DOI: 10.71317/jgst.2.7.2026.263
Journal of Global Social Transformation
Journal homepage: https://rjsaonline.org/index.php/JGST

A Model-Agnostic Framework for Transparent, Fair, and Reproducible Automated Essay Scoring

Ahsan Javed[1]
[1] IU international University of Applied Sciences
Email: ahsanjaved997@outlook.com

ARTICLE INFO

Submission:
June 05, 2026
Revised:
June 20, 2026
Accepted:
July 06, 2026
Available Online:
July 21, 2026

Keywords:
Automated Essay Scoring (AES), Explainable AI (XAI), Natural Language Processing (NLP), Algorithmic Fairness, Reproducibility, Transformer Models, DeBERTa, Prompt-Aware Binning

Corresponding author:
ahsanjaved997@outlook.com

ABSTRACT

The most accurate automated essay scoring systems function as opaque black boxes. This lack of transparency presents a significant obstacle to trust among the students, teachers, and policymakers who rely on these tools, as fairness and accountability are non-negotiable. The design and evaluation of a model-agnostic framework for explainable AI in essay scoring are detailed, with a foundational focus on reproducibility and algorithmic fairness. By addressing these longstanding challenges, the proposed framework delivers a tool that is understandable and trustworthy for students, educators, and policymakers. The contributions of this work are threefold. First, a prompt-aware binning strategy is introduced to normalize essay scores into three proficiency tiers, which enhances model stability and enables a more meaningful fairness analysis. Second, both classic and state-of-the-art transformer models are benchmarked within a reproducible pipeline that utilizes deterministic controls and a manifest system to track every data artifact, ensuring that all findings are auditable and verifiable. Third, rather than focusing on fairness or interpretability in isolation, this work integrates both dimensions, using SHAP and LIME to dissect model logic and to identify and measure disparities across demographic groups. On the ASAP2.0 dataset, a fine-tuned DeBERTa-v3-large model achieves a macro-F1 of 0.898. An XGBoost model trained on a fused set of linguistic and transformer features achieves a competitive 0.890. This framework aims to set a new standard for transparent and trustworthy AI in educational assessment and offers a foundation for future research in explainable educational technology.

Introduction

Automated essay scoring (AES) has changed how student writing is evaluated. The evolution of AES mirrors larger trends in natural language processing (NLP), beginning with early systems like Project Essay Grade (PEG) in the 1960s. PEG relied on proxy features such as essay length, word choice, and other surface-level statistics to approximate writing quality (Page, 1966). This was followed by more sophisticated feature-engineering approaches in commercial systems like e-rater and IntelliMetric during the 1990s and 2000s (Attali & Burstein, 2004). These earlier systems, while groundbreaking for their time, were limited by their reliance on hand-crafted, surface-level linguistic features. Their models could capture aspects of grammar, mechanics, and structure, but they struggled to grasp the deeper semantic content and argumentative coherence that are the hallmarks of proficient writing.

The deep learning revolution, particularly the advent of transformer architectures like BERT, has fundamentally altered this landscape (Devlin et al., 2019). By pre-training on vast corpora of text, these models learn rich, contextual representations of language. Their self-attention mechanisms allow them to weigh the importance of different words in a sentence, capturing complex dependencies and nuances inaccessible to previous methods. With this leap in representational power, however,

comes a new kind of opacity. As a result, modern AES systems built on transformers have made huge leaps in accuracy, sometimes matching or even exceeding the agreement rates of human graders (Rodriguez et al., 2019; R. Yang et al., 2020).

Motivation and Key Challenges in Existing Systems

However, as these models increase in predictive power, they often decrease in transparency. The very mechanisms that make them powerful also make them opaque. The knowledge is distributed across millions or billions of parameters in a way that is not directly legible to humans. It is difficult for any stakeholder, whether a teacher, student, or researcher, to unpack why a score was assigned. This article addresses that trade-off between performance and interpretability.

This challenge is compounded by the inherent complexity of the data itself. The widely used ASAP 2.0 dataset, which forms the empirical basis of this work, is not a monolithic entity. It comprises multiple distinct essay prompts, each with its own unique characteristics. Exploratory data analysis reveals that the number of essays per prompt varies dramatically.

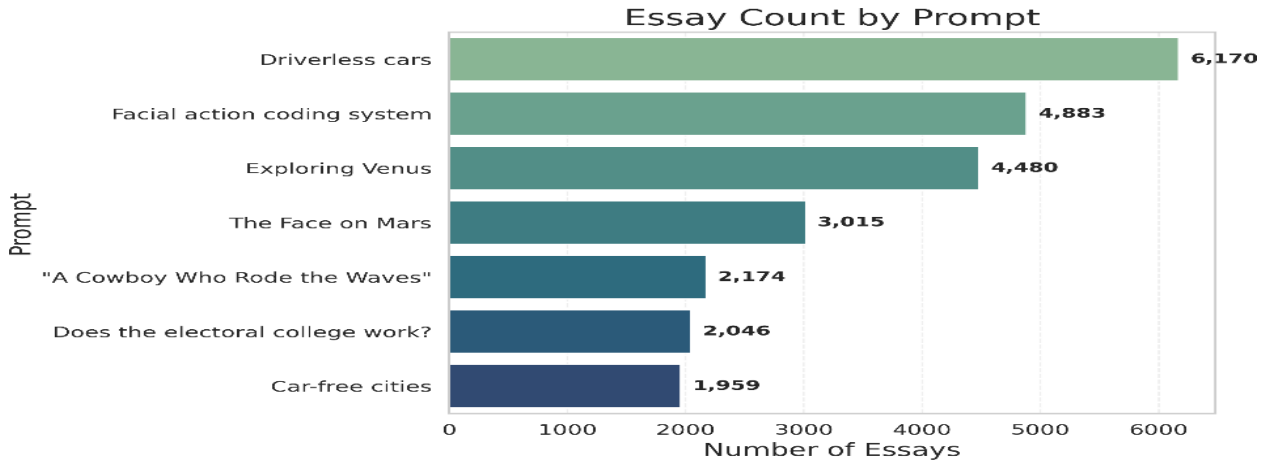


Figure 1: Essay Count by Prompt

The data is far from uniform, ranging from nearly 6,200 essays for "Driverless cars" to under 2,000 for "Car-free cities". This imbalance means a model trained naively on the aggregate data will be disproportionately influenced by the characteristics of the most frequent prompts.

More importantly, the scoring standards are inconsistent across prompts. A raw score of '3' on one prompt might signify average performance, while on another, it could be well below average.

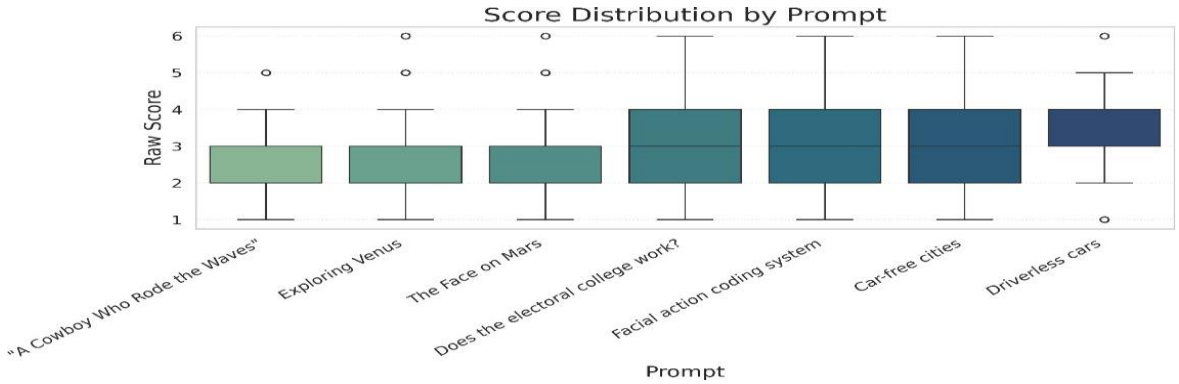


Figure 2: Score Distribution by Prompt

Figure 2 shows that both the range and median of raw scores differ substantially from one prompt to the next. For example, the prompt "A Cowboy Who Rode the Waves" has a tight distribution centred around a score of 2–3, whereas "Driverless cars" exhibits a much wider spread with a median of 3–4.

This heterogeneity poses a fundamental problem for building a single, general-purpose AES model. A model trained on this inconsistent data may learn to associate certain scores with prompt-specific artifacts rather than genuine writing quality. This creates a spurious correlation that threatens the model's validity and makes its decisions even harder to interpret.

Explainability in AES is more than a technical puzzle; it is an ethical requirement. The guidelines of the IU International University of Applied Sciences are clear: students are responsible for their work, and any AI-generated content must be critically reviewed (IU, 2025). If a model is a black box, it is impossible for a student to learn from its feedback, or for a teacher to contest a grade that feels wrong. The principle of "critical review" implies a dialogue, a process of understanding and questioning. An opaque score provides no basis for such a dialogue. The development of an explainable AI framework is a direct response to the need for tools that students and educators can trust and understand.

This ethical mandate extends to the critical issue of fairness. AI models are known to learn and amplify biases present in their training data (Noble, 2018). If an AES system performs differently for students from different backgrounds, it fails its primary purpose as an objective assessment tool and risks becoming a mechanism for perpetuating inequality. This is not a hypothetical risk; biased models can reinforce stereotypes, disadvantage already marginalised groups, and erode trust in education systems. A biased model might, for example, systematically under-score essays that use culturally specific language or references unfamiliar to the original graders. The ASAP 2.0 dataset contains demographic information that allows for an audit of this very issue.

Demographic Category	Group	Count	Percentage
Gender	Male	12,515	50.61%
	Female	12,212	49.39%
Race/Ethnicity	White	9,867	39.91%
	Hispanic/Latino	7,183	29.05%
	Black/African American	4,587	18.55%
	Two or more races/Other	1,525	6.17%
	Asian/Pacific Islander	1,417	5.73%
	American Indian/Alaskan Native	144	0.58%
Economically Disadvantaged	Economically disadvantaged	12,785	61.70%
	Not economically disadvantaged	7,937	38.30%

Table 1: ASAP 2.0 Dataset Demographics Summary

These distributions are used throughout the analyses of fairness and model performance. They are not just statistics; they represent the student populations that will be subject to the model's judgments. The potential for bias is significant. For example, if certain prompts contain culturally specific content, or if the human raters who produced the original scores held implicit biases, the model will learn to replicate these patterns. A core part of this work is to explicitly measure model performance across these groups to ensure that the pursuit of accuracy does not come at the cost of equity.

Problem Statement

Current Automated Essay Scoring (AES) systems based on deep learning achieve high predictive accuracy; however, they continue to face several significant challenges. One of the primary limitations is the lack of transparency in their decision-making processes, making it difficult for users to understand score assignments, receive meaningful feedback, or challenge evaluation outcomes. In addition, these systems are susceptible to prompt-specific biases caused by variations in scoring rubrics and the uneven distribution of prompt data in benchmark datasets such as ASAP 2.0, which can negatively affect their generalizability and consistency. Furthermore, demographic bias remains a critical concern, as model performance may vary across different gender, ethnicity, and socio-economic groups, raising questions about fairness and equity in automated assessment. Addressing these challenges requires the development of an AES framework that is transparent, fair, reproducible, and capable of adapting effectively to diverse essay prompts while ensuring consistent and unbiased evaluation across different learner populations.

