SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V11I4P114

Volume 11 Issue 4, 133-148, April 2024
© 2024 Seventh Sense Research Group®

Original Article

Enhancing Question Answering with a Multidirectional Transformer: Insights from Squad 2.0

R. Rejimoan[1], B. Gnanapriya[2], J.S. Jayasudha[3]

[1,2]Department of Computer Science and Engineering, Annamalai University, Tamil Nadu, India
[3]Department of Computer Science, Central University of Kerala, Kerala, India.

[1]Corresponding Author : rejimoanr@outlook.com

Received: 15 February 2024 Revised: 16 March 2024 Accepted: 14 April 2024 Published: 30 April 2024

Abstract - *Natural Language Processing (NLP), a field at the intersection of linguistics and artificial intelligence, aims to equip machines with the ability to understand, interpret, and generate human-like text. Focused on the relevance of Machine Reading Comprehension (MRC), a vital subset of NLP, the proposed approach addresses the intricate task of training a model to understand and respond to questions based on a given context, mimicking human-like comprehension. Leveraging the Squad 2.0 dataset, a benchmark in MRC, the methodology employs a Multidirectional Transformer architecture coupled with BERT, a pre-trained language representation model, to enhance the model's ability to grasp contextual nuances. The tokenization process is utilized to break down raw text into smaller units, allowing for effective analysis. The architecture incorporates embedding techniques, sub-string search mechanisms, and data generators, fostering a comprehensive understanding of the input data. Employing masked softmax and permutation techniques during training contributes to the model's robustness, particularly in handling long-range dependencies and diverse expressions of the same information. The results obtained reveal a high accuracy of 94.00%, with an Exact Match of 48.4% and an F1 score of 60.9882%. Visual representations further affirm the model's prowess in comprehension, showcasing aligned predictions with actual answers. In essence, this paper presents a comprehensive approach to MRC within the NLP domain, employing advanced techniques and achieving promising results on the Squad 2.0 dataset.*

Keywords - *Machine Reading Comprehension, Question answering, Natural Language Processing, BERT, Squad, Embedding.*

1. Introduction

It's a challenging effort to teach machines to read and understand essential information from texts written in natural language. To extract meaningful information from natural language documents, conventional keyword searches, pattern matching, and enhanced mathematical and statistical techniques based on similarity were insufficient. These techniques provided the answers without a thorough comprehension of the relevant context based on the retrieval of a database.

Evaluating a system's capacity for language comprehension through reading comprehension is a logical approach to address this problem. Answering questions about a textual environment using appropriate linguistic understanding is the challenge of reading comprehension. Machine Reading Comprehension (MRC) is a type of reading comprehension model that helps the machine learn from context. Four decades ago, was the beginning of the extensive history of machine reading comprehension systems. The QUALM question-answering program developed by Wendy Grace Lehnert (1977) [1] was one of the most prominent investigations. The system comprehended two stories; however, because of the small-scale data and domain-specific approach, the QUALM could not be extensively used. The lack of large-scale, high-quality datasets has caused research on MRC to stagnate for the past 20 years.

Twenty years later, Lynette Hirschman et al. (1999) [2] made modest progress with MRC systems and provided a dataset with sixty test tales covering content from third to sixth grade. Five interrogative terms-What, Where, When, Why, and who-were included in the dataset. These words have the ability to draw out pertinent details from a context. In independent learning, question words can be very beneficial. They can be used to highlight paragraphs with key information at random. During that time, the majority of question-answering systems relied on statistical or rule-based techniques, which led to poor accuracy and subpar performance.

A rule-based MRC system called Quarc was introduced by Michael Thelen and Ellen Riloff in 2000 [3]. It makes use of heuristic rules. The narrative was read by the system, which then extracted the sentence that would have contained the answers to a variety of interrogative ("Wh") inquiries. Morphological analysis techniques, including Named Entity Recognition (NER) and POS (Part-of-Speech) tagging, were used to extract the replies. At 40% accuracy-the best accuracy of that era-the Quarc system was one of the noteworthy reading comprehension systems.

Hoifung Poon et al. (2010) [4] used the bootstrapping method to suggest a unified reading strategy. For the extraction of knowledge and long-tail conquest, self-supervised learning and Markov logic inference were integrated. The MC Test was first presented by Matthew Richardson et al. (2013) [5] and consists of 2000 questions and 500 tales. Many researchers focused on the MC Test dataset, the predecessor of the contemporary MRC datasets, and started using the generated models on it.

With the broad adoption of deep learning and the development of sophisticated models and superior NLP architectures since 2015, MRC has prospered. Hermann et al. (2015) [6] developed a Deep Neural Network model based on attention and offered a large-scale supervised MRC dataset. The model was able to read and understand papers written in natural language, and it could provide context-based answers for complicated queries.

At that time, MRC had a quick development due to the introduction of multiple models for reading comprehension and the appearance of difficult datasets. A typical MRC system receives as inputs the textual background and the question in natural language and outputs the response. The MRC system consists of four primary components, as shown in Figure 1. Since natural language is difficult for a machine to understand directly, the first step is to translate the language into a form that the machine can comprehend. Consequently, the embedding module makes it easier to translate natural language into word vectors, which are then supplied to the feature extraction module in order to extract the pertinent data. The information retrieved is communicated with the Context-Question Interaction module in order to determine the correlation between the question and context after further pertinent aspects from the context and question have been collected. Finally, fed to the Answer Prediction module, which outputs the answer.

The embedding module uses vectorization, also known as word embedding, to turn the input words from the question and context into fixed-length vectors. Syntactic and semantic information can be preserved when words are represented as vectors in a vector space [7]. The vectors can be classified as either sparse (based on frequency) or dense (based on prediction) based on how well the information is preserved. While sparse vectors are capable of capturing a limited amount of semantic material and utilizing vast amounts of memory, dense vectors are able to hold semantic data efficiently without requiring a significant amount of memory.

The focus of the feature extraction module is on extracting contextual information from context and question embeddings, which it receives as input in the form of vectors from the embedding module. Traditional rule-based and Machine Learning (ML) models are less effective at capturing contextual information than deep learning-based models.

Lastly, Transformers [8] is a superb advancement in deep learning that outperforms current models in a number of areas. BERT-based MRC is shown in Figure 2. Transformers outperform previous approaches and achieve previously unattainable results in NLP challenges.

