**Information Research Communications, 2027; 4(1):145-154.**
https://inforescom.org

Research Article

# Development and Evaluation of a Lightweight BERT-Based Automatic Essay Scoring System for Higher Education

**Megat Norulazmi Megat Mohamed Noor*, Noormadinah Alias**

Universiti Kuala Lumpur, Malaysian Institute of Information Technology, MALAYSIA.

**ABSTRACT**

**Background:** The growing need for rapid and reliable assessment of university students has encouraged the development of automated essay evaluation systems. Automatic Essay Scoring (AES) can reduce grading workload while supporting consistency and timely feedback. **Objectives:** This study aimed to develop, implement, and evaluate a web-based AES system using a lightweight BERT-based model for assessing English essays at the tertiary level. **Methodology:** The web-based system incorporated three core modules: an automatic essay scoring engine, an AI-generated text detection module, and a sentence-level plagiarism checker. Role-based dashboards were developed for students, lecturers, and administrators to facilitate essay submission, assignment management, authorization, automated reporting, and lecturer review of machine-generated scores. The AES model was fine-tuned using a transformer-based approach and evaluated using the Quadratic Weighted Kappa (QWK) metric to determine agreement between machine-generated and lecturer scores. **Results:** Preliminary in-domain evaluation produced QWK scores exceeding the 0.80 acceptance threshold, indicating high agreement between automated and lecturer assessments. The AI-generated text detection and plagiarism-checking modules provided additional support for academic integrity evaluation. However, their outputs were considered advisory rather than definitive evidence of misconduct. **Conclusion:** The developed lightweight BERT-based AES system demonstrated strong potential for reducing grading burden and improving assessment consistency in tertiary education. Its integration with AI-text detection and plagiarism checking provides a foundation for intelligent and efficient assessment systems, while human oversight remains essential for final academic decisions.

**Keywords:** AI-Generated Text Detection, Automatic Essay Scoring, DistilBERT, Educational Technology, Higher Education, Plagiarism Checking, Transformer Models.

**Correspondence:**
**Megat Norulazmi Megat Mohamed Noor**
Universiti Kuala Lumpur, Malaysian Institute of Information Technology, MALAYSIA.
Email: megatnorulazmi@gmail.com

**Received:** 24-06-2026;
**Revised:** 02-08-2026;
**Accepted:** 16-10-2026.

## INTRODUCTION

Writing essays remains an important tool for evaluating students' understanding, reasoning, and communication skills in higher education institutions. Unlike questions that are designed objectively, essays require students to organize their thoughts, argue, and demonstrate their language skills. However, marking essays manually is laborious, tedious, and inconsistent, particularly when the instructor marks large groups of students.

Thus, AES has become an important area of research in the field of educational technology and natural language processing. While the early AES systems relied upon manually engineered linguistic, syntactic, and discourse features, the modern systems use deep learning techniques that generate semantic and contextual representations based on the input text. The transformer-based approaches such as BERT and its smaller versions can be used as an effective basis for constructing AES systems because of their ability to model bidirectional context and fine-tuning (Vaswani *et al.*, 2017; Devlin *et al.*, 2019; Sanh *et al.*, 2019).

This study develops an AES solution that can be implemented in educational settings. Notably, the proposed solution goes beyond simple scoring to incorporate academic integrity by providing features such as text detection and sentence plagiarism via AI. This is vital since the increasing availability of generative AI has increased the difficulty of evaluating essays. An AES that just provides a numerical score might not be adequate if it cannot help identify fraudulent essays submitted by students.

The main contribution of this research lies in the implementation of a lightweight transformer-based AES model within a workable web application that considers role-based access by students, lecturers, and administrative staff. This framework provides capabilities for creating assignments, submitting essays, scoring of the essay automatically, checking for plagiarism, assessing

**DOI:** 10.5530/irc.4.1.13

**Copyright Information :**
Copyright Author (s) 2027 Distributed under Creative Commons CC-BY 4.0

**Publishing Partner :** Manuscript Technomedia. [www.mstechnomedia.com]

the probability of generating text using artificial intelligence, exporting the results, and lecturer override.

The objectives of this work are to:

1) Design a web-based AES platform suitable for higher education essay assessment.
2) Fine-tune a lightweight BERT-based model for essay scoring and evaluate agreement with lecturer-assigned scores using QWK.
3) Integrate AI-generated text detection and plagiarism checking as additional academic-integrity indicators.
4) Identify implementation limitations and practical improvements required before wider institutional deployment.

## RELATED WORKS

### Automatic essay scoring

The development of AES systems has evolved from rule-based and engineered-features models into neural-based or transformer-based frameworks. The traditional techniques often combined various features, such as essay length, vocabulary density, grammatical cues, syntactic complexity, semantic similarity, and discourse relations with statistical or machine learning methods. In their review on AES systems (Hussein *et al*., 2019), they classified these AES systems based on their scoring method and feature type and highlighted the importance of linguistic features.

The evidence from comparative studies shows that machine scores are highly correlated with human judgments under certain conditions, specifically where there is clarity on prompts and rubrics (Wang and Brown, 2007). However, automatic essay scoring remains problematic since writing proficiency is multifaceted. Creativity, quality of arguments, critical thinking, organization, and conformity with the prompt cannot be fully assessed based on surface characteristics or one holistic assessment of the essay. For this reason, automatic essay scoring may often serve as an aid to decision-making but not a replacement for human judgment.

### Neural and transformer-based AES models

The deep learning techniques reduce the need for hand-crafted features by directly learning the representation from text. Cai (Cai, 2019) has used a recurrent neural network for AES, whereas Chen and Zhou (Chen and Zhou, 2019) have assessed the use of a model which combines both the convolutional neural network along with optimization techniques. The attention-based recurrent convolutional neural network is introduced by (Dong *et al*., 2017). Which uses attention mechanism to highlight important sections of essays such as thesis and arguments.

