





Evaluation of Short Answers Using Domain Specific Embedding and Siamese Stacked BiLSTM with Contrastive Loss

Shweta Patil^{*}, Krishnakant P. Adhiya^{*}

Department of Computer Engineering, SSBT's College of Engineering and Technology, Bambhori, Jalgaon 425001, Maharashtra, India

Corresponding Author Email: shwetampatil83@gmail.com

<https://doi.org/10.18280/ria.370320>

ABSTRACT

Received: 1 April 2023

Accepted: 21 May 2023

Keywords:

automated short answer grading, Siamese neural network, BiLSTM, LSTM, stacked BiLSTM, contrastive loss, Pearson correlation, RMSE

Automatic short answer evaluation is the most complex task to perform as compared to evaluation of multiple choices and true or false type questions. Short descriptive answer tries to capture the overall knowledge gained by student related to the course, his remembering and presentation capabilities of the same. But sometimes the evaluation of such short descriptive answer becomes cumbersome and time consuming. So in this study, we are trying to address the issue of automated evaluation of short descriptive answers for Data Structures course by proposing a Siamese stacked Bidirectional LSTM neural network. The model utilizes the domain specific embedding generated by training gensim Word2Vec model on Data Structures domain. Domain specific embedding helps to identify the context relevant to domain which is difficult to understand in pre-trained embedding's due to ambiguity in words. The proposed model is trained using contrastive loss function and finally evaluation is made to determine whether student answer is correct or incorrect based on model answer provided by evaluator. The proposed architecture is tested using widely used Mohler's dataset and the results obtained are compared to baseline approaches using Pearson correlation coefficient and RMSE score. Also the proposed architecture is utilized on the dataset generated for specifically Data Structures course. For Mohler's dataset proposed framework achieves the best Pearson correlation value 0.668 compared to related baseline approaches. The results obtained has shown that proposed architecture is effective in investigating the relationship between complex descriptive sentences and performs the task of evaluation more similar to that of human evaluator.

1. INTRODUCTION

The task of Automated Short Answer Scoring is the most studied area in Natural Language Understanding domain. The automated short answer grading (ASAG) task mainly concentrates on recognizing the semantic similarity between the student and model answer provided by student and instructor respectively. Grading short answer manually is the most tedious work and moreover even prone to errors. Many a time's bias is observed in the task of short answer evaluation. To address this issue the possible solution is to automate the task of evaluation. The solution should be robust to recognize the surface level similarity between the concepts such that though student answer may not consist of exact words as in reference answer; but the underlying semantics between the texts may be similar which is one of the most challenging task of automated evaluation. The solution should recognize such situations and evaluate the answer accordingly without errors. Study has shown that neural networks [1-6] have proven to be boon in such task of Natural Language Processing.

The work in the field of ASAG is being carried out since decades. The comprehensive studies carried out by various researchers are summarized in the study [2, 3, 7]. It is evident that ASAG task can be broadly divided into two categories:

1. Statistical and lexical approach wherein hand-crafted features are defined such as length of student answer and reference answer, number of concepts common in both

student's answer and model answer, which are then utilized by various Machine Learning methods to classify or generate scores for respective student answers.

2. Semantic approach wherein the features are learned and are utilized by Deep Neural Network models for training and predicting the appropriate answer category or scores.

In this paper to tackle the task of ASAG we have proposed the deep learning-based approach. Specifically, we make the following contributions:

1.1 Siamese stacked bidirectional LSTM network

We have used Siamese stacked Bidirectional Long Short Term Memory based model. The Siamese neural network has proven to be efficient model for recognizing semantic similarity in semantic textual similarity task [8-11]. In Siamese model both student answer and model answer are being provided as the input which are processed in parallel, their level of similarity is being computed and based on the same appropriate label is predicted as an output through model. Instead of single Bidirectional Long Short Term Memory (BiLSTM) as in the study [12] stacked BiLSTM with 2-layers of BiLSTM to capture the complex contextual information in the input sequence is used. BiLSTM mechanism has shown promising results in numerous NLP task such as sentence similarity, sentiment analysis and many more.

