

Rethinking Evaluation Practices in Automated Essay Scoring: A Quantitative Analysis over the Past Decade

Salam Albatarni, May Bashendy, Sohaila Eltanbouly, Marwan Sayed, Tamer Elsayed

Computer Science and Engineering Department, Qatar University

{sa1800633, ma1403845, se1403101, me2104862, telsayed}@qu.edu.qa

Abstract

Automated Essay Scoring (AES) aims to automatically assign quality scores to written essays and has been studied for several decades. Despite substantial methodological progress, little attention has been given to how AES systems are evaluated. In this position paper, we argue that several common AES evaluation practices are misaligned with broader machine learning evaluation standards. To examine current practices, we systematically analyzed 101 AES papers published in leading venues over the past decade, focusing on experimental setup, reproducibility, model development, and performance reporting and analysis. Consequently, we identify recurring shortcomings and highlight ways for more rigorous and transparent evaluation practices.

1 Introduction

Automated Essay Scoring (AES) is the task of automatically assigning quality scores to written essays, reducing time and cost, which is particularly valuable in large-scale assessments. AES encompasses both holistic and trait-based scoring; holistic scoring assigns a single overall quality score, whereas trait-based scoring evaluates the essay across multiple writing dimensions, such as organization and grammar. The development of AES systems typically follows two paradigms: prompt-specific and cross-prompt.¹ In prompt-specific setup, models are trained and evaluated on essays responding to the same prompt, allowing prompt specialization. In contrast, cross-prompt models are trained on essays from a set of source prompts and evaluated on unseen target prompts, aiming to improve generalization across different prompts. Approaches to these tasks include feature-engineered models built on traditional machine learning techniques (Chen and He, 2013; Phandi et al., 2015; Sayed

et al., 2025), neural network-based models (Tay et al., 2018; Xie et al., 2022), transformer-based models (Wang et al., 2022; Albatarni et al., 2026), and, more recently, large language model-based approaches (Yoshida, 2025a).

Despite this considerable methodological progress and long history (Page, 1966), the field has rarely paused to question whether its established practices are conceptually sound. Instead, most studies prioritize performance gains, often following established conventions, without critically examining their validity. Among the few exceptions, Beigman Klebanov and Madnani (2020) revisited foundational research on AES, showing that while human-rater agreement has been largely achieved, critical challenges around validity, fairness, and score interpretability remain unresolved. More recently, Li and Ng (2024a) broadened the discussion by calling for a shift in perspective, urging the community to move beyond aggregate metrics, diversify datasets, explore trait-based and cross-prompt scoring, and consider AES integration into educational contexts.

While those studies provide valuable reflections on the current state and future directions of AES, one key aspect that remains underexplored is *evaluation*. In this position paper, we critically analyze how AES systems are evaluated in practice, identifying systematic issues in experimental design and performance reporting. We argue that current AES evaluation paradigms are often misaligned with broader machine learning evaluation standards. To substantiate this claim, we present a systematic analysis of a *representative sample* of 101 AES studies published over the past decade in 10 leading NLP and AI venues.

Our analysis is informed by broader critiques of empirical practices in machine learning. In particular, Lones (2021) outlines common pitfalls and best practices in experimental design and evaluation. Likewise, Herrmann et al. (2024) calls for

¹Prompt refers to the text describing an essay writing task.

a reconsideration of established empirical conventions. While these works address machine learning broadly, we apply their perspective specifically to AES evaluation. Our contribution is four-fold: (1) We analyze 10 years of AES research across 10 leading publication venues, focusing specifically on evaluation practices. (2) We identify common evaluation pitfalls and quantify their prevalence through a systematic analysis of published papers. (3) We provide concrete recommendations and best practices to improve evaluation and reporting in future AES research.² To our knowledge, this is the first AES-specific work to offer a structured evaluation roadmap. (4) We make the annotations of our analysis public for transparency.³

2 Study Design

Recently, evaluation practices in AES research have raised concerns regarding their alignment with established machine learning standards, risking producing misleading conclusions and unreliable interpretations of model performance. To systematically examine these practices, we analyze a representative set of AES studies, focusing on how evaluation setups are designed, datasets are used, reproducibility is supported, and model performance is reported. Our study begins with the collection of relevant papers from major NLP and AI venues (§2.1), followed by an annotation process that captures key aspects of their evaluation practices (§2.2).

2.1 Paper Collection

We conduct a systematic analysis of AES papers published over the past decade (2016-2025 inclusive) in 7 leading NLP and AI conferences, namely ACL, EMNLP, NAACL, EACL, LREC, COLING, and AACL (covering main and findings tracks), where many influential AES papers have been introduced. We also include AES-relevant papers from educational NLP/AI venues, specifically the AIED and EDM conferences and the BEA Workshop.

Candidate papers were identified by searching titles for the following keywords: ‘essay(s)’, ‘student(s)’, ‘trait(s)’, ‘rubric(s)’, ‘scoring’, ‘writing’, ‘written’ and ‘proficiency’, yielding 941 papers. We note that our objective is *not* to provide an *exhaustive* survey

²We provide an AES evaluation checklist in Appendix C.

³Annotations and evaluation checklist is available at <https://gitlab.com/bigirqu/aes-evaluation-practices/>. For any annotation errors or issues, please contact us.

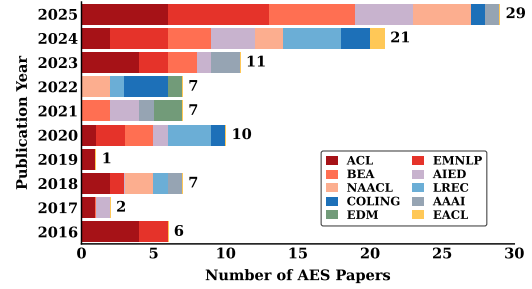


Figure 1: Distribution of reviewed AES papers.

of AES research, but rather to characterize evaluation practices in a *representative* set of AES studies published at influential NLP/AI venues. Accordingly, while our keyword-based strategy aims to maximize coverage, we do not claim complete recall of AES studies.

Next, we conducted a *two-stage screening*. In the first stage, we excluded papers that were not relevant to AES based on their titles (e.g., papers on teacher-student training paradigms), reducing the pool to 225 papers. In the second stage, we manually examined each remaining paper to determine whether it included an empirical evaluation of an AES system. We excluded conceptual papers with no empirical evaluation (e.g., position papers), papers focused on short-answer rather than essay scoring, and papers that introduced new datasets without benchmarking AES models. This resulted in a final set of 101 papers included in our analysis.

Figure 1 shows the distribution of the final 101 papers across years and venues. It indicates that, within our set, AES research has experienced a sharp increase in interest in recent years, rising from 11 papers in 2023 to 21 in 2024, and reaching 29 in 2025. Across venues, ACL emerged as the leading venue (21 papers), followed by EMNLP (18 papers), while EDM and EACL contributed relatively fewer works (3 and 1 papers, respectively).

2.2 Paper Annotation

This section describes the annotation process and the inter-annotator agreement analysis used to assess its consistency and reliability.

Annotation Process Our annotation framework focuses on 6 key evaluation aspects: *language and dataset coverage*, *model tuning*, *reproducibility*, *evaluation measures*, *performance evaluation*, and *performance analysis*. We prepared an annotation form (questionnaire) consisting of 23 questions covering the 6 aspects, with each linked to a subset

of questions (see Appendix A). The annotation process was conducted by four annotators (coauthors here) and consisted of three stages: *pilot study*, *main annotations*, and *disagreement resolution*.

In the pilot study, each annotator independently annotated two randomly selected papers, after which the annotation questionnaire was refined to address missing questions, clarify answer options, determine question formats, and introduce a "not applicable" option wherever necessary. This resulted in the finalized annotation form. In the main annotation phase, we adopted a double-annotation procedure to ensure reliability; each paper was independently evaluated by two distinct annotators. Annotators focused on the evaluation-related sections of their assigned papers to complete the questionnaire. Following that phase, annotators met to resolve any disagreements. In cases where the two could not reach agreement, a third annotator was consulted to determine the final decision. This structured process ensured rigor and reliability in the resulting annotations.