Main Research Objective

To develop and evaluate a transparent, reproducible, and fair automated essay scoring framework that integrates explainable artificial intelligence techniques with transformer-based models to improve scoring performance while ensuring interpretability and demographic fairness across multiple essay prompts.

Research Questions

This article is structured around five main objectives: to benchmark classic and deep AES models on the ASAP 2.0 dataset; to design and test a prompt-aware binning strategy for both performance and fairness; to audit the interpretability of model

decisions with SHAP and LIME; to establish and validate a reproducibility protocol so that the work can be checked and extended; and finally, to put the findings into context by comparing them with global AES benchmarks.

1. How do classic and transformer-based models compare when using prompt-aware, three-level binning?
2. Does prompt-aware binning improve fairness or demographic balance in scoring?
3. What can SHAP and LIME reveal about how these models make their decisions?
4. How reproducible are the results when every step is logged and controlled?
5. Where does this work fit among the best published AES results, and what new challenges does it uncover?

Literature Review

The Evolution of Automated Essay Scoring

The pursuit of automated writing evaluation is nearly as old as modern computing itself. The field has evolved through several distinct paradigms, each defined by the prevailing technological and theoretical approaches of its time. Understanding this evolution is crucial for contextualizing the contributions of the present work, as it highlights the persistent trade-off between the sophistication of the models and their inherent transparency.

The Pioneering Era: Surface-Level Feature Engineering

The earliest attempts at Automated Essay Scoring (AES), dating back to the 1960s, were characterized by a focus on easily quantifiable, surface-level features of a text. The seminal work in this area was Ellis Page's Project Essay Grade (PEG) in 1966. PEG operated on a simple yet revolutionary premise: that intrinsic qualities of good writing (what Page termed "trins") could be approximated by a set of extrinsic, measurable variables ("proxes"). These proxes included metrics such as essay length, word count, number of commas, and the use of certain prepositions (Page, 1966). A multiple regression model was then used to find the optimal weighting of these features to predict the scores assigned by human raters.

Subsequent systems in this era, such as the commercial e-rater developed by the Educational Testing Service (ETS), refined this feature-engineering approach. E-rater incorporated more sophisticated NLP techniques, including part-of-speech tagging and syntactic parsing, to extract features related to grammar, usage, mechanics, and style (Attali & Burstein, 2004). It also introduced a rudimentary form of content analysis by comparing the vocabulary of a student's essay to that of a model essay for the same prompt. These systems represented a significant step forward, moving beyond simple proxies to a more nuanced, rubric-aligned evaluation. However, they still struggled to capture the high-level, abstract qualities of writing, such as rhetorical effectiveness, logical reasoning, and creativity. Their logic, while more complex than PEG's, was still based on a set of explicit, human-defined features.

The Statistical and Machine Learning Era

The 2000s saw a shift away from purely rule-based and feature-driven systems towards more sophisticated statistical and machine learning models. This era was marked by the use of techniques like Latent Semantic Analysis (LSA), which allowed for a more robust form of content analysis (Landauer et al., 1998). LSA uses singular value decomposition to create a high-dimensional "semantic space" from a large corpus of text. The meaning of a word or a document is represented as a vector in this space. In the context of AES, the "content" of a student's essay could be represented as a vector and compared to the vector of a gold-standard essay, providing a more reliable measure of topical relevance than simple keyword matching.

This period also saw the application of a wider range of machine learning algorithms, including Support Vector Machines (SVMs), decision trees, and Bayesian classifiers. These models were capable of learning more complex, non-linear relationships between features and scores. The focus of research shifted towards identifying and engineering the most predictive set of features, which often included a combination of surface-level metrics, syntactic features, and content-based features derived from techniques like LSA. While these models offered improved accuracy, they also represented a step away from the inherent transparency of the early rule-based systems, marking the beginning of the "black box" problem in AES.

The Deep Learning Revolution: From LSTMs to Transformers

The most recent and transformative era in AES has been driven by the deep learning revolution in NLP. Early deep learning approaches utilized Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) networks, to model the sequential nature of text (Hochreiter & Schmidhuber, 1997). LSTMs could process an essay word by word, maintaining an internal "memory" of the preceding context. This allowed them to capture long-range dependencies and generate a holistic

representation of the entire essay, which could then be used to predict a score. This approach, pioneered by researchers like Taghipour and Ng (2016), eliminated the need for extensive manual feature engineering, as the model could learn relevant features directly from the raw text.

Models like BERT and its successors (e.g., RoBERTa, DeBERTa) have consistently set new state-of-the-art performance benchmarks in AES (He et al., 2020; Uto et al., 2020). Their ability to capture subtle semantic nuances and complex syntactic structures has made them incredibly powerful. However, this power comes at the cost of interpretability. The very complexity that makes these models so effective also turns them into "black boxes," which is the central problem that this article aims to address. The knowledge learned by these models is distributed across billions of parameters, making it impossible to trace a direct, human-understandable line of reasoning from the input text to the final score.

Explainable AI in Natural Language Processing

The Need for Interpretability: Model-Specific vs. Model-Agnostic Approaches

XAI methods can be broadly categorized into two groups: model-specific and model-agnostic. Model-specific explanation methods are designed for a particular class of models and often leverage the internal architecture of the model to generate explanations. For example, the attention mechanism in a transformer can itself be seen as a form of explanation. By visualizing the attention weights, one can see which words the model "paid attention to" when making a prediction (Vig, 2019). This can provide some insight into the model's focus. However, the relationship between attention and feature importance is not always straightforward, and research has shown that these visualizations can sometimes be misleading or fail to capture the true reasoning of the model (Jain & Wallace, 2019).

Model-agnostic methods, on the other hand, can be applied to any machine learning model, as they treat the model as a black box. These methods work by systematically perturbing the input to the model and observing how the output changes, thereby inferring which features were most influential. This approach makes them particularly well-suited for the framework of this article, which aims to compare the interpretability of fundamentally different model types (e.g., linear models, tree-based ensembles, and transformers). The two most prominent model-agnostic techniques, and the ones used in the interpretability audit in Chapter 4, are LIME and SHAP.

Local Explanations: The LIME Framework

LIME, which stands for Local Interpretable Model-agnostic Explanations, is a technique that explains the prediction of any classifier by learning an interpretable model locally around the prediction (Ribeiro et al., 2016a). The core idea behind LIME is that while a global model might be very complex and non-linear over the entire data distribution, its behaviour in the immediate vicinity of a single data point can often be accurately approximated by a much simpler, linear model.

To generate an explanation for a single essay, LIME creates a new dataset of "perturbed" versions of that essay. For text, this typically involves removing a random subset of words from the original sentence. It then gets predictions from the black box model for each of these perturbed instances. Finally, it trains a simple, interpretable model (like a linear regression with L2 regularization) on this new dataset, where the features are the presence or absence of words and the target is the black box model's prediction. The instances are weighted by their proximity to the original essay, giving more importance to perturbations that are more similar to the original. The learned coefficients of this simple model are then presented as the explanation, indicating which words or features had the most positive or negative influence on the original prediction. The primary strength of LIME is its intuitive, instance-level explanations that are easy for a human to understand. Its main weakness is that the explanations can be unstable, as they depend on the random nature of the perturbations and the simplicity of the local model.

Global and Local Explanations: The SHAP Framework and Shapley Values

SHAP (SHapley Additive exPlanations) is a unified approach to explaining the output of any machine learning model, based on the concept of Shapley values from cooperative game theory (Lundberg & Lee, 2017). The Shapley value is a method for fairly distributing the "payout" of a game among its players, where each player's contribution is considered. In the context of machine learning, the "game" is the prediction task, the "players" are the input features, and the "payout" is the difference between the model's actual prediction for an instance and the average prediction across all instances.

The SHAP value for a feature is its average marginal contribution to the prediction across all possible combinations (or "coalitions") of other features. This provides a solid theoretical grounding for the explanations and guarantees certain desirable properties, such as local accuracy (the sum of the SHAP values for all features equals the difference between the

prediction and the baseline) and consistency (a change in the model that increases the importance of a feature will not decrease its SHAP value).

For different model types, SHAP provides optimized algorithms. For tree-based models like the XGBoost model used in this article, TreeSHAP offers a fast and exact computation method. For deep learning models, DeepSHAP provides an efficient approximation. A key advantage of SHAP is its ability to provide both local, instance-level explanations (like LIME) and global explanations.

Existing Prompt-Aware Techniques and Their Trade-offs

A smaller but growing body of research has begun to explore prompt-aware modelling to address these limitations. These approaches can be broadly categorized into two types: model-level and feature-level adaptations, each with its own set of advantages and trade-offs.

Model-level adaptations involve training a separate, specialized model for each individual prompt (Phandi et al., 2015). This approach can lead to higher performance, as each model can learn the specific nuances and scoring criteria for its assigned prompt without interference from the data of other prompts. However, this method is computationally expensive, especially when the number of prompts is large, as it requires separate training and hyperparameter tuning for each model. More importantly, it is data inefficient. It prevents the transfer of knowledge between prompts, meaning that general patterns of good writing (e.g., grammatical correctness, coherent structure) must be learned from scratch for each prompt. This is particularly problematic for prompts with a small number of training examples, where the model may be prone to overfitting.

The approach taken in this article, prompt-aware dynamic score binning, can be seen as a novel form of data-level adaptation. Rather than modifying the model or its features, it normalizes the target variable (the score) to be consistent and meaningful across all prompts before the modelling process begins. This simple yet effective preprocessing step, detailed in Chapter 3, addresses the core problem of inconsistent scoring scales. The successful outcome of this normalization is a set of balanced score distributions for each prompt, as will be detailed and visualized in Figure 3 in the next chapter. This provides a more robust and equitable foundation for any subsequent modelling, whether it be prompt-agnostic or prompt-aware.

Research Gap

The existing literature on Automated Essay Scoring (AES) has extensively examined predictive performance, explainable artificial intelligence (XAI), prompt-specific modeling, and algorithmic fairness; however, these research areas have largely developed in isolation. Current AES studies primarily focus on improving accuracy through complex deep learning and transformer-based models, often at the expense of transparency and interpretability. Although XAI techniques have been applied to natural language processing tasks, their integration with AES remains limited and is rarely connected to systematic fairness evaluation. Similarly, research addressing prompt-specific bias has concentrated mainly on sophisticated model-level adaptations, while simpler and potentially more scalable data-level normalization strategies have received insufficient attention. Furthermore, fairness studies in AES are still emerging and seldom provide comprehensive audits of transformer-based models across demographic groups or examine how model explanations relate to fairness outcomes. Consequently, a significant research gap exists in the absence of a holistic, reproducible, and integrated AES framework that simultaneously addresses predictive performance, interpretability, prompt-specific bias, and demographic fairness as interconnected components of a trustworthy educational assessment system.

Methodology

This study adopts a quantitative experimental research design to develop and evaluate a model-agnostic Automated Essay Scoring (AES) framework. The methodology emphasizes reproducibility, transparency, and fairness by implementing a standardized evaluation protocol that can be replicated across different machine learning and transformer-based models. The proposed framework is designed to provide reliable performance while remaining understandable to educational stakeholders and consistent with current research standards (Loukina et al., 2019a; Pineau et al., 2020).

Data and Variables

This section describes the dataset, variables, and data quality procedures used throughout the study. The research utilizes the ASAP 2.0 dataset, which contains essays written in response to multiple writing prompts with different scoring scales. To ensure consistent model training and evaluation, the dataset is carefully validated, standardized, and prepared before any modelling process begins.

