Machine Learning Model for Automated Assessment of Short Subjective Answers

Zaira Hassan Amur[1], Yew Kwang Hooi[2], Hina Bhanbro[3], Mairaj Nabi Bhatti[4], Gul Muhammad Soomro[5]
Dept. Computer and Information Sciences, Universiti Teknologi PETRONAS, Perak, Malaysia[1, 2, 3]
Dept. Information Technology, Shaheed Benazir Bhuto University, Nawabshah, Pakistan[4]
Dept. Information Technology, Tomas Bata University, Zlin, Czech Republic[5]

***Abstract*—Natural Language Processing (NLP) has recently gained significant attention; where, semantic similarity techniques are widely used in diverse applications, such as information retrieval, question-answering systems, and sentiment analysis. One promising area where NLP is being applied, is personalized learning, where assessment and adaptive tests are used to capture students' cognitive abilities. In this context, open-ended questions are commonly used in assessments due to their simplicity, but their effectiveness depends on the type of answer expected. To improve comprehension, it is essential to understand the underlying meaning of short text answers, which is challenging due to their length, lack of clarity, and structure. Researchers have proposed various approaches, including distributed semantics and vector space models, However, assessing short answers using these methods presents significant challenges, but machine learning methods, such as transformer models with multi-head attention, have emerged as advanced techniques for understanding and assessing the underlying meaning of answers. This paper proposes a transformer learning model that utilizes multi-head attention to identify and assess students' short answers to overcome these issues. Our approach improves the performance of assessing the assessments and outperforms current state-of-the-art techniques. We believe our model has the potential to revolutionize personalized learning and significantly contribute to improving student outcomes.**

Keywords—Natural language processing; short text; answer assessment; BERT; semantic similarity

I. Introduction

Semantic similarity is a technique used to determine whether two separate texts have the same meaning. It is a crucial task in natural language processing (NLP) and can be applied to a range of downstream applications, such as text classification, summarization, and question-answering systems (QAS). In the early days of text similarity research, the emphasis was often on comparing lengthy texts, such as news articles, large corpora, and documents. Compared to lengthy writings, short texts have unique characteristics that pose challenges to traditional approaches for measuring similarity. First, short texts have a shorter form, which means that traditional approaches such as knowledge-based, and corpus-based which rely on examining common terms in two texts to determine similarity often lack statistical evidence to support them [1]. Second, short writings frequently use colloquial language and contain numerous typographical and grammatical errors. Third, due to the huge volume of short messages produced, they tend to be ambiguous and noisy [2]. Consequently, it is difficult to use traditional text similarity methods for short texts. There are three main methods for calculating the similarity of short texts. The first method is word-level semantic-based, which looks at the words in the texts and finds pairs of similar words. It then calculates the similarity of the whole text based on the similarity of these word pairs. The second method is semantic modeling-based, which looks at the overall structure of the texts and compares the two models to see how similar they are. The third method is deep learning-based, which converts the short texts into "word embeddings" and calculates how close the words are to each other using cosine similarity [3]. Other approaches such as convolutional neural networks (CNN) and recurrent neural networks (RNN) can take a long time to train due to their sequential processing of information. However, most of the supervised work in NLP is done using a human-annotated corpus. This approach involves two steps: first, candidate phrases are extracted using a heuristic method, and then a classification model is trained to determine whether the phrase is from an answer or a sentence [4-5]. Other deep learning approaches such as pre-trained sentence transformers, like GPT-1, BERT, XLNet, Roberta, and ELECTRA, have been incredibly successful because they can learn a universal language representation from vast amounts of unlabeled text data. Moreover, the transformer learning model has led to a lot of progress in machine learning. It uses a sequence-to-sequence architecture and an attention mechanism to determine the significance of words in a sequence. This mechanism imitates the way humans read and think. Transformer-based models use a feature extraction technique that creates a vector for each word in the sequence based on its relevance to the other words. The aim of this study is to evaluate students' short subjective answers with the help of a multi-head attention transformer learning model based on the BERT language model, which is a promising method for improving the accuracy of student assessment. Student assessment is a crucial aspect of classroom instruction, involving evaluating students' knowledge, understanding, and skills to inform instruction and support student learning. Various forms of assessment, including quizzes, exams, projects, and presentations, serve to measure student progress and identify areas where additional support is needed [6-7]. By providing feedback to both students and teachers, effective assessment practices can help ensure that students are meeting learning goals and enable teachers to tailor instruction to meet the needs of individual students.

The key contributions of this study are:

1) To develop a BERT-based transformer learning model that utilizes multi-head attention to accurately identify and assess students' short answers in personalized learning.

2) To evaluate the effectiveness of the proposed transformer learning model in improving the accuracy of assessments compared to current state-of-the-art techniques.

3) To investigate the challenges of assessing short answers using machine learning methods, such as transformer models with multi-head attention, and propose solutions to overcome these issues.

4) To contribute to the field of NLP and personalized learning by developing an advanced technique for understanding and assessing the underlying meaning of short text answers.

The paper is structured as follows:

Section II provides a comprehensive review of the related literature, Section III elaborates on the proposed method, Section IV presents the results and corresponding discussions, and presents the implications and suggested solutions, and Section V concludes the paper.

II. Related Work

The purpose of the literature review is to examine the various machine learning methods and techniques that are utilized in short text semantic similarity. In order to achieve this objective, we have meticulously analyzed 20 studies to provide a comprehensive overview of the related work.

The field of natural language processing (NLP) is experiencing an increasing use of deep learning techniques. Various attention-based neural network models, including Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Bidirectional Encoder Representations from Transformers (BERT), have captured the attention of numerous researchers. In the domain of short text semantic similarity (STSS), Wang et al. [8] conducted a study in the field of short text semantic similarity. They utilized a Convolutional Neural Network (CNN) to classify short text and identify related words. The authors also used external knowledge and Jaro-Winkler similarity to understand the conceptual meaning of words and detect grammatical errors in sentences, respectively. The research conducted by Shih et al. [9] resulted in the development of a short answer grading system that utilizes a Convolutional Neural Network (CNN). Their CNN model operates as a text classifier to interpret Chinese language responses submitted by students. The authors' use of binary classification methods enabled the grading of answers as either correct or incorrect with relative accuracy. This system has the potential to revolutionize the grading process of short answer responses in the Chinese language.