Inter-Annotator Agreement We compute inter-annotator agreement for each question using the raw responses (before disagreement resolution). We use Cohen’s Kappa κ (Cohen, 1960) for 16 questions with mutually exclusive answers, and Jaccard similarity (Jaccard, 1912) for 7 multi-label questions (Amigo and Delgado, 2022; Godbole and Sarawagi, 2004). This resulted in an average Cohen’s κ of 0.709 across 16 questions, which is considered *substantial* agreement (Landis and Koch, 1977). For the 7 multi-label questions, Jaccard similarity reaches an average of 0.83, indicating high overlap between annotators’ responses. A detailed breakdown of IAA is provided in Appendix B.

3 Current State and Pitfalls

Despite the growth of AES research in recent years, several issues persist in how AES systems are evaluated. In this section, we discuss these issues, focusing on dataset and language coverage, model tuning, reproducibility, evaluation measures, performance evaluation, and the depth of analysis provided across studies. Together, these aspects provide a critical view of current evaluation practices and highlight areas where they can be improved.

3.1 Language and Dataset Coverage

While our analysis reveals that the reviewed papers cover 14 different languages, it shows that

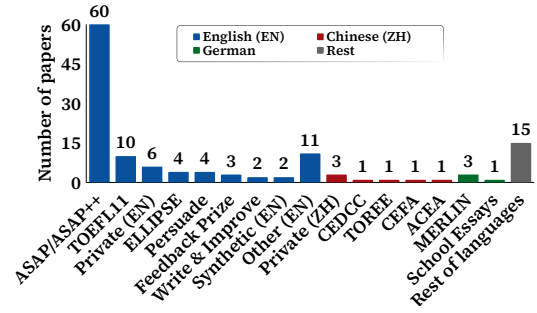


Figure 2: Datasets used across the reviewed AES papers.

the field remains highly imbalanced in that regard. Specifically, 82 of the 101 reviewed papers (81.2%) study English AES, while the remaining investigate other languages, e.g., Arabic (Qwaider et al., 2025), Chinese (Song et al., 2020), Korean (Song et al., 2025), and others. That dominance, illustrated in Figure 2, is further reflected in the availability of datasets, where English alone benefits from 25 AES datasets (including 6 private), compared to only 24 datasets across the other 13 languages combined (including 7 private). Our analysis further reveals limited dataset creation efforts: just 13 non-English datasets have been introduced in the past decade, averaging one per language. This clearly highlights the overwhelming concentration of both data resources and research efforts around English AES. As a result, benchmarking and systematic experimentation in low-resource languages remain limited, slowing progress toward advancing state-of-the-art performance. Over time, this imbalance risks biasing the AES research toward a single linguistic context, limiting global generalizability, and hindering broader innovation in the field.

An even more critical observation lies in the over-reliance within English AES research on using ASAP⁴ dataset for holistic scoring, and its extension ASAP++ (Mathias and Bhattacharyya, 2018) for trait-specific scoring, a concern previously noted by Li and Ng (2024a) and now reinforced and validated by our statistical evidence. As shown in Figure 2, among 25 identified English datasets, ASAP and ASAP++ are used in 60 of the 82 English studies (73.2%). In contrast, studies using other English datasets, such as TOEFL11 (Yoshida, 2025a), ELLIPSE (Ding et al., 2024), and PERSUADE (Lee et al., 2025), remain comparatively few. This heavy reliance on a single benchmark family (ASAP/ASAP++) introduces the risk of overfitting, where models are implicitly opti-

⁴<https://www.kaggle.com/c/asap-aes>

mized toward ASAP-specific population characteristics, rather than general representations of writing proficiency. Consequently, systems may perform well on ASAP/ASAP++, but fail to generalize to essays with different prompts, scoring rubrics, or characteristics. This is supported by Eltanbouly et al. (2026), who show that cross-dataset evaluation can substantially change AES model rankings.

Recommendations To promote a balanced and robust AES research field, we recommend: **(R1) Prioritizing the development and public release of datasets for low-resource languages** through dedicated data collection and annotation efforts. This would address the current resource gap, enable systematic benchmarking, and promote linguistic diversity in AES research. For example, recent efforts such QAES (Bashendy et al., 2024), TAQEEEM 2025 shared task (Bashendy et al., 2025), and LAILA dataset (Bashendy et al., 2026) represent valuable steps toward supporting Arabic AES. **(R2) Breaking the self-reinforcing cycle of over-reliance on ASAP/ASAP++ benchmark family.** While it served as a valuable resource in the community, greater attention should be directed toward other well-established public resources, such as ICLE++ (Li and Ng, 2024c) and ELLIPSE (Crossley et al., 2023a) for both holistic and trait-specific scoring, and PERSUADE (Crossley et al., 2023b) for holistic scoring. Moving beyond ASAP dominance mitigates the risk of benchmark-specific overfitting, and enhances generalizability while unlocking underexplored valuable resources.

3.2 Model Tuning

Reliable estimation and interpretation of model performance rely on how models are optimized, which depends on training and testing data splits, and how hyperparameters are selected and tuned.

Data Splitting In current AES research, different data splitting approaches are adopted depending on the task and dataset. Our analysis shows that 60 (59.4%) of the surveyed papers employ cross-validation (CV). For example, studies using ASAP/ASAP++ datasets, adopt the 5-fold CV splits introduced by Taghipour and Ng (2016) for prompt-specific setup (Kumar et al., 2022; Do et al., 2024a), and leave-one-prompt-out CV for cross-prompt setup (Ridley et al., 2021; Do et al., 2023). Similarly, leave-one-prompt-out CV is used with other datasets, such as ICLE++ (Li and Ng, 2024c) and TOEFL11 (Jiang et al., 2023).

Although CV is intended to reduce overfitting and provide reliable means for estimating model performance, certain practices can introduce *data leakage*. One such risk is *sequential overfitting*, defined by Lones (2021), where multiple models or configurations are evaluated in succession on the same test set. As results of earlier evaluations inform later decisions, information about the test set can leak into the training process. For example, selecting the best hyperparameter configuration based on average performance over all development folds, then applying it to all test folds (He et al., 2022; Jiang et al., 2023; Wang and Liu, 2025).

In contrast, 27 (26.7%) studies use fixed train/dev/test splits, as provided with datasets such as DREsS (Yoo et al., 2025). This approach avoids sequential overfitting; however, it may limit the generalizability of the results, as the model is optimized on a single (often small) fixed dev split. A more rigorous alternative is to use CV on the train/dev data with a held-out test set for final evaluation; however, this practice appears in only 2 papers (Doewes et al., 2022; Hwang, 2024).

Another practice observed in recent works is evaluating models on a small subset of the test set. This is particularly common in studies leveraging large language models (LLMs) (Lee et al., 2024; Jang et al., 2025), due to computational cost. While this choice is justified, it limits comparability with other studies that use the full test set.

Hyperparameter Tuning Our analysis reveals that 74 (73.3%) of the reviewed papers do not explicitly report performing hyperparameter tuning. Specifically, studies either present hyperparameters as fixed configurations without justification (Cummins et al., 2016; Chen et al., 2025), or adopt them from prior work (Do et al., 2024b, 2023), relying on the validation set only for early stopping. While these studies may have conducted hyperparameter tuning without documenting it, the lack of transparency raises questions about how hyperparameters were selected, whether the chosen configurations generalize to unseen data, and whether the models overfit to a specific dataset. Eltanbouly et al. (2026) show that hyperparameter tuning can alter model rankings by tuning multiple popular cross-prompt AES models, including PAES (Ridley et al., 2020), CTS (Ridley et al., 2021), and ProTACT (Do et al., 2023), which were originally evaluated using fixed hyperparameter configurations.

