AUTOMATED SHORT ANSWER GRADING USING LONG SHORT-TERM MEMORY OPTIMIZED WITH PARTICLE SWARM OPTIMIZATION

By

BASHIR SULAIMON ADEBAYO * NUSA AMINA MUHAMMAD KUTIRIKO **

ABDULLAHI IBRAHIM MUHAMMAD ***

*, ** Department of Computer Science, Federal University of Technology, Minna, Niger, Nigeria.
*** Department of Computer Engineering, Federal University of Technology, Minna, Niger, Nigeria.

Date Received: 23/01/2024 Date Revised: 08/02/2024 Date Accepted: 16/02/2024

ABSTRACT

Automated Short Answer Grading (ASAG) systems contribute immensely to providing prompt feedback to students, which eases the workload of instructors. This research focuses on the development of an optimized ASAG model using LSTM model and particle swarm optimization techniques to prevent model overfitting. The popular ASAG dataset by Mohler was utilized for the experiment. The dataset contains training samples from Computer Science department of the Federal University of Technology, Minna, Nigeria, with grades between 0 and 5. In order to effectively optimize the LSTM model parameters, which are learning rate and number of neurons in the LSTM layers, four experiments were performed, each with different particle population sizes (5, 10, 15 and 20). The results show that PS5 model produced the lowest RMSE and MAPE of 0.77697 and 44.5356%, respectively. The PS15 model, however, produced the highest RMSE and MAE of 0.80985 and 56.6192%, respectively. In order to validate the developed PSO-LSTM ASAG model, normal LSTM model for ASAG was implemented and tested. The PSO-LSTM has an RMSE value of 0.77687 and MAPE of 44.5356%, as compared with LSTM, which has an RMSE value of 0.9423 and MAPE of 85.73%. The results clearly show the superiority of the developed hybrid model in predicting the scores of short answer grading. The model's performance can be further improved by increasing the sample size and using other optimization algorithms, such as genetic algorithms or ant colony optimization. Further research can also investigate the effect of other variables, such as question complexity and student writing style, on the model's performance.

Keywords: Deep Learning, Automated Short Answer Grading, LSTM Recurrent Neural Network, Long Short-Term Memory, Particle Swarm Optimization.

INTRODUCTION

As a result of the COVID-19 pandemic, education system shifted to online mode. Almost all educational institutions, from schools to colleges, have adopted the online education method at this time. Automated Short Answer Grading (ASAG) is the use of statistical model to award grades to texts in an educational setting, which is a highly desired goal in education. Machine learning advances are bringing this objective closer to reality (Ghavidel et al., 2020). Assessment plays a significant role in measuring the learning ability of the student. Students can learn individually using online tutoring systems while answers are being evaluated in order to provide users with personalized feedback on their responses. There are an abundance of domain-related questions available on

i-manager Publications
Member of:
This paper has objectives related to SDGs
4 QUALITY EDUCATION
9 INDUSTRY, INNOVATION AND INFRASTRUCTURE

large tutoring platforms, but domain-related information is frequently required to evaluate an answer. Furthermore, the rising accuracy of short answer grading systems makes their use in exams possible. Natural Language Processing (NLP) and deep learning are two artificial intelligence subfields widely used in e-learning (Chanaa & El-Faddouli, 2018; Camus & Filighera, 2020; Robinson et al., 2016; Shehab et al., 2016). For automatic grading, natural language answers are divided into essays or short answers. Each student answer is graded on a nominal, ordinal, or ratio scale in both short answers and essays (Roy et al., 2018). On the nominal scale, grades are in the form of labels like correct, contradictory, and incorrect.

ASAG is an area of NLP where a different sort of dynamic network is widely employed. These dynamic networks are mostly called Recurrent Neural Networks (RNNs) and are powerful tools used to model and classify data that is sequential in nature. These types of networks have been used in engineering and science for the identification and modeling of complex systems (Jafari & Hagan, 2018). A sequence of words can be transformed into a sequence of vectors while preserving the semantic information by using an embedding. RNNs, in combination with embedding, have many applications in NLP tasks like sentiment analysis, topic labeling, language detection, and machine translation (Conneau et al., 2016).

