# AutoSCORE: Enhancing Automated Scoring with Multi-Agent Large Language Models via Structured Component Recognition

**Yun Wang**[1*], **Zhaojun Ding**[1*], **Xuansheng Wu**[1], **Siyue Sun**[2], **Ninghao Liu**[1†], **Xiaoming Zhai**[3†]
[1]School of Computing, University of Georgia
[2]Khoury College of Computer Sciences, Northeastern University
[3]AI4STEM Education Center, University of Georgia
{yun.wang1, zhaojun.ding, xuansheng.wu, ninghao.liu, xiaoming.zhai}@uga.edu, sun.siyue@northeastern.edu

## Abstract

Automated scoring plays a crucial role in education by reducing the reliance on human raters and offering scalable and immediate evaluation of student work. While large language models (LLMs) have shown strong potential in this task, their use as end-to-end raters faces challenges such as low accuracy, prompt sensitivity, limited interpretability, and rubric misalignment, which hinders practical implementation. To address the limitations, we propose AutoSCORE, a multi-agent LLM framework enhancing **auto**mated scoring via rubric-aligned **S**tructured **CO**mponent **RE**cognition. With two agents, AutoSCORE first extracts rubric-relevant components from student responses and encodes them into a structured representation (i.e., Scoring Rubric Component Extraction Agent), which is then used to assign final scores (i.e., Scoring Agent). This design ensures that model reasoning follows a human-like grading process, enhancing interpretability and robustness. We evaluate AutoSCORE on four benchmark datasets from the ASAP benchmark, using both proprietary and open-source LLMs (GPT-4o, LLaMA-3.1-8B, LLaMA-3.1-70B). Across diverse tasks and rubrics, AutoSCORE predominantly improves scoring accuracy, human-machine agreement (QWK, correlations), and reduces error metrics (MAE, RMSE) compared to single-agent baselines, with particularly strong benefits on complex, multi-dimensional rubrics, and especially large relative gains on smaller LLMs. These results demonstrate that structured component recognition combined with multi-agent design offers a scalable, reliable, and interpretable solution for automated scoring.

**Code** — https://github.com/AI4STEM-Education-Center/AutoSCORE

## Introduction

Grades are an integral component of educational systems, serving to quantify learning, measure achievement, and provide individualized feedback (Cain et al. 2022). However, assessing and assigning these grades to constructed response assessments often relies on manual scoring, which is time-consuming for educators (Aniche, Mulder, and Hermans 2021), failing to provide timely feedback for large classes or online settings. Moreover, human scoring does not always achieve ideal objectivity or inter-rater consistency (Hussein, Hassan, and Nassef 2019). These limitations highlight the need for scoring approaches that improve efficiency and scalability while enhancing consistency and fairness in assessment. Automated scoring offers a way forward by reducing grading time, lowering costs (Figueras et al. 2025), and providing more timely feedback (Nordquist 2007; Dikli and Bleyle 2014). Furthermore, with appropriate design and sufficient training data, automated scoring can yield higher accuracy and potentially reduce certain biases compared to human raters (Vo et al. 2023; Guo et al. 2025), thereby supporting fairness in assessment. As such, it is increasingly adopted as part of modern educational practice, with large-scale assessments such as the TOEFL and GRE incorporating automated scoring to evaluate written responses alongside human raters (Zhai and Pellegrino 2023).

While automated scoring has seen widespread uptake, challenges remain. Firstly, traditional deep learning scoring models require substantial amounts of labeled data to achieve high accuracy, which limits their scalability across tasks and contexts (Ridley et al. 2020). Moreover, they often operate as black-box systems, offering low transparency into the patterns and features used for scoring, which makes it difficult for educators to trust (Misgna et al. 2024). In addition, the feedback they provide is often generic to help students identify and address their most important areas for improvement, and it occasionally flags correct usage as errors or overlooks genuine mistakes (Zhai, Shi, and Nehm 2021). Addressing these challenges calls for more advanced approaches that can generalize across tasks, offer transparent decision-making, and deliver real-time feedback.