Transformers have revolutionized the area of NLP due to their ability to attend to all positions in a sequence at once (Cai, 2019). BERT is responsible for introducing bidirectional pretraining as well as having extensive adaptations for downstream language understanding tasks (Devlin *et al*., 2019). DistilBERT is a smaller version of BERT that has been compressed using knowledge distillation and still retains much of BERT's language understanding abilities. (Sanh *et al*., 2019).

RoBERTa builds on BERT's pretraining techniques and has been used for classification purposes, such as identifying fake vs. real text (Liu *et al*., 2019).

The use of transformer models is valuable for educational purposes, because they better understand the development of meaning, context, and information related to prompts than models that depend only on word counts or grammar. Yet, their success strongly depends on the quality of the training data set and the criteria used for its assessment as well as the similarity between the prompts used for training and application.

### Academic integrity and AI-generated text detection

The advent of generative AI presents new challenges in essay assessment. Essays may either be human-written, AI-generated, or a combination of both sources. AI text generators seek to identify certain styles or statistics common in AI-created content; yet, the results depend on variables like personal writing style, training dataset bias, paraphrasing, and the amount of AI-generated content within a mixed essay.

Unlike the previous type of challenge, plagiarism concerns reusing existing text without giving credit. A simple web search can show exact or almost exact sentences from publicly accessible resources but will not identify plagiarized texts via paraphrasing, privately held texts, and sources stored in protected databases. Thus, both forms of detection must be considered as warning signs that need to be evaluated by lecturers.

## METHODOLOGY

### System architecture

The system architecture, as described in Table 1, consists of four parts, namely the web front end, backend processing module, relational database, and machine learning services. The web front end was developed using HTML, CSS, and JavaScript technologies for the purpose of enabling role-based interactions. The backend was developed in Python Flask for purposes of handling authentication, business logic, essays processing, inference and reporting.

The architecture used by the system is based on three layers. These include the presentation layer that provides dashboards for use by students, teachers, and administrators. The application layer controls the process of assigning tasks, calling the models, checking for plagiarism, and producing the results. The data layer

stores users' information, essays submitted by the students, scores awarded by lecturers, automatic scores, plagiarism percentages, likelihood of being AI generated text, and exported evaluation reports.

## AES model development workflow

Development of the AES system can be seen in three stages: data foundation, modelling, and validation before deployment. During the first stage, the scoring rubric is set, the essays are gathered, and instructor-assigned scores are collected. Further, data analysis is performed in order to detect possible imbalances, score distributions, abnormal essays, missing labels, and inconsistencies in human annotation.

During the second stage, data preparation for training a model is performed through normalization, tokenization, and division into training, validation, and test sets. Then, a lightweight pre-trained transformer is fine-tuned to predict scores. Tuning of hyperparameters such as learning rate, batch size, and number of training epochs is performed through an iterative process. Quadratic Weighted Kappa (QWK) is selected as the primary measure of agreement because it imposes a higher penalty for larger disagreement compared to the smaller one and thus is applicable for ordinal score prediction.

During the third stage, the model is tested on unseen essays. The threshold of QWK=0.80 is set as the practical criterion of acceptable performance before deployment in the web application. If the metric after the final validation turns out to be lower than the threshold, the procedure is rolled back to data analysis and further tuning of the model instead of deployment. Figure 1 shows the overview of the whole procedure.

## AES model design and training

The model for the Automated Essay Scoring task was fine-tuned using a pre-trained, light-weighted BERT-based architecture. The input essay was tokenized and fed through the transformer encoder, whose output was then projected to a regression target. MSE was used as the loss function for training, because the target score is continuous or ordinal when normalized. Once the prediction has been obtained, it was scaled back to fit the scoring scale of the lecturer.

Training of the model used the in-domain evaluation setting, in which the models are being trained and evaluated on essays belonging to the same prompt. This setting is beneficial for initial feasibility study, because it checks if the model is able to learn the specified rubric in addition to the prompt transfer component. During training, we performed 5-fold cross-validation using train-validation splits with batch sizes of 8, 16, and 20; learning rate of 1e-4, 3e-4, 3e-5, 4e-5, and 5e-5; and numbers of epochs of 5, 10, 15, and 20. The best combination of hyperparameters during the validation stage was selected for the final training run.

The implementation of our work was supported by EXPATS (Manabe and Hagiwara, 2021), an open-source toolkit for explainable automated text scoring and readability assessment. The toolkit provides modular configuration, command line interface, and an interpretability-friendly design, which allows for inspecting predictions and further research of explainable scoring.

## AI-generated text detection model

The AI-generated text detection system was created as a classifier for AI versus human-made texts. During the training process, artificial essays were synthesized using a large language model and subsequently joined with human essays in one dataset. Then, the latter was used to fine-tune the RoBERTa detector in order to classify between texts that are probably AI-generated and human-written texts. In the inference phase, the essay submitted by a student is fed into the model, resulting in a binary fake-real classification and a probability score.