Developing a generic system for short-answer grading is difficult. Grading can be an expensive and time-consuming operation when there are a high number of student answers to the question. Since textual scoring is a component of practically every educational setting for student assessment procedures, there are many ASAG engines being used in large-scale formative and summative assessment (Shermis, 2015). The basic concept behind textual scoring is to assess a text against a rubric that takes into account characteristics such as grammar, text organization, and topic-specific information. An ASAG engine is designed to extract measurable features that may be used to approximate these attributes and, as a consequence, calculate a likely score using statistical inference. Researchers have begun training very deep language models, which are networks meant to predict some element of the text (typically words) based on the other parts, due to the volume of unlabeled text data accessible. Contextual information is eventually learned by these networks. In many NLP tasks, state-of-the-art outcomes have been attained by extending these language models to predict labels rather than words or sentences. Many of these models are made up of layers of transformers that use attention to locate the most relevant attributes for completing a specific task (Devlin et al., 2018; Mathew et al., 2021; Yang et al., 2019; Zhavoronkov et al., 2020).

The long-term memory (LSTM) is widely used in various areas of research, such as analyzing sentiment, recognizing speech, and modeling language. The performance of these models can be improved by the use of non-permanent data sources, which are crucial in predicting future trends. Unlike a feedforward neural network, the LSTM model takes into account the changes in the past time step to provide a forecast. It also generates a memory of past scenarios through its recurrent connection (Bouktif et al., 2020). The LSTM model has showcased immense potential in the field of natural text processing. Neural networks have the capability to exploit the context of natural text to create a single dense vector representation, also known as its embedding. This embedding is then used for mathematical models and computations. Neural networks have proven to be more robust than traditional machine learning models, which make use of handcrafted features (LeCun et al., 2015; Schmidhuber, 2015).

LSTM comprises cycles that feed network activations from the past time step as network input to motivate forecasts at the present time step. Although the recurrent connection enables the model to build a memory of past scenarios that is implicitly encoded in its hidden state variables, many techniques for ASAG have been proposed, including deep learning approaches such as LSTM, Bidirectional Encoder Representations from Transformers (BERT), and XLNET (Condor et al., 2021; Ghavidel et al., 2020; Prabhudesai & Duong, 2019;

Sabharwal & Agrawal, 2021). These models, despite being successful, are susceptible to overfitting, leading to poor performance and reducing the accuracy of the grading system (Mayfield & Black, 2020). This paper presents the ASAG system by optimizing the hyperparameters of the LSTM model using PSO. The performance of the proposed ASAG model was compared with that of the existing LSTM model for ASAG.

1. Methods

1.1 LSTM-PSO Model Architecture

Figure 1 shows the proposed LSTM-PSO ASAG model architectural diagram. It shows various steps that the system passed through to achieve the set-out aim of the research work. The process involves data collection and pre-processing, LSTM model design and training, model testing, hyperparameter optimization, and performance evaluation. For each optimization phase, the objective function was evaluated to obtain the optimal parameters of the model. The PSO-LSTM ASAG model was designed by

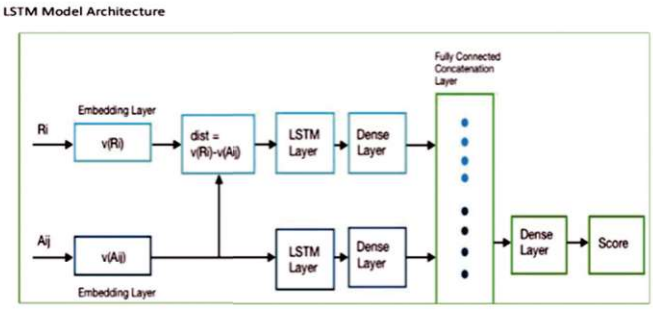

Figure 1. LSTM Model Architecture

setting appropriate parameters for the model. The parameters, such as the learning rate, number of epochs, number of inputs, and number of layers, were selected for the design.