Dataset

The study uses the ASAP 2.0 dataset, which consists of English-language student essays collected from seven different writing prompts. Each prompt has its own scoring rubric and score range, making the dataset suitable for evaluating multi-prompt automated essay scoring systems. In addition to essay text and human-assigned scores, the dataset provides demographic information that enables fairness evaluation without using sensitive attributes during model training (Loukina et al., 2019b).

Variables

The primary input variable is the essay text, which serves as the basis for both traditional feature engineering and transformer-based language models. The prompt identifier is used to distinguish writing tasks and support prompt-aware target construction, while the human-assigned raw score is transformed into standardized proficiency categories for classification. Demographic variables are reserved exclusively for post-hoc fairness analysis and are not included as input features during model training, thereby reducing the risk of introducing bias into the learning process (Hardt et al., 2016).

Inclusion, Exclusion, and Integrity Checks

Before model development, the dataset undergoes several quality assurance procedures to ensure data reliability. Essays with missing text, missing scores, duplicate records, or invalid prompt information are excluded from the analysis. Additional integrity checks verify that all required variables are present and that score values fall within the documented ranges for each prompt. These validation procedures improve data quality, reduce potential errors, and support reproducible machine learning experiments (Pineau et al., 2020).

Preprocessing

Preprocessing prepares the essay data for both traditional machine learning models and transformer-based architectures while preserving important linguistic characteristics. All preprocessing operations are performed separately within each cross-validation fold to prevent information leakage between training and validation data. Every preprocessing configuration is documented to ensure that the complete experimental pipeline can be reproduced in future studies (Pineau et al., 2020).

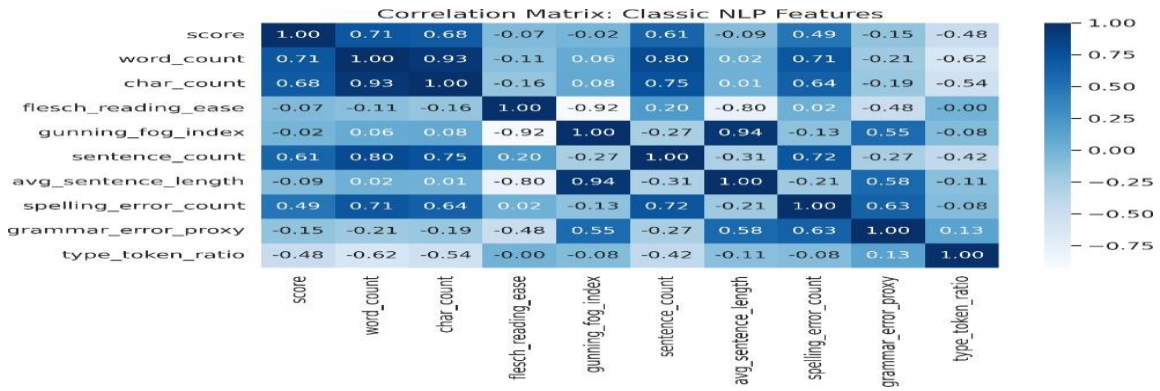
Text Normalization

Text normalization reduces unnecessary variation in the essays while maintaining linguistic features that contribute to writing quality. The preprocessing process includes Unicode normalization, removal of non-printable characters, whitespace standardization, and appropriate handling of punctuation and numerical tokens. Lowercase conversion is applied only for traditional machine learning features, whereas transformer models retain the original text because their tokenizers are designed to process casing information effectively (Devlin et al., 2019).

Classic Feature Extraction

Traditional linguistic features are extracted from the normalized essays to provide interpretable information about writing quality. These features include measures of essay length, sentence structure, readability, lexical diversity, and simple error indicators that collectively represent different aspects of written communication. Such handcrafted features establish a transparent baseline for comparison with transformer-based representations and facilitate the interpretation of model predictions in educational settings (James et al., 2021).

Figure 3: Correlation matrix of classic features



Transformer embeddings

Goal: Capture contextual semantics that classic features miss. The encoder is fine-tuned within each outer fold, then used to extract a fixed length embedding for every essay. This design keeps the downstream classifiers simple and makes ablations comparable across pathways (Devlin et al., 2019).

Fold safe fine tuning

1. **Data:** Use only the fold's training split to fit encoder parameters. The validation split is never used for fitting or early feature learning outside the fold.
2. **Objective:** Predict the prompt aware binned label.
3. **Sequence policy:** Fix a max sequence length that keeps truncation rare. Truncate from the tail if needed and log the fraction truncated.
4. **Optimization:** Use a small learning rate with a linear schedule and warmup, gradient clipping, and early stopping based on the fold's validation loss to control overfitting.
5. **Class imbalance:** Apply class weights or a weighted sampler on the training split when bins are imbalanced.
6. **Embedding extraction:** After fine tuning, take the final hidden state representation per token and pool to a single vector per essay. The same fold trained encoder produces embeddings for both the training and validation splits of that fold, which are then fed to simple classifiers or fused with classic features.
7. **Representation diagnostics:** Use t SNE to visualize the classic feature space and the embedding space for qualitative checks of structure. t SNE is used for inspection only, not for modelling, and its hyperparameters are kept stable to ensure comparability across folds (Maaten & Hinton, 2008).

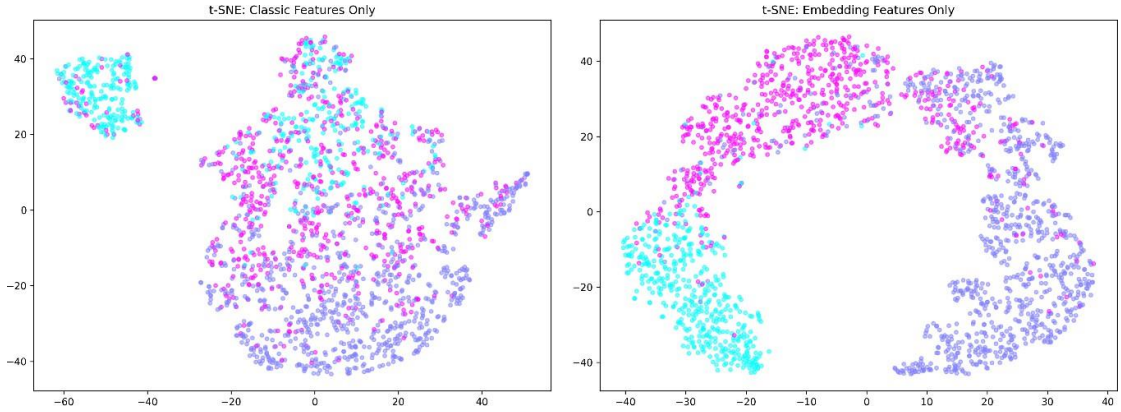


Figure 4: A t-SNE Comparison of Feature Spaces

Pseudocode: Fold safe embedding extraction

```
for each outer fold k:
    Train, Val = split_k(D)
    # Build prompt aware labels on Train only
    edges = compute_bin_edges(Train)
    y_tr = bin_with(edges, Train)
    y_va = bin_with(edges, Val)

    # Tokenize with the model tokenizer
    Ttr = tokenize(Train.text)
    Tva = tokenize(Val.text)

    # Fine tune encoder on Train only
    enc = finetune_encoder(Ttr, y_tr,
                           lr=small, scheduler="linear", warmup=short,
                           class_weights=True, early_stopping=True,
                           grad_clip=True, seed=42)

    # Extract fixed length embeddings with the fold trained encoder
    Ztr = encode(enc, Ttr)
    Zva = encode(enc, Tva)

    # Save embeddings and metadata for this fold
    persist({"fold": k, "edges": edges, "embeddings_train": Ztr, "embeddings_val": Zva})
```

Figure 5: Pseudocode for Fold Safe Embedding Extraction

Target construction by prompt

Target construction standardizes the learning objective across prompts with different scoring ranges while respecting the ordinal nature of scores. All computations in this section are performed inside the cross-validation harness to avoid leakage.

Rationale

Raw score ranges and distributions differ by prompt. Training a single global target mixes incomparable scales and can bias evaluation toward prompts with broader ranges. The protocol therefore proposes a prompt aware binning scheme that maps raw scores within each prompt to three proficiency levels low, medium, and high. This choice keeps the task ordinal, improves class balance for classification, and supports fairness analysis that focuses on recall in the high proficiency group (Cohen, 1968; Hardt et al., 2016).

Implementation

- Two pathways run under the same outer CV harness.
- Hyperparameter tuning

Hyperparameters are tuned inside the cross-validation harness so that no information from a fold's validation split can influence model selection. Searches are small and targeted to maintain runtime discipline and to reduce the risk of overfitting hyperparameters to the folds (James et al., 2021; Pineau et al., 2020).

Validation and diagnostics

Distribution checks: Plot class counts per prompt before and after binning to verify that the procedure improves balance without erasing genuine difficulty differences. Confirm that the proportion of high labels is not dominated by a single prompt.

Sensitivity checks: Recompute the bins with small perturbations to the cut points and verify that class assignments are stable for the vast majority of essays. This guards against brittle edges.

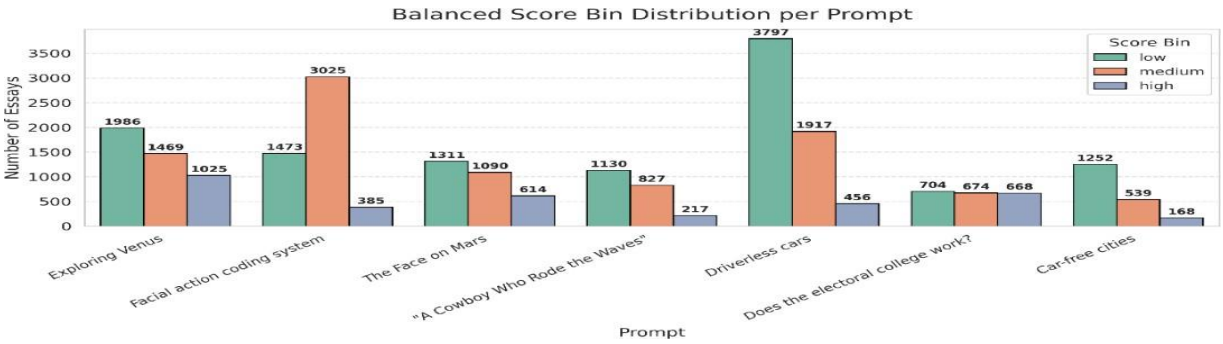


Figure 6: Binned score distribution by prompt

Demographic neutrality check

Cross tabulate the binned labels by demographic groups within each prompt on the training split to confirm that the binning rule itself does not introduce artificial skews. Report only aggregate proportions and keep raw sensitive attributes out of derived files used by later stages (Hardt et al., 2016).

Leakage guard

All cut points are computed on the training split of each fold and then applied unchanged to that fold's validation split. No bin thresholds, preprocessors, or embeddings are ever fit on validation data. This isolates evaluation from any information about the held-out essays and aligns with best practice in supervised learning (James et al., 2021).