Recommendations To improve model optimization and evaluation, we recommend: **(R3) Performing fold-independent hyperparameter tuning** when using CV, to ensure that no observations from the test fold influence the training process. This practice, though rarely adopted in AES, is demonstrated by [Li and Ng \(2024b\)](#), where hyperparameter tuning is conducted independently for each test fold. **(R4) Reserving a held-out test set** for final evaluation that remains isolated from all optimization and tuning decisions. However, this may be challenging in AES research due to the typically-small dataset size. **(R5) When a held-out test set is not feasible, adopt cross-dataset evaluation.** In this setup, a model trained on one dataset is evaluated on an *unseen dataset*, as demonstrated by [Lee et al. \(2025\)](#). This provides a more rigorous assessment of generalization and better reflects real-world conditions where AES systems must evaluate essays from students with varying writing quality and language proficiency. **(R6) Avoidance of relying on fixed values of hyperparameters** adopted from prior work without validation, as optimal configurations are often dataset- and task-dependent.

3.3 Reproducibility

Reproducibility is a key requirement for research progress, ensuring that proposed methods can be independently verified, extended, and importantly evaluated. Achieving reproducibility spans 3 factors: availability of *source code*, completeness of reported *implementation details*, and transparency of documented *hyperparameter values*.

Source Code A key barrier to reproducibility in AES research is the limited public availability of source code. In our analysis, only 33 (32.7%) studies provided working code repositories ([Wang et al., 2024](#); [Chu et al., 2025](#)), leaving more than half of the surveyed studies with only the re-implementation option using what is described in the paper, which often omit crucial implementation choices. This significantly limits the ability to verify reported results, reproduce experimental setups, and reuse existing methods in subsequent works.

Implementation Details When code is unavailable, reproducibility relies heavily on the clarity of reported implementation details. Encouragingly, our analysis shows that 86 (85.1%) studies report implementation details that support, to varying degrees, replication, such as model architecture (e.g., number of layers, hidden dimensions), training con-

figurations (e.g., optimizer, early stopping criteria), data preprocessing, libraries, model size, and hardware specifications. A good example is provided by [Eltanbouly et al. \(2025\)](#), whose work, although not accompanied by publicly released code, provides sufficient details that enable replication.

Hyperparameter Values Hyperparameter settings are also critical for ensuring reproducibility, yet often under-reported. Our analysis shows that only 27 (26.7%) studies perform hyperparameter tuning, among these, 15 report the method used for tuning, such as Bayesian Optimization ([Alikaniotis et al., 2016](#)), and 16 share the search space of tested hyperparameter values ([Zeng et al., 2023](#)). These findings highlight that, although most studies report the optimal hyperparameter values (21 out of 27), many omit essential details about the tuning process, which significantly affect reproducibility. [Li and Ng \(2024b\)](#) exemplifies good practice by documenting hyperparameter values, optimization tool, search space, and the best configurations.

Recommendations To strengthen reproducibility and enable fair comparison, we recommend: **(R7) Releasing complete source code** in publicly accessible repositories, along with setup instructions, preprocessing, training, and evaluation scripts, and, when feasible, trained model checkpoints. **(R8) Reporting comprehensive implementation details**, including model architecture, libraries, and any method-specific choices. For example, in LLM-based methods, we encourage providing the prompts to support reproducibility. **(R9) Providing full hyperparameter information**, specifying the optimization method, search space, and optimal configurations. Transparency by following these 3 practices not only builds trust but also accelerates innovation by providing a clear starting point for future work. [Li and Ng \(2025\)](#) demonstrates best practices in reproducibility providing the source code, sufficient implementation details, and complete hyperparameter values, thereby strengthening transparency and credibility.

3.4 Evaluation Measures

The selection of the evaluation measure shapes how performance is interpreted, guiding model development, optimization, and research conclusions and directions. Measures indeed are more than quantitative figures; they encode assumptions about what counts as “good” performance.

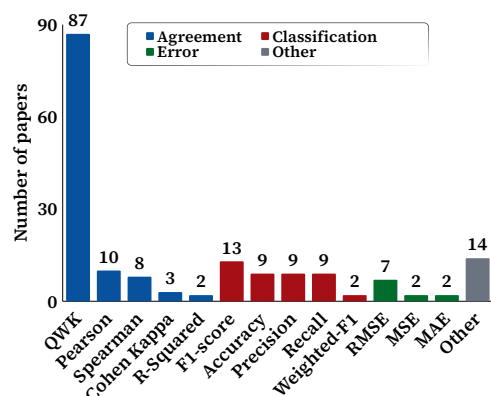


Figure 3: Evaluation measures in the reviewed papers.

Figure 3 shows the frequency of reported measures across three main categories in the literature: agreement, classification, and error measures, naturally, with agreement measures dominating. Overall, Quadratic Weighted Kappa (QWK) dominates, appearing in 87 papers (86.1%), followed by F1-score in 13 (12.9%) and Pearson’s r in 10 (9.9%).

QWK measures the level of agreement between two raters while penalizing disagreements according to their squared distance, making it a natural fit for ordinal scoring tasks such as essay grading. However, several studies have highlighted its limitations. Brenner and Kliebsch (1996) showed that QWK is sensitive to the number of scoring categories, especially in the 2–5 range typical of AES. Yannakoudakis and Cummins (2015) found it performed poorly on criteria like robustness and misclassification sensitivity. Albatarni and Elsayed (2026) showed that QWK is particularly sensitive to small datasets, while Loukina et al. (2020) highlighted its dependence on rater reliability, emphasizing the impact of reference human raters.

Our analysis shows the extent of the reliance on QWK: 64 studies (63.3% of the reviewed papers) report QWK as their *sole* measure. To provide a practical point of comparison, we also look at deployed AES systems from industry, e.g., ETS and Duolingo. They typically report complementary metrics such as F-score, Cohen’s Kappa, Spearman correlation, and exact agreement (Naismith et al., 2023; Farra et al., 2015). Some also include precision and recall (Burstein et al., 2010), providing a more nuanced picture of model performance.

Recommendations To better present AES performance, we recommend: **(R10) Reporting multiple evaluation measures rather than relying solely on QWK.** This is a standard best prac-

tice in fields like machine learning, ensuring a more complete and reliable assessment of system performance (Lones, 2021). **(R11) Choosing metrics deliberately based on task characteristics and potential limitations.** Prior work has proposed alternatives to QWK for AES. For example, Loukina et al. (2020) recommends using PRMSE when there are exactly two annotators whose annotations are available and label noise is present. Several studies have also recommended Gwet’s AC2 (Gwet, 2014) as an alternative to QWK. By reporting complementary measures alongside a primary metric aligned with the scoring objective, researchers can provide a more informative evaluation.

3.5 Performance Evaluation

The way AES studies present performance results is crucial to assessing progress in the field. Decisions about reporting granularity, baselines, and statistical testing shape the validity of comparisons and the credibility of the findings.

Result Reporting Granularity A key aspect of result presentation is granularity, which refers to the level of detail at which performance is reported. In AES studies, results may be broken down by individual prompt and trait, or summarized into broader averages. Fine-grained reporting is crucial as it exposes the strengths and weaknesses that aggregated scores can obscure, offering a more transparent and comprehensive view of performance. Yet, despite its importance, this practice remains uncommon; only 10 (22.2%) out of the 45 papers that focus on trait-specific scoring report fine-grained results for each trait across individual prompts (Mathias and Bhattacharyya, 2018, 2020). Most studies, however, rely on aggregated reporting: 35 (77.8%) of 45 studies report results as averages, either per trait, per prompt, or both (Do et al., 2023; Wang and Liu, 2025). While aggregation provides a convenient overview, it reduces the number of data points available for statistical analysis and hides informative variations in model performance.

