4th International Engineering Conference (IEC 2022)
Federal University of Technology, Minna, Nigeria

AN LSTM AND BiLSTM MODELS FOR AUTOMATED SHORT ANSWER GRADING: AN INVESTIGATIVE PERFORMANCE ASSESSMENT

*Nusa, A. M. K[1], Bashir, S. A[2] and Adepoju, S. A[3]

[1,2,3]Department of Computer Science, Federal University of Technology, PMB 65 Minna Niger State, Nigeria
*aminanusa1989@gmail.com +2347060776016

ABSTRACT
Automated Short Answer Grading (ASAG) systems contributes immensely in providing prompt feedback to students which eases the workload of instructors. In this paper, the performance of two deep learning models (LSTM and BiLSTM) were investigated to ascertain their effectiveness in grading short answers. The popular ASAG dataset by Mohler was utilized for the experiment. The dataset contains training samples from Computer Science department with grades between 0-5. The results show that LSTM model performs better interms of training time with lower RMSE and MAPE when compared with BiLSTM.

Keywords: Automated Short Answer Grading; Bidirectional LSTM; Deep learning; LSTM

1 INTRODUCTION

Regularly evaluating student comprehension is crucial in the education process. An automatic short answer grading system (ASAG) can assist with this by evaluating student responses and providing a score based on how closely they match the correct answer. This can be helpful for instructors as it lightens their workload and eliminates the potential for subjective grading. The feedback provided by an ASAG system can also be useful for both students and professors to identify areas where they can improve their understanding. The ASAG system provides timely and effective feedback, allowing both parties to focus on areas of potential improvement (Prabhudesai & Prabhudesai, 2019).

Over the past few decades, there has been significant progress in the fields of Natural Language Processing (NLP) and Machine Learning (ML), which has made it possible to create systems for grading short, subjective answers. One of the first such systems was Project Essay Grade (PEG), which was proposed by Ellis Page in 1960. Another system, called Intelligent Essay Assessor (IEA), uses Latent Semantic Analysis (LSA) and vectors to determine the similarity between student responses and the correct answer. E-rater, on the other hand, relies on NLP to grade English essays on a specific topic. In recent years, Recurrent Neural Networks (RNNs) have garnered a lot of attention for their ability to handle sequential information and understand deeper semantics. For example, RNNs, including variations like Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU), are mostly applied to capture or summarize the meaning of sentences or documents with great success (Wang et al., 2018).

2 LSTM and BiLSTM Models

LSTM is a specific kind of Recurrent Neural Network that uses "memory units" to overcome issues of vanishing and exploding gradients. Additionally, it is able to obtain long-term dependencies in data. In the research area of Natural Language Processing (NLP), LSTM is particularly useful for extracting high-level information from text. A further advancement of LSTM is Bidirectional Long Short Term Memory (BiLSTM), which combines the forward and backward hidden layers, allowing for access to both preceding and succeeding context. Therefore, BiLSTM can perform better in sequential modeling tasks compared to LSTM. Currently, both LSTM and BiLSTM have been used for text classification and have achieved a number of success (Liu et al., 2019).

LSTMs have been widely used in different NLP tasks, such as machine translation and question-answering systems. These tasks often involve sequence-to-sequence models, which are good at mapping input sequences to output sequences. However, these models may not capture sufficient context to generate coherent and meaningful sentences when used in text generation. Simply mapping a sequence of words to the next word may result in

grammatically correct but nonsensical sentences without the necessary contextual information. As a result, the generated sentences may lack coherence and specificity (Santhanam, 2020)

Only one direction Long-term dependencies can be handled by LSTM models, however they only take the preceding word's features into account. Bi-LSTM models, in contrast to LSTM, can take word context into account by examining the sequence from both directions, resulting in two different sequences of LSTM output vectors. These vectors are then combined and sent through a max pooling layer to provide a sentence representation that accounts for both the context immediately before and immediately after the sentence. (Zhang et al., 2018).

For the second type of question categorization, Bidirectional Long Short-Term Memory (Bi-LSTM) models are frequently used in the field of text classification. These models have the ability to categorize text without depending on predefined sentence structures. Numerous text classification research have used bi-LSTM models in fields like news categorization and film genre classification. (Anhar et al., 2019).