Figure 2 shows the LSTM architecture for ASAG with the layer connections and their functions. The input layer takes in pre-processed data from the tokenization and other preprocessing steps and converts it to numerical values for processing. The embedding layer serves as the mapping layer, where words and relationships are

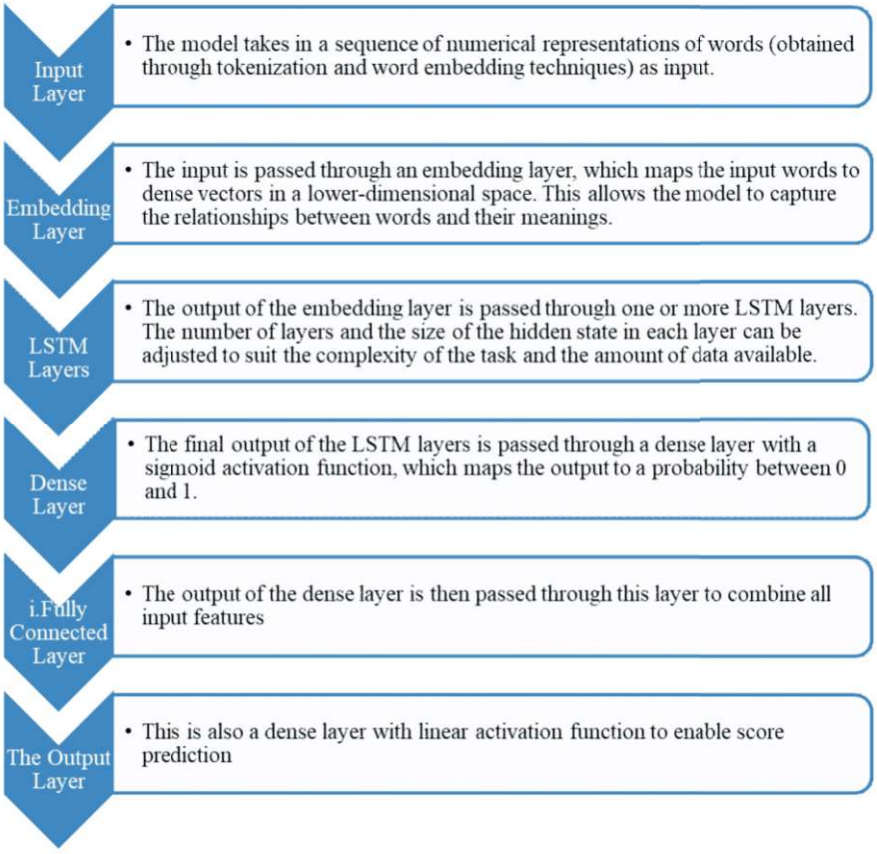

Figure 2. LSTM-PSO ASAG Architectural Description

mapped.

The PSO algorithm was used to optimize the hyperparameters of the LSTM model. The objective or fitness function used to evaluate the fitness of the particles is the root mean squared error. It is given as,

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

Where, A is the actual grade, P is the predicted grade, M is the total number of answers.

The particle size of the PSO algorithm is one important parameter that affects the performance of the algorithm and is common in all metaheuristic optimization algorithms which are population-based. The parameter affects not only the quality of the result but also the complexity of the solution. Investigating this parameter to determine its effect and determine the most appropriate value for fine-tuning the LSTM algorithm is essential to this research. This is due to the fact that this is one of the first studies to apply metaheuristic PSO to fine-tune LSTM model parameters. Four different particle sizes were investigated. They are [5, 10, 15, and 20]. These values are proposed due to the complexity of the model and time of simulation. For each parameter, the entire PSO process was repeated for Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE).

1.2 Dataset

The dataset contains 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 question, for a total of 2273 answers in the dataset. The answers are graded on a scale of 0 to 5, representing a completely correct and perfect answer. In this research work, the average grade was used. Table 1 shows the sample of the dataset collected.

For each class, the performance of the test dataset was evaluated using the metrics Mean Absolute Percentage Error (MAPE), and Root Mean Squared Error (RMSE). Mathematically, they are,

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

Question 1	Where do C++ program begin to execute?	
Reference Answer	At the main function.	
Students Answers		Grade
Answer (a)	The Function Main ().	5.0
Answer (b)	At the Root	2.5
Answer (c)	In the Testing Phase	0.0
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	5.0
Answer (b)	Several	4.5
Answer (c)	One	0.0

Table 1. Dataset Sample

$$MAPE = \frac{1}{M}\sum_{i=1}^{M}\left|\frac{A_i - P_i}{A_i}\right| * 100\% \quad (3)$$

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. A and P are the mean values of A and P distributions.