One promising direction comes from recent advances in large language models (LLMs), which offer strong language understanding and generative capabilities (Minaee et al. 2025) that can advance automated scoring, particularly for open-ended responses (Mansour et al. 2024; Seßler et al. 2025; Latif and Zhai 2024). These models can generalize across prompts, enabling scoring on diverse tasks without extensive retraining (Wu et al. 2023; Lee et al. 2024b). Beyond this, LLMs can explain their scoring decisions in natural language, improving transparency and interpretability for educators (Lee et al. 2024a; Chu et al. 2025). In addi-

[*]These authors contributed equally.
[†]Corresponding author.

tion, LLMs can generate more targeted, task-specific feedback and reduce misjudgments (Scarlatos et al. 2024), representing a notable improvement over traditional deep learning scoring systems. However, despite these advantages, the end-to-end fashion of current LLM-based scoring still leads to several limitations: (i) Even machine's assessments have the same scores with humans, their reasoning paths may differ and be untraceable, making audit difficult (Wu et al. 2025); (ii) Without an explicit step that checks each rubric criterion one by one, it is unclear whether LLM raters apply consistent standards across similar responses, consistency remains unverified (Shi et al. 2025); (iii) Using a prompt that concatenates the rubric, question, and student response into a single input may cause uneven coverage of rubric dimensions, with some criteria overlooked and others overemphasized (Zhu et al. 2025); (iv) Sensitivity to prompt or formatting variations, reducing reliability (Errica et al. 2025).

We argue that these limitations arise from the absence of an explicit step that first identifies evidence or components relevant to each rubric criterion before scoring. Without such a step, LLMs are more likely to deviate from criterion-grounded reasoning and produce inconsistent or incomplete evaluations. As a well-established tool in educational practice, rubrics enhance transparency by clarifying criteria (Brookhart 2018) and facilitate self-regulated learning, self-assessment, and self-efficacy (Panadero et al. 2019; Andrade 2019; Brookhart and Chen 2015). In performance assessments, rubrics can also improve human scoring reliability and inter-rater agreement (Jonsson and Svingby 2007; Reddy and Andrade 2010). Therefore, if LLMs are viewed as "rater," a more reasonable approach is not to assign scores all at once, but rather to align with the rubric, just as trained raters do: first identify evidence components corresponding to rubric items, then make judgments and assign scores, and finally produce traceable justifications linked to evidence.

Based on the above insights, we propose AutoSCORE, a rubric-guided multi-agent LLM scoring framework that explicitly identifies rubric-relevant content from student essays. This separated scoring process allows LLM raters to complete a distinct sub-task per inference, with an LLM specified for that sub-task serving as an agent. AutoSCORE comprises two agents: (i) a Scoring Rubric Component Extraction Agent that identifies structured components in student responses according to the rubric and produces rubric-aligned outputs, and (ii) a Scoring Agent that assigns scores based on these representations, responses, and the rubric. Optional Verification and Feedback Agents can be incorporated to denoise and align extraction results and to generate interpretable feedback, further enhancing quality control and transparency. We summarize our contributions as follows:

- Proposes a rubric-aligned structured scoring paradigm: first, scoring rubric component identification, then item-by-item scoring, forming a traceable evidence-judgment chain.
- Designs and implements a model-agnostic multi-agent framework that can leverage both proprietary models (such as GPT-4) and high-performance open-source models, facilitating localization and cost control.
- Evaluation is conducted on multiple datasets and question types, demonstrating improved consistency and robustness compared to single-step/single-model baselines.

## Related Work

Automated scoring broadly refers to the use of automated methods to evaluate constructed-response tasks in educational assessments (Zhai et al. 2020). Research on text-based automated scoring has two major strands: automated essay scoring (AES), which evaluates extended writing tasks such as essays (Shermis and Burstein 2013), and short answer scoring (SAS), which focuses on brief, content-specific responses (Mohler, Bunescu, and Mihalcea 2011). Earlier approaches in both strands predominantly relied on handcrafted features and statistical models, which achieved moderate agreement with human ratings but required prompt-specific engineering and offered limited transparency.

With the advent of deep learning and neural representation learning in NLP, automated scoring methods for both AES and SAS shifted from handcrafted feature engineering to end-to-end models that learn task-specific features directly from text (Taghipour and Ng 2016; Dong, Zhang, and Yang 2017). Early neural approaches used recurrent or convolutional architectures (Riordan et al. 2017; Taghipour and Ng 2016; Dong, Zhang, and Yang 2017), while more recent work leverages Transformer-based encoders, particularly BERT (Devlin et al. 2019; Latif and Zhai 2023), and pre-trained language models (Zhang and Litman 2019; Zhu, Wu, and Zhang 2022; Liu et al. 2023; Latif et al. 2024). These models generally achieve higher agreement and require less prompt-specific tuning, but often operate as black boxes and demand large labeled data (Mayfield and Black 2020; Dong, Zhang, and Yang 2017).