Due to their enhanced performance, Bidirectional Long Short-Term Memory (BLSTM) models have lately been more well-liked in question-answering tasks. In a BLSTM-based model for community question answering, each word in a user's query, as well as previous responses, is represented by a vector from the hidden layer. These vectors are then combined to form representations of the complete sentence, and the most similar historical description is selected based on the similarity of these sentence representations. However, a major challenge in this approach is minimizing the impact of irrelevant information in sentences and effectively merging hidden layer vectors to generate trustworthy sentence representations. Previous studies have attempted to address this challenge by incorporating various notification mechanisms at the hidden layer level (Bi et al., 2019).

The Long Short-Term Memory (LSTM) model includes three gates: i as input (Eqn.1), f representing forget in (Eqn.2) and o serving as output in (Eqn.3), c represent a well as a cell memory activation vector. An input vector x_t at a particular time step t was used, the previous output h_{t-1} and cell state c_{t-1}, an LSTM with hidden size k that calculate h_t as the next output and c_t represent the cell state as provided in:

$$i(t) = \sigma(W_i H + b_i) \quad (1)$$

$$f(t) = \sigma(W_f H + b_f) \quad (2)$$

$$o(t) = \sigma(W_o H + b_o) \quad (3)$$

$$H = (x(t), h(t-1))^T \quad (4)$$

$$c_t = f_t c_{t-1} i_t tanh(W_c H + b_c) \quad (5)$$

$$h_t = o_t tanh(c_t) \quad (6)$$

where W_i, W_f W_o and W_c are trained weighted matrices, and the b_i, b_f, b_o and b_c The gates and the input transformation in LSTM architecture are parameterized by biases. The cell memory vector σ is often represented by the sigmoid function, but it can also use in place of other activation functions such as tanh. The gates that denote the input and output are used by an LSTM to regulate the information flow within the cell (Nie et al., 2016).

A BiLSTM is made up of two LSTMs that work together to gather annotations of words by analyzing the sentiment of a sentence by considering information from both the beginning and end of the sentence For every time step, the forward LSTM computes a hidden state fh_t using the previous hidden state fh_{t-1} and the input vector x_t, while the backward LSTM computes the hidden state bh_t using the opposite hidden state bh_{t-1} and the input vector x_t. In the BiLSTM model, the vectors from the forward and backward directions are combined to generate the total hidden state. The LSTM parameters for each direction are distinct but use the same word embeddings. The output produced by the BiLSTM model at time step t, represented by h_t, in equation (7) is the final outcome (Chen et al., 2020).

$$h_t = [fh_t, bh_t] \quad (7)$$

In order to store context information in a cell memory vector, a BiLSTM employs two LSTMs, one of which manipulate the input process in the forward direction and the other of which processes it in the reverse direction. The cell memory vectors from both LSTMs are combined to generate the hidden or output vector at each stage, which means that the full input sequence is taken into consideration. Each stage's final output is a label that determines whether a potential response phrase should be chosen as the appropriate response to an input query. This

prompts the BiLSTMs to learn a weight matrix that produces a positive label when there is a match between the cell memory vectors of the two LSTMs. During training, mean pooling is used to all time step outputs, and mean, sum, and max pooling are used as features during the test phase (Wang & Nyberg, 2015).

An LSTM layer is composed of multiple sections known as storage blocks. Each block has different memory cells that are connected in a recurrent manner, and three units that regulate the input, output, and forget operations for the cells. LSTM has proven to be particularly effective in tasks such as handwriting and speech recognition.