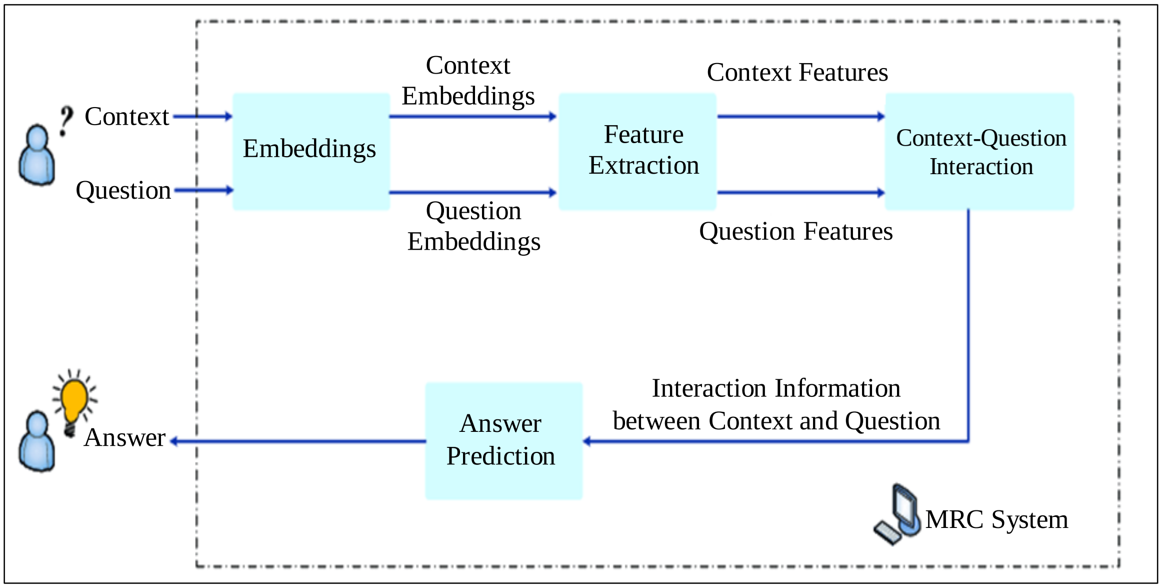


Fig. 1 Basic layout of MRC

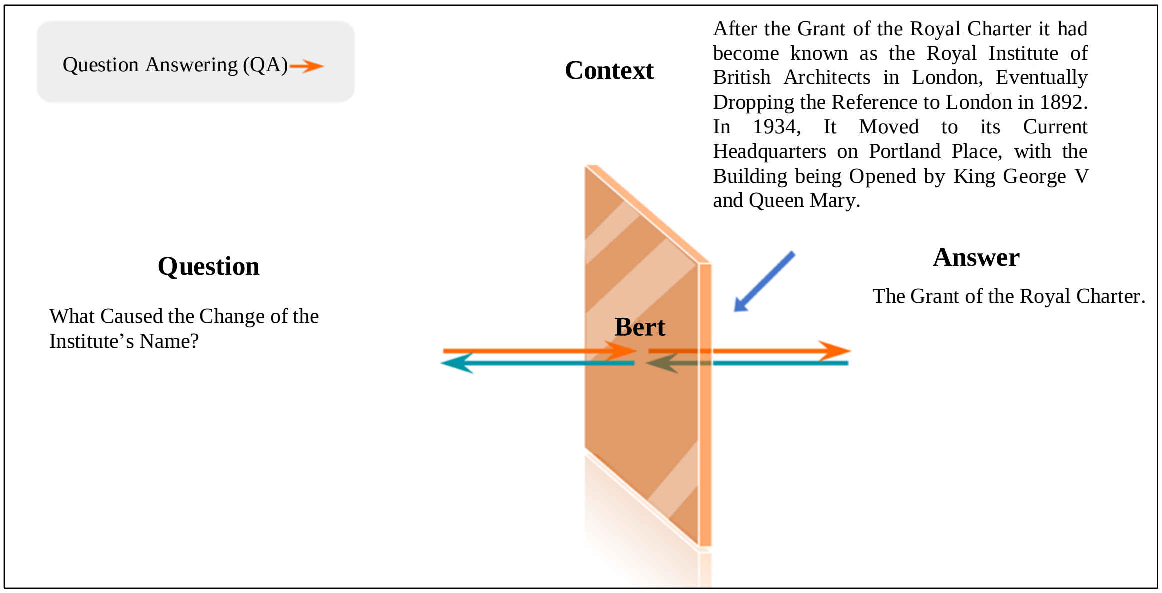

Question Answering (QA)
Context
After the Grant of the Royal Charter it had become known as the Royal Institute of British Architects in London, Eventually Dropping the Reference to London in 1892. In 1934, It Moved to its Current Headquarters on Portland Place, with the Building being Opened by King George V and Queen Mary.
Question
What Caused the Change of the Institute's Name?
Bert
Answer
The Grant of the Royal Charter.

Fig. 2 Bert-based MRC

One of the essential MRC modules, Context-Question Interaction, seeks to determine the relationship between the question and the context. In order to provide an accurate response, this module gathers the information from the preceding module and identifies the most pertinent sections. Answer prediction: the last module outputs the solution based on the data gathered from earlier modules. These four categories are used to group the final answers based on the type of response.

Reading Comprehension (RC) and Question Answering (QA) are related, and they both share some traits, including issue design, techniques, and evaluation. The creation of models that can respond to queries posed by individuals in natural language automatically is the focus of question answering systems. The ultimate objective of QA is to design and implement computer systems that are capable of automatically responding to user inquiries. Question-answering is one application of reading comprehension.

RC algorithms are able to extract the most accurate response from the relevant context or, depending on the situation, even produce a more sophisticated or adaptable response. While responding to inquiries, RC systems focus a great deal of attention on textual comprehension. Robust datasets, which are particularly created to evaluate various aspects of textual understanding, have contributed to the progress of reading comprehension.

With the aid of well-known comprehension models, reading comprehension can close the comprehension gap between humans and machines. As a result, RC systems perform better in dialogue and question-answering systems since they can access knowledge more quickly. The main contributions of the proposed paper are as follows:

- To implement a Deep Neural network based on a multidirectional Transformer for comprehension and question answering on the SQUAD 2.0 dataset to enhance the model's ability to capture contextual information from different directions.
- To explore how multidirectional attention mechanisms contribute to better comprehension and question-answering accuracy.
- To emphasize the importance of achieving higher levels of accuracy in question answering, indicating the model's proficiency in understanding and processing textual information.

The subsequent sections of this paper follow a structured framework. In section 2, the existing literature is examined, investigating relevant studies and insights within the proposed field. Section 3 provides a comprehensive outline of the methodology, detailing the approach taken to implement the Multidirectional Transformer on the SQUAD 2.0 dataset. Section 4 showcases the obtained results and initiates a discussion around them, shedding light on the performance and implications of the model. Finally, in section 5, conclusions are drawn based on the findings, encapsulating the key takeaways from this research endeavour.

2. Related Works

By addressing the difficulties in comprehending the questions themselves and the information sought, Zhang et al. [9] contribute to MRC. They present QA pairs, an additional question-answer matching task that is based on the SQUAD, QuAC, and COQA datasets. With a previous attention mechanism, the suggested PrA-MRC model integrates learnt question-type information and achieves an accuracy OF 84%. By incorporating previous knowledge into the BiDAF span-based model and using bi-attention flows for both query-aware and prior query-aware context representations, the method improves understanding.

The first large-scale machine-translated question-answering dataset for the Slovak language, SQUAD-sk, was created by Stas et al. [10]. Using the Helsinki-NLP Opus English-Slovak model with Marian neural machine translation, approximately 92% of the questions and answers from the original English SQUAD v2.0 were translated accurately. SQUAD-sk proves to be a useful tool for improving both monolingual and multilingual Q&A systems. One issue is that the existing dataset may not do as well at capturing finer details unique to the Slovak colloquial language.

The Persian Question Answering Dataset (PQuAD), a noteworthy addition with 80,000 questions drawn from Persian Wikipedia pages, is introduced by Darvishi et al. [11]. A noteworthy feature of the dataset is the purposeful design of 25% of the questions to be adversarially unanswerable. This new resource emphasizes diversity and varied difficulty levels while acting as a standard for Persian reading comprehension. Their goal is to spur improvements in the field of Persian reading comprehension research as well as the creation of Persian-language-specific question-answering systems. One drawback is that, although the dataset is vast and diverse, it does not fully reflect the subtleties of Persian language usage in conversational contexts, which could make it difficult for models to handle informal expressions.

The difficult task of Machine Reading Comprehension (MRC) with Unanswerable Questions is focused on by Yunjie Ji et al. [12]. They address the common mistake made by current models in identifying minute literal changes that turn an answerable question into an unanswerable one. They present an approach to contrast answered questions with their distorted and paraphrased equivalents at the answer span level: the span-based Contrastive Learning method (spanCL). With absolute EM increases ranging from 0.86 to 2.14 on the SQUAD 2.0 dataset, SpanCL considerably improves baselines by forcing MRC models to detect subtle semantic shifts. The research emphasizes how well spanCL uses produced questions and how effective it is regardless of the paradigm. They do recognize that the interoperability of question generation with spanCL may present a performance issue.

A multi-task fusion model based on BERT was presented by Ouyang et al. [13] for Machine Reading Comprehension (MRC). The model uses shared contextual representations from BERT for span extraction, yes/no question answering, and unanswerable questions. It handles extractive and non-extractive MRC tasks concurrently. Comprehensive training is made possible by the fused cross-entropy loss function and the fusing of sub-module outputs. Self-training improves model accuracy and generalization by producing pseudo-labeled data. Even with impressive results on SQUAD2.0 and CAIL2019 datasets, self-training is time-consuming due to its iterative nature. The applicability of the model to a wider range of MRC tasks is limited by its specificity.

Le Minh et al. [14] made a significant contribution to the field of natural language understanding by presenting UIT-VIQUAD 2.0, a benchmark dataset for Vietnamese Machine Reading Comprehension (MRC) that tackles the problem of unanswerable questions, a prevalent real-world situation. 77 teams participated in a competitive challenge at VLSP 2021 due to the dataset. This project promotes inquiry, generating, question-answering, and linguistic inference in Vietnamese MRC. Upcoming initiatives entail improving the performance of the MRC system by adding annotated questions. A potential limitation lies in need for cautious consideration in the augmentation of annotated questions for enhancing MRC system performance, as it may introduce biases or specific linguistic nuances that could impact the model's generalization.

Van Nguyen et al. [15] emphasize MRC while highlighting the changing field of natural language understanding. They offer UIT-VIQUAD 2.0, a benchmark dataset that encourages evaluation of MRC and question-answering systems for the Vietnamese language, in response to the shortcomings of current Vietnamese datasets that mostly concentrate on answerable questions. With 77 teams taking part, the Eighth Workshop on Vietnamese Language and Speech Processing had a substantial response. The dataset encourages researchers to dive into question-answering, creation, and natural language inference and serves as a catalyst for additional investigation in Vietnamese MRC and related tasks. One drawback, though, is that performance must be improved by adding both quantitative and qualitative annotations to annotated questions.

The machine reading comprehension model SSAG-Net, developed by Yu et al. [16], uses neural networks to integrate syntax and semantics. Their strategy enhances the model's capacity to exploit both MRC tasks by using distinct branches for syntax and semantics and explicit syntactic constraints, which set it apart from typical attention methods. Tested on SQUAD 2.0 and MC Test, SSAG-Net performed better on extractive and multiple-choice comprehension tasks than baseline models1. The research admits its shortcomings, discussing semantic framework links and offering possible improvements to the BERT model. To enhance overall model performance, it also falls short in terms of investigating sophisticated BERT variations and improving semantic analysis.