2. Results and Discussion

Figure 3 shows the convergence curve obtained when using Particle Swarm Optimization (PSO) algorithm to optimize a Long Short-Term Memory (LSTM) model for automated short answer grading, which represents the behavior of the optimization process over time. Typically, the convergence curve shows how the objective function (or fitness function) value changes over iterations or epochs of the PSO algorithm. The objective function measures how well the LSTM model is performing at grading short answers, and the PSO algorithm works by iteratively updating the weights and biases of the model in order to minimize the objective function.

As the PSO algorithm progresses, the convergence curve shows a decreasing trend in the objective function value over time. This indicates that the algorithm is finding better solutions to the optimization problem.

Table 2 shows the summary of the optimal parameters (learning rate, number of neurons) obtained with their corresponding fitness values for Population Size (PS) of 5, 10, 15, and 20, respectively.

At PS of 5, learning rate of 0.0528 and 54 neurons were obtained as optimal hyperparameters: 0.0422 and 56 for PS of 10, 0.0492 and 61 for PS of 15, and 0.050 and 53 for PS of 20. Overall, the experimental results obtained show

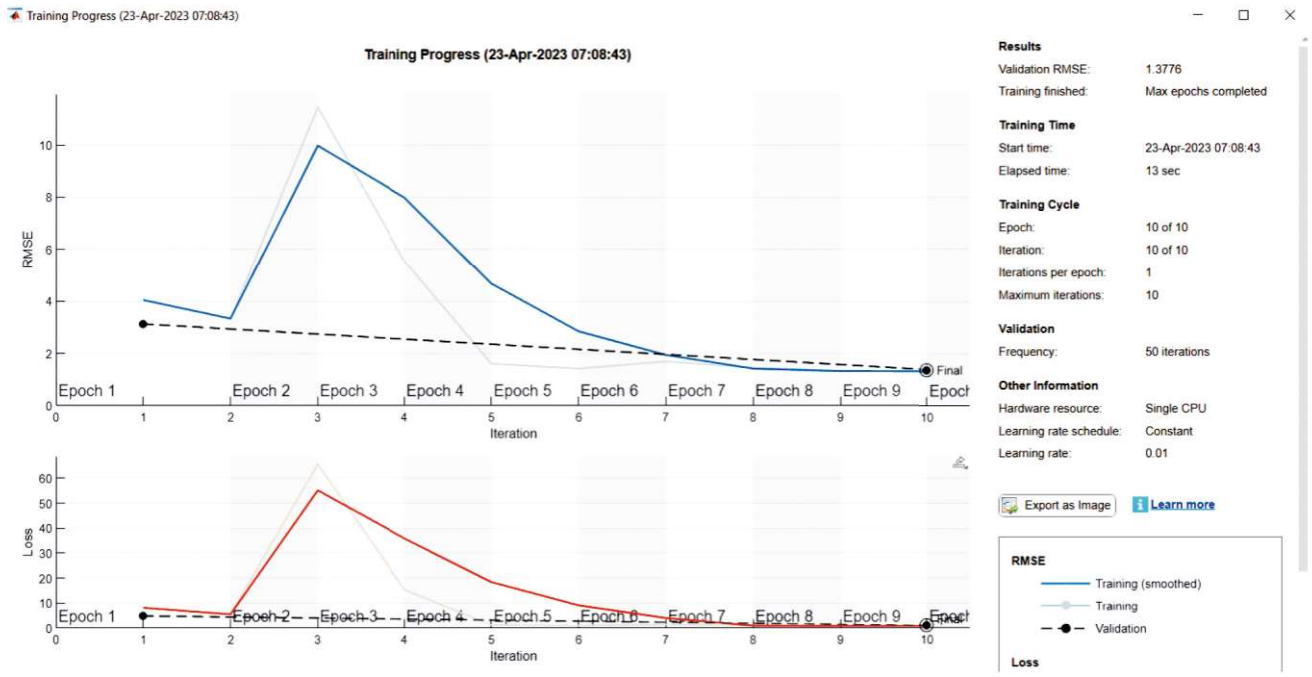

Figure 3. LSTM- PSO ASAG Training Process

Parameter	PS at 5 Iterations	PS at 10 Iterations	PS at 15 Iterations	PS at 20 Iterations
Learning Rate	0.0528	0.0422	0.0492	0.050
Number of Neurons	54	56	61	53
Fitness Value	0.7296	0.7296	0.7931	0.7296

