IJCCS (Indonesian Journal of Computing and Cybernetics Systems)
Vol.19, No.3, July 2025, pp. 271~280
ISSN (print): 1978-1520, ISSN (online): 2460-7258
DOI: https://doi.org/10.22146/ijccs.109333

Deep Learning for Automatic Assessment and Feedback in LMS-Based Education

Aniek Suryanti Kusuma*[1], Anak Agung Gde Ekayana[2], Desak Made Dwi Utami Putra[3]
[1] Master of Informatics, Institut Business and Technology Indonesia, Denpasar
[2,3] Informatics Dapartment, Institut Business and Technology, Denpasar, Indonesia
e-mail: ***[1]anieksuryanti@instiki.ac.id**, [2]gungekayana@instiki.ac.id, [3]desak.utami@instiki.ac.id

Abstrak

Sistem Manajemen Pembelajaran (LMS) memainkan peran penting dalam pendidikan modern dengan mengatur konten, memfasilitasi komunikasi, dan mendukung proses penilaian siswa. Namun, platform LMS saat ini masih banyak bergantung pada penilaian manual dan umpan balik yang bersifat umum, yang cenderung tidak efisien dan kurang personal. Penelitian ini bertujuan untuk mengembangkan sistem berbasis deep learning yang dapat mengotomatisasi penilaian esai dan memberikan umpan balik yang dipersonalisasi menggunakan teknik Natural Language Processing (NLP). Dataset yang digunakan terdiri dari 17.307 esai siswa, masing-masing diberi label skor dalam rentang 1 hingga 6. Sistem yang diusulkan menggunakan arsitektur Bidirectional Long Short-Term Memory (Bi-LSTM) untuk memprediksi skor berdasarkan masukan teks. Model ini dilatih dengan tujuan regresi dan dioptimalkan menggunakan fungsi kehilangan Mean Squared Error (MSE), yang menghasilkan nilai MSE sebesar 0,557 pada data uji. Visualisasi melalui scatter plot dan word cloud menunjukkan kemampuan model dalam menangkap struktur semantik dan konteks makna dari teks. Temuan ini menunjukkan bahwa arsitektur Bi-LSTM dapat digunakan secara efektif untuk penilaian otomatis esai dalam lingkungan LMS, serta memiliki potensi untuk mengurangi beban kerja pengajar dan meningkatkan kualitas umpan balik. Penelitian selanjutnya akan difokuskan pada pengembangan sistem melalui integrasi mekanisme perhatian (attention mechanism), perbandingan baseline, serta dukungan multibahasa untuk meningkatkan skalabilitas dan dampak praktis sistem.

Kata kunci— *LMS, Deep Learning, NLP, BiLSTM, Penilaian Otomatis, Umpan Balik Pendidikan*

Abstract

Learning Management Systems (LMS) play a critical role in modern education by organizing content, facilitating communication, and supporting student assessment. However, current LMS platforms often rely heavily on manual grading and generic feedback, which can be inefficient and lack personalization. This study aims to develop a deep learning-based system that automates essay assessment and provides personalized feedback using Natural Language Processing (NLP) techniques. A dataset of 17,307 student essays, each annotated with a score ranging from 1 to 6, was used to train and evaluate the model. The proposed system leverages a Bidirectional Long Short-Term Memory (Bi-LSTM) architecture to predict scores based on textual input. The model was trained using regression objectives and optimized with the Mean Squared Error (MSE) loss function, achieving an MSE of 0.557 on the test set. Visualization through scatter plots and word clouds illustrates the model's ability to capture semantic structure and contextual meaning. These findings demonstrate the feasibility of using Bi-LSTM for automated essay scoring in LMS environments and its potential to reduce instructors'

Received April 17th,2025; Revised June 28th, 2025; Accepted July 30th, 2025

workload while improving the quality of feedback. Future research will focus on enhancing the system through attention mechanisms, baseline comparisons, and multilingual support to increase scalability and practical impact.