More recently, LLMs such as GPT-4 (Achiam et al. 2023) and LLaMA (Touvron et al. 2023) have demonstrated strong general-purpose language understanding and generation capabilities, enabling automated scoring in zero-shot or few-shot settings without extensive retraining (Mansour et al. 2024; Seßler et al. 2025; Wu et al. 2023). LLM-based scoring offers several advantages over traditional neural approaches: the ability to generate human-readable explanations of scores (Chu et al. 2025), provide richer and more targeted feedback (Scarlatos et al. 2024), and generalize across prompts and tasks (Lee et al. 2024b). However, existing LLM-based approaches still face critical challenges: the reasoning path behind a score is often opaque or untraceable (Wu et al. 2025); without explicit criterion-level reasoning, it is unclear whether LLM raters apply consistent standards across similar responses (Shi et al. 2025); criterion coverage can be unbalanced when scoring against rubrics (Zhu et al. 2025); and outputs remain sensitive to prompt and formatting variations (Errica et al. 2025).

Recent work has sought to address these limitations by incorporating rubrics into the scoring process, aiming to align evaluations more closely with human grading criteria and ensure more balanced coverage of rubric dimensions (Wu et al. 2025; Gunjal et al. 2025; Eltanbouly, Albatarni, and Elsayed 2025; Hashemi et al. 2024). However, most existing rubric-guided approaches integrate rubric criteria only

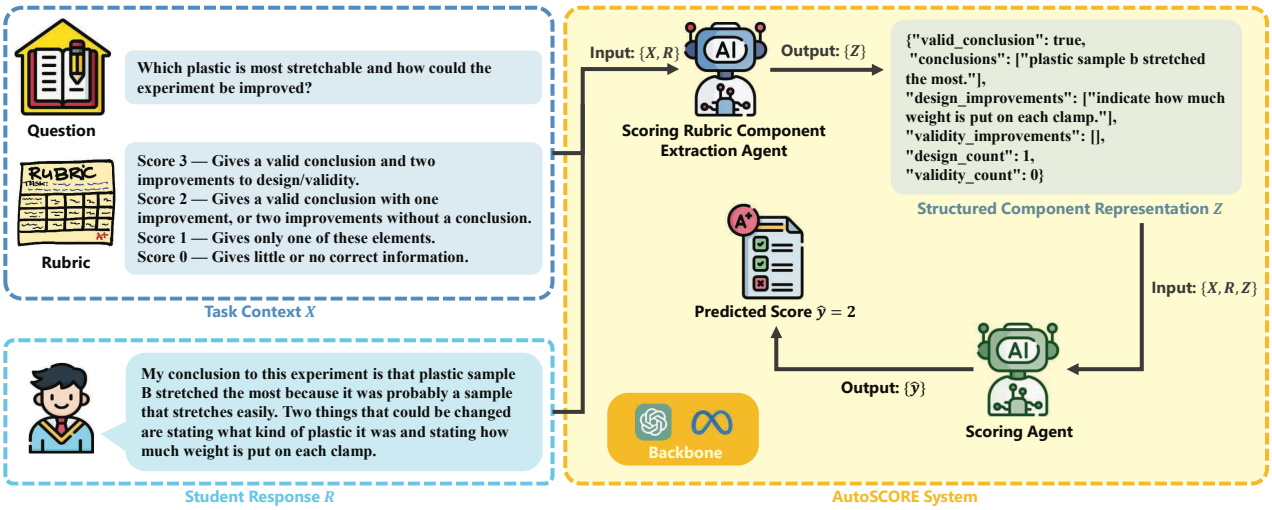


Figure 1: Overview of the proposed AutoSCORE multi-agent framework. The Scoring Rubric Component Extraction Agent identifies rubric-aligned components from the task context and student response, producing a structured representation $Z$. The Scoring Agent leveraged this representation together with the original inputs to assign the final score.

implicitly within the scoring prompt or model input, without explicitly identifying and structuring evidence for each criterion prior to scoring (Wu et al. 2025; Gunjal et al. 2025). As a result, they may still exhibit inconsistencies in applying standards across responses and offer limited auditability of the reasoning process. Motivated by these limitations, our work adopts a rubric-aligned scoring paradigm that first extracts criterion-specific evidence before scoring, enabling greater consistency, interpretability, and traceability, and further leverages a multi-agent design to enhance reliability.

## Methodology

We present AutoSCORE, a rubric-aligned, multi-agent LLM framework for automated scoring of constructed responses. It consists of two modules: Scoring Rubric Component Extraction Agent and Scoring Agent, which together promote rubric alignment and interpretability. The architecture is shown in Figure 1.

### Problem Formulation

We focus on the automated scoring task for assessing student-produced answers (Williamson, Xi, and Breyer 2012), such as short answers and essays. Each entry is represented as a triplet $(X, R, y)$, where $X$ denotes the task context, including the assessment question, reference materials (e.g., data tables or figures), and the scoring rubric; $R$ is the student's free-form written response in the text format; and $y$ is the score assigned by human raters. The score $y$ comes from a finite ordinal set defined by the corresponding rubric (e.g. $\{0, 1, 2, 3\}$). The objective is to design a system $f$ that, given $X$ and $R$, predicts a score

$$\hat{y} = f_\theta(X, R),$$

where $\hat{y}$ is expected to be as close as to the human assessment $y$. It is worth noting that, $f$ can be implemented with a single model or a framework with multiple LLMs.