Furthermore, the DE-CNN model introduced by Xu et al. [10] represents a significant advancement in the area of short-answer comprehension. By utilizing multiple embedding layers, the authors were able to gain a deeper understanding of the context and underlying concepts within short answers. The attention embedding layer further enabled the extraction of concept representations, enhancing the model's accuracy and effectiveness. These findings have implications for the development of future deep-learning models designed for short-answer comprehension. Moreover, the CNN model proposed by Perera et al. [11] represents a significant contribution to the field of web-based question-answering systems. By focusing on factoid questions with short answers, the authors were able to develop a model that effectively identifies irrelevant answers. However, it is worth noting that the model's inability to answer the complete factoid question set highlights a need for continued research in this area. Future studies may consider building on this work to enhance the accuracy and effectiveness of web-based QAS models. The character-level CNN developed by Surya et al. [12] also represents a promising approach to short-answer comprehension. By relying solely on character-level data, the model can learn to identify key information without any prior knowledge of language or semantics. Nevertheless, the challenges faced by the authors in scoring short answers highlight the need for continued research in this area. Future studies may consider developing novel tools and strategies to enhance the effectiveness of short-answer scoring in the context of character-level CNNs. The approach proposed by Liu et al. [13] represents a novel approach to mining and comprehending global features from a short text. By combining the strengths of both CNN and LDA, the authors were able to develop a method that effectively captures both local and global features. This approach may have significant implications for natural language processing tasks, particularly in the context of short text comprehension. The SVMCNN model developed by Hu et al. [14] represents a promising approach to short-text classification. By combining the strengths of both CNN and SVM, the authors were able to develop a more robust model capable of accurately classifying short text data. Moreover, by training the model on the Twitter social platform using TensorFlow, the authors demonstrated the applicability of their method to real-world scenarios. This study may have important implications for the development of more effective short-text classification methods.

Moreover, the LSTM model proposed by Yao et al. [15] represents a novel approach to short text similarity calculation. By utilizing cosine similarity and backward propagation, the authors were able to develop a more accurate and robust model capable of measuring the similarity between short texts. This method may have important implications for various natural language processing tasks, such as information retrieval and question-answering systems. The approach taken by Zhou et al. [16] represents an innovative method for short-text classification using both RNN and CNN. By combining semantic features extracted from both types of neural networks, the authors were able to develop a more comprehensive and effective model for Chinese short-text classification. This study may have important implications for various natural language processing tasks, such as sentiment analysis and document classification. The study conducted by Hassan et al. [17] sheds light on the challenges involved in measuring similarity among short texts and proposes a potential solution using RNN and Tf-IDf vectors. The findings of this research could be beneficial for improving the performance of short text classification and similarity tasks in various fields, such as social media analytics, customer service, and content analysis.

The use of RNNs to generate vector representations of short text is a promising approach for improving the accuracy and efficiency of natural language processing tasks. The evaluation of the model on a standard benchmark dataset like DSTC provides a reliable way to assess its performance and compare it with other models. The research conducted by Lee et al. [18] contributes to the ongoing efforts to develop effective and scalable solutions for processing short text in various domains, such as social media, e-commerce, and customer service.

Furthermore, Mozafari et al. [19] introduced a BERT-based answer selection model (BAS) to capture both the syntactic and semantic information in short question-answer pairs. The model employs pre-trained BERT embeddings to encode the input text and utilizes a binary classification approach to predict whether a given answer is correct or not. The authors evaluated the performance of their proposed model on several benchmark datasets and achieved state-of-the-art results in short answer selection tasks. Wijaya et al. [20] leveraged the BERT model to devise an automated grading system for short answers in the Indonesian language. The study employed Cohen's Kappa to assess the inter-rater reliability among student answers, and the model demonstrated high accuracy in grading the short answers. These findings suggest that the proposed approach holds promise for implementation in educational contexts, where efficient grading mechanisms are essential.

Luo et al. [21] explored the use of the BERT model for grading short answers, similar to the study by Wijaya et al. [20]. However, they utilized a different dataset for training the model, which was the short answer scoring V2.0 dataset. In addition, the study used the regression task function to check the linearity between answers, and found that the BERT model achieved high accuracy in grading short answers. These results indicate the potential of the BERT model in improving the efficiency of grading mechanisms in educational settings. However, further research is needed to investigate the generalizability of these findings across different languages and domains. In the study, Alammary et al. [22] introduced the use of BERT models for short text classification in Arabic. The researchers explored different versions of BERT models and evaluated their effectiveness in classifying Arabic short texts. Furthermore, the study compared the performance of the Arabic BERT models with their English counterparts. This research has significant implications for natural language processing tasks in the Arabic language and can lead to the development of more effective and accurate models for Arabic text classification. Heidari et al. [23] developed a short answer grading system for Indonesian students using domain-independent subjects such as biology and geography. The study employed the BERT model to detect word embeddings from sentences and analyze the contextual information for improved grading accuracy. By integrating domain-specific knowledge into the model, the proposed approach demonstrated high accuracy in grading short answers, suggesting its potential use in educational contexts for efficient and reliable grading. Gaddipati et al. [24] highlighted the distinctions between transformer-based language models such as BERT, GPT, GPT2, and ELMO. Unlike ELMO and GPT, BERT utilizes a transformer mechanism and extracts contextual embeddings in a bidirectional manner. The model is trained on large-scale datasets such as Book Corpus and Wikipedia, which consist of 800M and 2500M words, respectively. Overall, the study sheds light on the unique features and capabilities of these advanced language models. Furthermore, Zhu et al. [25] developed a BERT-based framework for grading short answers, which incorporated CNN and capsule networks, as well as a triple-hot loss strategy to encode key sentences. The approach was tested on a dataset of student short-answer responses and yielded superior results compared to other state-of-the-art methods. These findings indicate that the proposed framework has the potential to significantly improve the accuracy and efficiency of grading mechanisms in educational settings. In addition to the classification of ASAG systems, Burrow et al. [26] also identified several limitations in these systems. One of the main limitations identified was the inability of existing ASAG systems to handle complex or open-ended questions. Another limitation was the reliance of ASAG systems on pre-defined rubrics, which can limit the flexibility of the grading process. The authors suggested that future research should focus on addressing these limitations and developing more sophisticated ASAG systems that can handle a wider range of questions and provide more accurate and flexible grading mechanisms.