Keywords— *LMS, Deep Learning, NLP, BiLSTM, Automatic Assessment, Educational Feedback*

1. INTRODUCTION

The integration of deep learning into Learning Management Systems (LMS) presents a transformative opportunity in digital education, particularly in automating the processes of assessment and feedback. LMS platforms have become essential tools for managing online learning, offering centralized access to content delivery, student progress tracking, and interaction. However, traditional LMS implementations still rely heavily on manual intervention for grading and feedback, which makes the process time-consuming, labor-intensive, and prone to inconsistencies.

Recent advances in deep learning—especially in Natural Language Processing (NLP)—have enabled the development of intelligent educational tools capable of analyzing large-scale student data such as written assignments, forum interactions, and performance trends [1], [2]. These tools support the delivery of real-time, personalized feedback, thus reshaping the digital learning landscape by enabling more adaptive and responsive instructional methods [3].

Despite these technological advancements, several challenges persist in conventional LMS assessment practices. These include the limited ability to provide timely and personalized feedback, concerns about academic integrity in AI-mediated environments, and the substantial workload placed on educators [4]. Furthermore, traditional assessment approaches often fail to align with the dynamic and diverse nature of modern online learning environments, thereby affecting student motivation and engagement [5]. As student populations continue to grow and diversify, the need for scalable, intelligent assessment solutions becomes increasingly critical. Without automation and intelligent systems, LMS platforms risk becoming ineffective in meeting the pedagogical demands of 21st-century education.

This study explores the application of deep learning methods within LMS environments to automate the assessment and feedback process in a way that benefits both learners and educators. The main objective is to design and implement a system that leverages NLP techniques to evaluate student-generated content—such as essays and forum responses—and generate real-time, individualized feedback. This research is driven by the growing demand for scalable AI-driven solutions in education and the potential of these technologies to reduce instructor workload while enhancing learning outcomes [2], [6]. It also responds to the need for more adaptive and intelligent educational systems that align with contemporary pedagogical paradigms [7].

To address these objectives, the proposed framework integrates a deep learning model into the LMS ecosystem at INSTIKI, focusing on the automated assessment of textual data using NLP techniques. The key contributions of this research are twofold: (1) demonstrating the technical feasibility and educational impact of AI-enabled assessment tools in LMS contexts, and (2) providing empirical insights into how such systems can enhance student engagement while preserving instructional quality [8], [9]. The evaluation emphasizes the model's ability to deliver timely, consistent, and context-sensitive feedback across varied educational scenarios. Ultimately, this work aims to

advance the development of more intelligent, scalable, and student-centered LMS platforms, reinforcing the transformative potential of deep learning in shaping the future of digital education [10].

2. REALATED WORK

Several recent studies have highlighted the promising role of deep learning, particularly Natural Language Processing (NLP), in improving educational assessment within Learning Management Systems (LMS). One major research theme focuses on the use of NLP for analyzing educational text. For example, [11] conducted a comprehensive review of NLP strategies and emphasized their potential to enhance textual content comprehension in the context of educational quality assessment. Their work shows that NLP can help build interpretative connections between educators and learners, which are crucial for meaningful and individualized feedback. Complementing this, [12] explored current trends and challenges in feedback analysis using NLP, especially the difficulty of applying aspect-based sentiment analysis in educational domains. Their findings suggest that tailoring feedback through improved textual analysis could enhance the responsiveness and relevance of LMS-based instructional strategies.

In addition to text-level analysis, another line of research centers on predictive modeling using student interaction data. The study by [13] demonstrated the potential of Long Short-Term Memory (LSTM) networks to analyze time-series LMS logs for the early identification of students at risk of academic underperformance. This highlights the capability of deep learning not only to assess students retrospectively, but also to support proactive intervention through behavioral prediction. Such models reinforce the growing trend of data-driven education where timely decisions are made based on individual learning trajectories.