Baseline Selection Selecting suitable baselines and clearly justifying their inclusion is essential for positioning a new method within the existing AES literature. However, our analysis shows that this practice is often neglected. Among the 101 reviewed papers, 43 studies (42.6%) do not include external baselines. Of these, 14 are dataset papers, where the main contribution is typically resource construction rather than model comparison. The re-

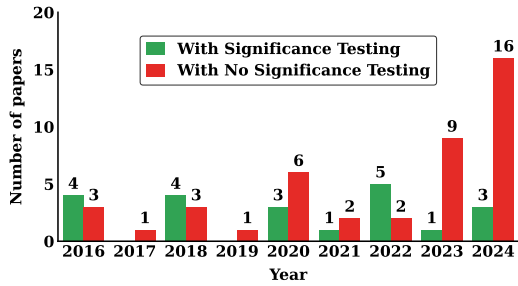


Figure 4: Statistical significance testing over time.

maining 29 papers compare only against variants of their own proposed method. In LLM-based work, this pattern is often linked to resource constraints (Yoshida, 2025a). Even when external baselines are included, their selection is often insufficiently justified. In fact, among the 58 papers that include external baselines, 27 do not justify their choice.

Statistical Significance Using statistical tests when comparing models is an important practice, as numerical differences alone are not sufficient to establish preference of one approach over another (Dror et al., 2018). However, as discussed earlier, the tendency to collapse results into averages reduces the number of comparable data points, making statistical significance testing difficult to achieve. Works like Mathias and Bhattacharyya (2020) show how finer reporting granularity enables this: by covering all ASAP++ prompts and traits, sufficient data points are available to conduct reliable significance testing as a natural part of evaluation. Despite this, 70 (69.3%) papers do not report significance testing, and Figure 4 suggests that this has become more prevalent during the last three years. Even when datasets provide sufficient data points, significance testing is often neglected; for example, ELLIPSE dataset includes 44 prompts, yet none of the three papers using ELLIPSE reported statistical significance testing.

Generalizability Naturally, evaluating models across multiple datasets is essential to ensure model generalizability (Lones, 2021). In practice, however, only 22 (21.7%) studies evaluate across multiple datasets. This is particularly evident in English AES, where 44 of the 60 studies using ASAP or ASAP++ (73.3%) evaluate on it exclusively.

Recommendations To improve performance reporting in AES research, we recommend: **(R12) Reporting fine-grained performance** presented per prompt and per trait, rather than col-

lapsed into averages. This not only improves transparency, but makes significance testing more feasible. **(R13) Comparing with proper baselines** wherever applicable. Baseline selection should be rational and cover state-of-the-art systems with respect to task performance and approach. **(R14) Conducting significance testing** to highlight statistically-meaningful differences between models. **(R15) Evaluating on multiple datasets** whenever feasible, to improve the generalizability of the reported findings.

3.6 Performance Analysis

Beyond reporting baseline comparisons and performance scores, deeper analysis is essential to understand why a model achieves its performance and what resources are required to obtain it.

Model Analysis For each reviewed paper, we examine the types of analyses conducted after the main experimental results. The most common form is an *ablation study*, appearing in 63 papers (62.4%). Other forms include failure analysis (24), trait-based analysis (16), feature analysis (20), embedding visualization (10), and qualitative analysis (7). Additionally, 30 papers report other forms of analysis tailored to their specific models, languages, or experimental settings, such as varying the number of training prompts to assess its effect (Cummins et al., 2016) or examining language similarity in multilingual settings (He and Li, 2024).

Overall, only 4 papers report no additional analysis beyond the main results, reflecting a positive common practice in AES research. However, the strong dominance of ablation studies is concerning. While they help identify which model components contribute to overall performance, they provide limited insights into model behavior, such as failure patterns, error types, or even success cases.

Model Efficiency Although educational settings do not typically require instant scoring, a model that matches state-of-the-art performance while being more efficient represents a stronger contribution. Efficiency encompasses both training time (affecting resource allocation and reproducibility) and inference time (indicating how a deployed system can scale). Model size is closely related to both, as larger models generally demand more memory, longer training runs, and slower inference, particularly for LLM-based methods, where essay scoring can take seconds compared to milliseconds for non-LLM approaches (Elbahnasawi et al., 2026). In

practice, however, such reporting remains uncommon: only 17 (17.3%) papers disclose training time (Li and Pan, 2025) and only 7 (7.1%) report inference time (Eltanbouly et al., 2025), out of 98 papers (excluding 3 not applicable cases). Similarly, only 27 (30.7%) papers report model size (excluding 13 not applicable cases). For papers studying LLM-based models (28), the monetary cost of using proprietary models is an additional relevant dimension, yet only 3 of 21 (excluding the 7 studies that used only open-source LLMs) such papers disclose it. Understanding effectiveness-efficiency trade-offs is important for assessing whether a proposed approach is practically viable.

Recommendations To address these shortcomings, we recommend: **(R16) Conducting performance analysis beyond scoring effectiveness.** A single performance figure is not sufficient to understand a model’s behavior. Complementary analyses, such as failure analysis or trait-level breakdowns, provide a deeper understanding of where a model succeeds and where it falls short. **(R17) Reporting efficiency and cost figures,** including training and inference time, model size, and monetary cost. This directly affects the feasibility of deployment in real-world or low-resource settings.

4 LLM-era Caveats

The recent adoption of LLMs in AES has grown rapidly, with our analysis showing that 28 out of 61 papers (45.9%) published over the past 3 years (2023-2025) used LLMs. This growing adoption highlights the need to examine the evaluation considerations specific to LLM-based AES systems. Unlike conventional AES models, LLM-based systems introduce additional sources of variability and uncertainty related to data contamination, non-determinism, prompt sensitivity, reproducibility, and computational cost. We therefore identify the following LLM-specific caveats that should be considered when evaluating LLM-based AES systems.

Data Contamination Widely used benchmarks, such as ASAP, have been publicly available for over a decade, and may have appeared in LLM pretraining corpora, raising concerns about data contamination. Despite the increasing use of LLMs in AES, few studies discuss contamination risks or evaluate models on newer or less-accessible datasets.

Non-deterministic Output Unlike traditional AES models, which produce deterministic predic-

tions once trained, LLM outputs may vary across runs due to decoding randomness, API updates, or configuration differences. While this variability in open-source models can often be reduced by controlling decoding parameters, reproducibility remains challenging for closed-source models, which may change over time without transparent versioning. Our analysis reveals that, out of the 28 (27.7%) papers that use LLMs, the majority of them (20 papers; 71.4%) use at least one open-source model (Lee et al., 2024). However, 8 of the 28 papers (28.6%) rely on closed-source models, raising additional concerns regarding reproducibility and comparability of results across studies.

Prompt Sensitivity Small changes in prompt wording, rubric structure, or few-shot examples can substantially influence LLM scoring behavior. This issue has been investigated in several AES studies (Mansour et al., 2024; Yoshida, 2025b), which show that even minor prompt modifications can substantially affect scoring performance.

Cost and Efficiency Closed-source LLMs introduce financial costs and API usage constraints rarely reported in AES evaluations, with only 3 out of 21 studies that used closed-source LLMs reporting cost figures (Mansour et al., 2024; Jang et al., 2025; Chu et al., 2025). Moreover, for inference, those models are tied to API speed and largely uncontrollable. In contrast, open-source models raise concerns about in-house computational resources and inference latency, as inference time depends heavily on hardware specifications, model size, and configurations like model quantization and batch size. Nevertheless, only 3 out of 20 studies that use open-source LLMs reported inference time.

Recommendations To address those caveats, we recommend: **(R18) Evaluating LLM-based systems on private or carefully controlled datasets** whenever possible to reduce the risk of data contamination and improve the reliability of evaluation. **(R19) Repeating experiments** and reporting average performance with standard deviation to provide more reliable estimates and account for output variability. **(R20) Reporting complete evaluation setup,** including exact prompts, few-shot examples, decoding settings, and model versions. **(R21) Validating methods with open-source LLMs** to support independent verification of results. **(R22) Reporting API costs and inference time,** along with hardware and model configurations.