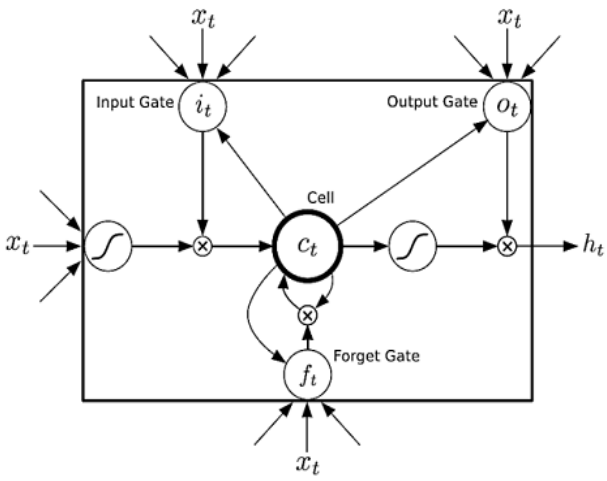

Figure 1: LSTM memory block with one cell

In Figure 1, the diagram depicts a single cell inside an LSTM memory block. The input is multiplied with x_t when the input gate is activated. The output gate activation multiplies the output to the network, while the forget gate activation multiplies the values of the previous cells. Only through the gates can the network communicate with the cells.

Bidirectional LSTM is a method in which a training sequence is processed in both the forward and backward directions by two distinct recurrent networks. Because these networks are linked to the same output layer, they can access the complete sequential data of any given point in the series, including points before and after it. Figure 2 depicts the BLSTM's structural layout.

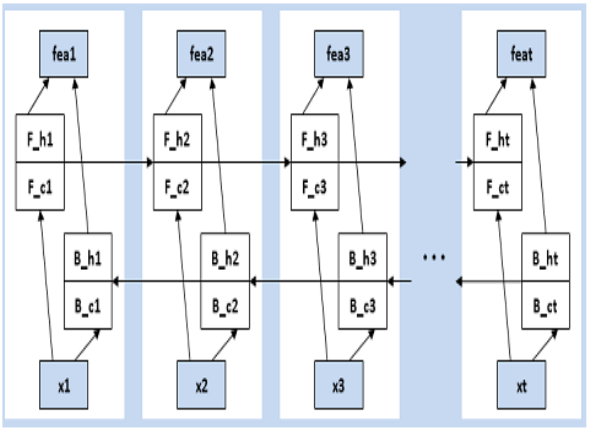

Figure 2: Bidirectional LSTM (Zhang 2015)

An LSTM is able to effectively capture long-range context by using specialized memory cells to store information. A bidirectional LSTM, in contrast to a unidirectional LSTM, is able to retain information from both the past and future by running in both directions and combining the hidden states from both runs. As a result, the bidirectional LSTM is able to preserve information from both past and future (Yang, 2020).

3 Experimentation

The dataset contains a total of 81 questions from 10 assignments and 2 examinations. The questions in the dataset consist of 24 to 31 student answers. An average of 28 answers is given per questions with a total of 2273 answers in the dataset. The answers are graded on a scale of 0 to 5 representing completely correct and perfect answer. In this research work, the average grade was used. Table 3.1 shows the sample of the dataset collected.

Table 1: Dataset Sample

Question 1	Where do c++ program begin to execute?	
Reference Answer	At the main function.	
Students Answers		**Grade**
Answer a	the Function main().	5
Answer b	At the root	2.5
Answer c	in the testing phase	0

Question 2	How many constructors can be made for a class?	
Reference Answer	At the main function.	
Students Answers		**Grade**
Answer a	Any number you want	5
Answer b	Several	4.5
Answer c	one	0

For each class, the performance of the test dataset will be evaluated using the metrics Mean Absolute Percentage Error (MAPE), Root Mean Squared Error (RMSE), and Pearson's correlation coefficient. Mathematically, they are;

$$RMSE = \sqrt{\frac{1}{M}\left(\sum_{i=1}^{M}(A_i - P_i)^2\right)} \quad (8)$$

$$MAPE = \frac{1}{M}\sum_{i=1}^{M}\left|\frac{A_i - P_i}{A_i}\right| * 100\% \quad (9)$$

$$r = \frac{\sum_{i=1}^{M}(A_i - \bar{A})\,(P_i - \bar{P})}{\sqrt{\sum_{i=1}^{M}(A_i - \bar{A})^2\,(P_i - \bar{P})^2}} \quad (10)$$

Where, A_i is the actual output of the i[th] sample while P_i is the i[th] output of the forecasting model, M is the number of samples. $\bar{A}$ and $\bar{P}$ are the mean values of A and P distributions.