Another important stream of research investigates automated assessment using deep learning models. For instance, [14] emphasized how NLP and machine learning technologies can be applied in language education to provide automated feedback for student-generated responses. Their findings highlight the dual role of such tools in both assessing student work and informing instructional design. Similarly, [15] successfully applied convolutional neural networks (CNNs) combined with advanced vector representations to automatically grade open-ended answers. These works demonstrate that deep learning can increase the objectivity, efficiency, and scalability of assessment practices in digital learning environments.

Although these studies contribute valuable insights, many focus on isolated components such as feedback analysis, behavioral prediction, or automated scoring. There remains a significant gap in research that attempts to integrate these capabilities into a unified, end-to-end system for real-time assessment and feedback within operational LMS platforms. Additionally, most prior studies rely on static or offline data without considering continuous integration into classroom environments. To address these limitations, the present study proposes a deep learning framework using stacked Bidirectional LSTM architecture for essay evaluation and real-time feedback generation. This model is designed to operate directly within LMS workflows, bridging existing gaps between theory and practice while supporting personalized, scalable digital learning.

3. METHODS

The method used in this research is deep neural network modeling based on bidirectional Long Short-Term Memory, specifically designed to handle sequential data, such as text. The model architecture is developed using the TensorFlow and Keras frameworks, with a stepwise approach that includes a word representation stage, bidirectional sequential feature extraction, and final value prediction through dense layers. The model is designed to perform regression, i.e., generate continuous value predictions based on processed text input.

3.1 Dataset Description

The dataset used in this research consists of 17,307 textual records of student-written essays, each accompanied by a corresponding score. Each data instance includes three attributes: an essay_id as a unique identifier, the full_text containing the complete essay content, and a score representing the manually assigned assessment score. The score is an integer ranging from 1 to 6, indicating the quality of the essay based on predefined rubric criteria. This dataset simulates real-world student submissions commonly found in Learning Management System (LMS) environments, making it suitable for developing and evaluating automated assessment systems. The average score across the dataset is approximately 2.95, with a standard deviation of 1.04, reflecting a diverse distribution of writing quality. The essays span a wide range of topics and writing styles, providing a rich and varied corpus for training natural language processing models. This dataset forms the foundation for training the deep learning model to predict essay scores based on textual features, enabling scalable and consistent feedback delivery in LMS-based education systems.

3.2 Data Preprocessing

Before model training, several preprocessing steps were applied to ensure data quality and compatibility with neural network inputs. First, all newline characters (\n), tabs (\t), and carriage returns (\r) were removed from the text using regular expressions to produce clean, uninterrupted textual inputs. Next, the number of words in each essay was calculated to assess length distribution. The average essay length was approximately 113 words, and the maximum observed length was 446 words. To maintain consistency, each essay was tokenized and subsequently padded or truncated to a fixed length of 446 tokens, corresponding to the maximum sequence length.

A Tokenizer object from the Keras API was used to convert the cleaned essays into sequences of integers, with support for out-of-vocabulary tokens using <OOV>. Tokenization was followed by zero-padding at the end of each sequence (post-padding) to ensure uniform input dimensions across the dataset. The final tokenized sequences were stored in NumPy arrays for use in model training and validation.

3.3 Dataset Splitting and Validation Strategy

To ensure robust model evaluation and avoid data leakage, the dataset was split into three parts using an 80-10-10 strategy. First, 80% of the data was used for training. The remaining 20% was equally divided into validation (10%) and test (10%) sets using the train_test_split() function from Scikit-learn with a fixed random seed (random_state=42) for reproducibility. The validation set was used for hyperparameter tuning and early stopping, while the test set was held out for final model evaluation.

3.4 Model Architecture

The model architecture is based on a stacked Bidirectional Long Short-Term Memory (Bi-LSTM) network. The input layer accepts sequences of tokenized words, which are embedded into a 128-dimensional dense vector space. The embedded sequences are passed through four stacked Bi-LSTM layers with progressively smaller dimensions (64, 32, 32, and 16 units). This stacking design was chosen to allow hierarchical extraction of syntactic and semantic features from the input text.