5 Roadmap

Considering the analysis of the reviewed studies, observed pitfalls, and corresponding recommendations, we outline a structured roadmap to guide the AES community towards stronger evaluation practices. We distinguish between *essential recommendations* (R3, R8–R17, R20–R21), which, we believe, are necessary for rigorous and reliable evaluation, and *strongly recommended practices* (R1, R2, R4–R7, R18–R19, R22), which further enhance evaluation robustness. To improve the usability of the proposed recommendations, we provide a checklist in Appendix C. The roadmap is organized across three stages, each linked to relevant recommendations.

Experimental Setup: In the first stage, several decisions should be established to ensure a sound evaluation setup: (1) **Select multiple datasets** (R2, R5, R15, R18) to assess the generalizability across different writing tasks, and learner populations. Multilingual evaluation is also encouraged, with attention to low-resource languages. (2) **Carefully define the data-split strategy** (R3, R4) following established best practices, and when feasible, reserve a held-out test set for final evaluation. (3) **Define the system’s goal and align it with the optimization measure** (R11). For example, real-time systems must balance predictive performance with computational efficiency. (4) **Choose strong and relevant baselines** (R13) before model development to avoid biased selection and establish clear performance target.

Model Development: In the second stage, researchers should adopt practices that ensure both effective tuning and transparency. In particular: (5) **Emphasize Hyperparameter tuning** (R3, R6), with efficient strategies such as Bayesian optimization preferred over grid search. (6) **Prioritize transparency and reproducibility** (R7, R8, R9, R19–R21), with training and optimization process clearly documented. Researchers should report all relevant experimental details, including preprocessing steps, hyperparameter configurations, model size, and model-specific design choices.

Performance Reporting and Analysis: In the final stage, performance should be presented in a way that provides a comprehensive understanding of the model’s behavior. (7) **Report fine-grained performance** (R10, R11, R12, R14), going beyond aggregated averages by providing

per-prompt and per-trait figures. Use multiple measures to capture diverse dimensions of performance. Improvements over baselines should be supported by statistical significance testing, while comparable performance with added efficiency and interpretability should also be highlighted. (8) **Conduct model analysis** (R5, R16, R17, R22) by reporting strengths and limitations of the approach through ablation studies, failure analysis, and method-specific investigations. Document practical factors such as inference time, model size, and financial cost, particularly for proprietary LLM-based approaches. (9) **Promote open resources and community efforts** (R1, R7) by releasing useful resources, such as extracted features, rationales, and well-documented code, to enable reproducibility and accelerate progress. Shared tasks should also be promoted to drive AES progress, especially in low-resource languages.

6 Conclusion

This paper examined evaluation practices in AES research by analyzing 101 representative papers published over the past decade in 10 leading NLP and AI venues. Across six evaluation dimensions, we identified several recurring issues: heavy reliance on English and few benchmarks, limited reproducibility, dependence on one evaluation measure, coarse performance reporting, poor baseline justification, lack of statistical significance testing, and insufficient analysis of results. Accordingly, we provided practical recommendations to help address these limitations. We further examined evaluation considerations that arise from using LLMs in AES. Finally, we outlined a roadmap that offers an applicable starting point for improving evaluation practices in future AES research.

Acknowledgments

This work was made possible by NPRP grant NPRP14S-0402-210127 from the Qatar Research Development and Innovation (QRDI) Council. The work of Sohaila Eltanbouly was supported by GSRA grant# GSRA12-L-0413-250111 from QRDI, and the work of Tamer Elsayed was also supported by the awarded research fund QUEXCENG-GOOGLE-2025-1 from Google. The statements made herein are solely the responsibility of the authors.

7 Limitations

This work has several methodological and implementation limitations that should be acknowledged.

First, our analysis focuses on papers published in 10 major NLP and AI venues over the past decade. While this scope does not encompass all AES research, it captures a representative body of work from leading venues where many influential AES methods are initially introduced and where broader research directions in the field are often established.

Second, our paper collection relies on keyword matching against paper titles, which may have caused relevant papers to be missed if they did not contain any of the specified terms. This introduces a potential coverage gap in our analysis.

Third, our analysis is based solely on what is explicitly reported in published papers. When experimental details are not clearly described, annotations reflect what could be inferred from the text, which may not always align with the actual practices followed by the authors.

Finally, this position paper focuses specifically on evaluation practices in AES research. Other important aspects of AES system development are beyond its scope, but may also warrant critical examination in future research.

References

- Salam Albatarni, May Bashendy, Sohaila Eltanbouly, and Tamer Elsayed. 2026. [MAPLE: A meta-learning framework for cross-prompt essay scoring](#). In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 19948–19961, San Diego, California, United States. Association for Computational Linguistics.
- Salam Albatarni and Tamer Elsayed. 2026. [Quadratic weighted kappa is not enough for evaluating automated essay scoring models](#). In *Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026)*, pages 4447–4456, Palma, Mallorca, Spain. European Language Resources Association (ELRA).
- Dimitrios Alikaniotis, Helen Yannakoudakis, and Marek Rei. 2016. [Automatic text scoring using neural networks](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 715–725, Berlin, Germany. Association for Computational Linguistics.
- Enrique Amigo and Agustín Delgado. 2022. [Evaluating extreme hierarchical multi-label classification](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5809–5819, Dublin, Ireland. Association for Computational Linguistics.
- May Bashendy, Salam Albatarni, Sohaila Eltanbouly, Walid Massoud, Houda Bouamor, and Tamer Elsayed. 2025. [TAQEEM 2025: Overview of the first shared task for Arabic quality evaluation of essays in multi-dimensions](#). In *Proceedings of The Third Arabic Natural Language Processing Conference: Shared Tasks*, pages 966–976, Suzhou, China. Association for Computational Linguistics.
- May Bashendy, Salam Albatarni, Sohaila Eltanbouly, Eman Zahran, Hamdo Elhuseyin, Tamer Elsayed, Walid Massoud, and Houda Bouamor. 2024. [QAES: First publicly-available trait-specific annotations for automated scoring of Arabic essays](#). In *Proceedings of The Second Arabic Natural Language Processing Conference*, pages 337–351, Bangkok, Thailand. Association for Computational Linguistics.
- May Bashendy, Walid Massoud, Sohaila Eltanbouly, Salam Albatarni, Marwan Sayed, Abrar Abir, Houda Bouamor, and Tamer Elsayed. 2026. [LAILA: A large trait-based dataset for Arabic automated essay scoring](#). In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3074–3091, Rabat, Morocco. Association for Computational Linguistics.
- Beata Beigman Klebanov and Nitin Madnani. 2020. [Automated evaluation of writing – 50 years and counting](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7796–7810, Online. Association for Computational Linguistics.
- Hermann Brenner and Ulrike Kliebsch. 1996. Dependence of weighted kappa coefficients on the number of categories. *Epidemiology*, pages 199–202.
- Jill Burstein, Joel Tetreault, and Slava Andreyev. 2010. [Using entity-based features to model coherence in student essays](#). In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 681–684, Los Angeles, California. Association for Computational Linguistics.
- Hongbo Chen and Ben He. 2013. [Automated essay scoring by maximizing human-machine agreement](#). In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1741–1752, Seattle, Washington, USA. Association for Computational Linguistics.
- Po-Kai Chen, Bo-Wei Tsai, Shao Kuan Wei, Chien-Yao Wang, Jia-Ching Wang, and Yi-Ting Huang. 2025. [Mixture of ordered scoring experts for cross-prompt essay trait scoring](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18071–18084, Vienna, Austria. Association for Computational Linguistics.

