ENHANCED AUTOMATIC SHORT-ANSWER GRADING SYSTEM USING BERT MODEL AND BI-LSTM NETWORK

ABSTRACT

Automatic short-answer grading (ASAG) plays a vital role in intelligent tutoring systems (ITS) by assessing students' responses to questions. Deep learning has emerged as a sophisticated technique for comprehensively understanding textual content. However, applying deep learning to ASAG poses challenges, particularly in achieving high-precision scoring due to the complexity of evaluating short answers. Moreover, the limited size of ASAG's corpus restricts the availability of sufficient training data for effective deep learning models. A novel approach using a BERT-based deep neural network for ASAG is introduced to address these challenges. The Bidirectional Encoder Representations from Transformers (BERT) model is leveraged to enhance understanding of short-answer texts. To further enhance the ASAG system's performance, a specialized semantic refinement layer called Bidirectional Long short-term memory (Bi-LSTM) is integrated to generate high precision scores. Overall, by introducing a BERT and Bi-LSTM in the ASAG system can overcome challenges related to understanding short answers, handling limited training data, and improving grading accuracy.

KEYWORDS

Automatic short-answer grading (ASAG), Bidirectional Encoder Representations from Transformers (BERT), Bi-LSTM network, Intelligent tutoring systems (ITSs), Textual entailment, Deep learning, Deep neural network.

INTRODUCTION

Teachers contribute significantly to developing competent future generations. One of the Key tasks of a teacher is to evaluate students based on their performance. The assessment method is essential for determining how well students can comprehend and achieve the learning objectives on a cognitive level. Evaluating student work is a time-consuming and laborious task for teachers. With the increasing demand for personalized and effective learning, there is a growing need for an “Automated answer grading systems”. Such systems can provide instant feedback to students, reduce teacher workload, and increase consistency in grading[31]. In a learning community, various grading methods are used to support the submitted assignments, homework, tests, and so on. The trend in grading methods is moving toward machine learning (ML) approaches which enable to make the assessment of short answers through various techniques [22].

Automatic Short-Answer Grading system is the key component of intelligent tutoring systems (ITSs) [6] [12], [31]. An Intelligent Tutoring System (ITS) is a computer-based instructional system that provides personalized and adaptive instruction to learners, mimicking the guidance and feedback provided by a human tutor. Adaptive testing and assessment in intelligent tutoring systems (ITSs) are necessary to provide personalized learning routes [6], [10], [12], [29], [31]. It highlights the significance of capturing students' cognitive levels accurately to tailor educational experiences effectively [8]. This work emphasizes the role of automatic short-answer grading (ASAG) in evaluating student responses efficiently within ITSs. Both students’ answers and reference answers appear in the form of natural language[22], [23, [32] ASAG is regarded as an application of recognizing textual entailment in educational technology[20], [31].

With the continuous development of artificial neural network technologies, many deep learning models, such as Long Short-Term Memory (LSTM)-based models [4], [9], convolutional neural network (CNN) [3], [27], [28] and LSTM-based model [19], and transformer-based model [1], are applied to short-answer grading. The existing methods for ASAG still remain challenging mainly for two reasons. First, students usually use different free texts to answer the same question, in which students’ answers may have

significant differences in sentence structure, language style, and text length[23], [33]. Therefore, it is necessary to utilize advanced learning techniques to combine different deep neural networks in the ASAG task in order to achieve a deeper semantic understanding of students' answers [14], [29], [35]. Second, the deep learning method for ASAG is fully supervised machine learning, which requires assigning a label to each students' answer in the training corpus[34]. Since it is very time-consuming to accurately assign a label to a freely expressed students' answer, the training corpus for ASAG is usually small in size, which is usually only a few thousand instead of the tens of thousands in conventional deep learning tasks [7]. Therefore, how to train a stable and effective deep neural network model on a small corpus is a major challenge faced by the deep learning methods of ASAG.

To address the issues and challenges mentioned above, We extend the application of the pretrained BERT model [13] in ASAG tasks from a fine-tuning approach to a combination with bidirectional LSTM (Bi-LSTM) [4], [26]. On top of the fine-tuned BERT model, we construct a semantic refinement layer to refine the semantics of BERT outputs [20], which consists of a Bi-LSTM network with position information [24]. Overall, by introducing a BERT-based deep neural network [18], [22] framework with a semantic refinement layer, the ASAG system can overcome challenges related to understanding short answers, handling limited training data, and improving grading accuracy.

The proposed framework that leverages Bidirectional Encoder Representations from Transformers (BERT)-based deep neural networks to enhance the grading accuracy of short answers. Bidirectional Encoder Representations from Transformers, is a natural language processing (NLP) model developed by Google in 2019. It utilizes a transformer architecture and is pre-trained on large amounts of text data to understand context and relationships between words. BERT is often used for various NLP tasks, such as text classification, question answering, and named entity recognition [20], [25]. A semantic refinement layer is also constructed, incorporating a bidirectional Long Short-Term Memory (LSTM) network with position information to refine the semantics of BERT outputs. Long Short-Term Memory is a type of recurrent neural network. LSTM has memory cells with gates that control the flow of information, allowing them to capture and store long-term dependencies in sequential data. This makes LSTMs particularly effective for tasks involving sequences, such as natural language processing, time series prediction, and speech recognition [9].

MOTIVATING FACTORS OF ASAG SYSTEM

The motivation behind the work lies in addressing the challenges faced in grading short answers manually, which can be time-consuming, subjective, and prone to inconsistencies. The primary motivation behind this research is to leverage advanced deep learning techniques, particularly BERT-based models [2], to automate the grading of short answers efficiently and accurately in educational settings, MOOCs, and online exam platforms [2]. By automating this process, the paper aims to improve the efficiency of grading tasks, ensure consistency in evaluation, scale grading capabilities to accommodate larger volumes of student responses, and enhance the overall accuracy of assessment.