After the recurrent layers, two fully connected (Dense) layers with 64 and 32 units respectively are used, each followed by a Dropout layer (rate = 0.1) to prevent overfitting. The final output layer uses a single neuron with ReLU activation to predict a non-negative continuous score. The model is optimized using the Adam optimizer with a learning rate of 1e-4, and trained with the Mean Squared Error (MSE) loss function, as appropriate for regression tasks.

This layered configuration was chosen based on prior works demonstrating that stacked Bi-LSTM structures perform well for long-form text regression, capturing both forward and backward dependencies. The use of ReLU in the output layer ensures that predicted scores remain in a non-negative range.

4. RESULTS AND DISCUSSION

4.1 Word Distribution and Semantic Patterns

To better understand the textual characteristics of student essays, a word cloud was generated from the training dataset. This visualization (Figure 1) highlights the most frequently occurring tokens after stopword removal and basic tokenization. Dominant words such as cars, driverless, people, venus, technology, and college suggest that students wrote on a wide range of topics, including science, society, transportation, and space.

However, the presence of out-of-context terms like venus, mars, aliens, and planet indicates potential topic drift or dataset noise, likely caused by open-ended essay prompts. Such diversity, while beneficial for model generalization, can introduce inconsistencies in automated scoring. To address this, stemming or lemmatization can be applied during preprocessing to unify similar terms—such as car and cars—into a canonical form, thereby improving the semantic consistency of input sequences.
Additionally, the lack of domain filtering may lead to semantically disjoint training data. While this does not necessarily compromise model training, it can reduce interpretability. In future iterations, topic modeling or semantic clustering could be applied to filter or categorize input data for more coherent training.

To standardize the model input, each essay was padded or truncated to a fixed length of 446 tokens, based on the maximum observed in the dataset. This ensures uniformity in input dimensions and compatibility with the LSTM-based architecture.

4.2 Distribution of Word Count in Text Input

Figure 1 shows the distribution of the number of words (num_words) in the text data corpus used as input for the model. This visualization uses a histogram combined with a density estimation curve to provide a smoother picture of the data distribution.

Most of the data has a text length between 200 and 400 words, with the peak of the distribution (mode) around 250 words. This indicates that the majority of user review inputs are of moderate length, neither too short nor extremely long. However,

this distribution has a long right tail (right-skewed), indicating the presence of a small number of very long texts, some of which exceed 1,000 words.

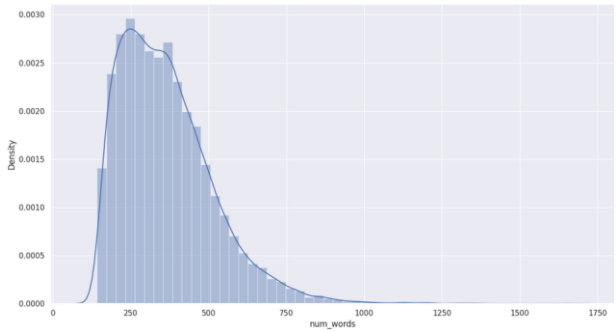

Figure 1. Histogram showing the distribution of input text lengths (in word count)

The figure 1 shows a histogram representing the distribution of the number of words (labeled as "num_words" on the x-axis) with corresponding density values (on the y-axis). The data appears to follow a skewed distribution, peaking around 200-300 words, where the density is highest (approximately 0.0025). The frequency decreases gradually as the number of words increases, with a long tail extending towards 1900 words. This suggests that most observations have a relatively low word count, while fewer instances have a higher word count. The shaded area under the curve indicates the density of the distribution across the range of word counts.