### Structured Component Representation

To support rubric-grounded scoring, we propose an isolated process to check student responses according to the rubrics, rule-by-rule, before assigning the overall scores. Specifically, AutoSCORE first extracts rubric-relevant *components* from each student response $R$, and then encodes them into a structured format $Z$ to support the judgment of the overall score. Here, a component is defined as a piece of text from response $R$, reflecting a rule emphasized by the rubric $X$. Each component can be encoded into a different structured format, such as a Boolean flag, a count number, or the original text span itself. For instance, consider a science investigation task scored with the following 3-point rubric $X$:

> *Score 3: The response draws a valid conclusion and describes two ways to improve either the experimental design and/or the validity of the results.*
> *Score 2: A valid conclusion with one improvement, or two improvements without a valid conclusion.*
> *Score 1: Only one of these elements present.*
> *Score 0: Little or no correct information from the investigation.*

This rubric consistently refers to three key components: (1) the presence of a valid conclusion, (2) improvements to the experimental design, and (3) improvements to the validity of the results. AutoSCORE first identifies them from a student response, and then organizes these components into a structured-format representation $Z$. Here, a *representation* refers to a formatted record that explicitly restores rubric-relevant components extracted from the student responses, rather than a dense vector. Note that, if the format of $Z$ is loose or inconsistent, LLMs often fail to reuse this evidence to support their assessments reliably. To ensure consistency and human-readability, we adopt JSON as the encoding format for $Z$. For the rubric above, the components can be encoded in the JSON-style representation $Z$ as follows:

```
Z = {
  "valid_conclusion": true|false,
  "conclusions": [string, ...],
  "design_improvements": [string, ...],
  "validity_improvements": [string,
      ...],
  "design_count": integer,
  "validity_count": integer
}
```

In this example, we could observe that the three key components from the rubric are clearly defined as "valid_conclusion," "design_improvements," and "validity_improvements," respectively. Here, `true|false`, `string`, `integer` are data structures of the values that are used to store certain components for each keyword.

## Multi-Agent Framework

To separate evidence extraction from score assignment and improve scoring accuracy and interpretability, AutoSCORE designs a multi-agent LLM framework, in which one agent identifies rubric-relevant components and encodes them into the structured representation $Z$ from the student response, and another agent determines the final score based on $Z$, the original response, and the rubric. This step-by-step design constrains the reasoning path of scoring through explicit checkpoints, making the decision process more transparent and enabling error isolation compared to single-pass scoring (Trirat, Jeong, and Hwang 2024).

**Scoring Rubric Component Extraction Agent.** To explicitly capture the evidence required by the rubric, AutoSCORE employs a Scoring Rubric Component Extraction Agent. The motivation is that scoring directly from free-form responses often leads to missed or inconsistently treated criteria, lacking transparency. Instead, this agent identifies rubric-relevant components in the student response $R$, under the task context $X$, and encodes them into the structured representation $Z$. Formally, we have

$$Z = f_{\text{extract}}(X, R),$$

which produces a structured representation $Z$, constrained by a prompt that enforces organize $Z$ as a valid JSON format, ensuring consistency and machine-readability. This explicit separation of evidence identification from scoring makes reasoning more transparent and provides rubric-aligned inputs for downstream scoring.

**Scoring Agent.** The scoring agent assigns the final score $\hat{y}$ by leveraging the extracted representation $Z$, together with the task context $X$ and the original response $R$. By focusing on rubric-relevant evidence encoded in $Z$ and using $R$ for verification and error correction when inconsistencies arise, the agent ensures that predictions remain aligned with the rubric. Formally, the scoring agent is defined as

$$\hat{y} = f_{\text{scoring}}(Z, X, R),$$

where $f_{\text{scoring}}$ predicts the final score $\hat{y}$. In particular, $f_{\text{scoring}}$ is asked to align with rubric guidelines, resolve ambiguities in favor of rubric definitions, and require strict integer-only JSON output. These requests further enhance interpretability of the scoring process and support targeted debugging.

While our AutoSCORE framework can be extended with optional verification and feedback agents for additional quality control and interpretability, in this work, we focus on the two core agents described above to provide a clear evaluation of the core framework. We leave a detailed investigation of the optional modules for future work.