1.2 Domain specific embedding

For the purpose of embedding, we have utilized the domain specific embedding instead of domain general embedding as utilized by Shweta and Adhiya [13]. Domain specific embedding helps to improve the system performance as it is trained on domain specific data which helps it to capture the patterns in domain that may not be present in domain general embedding. It also helps to increase the interpretability which means that it can easily give why certain words are being associated with each other in vector space. In the study [12], by utilizing pre-trained embedding the Pearson correlation coefficient reported is 0.655 whereas with domain specific embeddings correlation coefficient is 0.668.

1.3 Contrastive loss

Much of the research work carried out for ASAG task using deep neural network utilizes cross entropy loss to classify the student answer whether it is correct or incorrect; though it has shown good results but as it depends on number of samples used during training for each of the similar and dissimilar class it may produce bias results towards the majority class in imbalanced dataset. One solution to handle this bias issue may be to utilize more robust loss function which can deal with such imbalanced and produce the similarity between pair of sentences. One such loss function is contrastive loss. Contrastive loss is used in many similarity tasks such as image similarity or sentence similarity in Siamese neural network. It basically maximizes distance between dissimilar pairs while minimizes distance between similar pairs even though the pairs of input with varying similarity levels. Here we have trained the Siamese stacked BiLSTM model with contrastive loss to find similarity and dissimilarity between pair of student answer and model answer.

The proposed architecture is used in the evaluation of short answer for Data Structures course of Under Graduate class. The model determines whether the student answer is similar to the reference answer and finally predicts whether it is correct or incorrect. Detailed discussions on the stacked BiLSTM architecture with the previous research work, contrastive loss, and experimental results are discussed in the remaining part of paper.

2. LITERATURE REVIEW

The literature studied for the task of ASAG is majorly related to semantic textual similarity, Siamese neural networks and utilization of various neural network techniques for evaluation of short answers.

2.1 Siamese LSTM models

Authors [12] has proposed a Siamese Bidirectional LSTM neural network along with hand crafted features such as length of student answer, ratio of length of reference answer and student answer, number of words in answer, number of unique words in student answer. Author has tested the performance of proposed work on Mohler's dataset [6]. As the dataset has small number of samples, for training the neural network they have performed data augmentation. Once the data is ready they have preprocessed the data and later word embedding are computed using pre-trained Glove embedding which are later

fed into LSTM model for training. The output of LSTM layer is passed through fully connected layer with sigmoid activation function which then later is passed through dense layer which predicts the final score using linear activation function. The proposed work has shown Pearson correlation coefficient to be 0.655 which is considerably higher than state-of-art proposed system for ASAG task.

A hybrid approach using weighted fine-tuned BERT feature extraction with Siamese BiLSTM model is presented in the study [9]. The proposed approach is tested on Quora Question pair similarity dataset. Initially text features are extracted using BERT with weights which generate word embedding vector. The embedded vectors are then trained by using deep Siamese BiLSTM model and finally similarity scores are determined. The proposed architecture demonstrates 91% accuracy which is higher than state-of-art, proposed work in semantic textual similarity task. Research work in the study [14] is a novel approach wherein they have incorporated Siamese BiLSTM in combination with pooling layer based on the Sinkhorn distance between LSTM state sequences and support vector ordinal output layer instead of softmax. Author has tested the proposed approach on Mohler's dataset and SemEval dataset and has proved that the proposed work outperforms with higher accuracy as compared to baseline approaches.

In the study [10], the capabilities of Siamese neural network are demonstrated in the task of semantic textual similarity. In building blocks, they have tried to incorporate LSTM, BiLSTM, GRU, BiGRU, LSTM+Attention, GRU+Attention and has proved that the variants of GRU outperforms all other models with the Pearson correlation of 0.889 using the Manhattan distance formula as proposed in the study [11], wherein author has utilized Siamese LSTM neural network to compute semantic textual similarity between sentences and has given a Pearson correlation to be 0.8822. Both authors [2, 7] have tried to present the detailed study of ASAG task for various available dataset such as Mohler's dataset, SemEval, SRA and performance of various machine learning and deep learning models. In the study [7], author presented a study with detailed reporting of performance of major work carried in the domain of ASAG task.