On the other hand, Mohler et al. [27] proposed a different approach for grading short answers using lexical semantic similarity. The authors argue that deep learning techniques, such as the ones used in BERT and other transformer-based models, may not be the most effective method for grading short answers because they rely on large amounts of training data. Instead, Mohler et al. utilized a method based on semantic similarity to assess the quality of short answers. By comparing the semantic features of the correct answer and the student's answer, the system was able to assign a score that reflected the level of correctness. This approach may be particularly useful for assessing short answers in domains where training data is limited, and where deep learning models may not be effective. Ye et al. [28] leveraged the BERT model to generate context-sensitive representations and combined it with the GCN model to classify short text. The study demonstrated that the proposed approach achieved high accuracy in classifying short text, indicating its potential application in various natural language processing tasks. By incorporating both contextual and graph-based information, the proposed method may provide a more comprehensive understanding of the meaning of the short text.

III. Methodology

A. Data Collection

This study utilizes the computer science dataset, developed by Mohler et al. [27], which comprises 2443 student answers and 87 questions from 12 assignments in the field of computer science. The dataset includes both, the questions and reference answers, as well as the student responses, and was designed to evaluate the effectiveness of models in grading student answers by comparing them to the evaluator's desired answer. The dataset has been graded by human evaluators who are experts in the field of computer science, and the grading scale ranges from 0 (not correct) to 5 (totally correct). Table I provides a detailed overview of the dataset used in this study. It contains a total of 87 questions and 2442 student responses, which are distributed across 12 different assignments. The number of questions in each assignment varies. The grading of each student response is done by two human graders, and the average of their scores is used as the standard score for each response.

TABLE I. MOHLER DATASET FOR THE ASSESSMENT

Institute	University of North Texas
Class/domain	Introductory computer science class
Course	Data structure
Assignments	12 assignments
Questions	87
Answers	2442
Score by human evaluators	0-5
Data collection	WebCT online learning environment
Type of questions	Open-ended

Additionally, it is worth noting that the dataset used in this study exhibits a bias toward correct answers. The dataset comprises both very short and very long answers. However, to ensure a fair evaluation, we have only included answers containing 10-20 words from the test dataset, which is considered an ideal size for short answers. Fig. 1(a) and Fig. 1(b) present the score assigned by human evaluators which is inconsistent, and Fig. 1(b) illustrates the biased nature of the dataset toward correct answers.

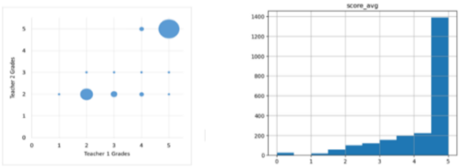


Fig. 1. (a) Distribution of grades and (b) Histogram of scores assigned by human evaluators [29].

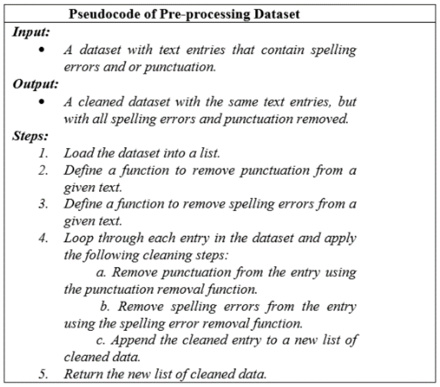
Table II illustrates the sample example from the dataset which contains questions, teacher answers, student answers, and scores assigned by teachers.

TABLE II. ILLUSTRATES THE SAMPLE EXAMPLE FROM THE DATASET

	Sample of questions, teacher answers, and student answers	Grades
Question. Teacher answer.	What is a variable? A location in memory that can store a value.	
Student answer: Student answer:	A block of memory that holds a specific type of data. A pointer to a location in memory.	5,5 3,5
Question. Teacher answer.	What is a pointer? A variable that contains the address in memory of another variable.	
Student answer: Student answer:	A pointer holds a memory location. Is a reference call to the place in memory where the object is stored.	5,4 3,4

B. Pre-processing

Before training and testing our machine learning model, we performed several pre-processing steps on the dataset to ensure that the text data was in a clean and normalized format. One issue we encountered was that many of the student answers contained spelling errors and unnecessary punctuation, which can introduce noise and sparsity into the dataset and negatively impact the performance of our model. To address this issue, we implemented techniques such as spell-check and punctuation removal to clean and normalize the text data. These steps involved identifying and correcting misspelled words, and removing unnecessary punctuation marks that may interfere with our analysis. Additionally, we also performed other pre-processing steps as mentioned in Fig. 2 such as removing stop words and converting text to lowercase to further enhance the quality and consistency of the data. By carefully pre-processing the dataset, we were able to significantly improve the accuracy and effectiveness of our machine-learning model, and ultimately generate more reliable and informative results. The following algorithm 1 mentions the pre-processing steps applied to the dataset (see Fig. 2).



```
Pseudocode of Pre-processing Dataset
Input:
  • A dataset with text entries that contain spelling errors and or punctuation.
Output:
  • A cleaned dataset with the same text entries, but with all spelling errors and punctuation removed.
Steps:
  1. Load the dataset into a list.
  2. Define a function to remove punctuation from a given text.
  3. Define a function to remove spelling errors from a given text.
  4. Loop through each entry in the dataset and apply the following cleaning steps:
       a. Remove punctuation from the entry using the punctuation removal function.
       b. Remove spelling errors from the entry using the spelling error removal function.
       c. Append the cleaned entry to a new list of cleaned data.
  5. Return the new list of cleaned data.
```

Fig. 2. Cleaning of the dataset.