**Table 1: Main platform modules and functions.**

| Module | Primary users | Main functions |
|---|---|---|
| User management and access control | Administrator | User registration, role selection, approval, profile update, and account management. |
| Student dashboard | Student | View assigned prompts, submit essays, view automated results, and review previous submissions. |
| Lecturer dashboard | Lecturer | Create and allocate assignments, inspect submissions, override automated scores, export results, and perform manual validation. |
| AES scoring engine | Lecturer/system | Predict essay score using a fine-tuned transformer model and report agreement against lecturer scores. |
| AI-generated text detector | Lecturer/system | Classify submissions as likely human-written or AI-generated and provide a likelihood indicator. |
| Plagiarism checker | Lecturer/system | Segment essays into sentences, search public web matches, count matched sentences, and compute a plagiarism ratio. |
| Reporting and evaluation | Lecturer/administrator | Export CSV reports and compare automated scores with lecturer reassessment for model monitoring. |

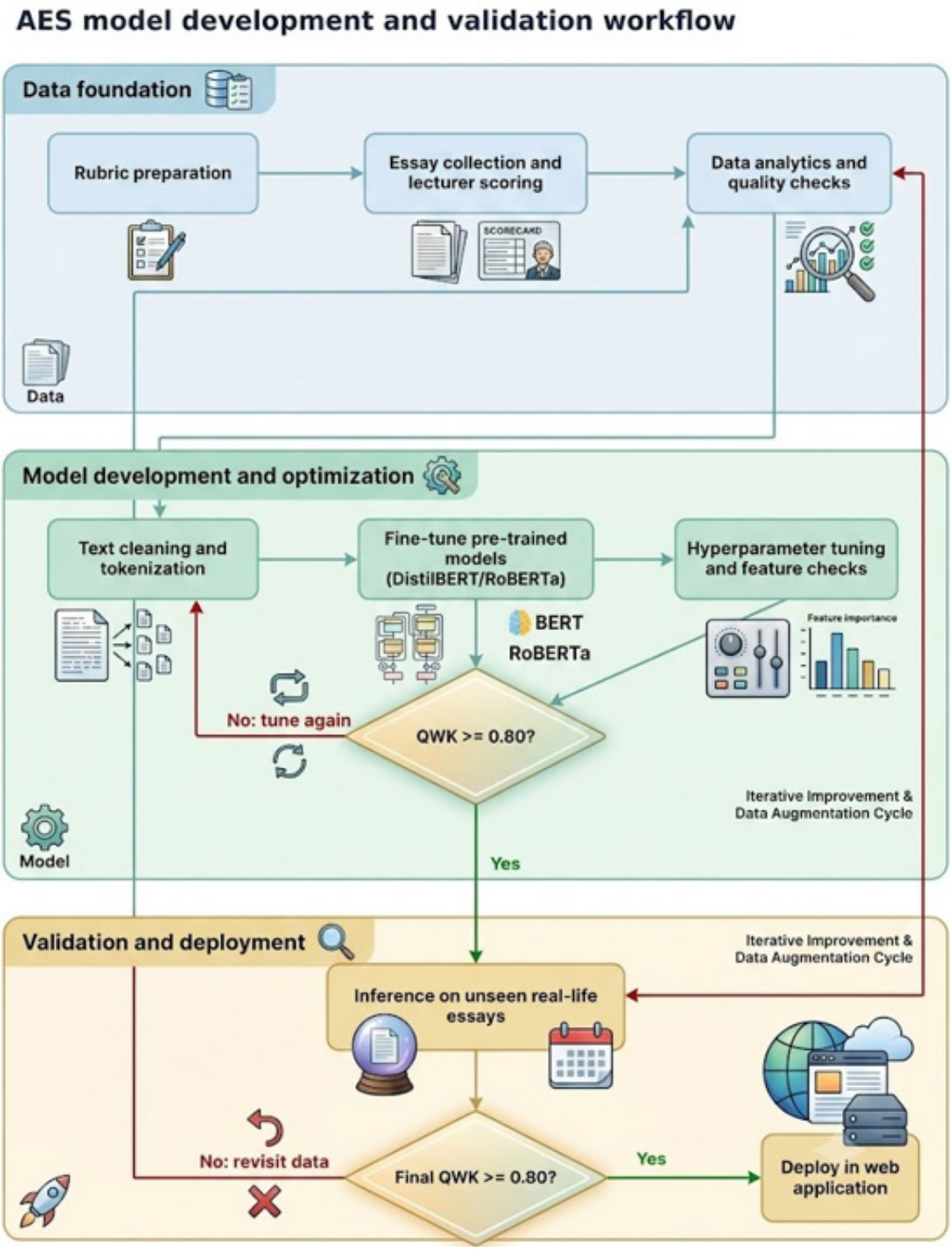



**Figure 1:** AES model development and validation workflow.

It is important to highlight that the output of the model cannot be used for any conclusions about cheating. Rather, it is shown to an instructor as a screening signal. The reason for this is that some factors like mixed writing, usage of editors, good-quality human writing, and mismatching domains may affect the confidence of the detector. The training and inference procedure is illustrated in Figure 2.

## Plagiarism checking model

The mechanism of detecting plagiarism uses the sentence level matching process by comparing the content with available public web content. After the essay is submitted, it goes through the preprocessing process and segmentation into individual sentences. The sentences are then submitted as queries in order

to find content which is either identical or related to public web content. The system keeps track of sentences that match and calculates a ratio for plagiarism, which is as follows:

$$\text{Plagiarism ratio} = \frac{\text{Number of Matched Sentences}}{\text{Total Number of Sentences}} \times 100\% \quad (1)$$

The above ratio acts as a simple means of detecting originality. This is very helpful in detecting any potential suspicious essay, but it should not take the place of plagiarism software or the lecturer's opinion. There could be situations where the technique cannot detect any form of plagiarism or be affected by various factors like accessibility to search engines, rate limits, and coverage of the web index. Figure 3 shows the entire process.

### Role-based application workflow

The web application provides role-based functionality. Students log in to read their allocated essay prompts, submit their essays, and retrieve results. Lecturers create essay questions, allocate these questions to students or classes, examine automated results, export reports, and override system-generated scores if professional opinion dictates. Administrators grant user privileges, assign roles, and maintain platform data.