- SeongYeub Chu, Jong Woo Kim, Bryan Wong, and Mun Yong Yi. 2025. [Rationale behind essay scores: Enhancing S-LLM’s multi-trait essay scoring with rationale generated by LLMs](#). In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5811–5829, Albuquerque, New Mexico. Association for Computational Linguistics.
- Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.
- Scott Crossley, Yu Tian, Perpetual Baffour, Alex Franklin, Youngmeen Kim, Wesley Morris, Meg Benner, Aigner Picou, and Ulrich Boser. 2023a. The English language learner insight, proficiency and skills evaluation (ELLIPSE) corpus. *International Journal of Learner Corpus Research*, 9(2):248–269.
- Scott Andrew Crossley, Perpetual Baffour, Yu Tian, Alex Franklin, Meg Benner, and Ulrich Boser. 2023b. A large-scale corpus for assessing written argumentation: PERSUADE 2.0. Available at SSRN 4795747.
- Ronan Cummins, Meng Zhang, and Ted Briscoe. 2016. [Constrained multi-task learning for automated essay scoring](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–799, Berlin, Germany. Association for Computational Linguistics.
- Yuning Ding, Omid Kashefi, Swapna Somasundaran, and Andrea Horbach. 2024. [When argumentation meets cohesion: Enhancing automatic feedback in student writing](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 17513–17524, Torino, Italia. ELRA and ICCL.
- Heejin Do, Yunsu Kim, and Gary Lee. 2024a. [Autoregressive score generation for multi-trait essay scoring](#). In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1659–1666, St. Julian’s, Malta. Association for Computational Linguistics.
- Heejin Do, Yunsu Kim, and Gary Geunbae Lee. 2023. [Prompt- and trait relation-aware cross-prompt essay trait scoring](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1538–1551, Toronto, Canada. Association for Computational Linguistics.
- Heejin Do, Sangwon Ryu, and Gary Lee. 2024b. [Autoregressive multi-trait essay scoring via reinforcement learning with scoring-aware multiple rewards](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16427–16438, Miami, Florida, USA. Association for Computational Linguistics.
- Afrizal Doewes, Akirati Saxena, Yulong Pei, and Mykola Pechenizkiy. 2022. [Individual fairness evaluation for automated essay scoring system](#). In *Proceedings of the 15th International Conference on Educational Data Mining*, pages 206–216, Durham, United Kingdom. International Educational Data Mining Society.
- Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. [The Hitchhiker’s guide to testing statistical significance in natural language processing](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1383–1392, Melbourne, Australia. Association for Computational Linguistics.
- Hoor Elbahnasawi, Marwan Sayed, Sohaila Eltanbouly, Fatima Zahra Brahamia, and Tamer Elsayed. 2026. [Qayyem: A real-time platform for scoring proficiency of Arabic essays](#). In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 536–545, San Diego, California, United States. Association for Computational Linguistics.
- Sohaila Eltanbouly, Salam Albatarni, and Tamer Elsayed. 2025. [TRATES: Trait-specific rubric-assisted cross-prompt essay scoring](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20528–20543, Vienna, Austria. Association for Computational Linguistics.
- Sohaila Eltanbouly, Marwan Sayed, and Tamer Elsayed. 2026. [Is one dataset enough for evaluation? studying generalizability of automated essay scoring models](#). In *Proceedings of the Fifteenth Language Resources and Evaluation Conference*, pages 431–440, Palma de Mallorca, Spain. ELRA Language Resource Association.
- Noura Farra, Swapna Somasundaran, and Jill Burstein. 2015. [Scoring persuasive essays using opinions and their targets](#). In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 64–74, Denver, Colorado. Association for Computational Linguistics.
- Shantanu Godbole and Sunita Sarawagi. 2004. Discriminative methods for multi-labeled classification. In *Advances in Knowledge Discovery and Data Mining*, pages 22–30, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Kilem Li Gwet. 2014. The definitive guide to measuring the extent of agreement among raters. *Handbook of inter-rater reliability. 4th Edition. Advanced Analytics, LLC*, 429.
- Junyi He and Xia Li. 2024. [Zero-shot cross-lingual automated essay scoring](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 17819–17832, Torino, Italia. ELRA and ICCL.
- Yaqiong He, Feng Jiang, Xiaomin Chu, and Peifeng Li. 2022. [Automated Chinese essay scoring from multiple traits](#). In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3007–3016, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

- Moritz Hermann, F. Julian D. Lange, Katharina Eggersperger, Giuseppe Casalicchio, Marcel Wever, Matthias Feurer, David Rügamer, Eyke Hüllermeier, Anne-Laure Boulesteix, and Bernd Bischl. 2024. [Position: Why we must rethink empirical research in machine learning](#). In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 18228–18247. PMLR.
- Haerim Hwang. 2024. The role of first language in automated essay grading for second language writing. In *Artificial Intelligence in Education*, pages 302–310, Cham. Springer Nature Switzerland.
- Paul Jaccard. 1912. The distribution of the flora in the alpine zone. 1. *New phytologist*, 11(2):37–50.
- Jinhee Jang, Ayoung Moon, Minkyung Jung, Young-Bin Kim, and Seung Jin Lee. 2025. [LLM agents at the roundtable: A multi-perspective and dialectical reasoning framework for essay scoring](#). In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 19674–19687, Suzhou, China. Association for Computational Linguistics.
- Zhiwei Jiang, Tianyi Gao, Yafeng Yin, Meng Liu, Hua Yu, Zifeng Cheng, and Qing Gu. 2023. [Improving domain generalization for prompt-aware essay scoring via disentangled representation learning](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12456–12470, Toronto, Canada. Association for Computational Linguistics.
- Rahul Kumar, Sandeep Mathias, Sriparna Saha, and Pushpak Bhattacharyya. 2022. [Many hands make light work: Using essay traits to automatically score essays](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1485–1495, Seattle, United States. Association for Computational Linguistics.
- J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.
- Sanwoo Lee, Yida Cai, Desong Meng, Ziyang Wang, and Yunfang Wu. 2024. [Unleashing large language models’ proficiency in zero-shot essay scoring](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 181–198, Miami, Florida, USA. Association for Computational Linguistics.
- Sanwoo Lee, Kun Liang, and Yunfang Wu. 2025. [Composable cross-prompt essay scoring by merging models](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 24386–24400, Suzhou, China. Association for Computational Linguistics.
- Shengjie Li and Vincent Ng. 2024a. [Automated essay scoring: A reflection on the state of the art](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17876–17888, Miami, Florida, USA. Association for Computational Linguistics.
- Shengjie Li and Vincent Ng. 2024b. [Conundrums in cross-prompt automated essay scoring: Making sense of the state of the art](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7661–7681, Bangkok, Thailand. Association for Computational Linguistics.
- Shengjie Li and Vincent Ng. 2024c. [ICLE++: Modeling fine-grained traits for holistic essay scoring](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8465–8486, Mexico City, Mexico. Association for Computational Linguistics.
- Shengjie Li and Vincent Ng. 2025. [Graph-based multi-trait essay scoring](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 33325–33351, Suzhou, China. Association for Computational Linguistics.
- Xia Li and Wenjing Pan. 2025. [CEAES: Bidirectional reinforcement learning optimization for consistent and explainable essay assessment](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 26267–26279, Vienna, Austria. Association for Computational Linguistics.
- Michael A Lones. 2021. How to avoid machine learning pitfalls: a guide for academic researchers. *arXiv preprint arXiv:2108.02497*.
- Anastassia Loukina, Nitin Madnani, Aoife Cahill, Lili Yao, Matthew S. Johnson, Brian Riordan, and Daniel F. McCaffrey. 2020. [Using PRMSE to evaluate automated scoring systems in the presence of label noise](#). In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 18–29, Seattle, WA, USA → Online. Association for Computational Linguistics.
- Watheq Ahmad Mansour, Salam Albatarni, Sohaila Eltanbouly, and Tamer Elsayed. 2024. [Can large language models automatically score proficiency of written essays?](#) In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2777–2786, Torino, Italia. ELRA and ICCL.
- Sandeep Mathias and Pushpak Bhattacharyya. 2018. [ASAP++: Enriching the ASAP automated essay grading dataset with essay attribute scores](#). In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