4 Results

Figure 3.3 shows the training progress of BiLSTM having the validation RMSE of 1.911, maximum iteration of 15, epoch value of 15, training time of 50s and learning rate of 0.01.

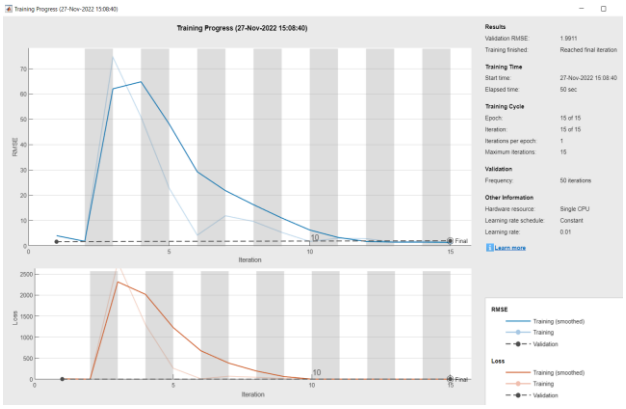

Figure 3:BiLSTM Training progress

Figure 4 indicate the training progress of LSTM with validation RMSE of 1.6968, maximum iteration of 15, epoch value of 15, training time of 31s and learning rate of 0.01.

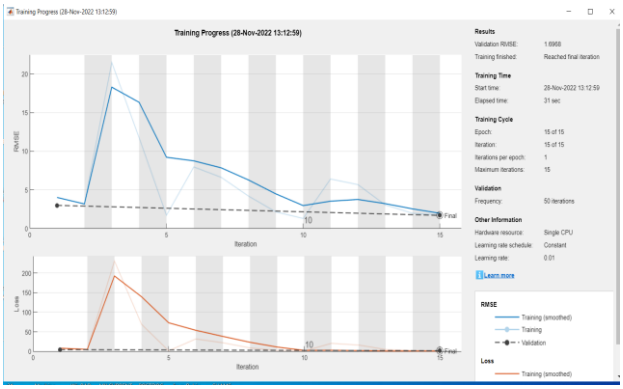

Figure 4: LSTM Training progress

Table 2 is the testing prediction for BiLSTM and LSTM and the actual score for the test sample. The results shows that the BiLSTM produce the maximum score of 5.2803392 and the minimum score of 4.9121804. The LSTM produce the maximum score of 4.8965182 and the minimum score of 4.5399179.

Table 2: Testing Predictions for BiLSTM and LSTM

Test Sample	Actual Score	BiLSTM Predicted Score	LSTM Predicted Score
1	4	4.9522724	4.8965182
2	2	5.2413387	4.6470914
3	4.5	5.2044516	4.630733
4	4.5	5.1670899	4.8637981
5	4.5	5.0219898	4.8651414
6	5	4.9986153	4.8754983
7	5	5.1805205	4.7512088
8	5	5.1927252	4.919198
9	5	5.2283363	4.7981644
10	5	5.2803392	4.7987375
11	5	5.2167382	4.7821507
12	5	5.1415143	4.6602283
13	5	5.1333241	4.6984763
14	5	4.9568081	4.8770456
15	5	5.167491	4.8402829
16	4.5	5.1960979	4.6659188
17	5	5.1492233	4.7653561
18	4	5.0326633	4.6364813
19	5	4.9121804	4.7771959
20	5	5.1628881	4.7738891
21	5	5.083159	4.790328
22	5	5.0674624	4.7035351
23	5	5.1566219	4.7714458
24	3.5	5.1633725	4.6890063
25	3.5	5.0098653	4.6350412
26	5	5.2080369	4.8302584
27	5	5.1249018	4.7602272
28	5	5.1104202	4.7165203
29	5	5.109324	4.8671012
30	1	5.2041483	4.5399179
31	5	4.9276028	4.7814374