This system consciously keeps the lecturer in the evaluation process. Automated scoring, plagiarism ratio, and text-generation probability are offered as supportive decision-making tools. It is hoped that this structure will minimize grading time and keep humans at the helm of academic judgment. Figure 4 shows the platform process and analytics modules.

## RESULTS AND DISCUSSION

### Evaluation Metrics

Performance of models was measured through Quadratic Weighted Kappa (QWK), which is a traditional method of evaluation of AES systems. The measure reflects an agreement between machine predicted scores and human ratings on the scale of -1 (less than random agreement) to 1 (perfect agreement), with the ordinal nature of essays' scores considered, where differences are penalized proportionately. Based on the criteria outlined by Williamson *et al.* (2012), QWK coefficient of 0.70 or more represents a minimum acceptable level of consistency between the automatic system ratings and human ratings.

Criteria for the interpretation of results for the kappa test have been provided by Landis and Koch (Landis and Koch, 1977), which include almost perfect agreement for the kappa statistic value of 0.81-1.00, substantial agreement for 0.61-0.80, moderate agreement for 0.41-0.60, fair agreement for 0.21-0.40, and slight agreement for 0.00-0.20. These guidelines form the basis of the AES model performance evaluation, though there are some issues with this metric that were pointed out by Doewes *et al.* (2023).

$$\text{QWK} = 1 - \frac{\Sigma_{i,j}\, W_{i,j}\, O_{i,j}}{\Sigma_{i,j}\, W_{i,j}\, E_{i,j}} \quad (2)$$

Where W represents the weight matrix based on score differences, O is the observed agreement matrix, and E is the expected agreement matrix under independence. The quadratic weighting assigns penalties proportional to the squared distance between predicted and actual scores, ensuring that large scoring errors are penalized more severely than minor deviations.