Table 2. Effect of Particle Size on Key LSTM Parameter Values

that with population sizes of 5, 10, and 20, the same optimal fitness was obtained. However, the convergence rate differs with population size. A population size of 10 provides the best fitness and convergence rate.

2.1 LSTM-PSO ASAG Testing Performance Results

Table 3 shows the testing prediction for the LSTM-PSO ASAG model and the actual score for the test samples. The results show that the model produces a maximum score of 5.24 and a minimum score of 4.89 for the PS of 5. Similarly, the results show that the model produces a maximum score of 5.44 and a minimum score of 4.11 for the PS of 10. The results also show that the model produces a maximum score of 5.52 and a minimum score of 4.06 for the PS size of 15. As for the PS of 20, the model produces 5.47 as its maximum score and 4.5958867 as the minimum score for this PS. From the results, it can be observed that all the models fail to accurately predict scores that are low, especially the

Test Sample ID	Actual Score	PS 5 Score	PS 10 Score	PS 15 Score	PS 20 Score
1	4.00	4.92	5.16	4.89	4.90
2	2.00	5.19	5.44	4.69	5.16
3	4.50	5.13	4.75	5.45	5.13
4	4.50	5.12	4.70	5.52	5.11
5	4.50	4.89	4.66	4.50	5.04
6	5.00	5.18	4.21	5.45	4.90
7	5.00	5.17	4.57	5.19	5.36
8	5.00	5.16	4.89	4.92	5.32
9	5.00	5.15	5.11	5.40	5.35
10	5.00	5.24	5.40	5.14	5.19
11	5.00	5.16	4.92	5.41	5.38
12	5.00	5.13	4.55	5.29	5.31
13	5.00	5.15	4.71	5.26	5.21
14	5.00	5.01	4.89	5.36	5.02
15	5.00	5.21	4.28	5.44	5.08
16	4.50	5.10	4.47	5.15	4.60
17	5.00	5.10	4.61	5.38	5.47
18	4.00	5.09	4.27	5.43	5.39
19	5.00	4.94	4.49	4.72	4.73
20	5.00	5.16	4.19	5.39	5.28
21	5.00	5.07	4.66	5.45	5.45
22	5.00	5.14	4.11	5.41	5.03
23	5.00	5.10	4.78	4.71	4.71
24	3.50	5.08	4.44	5.41	5.03
25	3.50	4.95	4.14	4.60	4.52
26	5.00	5.16	4.89	4.92	5.34
27	5.00	5.24	5.02	5.03	5.22
28	5.00	5.33	4.75	5.14	4.81
29	5.00	5.08	4.63	5.41	5.05
30	5.00	5.28	5.28	5.06	5.07
31	5.00	4.94	4.65	4.06	4.93

Table 3. Test Predictions for the LSTM-PSO ASAG Model

second sample with a target score of 2. All the models predicted about 5, except model 3 with PS 5, which predicted 4.69. The results also show that the models predicted higher scores better than lower scores. This could be attributed to the number of samples with lower scores in the target samples.

Figure 4 shows the line curve of actual scores versus predicted scores for the developed PSO-LSTM ASAG Model. From the graph, it can be noticed that the model that closely follows the actual score (black curve) at higher scores is the red curve (at PS 5) followed by blue curve (at PS 20). At lower actual scores, the green curve follows the actual curve by predicting lower values for some cases better than models for PS 5, 20 and 15.

Table 4 shows the performance of the PSO-LSTM models for different population sizes. The results show that PS5 model produced the lowest RMSE and MAPE of 0.77697 and 44.5356%, respectively. The PS15 model, however, produced the highest RMSE and MAPE of 0.80985 and 56.6192%, respectively. Figure 5 shows the RMSE comparison for PSO-LSTM models, and Figure 6 shows the MAPE comparison for PSO-LSTM models.