2.2 Attention based models

For textual similarity in the study [8] attentive Siamese LSTM network is used for measuring semantic textual similarity. They proposed an architecture which consists of 5 layers: Input Layer to which sentence pairs are given. Embedding Layer which comprises of sentence pair words represented in lower dimension. Pre-trained embeddings are used to train on Wikipedia corpus using fasttext with 300-dimension. Hidden Layer which learns high level features of the sentences. Attention Layer which produces weight vectors. To this instead of just last hidden state of sentence pair authors proposed to use all hidden states as it will help to capture more information. Output Layer predicts the similarity between ranges of 1 to 5 for sentence pairs. They have tested their model performance on 3 tasks which include SemEval semantic relatedness task, Microsoft Research Paraphrase Identification task, and Chinese Mandarin and Tibetan corpora translated from SemEval task. They have also adopted BiLSTM model for comparative study. It was shown that attention based Siamese LSTM model shows substantial increase in Pearson correlation for all the 3 tasks as well as

over attention based Siamese BiLSTM network.

In the study [15], a hybrid attention model using CNN and BiLSTM is proposed for ASAG task which gives outstanding accuracy of 96%.

2.3 Paragraph embedding

Hassan et al. [16] has utilized various paragraph embedding techniques for short answer evaluation. They have computed paragraph embedding for respective answers using skip thought, InferSent and doc2vec. Later the embedding for both student and model answer are compared based on cosine similarity. The proposed approach has shown best result of 0.569 for Pearson correlation for doc2vec and 0.862 RMSE for InferSent. In the study [17], similar sentence embedding as in the study [16] is proposed using skip thought approach. After computing the vectors using skip thought for student answer and model answer, component wise product and absolute difference are computed and concatenated together. Finally, scores are predicted by training logistic linear classifier. Author has applied the proposed approach on Mohler's dataset and computed the Pearson correlation of 0.63 and RMSE of 0.91.

A comparative study of pertained transfer learning models such as ELMo, BERT, GPT and GPT2 is performed in the study [18] for ASAG task and has proved ELMo outperformed all baseline models with the Pearson correlation coefficient to be 0.485 as it has significant amount of domain data in pre-trained corpus as compared to other studied transfer learning models.

The study performed to compute the semantic similarity between student and model answer has shown that many researchers have incorporated various embedding vector along with numerous variants of neural network and shown competitive results. In this paper we are trying to make following contributions:

1. To utilize domain specific feature vectors generated through skipgram technique [13] instead of utilizing pre-trained embedding. As they help to capture the domain specific context more efficiently.

2. By utilization of domain specific embedding, train Siamese stacked BiLSTM network to generate the similarity or dissimilarity between both student answer and model answer using contrastive loss. Siamese stacked BiLSTM helps to capture the context for complex and compound sentences and contrastive loss assist in identifying the level of similarity between two sentences.

3. To evaluate the prediction using evaluation metrics such as Pearson correlation and RMSE and compare with the baseline models [12, 14, 16, 17].

3. MOTIVATION

Automated essay scoring and automated short answer scoring has been area of interest for many researchers since decades. The work carried out in this domain are categorized either by making evaluation based on lexical features or based on semantic features or by using Machine Learning technique or Deep Learning technique. Earlier work concentrates on mere existence of relevant terms in answer. Even they lack to recognize the sequence of information presented in sentence. With the advancement in Deep Learning and Natural Language Processing most of the complex tasks are made

more feasible to work with. Evaluation of short answer using these Deep Learning techniques will not only provide timely evaluation but will also be unbiased [19, 20].

The proposed approach not only concentrates on the existence of term in answer, but also recognizes the context behind the word. Our aim to carry out this research work is to perform the evaluation and reduce the burden on instructor so that they can devote their time in teaching learning task. In this paper we have tried to propose a Siamese stacked BiLSTM Neural network to evaluate student answers with reference model answers and categorize whether it is correct or incorrect. As ASAG task performed in the paper is related to Data Structures course of computer engineering; instead of utilizing the pre-trained domain general embedding we have tried to utilize the domain specific embedding so that more relevant context will be captured.