In order to solve the problem of Machine Reading Comprehension (MRC) for scholarly works, Saikh et al. [17] provide Science QA, a dataset of more than 100,000 context-question-answer triples that have been human-annotated. They use both basic and advanced models, such as SciBERT and SciBERT with Bi-DAF, and they get an amazing 75.46% F1 score. Tokenization disparities and sequence length constraints present difficulties that affect the accuracy of the model. Future objectives include adopting Generative Pre-trained Transformer (GPT) - 3 models, extending the task to full-text articles, improving Bi-DAF with multi-hop attention, increasing the size of the dataset, and investigating visual question answering. The research establishes a platform for future developments in managing complex details found in scientific literature and emphasizes the importance of MRC in information extraction from academic papers. One drawback is the uncertainty that tokenization variants introduce.

The lack of models and datasets for Machine Reading Comprehension (MRC) in the field of anti-terrorism is addressed by Gao et al. [18]. Using domain-related triples to improve semantic information, they present the Anti-Terrorism Domain Dataset (ATSMRC) and the KG-ATT-MRC model. On both the ATSMRC and cmrc2018 datasets, knowledge noise is reduced through mixed mutual attention, leading to notable gains in EM and F1 measures. In vertical domains with prominent data features and specialized terminology, the model performs better. Optimizing the model's effectiveness on fragmented characteristics in general domain datasets presents an issue.

2.1. Research Gap

Prior research has demonstrated notable progress in Machine Reading Comprehension (MRC) in a number of domains and languages. Models such as KG-ATT-MRC and SSAG-Net, as well as specific datasets like ATSMRC, SQUAD-sk, and PQUAD, have proven to be useful models. The reviewed models use novel techniques such as adversarially unanswerable questions, syntax-semantics integration, and knowledge-based attention mechanisms. However, a unified model that capitalizes on BERT's advantages is still required for improved MRC tasks. Even if they are quite effective, the current models tend to focus on certain languages or domains, and there isn't yet a systematic investigation into a deep multidirectional transformer that incorporates all of these different elements. The proposed work is based on this constraint. It seeks to solve the shortcomings of current models and datasets by combining these approaches into a cohesive architecture for better understanding and question-answering tasks.

3. Materials and Methods

The proposed framework focuses on advancing Natural Language Understanding through the design and development of a deep multidirectional transformer. The model leverages the power of BERT to provide contextualized embeddings. Dense layers and Masked Softmax contribute to the prediction process. The Permutation step introduces variability during training. The chosen dataset for evaluation is SQUAD 2.0, a benchmark in machine reading comprehension tasks. Performance metrics include accuracy, exact match, and F1 score. By combining these elements, the proposed paper aims to create a comprehensive and effective deep multidirectional transformer for superior performance in comprehension and question-answering tasks. The block diagram of the proposed model is shown in Figure 3.

3.1. Dataset

One of the best examples of a large-scale, labelled dataset for reading comprehension is the Stanford Question Answering Dataset (SQuAD). For the objective of this research, the more advanced SQuAD 2.0, which was introduced in mid-2018, is the emphasis.

SQuAD 2.0 is a popular MRC benchmark dataset that mixes over 50,000 new, unanswerable questions that are crowd-sourced and intentionally crafted to resemble answerable questions with the 100,000 questions from SQuAD 1.1 [19]. There are 43,000 unanswerable questions and 87,000 solvable questions in the training dataset. Question-answer combinations for an example passage from the SQUAD dataset are displayed in Figure 4. A portion of text from the passage appears in each of the response options.

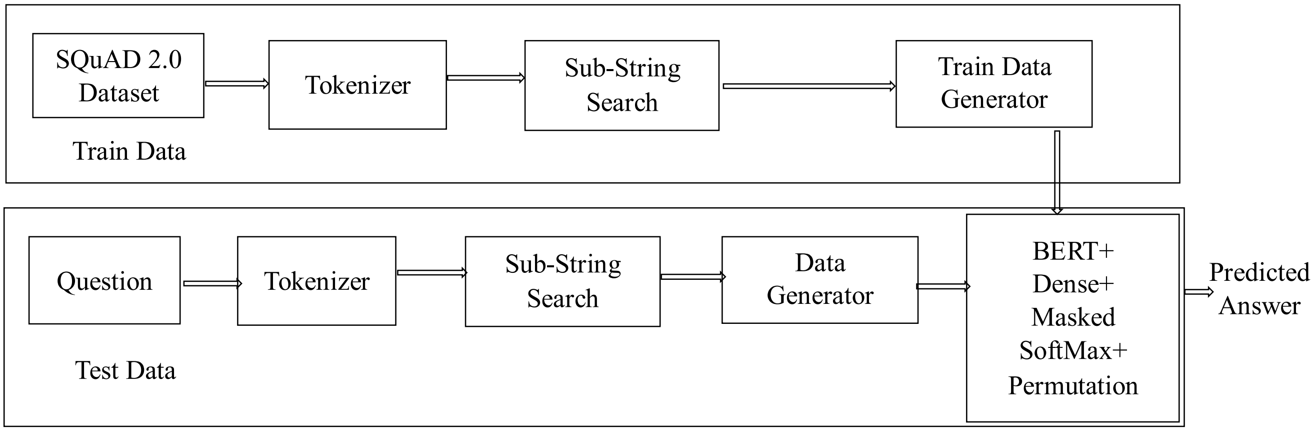
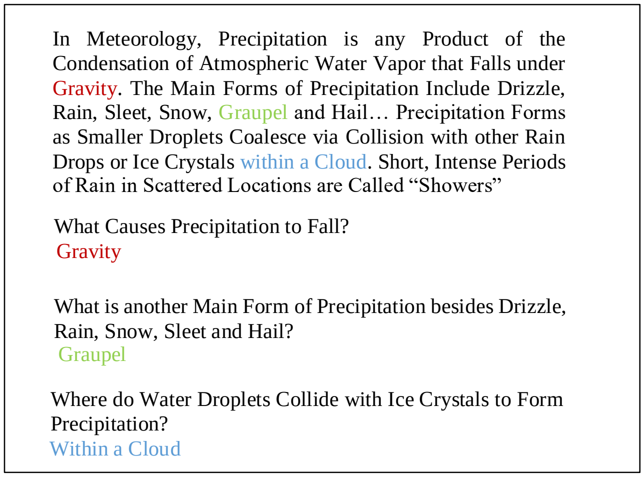


Fig. 3 Block diagram of the proposed methodology



In Meteorology, Precipitation is any Product of the Condensation of Atmospheric Water Vapor that Falls under Gravity. The Main Forms of Precipitation Include Drizzle, Rain, Sleet, Snow, Graupel and Hail… Precipitation Forms as Smaller Droplets Coalesce via Collision with other Rain Drops or Ice Crystals within a Cloud. Short, Intense Periods of Rain in Scattered Locations are Called "Showers"

What Causes Precipitation to Fall?
Gravity

What is another Main Form of Precipitation besides Drizzle, Rain, Snow, Sleet and Hail?
Graupel

Where do Water Droplets Collide with Ice Crystals to Form Precipitation?
Within a Cloud

Fig. 4 A portion of text from Dataset

Prior to the evolution of BERT by Devlin et al. [20], early models developed for SQuAD 2.0 significantly underperformed human-level performance. By depending significantly on BERT, the best contributions on SQuAD 2.0 have nearly attained human-level performance; as of March 2019, 19 out of the top 20 submissions on the leaderboard use BERT in some way. In this research, BERT's potential is thoroughly examined, making it a cornerstone of modern machine reading comprehension methods.

3.2. Tokenization

In the proposed model, the process of tokenizer plays a crucial role as one of the initial steps in the NLP pipeline. Tokenization involves breaking down the raw text into smaller units, known as tokens. This can be done at the word level, referred to as 'Word Tokenization,' or at the sentence level, known as 'Sentence Tokenization.' In the proposed paper, where comprehension and question answering are the focus, word tokenization is employed. For 'Word Tokenization,' the text is typically split based on spaces between words, as shown in Figure 5.

This method is effective for tasks where understanding the individual words is crucial, such as in comprehension and question-answering tasks. During the tokenization process, certain characters, such as spaces and punctuation, are ignored to ensure that they do not become part of the final list of tokens. The significance of tokenization lies in the fact that the meaning of a sentence is derived from the words it contains.

Analyzing these words allows us to interpret the overall meaning of the text. Once a list of words is obtained through tokenization, various statistical tools and methods can be applied to gain deeper insights. For instance, word count and word frequency analysis are employed to determine the importance of specific words within a sentence or document.