The main contribution of the work involves developing and evaluating a novel BERT-based deep neural network model specifically tailored for automatic short-answer grading. The incorporation of a semantic refinement layer in the BERT-based model to deepen the understanding of answer texts and improve feature generalization abilities within the domain. This enhancement aims to promote a deeper semantic understanding of student responses [35] and enhance the model's ability to generalize features for accurate grading. This model is designed to address the challenges associated with manual grading by providing a more objective and scalable solution for evaluating student responses in educational settings, MOOCs, and online exam platforms [7]. Through the implementation and fine-tuning of the BERT model [21], the work aims to demonstrate the potential of deep learning techniques in revolutionizing the assessment process and advancing intelligent tutoring systems and automated grading mechanisms in educational environments.

RELATED WORKS

Automatic text scoring using neural networks (D. Alikaniotis, H. Yannakoudakis, and M. Rei 2016) involves leveraging artificial neural network architectures to assess the quality or score of written text. Here's a simplified overview of the process. Collect and preprocess a labeled dataset containing examples of text along with corresponding scores or grades. Convert the textual data into a suitable format for neural network input. This often involves techniques like tokenization and embedding to represent words or phrases numerically. Design a neural network architecture that can effectively capture the features and patterns in the input text data. This may include recurrent neural networks (RNNs), long short-term memory networks (LSTMs), or more advanced models like BERT integrate the trained neural network into the text scoring system, allowing it to automatically evaluate new or unseen text based on the learned patterns during training. Neural networks can capture complex, non-linear relationships within the text, allowing them to model intricate patterns and dependencies that may be challenging for traditional scoring methods. But Neural networks are often considered black-box models, making it difficult to interpret how the model arrives at a particular score. Lack of interpretability can be a concern in educational settings where transparency is crucial.

The paper "BERT: Pre-training of deep bidirectional transformers for language understanding" by (Jacob Devlin and his colleagues, published in 2018), introduces BERT (Bidirectional Encoder Representations from Transformers), a groundbreaking natural language processing (NLP) model. BERT is designed to capture bidirectional context for each word in a sentence, unlike previous models that only considered left-to-right or right-to-left context. This bidirectional approach allows BERT to understand the relationships and dependencies between words in both directions [13], [20]. BERT is pre-trained on a massive amount of unlabeled text data. The pre-training objective involves predicting missing words within sentences, both masked words and randomly selected words. This unsupervised pre-training helps BERT learn contextual representations. This encourages the model to understand the context and relationships between the words. BERT's bidirectional architecture captures context from both left and right directions, allowing it to better understand the nuances and relationships between words in a sentence. BERT generates contextualized word representations. The Transformer architecture used in BERT facilitates parallelization during training, making it computationally efficient and scalable. But the training and fine-tuning of BERT can be computationally intensive, requiring substantial resources. This may pose challenges for researchers or organizations with limited access to high-performance computing infrastructure.

Automatic short answer grading using semantic spaces and MaLSTM (C. N. Tulu, O. Ozkaya, and U. Orhan 2021) the "SemSpace" typically refers to a semantic space, which is a mathematical representation where words or phrases are placed based on their semantic similarity. Sense vectors may indicate that the model considers different senses or meanings of words. Utilizing semantic space and sense vectors helps capture the semantic relationships between words, enhancing the understanding of the content. MaLSTM is a model architecture that combines Matching Networks and Long Short-Term Memory (LSTM) networks. LSTM is effective for handling sequential data, and in the context of short answer grading, MaLSTM likely aims to capture the similarity or matching degree between the student's answer and a reference answer. The overall goal of the paper seems to be improving the automatic grading of short answers. Using SemSpace sense vectors suggests a focus on semantic understanding, enabling the model to recognize not only surface-level similarities but also semantic nuances in the responses. This model captures the nuances and multiple senses of words in short answers. MaLSTM combines Matching Networks with LSTM, enabling the model to perform contextual matching. Improved Relevance. The use of semantic vectors and MaLSTM likely contributes to a more nuanced relevance assessment. It has some disadvantages they are, Utilizing semantic vectors and MaLSTM may require significant computational resources, making the approach computationally intensive, especially during training and fine-tuning. Success in automatic grading often depends on the availability of large and diverse datasets for training.

The combination of Deep Neural Networks (DNNs) and Item Response Theory (IRT) in Automatic Short Answer Grading (M. Uto and Y. Uchida 2020) represents a hybrid approach that leverages the strengths of both techniques. Here's an explanation of how these elements are typically integrated. DNNs are used for their capability to learn complex patterns and representations from data. In ASAG, DNNs can be employed to automatically extract features and representations from short answers. These neural networks are trained on a large dataset of short answers and corresponding grades. The initial layers of the DNN are responsible for feature extraction. These layers capture various aspects of the input short answers, learning to represent them in a high-dimensional feature space. This process is essential for capturing the nuanced information present in language. Deep Neural Networks (DNNs) contribute to the model's ability to understand the semantic content of short answers,[35] capturing contextual information and relationships between words. DNNs excel at automatically extracting relevant features from input data. In the context of Automatic Short Answer Grading (ASAG), this capability allows the model to discern important patterns and characteristics in student responses. The integration of Deep Neural Networks (DNNs) and Item Response Theory (IRT) can result in a computationally complex model, requiring substantial computational resources for both training and inference. The use of DNNs can make the model less interpretable. Tuning parameters for both the DNN and IRT components can be challenging.