4.3 Model Performance and Error Distribution

To evaluate the predictive performance of the proposed Bidirectional LSTM model, we measured the Mean Squared Error (MSE) on the test set, which yielded a score of 0.5572. This indicates that the model is reasonably capable of predicting essay scores that approximate the actual human-graded values. While not perfect, this performance suggests that the model can serve as a foundational layer in automated assessment systems for LMS environments.

A scatter plot of actual versus predicted scores provides visual insight into the distribution of errors. Ideally, predicted values should align closely with the red diagonal line (representing perfect predictions). The plot demonstrates that most predictions cluster around the correct scores, especially for lower to mid-range values (1–4). However, some dispersion is observed in higher score ranges (5–6), where the model tends to slightly underpredict or overpredict. This variance could be due to insufficient training data in the higher score spectrum or increased linguistic complexity in those responses.

The residual distribution (i.e., the vertical spread of points from the diagonal) is generally narrow, indicating low error variance. However, a small number of outliers—predictions that significantly deviate from the actual scores—highlight the need for further refinement in handling ambiguous or atypical inputs.

Future work may include evaluating the impact of techniques such as class rebalancing, ensemble modeling, or attention mechanisms to further reduce error and stabilize prediction variance across all score ranges.

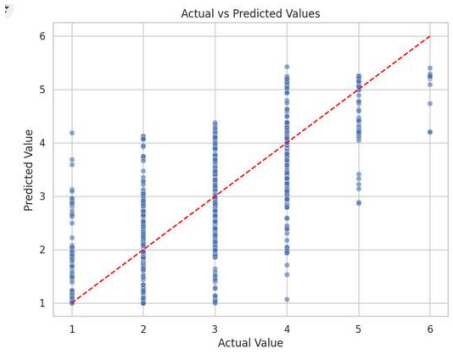

Figure 2. Actual vs. Predicted Scores on Test Data.

The Figure 2 is a scatter plot titled "Actual vs Predicted Values," which compares the actual values (on the x-axis) against the predicted values (on the y-axis) of a model. The x-axis, labeled "Actual Value," ranges from 1 to 6, while the y-axis, labeled "Predicted Value," also ranges from 1 to 6. Each point on the plot represents a pair of actual and predicted values, with blue dots indicating individual data points.

A prominent red dashed line runs diagonally from the bottom-left (1, 1) to the top-right (6, 6), representing the line of perfect prediction where the predicted value equals the actual value. The distribution of the blue dots around this line provides insight into the model's performance:

- Most data points are clustered near the line, suggesting that the predictions are generally close to the actual values.
- There is some vertical scatter, indicating variability in the predictions. For instance, at an actual value of around 3, the predicted values range from approximately 2 to 5, showing a moderate spread.
- The density of points appears higher around the middle range (actual values 2 to 4), with fewer points at the extremes (actual values 1 and 6).
- Some outliers are visible, such as points where the predicted value deviates significantly from the actual value (e.g., a predicted value of around 5 when the actual value is 3).

Overall, the plot suggests that the model has reasonable predictive accuracy, as many points align closely with the perfect prediction line, though there is room for improvement due to the observed scatter and outliers.

4.4 Outlier Analysis

During evaluation, a small subset of predictions exhibited significant deviation from the actual scores, defined as errors exceeding ±2 points on a 1–6 scale. Upon manual inspection, these outliers were often associated with atypical or ambiguous inputs—such as extremely short essays or responses containing off-topic content. For example, several essays that lacked context or coherent structure received much lower predicted scores than their human-assigned values. Conversely, verbose but low-quality responses were sometimes overestimated by the model. These mispredictions indicate the model's sensitivity to both length and linguistic patterns, and they highlight areas where the training data may lack sufficient representation. Addressing these outliers will

require improved data preprocessing, outlier-aware training, and possibly integrating attention mechanisms for more contextual understanding.