Pseudocode: Prompt aware binning

```
# Inputs: df[text, prompt, raw_score]
# Output: binned labels and per prompt cut points per fold
for fold in 1..5:
    Train, Val = stratified_kfold(df, key=(prompt, placeholder_label), k=5, seed=42)[fold]

    # Compute cut points on Train only
    edges = {}
    for p in unique(Train.prompt):
        s = Train[Train.prompt == p].raw_score
        e = try_quantile_edges(s, [0, 1/3, 2/3, 1]) or linspace_edges(min(s), max(s), 3)
        edges[p] = e

    # Apply the same mapping to Train and Val
    Train.score_bin = apply_edges(Train.raw_score, Train.prompt, edges)
    Val.score_bin   = apply_edges(Val.raw_score,   Val.prompt,   edges)

    # Persist the cut points and summary counts for auditing
    persist({"fold": fold, "edges": edges,
             "counts_train": counts_by_prompt(Train.score_bin),
             "counts_val": counts_by_prompt(Val.score_bin)})
```

Figure 7: Pseudocode: Prompt aware binning

Experimental design and cross-validation

Outer protocol: Use 5 fold stratified cross validation. Each fold acts once as validation and four times as training. The evaluation split remains completely untouched until a single scoring pass is made at the end of each fold.

Stratification key: Concatenate prompt and score_bin to preserve the joint distribution of prompts and proficiency levels in every fold. This prevents label only stratification from over or underrepresenting prompts with skewed class proportions (James et al., 2021).

Fold integrity: All preprocessing, bin construction, tokenizer fitting, transformer fine tuning, and feature scaling are learned on the training split of the fold and then applied to the validation split with no refitting. Random seeds are fixed per fold to stabilize stochastic components while keeping folds independent. Any essays duplicated across prompts are removed upstream during integrity checks to avoid leakage.

Imbalance handling: Within each fold, residual class imbalance after binning is handled by class weights or a weighted sampler on the training split only. Validation splits are never resampled, so that reported metrics reflect true prevalence.

Validation reporting: For each fold, report the primary and secondary metrics on the untouched validation split and retain per essay predictions to support later explainability and fairness analysis. Aggregate across folds by the mean and standard deviation and construct confidence intervals by simple bootstrap over the five fold scores. Statistical comparisons between feature sets use paired tests across folds and are described in Section 3.8 (Cohen, 1968; James et al., 2021).

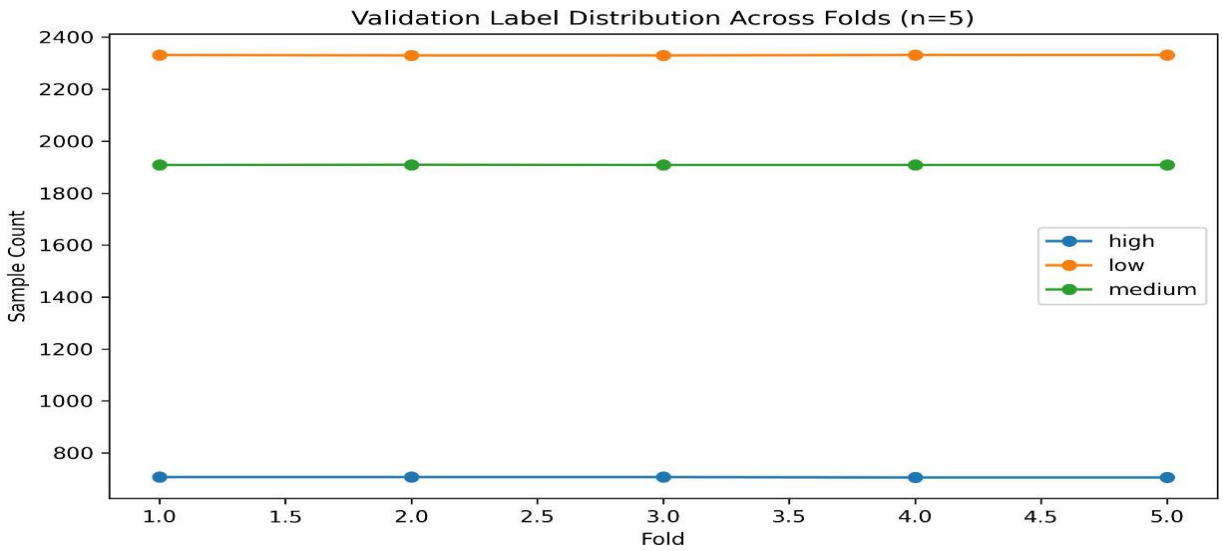


Figure 8: Validation Label Distribution Across Folds

Pseudocode: Leakage safe stratified CV

```
Input: D (essays with text, prompt, raw_score, demographics)
Output: Per fold predictions, metrics, saved models

folds = stratified_kfold(D, key=(prompt, placeholder_label), k=5, seed=42)
for k, (Train, Val) in enumerate(folds):
    # Build prompt aware bins on Train only
    edges = {p: compute_edges(Train[Train.prompt==p].raw_score)
             for p in unique(Train.prompt)}
    y_tr = bin_with(edges, Train)
    y_va = bin_with(edges, Val)

    # Fit preprocessing on Train only
    prep = fit_preprocessing(Train.text)
    Xtr  = transform(prep, Train.text)
    Xva  = transform(prep, Val.text)

    # Transformer path
    tr_clf = finetune_transformer(Train.text, y_tr,
                                  class_weights=True,
                                  early_stopping=True,
                                  seed=42)
    Ztr = extract_embeddings(tr_clf, Train.text)
    Zva = extract_embeddings(tr_clf, Val.text)

    # Classic models on Classic Only and Fused
    models = train_tabular_models({Xtr, Ztr}, y_tr, tune=True)

    # One time evaluation on untouched Val
    reports = evaluate_all(models ⊎ {tr_clf}, {Xva, Zva}, y_va)

    # Persist models, configs, and fold level reports for audit
    persist({"fold": k, "edges": edges, "prep": prep,
             "models": models, "transformer": tr_clf,
             "reports": reports})
```

Figure 9: Pseudocode: Leakage safe stratified CV

Model families

Two modelling pathways run under the same cross validation harness. The first uses interpretable tabular features. The second learns text representations with a pretrained encoder and then uses those representations directly or in fusion with the tabular features. Choices reflect common practice in supervised learning and gradient boosted trees for tabular data and transformer encoders for text (Chen & Guestrin, 2016; Devlin et al., 2019; James et al., 2021).

Classic pathway

Algorithms: Logistic Regression, Random Forest, and XGBoost trained on Classic Only and Fused features.

Feature preparation: Classic features are standardized by the parameters learned on the training split of each fold. When fusing with embeddings, concatenate the z scored classic features with the embedding vector and fit models on the combined matrix. No dimensionality reduction is applied so that feature level attributions remain interpretable.

Model specifics

Logistic Regression. Multinomial objective with L2 regularization. Class weights applied when imbalance persists after binning. Probability estimates are optionally calibrated on the training split via cross validation to improve probability quality for fairness metrics (James et al., 2021).

Random Forest. Hundreds of trees with square root feature sampling, depth limited to avoid overfitting, and out of bag estimates used only for internal diagnostics, never for final scoring.

XGBoost. Gradient boosted trees with tree method gpu_hist when available, learning rate in the low range, shallow trees, and subsampling for regularization.

Evaluation Metrics and Statistical Testing

Model performance was primarily evaluated using Quadratic Weighted Kappa (QWK), as it accounts for the ordinal nature of essay scores. Macro F1-score and accuracy were also calculated to provide complementary performance measures, while probability calibration was assessed to evaluate prediction reliability. Performance results were averaged across all cross-validation folds to ensure robust evaluation (Cohen, 1968; Guo et al., 2017).

Explainability Protocol

Model explainability was incorporated to improve transparency and interpretability of predictions. SHAP was used to identify globally important features, while LIME generated local explanations for individual essays. All explanations were produced

only on validation data to avoid information leakage and ensure unbiased interpretation of model behavior (Lundberg & Lee, 2017; Ribeiro et al., 2016).

Fairness Audit Protocol

A fairness assessment was conducted using demographic attributes only after model prediction. Equal Opportunity Difference was calculated to compare recall across demographic groups for the high-proficiency class. Sensitive attributes were excluded from model training and were used solely for evaluating fairness on validation predictions (Hardt et al., 2016).

Reproducibility and Environment Controls

To ensure reproducibility, all experiments were executed using fixed random seeds and fully documented software and hardware configurations. Dataset versions, preprocessing settings, model parameters, and cross-validation splits were recorded in a run manifest, enabling other researchers to reproduce the complete experimental pipeline under the same conditions (Pineau et al., 2020).

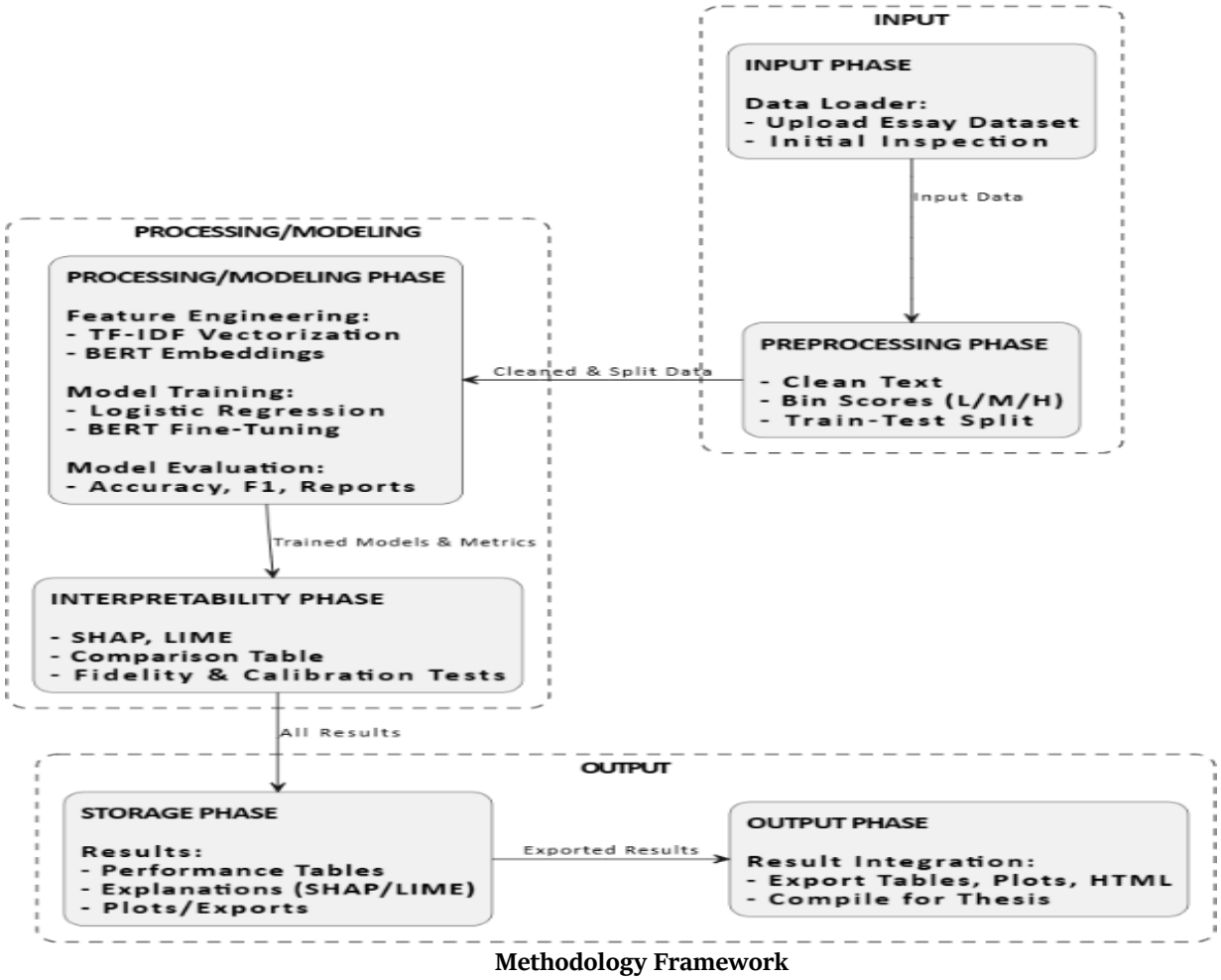
Classic models

Search strategy: For each fold, run a compact grid or random search on the training split. Use a stratified inner split taken only from the training portion to select hyperparameters. The winner is refit on the full training split of the fold and evaluated once on the untouched validation split.