C. BERT-Multi-Head Attention Model

BERT is a transformer-based model that utilizes bidirectional processing and attention mechanism to understand language. Several versions of the BERT model such as Roberta, KeyBERT, M-BERT, and SBERT have been introduced. We used the BERTbase-uncased model as it has shown the best performance on NLP tasks. This model can encode various languages but utilizes the default English vocabulary. The architecture of the BERT model with tokens is shown in Fig. 3, where E1-EN generates the input tokens, and T1-TN are output tokens that categorize phrases using binary representations and deliver them to the C-label. BERT employs a masked language modeling technique that predicts incoming words based on the surrounding context. This technique changes 15% of the words in a sentence presented in Fig. 4 where 80% are converted to "mask" tokens, 10% to random words, and 10% to their previous representations. BERT evaluates the accuracy of its predictions and fine-tunes accordingly. Compared to an implementation of BERT that operates without masking, BERT coverage is slower but reaches a higher threshold. The next sentence prediction (NSP) examines whether the two sentences are connected logically, providing contextual information for both sentences.

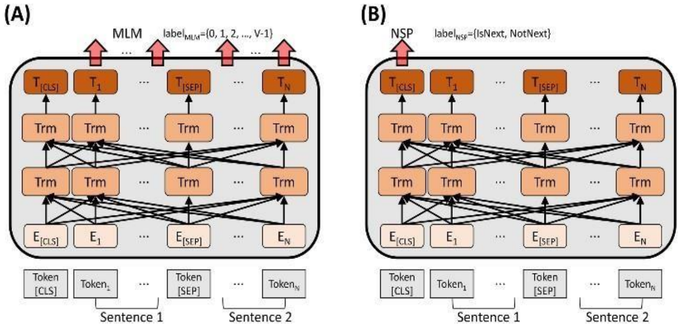


Fig. 3. BERT language model.

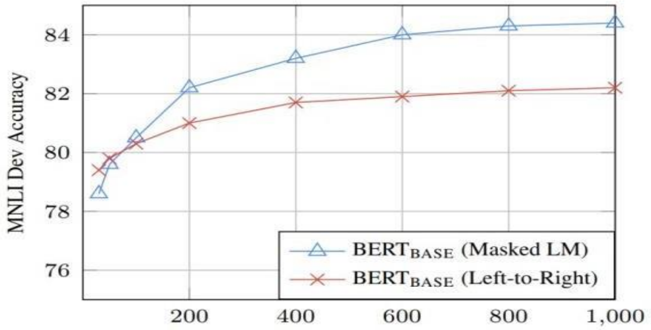


Fig. 4. BERT model with and without masked language modeling.

It uses a transformer-based architecture to process text data and learn contextual representations. One of the key components of transformer architecture is multi-head attention. Multi-head attention is a type of attention mechanism that allows the model to attend to different parts of the input sequence simultaneously. In the case of BERT, multi-head attention is used in both the encoder and decoder components of the transformer architecture. The BERT multi-head attention model consists of three components: query, key, and value matrices as shown in Fig. 5. These matrices are used to compute the attention scores, which determine how much attention each token in the input sequence should receive. The query matrix represents the current token that is being processed, while the key and value matrices represent all the other tokens in the sequence. The multi-head attention mechanism in BERT involves splitting the query, key, and value matrices into multiple heads. Each head has its own set of parameters and is trained to attend to a different part of the input sequence. This allows the model to capture different aspects of the input sequence and learn more complex relationships between the tokens. The output of the multi-head attention mechanism is computed as the weighted sum of the values, where the weights are determined by the attention scores. The attention scores are computed by taking the dot product of the query matrix and the key matrix and then applying a SoftMax and linear function to normalize the scores. The resulting attention vector is then multiplied by the value matrix to obtain the output. The BERT multi-head attention model also includes a layer normalization step after the output is computed. Layer normalization ensures that the output has a mean of zero and a standard deviation of one, which helps to improve the stability and performance of the model.

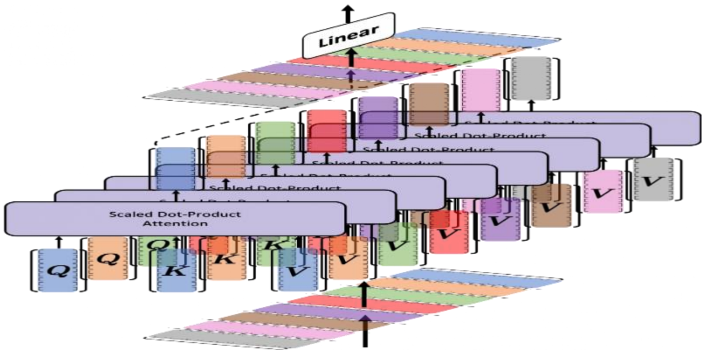


Fig. 5. Multi-head attention.

The implementation method utilized the BERT multi-head attention model, a monolingual model solely evaluated on the English dataset. The process involved setting up sentence transformers on a dataset and fine-tuning the model through a question-answer task. The results were evaluated using statistical approaches. Fine-tuning the model ensured that it could accurately understand the context of the given task, making it suitable for various natural language processing applications. Additionally, the implementation method applied the attention mechanism to highlight the answers that best matched the teacher's answer. This approach enabled the model to provide more accurate responses and perform better in question-answering tasks. Fig. 6 presents a visual representation of the implementation process. By following this process, the model can be optimized for specific tasks, resulting in better performance. It is worth noting that this implementation method is limited to the English language, as BERT is a monolingual model. However, there are other models available that support multiple languages. Overall, the BERT multi-head attention model is a powerful method for natural language processing tasks and has been widely adopted in various applications, including chatbots, sentiment analysis, and language translation.