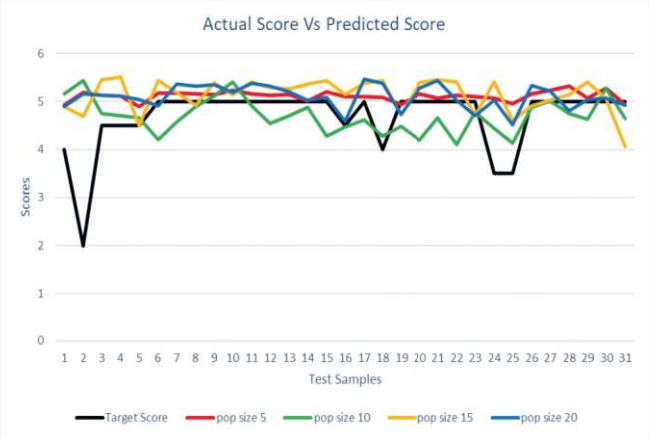

Figure 4. Actual Vs Prediction Score for PSO-LSTM

Models	RMSE	MAPE (%)
PS5	0.77687	44.5356
PS10	0.781674	48.8759
PS15	0.80985	56.6192
Ps20	0.777301	48.1482

Table 4. Performance of the PSO-LSTM Models

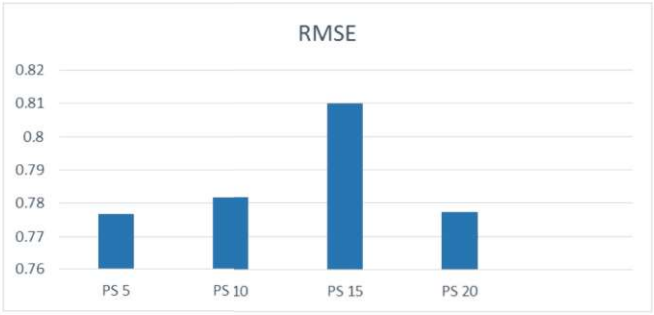

Figure 5. RMSE Comparison for PSO-LSTM Models

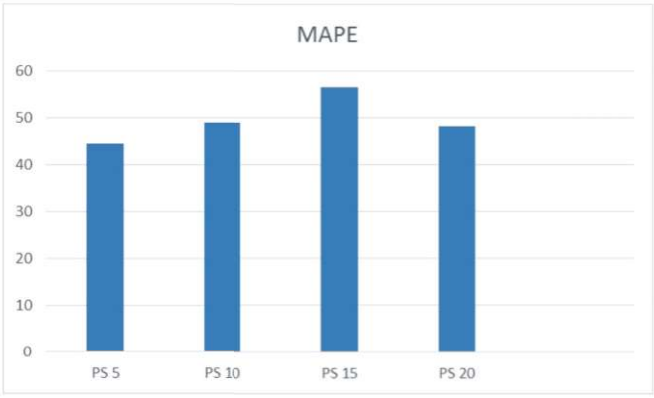

Figure 6. MAPE Comparison for PSO-LSTM Models

2.2 Performance Validation

In order to validate the developed PSO-LSTM ASAG model, normal LSTM model for ASAG was implemented and tested. Table 5 shows the testing prediction for the LSTM ASAG model and the actual score for the test samples. The results show that the model produces a maximum score of 4.2576 and a minimum score of 3.782165.

Figure 7 shows that the results generated are plotted and compared with the actual scores. The results show that the LSTM model predicted scores lower than the actual scores, except in three cases with very low actual scores. This result, when compared with the proposed PSO-LSTM ASAG model, shows that the optimization of LSTM hyperparameters allows for improved score prediction.