4. PROPOSED METHODOLOGY

4.1 Siamese neural network

In this paper, we have utilized Siamese neural network to compute the degree of similarity between student answer and model answer. The basic architecture of Siamese neural network utilized is shown below in Figure 1:

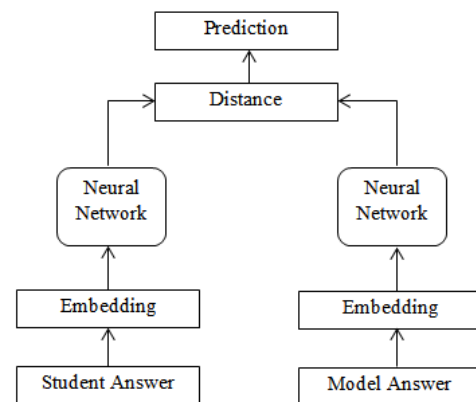


Figure 1. A Siamese architecture

Siamese neural network employs two identical neural network trained on same set of parameters for two different inputs and helps to identify the degree of similarities between both inputs. Here in this research work author has utilized stacked BiLSTM Siamese neural network to train over a pair of model and student answer and employed contrastive loss to compute distance between both pair of answers. Recent study has proven that the utilization of deep learning techniques has showed promising results in numerous natural language processing task.

4.2 Long short term memory

Many researchers have utilized LSTM, RNN and BiLSTM to address the issue of ASAG task. Recurrent Neural Network (RNN) faced the issue of vanishing gradient, which occurs when the gradients used to update network parameters become very small and make it very difficult for network to learn. LSTM helps to handle vanishing gradient issue of RNN by making use of more complex cell state that is updated and

controlled by three gating mechanisms: Input gate, Forget gate and Output gate.

These gates are implemented as sigmoid functions which control how much information is allowed to pass through cell state.

In addition to 3 gates it [21] has a set of learning parameters that are used to update cell state and update state.

$$\begin{aligned}
 i_t &= \sigma(W_i [h_{t-1}, X_t] + b_i) \\
 f_t &= \sigma(W_f [h_{t-1}, X_t] + b_f) \\
 g_t &= \tanh(W_g [h_{t-1}, X_t] + b_g) \\
 C_t &= f_t * c_{t-1} + i_t * g_t \\
 o_t &= \sigma(W_o [h_{t-1}, X_t] + b_o) \\
 h_t &= o_t * \tanh(C_t) \\
 y_t &= f(h_t)
 \end{aligned}$$

where, W_i, W_f, W_g, W_o are weights, b_i, b_f, b_g, b_o are bias for input gate, forget gate, cell state and output gate respectively. While h_t and h_{t-1} are hidden cell state at time t and $t-1$. Input, forget and output gate helps to control flow of information to and from the cell, it has a memory cell that can retain the information for long period of time.

4.3 Bidirectional LSTM

Bidirectional LSTM is an extension of standard LSTM which allows the network to process both backward and forward information about the sequence at each time step t . Basically BiLSTM [22] processes the information from left to right and right to left using two separate LSTM layers and then combines the outputs from these layers at each time step t . The hidden states of two LSTM layers at time step t denoted by h_{tf} and h_{tb} are concatenated to form a final hidden state h_t . The output of BiLSTM is computed by passing the hidden state at each time step t through a linear layer by an activation function as in standard Feed forward neural network.

In this research work we have utilized stacked BiLSTM as it has proved effective in capturing long term dependencies in sequential data and has achieved state-of-art performance in many NLP tasks [23]. Stacking multiple BiLSTM layers allows the network to learn increasingly complex representations of input with each layer build on the learnt representations of previous layer.

4.4 Contrastive loss

Here we have tried to use contrastive loss function to train model as it aims to learn similarity between pair of inputs. It basically encourages the model minimize distance between similar pair of sentences and maximize distance for dissimilar pair of sentences. The loss function computes a penalty; based on how far apart similar pairs are and how close dissimilar pairs are thereby encouraging models to learn useful representation of input space.

For every pair of sentence X and Y the Euclidian distance between their vector representation is calculated which is given as $d(f(x), f(y))$. To train the model the contrastive loss for pair of sentences is given as:

$$L(X, Y, t) = (1 - t) * d(f(x), f(y))^2 + t * \max(0, m - d(f(x), f(y)))^2$$

where, t is target label and m is margin hyper-parameter that controls minimum distance between dissimilar pairs. The

overall loss for the batch of answer pairs is the average of individual contrastive loss:

$$L_{batch} = (1 / N) * \sum (L(X_i, Y_i, t_i))$$

where, N is batch size, X_i, Y_i are i th sentence pair and t_i is corresponding similarity label.