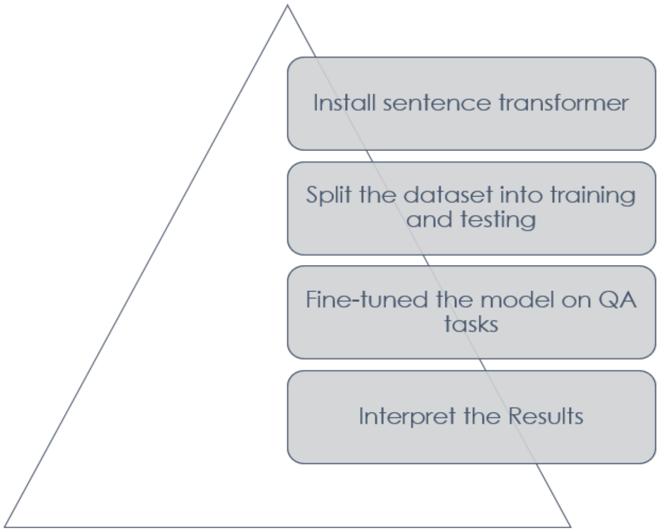

Install sentence transformer
Split the dataset into training and testing
Fine-tuned the model on QA tasks
Interpret the Results

Fig. 6. Implementation details.

D. Training and Testing

To evaluate the performance of our approach, we utilized the BERT multi-head attention model. We randomly split the Mohler dataset into 70% training and 30% testing data, ensuring that the split was representative of the entire dataset. We trained the model for 1000 iterations, using different training and testing data for each iteration to improve the generalization of the results. The cosine similarity feature was trained using isotonic, linear, and non-linear (ridge) regression models, and the performance was compared to previously established models such as Mohler et al. [27]. Following training, we evaluated the model by testing it on the unseen test data. The similarity scores of the test data were input through the trained regression model, resulting in predicted grades that were compared to the desired scores. We calculated the Root Mean Square Error (RMSE) and Pearson correlation to evaluate the model's performance. Our utilization of the BERT multi-head attention model allowed us to effectively analyze and evaluate the performance of our approach on the Mohler dataset. The results of this study could have significant implications for future natural language processing applications, particularly those that require accurate grading of written responses.

IV. Results

A. Feature Extraction

In the feature extraction, we use the pre-trained embeddings from each transfer learning model and assign them to the tokens of every word in all the answers. To create answer embeddings, we use the Sum of Word Embeddings (SOWE) method, as shown in Eq. (1). Here, ai j denotes the j-th answer vector of question qi, and wk represents the vector of the k-th word in the answer ai j. By applying this method, we obtain a single vector that represents each answer in a high-dimensional hypothesis space. The resulting sentence embeddings have the same size as the word embeddings. This approach allows us to capture the semantic and syntactic properties of each answer and create a compact representation of it.

$$aij = \sum k=1 \text{ to } nj\ wk \qquad (1)$$

In this equation, "aij" represents the vector of the jth answer of the question "qi", "wk" represents the vector of the kth word in the answer "aij", and "nj" represents the number of words in the answer "aij". The equation calculates the sum of the word embeddings for each word in the answer to create a single vector representing the entire answer. To create a Question-Answering model using BERT, the tokenizer utilizes two special tokens, namely [CLS] and [SEP]. These tokens serve the purpose of encoding the sentence sequence. The [CLS] token is a classification token, whereas the [SEP] token separates the Key and response answer, as exemplified below. The sequence of sentences is then passed as a token input to the BERT model for training [32,35]. The model generates high-dimensional embeddings for input tokens, which are then used to predict the grades within a specified range. An example of BERT embeddings is given below.

Question: What is a variable??

Key Answer: A location in memory that can store a value.

Student Answer: a block of memory that holds a specific type of data.

[CLS] and [SEP]: [CLS] a location in memory that can store a value [SEP] A block of memory that holds a specific type of data

Token ids of both responses:

[101, 10408, 1996, 9896, 1998, 5468, 1995, 3558, 2770,

2051, 1012, 102, 270, 2019, 9896, 2006, 103, 3563, 102]

Example of Response pair and Token Id’s

Additionally, the multi-head attention mechanism is utilized to visualize the relationships between words. The lines that are darker in color indicate a closer relationship between words at layers 1,2,3,4.

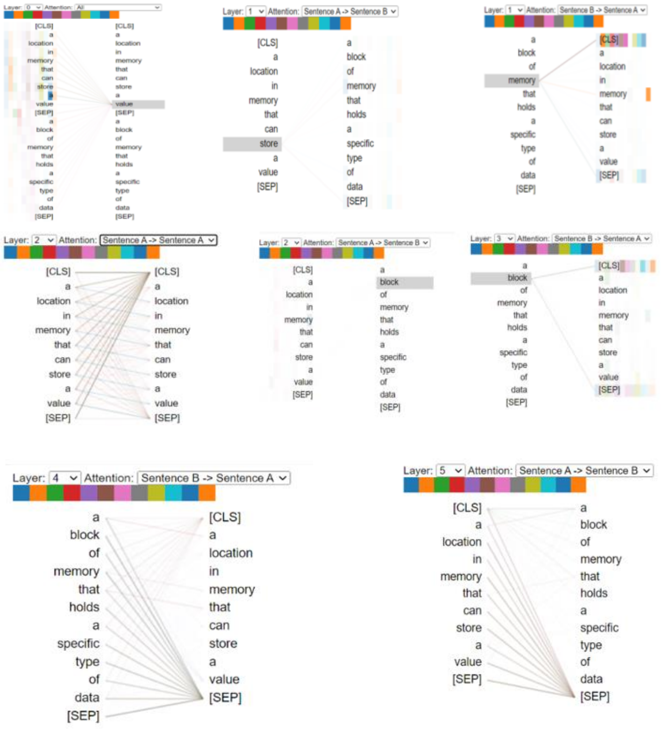

Layer: 0 Attention: All
Layer: 1 Attention: Sentence A -> Sentence B
Layer: 1 Attention: Sentence B -> Sentence A
Layer: 2 Attention: Sentence A -> Sentence A
Layer: 2 Attention: Sentence A -> Sentence B
Layer: 3 Attention: Sentence B -> Sentence A
Layer: 4 Attention: Sentence B -> Sentence A
Layer: 5 Attention: Sentence A -> Sentence B
[CLS] a location in memory that can store a value [SEP]
a block of memory that holds a specific type of data [SEP]