Automatic short answer grading by encoding student responses via a graph convolutional network by (H. Tan, C. Wang, Q. Duan, Y. Lu, and R. Li, 2020). In this work each word or phrase in the student response becomes a node in the graph. The relationships between these nodes, such as co-occurrence or semantic connections, are modeled as edges. A Graph Convolutional Network employs an embedding layer to convert the nodes (words or phrases) into high-dimensional vectors, capturing their semantic meaning. Convolutional Operations: GCN applies convolutional operations on the graph to aggregate information from neighboring nodes, allowing the model to understand the contextual relationships between words. Scoring and Classification: After the convolutional operations, the model generates a representation of the student response. This representation is then used for scoring or classification tasks, determining the correctness or quality of the answer. Training on Labeled Data: The GCN model is trained on a dataset with labeled examples, where the correct grades or classifications are provided for various student responses. This enables the model to learn the patterns and features associated with correct or incorrect answers. GCNs excel at capturing contextual relationships within student responses, allowing for a more nuanced comprehension of the meaning and coherence in answers. GCNs heavily rely on labeled data for training[10]. In cases where there are limited or biased training data, the model's performance may be hindered. Implementing and training GCNs can be computationally expensive and resource-intensive, particularly when dealing with large datasets or complex graph structures.

Automatic short answer grading and feedback using text mining methods (N. Seuzen, A. N. Gorban, J. Levesley, and E. M. Mirkes 2020) involves the application of computational techniques to analyze and assess students' written responses. The student's written response is subjected to text mining techniques, including preprocessing steps like removing stop words, stemming, or lemmatization, to prepare the text for analysis. Relevant features, such as word frequencies, n-grams, or other linguistic patterns, are extracted from the processed text. These features serve as input for the grading model. The trained model is applied to grade new student responses. It classifies the answers into categories indicating correctness or quality, providing an automated grading score. Based on the grading results, tailored feedback is generated for the student. Text mining identifies specific strengths and weaknesses in the response, allowing for personalized and constructive feedback. Text mining methods provide a consistent and standardized approach to grading, subjective variations in assessment across different graders. Text mining methods [11] may struggle with handling ambiguous language or varied interpretations, leading to challenges in accurately grading answers that lack clarity. Grading short answers requires a nuanced understanding of natural language, and text mining models may struggle with the complexity of language nuances, idioms, or context.

Table 1 Comparison of existing works

AUTHOR NAME AND YEAR	TECHNIQUES USED	MERITS	DEMERITS
C. N. Tulu, M. Kaya, and M. Kocamanlar (2021)	SemSpace Algorithm, Word Sense Disambiguation(WSD) and MaLSTM Model	It enhances the accuracy. Improved efficiency. It captures semantic similarity.	It require a large amount of training data. Lack of generalizability and robustness.
Masaki Uto and Yuto Uchida (2020)	Item-Response Theory (IRT) model and LSTM	Improved Scoring accuracy. Automated feature extraction. It captures non-linear relations.	It requires a large amount of data. Complex implementation.
H. Tan, C. Wang, Q. Duan, Y. Lu, and R. Li. (2020)	Graph Convolutional Networks.	Improved Accuracy. Efficient grading.	Complex Implementation. It can still be complex to understand.
N. Siiuzen, A. N. Gorban, J. Levesley, and E. M. Mirkes. (2020)	Text mining methods	It gives quick feedback. Objective evaluation of student responses.	Complex Implementation. May not capture a deeper conceptual understanding
J. Devlin, M. W. Chang, K. Lee, and K. Toutanova (2019)	BERT	It capture both left and right context from the input. It is pre-trained on large scale.	Requires large amount of computational resources.
D. Alikanioti, H.Yannakoudakis, and M. Rei (2016)	LSTM and C&W embedding	Enhanced interpretability. Superior performance.	Large and diverse datasets are often required. Lack of Context Understanding.

The above-mentioned deep learning methods achieve grading and scoring in an end-to-end manner, but they require a large amount of labeled corpus for training their model, which is what most ASAG corpora lack. To solve this problem, various pretrained transfer learning models, such as embeddings from language models (ELMo), GCN [10], BERT, RoBERT [5] generative pretrained transformer (GPT), and GPT-2, are applied to the ASAG task. Among them, BERT [13], [16], [17], [20] is especially outstanding and has achieved state-of-the-art grading results.

PROPOSED WORK

The system model proposed in this work "Automatic Short-Answer Grading" integrates a BERT-based deep neural network framework [20] with bidirectional LSTM (Bi-LSTM) network to automate and enhance the process of short-answer grading. By dynamically encoding answer texts using the BERT model and refining semantics through the Bi-LSTM network [9], [13], the system achieves a deep understanding of student responses, leading to accurate and context-aware grading. Leveraging transfer learning and pre-trained models, the system overcomes the limitations of small labeled corpora in ASAG tasks and demonstrates state-of-the-art performance on benchmark datasets. Designed for end-to-end grading, the system streamlines the evaluation process in educational environments, offering a scalable and efficient solution for evaluating student responses in intelligent tutoring systems, MOOCs, and online exam platforms.