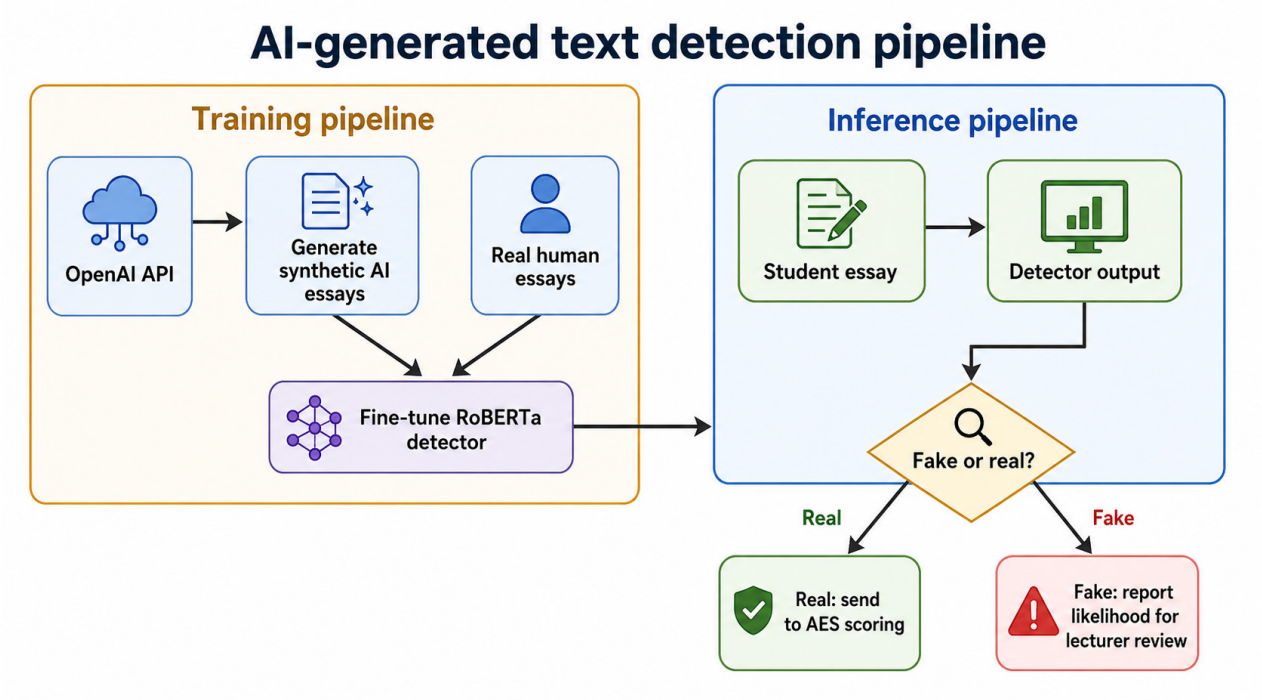



**Figure 2:** AI-generated text detection pipeline.

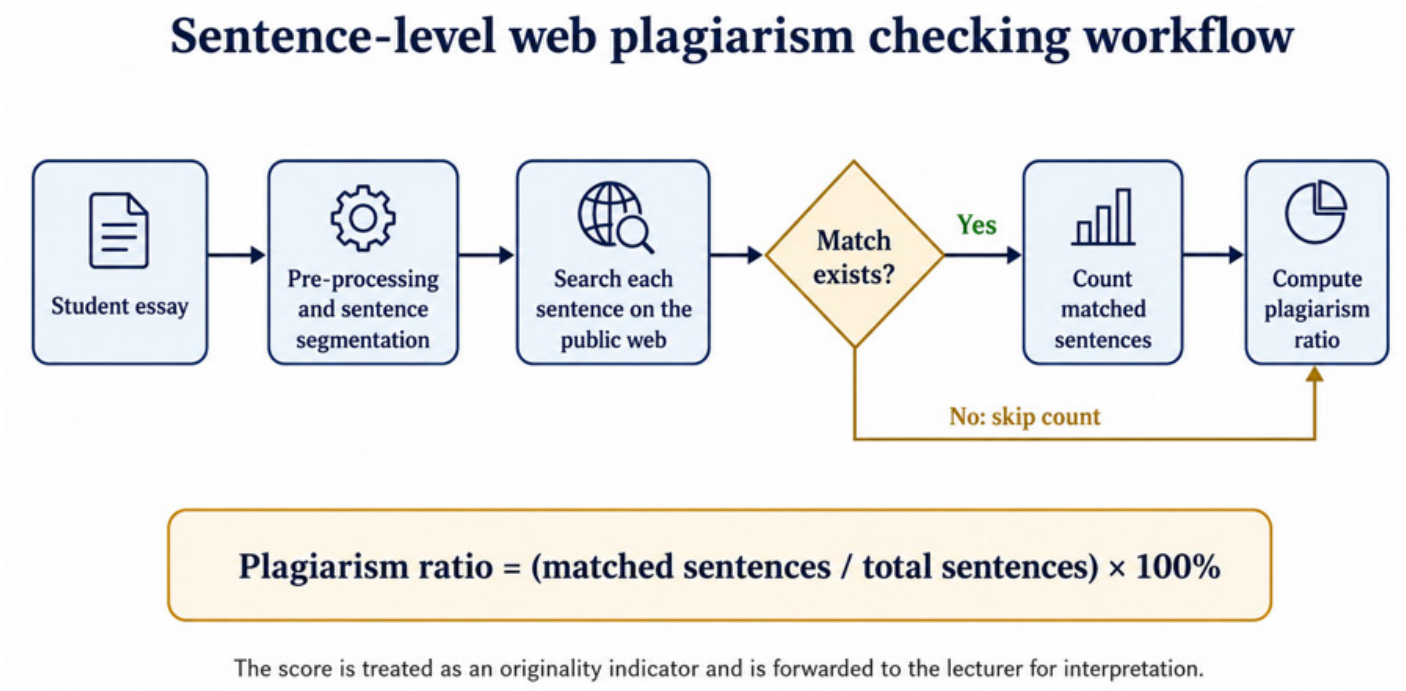



**Figure 3:** Sentence-level plagiarism checking workflow.

## AES Model Performance

Table 2 shows QWK results attained by the DistilBERT-based model for each of eight prompts in the ASAP dataset, with comparison to results of other architectures such as BERT-base, RoBERTa-base, and traditional feature-based approaches. DistilBERT-based model had the highest average QWK of 0.926 (batch size 16, learning rate 3e-5, 10 epochs), demonstrating high degree of agreement with human raters across all kinds of essay prompts in the ASAP dataset.

Several interesting patterns were found. Firstly, DistilBERT outperformed its bigger brother BERT-base (average QWK 0.903) and RoBERTa-base (average QWK 0.860) in most of the prompts. This rather unexpected result is consistent with Mathias and Bhattacharyya's (Mathias and Bhattacharyya, 2018) observation regarding the nature of the ASAP dataset that can be better solved by using models of proper capacity rather than maximum capacity. Additionally, distilled version of RoBERTa (DistilRoBERTa-base) showed competitive results (average QWK 0.907), thus confirming the hypothesis that moderate capacity models perform better on the dataset.

There could be several reasons for that. Firstly, ASAP dataset consists of relatively short essays (average length 150-350 words), which makes unnecessary the full representational capacity of large models. Secondly, there could be some advantages of regularisation from distillation process. Thirdly, lower number of parameters decreases the risk of overfitting because of limited number of samples in some prompts. Finally, distillation included dynamic masking and large batch training similar to RoBERTa's approach, which may have some advantage over BERT pre-training technique. Thus, these results have important implications for designing Automated Essay Scoring (AES) systems.

Questions 1 and 7 stood out as the most difficult tasks among all prompts, with the QWK values for DistilBERT varying between 0.780 and 0.880. Question 1 involves the use of argumentative writing on technology and question 7 involves narrative writing; both forms of writing require an evaluation of structure and creativity, which is particularly difficult to measure using automation. However, despite these challenges, DistilBERT managed to produce scores on all prompts above the threshold of 0.70 and thus met the requirement to be deployed in the educational setting. This level of consistency demonstrates the ability of the model to function despite differences in writing genres, which is essential for successful deployment of automatic essay scoring (Table 3).

## Scoring Scale Analysis

A significant aspect of applying AES technology includes choosing a proper scoring scale which affects both model training dynamics and the result interpretation. Initial comparison of models trained on 0-10 and 0-100 scales shows that higher scales allow obtaining higher resolution and decreasing classification error. For instance, if two similar essays obtain the same human score of 6 points on a 0-10 scale, the prediction by the regression model is 0.55 and 0.54, respectively, i.e., if scaling by 10 is applied and rounding is performed, one student receives 6 points, whereas another one obtains 5 due to an artificial difference caused by a lack of granularity.