Fig. 7. BERT-based multi-head attention model for layers 0, 1, 2, 3, 4, 5

The findings obtained from the BERT-based multi-head attention model are extremely encouraging (see Fig. 7). Layer 0 has highlighted the importance of the [CLS] and [SEP] tokens, as they effectively emphasize the embeddings from the text. Layer 1 has shown a strong correlation between the words "store" and "memory," indicating that they are related. In layers 2, 3, and 4, words such as "block," "value," "memory," "type," and "hold" also share strong embeddings, indicating their interconnectedness. Moreover, the relationship between student and teacher responses has also been established, as their embeddings show strong correlations. Layer 0 also performed self-attention with multi-heads to determine the relationships between different parts of the answer itself. The multi-head attention module utilized teacher-to-teacher, teacher-to-student, and student-to-student attention to determine the strong impact of different words. This approach has proved to be highly effective in identifying the key components of the response and the underlying relationships between them. The results obtained through this model have significant implications for natural language processing, particularly in the field of language understanding and interpretation. The multi-head attention mechanism can be used to improve the accuracy of machine translation, question-answering systems, and text classification algorithms, among others. Overall, the BERT-based multi-head attention model has demonstrated its ability to capture complex relationships between different components of natural language text. This approach has the potential to revolutionize the field of natural language processing and enable more accurate and efficient analysis of text data.

B. Automated Scoring

To evaluate student answers, we employed text similarity techniques that match the embeddings of teacher answers to student responses. We used Python programming with Spyder IDE to implement this process. After extracting the embeddings using the BERT-multi-head attention model, we used four similarity methods to obtain a similarity value, which we used as a scoring rubric. These four methods are the longest common subsequence (LCS), cosine coefficient (SC), Jaccard coefficient (JC), and Dice coefficient (DC). Additionally, we chose these methods as they are commonly used in Natural Language Processing (NLP) and provide reliable results for text similarity comparisons.

$$Score = \frac{(Score(sim)*answer\ score)+(Score(keymatch)*answer\ score)}{2} \tag{2}$$

Example to calculate the similarity of teacher and student answers:

Teacher answer: a location in memory that can store a value.

Student answer: a block of memory that holds a specific type of data.

$$\mathrm{Sim}_{\mathrm{lcs}} = \frac{2*40}{58*40} = 0,81633 \quad sim(cosine) = \sqrt[6]{9*6} = 0,81650 \quad score(keymatch) = 6/9 = 0,66667$$

$$\mathrm{Sim}_{\mathrm{jaccard}} = \frac{6}{9} = 0,66667 \quad Simdice = \frac{2*6}{9+6} = 0,80000$$

To determine the score for each student response, we derived a scoring rubric by averaging the similarity values obtained through two different methods: the String-based method and the keyword-matching technique. The String-based method involves comparing the strings in the form of embedding values of the teacher's and student's answers to identify common sub-sequences and measure their similarity. On the other hand, the keyword matching technique involves identifying the presence of specific keywords in the student's answer that are expected based on the question or prompt.

In the previous example, we obtained a similarity score of 0.66667. However, it should be noted that we have multiple references available, and we select the highest similarity score among them. In this case, the highest similarity score is 0.85714.

$$\text{Score} = \frac{(0.81633*4)+(0.85714*4)}{2} = 3.34695$$

Therefore, we consider this score as the final similarity score for the response in question. This approach helps to ensure that the students receive a fair and accurate evaluation, as we consider all available references and select the most appropriate one.

C. Comparative Evaluation

Our model's performance was evaluated using RMSE and Pearson correlation scores, and we conducted a comparative analysis of our model's performance with various pre-trained models, such as ELMO, GPT, and GPT2, as reported by Gaddipati et al. [24], on the Mohler dataset. We further compared our model's performance with other approaches and showed that the BERT-multi-head attention method outperformed other techniques in terms of effectiveness. Table III displays the Root Mean Square Error (RMSE) and Pearson correlation (ρ) results of various Pre-trained transfer learning models on the Mohler Dataset.

Table IV compares the performance of different models and approaches on the Mohler dataset. The models are evaluated based on their RMSE (Root Mean Square Error) and Pearson correlation scores. The results for the BOW (Bag of Words) approach with SVMRank, achieve an RMSE of 1.042 and a Pearson correlation score of 0.480. The results for the tf-idf approach with SVR, which performs slightly better than the BOW approach with an RMSE of 1.022 and a Pearson correlation score of 0.327. The results for the tf-idf approach with LR (Logistic Regression) and SIM (Semantic Information), which outperforms the previous two approaches with an RMSE of 0.887 and a Pearson correlation score of 0.592. Furthermore, the results for three different word embedding models - Word2Vec, GloVe, and FastText; all of these models use SOWE (Sum of Word Embeddings) and Verb phrases features, and they achieve RMSE values ranging from 1.023 to 1.036 and Pearson correlation scores ranging from 0.425 to 0.465. The results for deep learning models - ELMo, GPT-2, and Roberta; ELMo uses a 5-layer BiLSTM (Bidirectional Long Short-Term Memory) with max-pooling and achieves an RMSE of 0.875 and a Pearson correlation score of 0.655. GPT-2 uses a 12-layer Transformer and achieves an RMSE of 0.911 and a Pearson correlation score of 0.610. Roberta uses a 24-layer Transformer and achieves the best performance among all the models with an RMSE of 0.851 and a Pearson correlation score of 0.692. Results of our BERT-based model (our approach) with Multihead Attention as the feature has RMSE value as 1.990, which means that on average, our model's predictions deviate from the actual values by 1.990 points. The Pearson correlation coefficient is 0.773, which indicates a strong positive correlation between our model's predicted scores and the actual scores. Overall, the RMSE value of 1.990 is higher than the RMSE value of the RoBERTa model (0.851), which indicates that our model has a higher prediction error than RoBERTa. However, the Pearson correlation coefficient of our model (0.773) is higher than that of RoBERTa (0.692), indicating that our model's predicted scores are more strongly correlated with the actual scores than RoBERTa's predictions.