Objectives and scoring: The selection score is mean Quadratic Weighted Kappa over the inner resamples because the target is ordinal (Cohen, 1968). Macro F1 is monitored as a secondary objective when two settings tie on QWK.

Search spaces

- **Logistic Regression:** C in a logarithmic range, penalty L2, solver that supports multinomial loss. Class weights on or off as a discrete switch.
- **Random Forest:** Number of trees, maximum depth, minimum samples per split, maximum features as a proportion.
- **XGBoost:** Learning rate in a low range, maximum depth, subsampling and column sampling proportions, minimum child weight, number of trees. Tree method gpu_hist when available (Chen & Guestrin, 2016).
- **Calibration:** For probability quality needed by fairness metrics, optional calibration is performed on the training split via cross validation and then frozen. Validation predictions are produced with the calibrated model only once (James et al., 2021).



Methodology Framework

Figure 10: Methodology Diagram

Data Analysis

Data Splitting: Stratified K-Fold Cross-Validation

To obtain a reliable estimate of model generalization, this study employed 5-fold stratified cross-validation. In each iteration, four folds were used for model training, while the remaining fold was reserved for validation, ensuring that every essay was evaluated exactly once. Performance metrics reported in this study represent the average results across the five validation folds, providing a more stable and robust evaluation than a single train–test split (Allgaier & Pryss, 2024; Bates et al., 2024).

Considering the multi-prompt nature of the ASAP 2.0 dataset, stratification was performed using a combined prompt_label, created by merging the essay prompt with its corresponding score bin. This strategy preserved the joint distribution of prompts and proficiency levels across all folds, reducing sampling bias and improving the reliability of model evaluation. Figure 14 illustrates the distribution of score bins across the validation folds, confirming that the stratified cross-validation procedure maintained consistent class proportions throughout the experimental process.

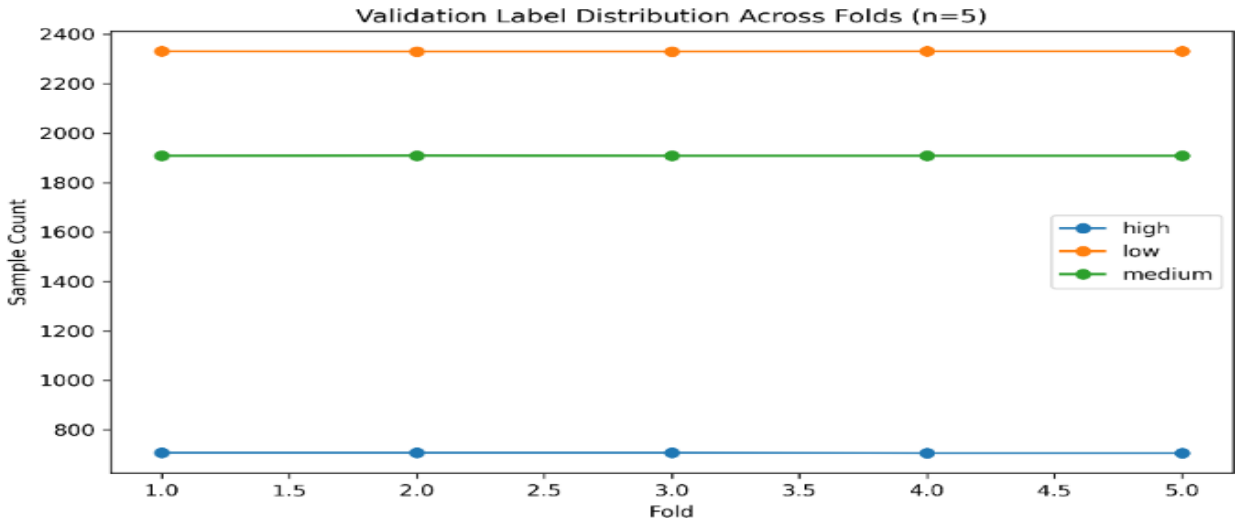


Figure 11: Validation Label Distribution Across Folds

As the figure illustrates, the number of samples for each class is remarkably consistent across all five folds. The lines for the 'low', 'medium', and 'high' bins are nearly flat, indicating that each fold presents the models with a validation set that has a very similar class distribution. This stability is a direct result of the careful stratification process. It confirms that the cross-validation procedure is well-balanced and that the performance metrics reported in the next chapter are not the result of a chance artifact of the data splitting process. This careful approach to data splitting is essential for the validity of the subsequent performance and fairness analyses, as it guarantees that each model is evaluated on a representative and consistent sample of the full data distribution.

Evaluation Framework

Model performance was evaluated using Accuracy, Macro F1-score, and Quadratic Weighted Kappa (QWK). Accuracy provides an overall measure of classification performance, whereas Macro F1-score evaluates predictive performance across all proficiency classes while accounting for class imbalance. QWK was included as the primary agreement metric because it measures ordinal consistency between predicted and human-assigned essay scores and is widely adopted in automated essay scoring research (Chicco & Jurman, 2020; Cohen, 1968).

Ablation Study

An ablation study was performed to quantify the contribution of different feature representations to automated essay scoring performance. Three feature configurations were evaluated: handcrafted linguistic features (Classic-Only), contextual transformer embeddings (Embeddings-Only), and a fused representation combining both feature types. Each feature set was evaluated using Logistic Regression, Random Forest, and XGBoost under the same cross-validation protocol, while BERT and DeBERTa were trained directly on raw essay text. This experimental design enabled a systematic comparison of feature engineering and contextual language representations, providing insight into their individual and combined contributions to predictive performance (Wang, 2024; Yang et al., 2024).

Performance Benchmarking

Performance benchmarking compared all machine learning and transformer-based models under identical experimental conditions. Results from the ablation study were analyzed to determine the influence of feature representation and model architecture on automated essay scoring performance. Comparative analysis across traditional, ensemble, and transformer models established the relative effectiveness of each approach and identified the best-performing framework for subsequent explainability and fairness evaluation.

1	RawText	DeBERTa-v3-	0.89	0.02	0.89	0.02	0.81	Top
		large	5	3	8	2	4	transformer model
2	Fused	XGBoost	0.88	0.00	0.89	0.00	–	Best classic (fused)
			3	4	0	3		
3	Embeddings	XGBoost	0.88	0.00	0.88	0.00	–	
	-Only		2	5	7	3		
4	Fused	RandomForest	0.88	0.00	0.88	0.00	–	
			0	4	6	3		
5	Embeddings	RandomForest	0.88	0.00	0.88	0.00	–	
	-Only		0	5	5	4		
6	RawText	BERT-base-	0.86	0.01	0.86	0.01	0.75	Baseline
		uncased	2	9	2	9	3	transformer
7	Fused	LogisticRegressio	0.87	0.00	0.87	0.00	–	
		n	2	6	5	5		
8	Embeddings	LogisticRegressio	0.87	0.00	0.87	0.00	–	
	-Only	n	1	7	3	6		
9	Classic-Only	XGBoost	0.64	0.00	0.61	0.00	–	Best classic-only
			3	1	6	2		
1	Classic-Only	RandomForest	0.64	0.00	0.60	0.00	–	
0			3	5	5	8		
11	Classic-Only	LogisticRegressio	0.60	0.00	0.58	0.00	–	
		n	9	3	2	3		
1	Majority	Baseline	0.47	0.00	0.21	0.00	–	Most
2	Class		1	0	4	0		frequent
								class
								baseline

Table 2: Comprehensive comparison of classic and transformer-based models.

All results, from the most advanced model to the naïve baseline, are reported in accordance with best practices recommended in recent AES literature (Li & Ng, 2024; Misgna et al., 2024). This ensures a fair basis for comparison and provides a realistic view of progress, not just incremental gains. Including the Majority Class predictor, which delivers a Macro-F1 of 0.214, establishes a transparent baseline for evaluating all subsequent results and prevents inflated performance claims.

Cross-Validation Consistency

The stability of model performance across the five validation folds was assessed using the distribution of Macro-F1 scores presented in Figure 15. Transformer-based models exhibited consistently high performance with relatively low variation across folds, indicating strong generalization to unseen data. In contrast, models trained on handcrafted features showed greater variability, reflecting their higher sensitivity to differences in training data.

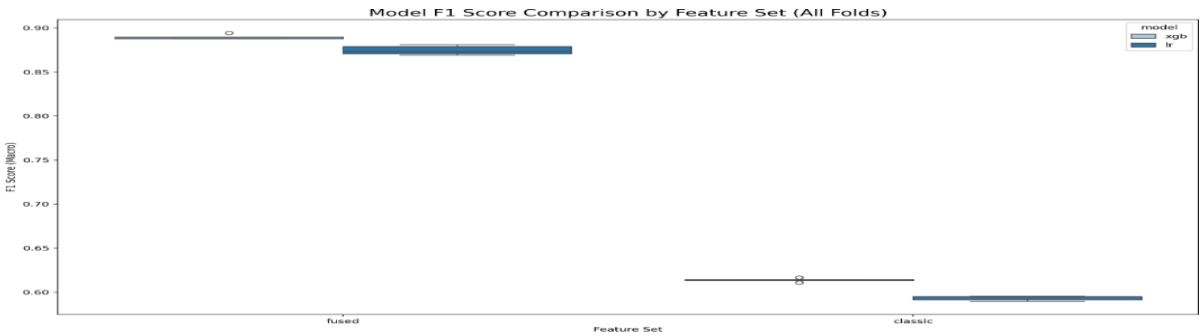


Figure 12: Model F1 Score Comparison by Feature Set (All Folds)

The pattern revealed in the figure is both striking and significant. Classic-Only models cluster at the lower end of the scale, with lower median F1 scores and noticeably wider interquartile ranges. This variability signals that traditional models are sensitive to the specific distribution of essays in each fold, resulting in inconsistent performance, an unacceptable risk for high-stakes assessment.

Aggregated Prediction Outcomes Analysis

A standard confusion matrix is often used to examine a single model's predictions in one fold, but it can miss broader trends. Here, the analysis is based on a comprehensive aggregated outcomes table, which sums up the model predictions and true class labels across all five cross-validation folds for each major model and feature set. This table captures the real-world distribution of predictions and misclassifications on the entire dataset, providing a much more robust and reliable understanding of model behaviour.