4.5 Proposed architecture

The overall architecture of the network utilized in this study is shown in Figure 2. The shown architecture consists of following major building blocks:

1. **Input layer:** It consist of two sequences of model answer and student answer.
2. **Tokenization:** Each of the input sequences are tokenize into tokens.
3. **Embedding Layer:** Each of the tokens are mapped to vector representation using the domain specific embedding's generated.
4. **BiLSTM Layer:** BiLSTM layer above embedding layer captures the contextual information of the sequence of tokens in both forward and backward directions. We have applied two such BiLSTM layers in stacked configuration to learn more sophisticated representation of input sequences.
5. **Merge Layer:** Merge layer obtains forward and backward sequences from both the branches by concatenating them.
6. **Contrastive Loss:** Applied a contrastive loss function to merged sequences to minimize the distance between similar pairs of answers and maximize the distance for dissimilar pair of answers.
7. **Output Layer:** Using the output of contrastive loss predicts the output either 0 or 1 for input.

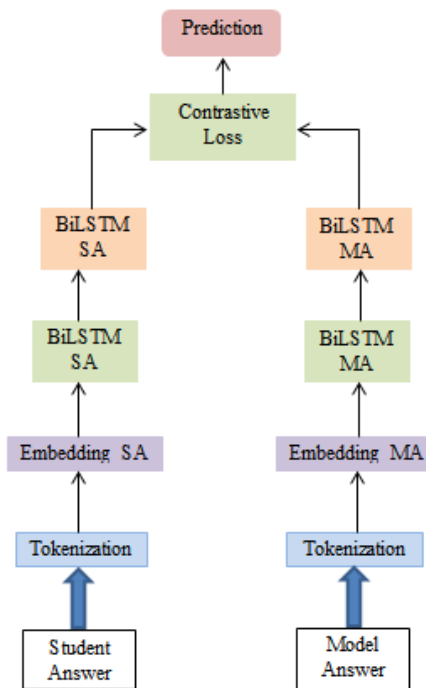


Figure 2. Proposed model architecture

5. EXPERIMENTAL SETUP

In this section, we will explain the complete experimental setup performed to address questions such as Does the domain specific embedding generated for Data Structures domain helps to enhance overall model performance in assessing student answer as compared to pre-trained embedding such as Glove? Does BiLSTM outperform LSTM network for the task of automated short answer grading? Does length of model answer and student answer affect the scoring? Does stacking BiLSTM helps to improve network performs as compared to single BiLSTM network? All these questions will be addressed in next section based on the experimental setup explained here.

The Siamese BiLSTM worked on has utilized 50 dimensional hidden representations. The domain specific embedding as suggested in the study [13] are generated by utilizing skip-gram which are specifically trained on Data Structures domain which are of 300 dimensions. The results are evaluated using two evaluation metrics such as Pearson correlation and RMSE score on Mohler’s dataset [24] and Data Structure dataset mentioned in the study [25]. Mohler’s dataset consist of 80 different questions and approximately 2270 answers. Whereas, dataset mentioned in the study [25] contains approximately 1,820 answers collected through 2-assignments from around 200 students.

Pearson correlation identifies the relationship between two variables whereas RMSE provides as estimation of how well the model is able to predict the values.

$$r = \frac{\sum(x-\bar{x})(y-\bar{y})}{\sqrt{\sum(x-\bar{x})^2(y-\bar{y})^2}} \quad (1)$$

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(Predicted-Actual)^2}{n}} \quad (2)$$

For initial preprocessing such as tokenization, python NLTK library is used. For generating domain specific embedding, we have utilized the same technique as suggested by Shweta and Adhiya [13], for the same corpus related to Data Structures is downloaded in text file. The text file is then preprocessed for removing stop words, removal of junk tokens and then tokenization using NLTK library.

Later Gensim Word2Vec is trained on the cleaned text of tokens with 100 epochs and with window size of 5. And finally, word embedding is generated for 3029 tokens specific to Data Structure domain. Utilizing domain specific embedding helps in capturing the domain related contextual information. The embedding generated are of 300 dimensions.