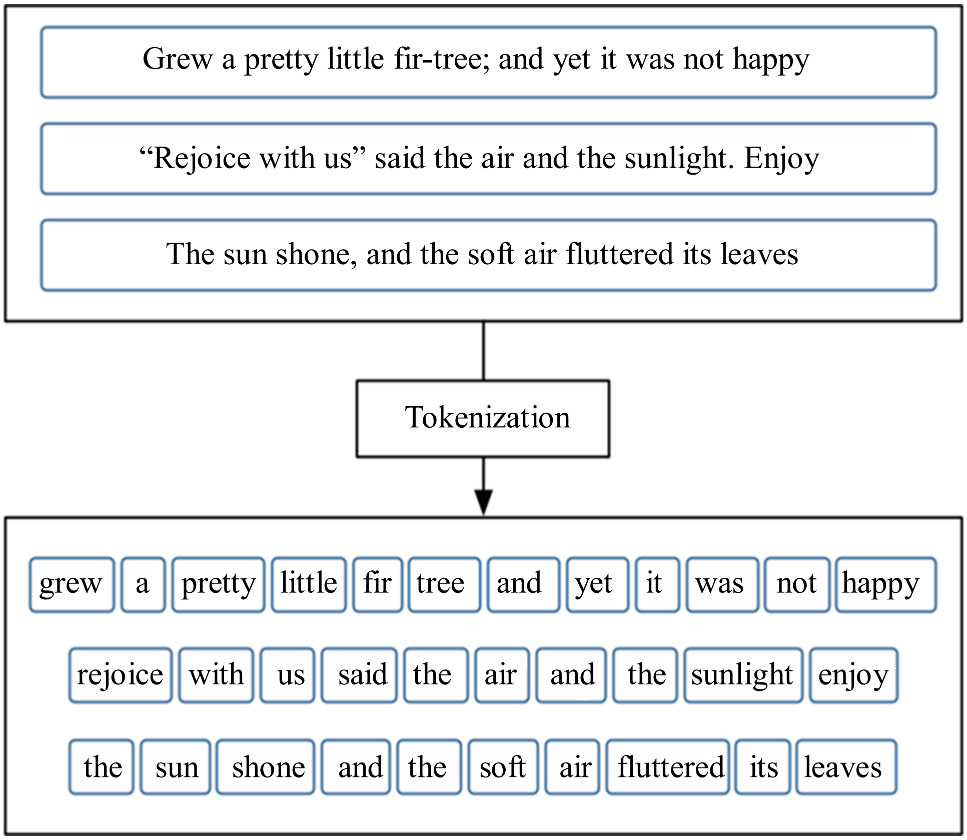

Grew a pretty little fir-tree; and yet it was not happy
"Rejoice with us" said the air and the sunlight. Enjoy
The sun shone, and the soft air fluttered its leaves
Tokenization
grew a pretty little fir tree and yet it was not happy
rejoice with us said the air and the sunlight enjoy
the sun shone and the soft air fluttered its leaves

Fig. 5 Word tokenization

3.3. Sub-String Search

A substring is an entire string contained inside another string or a contiguous portion of a string. The process involves identifying the patterns and sequences. An example of sub-string is shown in Figure 6. The substring search phase identifies pertinent data in the questions and sections of tokenized context.

To complete the comprehension and question-answering tasks it entails locating particular substrings or sentences that contain important information. This is accomplished using a variety of methods, including regular expressions, precise matching, and more advanced algorithms that look for significant patterns in the text. The aim is to identify pertinent sections that enhance comprehension of the situation and aid in producing precise responses. As an important pre-processing phase, the substring search design helps extract relevant data that the model's later stages can use properly.

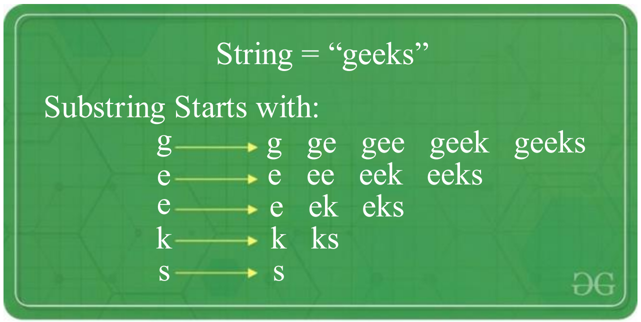


Fig. 6 An example of a substring

3.4. Data Generator

Using the tokenized and substring-searched data, the data generator creates organized training and testing samples. To enable efficient model training during the data generation phase, machine-readable language is transformed from its raw form. The goal of this component is to prepare the input data so that the model may be trained efficiently and tested with correct evaluation. To help the model learn the relationships between contextual information and appropriate answers, the data generator creates input-output pairs for the training set by pairing tokenized context passages with related questions and their proper answers. In order to prepare input samples for testing, the data generator associates tokenized questions with any pertinent substrings or context found using the substring search. The model uses these prepared samples as input to predict answers.

To ensure that the model is trained and tested on well-structured and meaningful data, the data generator's architecture manages the information flow from tokenization and substring search to the construction of comprehensible input-output pairs. Padding is also done to ensure compatibility inside a neural network architecture and prepare the data for training. In order to ensure that every sequence in a batch has the same length, padding is the process of adding zeros or special tokens to sequences. Contiguous data batches must be created using this step in order to facilitate effective parallel processing during model training. Sequences are either padded or truncated to a specified length because neural networks work with fixed-size inputs, guaranteeing consistency throughout the dataset.

3.5. Proposed Model

The model architecture for the proposed approach combines several components to create a comprehensive system for comprehension and question-answering tasks. The overall process is shown in Figure 7.

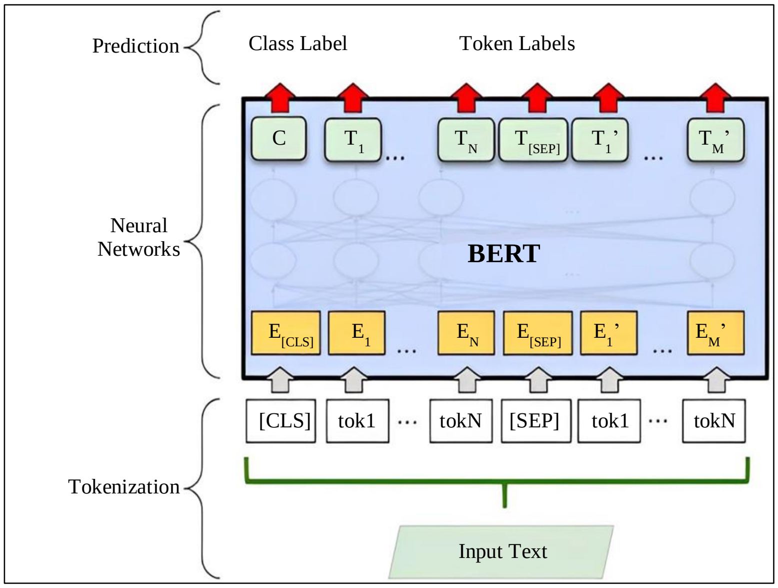


Fig. 7 Model architecture for the proposed methodology

3.5.1. BERT

In natural language issues, text representation plays a crucial role in representing texts that are machine-understandable. There are various difficulties in the text representation process. Delivering text representation involves major hurdles, two of which are semantic and syntactic [21]. The best approach available right now for handling text representation is BERT (Bi-directional Encoder Representations from Transformer). BERT is a pre-trained language representation model based on the Transformer architecture.

The model is trained on two tasks, namely the Next Sentence Prediction (NSP) and Masked language Modelling (MLM) methods, to achieve this Outcome. A random word in a phrase is hidden using the MLM method, and the model learns the context by estimating the masked text based on the surrounding masked word [22]. In order to execute binarized next sentence prediction for the NSP method, two phrases are introduced. Whether the second of two sentences is the pair of the first sentence or not will be ascertained by the NSP mechanism. The pre-trained model may be able to retain the connections between the texts with the aid of this learning process.

Token embeddings, segment embeddings, and position embeddings make up the components of the BERT input representation, as illustrated in Figure 8. Word tokens acquired via the Word Piece approach make up the token embeddings in the BERT architecture; the token embeddings always begin with the (CLS) token and end with the (SEP) token. Every token has an embedding vector associated with it, including special tokens and sub-word units acquired through tokenization. The basis for BERT's comprehension of the input is its token embeddings, which capture the semantic meaning of individual tokens. Next, each token that belongs to a given sentence is identified by its segment embedding. BERT is made to handle tasks like answering questions that require several text segments. BERT uses segment embeddings to distinguish between these segments.

Each token has a segment embedding attached to it that specifies the phrase or segment it belongs to. BERT needs to take into account each token's position because it processes the full input sequence at once. To give information about a token's position in the sequence, position embeddings are included in its embedding. This aids BERT by maintaining the input's word order intact. These three embeddings are often mathematically summed or concatenated to produce a single representation for every token in the input sequence.

NLP uses the Transformer design [8], which successfully handles long-range relationships, to offer a novel method for sequence-to-sequence problems. Relying on self-attention techniques to compute representations for both input and output sequences is a fundamental characteristic of the Transformer. With the help of self-attention, the model can predict the output sequence while concentrating on distinct segments of the input sequence, which helps it to identify complex dependencies and relationships in the data.

The Encoder Attention within the Transformer, as shown in Figure 9, involves establishing connections between the input and output sequences. The Encoder block, a fundamental component, consists of three types of residual sub-layers, incorporating the addition of positional embeddings. Positional embeddings are essential for providing information about the relative or absolute position of tokens in the sequence, helping the model understand the sequential order of the input.