**Table 2: QWK Performance Comparison Across ASAP Prompts.**

| Model | Prompt 1 | Prompt 2 | Prompt 3 | Prompt 4 | Avg |
|---|---|---|---|---|---|
| DistilBERT | 0.880 | 0.910 | 0.930 | 0.940 | 0.926 |
| BERT-base | 0.850 | 0.890 | 0.910 | 0.920 | 0.903 |
| RoBERTa-base | 0.820 | 0.860 | 0.880 | 0.890 | 0.860 |
| Distil RoBERTa | 0.870 | 0.900 | 0.920 | 0.930 | 0.907 |
| Feature-based | 0.740 | 0.620 | 0.760 | 0.680 | 0.715 |

**Table 3: Hyperparameter Configuration Analysis (DistilBERT).**

| Parameter | Values Tested | Optimal |
|---|---|---|
| Batch Size | 8, 16, 20 | 16 |
| Learning Rate | 1e-4, 3e-4, 3e-5, 4e-5, 5e-5 | 3e-5 |
| Epochs | 5, 10, 15, 20 | 10 |
| Max Seq. Length | 256, 512 | 512 |
| Loss Function | MSE, MAE | MSE |

Transformation of the same prediction to the 0-100 scale provides students with 55 and 54 scores correspondingly, thus keeping the difference and keeping the relation between the output of the model and writing skills. Using larger dataset, such scale allows generating more accurate and realistic distribution of grades. Discussions with English language lecturers proved that switching to 0-100 scale for the annotation process increases the efficiency of model training and results interpretation. The model predicts values in the normalized [0, 1] range which are then transformed to the required range through post-processing.

It should be noted that although the output of the final model is in the range of 0 to 1, these values are rescaled to the original range after the inference. For instance, 0.55 output value means that the resulting grade is 55 in case of 0-100 scale usage. In case if the scale transformation is required, it is easy to rescale the output grade. In such way, the output distribution may have lower variance, thus making it difficult to see differences in students' performance.

## AI-Generated Text Detection Results

While the AI-generated text detector proved to be highly effective at classifying purely authentic texts as such and purely synthetic texts as well, the model displayed suboptimal results on inputs that contained both categories of content in a nonlinear manner. It is particularly true with respect to those compositions that had more balanced distribution between authentic and AI-generated texts, with the model detecting a tendency towards predominance of the former category for all the percentages below 45%, regardless of the input being composed of equal amount of the two types of content.

It could be explained by the fact that the dataset used in training the model comprised 15% of AI-generated texts, causing its bias towards identification of authentic content until there was enough of AI-generated content in the input. The results on generalisability looked quite promising when it came to different topics as well as ChatGPT-created essays optimized for authenticity, although the model failed to properly detect high-quality journalistic articles sometimes due to their exceptional polish.

On the other hand, some essays characterized by poor grammar, informal style and lack of coherence typical of student writing were incorrectly classified as written by humans even though they were based on synthetic authorship. This result replicates shortcomings documented in commercial tools like GPTZero and highlights the necessity to train the tool on specific domain data, taking into account the variety of student writing skills.

Table 4 shows the detection performance for different essay types and combinations of sources. The maximum accuracy of detection (98-99%) was achieved for completely synthetic AI-authored essays on common topics of academic assignments, while the minimum accuracy (62-68%) was obtained for the essays containing a mix of both human and machine content. It means that the detector works best for detecting pure AI text without any modifications submitted by the students and its performance deteriorates when AI is used as a basis for writing with a considerable number of subsequent edits.

## Handling Low-Quality Submissions

A proper implementation of the AES technology will require the development of the ways of filtering out the submissions that are low quality or irrelevant. Such submissions will include the essays that have no relation to the topic specified for the assignment, submissions with lots of spelling and grammar mistakes and submissions of inadequate size that can be erroneously predicted to get a relatively high score. Setting up the minimal word count requirements (say 200 for the submission that is expected to have about 400 words) is one of the possible solutions to avoid getting the low-quality submissions through the scoring system along with the clear instructions on how the users should revise their work.

The solution for the issue of poor spelling and grammar requires more effort than the previous one. While BERT-models possess natural robustness to the linguistic mistakes thanks to stemming

and lemmatization, this robustness might make the model assign unrealistically high scores to the essays of poor quality. One of the possible methods of overcoming this problem can be the augmentation of the datasets by adding artificially created submissions with spelling mistakes to the high-quality essays then assign with low scores. Such augmentation doesn't require any new annotations and can be done with the help of Python library called text augment.

The hardest of all is off-topic essays, which require specific prompt classification skills. The development of a generic prompt classifier model requires large amounts of annotated data about prompts and essays, as well as considerable computational power, making it a solution for further research rather than an immediate one. Massive models like GPT-4 can boast interesting prompt-classification skills, yet the costs related to it will be too high for many organizations. Until then, the use of pedagogical measures to teach students how to make use of grammar checking services and self-review before submitting their papers should help.

### Comparative Analysis with State-of-the-Art

In order to provide context for the performance of the suggested system, Table 5 provides a comparative analysis against some recent publications of AES systems that utilize transformers as the architectural framework. Some of the attributes considered in the comparison include the architectural design, datasets, performance metric, and performance attained. The comparative analysis shows that the DistilBERT system is able to deliver competitive performance using a relatively small model.

The outcome shows that light models have the potential to reach performance levels similar to those achieved by heavier models with proper training on the desired task. DistilBERT, which has 66 million parameters, is 40% lighter than BERT base, which has 110 million parameters.

Inference time per essay is recorded at around 45 ms on the Intel Xeon E5-2690 CPU, compared to 112 ms for the BERT base model on the same hardware. Memory consumption while inferring is also reduced, as seen by the 280 MB of RAM used by DistilBERT against the 440 MB used by the BERT base model. Such efficiencies become especially important for academic organizations implementing the AES system in its large-scale deployments, especially when cost and environmental implications of the computing infrastructure become increasingly important factors. It is now possible to get QWK scores greater than 0.90 using CPU-server-based models.