Model & Features	True Label	Pred: High	Pred: Low	Pred: Medium	Total Samples
XGBoost, Fused	High	3192	9	304	3505
	Low	24	10468	1221	11713
	Medium	298	1036	7554	8888
Logistic Reg., Fused	High	3163	14	328	3505
	Low	46	10438	1229	11713
	Medium	460	1080	7348	8888
XGBoost, Classic	High	1797	187	1521	3505
	Low	263	8785	2665	11713
	Medium	1180	2800	4908	8888
Logistic Reg., Classic	High	2243	82	1180	3505
	Low	446	8483	2784	11713
	Medium	1664	2600	5624	8888

Table 3: Aggregated Prediction Outcomes for Major Models

Analysis of this table uncovers a clear difference in the quality and type of errors produced by fused models compared to classic-only approaches. Fused models like XGBoost and Logistic Regression achieve both high accuracy and high precision in their predictions. For these models, more than ninety percent of the predictions for the "Low" and "High" classes are correct, and most errors fall into adjacent categories. In practice, mistakes typically involve misclassifying a "Medium" essay as "High" or vice versa, rather than making wide, implausible jumps across the rating scale. Such localized mistakes mirror the kind of disagreement seen even among expert human graders (Williamson et al., 2012a). This kind of error discipline is starting to be recognized as part of robust AES system design.

Prompt-Specific Performance Breakdown

Aggregate scores can often obscure important variation in model performance across different types of essay prompts. To address this, a prompt-level breakdown is performed for the best model (DeBERTa-v3-large), providing insight into both the generalizability and specific vulnerabilities of the approach.

Prompt Name	Rhetorical Mode	Mean Macro-F1
Does the electoral college work?	Persuasive	0.921
Driverless cars	Persuasive	0.915
Facial action coding system	Expository	0.903
Exploring Venus	Expository	0.895
The Face on Mars	Expository	0.887
Car-free cities	Persuasive	0.881
A Cowboy Who Rode the Waves	Narrative	0.879

Table 4: DeBERTa-v3-large Performance by Prompt (Mean Macro-F1)

The table reveals clear and actionable trends. DeBERTa-v3-large consistently achieves its highest performance on persuasive and expository prompts. These genres tend to have predictable structures, clear article statements, and argument patterns, making them easier for transformer models to parse and evaluate with high fidelity. This observation aligns with recent neural AES findings. Atkinson and Palma (Atkinson & Palma, 2025) report improved scoring accuracy on persuasive and expository essays when hybrid transformer models are used. A systematic review further confirms that models perform better on structured, formulaic writing tasks, noting that genre and structure influence model confidence and accuracy.

Training Dynamics and Model Behaviour

A complete evaluation of model performance requires not only final scores, but also a careful analysis of the training process and how the model behaves during learning. To this end, the fine-tuning procedure for DeBERTa-v3-large was monitored across all folds, recording both training and validation loss, as well as validation QWK, after each epoch. Figure 16 presents a representative learning curve for fold 3, which reflects typical model behaviour observed across the study.

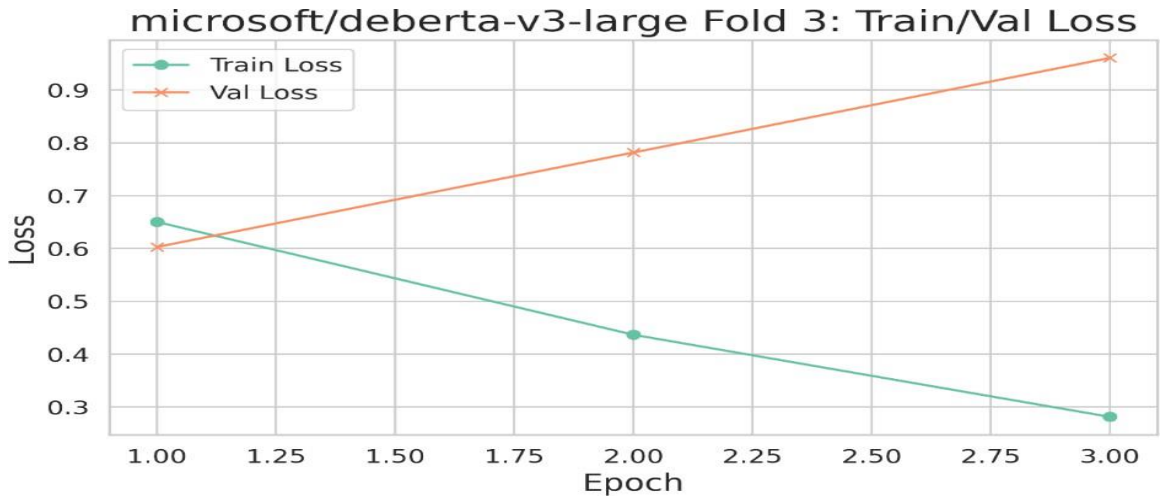


Figure 13: Transformer Training and Validation Curve

The Rule-Based Logic of Classic Models

To begin understanding how classic models make decisions, it is important to identify the features they consistently rely on. Global interpretability methods such as SHAP (SHapley Additive exPlanations) provide a high-level overview of each model's decision-making process. By aggregating these SHAP values, we can summarize the core principles each model has learned to associate with writing quality.

The models trained on the Classic-Only feature set represent a traditional, feature-engineered approach to automated essay scoring. As the SHAP summary plots in Figure 17 and Figure 18 demonstrate, their decision-making is dominated by a small set of surface-level linguistic features. This reveals a learned principle where writing quality is equated with a checklist of easily quantifiable metrics.

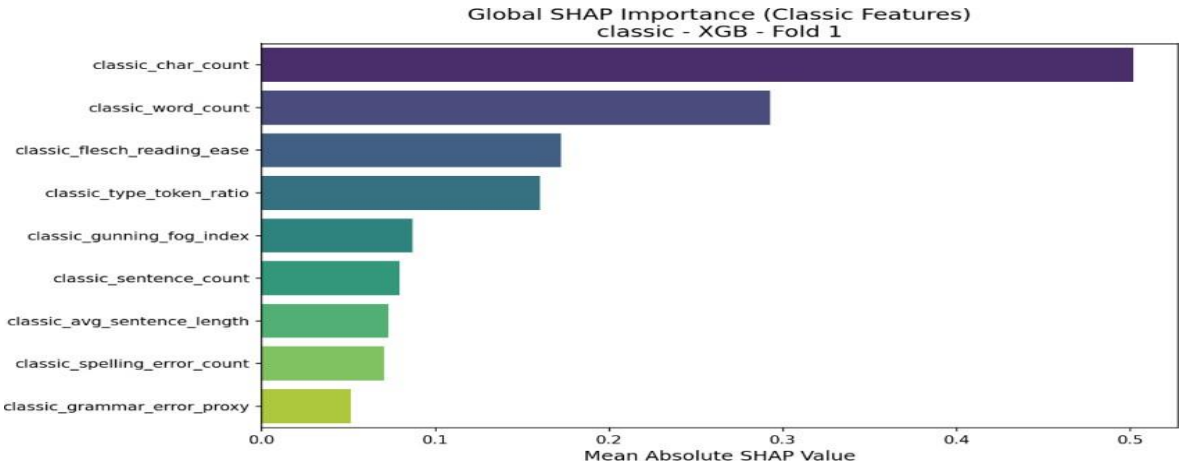


Figure 14: Global SHAP Feature Importance for XGBoost (Classic-Only, Fold 1)

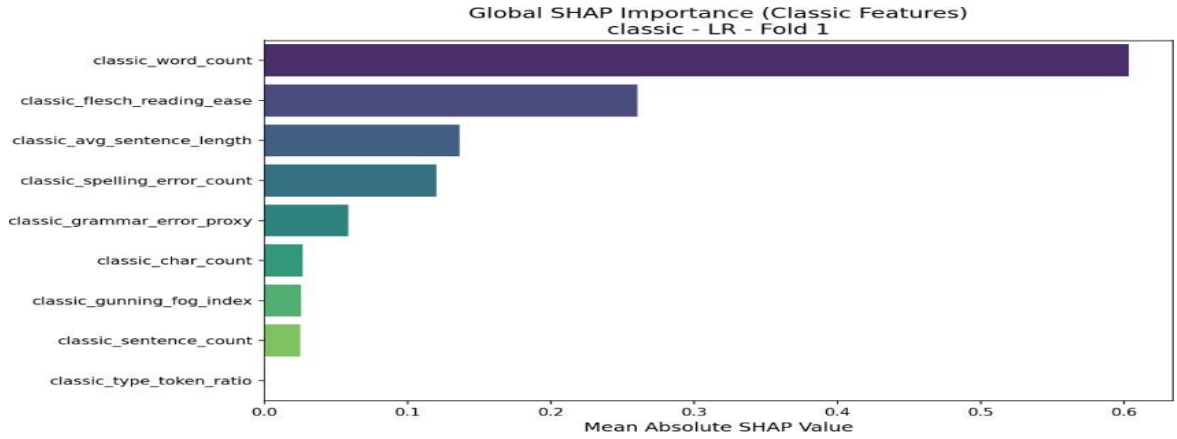


Figure 15: Global SHAP Feature Importance for Logistic Regression (Classic-Only, Fold 1)

The Shift to Meaning-Based Analysis in Fused Models

The adoption of numerical representations from large language models, often called embeddings, marks a decisive turning point in automated essay scoring. Fused models combine these embeddings with traditional linguistic features, leading to a major shift in how decisions are made. Instead of relying mainly on explicit, hand-crafted rules, the model now operates on a far deeper semantic level, and this change is visible in its behaviour and performance.

A close examination of the global SHAP feature importance plots for the XGBoost model trained on the Fused feature set makes this transformation clear. As shown in Figure 19 (inserted below), the highest-ranked features are almost entirely the individual dimensions of the transformer's embedding, labelled embedding_0 through embedding_1023. Traditional linguistic features that were previously central, such as word count or sentence complexity are now mostly absent from the top contributors when these embeddings are included.

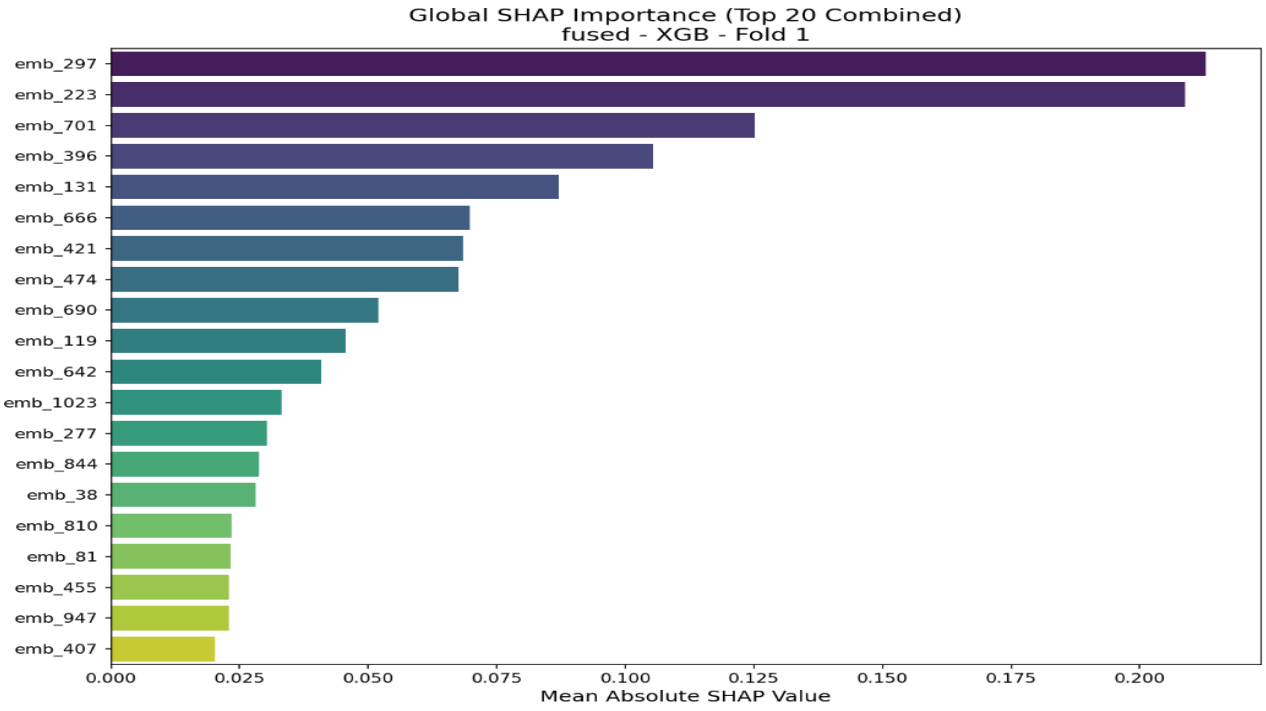


Figure 16: Global SHAP Feature Importance for XGBoost (Fused, Fold 1)

Comparative Analysis of Decision Policies

The SHAP analysis revealed clear differences in the decision-making strategies of traditional and transformer-enhanced models. Models trained on handcrafted linguistic features primarily relied on surface-level characteristics, including essay length, lexical complexity, and spelling-related indicators. Although these features provided interpretable decision rules, they were less effective in capturing semantic quality and discourse coherence, which limited their predictive performance. Similar

observations have been reported in previous AES studies, where handcrafted features were found to represent only limited aspects of writing quality (Phandi et al., 2015).