## Design Rationale: Reasoning as a Constrained Path

The two-stage design of AutoSCORE can be viewed through a graph reasoning perspective. If we conceptualize the scoring process as a path from the task description to the final score, a human rater's reasoning path passes through a sequence of intermediate checkpoints, i.e., identifying specific rubric-relevant components, before reaching at a decision. In this view, each checkpoint corresponds to a node in a reasoning graph, with multiple edges representing alternative reasoning steps. The Scoring Rubric Component Extraction Agent explicitly anchors these nodes by enforcing the extraction of rubric-grounded components into the structured representation $Z$. This constrains the LLM's reasoning path to pass through human-aligned intermediate states.

Under the assumption that aligning these intermediate nodes increases the overlap between the model's reasoning path and that of expert raters, this design improves scoring accuracy and interpretability to spurious correlations.

# Experiments and Analysis

We evaluate our proposed AutoSCORE framework on four datasets and three LLMs, covering both short- and long-text scoring tasks while mitigating the randomness and model-specific bias of single-model evaluations.

## Experimental Setup

**Datasets.** Our primary experiments are conducted on the Automated Student Assessment Prize (ASAP) dataset, which was developed to evaluate whether computer systems can reliably score written responses for educational assessment. The dataset consists of two components: ASAP-SAS (Short Answer Scoring) (Barbara et al. 2012) and ASAP-AES (Automated Essay Scoring) (Hamner et al. 2012). The ASAP-SAS dataset contains short student responses (fewer than 50 words) across multiple subjects, including **Science**, **Biology**, **English**, and other subjects, with rubric score ranges of 0 to 2 or 0 to 3. It covers two response types: source-dependent and non-source-dependent. To ensure diversity in subject matter, rubric range, and response type, we select three subsets: **Science** (rubric 0–3, source-dependent), **Biology** (rubric 0–3, non-source-dependent), and **English** (rubric 0–2, source-dependent). All selected SAS responses are from grade 10 students.

To further evaluate the performance of the framework on long-text scoring, we conducted experiments on ASAP-AES **EssaySet** #1, where the average essay length is approximately 350 words. The dataset consists of responses from grade 8 students, with scores ranging from 1 to 6, and includes persuasive, narrative, and expository essays.

In total, our evaluation covers 6,656 student responses: 4,871 from ASAP-SAS and 1,785 from ASAP-AES.

**Model Selection.** To evaluate AutoSCORE's generality, we experiment with both proprietary and open-source LLMs. The primary proprietary model is GPT-4o (Hurst et al. 2024), chosen for its state-of-the-art performance in diverse reasoning and language understanding tasks, and its stable API access. To further examine our framework's model-agnostic nature, we include high-performing open-source LLMs that can be deployed locally. Specifically, we use LLaMA-3.1-8B-Instruct (Dubey et al. 2024) and LLaMA-3.1-70B-Instruct, both competitive in instruction-following and reasoning benchmarks. Evaluating across models of different origins and architectures verifies that the multi-agent framework's benefits are not limited to a single LLM.

**Baselines.** We compare the proposed multi-agent scoring framework with a single-agent baseline for each selected LLM. In the baseline setting, the model is prompted once to directly generate a score for a given response according to the official scoring rubric. All other experimental conditions are identical between the two settings.

**Evaluation Metrics.** The performance of our AutoSCORE framework was evaluated using six complementary metrics. Accuracy measures exact agreement with human raters, while Quadratic Weighted Kappa (QWK) captures weighted agreement by rewarding closer predictions and penalizing larger discrepancies. The Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) quantify absolute and squared errors, and Pearson and Spearman correlations reflect linear and rank-based consistency with human scores.

**Implementation.** All experiments were conducted on a workstation equipped with 3 × NVIDIA RTX A6000 GPUs and Intel Xeon Silver 4214R CPU @ 2.40GHz.

## Main Scoring Performance Comparison

Table 1 summarizes the main results across four datasets and three LLMs. Overall, AutoSCORE achieves consistent improvements in most dataset–metric combinations compared to single-agent baselines. For instance, on Science subset with GPT-4o, Accuracy rises from 0.588 to 0.632 (+7.5%), while on English subset, QWK improves from 0.540 to 0.629 (+16.5%). On the long-text task Essay Set, QWK increases from 0.251 to 0.344 (+37.1%) with Accuracy also rising by 26.9%. Although a few settings show marginal declines (e.g., QWK for the LLaMA-3.1-8B model on the Biology subset), the overall trend across Accuracy, QWK, MAE/RMSE, and correlations indicates that rubric-aligned multi-agent scoring provides consistent benefits, particularly for tasks requiring complex rubric coverage.