The Siamese BiLSTM model is developed by python code. The layers are imported from Keras library. Two BiLSTM layers are stacked one above the other such that embedding are fed as input to initial layer which are then processed and the output of it is passed as an input to the next layer. The dropout rate is set to 0.2. Two identical networks are generated in which model answer is passed as input to one network and student answer is passed as input to another network. In the output the distance between both student and model answer is computed using Euclidean distance. Finally, the model is trained using contrastive loss function.

To save the training time we have utilized GPU provided by Google Colab. The dataset is randomly splitted into 80:20 with 80% data utilized for training and remaining 20% for testing. The model is trained with the batchsize of 512 with 50 epochs

and with the learning rate of 0.001 with adam optimizer.

6. RESULTS AND DISCUSSION

In this section, we will discuss the comparison between the proposed architecture with few baseline approaches. We will also try to address the questions mentioned in section 5.

- Does domain specific embedding help to enhance model performance?

Pre-trained embedding is trained on general domain corpus. When utilizing the same on domain specific task; it becomes difficult to capture the context due to ambiguity in words. To address this issue, we have tried to apply the approach proposed in the study [13] to train domain specific embedding. It shows that with the help of domain specific embedding there is quite improvement in the evaluation scores. The correlation coefficient reported using a pre-trained word embedding is 0.655 by Prabhudesai and Duong [12], whereas utilizing domain specific embedding in our proposed architecture we got the correlation coefficient of 0.668 on Mohler’s dataset.

- Does BiLSTM perform better that LSTM network?

BiLSTM network helps to capture the intent of text in the sentences by looking in the sentence from backward and forward direction as compared to the LSTM network. We applied the LSTM network in Siamese neural network along with domain specific embedding on Mohler’s dataset. Table 1 represents Pearson correlation by utilization of LSTM comes to be 0.664 whereas in case of BiLSTM it is 0.678.

Table 1. Pearson correlation and RMSE score for BiLSTM and LSTM network

Dataset	Network	Pearson Correlation	RMSE
Mohler’s Dataset	LSTM	0.664	0.82
	BiLSTM	0.678	0.828

- Does length of model answer and student answer affect the scoring?

In mohler’s dataset we observed that 12% model answer have the same length as the student answer whereas 88% of answers of different length but in it 14% answer length difference is 1. When we computed the results on 20% test data it was observed that though the length of model answer and student answer were different substantially but the answers where evaluated correctly based upon how similar and dissimilar answers are based on context instead of just the length of answer. The results presented in Table 2 are for few such evaluations performed by stacked BiLSTM network with varying model and student answer length.

- Does stacking BiLSTM help to improve the results as compared to single BiLSTM network?

We tried single BiLSTM network in Siamese neural network. It was observed there was no significant improvement in results on Mohler’s Dataset. But when we tested on the Data Structures dataset [25] it showed improvement in Pearson correlation and RMSE score for

stacked BiLSTM as compared to single BiLSTM network results are represented in Table 3.

Table 2. Sample of evaluated results using stacked BiLSTM network

Reference Answer	Student Answer	Predicted Label	Manual Label
The last element in a circular linked list points to the head of the list.	In a circular linked list, the last node contains a pointer that goes back to the first node; in a basic linked list, the last node contains a null pointer.	Correct	Correct
	They are passed by reference because you want the function to change the pointer.	Incorrect	Incorrect
	A binary tree in which the data is in order from left to right.	Correct	Correct
A binary tree that has the property that for any node the left child is smaller than the parent which in turn is smaller than the right child.	A tree which is split based on values. This makes it very easy to search. One can compare the desired value to the root, and if the root is greater than, we search the left side of the tree, if it is less than, we search the right side... and do the same thing recursively	Correct	Correct

Table 3. Pearson correlation and RMSE score for stacked BiLSTM and BiLSTM network

Dataset	Network	Pearson Correlation	RMSE
Mohlers Dataset	BiLSTM	0.678	0.828
	Stacked BiLSTM	0.668	0.889
Data Structure dataset [25]	BiLSTM	0.711	0.914
	Stacked BiLSTM	0.724	0.949

As we are successful in addressing the mentioned questions above; we compared the results generated by proposed architecture of Siamese stacked BiLSTM with the baseline approaches as shown in Table 4 below.