Aggregate Fairness Analysis: A High-Level View of Equity

The first step in the fairness audit is to examine how the DeBERTa-v3-large model performs across the key demographic groups represented in the dataset. To do this, predictions from all five cross-validation folds were pooled, and performance metrics were calculated separately for each demographic subgroup. Table 5 summarizes the Macro-F1 score, recall for the 'high' score bin, and the Equal Opportunity Difference, providing a clear snapshot of where disparities exist and where the model delivers equitable outcomes.

Demographic Category	Group	Macro-F1	Recall ('high' bin)	Equal Opportunity Difference
Gender	Female	0.899	0.892	-0.008
	Male (Privileged)	0.897	0.900	
Economic Status	Econ. Disadvantaged	0.891	0.869	+0.041
	Not Disadvantaged (Privileged)	0.901	0.910	
Race/Ethnicity	Black	0.885	0.855	+0.065
	Hispanic	N/A	0.868	+0.052
	White (Privileged)	0.903	0.920	

Table 5: Fairness Audit Results for DeBERTa-v3-large Model

Fairness Analysis

Analysis by Gender

The gender-based fairness analysis demonstrated highly consistent model performance across male and female students. Both groups achieved nearly identical Macro-F1 scores and recall for the high-proficiency class, resulting in an Equal Opportunity Difference close to zero. These findings indicate that the model identified high-quality essays with comparable effectiveness for both gender groups, suggesting minimal gender-related bias. Similar observations have been reported in recent studies indicating that transformer-based AES models generally exhibit limited gender disparities when evaluated using fairness metrics (Baker & Hawn, 2022).

Analysis by Economic Status

Differences in model performance were more apparent across socioeconomic groups. Students classified as non-economically disadvantaged achieved slightly higher Macro-F1 scores and recall than economically disadvantaged students. The Equal Opportunity Difference was approximately 4%, indicating a modest but measurable disparity in recognizing high-performing essays. Although demographic attributes were excluded from model training, the observed differences suggest that socioeconomic characteristics may be indirectly reflected through linguistic patterns, such as vocabulary richness, syntactic complexity, and writing style. This finding is consistent with previous research demonstrating that language models can inadvertently encode socioeconomic disparities present in educational datasets (Mitchell et al., 2019; Baker & Hawn, 2022).

Analysis by Race and Ethnicity

The largest performance disparities were observed across racial and ethnic groups. White students achieved the highest Macro-F1 score (0.903), whereas Black students obtained the lowest (0.885). The Equal Opportunity Difference reached 0.065 between White and Black students and 0.052 between White and Hispanic students, indicating lower recall for high-performing essays from minority groups. These results suggest that the model was less effective at identifying high-quality essays written by Black and Hispanic students than those written by White students.

Prompt-Specific Fairness Analysis

Prompt-level analysis revealed that fairness varied according to the writing task. For the structured persuasive prompt ("Driverless Cars"), model performance remained relatively consistent across demographic groups, indicating equitable scoring under well-defined writing conventions. In contrast, larger disparities emerged for the creative narrative prompt ("A Cowboy Who Rode the Waves"), where performance for Black and Hispanic students declined relative to White students. These findings suggest that demographic fairness is prompt-dependent and that creative writing tasks present greater

challenges for maintaining equitable automated assessment. Similar prompt-dependent bias has recently been reported in automated essay scoring research (Yang et al., 2024).

Findings

- Transformer-based models consistently outperformed traditional feature-engineered approaches across all evaluation metrics, demonstrating the effectiveness of contextual language representations for automated essay scoring.
- DeBERTa-v3-large achieved the highest overall performance, recording the best Accuracy, Macro-F1, and Quadratic Weighted Kappa (QWK) scores among all evaluated models.
- Classical machine learning models trained on transformer embeddings achieved performance comparable to end-to-end transformer models, highlighting that feature representation contributed more to predictive performance than classifier complexity.
- Combining handcrafted linguistic features with contextual embeddings produced only marginal improvements over embedding-based features alone, indicating that transformer embeddings captured most of the information required for essay scoring.
- Cross-validation results showed low variability across folds, confirming the robustness and generalizability of the proposed evaluation framework.
- SHAP and LIME analyses demonstrated that transformer-enhanced models relied primarily on contextual semantic information rather than surface-level linguistic features, providing insights into model decision-making.
- Fairness analysis revealed negligible performance differences between male and female students, indicating minimal gender-related bias.
- Moderate disparities were observed across socioeconomic groups, with slightly higher predictive performance for non-economically disadvantaged students.
- The largest fairness gaps occurred across racial and ethnic groups, where lower recall was observed for high-performing essays written by Black and Hispanic students compared with White students.

Discussion

The findings of this study demonstrate that transformer-based language models provide substantial improvements over traditional machine learning approaches for automated essay scoring (AES), particularly when combined with an integrated framework for explainability and fairness evaluation. Consistent with previous studies (Devlin et al., 2019; He et al., 2021), contextual language representations captured semantic and syntactic information more effectively than handcrafted linguistic features, resulting in higher predictive performance across multiple essay prompts. While classical features remained valuable for interpretability, they were less capable of modeling complex writing characteristics such as coherence, semantic relationships, and contextual meaning.

An important contribution of this research is the integration of predictive performance, explainability, and fairness within a single evaluation framework. Previous AES studies have primarily emphasized predictive accuracy, often reporting metrics such as Quadratic Weighted Kappa (QWK) while providing limited insight into model behavior or demographic equity (Taghipour & Ng, 2016; Uto et al., 2020). In contrast, the present study demonstrates that high predictive performance alone does not guarantee trustworthy educational AI. The incorporation of SHAP and LIME explanations alongside fairness auditing revealed that model decisions and subgroup performance should be interpreted together rather than independently.

The explainability analysis indicated that transformer-based models relied predominantly on semantic representations learned from contextual embeddings, whereas traditional models were influenced primarily by surface-level linguistic features such as essay length, readability, and lexical diversity. These findings align with recent explainable AI research, which suggests that contextual language models capture deeper linguistic patterns than conventional feature-engineering approaches (Lundberg & Lee, 2017; Ribeiro et al., 2016). Although post-hoc explanation techniques cannot fully reveal the internal reasoning of deep neural networks, they provide valuable evidence regarding the features that contribute most strongly to model predictions.

The fairness analysis further demonstrated that algorithmic performance was not uniform across writing tasks. Performance disparities remained relatively small for structured argumentative essays but became more pronounced for narrative prompts. This observation suggests that fairness in automated essay scoring is highly dependent on prompt characteristics rather than representing a fixed property of the model. Similar concerns have been reported in recent educational AI literature, where linguistic diversity and cultural variation influence model predictions despite strong overall performance (Hardt et al., 2016; Gohar & Cheng, 2023). These findings emphasize that fairness evaluation should be conducted separately for different assessment contexts instead of relying solely on aggregate performance metrics.

From an educational perspective, the findings reinforce the importance of maintaining human oversight in automated assessment. Although transformer-based models achieved competitive predictive performance, explainability and fairness analyses identified cases where model predictions could be influenced by dominant linguistic patterns within the training data. Consequently, automated essay scoring systems should be viewed as decision-support tools that assist educators by providing rapid and consistent preliminary evaluations rather than replacing expert human judgment. This recommendation is consistent with current research advocating human-centered artificial intelligence in educational assessment (Holstein & Aleven, 2021).

Limitations and Future Work

Despite the encouraging results, several limitations should be acknowledged. First, the experiments were conducted exclusively on the ASAP 2.0 dataset, which may limit the generalizability of the findings to other educational contexts, writing genres, and languages. Although ASAP 2.0 remains one of the most widely used benchmarks for automated essay scoring, it represents a specific population and assessment setting.

Second, fairness evaluation was constrained by the demographic attributes available within the dataset. The analysis considered only the provided demographic categories and therefore could not investigate more detailed intersectional fairness issues. Previous research has shown that aggregate fairness metrics may conceal disparities affecting smaller demographic subgroups, highlighting the need for richer educational datasets that support comprehensive fairness assessment (Buolamwini & Gebru, 2018; Kearns et al., 2018).

Conclusion

This study developed and evaluated a reproducible and model-agnostic automated essay scoring (AES) framework that integrates traditional machine learning models, transformer-based language models, explainability techniques, and fairness auditing within a unified evaluation pipeline. The experimental findings demonstrate that transformer-based representations consistently outperform handcrafted linguistic features by capturing richer semantic and contextual information, resulting in improved scoring performance across multiple essay prompts. Furthermore, the integration of explainability methods provided meaningful insights into model decision-making, while fairness analysis revealed that predictive performance alone is insufficient for evaluating educational AI systems. Instead, transparency, reproducibility, and equity should be considered alongside conventional performance metrics to ensure responsible deployment in educational assessment.

The findings also highlight that fairness in automated essay scoring is context-dependent, with performance disparities varying across different writing prompts despite strong overall accuracy. This observation reinforces the need for comprehensive evaluation frameworks that assess not only predictive effectiveness but also model interpretability and demographic fairness. Although the proposed framework is constrained by the characteristics of the ASAP 2.0 dataset and the limitations of post-hoc explanation methods, it provides a robust foundation for developing trustworthy educational AI systems. Future research should validate the framework on more diverse datasets, investigate advanced bias mitigation techniques, and explore the integration of explainable AI with intelligent feedback systems to further improve the reliability, transparency, and educational value of automated essay scoring.

Future Implications

The findings of this study provide several important directions for future research and practical implementation in educational technology. Future studies should evaluate the proposed framework using larger, multilingual, and more diverse essay datasets to improve its generalizability across different educational contexts. In addition, integrating recent large language models with advanced bias mitigation techniques and inherently interpretable AI methods may further enhance both predictive performance and fairness. Beyond automated scoring, the framework can be extended to provide personalized formative feedback that supports students' writing development while assisting educators in assessment. As AI becomes increasingly integrated into educational environments, combining accuracy with explainability, fairness, and reproducibility

will be essential for developing trustworthy and ethically responsible automated assessment systems that can be confidently adopted in real-world educational practice.