**Task-Level Analysis.** Performance improvements vary with the complexity of both the rubric and the underlying task. On Biology subset, where scoring essentially reduces to counting key elements, the baseline already aligns well with the rubric. Here AutoSCORE shows limited benefits: GPT-4o improves slightly (QWK +9.1%), while the LLaMA-3.1-8B model even declines. This may be due to (i) a ceiling effect, where the task is too simple to benefit from extra structure, and (ii) error propagation, where noise in component extraction can outweigh potential benefits. In contrast, the English subset and Essay Set involve more complex responses and multi-dimensional rubrics. These settings require accurate extraction of rubric-relevant components to support scoring, where AutoSCORE yields the largest and most consistent improvements. For example, on Essay Set, QWK increases by +37.1%. These results confirm that the framework is particularly valuable for tasks with complex rubrics or longer responses, where accurate component recognition becomes critical for reliable scoring.

**Model-Level Analysis.** We examine how model capacity affects the relative benefits of AutoSCORE. The relative gains are larger for smaller models. On Science subset, the LLaMA-3.1-8B model improves QWK from 0.150 to 0.261 (+74.0%), whereas GPT-4o achieves a much smaller relative gain (+2.3%) under the same task. Similar patterns are observed on English subset, where the relative gain is +26.3% for the LLaMA-3.1-8B model compared to +16.5% for GPT-4o, and on Essay Set (+43.0% for LLaMA-3.1-8B model and +37.1% for GPT-4o). This indicates that AutoSCORE is especially effective when the backbone model has limited capacity: by decoupling component recognition from final scoring, AutoSCORE compensates for weaker reasoning ability. Practically, this implies that institutions with limited computational resources can deploy smaller models and still achieve substantial accuracy gains.

## Ablation and Robustness

The single-agent scoring baseline can be viewed as an ablation that removes the Scoring Rubric Component Extraction Agent, collapsing the framework into direct end-to-end scoring. The consistent performance gap between AutoSCORE and this baseline across datasets quantifies the contribution of structured component recognition. Moreover, the improvements appear not only in correlation-based metrics (QWK, Pearson, Spearman) but also in error-based metrics (MAE, RMSE), suggesting robustness across evaluations.

## Validation of Component Recognition Reliability

To assess the reliability of the component recognition agent, we conducted a double-annotation study on two ASAP-SAS subsets: Science subset (multi-component rubric) and Biology subset (key-element counting). For each dataset, we randomly sampled 20% of responses (n=258 for Science subset and n=370 for Biology subset). Two trained annotators independently identified rubric-relevant components, with adjudication yielding a gold reference.

On Science subset, the agent achieved strong reliability on the binary valid conclusion label (accuracy 0.899, $F_1$ 0.918, Cohen's $\kappa$ 0.788). For component counts, the agent reached MAE 0.295, RMSE 0.564, Pearson $r$ 0.688 for design_count, and MAE 0.116, RMSE 0.374, Pearson $r$ 0.777 for validity_count. The exact-match rates were 0.717 and 0.895, respectively, confirming moderate-to-high fidelity in recovering rubric-relevant counts. While minor deviations remained in exact matches, the low error and moderate-to-high correlation suggest that the agent provides dependable component counts for subsequent scoring.

On Biology subset, where the rubric reduces to enumerating key elements, predictions were even more stable. The agent achieved an exact-match rate of 0.859, with MAE 0.157, RMSE 0.441, and a Pearson correlation of 0.893