Table 4. Comparison of proposed architecture with baseline approaches on Mohler’s dataset

Approach	Pearson Correlation	RMSE
Kumar, et al. [14]	0.550	0.830
Hassan et al. [16]	0.569	0.797
Prabhudesai and Duong [12]	0.655	0.889
Gomma and Fahmy [17]	0.63	0.91
Proposed Approach	0.668	0.889

It is clearly evident that proposed framework outperforms with Pearson correlation of 0.668 and RMSE score of 0.889

when compared with the baseline approaches for evaluation of short answer responses on Mohler’s dataset. The domain specific embedding has shown a positive contribution in improvement of the proposed architecture. To some extent stacking BiLSTM network has also contributed to the success of model.

We have also tried to utilize the proposed approach on the Data Structure dataset proposed in the study [25] which has more number of samples as compared to Mohler’s dataset. Even the answers collected in the mentioned dataset are more descriptive as compared to Mohler’s dataset wherein the answer length is within the range of 1-3 sentences. The results presented in Table 5 below clearly shows that even the complex descriptive sentences are correctly identified as correct or incorrect by the proposed approach with the Pearson correlation of 0.724 and RMSE score of 0.949.

The model fails to assign correct label to too short answer which consists of 1-3 tokens as it is unable to generate the context for the same and when compare with the reference answer it assigns incorrect label.

Table 5. Pearson correlation and RMSE score on both datasets

Dataset	Network	Pearson Correlation	RMSE
Mohlers Dataset	Stacked BiLSTM	0.668	0.889
Data Structure dataset [25]	Stacked BiLSTM	0.724	0.949

7. CONCLUSION AND FUTURE SCOPE

In this study, approach for short answer evaluation is proposed using Siamese stacked BiLSTM with domain specific embedding. The proposed work is tested on popular dataset for ASAG task that is Mohler’s dataset [24] and also on Data Structures dataset generated by the study [25] by conducting 2 assignments on undergrad class. The study has tried to address few questions related to embedding and network. The study has shown significant results on both datasets with domain specific embedding and stacked BiLSTM network. The model has shown to perform well on even complex and compound sentences as compared to simple sentences. Whereas it fails to assign correct label to too short answers which consist of only 1-3 number of words in the same.

In near future, we plan to increase the vocabulary size of corpus for generating domain specific embedding by collecting more data related to Data Structures domain through multiple online platforms. This will help to understand the domain even better. Even we are planning to replicate the proposed framework on other courses of Computer Engineering such as System Programming. As the system proposed currently performs evaluation of textual sentences only; study need to perform how architecture can be modified for evaluation of regular expressions too in near future.

ACKNOWLEDGMENT

I am grateful to Dr. Krishnakant P. Adhiya for his continuous guidance and support throughout this research

work. His constant suggestions have helped to improve the results. We would also like to thank the Computer Engineering Department of SSBT's College of Engineering and Technology, for allowing us to carry out this research work.