TABLE III. A SUMMARY OF THE RESULTS OBTAINED BY VARIOUS APPROACHES ON THE MOHLER DATASET IS PRESENTED

Model	**Isotonic regression**		**Linear regression**		**Ridge regression**	
	RMSE	**P**	**RMSE**	**p**	**RMSE**	**P**
ELMO [24]	0.978	0.485	0.995	0.451	0.996	0.449
GPT [24]	1.082	0.248	1.088	0.222	1.089	0.217
GPT2 [24]	1.065	0.311	1.077	1.075	1.079	0.269
BERT-multi-head attention **Our model**	1.089	0.456	1.990	0.773	1.536	0.876

TABLE IV. COMPARISON WITH OTHER METHODS

Model	**Features**	**RMSE**	**Pearson Correlation**
BOW [29]	SVMRank	1.042	0.480
TF-idf [30]	SVR	1.022	0.327
Tf-idf [31]	LR+SIM	0.887	0.592
Word2Vec[33]	SOWE + Verb Phrases	1.025	0.458
Glove [34]	SOWE +Verb Phrases	1.036	0.425
FastText[35]	SOWE+Verb Phrases	1.023	0.465
ELMO [24]	5-layer BiLTSM +max-pooling	0.875	0.655
GPT-2 [24]	12-layer transformer	0.911	0.610
Roberta[36]	24-layer transformer	0.851	0.692
BERT (our approach)	Multihead Attention	1.990	0.773

D. Model Implications

One of the primary challenges is the limited availability of training data for short answers. Short answers are usually context-dependent and diverse in nature, making it difficult to generate large amounts of high-quality training data. Additionally, there is often ambiguity and variation in short answers, which makes it challenging for machine learning models to accurately evaluate them. Another challenge is the need for efficient methods to encode short answers and generate embeddings that can be used for similarity matching. Transformer models with multi-head attention have shown promise in this regard, but there is a need for further research to optimize their performance for short answer evaluation. Furthermore, there is a need for developing robust methods to handle outliers, exceptions, and edge cases that are often encountered in short answer assessment. This requires careful consideration of the characteristics of short answers and the design of models and algorithms that can handle such situations effectively [36-40]. To overcome these challenges, potential solutions include utilizing data augmentation techniques to generate more diverse training data, developing novel algorithms and models specifically tailored for short answer assessment, and leveraging domain-specific knowledge and expertise to enhance the performance of machine learning models. Additionally, the use of ensemble methods and human-in-the-loop approaches may improve the accuracy and reliability of short answer evaluation.

V. CONCLUSION AND FUTURE WORK

This study aimed to assess short subjective answers using a BERT-based multi-head attention model and string-based methods such as cosine co-efficient, longest common subsequence, Dice coefficient, and Jaccard coefficient to score the answers. Additionally, we compared the performance of the BERT multi-head attention model with former approaches using isotonic, linear, and ridge regression. The findings suggest that the BERT multi-head attention model outperforms other approaches, indicating its effectiveness in understanding and assessing the underlying meaning of short answers. Our study highlights the potential of machine learning methods in improving the efficiency of personalized learning, particularly in the assessment of open-ended questions. Overall, this study contributes to the growing body of research on NLP techniques and their applications in the education domain. Further research can explore the generalizability of our proposed model in different educational settings and subject domains.

ACKNOWLEDGMENT

Appreciation goes to the Pre-commercialization-External: YUTP-PRG Cycle 2022 (015PBC-005)

REFERENCES

[1] B. Agarwal, H. Ramampiaro, H. Langseth, M. J. I. P. Ruocco, and Management, "A deep network model for paraphrase detection in short text messages," vol. 54, no. 6, pp. 922-937, 2018.

[2] Z. H. Amur, Y. Kwang Hooi, H. Bhanbhro, K. Dahri, and G. M. J. A. S. Soomro, "Short-Text Semantic Similarity (STSS): Techniques, Challenges and Future Perspectives," vol. 13, no. 6, p. 3911, 2023.

[3] Z. H. Amur, Y. K. Hooi, and G. M. Soomro, "Automatic Short Answer Grading (ASAG) using Attention-Based Deep Learning MODEL," in 2022 International Conference on Digital Transformation and Intelligence (ICDI), 2022.

[4] H. Bhanbhro, Y. K. Hooi, and Z. Hassan, "Modern Approaches towards Object Detection of Complex Engineering Drawings," in 2022 International Conference on Digital Transformation and Intelligence (ICDI), 2022.

[5] N. Carlini et al., "Extracting training data from large language models," in 30th USENIX Security Symposium (USENIX Security 21), 2021.

[6] H. Bhanbhro, Y. Kwang Hooi, W. Kusakunniran, and Z. H. J. A. S. Amur, "A Symbol Recognition System for Single-Line Diagrams Developed Using a Deep-Learning Approach," vol. 13, no. 15, p. 8816, 2023.

[7] L. B. Galhardi and J. D. Brancher, "Machine learning approach for automatic short answer grading: A systematic review," in Ibero-american conference on artificial intelligence, 2018.

[8] H. Wang, K. Tian, Z. Wu, and L. J. I. J. o. C. I. S. Wang, "A short text classification method based on convolutional neural

network and semantic extension," vol. 14, no. 1, pp. 367-375, 2021.

[9] S.-H. Wu and C.-Y. Yeh, "A Short Answer Grading System in Chinese by CNN," in 2019 IEEE 10th International Conference on Awareness Science and Technology (iCAST), 2019.

[10] J. Xu et al., "Incorporating context-relevant concepts into convolutional neural networks for short text classification," vol. 386, pp. 42-53, 2020.