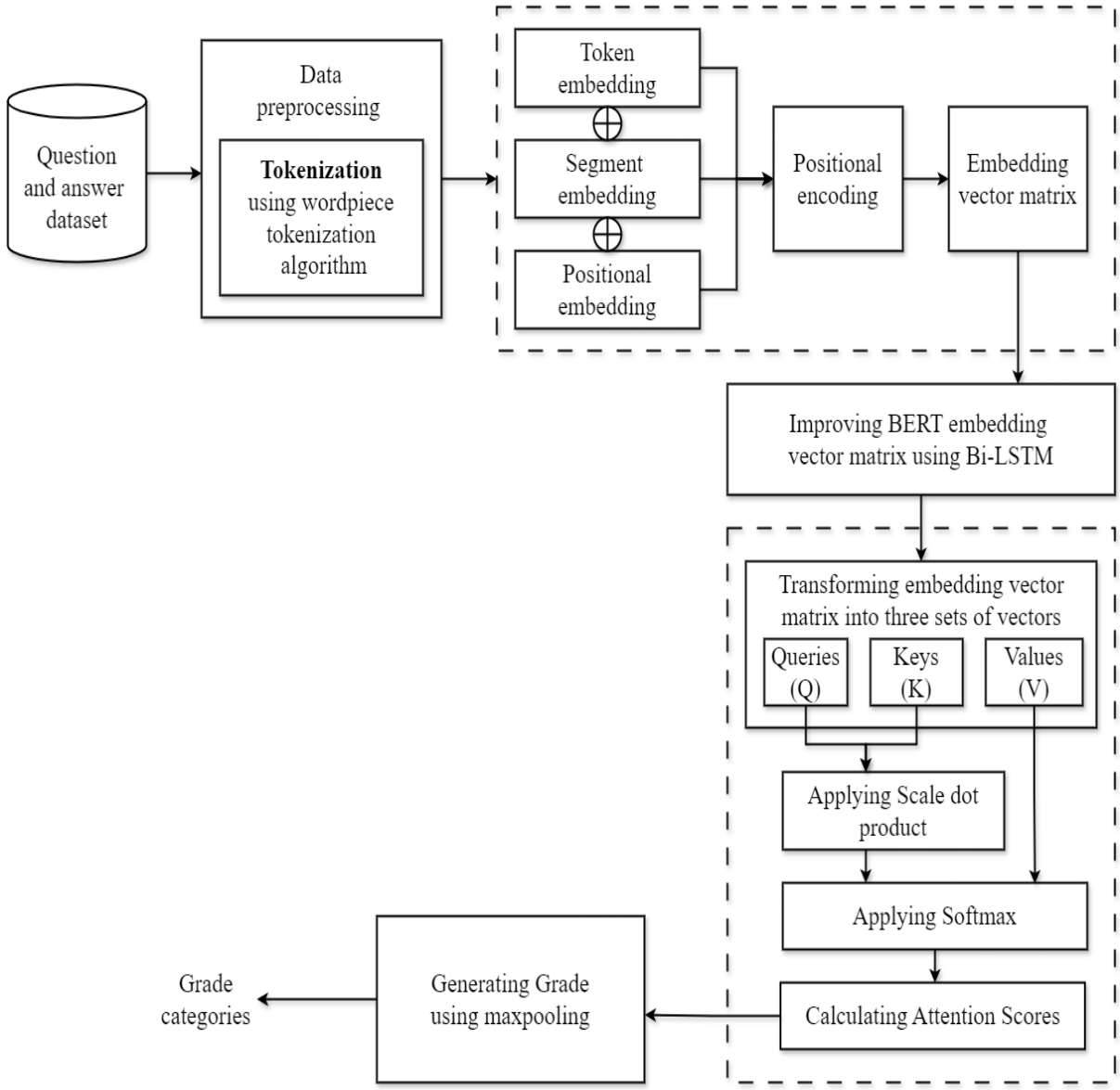

Figure 1: Proposed System architecture

BERT ENCODING

The process involves initializing all parameters of the BERT model [13] and then jointly fine-tuning these parameters along with other layers in the model. The input sequence for the fine-tuned BERT layer consists of a sentence pair comprising the student's answer and the reference answer. The input to a BERT model [17] is tokenized text, which is converted into embeddings which include token embeddings, segment embedding, positional embedding and positional encoding. Token embeddings represent the meaning of individual words or sub words as a vector. Positional encodings encode the position of tokens in the sequence using sinusoidal functions to enable understanding of sequential order. It learns to represent a token's position in a sequence as a vector which is known as a hidden state. For example, the position of the first token in a sequence is represented by a (learned) vector $\boldsymbol{Wp}$ [:, 1], the position of the second token is represented by another (learned) vector $\boldsymbol{Wp}$ [:, 2], etc. The purpose of the positional embedding is to allow a Transformer to make sense of word ordering; in its absence the representation would be permutation invariant and the model would perceive sequences as "bags of words" instead.

BERT POSITIONAL ENCODING ALGORITHM

Input: $\boldsymbol{l} \in [\boldsymbol{l_{max}}]$, position of a token in the sequence.

Step 1: Extract the positional encoding of a token's position in a sequence.

Step 2: Takes a position $\boldsymbol{l}$ and a positional encoding $\boldsymbol{Wp}$ as inputs.

$$\boldsymbol{Wp}\,[2i-1, t] = \sin(t/l_{max}^{2i/d_e})$$

$$\boldsymbol{Wp}[2i, t] = \cos(t/l_{max}^{2i/d_e}),$$

Step 3: Extracting the column vector corresponding to the given position $\boldsymbol{l}$.

Step 4: Return $\boldsymbol{ep}$ as the output, representing the positional embedding vector matrix for the token at position $\boldsymbol{l}$.

Output: $\boldsymbol{ep} \in \mathbb{R}^{de}$, the embedding vector-matrix representation of the position.

Bi-LSTM NETWORK

In the semantic refinement layer of the ASAG system, a bidirectional Long Short Term Memory (LSTM) network with position information [4] is employed to refine the semantics of the BERT outputs. The Bi-LSTM network extracts fine global context[24] for the BERT outputs to the hidden states of the BERT model. By combining these components and utilizing complex gate structures in the Bi-LSTM network improves the hidden states of the BERT output. The aim is to enhance the system's ability to understand and evaluate short answers accurately. At the end of each sub layer, layer normalization [30] is applied to normalize the activations.

BI-LSTM ALGORITHM

Input: $\boldsymbol{ep}$ as Embedding vector matrix

Step 1: Calculate the forget gate activation vector

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)$$

Step 2: Calculate the input gate activation vector:

$$i_t = \sigma(w_i \cdot [ht^{-1}, x_t] + b_i)$$

Step 3: Calculate the candidate cell state vector:

$$\tilde{c}_t = \tanh(W_c.[h_{t-1}, x_t] + b_c)$$

Step 4: Update the cell state:

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

Step 5: Calculate the output gate activation vector:

$$0_t = \sigma(W_o.[h_{t-1}, x_t] + b_o$$

Step 6: Calculate the new hidden state vector

$$h_t = O_t \odot \tanh(c_t)$$

Output: h_t as Improved vector matrix.