- Sandeep Mathias and Pushpak Bhattacharyya. 2020. [Can neural networks automatically score essay traits?](#) In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 85–91, Seattle, WA, USA → Online. Association for Computational Linguistics.
- Ben Naismith, Phoebe Mulcaire, and Jill Burstein. 2023. [Automated evaluation of written discourse coherence using GPT-4.](#) In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 394–403, Toronto, Canada. Association for Computational Linguistics.
- Ellis B Page. 1966. The imminence of... grading essays by computer. *The Phi Delta Kappan*, 47(5):238–243.
- Peter Phandi, Kian Ming A. Chai, and Hwee Tou Ng. 2015. [Flexible domain adaptation for automated essay scoring using correlated linear regression.](#) In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 431–439, Lisbon, Portugal. Association for Computational Linguistics.
- Chatrine Qwaider, Bashar Alhafni, Kirill Chirkunov, Nizar Habash, and Ted Briscoe. 2025. [Enhancing Arabic automated essay scoring with synthetic data and error injection.](#) In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pages 549–563, Vienna, Austria. Association for Computational Linguistics.
- Robert Ridley, Liang He, Xin-Yu Dai, Shujian Huang, and Jiajun Chen. 2021. [Automated cross-prompt scoring of essay traits.](#) In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13745–13753. AAAI Press.
- Robert Ridley, Liang He, Xinyu Dai, Shujian Huang, and Jiajun Chen. 2020. Prompt agnostic essay scorer: a domain generalization approach to cross-prompt automated essay scoring. *arXiv preprint arXiv:2008.01441*.
- Marwan Sayed, Sohaila Eltanbouly, May Bashendy, and Tamer Elsayed. 2025. [Feature engineering is not dead: A step towards state of the art for Arabic automated essay scoring.](#) In *Proceedings of The Third Arabic Natural Language Processing Conference*, pages 231–245, Suzhou, China. Association for Computational Linguistics.
- SeungWoo Song, Junghun Yuk, ChangSu Choi, HanGyeol Yoo, HyeonSeok Lim, KyungTae Lim, and Jungyeul Park. 2025. [Unified automated essay scoring and grammatical error correction.](#) In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4412–4426, Albuquerque, New Mexico. Association for Computational Linguistics.
- Wei Song, Kai Zhang, Ruiji Fu, Lizhen Liu, Ting Liu, and Miaomiao Cheng. 2020. [Multi-stage pre-training for automated Chinese essay scoring.](#) In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6723–6733, Online. Association for Computational Linguistics.
- Kaveh Taghipour and Hwee Tou Ng. 2016. [A neural approach to automated essay scoring.](#) In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1882–1891, Austin, Texas. Association for Computational Linguistics.
- Yi Tay, Minh C. Phan, Luu Anh Tuan, and Siu Cheung Hui. 2018. Skipflow: incorporating neural coherence features for end-to-end automatic text scoring. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI’18/IAAI’18/EAAI’18*. AAAI Press.
- Jiong Wang and Jie Liu. 2025. [T-MES: Trait-aware mix-of-experts representation learning for multi-trait essay scoring.](#) In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 1224–1236, Abu Dhabi, UAE. Association for Computational Linguistics.
- Yongjie Wang, Chuang Wang, Ruobing Li, and Hui Lin. 2022. [On the use of bert for automated essay scoring: Joint learning of multi-scale essay representation.](#) In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3416–3425, Seattle, United States. Association for Computational Linguistics.
- Yupei Wang, Renfen Hu, and Zhe Zhao. 2024. [Beyond agreement: Diagnosing the rationale alignment of automated essay scoring methods based on linguistically-informed counterfactuals.](#) In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8906–8925, Miami, Florida, USA. Association for Computational Linguistics.
- Jiayi Xie, Kaiwei Cai, Li Kong, Junsheng Zhou, and Weiguang Qu. 2022. [Automated essay scoring via pairwise contrastive regression.](#) In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2724–2733, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.
- Helen Yannakoudakis and Ronan Cummins. 2015. Evaluating the performance of automated text scoring systems. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 213–223.
- Haneul Yoo, Jieun Han, So-Yeon Ahn, and Alice Oh. 2025. [DREsS: Dataset for rubric-based essay scoring on EFL writing.](#) In *Proceedings of the 63rd Annual Meeting of the Association for Computational*

Linguistics (Volume 1: Long Papers), pages 13439–13454, Vienna, Austria. Association for Computational Linguistics.

Lui Yoshida. 2025a. [Are the reasoning models good at automated essay scoring?](#) In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 8388–8394, Suzhou, China. Association for Computational Linguistics.

Lui Yoshida. 2025b. [Do we need a detailed rubric for automated essay scoring using large language models?](#) In *Artificial Intelligence in Education – 26th International Conference, AIED 2025, Palermo, Italy, July 22–26, 2025, Proceedings, Part VI*, volume 15882 of *Lecture Notes in Computer Science*, pages 60–67. Springer.

Zijie Zeng, Dragan Gasevic, and Guangliang Chen. 2023. [On the effectiveness of curriculum learning in educational text scoring](#). *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(12):14602–14610.

A Annotated Questions

We provide the full annotated questions we used in this study in Table 1.

B Inter-annotator Agreement Breakdown

This section presents the inter-annotator agreement (IAA) for each question. Since our questionnaire contains both mutually exclusive and multi-label questions, different agreement measures are used accordingly. For mutually exclusive questions, we compute Cohen’s Kappa, a standard measure for multi-class inter-annotator agreement. For multi-label questions, Cohen’s Kappa is not directly applicable; therefore, following Amigo and Delgado (2022); Godbole and Sarawagi (2004), we use Jaccard similarity instead. In total, seven questions were annotated as multi-label. Figures 5 and 6 summarize the IAA results across all questions.

For the mutually exclusive questions shown in Figure 5, the average IAA is 0.71, which corresponds to substantial agreement (Landis and Koch, 1977). However, Q15 achieved only fair agreement. This question asked: “Do they provide implementation details?” The lower agreement stems from the subjective nature of the question. For example, some LLM-based papers lack detailed implementation descriptions; however, some annotators considered simply including the prompt used as sufficient to qualify as implementation details. To ensure consistency, all disagreements for this and other questions were discussed and resolved by the annotators.

Beyond these, two questions achieved only moderate agreement: Q6 (“If cross-validation is used, is there a held-out test set?”) and Q16 (“Do they use a hyperparameter tuning tool?”). For Q6, disagreement mainly arose because the data-splitting procedure was not always explicitly described, making it difficult for annotators to determine whether a separate test set was used. All remaining questions achieved either substantial or almost perfect agreement.

For the multi-label questions, all questions achieved a Jaccard similarity above 0.75, except for Q23 (“What analyses are performed?”). This question was inherently more challenging, as, although we attempted to cover common analysis categories, many papers include method-specific analyses that are difficult to fully capture in a pre-defined set of labels. As a result, annotations were more dependent on individual interpretation. Nevertheless, all disagreements for this question were ultimately resolved through discussion.

C Evaluation Checklist

This checklist is intended to support the evaluation and reporting of automated essay scoring (AES) systems.⁵ It summarizes common best practices recommended to be adopted in AES research. For a broader checklist on machine learning experimentation and reproducibility, refer to REFORMS checklist (Lones, 2021).⁶

Ideally, your answer should be “Yes” to all questions; if not, you should clearly justify why not in your paper.