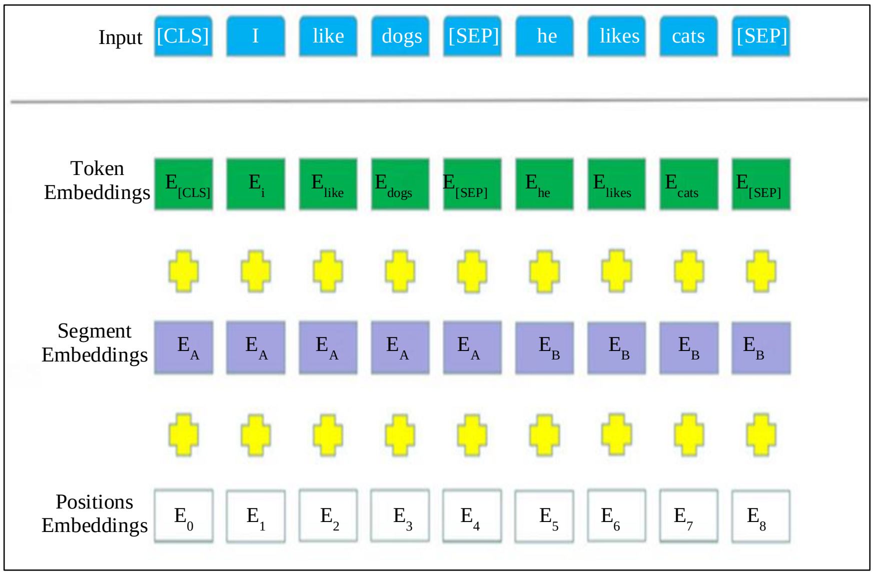


Fig. 8 Embedding of BERT

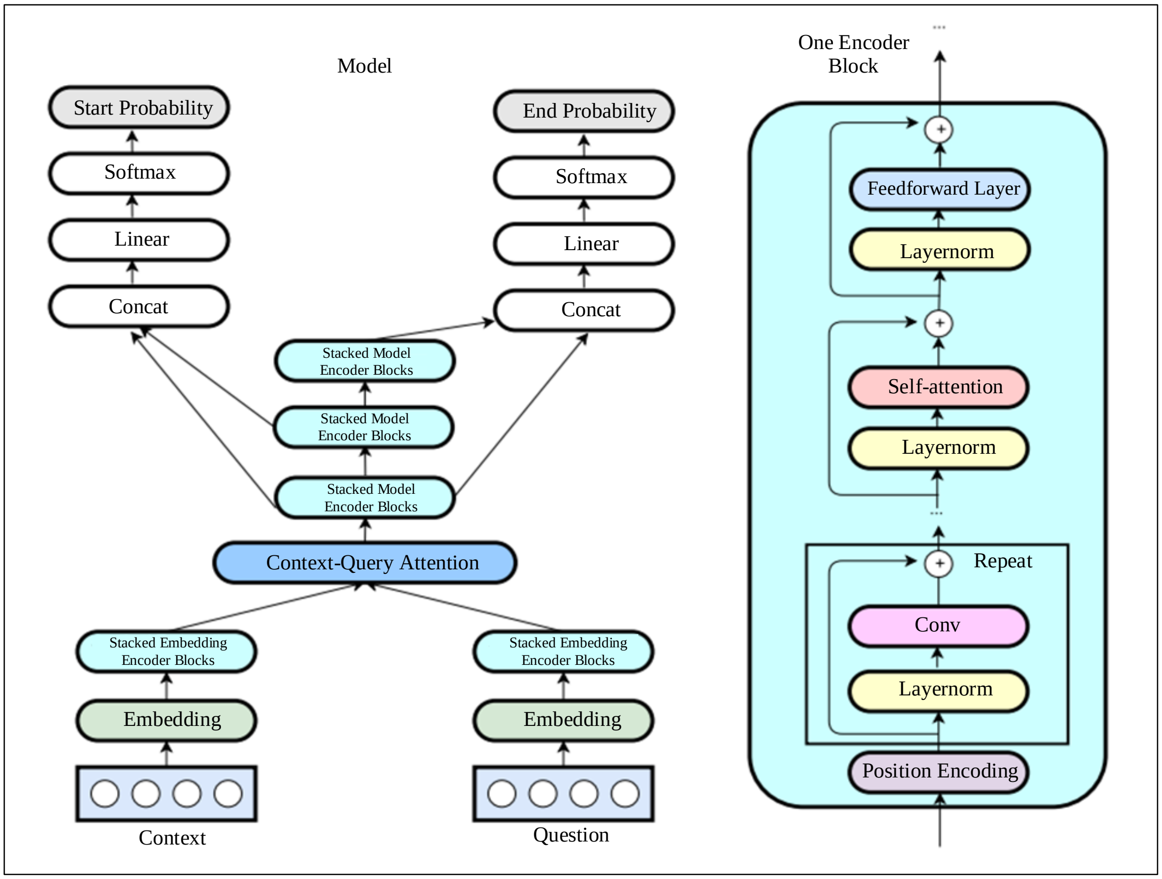


Fig. 9 Proposed QA architecture using BERT

Each operation in the encoder block, which consists of one encoder block with four convolution layers and a kernel size of 7, is placed inside a residual block, the output of which is defined by Equation 1.

$$f\big(layernorm(x)\big) + x \qquad (1)$$

The Context-Query Attention Layer calculates the attention output as well as the context-to-query and query-to-context attention distributions after calculating pairwise similarities between context and query words. More specifically, this layer performs the following calculations:

Context-to-Query Attention: With C denotes the encoded context and Q for the encoded question, a pairwise similarities matrix S is calculated. Utilizing a tri-linear function [9] as described in Equation 2, these similarities are calculated.

$$f(q, c) = W_0[q, c, q \odot c] \qquad (2)$$

Then, using softmax row normalization, $(softmax_{row})$ the context-to-query attention is expressed as in Equation 3.

$$A = softmax_{row}(S)Q^T \qquad (3)$$

Query-to-Context Attention: It is then calculated as in Equation 4, wherein $softmax_{col}$ denotes softmax column normalization.

$$S = \; softmax_{col}(S)C^T \qquad (4)$$

Attention Output: The output of the layer is calculated as in Equation 5.

$$[C, A, C \odot A, C \odot B] \qquad (5)$$

The embedding encoder layer and the model encoder are constructed from the same encoder blocks. The outputs M_0, M_1, and M_2 of the three stacked encoder blocks, respectively, each contributes to a distinct output layer. Lastly, the output layer uses the first two blocks of the model encoder layer for p_{start} and the first and third blocks for p_{end} to calculate the likelihood that each point in the context is either the start $(p_{start}(i))$ or the end $(p_{end}(i))$ of the answer span. The probabilities are computed as in Equation 6 and 7.

$$p_{start} = softmax(W_1[M_0, M_1]) \qquad (6)$$

$$p_{end} = softmax(W_2[M_0, M_2]) \qquad (7)$$

3.5.2. Dense Layer

After BERT's initial contextualization, the dense layer offers more processing and feature extraction capabilities. A dense layer is a kind of fully connected layer in which every neuron in its preceding layer is intricately related to every other neuron.

The model can identify complicated patterns and representations within the data owing to this connection pattern, which makes it possible to capture complex relationships between the features collected by the previous layers. Every neuron in the layer above feeds information into the neurons of a dense layer; these connections are encoded as weights. The weights are changed when the model is being trained in order to maximize its effectiveness for the given task.

A matrix-vector multiplication technique is used to determine each neuron's output in the dense layer. For the multiplication to be valid, the row vector from the previous layer needs to match the column vector of the weight matrix. Therefore, the number of neurons in the layer above and the required number of neurons in the dense layer dictate the weight matrix's size. By converting the input data into a new set of features, the dense layer is able to improve the model's capacity for accurate prediction in subsequent tasks by giving it a richer representation.

3.5.3. Masked Softmax

A mathematical function called the softmax operation normalizes scores to make them positive and guarantees that their sum equals 1. Tokens in MRC are categorized using softmax, which also provides probabilities for each token's relevance or importance in producing an answer. The mathematical expression for softmax is as in Equation 8.

$$softmax_i(x) = \frac{\exp(x_i)}{\sum_j exp(x_j)} \qquad (8)$$

Wherein, x_i is the attention score for position i, x_j is the attention score for position j in the sequence, and the sum is taken over all positions in the sequence.

The model's self-attention mechanisms allow the decoder to take into account data at various points in the input stream. But if a masking mechanism wasn't included, the decoder would unintentionally peek at future positions, which would go against the causality principle. A masking method is provided prior to the softmax operation in order to overcome this problem. One important stage in the training process is to introduce a mask prior to the softmax procedure to address the issue of "looking into the future". In order to make sure the model learns to predict answers sequentially and causally, masking is used to hide some answers during training. The model is discouraged from producing responses based on any information accessible up to that moment by hiding some answers, which keeps it from learning future positions while it is being trained. This masking technique is essential to preserving the integrity of the training process and enhancing the model's successful generalization to previously unknown data.