| Datasets | Models | QWK↑ | Accuracy↑ | Pearson↑ | Spearman↑ | MAE↓ | RMSE↓ |
|---|---|---|---|---|---|---|---|
| Science subset (3 Components) | GPT-4o | 0.701 | 0.588 | 0.707 | 0.696 | 0.451 | 0.733 |
| | + AutoSCORE | **0.717** | **0.632** | **0.728** | **0.718** | **0.418** | **0.726** |
| | Δ (%) | +2.28% | +7.48% | +2.97% | +3.16% | -7.31% | -0.96% |
| | LLaMA-3.1-8B-Instruct | 0.150 | 0.293 | 0.339 | 0.222 | 0.879 | 1.105 |
| | + AutoSCORE | **0.261** | **0.350** | **0.370** | **0.269** | **0.763** | **0.994** |
| | Δ (%) | +74.00% | +19.45% | +9.15% | +21.17% | -13.20% | -10.05% |
| | LLaMA-3.1-70B-Instruct | **0.533** | 0.426 | **0.617** | **0.542** | 0.663 | 0.925 |
| | + AutoSCORE | 0.504 | **0.462** | 0.543 | 0.431 | **0.588** | **0.837** |
| | Δ (%) | -5.44% | +8.45% | -11.99% | -20.48% | -11.31% | -9.51% |
| Biology subset (1 Component) | GPT-4o | 0.681 | **0.819** | 0.695 | 0.687 | **0.188** | **0.450** |
| | + AutoSCORE | **0.743** | 0.806 | **0.750** | **0.695** | 0.198 | 0.453 |
| | Δ (%) | +9.10% | -1.59% | +7.91% | +1.16% | +5.32% | +0.67% |
| | LLaMA-3.1-8B-Instruct | **0.087** | 0.182 | **0.472** | **-0.230** | **0.826** | **0.918** |
| | + AutoSCORE | 0.063 | **0.184** | 0.422 | -0.255 | 0.827 | 0.921 |
| | Δ (%) | -27.59% | +1.10% | -10.59% | -10.87% | +0.12% | +0.33% |
| | LLaMA-3.1-70B-Instruct | **0.704** | **0.816** | **0.709** | 0.517 | **0.188** | **0.445** |
| | + AutoSCORE | 0.660 | 0.739 | 0.699 | **0.570** | 0.278 | 0.560 |
| | Δ (%) | -6.25% | -9.44% | -1.41% | +10.39% | +47.87% | +25.84% |
| English subset (4 Components) | GPT-4o | 0.540 | 0.548 | 0.603 | 0.501 | 0.468 | 0.708 |
| | + AutoSCORE | **0.629** | **0.604** | **0.653** | **0.553** | **0.398** | **0.633** |
| | Δ (%) | +16.48% | +10.22% | +8.29% | +10.38% | -14.96% | -10.59% |
| | LLaMA-3.1-8B-Instruct | 0.354 | 0.430 | 0.482 | 0.408 | 0.628 | 0.863 |
| | + AutoSCORE | **0.447** | **0.492** | **0.505** | **0.449** | **0.547** | **0.790** |
| | Δ (%) | +26.27% | +14.42% | +4.77% | +10.05% | -12.9% | -8.46% |
| | LLaMA-3.1-70B-Instruct | 0.422 | 0.463 | 0.564 | 0.439 | 0.568 | 0.794 |
| | + AutoSCORE | **0.490** | **0.532** | **0.572** | **0.470** | **0.486** | **0.721** |
| | Δ (%) | +16.11% | +14.90% | +1.42% | +7.06% | -14.44% | -9.19% |
| Essay Set (5 Components) | GPT-4o | 0.251 | **0.280** | 0.412 | 0.343 | 1.076 | 1.396 |
| | + AutoSCORE | **0.344** | 0.269 | **0.499** | **0.453** | **1.035** | **1.327** |
| | Δ (%) | +37.05% | -3.93% | +21.12% | +32.07% | -3.81% | -4.94% |
| | LLaMA-3.1-8B-Instruct | 0.135 | 0.128 | 0.361 | **0.301** | 1.364 | 1.596 |
| | + AutoSCORE | **0.193** | **0.165** | **0.436** | 0.273 | **1.222** | **1.451** |
| | Δ (%) | +42.96% | +28.91% | +20.78% | -9.30% | -10.41% | -9.09% |
| | LLaMA-3.1-70B-Instruct | 0.428 | 0.340 | 0.599 | 0.525 | 0.862 | 1.143 |
| | + AutoSCORE | **0.575** | **0.488** | **0.649** | **0.549** | **0.612** | **0.914** |
| | Δ (%) | +34.35% | +43.53% | +8.35% | +4.57% | -29.00% | -20.03% |

Table 1: Main scoring performance comparison across datasets, models, and evaluation metrics, highlighting the impact of AutoSCORE over baseline models. Bold numbers denote the best performance within each dataset–metric pair.

against human annotations. The higher correlation and lower error in this setting reflect the simpler rubric structure, confirming that the agent can reproduce key-element counts.

Taken together, these findings demonstrate that the Component Recognition Agent reliably extracts rubric-grounded components across tasks of varying complexity. This provides a solid foundation for the AutoSCORE framework: accurate intermediate representations can be trusted as inputs to the downstream scoring stage, ensuring that observed improvements in overall scoring performance are not undermined by unreliable component recognition.

## Averaged Inference Time and QWK Tradeoff

To better analyze the changes in both efficiency and performance after applying our proposed AutoSCORE framework, we conducted experiments on the English subset as a representative case. Specifically, we evaluated the tradeoff between QWK, a widely used performance metric in automated scoring, and the averaged inference time per instance, which reflects the efficiency of LLM-based scoring systems. Three models were tested: GPT-4o, LLaMA-3.1-70B-Instruct, and LLaMA-3.1-8B-Instruct.

As shown in Figure 2, the results consistently demonstrate

| *Human Score = 1, AutoSCORE = 1, Baseline = 0* | |
|---|---|
| **Assessment Question** | How does the author organize the article? Support your response with details from the article. |
| **Student Response** | The author organizes the article in parts. He starts with an introduction to pull the reader in and then quickly changes the tone to show that he is still taking the article seriously. |
| **Rubric Excerpt** | **1 pt (Partially Proficient):** Fulfills some requirements, but may be general or simplistic.<br>**0 pt (Not Proficient):** Inaccurate, incomplete, or missing information. |
| **Selected Extracted Components** | **Organization Method:** The author organizes the article in parts.<br>**Supporting Details:** (i) He starts with an introduction to pull the reader in. (ii) Then quickly changes the tone to show seriousness. |

Table 2: Case study example from English subset.