ATTENTION MECHACNISM

After receiving the output from the Bi-LSTM network, the attention layer in a model like BERT [25] processes this output to capture important contextual information and relationships within the sequence. This process is achieved by using a multi-head attention mechanism [15]. The contributions of these contexts to the fused semantics are automatically adjusted by a weight matrix. After this combination process, the fused semantic representation is processed by layer normalization [30], [35] before being fed into the max-pooling layer for further evaluation. The attention layer plays a crucial role in integrating different levels of contextual information to enhance the system's ability to evaluate short answers accurately.

ATTENTION MECHANISM ALGORITHM

Input: $\boldsymbol{X} \in \mathbb{R}^{dx l_x}$, $\boldsymbol{Z} \in \mathbb{R}^{dz \times l_z}$, vector representations of primary and context sequence.
Parameters: W$\boldsymbol{qkv}$ consisting of:

$W_q \in \mathbb{R}^{d_{attn} \times d_x}, b_q \in \mathbb{R}^{d_{attn}}$

$W_k \in \mathbb{R}^{d_{attn} \times d_z}, b_k \in \mathbb{R}^{d_{attn}}$

$W_v \in \mathbb{R}^{dout \times d_z}, b_v \in \mathbb{R}^{dout}$,

Hyperparameters: Mask∈{0,1}{0,1}$^{lz \times l_x}$↑(3)

Step 1: Compute Query, Key, and Value Matrices such as Q, K and V.

$Q \leftarrow W_q X + b_q 1^T$ ⟦Query $\in \mathbb{R}^{dattn \times l_x}$⟧

$K \leftarrow W_k Z + b_k 1^T$ ⟦Key $\in \mathbb{R}^{dattn \times l_z}$⟧

$V \leftarrow W_v Z + b_v 1^T$ ⟦$Value \in \mathbb{R}^{dout \times l_z}$⟧

Step 2: Compute the scale dot product of K and the transpose of Q to get the attention scores matrix S, representing the similarity between queries and keys.

$S \leftarrow K^T Q$ ⟦Score $\in \mathbb{R}^{lz \times l_x}$⟧

Step 3: Apply the softmax function to the rows of S scaled by attention to compute the attention weights.

Step 4 : To compute the output multiply the attention weights by V to get the output matrix which aggregates information from the context sequence based on attention weights.

$\boldsymbol{X}^{(h)} = V \cdot \text{softmax}(S/\sqrt{d_{attn}})$

Output: $\boldsymbol{X}^{(h)} \in \mathbb{R}^{dout \times l_x}$, updated representations of tokens in $\boldsymbol{X}$, folding in information from tokens in $\boldsymbol{Z}$.

MAX-POOLING

In the max-pooling layer of the ASAG system, a max-pooling operation is performed on the semantic representation generated by the attention mechanism of the BERT model [25]. During the max-pooling operation in the ASAG system, the semantic representation is processed to extract the most relevant and

significant features that are crucial for evaluating the student's answer. In the Prediction Layer, a max-pooling operation is performed on the updated representation of tokens to obtain the final semantic representation Z for the answer pair (q, p). The max-pooling operation selects the maximum value from the fused semantic representation across a specific dimension or window. By selecting the maximum value from the set of values in the representation, the max-pooling operation helps in capturing the most salient information that contributes to the overall understanding and assessment of the answer pair (question, student's answer). To calculate the probability of each grade category, $\boldsymbol{Z}$ is fed into a linear transformation followed by a softmax function. The output of the max-pooling operation is the final semantic representation Z. The max-pooling operation helps in capturing the most relevant and significant features from the fused semantic representation for further processing and grading of the short answer pair. The result of the max pooling operation is a final semantic representation. Finally, grade categories are generated for student's answer using a regression task.

MAX-POOLING ALGORITHM

Input: $\boldsymbol{X}^{(h)}$ as updated representation of tokens.

Step 1: Perform a max-pooling operation on the $\boldsymbol{X}^{(h)}$ attention scores.

$\boldsymbol{Z}$ = Max-pooling($\boldsymbol{X}^{(h)}$)

Step 2: To calculate the probability of each grade category, $\boldsymbol{Z}$ is fed into a linear transformation followed by a softmax function.

$$\boldsymbol{G = MZ+b}$$

$$\boldsymbol{P(y|Z)} = \frac{\boldsymbol{exp}\ (\boldsymbol{G_y})}{\sum_i^{\boldsymbol{dy}} \boldsymbol{exp}(\boldsymbol{G_i})}$$

where $\boldsymbol{M}$ is the representation matrix of grade categories, b is a bias vector, $\boldsymbol{G_y}$ is the number of grade categories, $\boldsymbol{G}$ is the vector of scores.

Output: $\boldsymbol{P(y|Z)}$ denotes the predicted probability of grade category $\boldsymbol{y}$.

PERFORMANCE EVALUATION

There are two measurement scales have been used for this proposed work.

Cohen's Kappa: Cohen's kappa is an effective measure, used for interrater reliability between items. The formula is as follows:

$$\boldsymbol{K} = \frac{\boldsymbol{Pr(a)-Pr(e)}}{\boldsymbol{1-Pr(e)}}$$

Table 2 Cohen's Kappa Description

Item	Description
Pr(a)	A total number of the percentage that is consistent between raters.
Pr(e)	If a percentage of total measurement changes, then the number of measurements changes between raters.

The short answer grading has been evaluated in Roshan Tara School, Mehrabpur, Sindh Pakistan. Out of 78 students, 60 students in 10th grade participated in the quiz. Out of 60 students, 52 responses were complete. For the assessment, we have involved a human expert to evaluate the performance and grade the assessment. Later, we compared the human assigned grades with a deep learning model that grades the answers automatically. However, the measurement provides consistency between the two raters. Moreover, Cohen's kappa scale is only applied to qualitative measurement. So for that, the data must be in a categorical form. The experiment shows the consistent equivalency between raters. Furthermore, we have evaluated the categories with the help of a confusion matrix to check the accuracy of the model.