To validate the performance of the proposed and normal LSTM ASAG models, the PSO-LSTM has an RMSE value of 0.77687 and a MAPE of 44.5356%, as compared with the LSTM, which has an RMSE value of 0.9423 and a MAPE of 85.73%. The results clearly show the superiority of the

Actual Score	LSTM Prediction	Actual Score	LSTM Prediction
4.00	4.04	5.00	3.97
2.00	4.22	4.00	3.90
4.50	3.98	5.00	3.87
4.50	4.00	5.00	3.96
4.50	3.82	5.00	3.92
5.00	3.96	5.00	3.86
5.00	4.06	5.00	4.08
5.00	4.16	3.50	3.94
5.00	4.08	3.50	3.81
5.00	4.21	5.00	4.18
5.00	4.04	5.00	4.24
5.00	3.94	5.00	4.24
5.00	3.90	5.00	3.94
5.00	3.98	5.00	4.26
5.00	3.99	5.00	3.78

Table 5. Sample Test Predictions for the LSTM ASAG Model

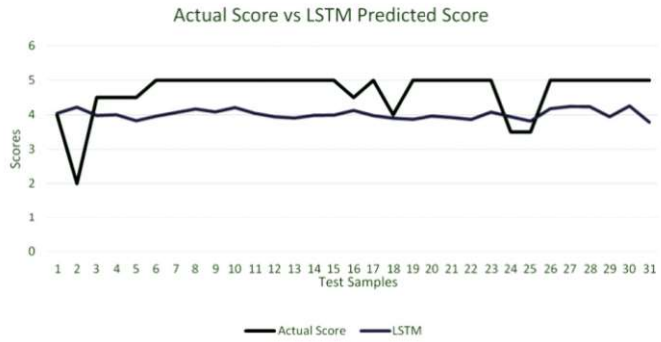

Figure 7. Actual Vs Prediction Score for LSTM

developed hybrid model in predicting the scores of short answer grading.

Conclusion

In conclusion, this paper presents a significant contribution to the field of Automated Short Answer Grading (ASAG) through the development of an optimized model using Long Short-Term Memory (LSTM) architecture enhanced with Particle Swarm Optimization (PSO) techniques. The research aimed at addressing the challenges of overfitting in ASAG models, which is crucial for providing accurate and efficient grading, especially in the evolving landscape of online education accelerated by the COVID-19 pandemic.

The experiments conducted with different particle population sizes demonstrated that the PSO-LSTM model, particularly with a population size of 5, outperformed other variants by producing the lowest Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE). The comparison with a traditional LSTM model underscored the superiority of the proposed hybrid approach in terms of accuracy, as evident from the lower RMSE and MAPE values.

The research developed ASAG model using LSTM and PSO, which was validated using RMSE. The model's performance was satisfactory, indicating that it could be used for grading short answers automatically. The particle swarm optimization algorithm was used to optimize the model's performance, which significantly improved the model's accuracy. From the findings, it can also be concluded that the optimized PSO-LSTM model is good at predicting high scores rather than lower scores, while the LSTM model predicted performs poorly in predicting higher scores.

References

[1]. Bouktif, S., Fiaz, A., Ouni, A., & Serhani, M. A. (2020). Multi-sequence LSTM-RNN deep learning and metaheuristics for electric load forecasting. *Energies,* 13(2), 391. https://doi.org/10.3390/en13020391

[2]. Camus, L., & Filighera, A. (2020). Investigating transformers for automatic short answer grading. In *Artificial Intelligence in Education: 21st International Conference, AIED 2020,* Ifrane, Morocco, July 6–10, 2020, Proceedings, Part II 21 (pp. 43-48). Springer International Publishing. https://doi.org/10.1007/978-3-030-52240-7_8

[3]. Chanaa, A., & El Faddouli, N. E. (2018, October). Deep learning for a smart e-learning system. In *Proceedings of the 2nd International Conference on Smart Digital Environment* (pp. 197-202). https://doi.org/10.1145/3289100.3289132

[4]. Condor, A., Litster, M., & Pardos, Z. (2021). Automatic short answer grading with SBERT on out-of-sample questions. *International Conference on Educational Data Mining (EDM)* (pp. 345-352).

[5]. Conneau, A., Schwenk, H., Barrault, L., & Lecun, Y. (2016). Very deep convolutional networks for natural language processing. *arXiv,* 2(1).