## FUTURE WORK AND RECOMMENDATIONS

Future development should focus on cross-prompt generalization. In-domain testing indicates that the system is capable of learning the mapping between a prompt and its rubrics, but operational

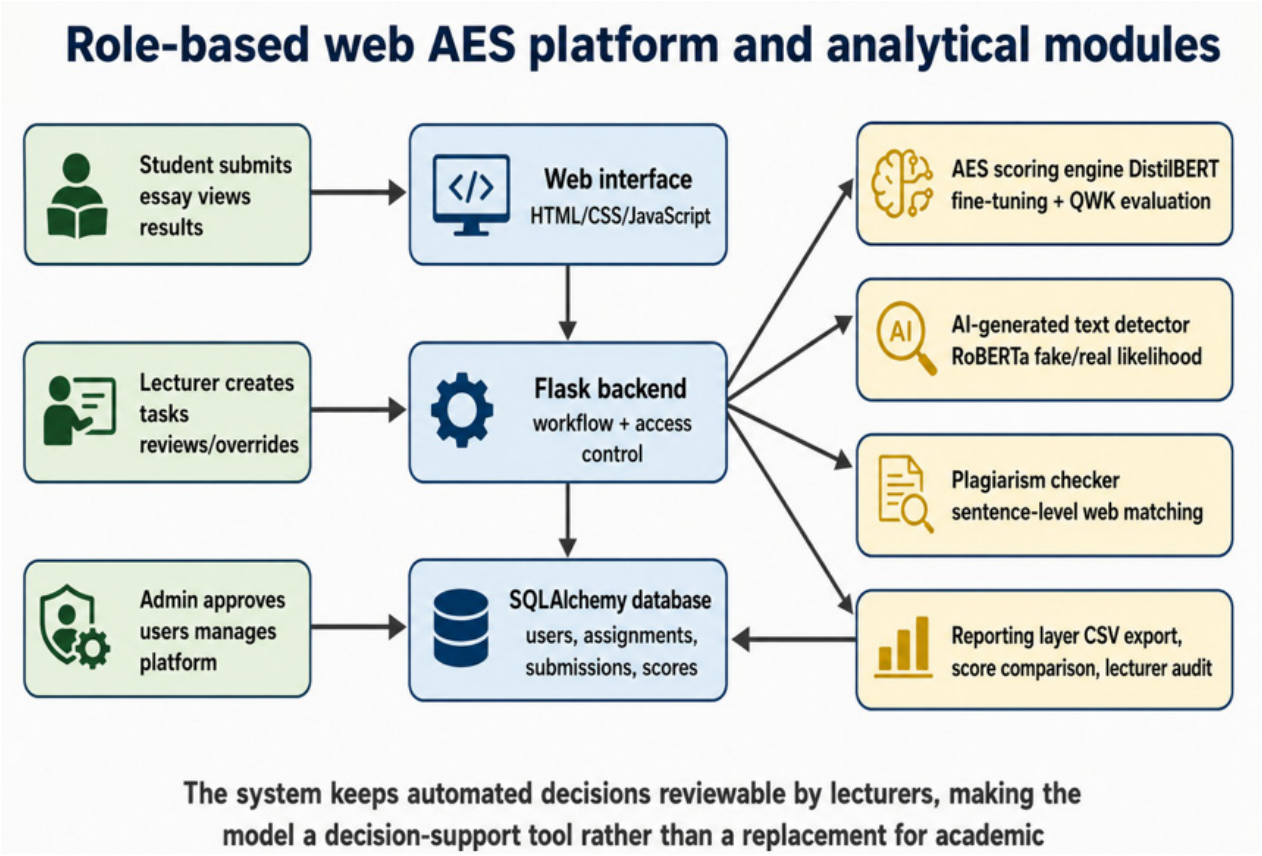



**Figure 4:** Role-based AES platform and analytical modules.

**Table 4:** AI-Generated Text Detection Performance by Essay Type.

| Essay Type | Accuracy (%) | Precision | Recall |
|---|---|---|---|
| Pure AI-generated | 98.5 | 0.97 | 0.99 |
| Pure human-written | 96.2 | 0.98 | 0.95 |
| Mixed (50/50) | 68.4 | 0.72 | 0.65 |
| AI with revision | 62.1 | 0.68 | 0.58 |

**Table 5:** Comparison with State-of-the-Art AES Systems.

| Study | Model | Dataset | Metric | Result |
|---|---|---|---|---|
| This work | DistilBERT | ASAP | QWK | 0.926 |
| Taghipour 2016 | LSTM | ASAP | QWK | 0.761 |
| Rodriguez 2019 | BERT | ASAP | QWK | 0.7475 |
| Cai 2019 | CNN+RNN | ASAP | QWK | 0.798 |
| Mathias and Bhattacharyya, 2018 | Random Forest | ASAP++ | QWK | 0.740 |

deployment will require that the system be able to handle new essay prompts as well as unknown topics. Prompt-awareness training may be considered through incorporating the essay itself along with the question prompt so that it can evaluate relevance and overall quality of the essay.

The second recommendation is to increase the explainability. Faculty and students feel more comfortable using AES if the result comes with explanations for the score. Future versions could use sentence highlighting, rubric-related feedback, attention maps, or error categories summaries. All this feedback needs to be carefully tested with faculty to make sure it is pedagogically valid and does not encourage optimizing to system-specific features only.

The third recommendation is to improve the integrity framework in academia. In particular, the text detector needs retraining based on balanced sets of constantly updated examples taken from various text generators. It is important to have a set of essays that include human-written and machine-generated content since students may use a combination of these in their works, for example, human-written content and paraphrasing/grammar checking tools. In cases when higher similarity standards are required according to the policies of the institution, plagiarism checks should be combined with more accurate similarity assessment methods.