Table 3 Comparison of grading between manual and automatic grading

Student no	Question no	Automatic short answer grading	Manual grading
1	1	Incorrect	Incorrect
1	2	Correct	Correct
1	8	Correct	Incorrect
2	1	Correct	Correct
2	8	Correct	Correct
3	5	Correct	Correct
4	1	Correct	Correct
52	1	Correct	Correct
3	8	Incorrect.	Correct
4	8	Correct	Correct
52	8	Correct	Correct
52	1	Correct	Correct

Table 4 Values of consistency between grades.

Grading		Correct	Incorrect (Incomplete or contradictory)
Manual grading	Correct	77%	4%
Automatic grading	Incorrect	3%	15%

$$K= \frac{(0.77*0.15)-(0.81*0.79)+(0.19+0.21)}{1-(0.81*0.079)+(0.19+0.21)}$$

K=0.77%

Confusion matrix: This matrix is used to check the performance of the machine learning algorithm. The confusion matrix represents the predictions and actual conditions generated by the algorithm. Following are the performance evaluator's parameters of the confusion matrix.

- True Positive: Predicts positive classes as positive classes
- True Negative: Predicts negative class as negative class
- False Positive: Predicts the negative class as positive class
- False Negative: Predicts positive class as negative class

Table 5 Prediction type

	Predicted Positive (PP)	Predicted Negative (PN)
True	True Positive (TP) 77%	True Negative (TN) 15%
False	False Positive (FP) 4%	False Negative (FN) 3%

Table 6 Definition of parameters included in the confusion matrix

Factors	Definition
Precision	"The ratio of accurately predicted positive observations to total expected positive observations".
Recall	"The ratio of actual positive instances that were predicted properly".
F1 Score	"F1 score is the harmonic mean between Precision and Recall".
Accuracy	"The number of true negatives (TN) and true positive (TP) samples divided by the total number of samples."