The masked softmax operation is expressed mathematically as in Equation 9.

$$Masked\ softmax_i(x) = \frac{\exp(x_i) Mask_i}{\sum_j exp(x_j).Mask_j} \qquad (9)$$

Wherein, $Mask_i$ is a binary mask applied to the attention scores. If a position is masked, its corresponding value in the mask is set to 0, effectively zeroing out the contribution of that position during the softmax computation. This ensures that the softmax operation only considers positions that are relevant up to the current position, preventing information leakage from future positions.

3.5.4. Permutation

Following the masked softmax operation, the model produces probability distributions across possible sequence answer positions. It is ensured by the softmax operation that the predicted responses are generated from pertinent positions in the input context. Still, the model may not be able to cope with multiple answer expressions or ways to phrase the right response if it only uses the deterministic output of the softmax. Exposing the model to different permutations of the same data is the intent of incorporating permutation.

The primary objective of permutation is to enhance the training data by rearranging or shuffling the predicted answers. This shuffling introduces diversity in the predicted output, encouraging the model to be less sensitive to the specific order of its predictions. For instance, if the masked softmax outputs a probability distribution $[P_1, P_2, \dots, P_n]$ for potential answer positions, the permutation step could shuffle this distribution to create variations like $[P_3, P_1, \dots, P_2]$. This is crucial in MRC tasks where the same information can be expressed in multiple ways, and the model needs to learn to identify the correct answer regardless of the specific order or arrangement of the predicted candidates.

The permutation step adds a layer of complexity to the training process, encouraging the model to focus on the intrinsic features and relationships between tokens rather than memorizing specific sequences. This is particularly important in the proposed paper, where the model needs to comprehend and answer questions based on the context, irrespective of the specific ordering of potential answers.

4. Results and Discussion

4.1. Hardware and Software Setup

The computational setup for this research utilized a machine with robust specifications, featuring an Intel Core i7 processor, 32GB of RAM, and the formidable NVIDIA GeForce GTX 1080Ti GPU. Model implementation was seamlessly carried out through the Keras library, functioning as a prototype built upon the Tensorflow framework and executed using the versatile Python language. Keras, known for its user-friendly interface and powerful capabilities, proved instrumental in crafting intricate Neural Network architectures. This framework ensures efficient utilization of computing resources, seamlessly accommodating CPU, GPU, and TPU environments.

To leverage extensive computational capabilities and streamline model training, the deployment was orchestrated on Google Colab. This cloud-based Python notebook environment not only provides complimentary access to robust computational resources but also facilitates collaborative development, making it an optimal choice for training models.

Hyperparameters are essential configuration settings that define the behaviour and characteristics of a machine learning framework throughout the training process. Unlike the parameters of the model, which are learned from the data itself, hyperparameters are set by the user before training begins. The neural network model uses the Adam optimizer with a 0.00002 learning rate and 108,893,186 trainable parameters. The sparse categorical cross-entropy loss function guides the training process.

During training, the model processes input data in batches of 4 samples per iteration. The training is carried out over 5 epochs, signifying the number of times the model processes the entire training dataset. These hyperparameter choices, such as the optimizer, learning rate, loss function, batch size, and number of epochs, collectively define the configuration for training the neural network model, aiming to optimize its performance on the given MCR task. The model configuration of the suggested approach is tabulated in Table 1.

Table 1. Model configurations

Model Parameters	Values
Trainable Parameters	108,893,186
Optimizer	Adam
Maximum Sentence Length	512
Learning Rate	0.00002
Epochs	5
Batch Size	4
Loss Function	Sparse Categorical Cross-Entropy

4.2. Performance Evaluation

The proposed model employs several key performance evaluation metrics to assess the effectiveness of the deep multidirectional transformer model in comprehension and question-answering tasks. These metrics provide a comprehensive understanding of the model's performance across different aspects of its predictions. The primary evaluation metrics include:

4.2.1. Accuracy

The percentage of questions that a system correctly answers is called its accuracy. Accuracy can be mathematically modelled as in Equation 10.

$$Accuracy = \frac{M}{N} \tag{10}$$

Where N be the total number of questions in the assessment dataset, and M be the number of questions a system successfully answered.

4.2.2. Exact Match

The system-generated response may contain some words that are correct replies while leaving other words incorrect, especially if the question requires a sentence or phrase as its response. The percentage of questions for which the answer provided by the system matches the right response exactly in this instance-that is, in terms of word for word is known as the Exact Match. An exact match can be mathematically modeled as in Equation 11.

In the event that an MRC task comprises N questions, the system will properly answer M of the questions; each question has a single valid answer, which may consist of a word, phrase, or sentence. Though they might not precisely match the ground truth answer, some of the remaining N − M solutions might contain some ground truth answer terms.

$$Exact\ Match = \frac{M}{N} \tag{11}$$

Furthermore, it is typical to gather more than one accurate response for every question in order to increase the evaluation's dependability. Therefore, the exact match score is only required to match any of the correct answers.

4.2.3. F1-score

The harmonic mean of recall and precision is the F1 score, as depicted in Equation 12. The ratio of the number of tokens in a prediction that overlaps with the right response to the total number of tokens in the prediction is known as the precision in textual QA. The recall calculates the ratio of all the tokens in a correct response that has been predicted to all the tokens in the correct answer.

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{12}$$

The greatest F1 score is obtained when a question contains multiple reference answers. The overall F1 score of the system is the average of all the predictions.

In NLP, tasks often deal with predicting the next word from a vocabulary that consists of thousands of classes. This results in a situation where the true predictions are represented by a large matrix with the majority of its elements being zeros, indicating the absence of the predicted class. Sparse categorical accuracy is a specialized metric designed to handle such sparse target scenarios where the task involves predicting from a multitude of classes. The sparse matrix representation efficiently handles the vast vocabulary and reduces computational overhead. The sparse accuracy plot provides insights into how well the model is able to predict the correct classes over the course of training. Across the 5 epochs, a steady increase in accuracy, as shown in Figure 10, suggests that the model is learning and adapting to the complexities of the Squad 2.0 dataset, successfully capturing the relationships within the input sequences and making accurate predictions.