References

1. Ackerman, P. L. (2021). Commentary: The Future of College Admissions Tests.
2. Ait Khayi, N., & Rus, V. (2024). Automated Essay Scoring Using Discourse External Knowledge. Proceedings of the Thirty-ThirdInternational Joint Conference on Artificial Intelligence, 7154–7160. https://doi.org/10.24963/ijcai.2024/791
3. Alfredo, R., Echeverria, V., Jin, Y., Yan, L., Swiecki, Z., Gašević, D., & Martinez-Maldonado, R. (2023). Human-Centred Learning Analytics and AI in Education: A Systematic Literature Review. https://doi.org/10.48550/ARXIV.2312.12751
4. Allgaier, J., & Pryss, R. (2024). Practical approaches in evaluating validation and biases of machine learning applied to mobile health studies. Communications Medicine, 4(1), 76. https://doi.org/10.1038/s43856-024-00468-0
5. An, J., Huang, D., Lin, C., & Tai, M. (2024). Measuring Gender and Racial Biases in Large Language Models (No. arXiv:2403.15281). arXiv. https://doi.org/10.48550/arXiv.2403.15281
6. Anchiêta, R. T., De Sousa, R. F., & Moura, R. S. (2024). A Robustness Analysis of Automated Essay Scoring Methods. Anais Do XV Simpósio Brasileiro de Tecnologia Da Informação e Da Linguagem Humana (STIL 2024), 75–80. https://doi.org/10.5753/stil.2024.245419
7. Baker, R. S., & Hawn, A. (2022). Algorithmic Bias in Education. International Journal of Artificial Intelligence in Education, 32(4), 1052–1092. https://doi.org/10.1007/s40593-021-00285-9
8. Buda, M., Maki, A., & Mazurowski, M. A. (2018). A systematic study of the class imbalance problem in convolutional neural networks. Neural Networks, 106, 249–259. https://doi.org/10.1016/j.neunet.2018.07.011
9. Chakravarty, A., Brenchley, M., Breakspear, T., Lewin, I., & Huang, Y. (2025). Enhancing Marker Scoring Accuracy through Ordinal Confidence Modelling in Educational Assessments (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2505.23315
10. Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794. https://doi.org/10.1145/2939672.2939785
11. Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC Genomics, 21(1), 6. https://doi.org/10.1186/s12864-019-6413-7
12. Chinta, S. V., Wang, Z., Yin, Z., Hoang, N., Gonzalez, M., Quy, T. L., & Zhang, W. (2024).
13. Crossley, S. A., Baffour, P., Burleigh, L., & King, J. (2025). A large-scale corpus for assessing source-based writing quality: ASAP 2.0. Assessing Writing, 65, 100954. https://doi.org/10.1016/j.asw.2025.100954
14. Cui, Y., Jia, M., Lin, T.-Y., Song, Y., & Belongie, S. (2019). Class-Balanced Loss Based on Effective Number of Samples. 2019 IEEE/CVF Conference on Computer Vision and
15. Gohar, U., & Cheng, L. (2023). A Survey on Intersectional Fairness in Machine Learning: Notions, Mitigation, and Challenges. Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, 6619–6627. https://doi.org/10.24963/ijcai.2023/742
16. Gunasekara, S., & Saarela, M. (2025). Explainable AI in Education: Techniques and Qualitative Assessment. Applied Sciences, 15(3), 1239. https://doi.org/10.3390/app15031239
17. He, P., Liu, X., Gao, J., & Chen, W. (2021). DeBERTa: Decoding-enhanced BERT with Disentangled Attention (No. arXiv:2006.03654). arXiv. https://doi.org/10.48550/arXiv.2006.03654
18. Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

19. Hofmann, V., Kalluri, P. R., Jurafsky, D., & King, S. (2024). AI generates covertly racist decisions about people based on their dialect. Nature, 633(8028), 147–154. https://doi.org/10.1038/s41586-024-07856-5

20. Holstein, K., & Aleven, V. (2021). Designing for human-AI complementarity in K-12 education (Version 2). arXiv. https://doi.org/10.48550/ARXIV.2104.01266

21. https://doi.org/10.1145/3287560.3287596

22. Jain, S., & Wallace, B. C. (2019). [No title found]. Proceedings of the 2019 Conference of the North, 3543–3556. https://doi.org/10.18653/v1/N19-1357

23. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). An introduction to statistical learning: With applications in R (Second edition). Springer.

24. Jiang, Z., Gao, T., Yin, Y., Liu, M., Yu, H., Cheng, Z., & Gu, Q. (2023). Improving Domain

25. Kabra, A., Bhatia, M., Kumar, Y., Li, J. J., & Shah, R. R. (2021). Evaluation Toolkit For Robustness Testing Of Automatic Essay Scoring Systems (No. arXiv:2007.06796). arXiv. https://doi.org/10.48550/arXiv.2007.06796

26. Kasneci, G., & Kasneci, E. (2024). Enriching Tabular Data with Contextual LLM Embeddings: A Comprehensive Ablation Study for Ensemble Classifiers (No. arXiv:2411.01645). arXiv. https://doi.org/10.48550/arXiv.2411.01645

27. Kearns, M., Neel, S., Roth, A., & Wu, Z. S. (2018). Preventing Fairness Gerrymandering: Auditing and Learning for Subgroup Fairness (No. arXiv:1711.05144). arXiv. https://doi.org/10.48550/arXiv.1711.05144

28. Kleinberg, J., Mullainathan, S., & Raghavan, M. (2016). Inherent Trade-Offs in the Fair Determination of Risk Scores (Version 2). arXiv. https://doi.org/10.48550/ARXIV.1609.05807

29. Kumar, V., & Boulanger, D. (2020). Explainable Automated Essay Scoring: Deep Learning Really Has Pedagogical Value. Frontiers in Education, 5, 572367. https://doi.org/10.3389/feduc.2020.572367

30. Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. Discourse Processes, 25(2–3), 259–284. https://doi.org/10.1080/01638539809545028

31. Lee, H., Stinar, F., Zong, R., Valdiviejas, H., Wang, D., & Bosch, N. (2025). Learning Behaviors Mediate the Effect of AI-powered Support for Metacognitive Calibration on Learning Outcomes. Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems, 1–18. https://doi.org/10.1145/3706598.3713960

32. Leinonen, T., Wong, D., Vasankari, A., Wahab, A., Nadarajah, R., Kaisti, M., & Airola, A. (2024). Empirical investigation of multi-source cross-validation in clinical ECG classification. Computers in Biology and Medicine, 183, 109271. https://doi.org/10.1016/j.compbiomed.2024.109271

33. Li, S., & Ng, V. (2024). Automated Essay Scoring: A Reflection on the State of the Art.

34. Linguistics (Volume 1: Long Papers), 12456–12470. https://doi.org/10.18653/v1/2023.acl-long.696

35. Loukina, A., Madnani, N., & Zechner, K. (2019b). The many dimensions of algorithmic fairness in educational applications. Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, 1–10. https://doi.org/10.18653/v1/W19-4401

36. Lundberg, S., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions

37. Maaten, L. van der, & Hinton, G. (2008). Visualizing Data using t-SNE. Journal of Machine Learning Research, 9(86), 2579–2605.

38. Misgna, H., On, B.-W., Lee, I., & Choi, G. S. (2024). A survey on deep learning-based automated essay scoring and feedback generation. Artificial Intelligence Review, 58(2), 36. https://doi.org/10.1007/s10462-024-11017-5

39. Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I. D., & Gebru, T. (2019a). Model Cards for Model Reporting. Proceedings of the Conference on Fairness, Accountability, and Transparency, 220–229. https://doi.org/10.1145/3287560.3287596

40. Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I. D., & Gebru, T. (2019b). Model Cards for Model Reporting. Proceedings of the Conference on Fairness, Accountability, and Transparency, 220–229.

41. Nauta, M., Trienes, J., Pathak, S., Nguyen, E., Peters, M., Schmitt, Y., Schlötterer, J., van Keulen, M., & Seifert, C. (2022). From Anecdotal Evidence to Quantitative Evaluation Methods: A Systematic Review on Evaluating Explainable AI. https://doi.org/10.48550/ARXIV.2201.08164

42. Noble, S. U. (2018). Algorithms of oppression: How search engines reinforce racism. New York university press.

43. Page, E. B. (1966). The imminence of grading essays by computer. Phi Delta Kappan, 238-243.

44. Papanikou, V., Karidi, D. P., Pitoura, E., Panagiotou, E., & Ntoutsi, E. (2025). Explanations as Bias Detectors: A Critical Study of Local Post-hoc XAI Methods for Fairness Exploration (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2505.00802

45. Parekh, S., Singla, Y. K., Chen, C., Li, J. J., & Shah, R. R. (2020). My Teacher Thinks The World Is Flat! Interpreting Automatic Essay Scoring Mechanism (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2012.13872

46. Pattern Recognition (CVPR), 9260–9269. https://doi.org/10.1109/CVPR.2019.00949

47. Pearl, J. (2009). Causality: Models, Reasoning, and Inference (2nd edn). Cambridge University Press. https://doi.org/10.1017/CBO9780511803161

48. Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, 17876–17888. https://doi.org/10.18653/v1/2024.emnlp-main.991

49. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 37–42. https://doi.org/10.18653/v1/P19-3007

50. Vig, J. (2019). A Multiscale Visualization of Attention in the Transformer Model.

51. Voss, E. (2025). Comparison of traditional machine learning and neural network approaches for automated scoring of second language English essays. Language Testing, 02655322251348959. https://doi.org/10.1177/02655322251348959

52. Wang, J., Wiens, J., & Lundberg, S. (2021). Shapley Flow: A Graph-based Approach to Interpreting Model Predictions (No. arXiv:2010.14592). arXiv. https://doi.org/10.48550/arXiv.2010.14592

53. Wang, Q. (2024). A multifaceted architecture to Automate Essay Scoring for assessing english article writing: Integrating semantic, thematic, and linguistic representations. Computers and Electrical Engineering, 118, 109308. https://doi.org/10.1016/j.compeleceng.2024.109308

54. Williamson, D. M., Xi, X., & Breyer, F. J. (2012a). 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

55. Williamson, D. M., Xi, X., & Breyer, F. J. (2012b). 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

56. Williamson, D. M., Xi, X., & Breyer, F. J. (2012c). 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

57. Yang, K., Raković, M., Li, Y., Guan, Q., Gašević, D., & Chen, G. (2024). Unveiling the Tapestry of Automated Essay Scoring: A Comprehensive Investigation of Accuracy, Fairness, and Generalizability (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2401.05655

58. Yang, R., Cao, J., Wen, Z., Wu, Y., & He, X. (2020). Enhancing Automated Essay Scoring Performance via Fine-tuning Pre-trained Language Models with Combination of Regression and Ranking. Findings of the Association for Computational Linguistics: EMNLP 2020, 1560–1569. https://doi.org/10.18653/v1/2020.findings-emnlp.141

59. Zhao, J., Wang, T., Yatskar, M., Ordonez, V., & Chang, K.-W. (2018). Gender Bias in Coreference Resolution: Evaluation and Debiasing Methods. Proceedings of the 2018 Conference of the North American Chapter of the

Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), 15–20. https://doi.org/10.18653/v1/N18-2003

60. Zhi-Hua Zhou & Xu-Ying Liu. (2006). Training cost-sensitive neural networks with methods addressing the class imbalance problem. IEEE Transactions on Knowledge and Data Engineering, 18(1), 63–77. https://doi.org/10.1109/TKDE.2006.17