$$Precision = \frac{TP}{TP+FP} = \frac{77}{77+4} * 100 = 0.95\%$$

$$Recall = \frac{TP}{TP+FN} = \frac{77}{77+3} * 100 = 0.96\%$$

$$F1 = \frac{Precision * Recall}{Precision+Recall} * 2 = 0.96\%$$

$$ACC = \frac{TP+TN}{TP+TN+FP+FN} = 0.93\%$$

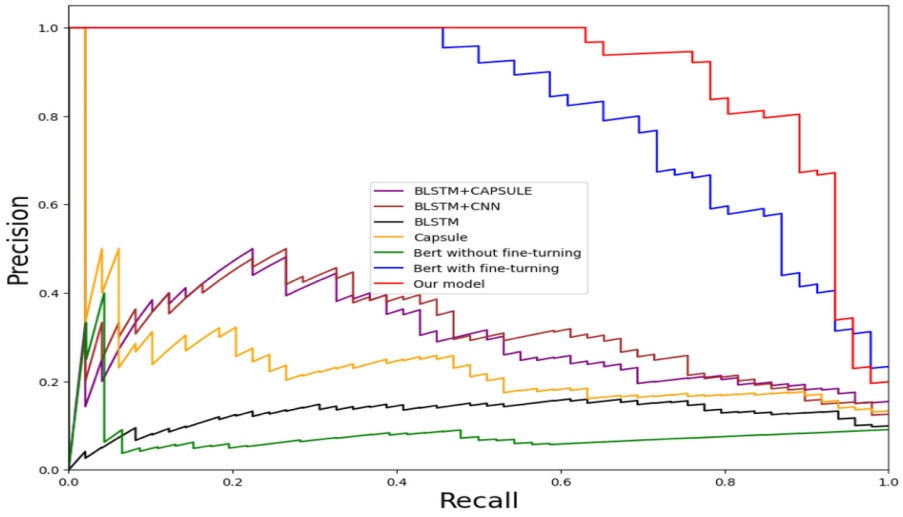

Fig 2 Comparison on Precision and recall with other models

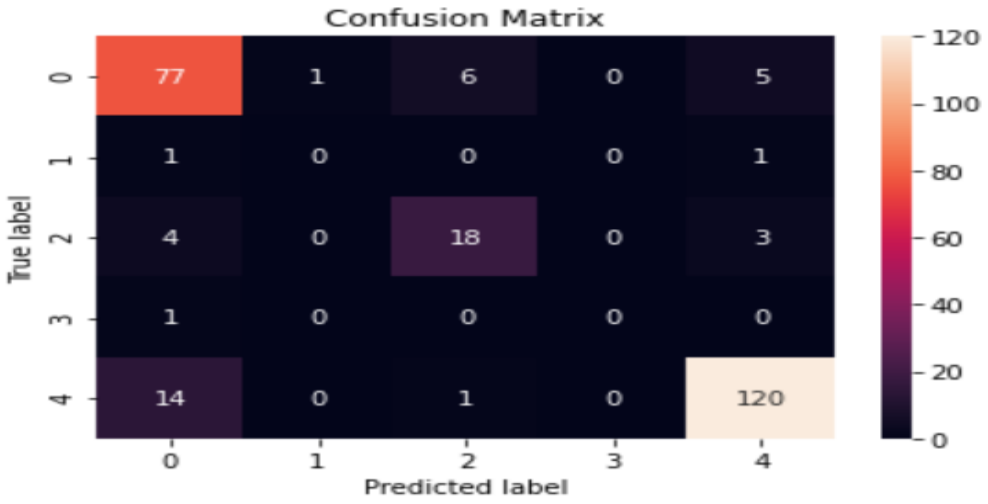

Fig 3 Confusion matrix

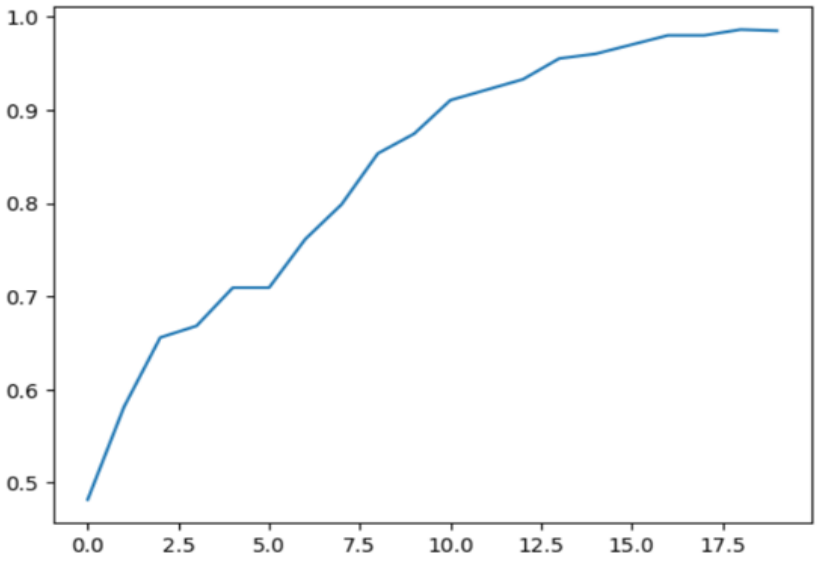

Fig 4 ASAG accuracy

Performance Analysis

ASAG using BERT and Bi-LSTM network obtained the best average performance. In most cases, it achieves the best accuracies compared to the other models. In Table 7 the ablation result of w/o refinement shows that our semantic refinement layer significantly improves the grading accuracy of the BERT model on the Mohler dataset and the SemEval-2013 UQ subset [10]. As with the UQ subset in the SemEval-2013 dataset, the testing questions and training questions on the Mohler dataset [26], [35] originate from the same domain. Therefore, this ablation result means that our semantic refinement layer can significantly improve the generalization ability of the BERT model to the features in the domain. The ablation result of w/o Bi-LSTM with BERT shows that after directly replacing Bi-LSTM with the output of BERT, the grading accuracy of the model on two datasets has decreased to varying degrees. This means that the complex gate structures in the Bi-LSTM network can extract more refined context information from the output of the BERT model, just as it performs in other natural language processing tasks, such as Named Entity Recognition.

Table 7 Accuracy obtained from different models

Different models	Unseen answer	Unseen questions	Unseen domain
Our model	76.5	69.2	66.0
CNN	70.1	68.8	63.4
BERT without Bi-LSTM	72.3	65.7	65.7
Without multihead	70.9	68.4	64.5

Table 8 Comparison of different models on Mohler dataset

Different models	MAE	RMSE	Pearson
Our model	0.248	0.827	0.897
CNN	0.309	0.821	0.884
BERT without Bi-LSTM	0.254	0.833	0.884
Without multihead	0.483	1.341	0.751

From the result we can observe that our semantic refinement layer can significantly improve the generalization ability of the BERT model to the features in the domain. Lower is better for MAE and RMSE; higher is better for accuracy and Pearson's r. The ablation result of w/o Bi-LSTM with BERT shows that after directly replacing Bi-LSTM with the output of BERT, the grading accuracy of the model on two datasets has decreased to varying degrees. This means that the complex gate structures in the Bi-LSTM network can extract more refined context information.

CONCLUSION

In this article, we proposed a novel BERT-based deep neural network model for ASAG. Through extensive experimental comparison, this article reveals the following theoretical implications:

- Word-embedding-based self-training neural networks, such as CNN and LSTM cannot play an important role in ASAG tasks with small datasets.
- The BERT model, which has a deep transformer coding structure and is pretrained on a large-scale corpus, can perform better by simply fine-tuning in the ASAG task.
- On the fine-tuned BERT model, the LSTM can extract more fine semantics to deepen the BERT model's understanding of the answer text and further improve the feature generalization ability of the BERT model in the domain.