REFERENCES

- [1] Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61: 85-117. <https://doi.org/10.1016/j.neunet.2014.09.003>
- [2] Burrows, S., Gurevych, I., Stein, B. (2015). The eras and trends of automatic short answer grading. *International Journal of Artificial Intelligence in Education*, 25: 60-117. <https://doi.org/10.1007/s40593-014-0026-8>
- [3] Young, T., Hazarika, D., Poria, S., Cambria, E. (2018). Recent trends in deep learning based natural language processing. *IEEE Computational Intelligence Magazine*, 13(3): 55-75. <https://doi.org/10.1109/MCI.2018.2840738>
- [4] Dikli, S. (2006). An overview of automated scoring of essays. *The Journal of Technology, Learning and Assessment*, 5(1).
- [5] Taghipour, K., Ng, H.T. (2016). A neural approach to automated essay scoring. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1882-1891.
- [6] Mohler, M., Bunescu, R., Mihalcea, R. (2011). 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*, pp. 752-762.
- [7] Haller, S., Aldea, A., Seifert, C., Strisciuglio, N. (2022). Survey on automated short answer grading with deep learning: From word embeddings to transformers. *arXiv preprint arXiv:2204.03503*. <https://doi.org/10.48550/arXiv.2204.03503>
- [8] Bao, W., Bao, W., Du, J., Yang, Y., Zhao, X. (2018). Attentive siamese LSTM network for semantic textual similarity measure. In *2018 International Conference on Asian Language Processing (IALP)*, Bandung, Indonesia, pp. 312-317. <https://doi.org/10.1109/IALP.2018.8629212>
- [9] Viji, D., Revathy, S. (2022). A hybrid approach of weighted fine-tuned BERT extraction with deep siamese Bi-LSTM model for semantic text similarity identification. *Multimedia Tools and Applications*, 81(5): 6131-6157. <https://doi.org/10.1007/s11042-021-11771-6>
- [10] Ranasinghe, T., Orašan, C., Mitkov, R. (2019). Semantic textual similarity with siamese neural networks. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pp. 1004-1011. 10.26615/978-954-452-056-4_116
- [11] Mueller, J., Thyagarajan, A. (2016). Siamese recurrent architectures for learning sentence similarity. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1). <https://doi.org/10.1609/aaai.v30i1.10350>
- [12] Prabhudesai, A., Duong, T.N. (2019). Automatic short answer grading using Siamese bidirectional LSTM based regression. In *2019 IEEE International Conference on Engineering, Technology and Education (TALE)*, Yogyakarta, Indonesia, pp. 1-6. <https://doi.org/10.1109/TALE48000.2019.9226026>
- [13] Shweta, P., Adhiya, K. (2022). Comparative study of feature engineering for automated short answer grading. In *2022 IEEE World Conference on Applied Intelligence and Computing (AIC)*, Sonbhadra, India, pp. 594-597. <https://doi.org/10.1109/AIC55036.2022.9848851>
- [14] Kumar, S., Chakrabarti, S., Roy, S. (2017). Earth mover's distance pooling over siamese LSTMs for automatic short answer grading. *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17)*, pp. 2046-2052. <https://doi.org/10.24963/ijcai.2017/284>
- [15] Qi, H., Wang, Y., Dai, J., Li, J., Di, X. (2019). Attention-based hybrid model for automatic short answer scoring. In: Song, H., Jiang, D. (eds) *Simulation Tools and Techniques. SIMUtools 2019. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, vol. 295. Springer, Cham. https://doi.org/10.1007/978-3-030-32216-8_37
- [16] Hassan, S., Fahmy, A.A., El-Ramly, M. (2018). Automatic short answer scoring based on paragraph embeddings. *International Journal of Advanced Computer Science and Applications*, 9(10): 397-402. <http://dx.doi.org/10.14569/IJACSA.2018.091048>
- [17] Gomaa, W.H., Fahmy, A.A. (2020). Ans2vec: A scoring system for short answers. In the *International Conference on Advanced Machine Learning Technologies and Applications (AMTLA2019)*. *AMTLA 2019. Advances in Intelligent Systems and Computing*, vol. 921. Springer, Cham. https://doi.org/10.1007/978-3-030-14118-9_59
- [18] Gaddipati, S.K., Nair, D., Plöger, P.G. (2020). Comparative evaluation of pretrained transfer learning models on automatic short answer grading. *arXiv preprint arXiv:2009.01303*. <https://doi.org/10.48550/arXiv.2009.01303>
- [19] Gong, T., Yao, X. (2019). An attention-based deep model for automatic short answer score. *International Journal of Computer Science and Software Engineering*, 8(6): 127-132
- [20] Riordan, B., Horbach, A., Cahill, A., Zesch, T., Lee, C. (2017). Investigating neural architectures for short answer scoring. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 159-168. <http://dx.doi.org/10.18653/v1/W17-5017>
- [21] Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8): 1735-1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- [22] Schuster, M., Paliwal, K.K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11): 2673-2681. <https://doi.org/10.1109/78.650093>
- [23] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12: 2493-2537
- [24] Mohler, M., Mihalcea, R. (2009). Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pp. 567-575.
- [25] Patil, S., Adhiya, K.P. (2022). Automated evaluation of short answers: A systematic review. In: Hemanth, D.J., Pelusi, D., Vuppalapati, C. (eds) *Intelligent Data*