[11] N. Perera, C. Priyankara, and D. Jayasekara, "Identifying Irrelevant Answers in Web Based Question Answering Systems," in 2020 20th International Conference on Advances in ICT for Emerging Regions (ICTer), 2020.

[12] K. Surya, E. Gayakwad, and M. J. I. J. R. T. E. Nallakaruppan, "Deep learning for short answer scoring," vol. 7, no. 6, pp. 1712-1715, 2019.

[13] J. Liu, H. Ma, X. Xie, and J. J. E. Cheng, "Short Text Classification for Faults Information of Secondary Equipment Based on Convolutional Neural Networks," vol. 15, no. 7, p. 2400, 2022.

[14] Y. Hu, J. Ding, Z. Dou, H. J. C. I. Chang, and Neuroscience, "Short-text classification detector: a bert-based mental approach," vol. 2022, 2022.

[15] L. Yao, Z. Pan, and H. J. I. A. Ning, "Unlabeled short text similarity with LSTM encoder," vol. 7, pp. 3430-3437, 2018.

[16] Y. Zhou, B. Xu, J. Xu, L. Yang, and C. Li, "Compositional recurrent neural networks for chinese short text classification," in 2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI), 2016.

[17] Z. H. Amur and Y. K. J. I. S. L. Hooi, "State-of-the-Art: Assessing Semantic Similarity in Automated Short-Answer Grading Systems," vol. 11, pp. 1851-1858, 2022.

[18] J. Y. Lee and F. J. a. p. a. Dernoncourt, "Sequential short-text classification with recurrent and convolutional neural networks," 2016.

[19] J. Mozafari, A. Fatemi, and M. A. J. a. p. a. Nematbakhsh, "BAS: an answer selection method using BERT language model," 2019.

[20] M. Wijaya, "Automatic Short Answer Grading System in Indonesian Language Using BERT Machine Learning," vol. 35, no. 6, pp. 503-509, 2021.

[21] J. Luo, "Automatic Short Answer Grading Using Deep Learning," Illinois State University, 2021

[22] A. S. J. A. S. Alammary, "BERT Models for Arabic Text Classification: A Systematic Review," vol. 12, no. 11, p. 5720, 2022.

[23] M. Heidari, J. H. Jones, and O. Uzuner, "Deep contextualized word embedding for text-based online user profiling to detect social bots on twitter," in 2020 International Conference on Data Mining Workshops (ICDMW), 2020, pp. 480-487: IEEE.

[24] S. K. Gaddipati, D. Nair, and P. Plöger, "Comparative evaluation of pretrained transfer learning models on automatic short answer grading," 2020.

[25] X. Zhu, H. Wu, and L. J. I. T. o. L. T. Zhang, "Automatic Short-Answer Grading via BERT-Based Deep Neural Networks," vol. 15, no. 3, pp. 364-375, 2022.

[26] S. Burrows, I. Gurevych, and B. J. I. J. Stein, "The eras and trends of automatic short answer grading," vol. 25, no. 1, pp. 60-117, 2015.

[27] M. Mohler, R. Bunescu, and R. Mihalcea, "Learning to grade short answer questions using semantic similarity measures and dependency graph alignments," in Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies, 2011, pp. 752-762.

[28] Z. Ye, G. Jiang, Y. Liu, Z. Li, and J. Yuan, "Document and word representations generated by graph convolutional network and bert for short text classification," in ECAI 2020: IOS Press, 2020, pp. 2275-2281.

[29] N. Süzen, A. N. Gorban, J. Levesley, and E. M. J. P. c. s. Mirkes, "Automatic short answer grading and feedback using text mining methods," vol. 169, pp. 726-743, 2020.

[30] R. Mihalcea and P. Tarau, "Textrank: Bringing order into text," in Proceedings of the 2004 conference on empirical methods in natural language processing, 2004, pp. 404-411.

[31] S. Jimenez, S.-P. Cucerzan, F. A. Gonzalez, A. Gelbukh, G. J. J. o. I. Dueñas, and F. Systems, "BM25-CTF: Improving TF and IDF factors in BM25 by using collection term frequencies," vol. 34, no. 5, pp. 2887-2899, 2018

[32] K. W.Church, "Word2Vec," vol. 23, no. 1, pp. 155-162, 2017.

[33] H. Al-Bataineh, W. Farhan, A. Mustafa, H. Seelawi, and H. T. Al-Natsheh, "Deep contextualized pairwise semantic similarity for arabic language questions," in 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI), 2019.

[34] D. Cer et al., "Universal sentence encoder," 2018.

[35] W. Hua, Z. Wang, H. Wang, K. Zheng, and X. Zhou, "Short text understanding through lexical-semantic analysis," in 2015 IEEE 31st international conference on data engineering, 2015, pp. 495-506: IEEE.

[36] A. Hassan and A. Mahmood, "Deep learning approach for sentiment analysis of short texts," in 2017 3rd international conference on control, automation and robotics (ICCAR), 2017.

[37] Z. H. Amur, Y. K. Hooi, G. M. Soomro, H. Bhanbhro, S. Karyem, and N. J. A. S. Sohu, "Unlocking the Potential of Keyword Extraction: The Need for Access to High-Quality Datasets," vol. 13, no. 12, p. 7228, 2023.

[38] M. A. Memon, Z. Hassan, K. Dahri, A. Shaikh, and M. A. J. I. Nizamani, "Aspect Oriented UML to ECORE Model Transformation," vol. 11, no. 3, 2019.

[39] Z. Hassan, Z. Bhatti, K. J. U. o. S. J. o. I. Dahri, and C. Technology, "A conceptual framework development of the social media learning for undergraduate students of University of Sindh," vol. 3, no. 4, pp. 178-184, 2019.

[40] A. R. Gilal, A. Waqas, B. A. Talpur, R. A. Abro, J. Jaafar, and Z. H. Amur, "Question Guru: An Automated Multiple-Choice Question Generation System," in International Conference on Emerging Technologies and Intelligent Systems, 2022.