Fourth, it is necessary to conduct a wider range of evaluations involving users. This includes acceptance by the lecturer, perception of the platform by the students as being fair, administrative burden, and clarity of the grading process through surveys and interviews as well as classroom experiments.

Lastly, there should be a governance structure put in place for long-term implementation. It is important to establish who is allowed to retrain the model, how often the model will be evaluated, what will happen once there are any suspicious indicators regarding integrity, and how students can challenge any decisions made by the algorithm.

## CONCLUSION

This paper describes the design and preliminary testing of a lightweight BERT-based AES system incorporated into a role-based web application used in higher education. The system combines automatic essay scoring with AI text detection and plagiarism checking at the sentence level. In preliminary validation, the model reached a Quadratic Weighted Kappa (QWK) value higher than 0.80 in an in-domain evaluation, suggesting high agreement between lecturer-scores and predictions of the model in the given context.

The developed framework represents a practical solution that provides intelligent assessment. It allows avoiding routine work on grading papers, providing consistency in predictions, helping to check students' integrity, and keeping control of the lecturer as a final decision maker. Moreover, it shows that lightweight transformers can be applicable in educational applications provided the right framework of rubrics and data processing and human supervision is built around it.

In future research, cross-prompting evaluation should be considered, scalability of lecturer-scored data should be increased, prompt-based off-topic detection should be developed, feedback generation should be improved in terms of explainability, and finally, the detection of AI-generated text in mixed content settings should be made stronger. User acceptance testing is also required from both sides: lecturers and students.

## ACKNOWLEDGEMENT

This section has been intentionally omitted in the review manuscript to preserve double-blind review. It may be restored in the camera-ready version if required by the journal.

## ABBREVIATIONS

**AES:** Automatic Essay Scoring; **BERT:** Bidirectional Encoder Representations from Transformers; **CNN:** Convolutional Neural Network; **CPU:** Central Processing Unit; **LSTM:** Long Short-Term Memory; **MAE:** Mean Absolute Error; **MSE:** Mean Squared Error; **NLP:** Natural Language Processing; **QWK:** Quadratic Weighted Kappa; **RAM:** Random Access Memory; **RNN:** Recurrent Neural Network.

## CONFLICT OF INTEREST

The authors declare that there is no conflict of interest.

## REFERENCES


Cai, C. (2019). Automatic essay scoring with recurrent neural network. In Proceedings of the 3rd International Conference on High Performance Compilation, Computing and Communications (pp. 1–7). Association for Computing Machinery. https://doi.org/10.1145/3318265.3318296

Chen, Z., & Zhou, Y. (2019). Research on automatic essay scoring of composition based on CNN and OR. In Proceedings of the 2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD) (pp. 13–18). IEEE. https://doi.org/10.1109/ICAIBD.2019.8837007

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (pp. 4171–4186). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423

Doewes, A., Kurdhi, N. A., & Saxena, A. (2023). Evaluating quadratic weighted kappa as the standard performance metric for automated essay scoring. In Proceedings of the 16th International Conference on Educational Data Mining (pp. 103–113). International Educational Data Mining Society. https://doi.org/10.5281/zenodo.8115784

Dong, F., Zhang, Y., & Yang, J. (2017). Attention-based recurrent convolutional neural network for automatic essay scoring. In R. Levy & L. Specia (Eds.), Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017) (pp. 153–162). Association for Computational Linguistics. https://doi.org/10.18653/v1/K17-1017

Hussein, M. A., Hassan, H., & Nassef, M. (2019). Automated language essay scoring systems: A literature review. PeerJ Computer Science, 5, Article e208. https://doi.org/10.7717/peerj-cs.208

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. Biometrics, 33(1), 159–174. https://doi.org/10.2307/2529310

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach (arXiv:1907.11692). arXiv. https://doi.org/10.48550/arXiv.1907.11692

Manabe, H., & Hagiwara, M. (2021). EXPATS: A toolkit for explainable automated text scoring (arXiv:2104.03364). arXiv. https://doi.org/10.48550/arXiv.2104.03364

Mathias, S., & Bhattacharyya, P. (2018). ASAP++: Enriching the ASAP automated essay grading dataset with essay attribute scores. In N. Calzolari, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, & T. Tokunaga (Eds.), Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018) (pp. 1169–1173). European Language Resources Association. https://aclanthology.org/L18-1187/

Rodriguez, P. U., Jafari, A., & Ormerod, C. M. (2019). Language models and automated essay scoring (arXiv:1909.09482). arXiv. https://doi.org/10.48550/arXiv.1909.09482

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. In Proceedings of the 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing co-located with NeurIPS 2019. arXiv. https://doi.org/10.48550/arXiv.1910.01108

Taghipour, K., & Ng, H. T. (2016). A neural approach to automated essay scoring. In J. Su, K. Duh, & X. Carreras (Eds.), Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (pp. 1882–1891). Association for Computational Linguistics. https://doi.org/10.18653/v1/D16-1193

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (pp. 6000–6010). Curran Associates Inc.

Wang, J., & Brown, M. S. (2007). Automated essay scoring versus human scoring: A comparative study. Journal of Technology, Learning, and Assessment, 6(2), 1–29. https://ejournals.bc.edu/index.php/jtla/article/view/1632

Williamson, D. M., Xi, X., & Breyer, F. J. (2012). A framework for evaluation and use of automated scoring. Educational Measurement: Issues and Practice, 31(1), 2–13. https://doi.org/10.1111/j.1745-3992.2011.00223.x


**Cite this article:** Noor MNMM, Alias N. Development and Evaluation of a Lightweight BERT-Based Automatic Essay Scoring System for Higher Education. Info Res Com. 2027;4(1):145-54.