4.5 Baseline Comparison

Due to the limited scope of this study, no baseline models such as Linear Regression, Support Vector Regression, or Random Forest were implemented. As such, a direct comparison between traditional machine learning methods and the Bi-LSTM model could not be conducted. However, existing literature suggests that deep learning architectures—particularly sequence-aware models like Bi-LSTM—are generally more effective in capturing linguistic patterns in natural language tasks compared to feature-based classical models. Future work should include benchmark models using TF-IDF or embedding-based features to quantitatively measure performance gains and validate the robustness of the proposed architecture. Including such comparisons will strengthen the generalizability and reproducibility of the findings.

4.6 Error Analysis

A deeper analysis of prediction errors across different score levels reveals some important trends. The model performs best in the mid-range scores (levels 3 and 4), where the data distribution is most balanced. However, higher variance is observed at the extremes—particularly scores 1 and 6—likely due to class imbalance and semantic complexity. These patterns suggest a mild prediction bias toward the mean. Additionally, the use of a single evaluation metric (MSE) may not fully capture nuanced performance. Future research may benefit from introducing Mean Absolute Error (MAE), R^2 score, or coefficient of determination to provide a more comprehensive evaluation. Incorporating cross-validation would also strengthen the reliability of the results, especially in cases where class frequencies vary significantly.

4.7 Implications for LMS Integration

The proposed Bi-LSTM model has strong potential for integration into real-world Learning Management Systems (LMS), particularly in automating formative assessment workflows. Once deployed, the model can serve as an intelligent backend service that evaluates open-ended student responses and provides immediate, personalized scores. For example, in essay-based courses or language learning modules, the model can be used to offer real-time feedback on structure, clarity, and content relevance—functions that typically require manual grading. Furthermore, the model can be extended to highlight strong and weak sections of a student's writing, offer revision suggestions, and track student progress over time. By automating these tasks, the LMS can reduce instructor workload, provide consistent evaluation, and enhance student engagement. These capabilities align with broader efforts to make online education more adaptive, data-driven, and scalable.

5. CONCLUSIONS

This study presented the development of a Bi-LSTM-based deep learning model for automatic assessment and feedback in LMS-based education environments. The primary objective was to explore how Natural Language Processing (NLP) techniques—particularly sequence modeling—can be used to evaluate student essay responses and generate reliable score predictions. The model demonstrated promising results with a test-set Mean Squared Error (MSE) of 0.557, indicating its effectiveness in approximating human-assigned scores.

From a pedagogical standpoint, the model's integration into LMS platforms offers a scalable solution for automated, real-time feedback delivery, which can significantly reduce instructor workload and enhance student engagement. The approach supports personalized learning by offering immediate evaluation and can serve as a foundational component for intelligent tutoring systems.

The key contributions of this work include: (1) implementing a deep learning framework that handles unstructured student-generated text data, (2) demonstrating the feasibility of Bi-LSTM in an educational scoring context, and (3) proposing a conceptual path toward integration within existing LMS infrastructures.

Nevertheless, the study has several limitations. The evaluation relied solely on the MSE metric, which, while useful, does not capture all aspects of prediction quality. Future research should explore additional evaluation metrics, incorporate baseline models for comparison, and examine the impact of class imbalance more systematically. Enhancements such as attention mechanisms, topic modeling, and ensemble learning may further improve prediction robustness. In conclusion, the findings support the viability of deep learning approaches for automating assessment and feedback in digital education, providing a foundation for more intelligent, adaptive, and learner-centered LMS solutions.

REFERENCES

[1] A. Akavova, Z. Temirkhanova, and Z. M. Lorsanova, "Adaptive Learning and Artificial Intelligence in the Educational Space," *E3s Web Conf.*, vol. 451, p. 6011, 2023, doi: 10.1051/e3sconf/202345106011.

[2] H. Munir, B. Vogel, and A. Jacobsson, "Artificial Intelligence and Machine Learning Approaches in Digital Education: A Systematic Revision," *Information*, vol. 13, no. 4, p. 203, 2022, doi: 10.3390/info13040203.