[6]. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional

transformers for language understanding. *arXiv.* https://doi.org/10.48550/arXiv.1810.04805

[7]. Ghavidel, H. A., Zouaq, A., & Desmarais, M. C. (2020). Using BERT and XLNET for the automatic short answer grading task. In *Proceedings of the 12th International Conference on Computer Supported Education,* 1, 58-67. https://doi.org/10.5220/0009422400580067

[8]. Jafari, A. H., & Hagan, M. T. (2018). Application of new training methods for neural model reference control. *Engineering Applications of Artificial Intelligence,* 74, 312-321. https://doi.org/10.1016/j.engappai.2018.07.005

[9]. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature,* 521(7553), 436-444. https://doi.org/10.1038/nature14539

[10]. Mathew, A., Amudha, P., & Sivakumari, S. (2021). Deep learning techniques: An overview. Advanced Machine Learning Technologies and Applications: *Proceedings of AMLTA 2020* (pp.599-608). https://doi.org/10.1007/978-981-15-3383-9_54

[11]. Mayfield, E., & Black, A. W. (2020, July). Should you fine-tune BERT for automated essay scoring? In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications* (pp. 151-162). https://doi.org/10.18653/v1/2020.bea-1.15

[12]. Prabhudesai, A., & Duong, T. N. (2019, December). Automatic short answer grading using siamese bidirectional LSTM based regression. In 2019 *IEEE International Conference on Engineering, Technology and Education (TALE)* (pp. 1-6). IEEE. https://doi.org/10.1109/TALE48000.2019.9226026

[13]. Robinson, C., Yeomans, M., Reich, J., Hulleman, C., & Gehlbach, H. (2016, April). Forecasting student achievement in MOOCs with natural language processing. In *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge* (pp. 383-387). https://doi.org/10.1145/2883851.2883932

[14]. Roy, S., Rajkumar, A., & Narahari, Y. (2018). Selection of automatic short answer grading techniques using contextual bandits for different evaluation measures. *International Journal of Advances in Engineering Sciences and Applied Mathematics,* 10(1), 105-113. https://doi.org/10.1007/s12572-017-0202-9

[15]. Sabharwal, N., & Agrawal, A. (2021). *Hands-on Question Answering Systems with BERT - Applications in Neural Networks and Natural Language Processing.* Hands-on Question Answering Systems with BERT. Apress.

[16]. Schmidhuber, J. (2015). Deep learning. *Scholarpedia,* 10(11), 32832. https://doi.org/10.4249/scholarpedia.32832

[17]. Shehab, A., Elhoseny, M., & Hassanien, A. E. (2016, December). A hybrid scheme for automated essay grading based on LVQ and NLP techniques. In 2016 *12th International Computer Engineering Conference (ICENCO)* (pp. 65-70). IEEE. https://doi.org/10.1109/ICENCO.2016.7856447

[18]. Shermis, M. D. (2015). Contrasting state-of-the-art in the machine scoring of short-form constructed responses. *Educational Assessment,* 20(1), 46-65. https://doi.org/10.1080/10627197.2015.997617

[19]. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., & Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in Neural Information Processing Systems,* 32.

[20]. Zhavoronkov, A., Aladinskiy, V., Zhebrak, A., Zagribelnyy, B., Terentiev, V., Bezrukov, D. S., ... & Ivanenkov, Y. (2020). Potential 2019-nCoV 3C-like protease inhibitors designed using generative deep learning approaches. *Theoretical and Computational Chemistry*. https://doi.org/10.26434/chemrxiv.11829102.v1

ABOUT THE AUTHORS

Bashir Sulaimon Adebayo, Department of Computer Science, Federal University of Technology, Minna, Niger, Nigeria.

Nusa Amina Muhammad Kutiriko, Department of Computer Science, Federal University of Technology, Minna, Niger, Nigeria.

Abdullahi Ibrahim Muhammad, Department of Computer Engineering, Federal University of Technology, Minna, Niger, Nigeria.

i-manager® Publications

3/343, Hill view, Town Railway Nager, Nagercoil
Kanyakumari Dist. Pin-629 001.
Tel: +91-4652-231675, 232675, 276675

e-mail: info@imanagerpublications.com
contact@imanagerpublications.com
www.imanagerpublications.com