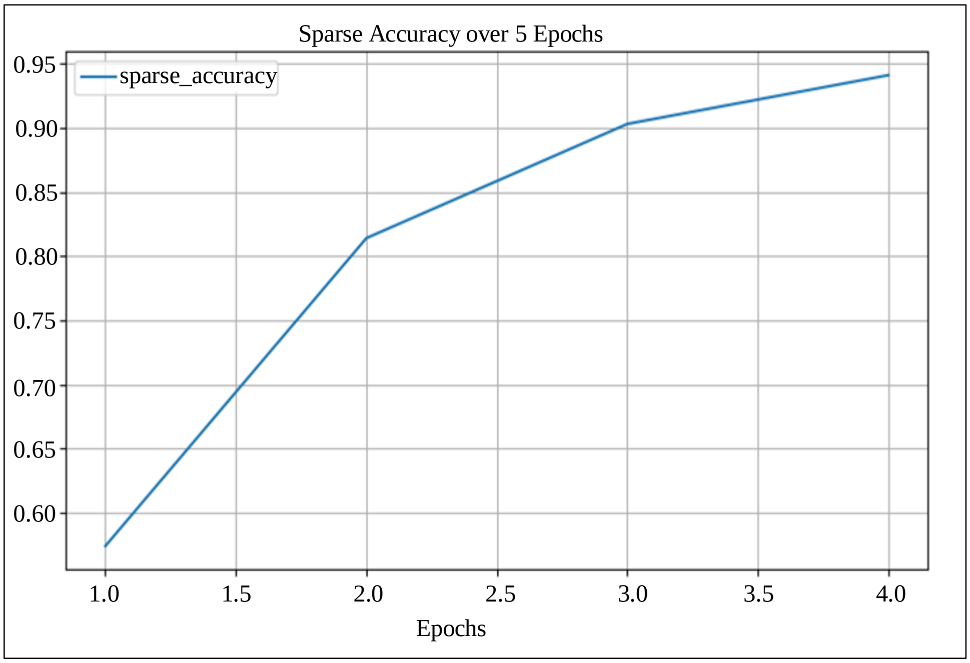


Fig. 10 Accuracy plot of the proposed model

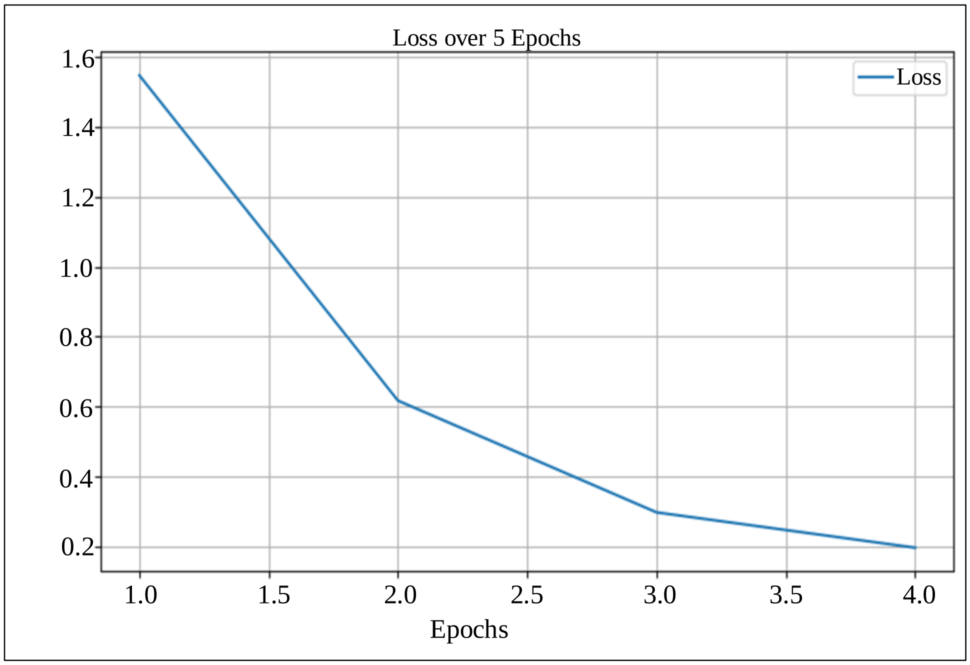


Fig. 11 Loss plot of the proposed model

The loss function serves as a measure of how well the model is performing in terms of minimizing prediction errors. As the model iteratively updates its parameters during training, the loss is continuously minimized, as shown in Figure 11, guiding the network towards more accurate predictions.

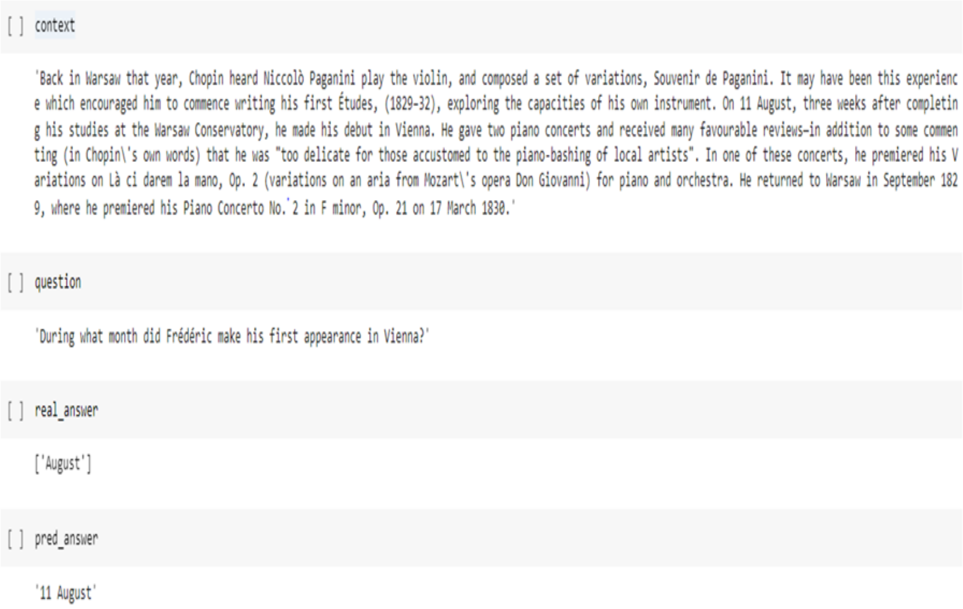
Table 2 shows the performance metrics obtained for the proposed methodology. The outcomes of the suggested paper demonstrate an excellent degree of performance in relation to certain evaluation requirements. The model's capacity to produce accurate predictions is demonstrated by its accuracy of 94.00%, which also shows that it performed well overall on comprehension and question-answering challenges using the Squad 2.0 dataset.

With an Exact Match score of 48.4%, the model demonstrates its accuracy in capturing the subtleties of the input context and demonstrates its ability to provide answers that precisely match the ground truth. The model's ability to achieve a fair equilibrium between accurate positive predictions and avoiding false positives is demonstrated by the remarkable 60.9882% F1 score. The prediction results visualization is shown in Figures 12(a)-12(d), demonstrating the model's ability to perform comprehension and question-answering tasks. The context is the given comprehension, and the model skilfully produces precise answers to the related queries.

The remarkable degree of agreement between the actual and predicted responses suggests that the model was able to comprehend the input data with a high degree of precision. This alignment indicates that the suggested methodology is able to capture every aspect of the provided setting in an effective manner, allowing the model to produce results that are almost identical to the real thing. These promising visual representation results confirm the robustness of the suggested methodology and demonstrate its capacity to produce precise results in the context of challenging natural language processing tasks.

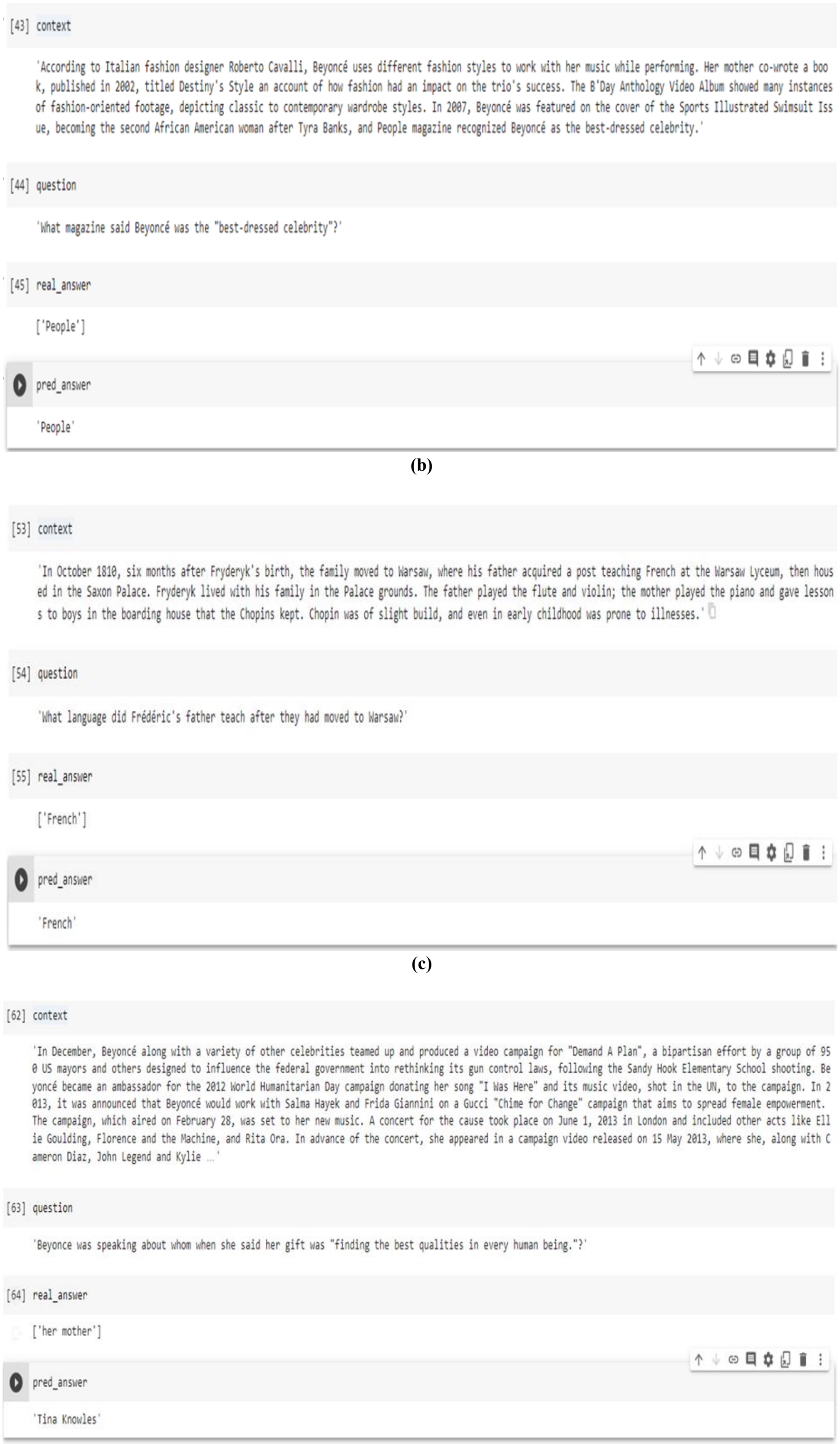
Table 2. Performance metrics

Performance Metrics	Results Obtained (%)
Accuracy	94.00
Exact Match	48.4
F1	60.9882



```
[ ] context

    'Back in Warsaw that year, Chopin heard Niccolò Paganini play the violin, and composed a set of variations, Souvenir de Paganini. It may have been this experienc
    e which encouraged him to commence writing his first Études, (1829-32), exploring the capacities of his own instrument. On 11 August, three weeks after completin
    g his studies at the Warsaw Conservatory, he made his debut in Vienna. He gave two piano concerts and received many favourable reviews–in addition to some commen
    ting (in Chopin\'s own words) that he was "too delicate for those accustomed to the piano-bashing of local artists". In one of these concerts, he premiered his V
    ariations on Là ci darem la mano, Op. 2 (variations on an aria from Mozart\'s opera Don Giovanni) for piano and orchestra. He returned to Warsaw in September 182
    9, where he premiered his Piano Concerto No. 2 in F minor, Op. 21 on 17 March 1830.'

[ ] question

    'During what month did Frédéric make his first appearance in Vienna?'

[ ] real_answer

    ['August']

[ ] pred_answer

    '11 August'
```