C.1 Stage A: Experimental Setup

- A1** Did you explicitly specify whether your method is prompt-specific (trained and tested on the same prompt) or cross-prompt (trained on some prompts and evaluated on unseen ones)?
- A2** Did you evaluate your method on more than one dataset (e.g., not only on ASAP)?
- A3** If your method is not language-specific, did you evaluate it on another language?
- A4** Did you consider cross-dataset evaluation (training a model on one dataset, and testing it

⁵The checklist is available at <https://gitlab.com/bigirqu/aes-evaluation-practices/>

⁶<https://reforms.cs.princeton.edu/checklist.docx>

Aspect	Q#	Question	Choices
Language and Dataset Coverage	1	Does the paper introduce a new dataset?	Yes; No
	2	What are the supported languages?	English; Other.
	3	What is the dataset used?	ASAP/ASAP++; TOEFL11; PERSUADE; ELLIPSE; Private/Non-public dataset; Other.
Model Tuning	4	Is hyperparameter tuning performed?	Yes (Other than early stopping, any hyperparameter tuning is considered); No.
	5	What is the dataset split used in the paper?	No split (e.g. Zero-shot); Cross-validation; Cross-validation (on training and development) with fixed test; Fixed split; Not clear
	6	If cross-validation is used, is there a held-out test set?	Yes; No.
	7	Does the paper perform cross-dataset evaluation (training on one, testing on another)?	Yes; No.
Evaluation Measures	8	What is the reported evaluation measure for scoring?	QWK; MSE; Accuracy; F1-score; Pearson correlation; Recall; Other.
Performance Evaluation	9	What is the scoring focus?	Holistic; Single-trait; Multi-trait
	10	Do the authors perform statistical significance testing when comparing their method with existing approaches?	Yes; No.
	11	What is the result reporting granularity?	Average per-prompt (average overall traits for each prompt); Average per-trait (average overall prompts for each trait); Per-prompt and per-trait (values for each prompt and for each trait); Other.
	12	Are they evaluating on multiple datasets?	Yes; No.
	13	Is the selection of the baselines justified?	Yes: <i>The paper specify a justification for choosing their baselines</i> ; No: <i>The paper either lists the baselines or provide a description without a justification or don't even mention what baselines they will compare with and just report them in their results table</i> ; Not applicable: <i>The paper does not compare with other studies.</i>
Reproducibility	14	Is the code available?	Yes, there is a link in the paper, and the code is actually there; No, there is a link in the paper, but the code is not there or the link is broken; No, there is no link in the paper.
	15	Do they provide implementation details?	Yes; No.
	16	Do they use a hyperparameter tuning tool?	Yes and they report the tool/method (e.g. grid search, random search, Optuna, ...); Yes, but they don't report the tool/method; Not applicable.
	17	If hyperparameter tuning is performed, are they providing the hyperparameter list?	Yes (They provide the hyperparameter values they tried); No; Not applicable.
	18	If hyperparameter tuning is performed, are they mentioning the best hyperparameters?	Yes; No; Not applicable.
	19	For LLM-based methods, are they using closed-source models or open-source models?	Closed-source models; Open-source models; Both; Not applicable.
Performance Analysis	20	Do they include their model size?	Yes; No; Not applicable (e.g., traditional machine learning models such as Random Forest, and closed-source LLM-based models).
	21	If LLMs was used, do they report financial cost?	Yes; No; Not applicable (either do not use LLMs or rely exclusively on open-source LLMs).
	22	Do they report the efficiency of their model?	They report training time; They report inference time; They do not report; Not applicable (e.g. reproducibility paper or used closed source models only).
	23	What are the analyses performed?	Failure analysis (e.g., confusion matrix); Trait-based analysis; Prompt-types analysis; Ablation study; None; Other.

Table 1: Annotated questions used in our analysis.

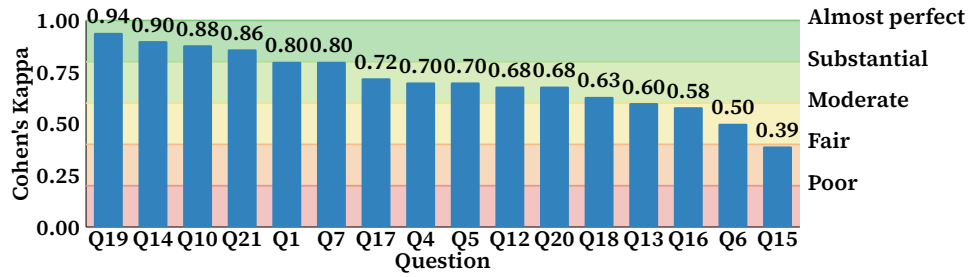


Figure 5: Cohen's Kappa for mutually exclusive questions.

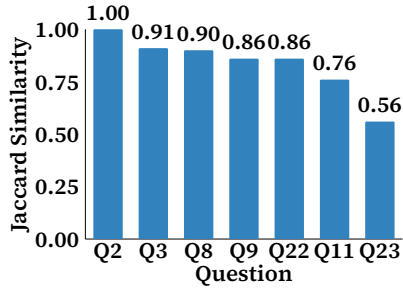


Figure 6: Jaccard similarity for multi-label questions.

on an unseen dataset)?

- A5** Did you discuss the score distribution of your dataset and how class imbalance (if any) was handled?
- A6** If your method is LLM-based, did you experiment with open-source models?
- A7** If your method is LLM-based and utilizes closed-source models, did you evaluate your method on a private dataset (to avoid data contamination)?
- A8** Did you explicitly describe your data-split strategy?
 - A8.1** Did you specify how many prompts are in the training, development, and test splits?
 - A8.2** If you used few-shot learning, did you specify how many and from which prompts the examples were selected?
 - A8.3** If your method is LLM-based, are you testing it on the entire test set?
- A9** Did you reserve a held-out test set for final evaluation (i.e., a completely unseen data used only for testing and not used during model optimization)?
 - A9.1** If a held-out test set was not possible, did you consider cross-dataset evaluation?

A10 Did you choose an evaluation measure that aligns with the system's objective?

A10.1 Is the selected measure appropriate for the characteristics of the dataset (e.g., QWK is not stable if applied on small datasets)?

A10.2 Did you report a measure other than QWK?

A10.3 When feasible, did you report inter-annotator agreement as an upper-bound reference for the evaluation measure?

A11 Did you choose strong and relevant baselines?

A11.1 Did you compare against multiple baselines rather than a single one?

A11.2 Did you include the current state-of-the-art model(s) among the baselines?

A11.3 Are all baselines evaluated under the same data splits, measures, and conditions as your proposed method?

C.2 Stage B: Model Development

B1 Did you perform hyperparameter tuning?

B1.1 Is it performed on the development set?

B1.2 Did you document the list or ranges of hyper-parameter values considered in the tuning?

B1.3 Did you provide the optimal hyper-parameter values used after tuning?

B2 Did you provide the model size (in terms of number of parameters)?

B3 Did you report important implementation details?

B3.1 Did you report general implementation details (e.g., hardware configuration, libraries, versions, etc.)?

- B3.2** Did you report method-specific implementation details (e.g., feature extraction procedures, embedding strategies, etc.)?
- B4** If the dataset has different prompt-specific or trait-specific score ranges, did you describe how scores were normalized or rescaled (if any)?
- B5** If your method utilizes engineered features, did you discuss how the features were normalized?
 - B5.1** Did you ensure that normalization was not applied at the dataset level (e.g., using global statistics across all prompts) rather than per-prompt, which may introduce data leakage across prompts?
 - B5.2** Did you specify how the features computed on the test set were normalized?
 - B5.3** If feature selection was done, did you clearly describe how it was performed?
- B6** If your method is LLM-based, did you provide all related details? (e.g. model version, checkpoint link if exists, prompts, and decoding parameters)

D Stage C: Performance Reporting and Analysis

- C1** Did you report per-prompt and per-trait results?
 - C1.1** If your method predicts one trait, did you report results per-prompt?
 - C1.2** If your method predicts multiple traits, did you report results per-prompt for each trait?
- C2** Did you perform a statistical significance test against the baselines?
- C3** Did you report the computational efficiency of your method?
 - C3.1** Did you report the average time taken to score one essay?
 - C3.2** Did you report the average time taken to train your model?
- C4** If your method uses proprietary models, did you report monetary cost?
- C5** Did you conduct an analysis of your method?

- C5.1** Did you conduct an ablation study to assess the contribution of individual components of your method?
- C5.2** Did you analyze which prompt types your method performs best and worst on, and discuss why?
- C5.3** Did you analyze failure cases to identify recurring error patterns?
- C5.4** Did you investigate whether your method struggles with specific score levels, and explain why?
- C6** Will you make the code publicly available to support reproducibility?