[3] H. Ji, L. Suo, and C. Hua, "AI Performance Assessment in Blended Learning: Mechanisms and Effects on Students' Continuous Learning Motivation," *Front. Psychol.*, vol. 15, 2024, doi: 10.3389/fpsyg.2024.1447680.

[4] E. Kldiashvili, M. G. de S. Ana, and Z. Maia, "Academic Integrity Within the Medical Curriculum in the Age of Generative Artificial Intelligence," *Heal. Sci. Reports*, vol. 8, no. 2, 2025, doi: 10.1002/hsr2.70489.

[5] K. A. A. Gamage, S. C. P. Dehideniya, Z. Xu, and X. Tang, "ChatGPT and Higher Education Assessments: More Opportunities Than Concerns?," *J. Appl. Learn. Teach.*, vol. 6, no. 2, 2023, doi: 10.37074/jalt.2023.6.2.32.

[6] M. Sallam and K. Al-Salahat, "Below Average ChatGPT Performance in Medical Microbiology Exam Compared to University Students," *Front. Educ.*, vol. 8, 2023, doi: 10.3389/feduc.2023.1333415.

[7] M. A. AlAfnan, S. Dishari, M. Jovic, and K. Lomidze, "ChatGPT as an Educational Tool: Opportunities, Challenges, and Recommendations for Communication, Business Writing, and Composition Courses," *J. Artif. Intell. Technol.*, 2023, doi: 10.37965/jait.2023.0184.

[8] A. W. Fazil, M. Hakimi, A. K. Shahidzay, and A. Hasas, "Exploring the Broad Impact of AI Technologies on Student Engagement and Academic Performance in University Settings in Afghanistan," *Riggs J. Artif. Intell. Digit. Bus.*, vol. 2, no. 2, pp. 56–63, 2024, doi: 10.31004/riggs.v2i2.268.

[9] Y. Sun, "A Comprehensive Evaluation Scheme of Students' Classroom Learning Status Based on Analytic Hierarchy Process," *Educ. Innov. Emerg. Technol.*, vol. 3, no. 4, pp. 1–10, 2023, doi: 10.35745/eiet2023v03.04.0001.

[10] C. D. González-Carrillo, F. Restrepo-Calle, J. J. R. Echeverry, and F. A. González, "Automatic Grading Tool for Jupyter Notebooks in Artificial Intelligence Courses," *Sustainability*, vol. 13, no. 21, p. 12050, 2021, doi: 10.3390/su132112050.

[11] E. A. E. Lukwaro, K. Kalegele, and D. G. Nyambo, "A Review on NLP Techniques and

Associated Challenges in Extracting Features From Education Data," *Int. J. Comput. Digit. Syst.*, vol. 15, no. 1, pp. 961–979, 2024, doi: 10.12785/ijcds/160170.

[12] T. Shaik *et al.*, "A Review of the Trends and Challenges in Adopting Natural Language Processing Methods for Education Feedback Analysis," *Ieee Access*, vol. 10, pp. 56720–56739, 2022, doi: 10.1109/access.2022.3177752.

[13] F. Chen and Y. Cui, "Utilizing Student Time Series Behaviour in Learning Management Systems for Early Prediction of Course Performance," *J. Learn. Anal.*, vol. 7, no. 2, pp. 1–17, 2020, doi: 10.18608/jla.2020.72.1.

[14] J. Wei, "The Feasibility of Integrating Natural Language Model in Daily English Education," *Lect. Notes Educ. Psychol. Public Media*, vol. 73, no. 1, pp. 130–134, 2024, doi: 10.54254/2753-7048/73/20241031.

[15] G. Smith, R. Haworth, and S. Žitnik, "Computer Science Meets Education: Natural Language Processing for Automatic Grading of Open-Ended Questions in eBooks," *J. Educ. Comput. Res.*, vol. 58, no. 7, pp. 1227–1255, 2020, doi: 10.1177/0735633120927486.