(a)



```
[43] context

'According to Italian fashion designer Roberto Cavalli, Beyoncé uses different fashion styles to work with her music while performing. Her mother co-wrote a boo
k, published in 2002, titled Destiny's Style an account of how fashion had an impact on the trio's success. The B'Day Anthology Video Album showed many instances
of fashion-oriented footage, depicting classic to contemporary wardrobe styles. In 2007, Beyoncé was featured on the cover of the Sports Illustrated Swimsuit Iss
ue, becoming the second African American woman after Tyra Banks, and People magazine recognized Beyoncé as the best-dressed celebrity.'

[44] question

'What magazine said Beyoncé was the "best-dressed celebrity"?'

[45] real_answer

['People']

pred_answer

'People'
```

(b)

```
[53] context

'In October 1810, six months after Fryderyk's birth, the family moved to Warsaw, where his father acquired a post teaching French at the Warsaw Lyceum, then hous
ed in the Saxon Palace. Fryderyk lived with his family in the Palace grounds. The father played the flute and violin; the mother played the piano and gave lesson
s to boys in the boarding house that the Chopins kept. Chopin was of slight build, and even in early childhood was prone to illnesses.'

[54] question

'What language did Frédéric's father teach after they had moved to Warsaw?'

[55] real_answer

['French']

pred_answer

'French'
```

(c)

```
[62] context

'In December, Beyoncé along with a variety of other celebrities teamed up and produced a video campaign for "Demand A Plan", a bipartisan effort by a group of 95
0 US mayors and others designed to influence the federal government into rethinking its gun control laws, following the Sandy Hook Elementary School shooting. Be
yoncé became an ambassador for the 2012 World Humanitarian Day campaign donating her song "I Was Here" and its music video, shot in the UN, to the campaign. In 2
013, it was announced that Beyoncé would work with Salma Hayek and Frida Giannini on a Gucci "Chime for Change" campaign that aims to spread female empowerment.
The campaign, which aired on February 28, was set to her new music. A concert for the cause took place on June 1, 2013 in London and included other acts like Ell
ie Goulding, Florence and the Machine, and Rita Ora. In advance of the concert, she appeared in a campaign video released on 15 May 2013, where she, along with C
ameron Diaz, John Legend and Kylie …'

[63] question

'Beyonce was speaking about whom when she said her gift was "finding the best qualities in every human being."?'

[64] real_answer

['her mother']

pred_answer

'Tina Knowles'
```

(d)

Fig. 12 Prediction results

5. Conclusion

The proposed paper presents a comprehensive exploration into the realm of natural language processing, with a specific focus on advancing MRC. Leveraging a Multidirectional Transformer architecture integrated with BERT, the model addresses the complex task of comprehending textual information and generating accurate responses to associated questions. The utilization of the Squad 2.0 dataset facilitates a rigorous evaluation, demonstrating the model's robust performance over five training epochs. The obtained results, characterized by a high accuracy of 94.00%, an exact match of 48.4%, and a commendable F1 score of 60.9882%, signify the effectiveness of the proposed methodology. The visualization of prediction results further reinforces the model's ability to provide precise answers, showcasing its proficiency in handling diverse comprehension passages. This research contributes valuable insights to the field of NLP, highlighting the efficacy of a Multidirectional Transformer architecture for enhancing machine comprehension. The demonstrated success of the model opens avenues for improved natural language understanding in various applications, from question-answering systems to advanced language processing tasks. As technology advances, the findings of this paper pave the way for continued refinement and innovation in NLP, with implications for more sophisticated language models and applications in the broader landscape of artificial intelligence.

Acknowledgements

The author expresses profound appreciation to the supervisor for providing guidance and unwavering support throughout the course of this research.

References

[1] Wendy G. Lehnert, *The Process of Question Answering, A Computer Simulation of Cognition*, 1st ed., Routledge, 1978. [CrossRef] [Google Scholar] [Publisher Link]

[2] Lynette Hirschman et al., "Deep Read: A Reading Comprehension System," *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pp. 325-332, 1999. [CrossRef] [Google Scholar] [Publisher Link]

[3] Ellen Riloff, and Michael Thelen, "A Rule-Based Question Answering System for Reading Comprehension Tests," *Proceedings of the 2000 ANLP/NAACL Workshop on Reading Comprehension Tests as Evaluation for Computer-Based Language Understanding*, vol. 6, pp. 13-19, 2000. [CrossRef] [Google Scholar] [Publisher Link]

[4] Hoifung Poon et al., "Machine Reading at the University of Washington," *Proceedings of the NAACL HLT2010 First International Workshop on Formalisms and Methodology for Learning by Reading*, pp. 87-95, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[5] Matthew Richardson, Christopher J.C. Burges, and Erin Renshaw, "Mctest: A Challenge Dataset for the Open-Domain Machine Comprehension of Text," *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 193-203, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[6] Karl Moritz Hermann et al., "Teaching Machines to Read and Comprehend," *Advances in Neural Information Processing Systems 28(NIPS 2015)*, 2015. [Google Scholar] [Publisher Link]

[7] Nisha Varghese, and M. Punithavalli, "Lexical and Semantic Analysis of Sacred Texts Using Machine Learning and Natural Language Processing," *International Journal of Scientific & Technology Research*, vol. 8, no. 12, pp. 3133-3140, 2019. [Google Scholar] [Publisher Link]

[8] Ashish Vaswani et al., "Attention is All You Need," *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, 2017. [Google Scholar] [Publisher Link]

[9] Yo Zhang, Bo Shen, and Xing Cao, "Learn A Prior Question-Aware Feature for Machine Reading Comprehension," *Frontiers in Physics*, vol. 10, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] Ján Staš, Daniel Hládek, and Tomáš Koctúr, "Slovak Question Answering Dataset Based on the Machine Translation of the Squad V2.0," *Journal of Linguistics/Jazykovedný Casopis*, vol. 74, no. 1, pp. 381-390, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] Kasra Darvishi et al., "PQuAD: A Persian Question Answering Dataset," *Computer Speech & Language*, vol. 80, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[12] Yunjie Ji et al., "To Answer or Not To Answer? Improving Machine Reading Comprehension Model with Span-Based Contrastive Learning," *arXiv Computation and Language*, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] Jianquan Ouyang, and Mengen Fu, "Improving Machine Reading Comprehension with Multi-Task Learning and Self-Training," *Mathematics*, vol. 10, no. 3, pp. 1-11, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] Dang Van Nhan, and Nguyen Le Minh, "ViMRC-VLSP 2021: Using XLM-RoBERTa and Filter Output for Vietnamese Machine Reading Comprehension," *VNU Journal of Science: Computer Science and Communication Engineering*, vol. 39, no. 2, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[15] Nguyen Van Kiet et al., "VLSP 2021-ViMRC Challenge: Vietnamese Machine Reading Comprehension," *VNU Journal of Science: Computer Science and Communication Engineering*, vol. 8, no. 2, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] Chenxi Yu, and Xin Li, "SSAG-Net: Syntactic and Semantic Attention-Guided Machine Reading Comprehension," *Intelligent Automation & Soft Computing*, vol. 34, no. 3, pp. 2023-2024, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] Tanik Saikh et al., "ScienceQA: A Novel Resource for Question Answering on Scholarly Articles," *International Journal on Digital Libraries*, vol. 23, no. 3, pp. 289-301, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] Feng Gao et al., "Knowledge Graph Based Mutual Attention for Machine Reading Comprehension over Anti-Terrorism Corpus," *Data Intelligence*, vol. 5, no. 3, pp. 685-706, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[19] Pranav Rajpurkar et al., "SQuAD: 100,000+ Questions for Machine Comprehension of Text," *Proceeding of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas, pp. 2383-2392, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[20] Jacob Devlin et al., "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding," *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4171-4186, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[21] Mohammadreza Samadi, Maryam Mousavian, and Saeedeh Momtazi, "Deep Contextualized Text Representation and Learning for Fake News Detection," *Information Processing & Management*, vol. 58, no. 6, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[22] Francisca Adoma Acheampong, Henry Nunoo-Mensah, and Wenyu Chen, "Transformer Models for Text-Based Emotion Detection: A Review of BERT-Based Approaches," *Artificial Intelligence Review*, vol. 54, pp. 5789-5829, 2021. [CrossRef] [Google Scholar] [Publisher Link]