REFERENCES

1. Xinhua Zhu, Han Wu, and Lanfang Zhang, "Automatic Short-Answer Grading via BERT-Based Deep Neural Networks", IEEE TRANSACTIONS ON LEARNING TECHNOLOGIES, vol.15, NO. 3, pp.364-371, JUNE 2022.
2. W. Song, Z. Wen, Z. Xiao, and S. Park, "Semantics perception and refinement network for aspect-based sentiment analysis," Knowl.-Based Syst., vol. 214, Art. no. 106755, 2021.
3. M . T. Nguyen, D. T. Le, and L. Le, "Transformers-based information extraction with limited data for domain-specific business documents," Eng. Appl. Artif. Intell., vol. 97,Art. no. 104100, 2021.
4. C. N. Tulu, O. Ozkaya, and U. Orhan, "Automatic short answer grading with SemSpace sense vectors and MaLSTM," IEEE Access, vol. 9, pp. 19270–19280, 2021.
5. W. X. Liao, B. Zeng, X. W. Yin, and P. F. Wei, "An improved aspect category sentiment analysis model for text sentiment analysis based on RoBERTa," Appl. Intell., vol. 51, pp. 3522–3533, 2021.
6. I. Katsaris and N. Vidakis, "Adaptive e-learning systems through learning styles: A review of the literature," Adv. Mobile Learn. Educ. Res., vol. 1, no. 2, pp. 124–145, 2021.
7. N. Khodeir, "Bi-GRU urgent classification for MOOC discussion forums based on BERT," IEEE Access, vol. 9, no. 1, pp. 58243–58255, Apr.2021.
8. G. G. Smith, R. Haworth, and S. Zitnik, "Computer science meets education: Natural language processing for automatic grading of open-ended questions ineBooks," J. Educ.Comput. Res., vol. 58, no. 7, pp.1227–1255, 2020
9. M. Uto and Y. Uchida, "Automated short-answer grading using deep neural networks and item response theory," in Proc. Int. Conf. Artif. Intell. Educ., pp. 334– 339, 2020.
10. H. Tan, C. Wang, Q. Duan, Y. Lu, and R. Li, "Automatic short answer grading by encoding student responses via a graph convolutional network," Interact. Learn. Environ., vol. 2020, pp. 1–15, 2020.
11. N. Seuzen, A. N. Gorban, J. Levesley, and E. M. Mirkes, "Automatic short answer grading and feedback using text mining methods," Procedia Comput. Sci., vol. 169, no. 2020, pp. 726–743, 2020.
12. A . Sahu and P. K. Bhowmick, "Feature engineering and ensemble-based approach for improving automatic short-answer grading performance," IEEE Trans. Learn. Technol., vol. 13, no. 1, pp. 77–90, Jan.–Mar. 2020.
13. C. Leon and F. Anna, "Investigating transformers for automatic short answer grading," in Proc. Int. Conf. Artif. Intell. Educ.,pp. 43–48, 2020.
14. Y. Zhang, C. Lin, and M. Chi, "Going deeper: Automatic short-answer grading by combining student and question models," User Model. UserAdapted Interact., vol. 30, no. 1, pp. 51–80, 2020.
15. H. Yang, B. Zeng, J. Yang, Y. Song, and R. Xu, "A multi-task learning model for Chinese-oriented aspect polarity classification and aspect term extraction," Neurocomputing, vol. 419, pp. 344–356, 2020.
16. J. Zhou, X. Huang, Q. Hu, and L. He, "SK-GCN: Modeling syntax and knowledge via graph convolutional network for aspect-level sentiment classification," Knowl.-Based Syst., vol. 205, no. 3, Art. no. 106292, 2020
17. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. Annu. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol., pp. 4171–4186, 2019.
18. Y. Liu et al., "RoBERTa: A robustly optimized BERT pretraining approach," arXiv:1907.11692v1, 2019.
19. T. Liu, W. Ding, Z. Wang, J. Tang, G. Huang, and Z. Liu, "Automatic short answer grading via multiway attention networks," in Proc. Int. Conf. Artif. Intell. Educ.,pp. 169–173, 2019.

20. C. Sung, T. I. Dhamecha, and N. Mukhi, “Improving short answer grading using transformer-based pre-training,” in Proc. Conf. Artif. Intell. Educ., pp. 469–481, 2019.
21. B. Yang, L. Wang, D. F. Wong, L. S. Chao, and Z. Tu, “Convolutional self-attention networks,” in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol., vol. 1, pp. 4040–4045, 2019.
22. S. Saha, T. I. Dhamecha, S. Marvaniya, R. Sindhgatta, and B. Sengupta, “Sentence level or token level features for automatic short answer grading?: Use both,” in Proc. Int.Conf. Artif. Intell. Educ., pp. 503–517, 2018.
23. S. Marvaniya, S. Saha, T. I. Dhamecha, P. Flotz, R. Sindhgatta, and B. Sengupta, “Creating scoring rubric from representative student answers for improved short answer grading,” in Proc. 27th ACM Int. Conf. Inf. Knowl. Manage.,pp. 993–1002, 2018.
24. W. Zhao, J. Ye, M. Yang, Z. Lei, S. Zhang, and Z. Zhao, “Investigating capsule networks with dynamic routing for text classification,” in Proc. Conf. Empirial Methods Natural Lang. Process., pp. 43–48. 2018.
25. A. Vaswani et al., “Attention is all you need,” in Proc. 31st Conf. Neural Inf. Process. Syst., pp. 5998–6008,2017.
26. S. Kumar, S. Chakrabarti, and S. Roy, “Earth movers distance pooling over Siamese LSTMs for automatic short answer grading,” in Proc. Int. Joint Conf. Artif.Intell., pp. 2046–2052, 2017.
27. B. Riordan, A. Horbach, A. Cahill, T. Zesch, and M. L. Chong, “Investigating neural architectures for short answer scoring,” in Proc. 12th WorkshopInnov. Use NLP Building Educ.Appl., pp. 159–168, 2017.
28. D. Alikaniotis, H. Yannakoudakis, and M. Rei, “Automatic text scoring using neural networks,” in Proc. 54th Annu. Meeting Assoc. Comput Linguistics, vol. 1, pp. 715–725, 2016.
29. M. A. Sultan, C. Salazar, and T. Sumner, “Fast and easy short answer grading with high accuracy,” in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol., pp. 1070–1075, 2016.
30. J.L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,”, arXiv:1607.06450, 2016.
31. S. Drissi and A. Amirat, “An adaptive e-learning system based on student’s learning styles: An empirical study,” Int. J. Distance Educ. Technol., vol. 14, no. 3, pp. 34–51, 2016.
32. G. Jorge-Botana, J. M. Luzon, I. Gomez-Veiga, and M. Cordero, “Automated LSA assessment of summaries in distance education: Some variables to be considered,” J. Educ. Comput. Res., vol. 52, no. 3, pp. 341–364, 2015.
33. M. Heilman and N. Madnani, “ETS: Domain adaptation and stacking for short answer scoring,” in Proc. Joint Conf. Lexical Comput. Semantics, vol. 2, pp. 275–279, 2013.
34. M. O. Dzikovska et al., “SemEval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge,” in Proc. 2nd Joint Conf. Lexical Comput. Semantics,vol. 2, pp. 263–274, 2013.
35. M. Mohler, R. Bunescu, and R. Mihalcea, “Learning to grade short answer questions using semantic similarity measures and dependency graph alignments,” in Proc. Annu. Meeting Assoc. Comput. Linguistics, Hum. Lang. Technol., pp. 752–762, 2011.