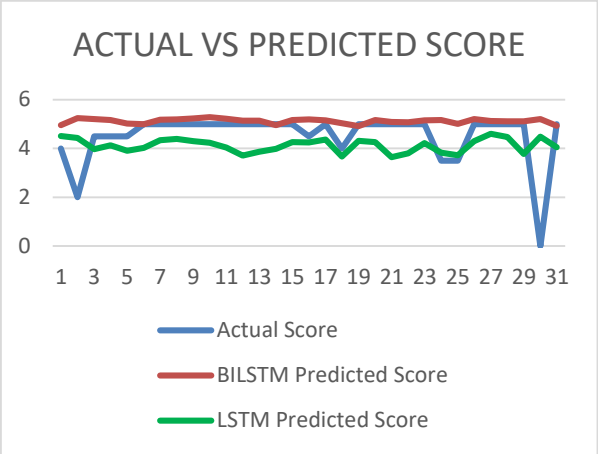

Figure 5: Actual VS Prediction score for BiLSTM and LSTM

The performance evaluation results for BiLSTM and LSTM is given in Table 3. The BiLSTM is having the RMSE value of 1.0983 and MAPE value of 27.05% compare with LSTM with the RMSE value of 0.8943 and MAPE value of 22.68%.

Table 3: Performance Evaluation Results

Performance Metrics	BiLSTM	LSTM
RMSE	1.09831358	0.89426392
MAPE	27.0539615	22.6806

Conclusion

in this paper the performance of deep learning model (BiLSTM and LSTM) were compared for ASAG model problem. The ASAG dataset from Mohlar *et al.* was used for the experiment and RMSE and MAPE was measured. The results show that the LSTM gives a better prediction with faster training time than BiLSTM. The RMSE value for LSTM was obtained at 0.89 and for BiLSTM was 1.10. The MAPE for the LSTM was obtained at 23% and 27% for BiLSTM. The results indicate that the LSTM is a better prediction for ASAG than BiLSTM. However, more research on the LSTM model like optimization of its parameters need to be carried out to improve it performance on ASAG.

Reference

Anhar, R., Mada, U. G., Mada, U. G., & Mada, U. G. (2019). *Question Classification on Question-Answer System using Bidirectional-LSTM*. 1–5.

Bi, M., Zhang, Q., Zuo, M., Xu, Z., & Jin, Q. (2019). Bi - directional LSTM Model with Symptoms - Frequency Position Attention for Question Answering System in Medical. *Neural Processing Letters*, *0123456789*. https://doi.org/10.1007/s11063-019-10136-3

Chen, C., Tseng, S., Kuan, T., & Wang, J. (2020). *Outpatient Text Classification Using Attention-Based Bidirectional LSTM for Robot-Assisted Servicing in Hospital*. https://doi.org/10.3390/info11020106

Liu, G., & Guo, J. (2019). PT US CR. *Neurocomputing*.https://doi.org/10.1016/j.neucom.2019.01.078

Liu, L., Li, Y, Zhang, J., Yu, Z., & Chen, Y. (2019). Attention-Based BILSTM Model for Answer Extraction in Question Answering System. IEEE Xplore. 1827–1831.

Nie, Y., An, C., Huang, J., Yan, Z., & Han, Y. (2016). *A Bidirectional LSTM Model for Question Title and Body Analysis in Question Answering*.307–311. https://doi.org/10.1109/DSC.2016.72

Prabhudesai, A., & Prabhudesai, A. (2019). *Automatic short answer grading using Siamese bidirectional LSTM based regression Automatic Short Answer Grading using Siamese Bidirectional LSTM Based Regression*.

Santhanam, S. (2020). *Context-Based Text-Generation Using LSTM Networks. 1-10.*

Wang, D., & Nyberg, E. (2015). *A Long Short-Term Memory Model for Answer Sentence Selection in Question Answering*. 707–712.

Wang, Z, Liu, J., & Dong, R (2018). *Intelligent Auto-Grading System*. IEEE Proceedings of CCIS 2018. 430–435.

Yang, Z. (2020).Deep *Automated Text Scoring model Based on Memory Network*. 2020 International conference on Computer Vision. 480-484.

Zhang, Q., Mu, L., Zhang, K., & Zan, H. (2018). *Research on Question Classi fi cation Based on Bi-LSTM*. 519–531. https://doi.org/10.1007/978-3-030-04015-4