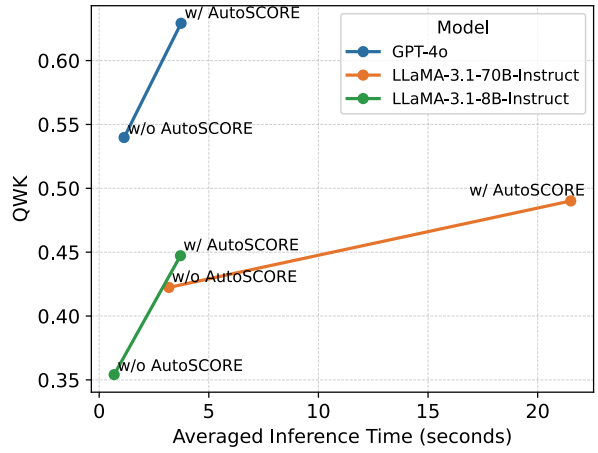


Figure 2: QWK compared with averaged inference time across different LLMs with and without AutoSCORE.

that incorporating AutoSCORE leads to higher QWK values at the cost of increased inference time across all three models. This highlights a clear efficiency–performance trade-off: while AutoSCORE requires more time per inference, it yields more reliable scoring outcomes. Importantly, AutoSCORE provides a practical pathway to improving performance without necessitating access to substantially larger open-source models (which demand more computational resources) or more expensive proprietary models.

## Case Study

To illustrate why our proposed AutoSCORE framework achieves more reliable scoring than directly applying an LLM to assign scores, we conduct a case study on the English subset. In this dataset, students were asked to read an article about space junk and respond to the assessment question about "How does the author organize the article?"

As shown in Table 2, the rubric differentiates between responses that merely provide vague or incomplete statements and those that articulate how the article is organized with at least some supporting detail. In this example, the student identified both the organizational structure and supporting evidence. AutoSCORE explicitly extracted these rubric-relevant components, guiding the LLM to follow a reasoning path aligned with human raters and yielding the correct score of 1. In contrast, when GPT-4o scored the response directly, it overlooked key details and produced a score of 0. This comparison highlights how AutoSCORE constrains the model's reasoning through rubric-grounded intermediate states, thereby improving scoring accuracy.

## Conclusion and Discussion

In this work, we introduced AutoSCORE, a rubric-aligned multi-agent LLM framework for automated scoring. By explicitly separating component extraction from scoring, AutoSCORE constrains the reasoning trajectory of LLMs to pass through human-aligned intermediate states, thereby addressing key limitations of end-to-end scoring such as rubric misalignment and lack of interpretability. Our experiments on four benchmark datasets (three ASAP-SAS subsets and one ASAP-AES set) and three LLMs demonstrate that AutoSCORE predominantly outperforms single-agent baselines across Accuracy, QWK, correlation metrics, and error reduction. Notably, the framework shows the largest gains on tasks with multi-dimensional rubrics and long-form responses, confirming that explicit component recognition is crucial for reliable scoring in complex educational assessments. We also observe that AutoSCORE yields especially large relative improvements on smaller LLMs, highlighting its practicality for cost-sensitive educational applications.

While AutoSCORE demonstrates consistent improvements across datasets and models, several limitations remain. First, our experiments primarily focus on non-reasoning LLMs, which may differ from reasoning-oriented models (e.g., ChatGPT-o1) and could align differently with rubric-guided scoring, and evaluating AutoSCORE on such models warrants further investigation. Second, although our experiments provide theoretical evidence of effectiveness, AutoSCORE has not yet been deployed in a real classroom or large-scale student assessment settings. Future validation in authentic classrooms will be crucial to understanding AutoSCORE's potential impact on student learning, teacher workload, and assessment reliability. Moreover, our work is limited to text-based assessments, whereas modern education increasingly involve multimodal inputs like images and audio, and extending AutoSCORE to handle such modalities remains an important direction for future research. Currently, rubric component templates are manually derived for precise alignment, and future work could automate this step using an auxiliary agent.

## Acknowledgments

This work is supported by the National Science Foundation (NSF) under Grant Nos. 2507128 and 2101104, and by the Institute of Education Sciences (IES) under Grant No. R305C240010. The views and conclusions expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

## References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Andrade, H. L. 2019. A critical review of research on student self-assessment. In *Frontiers in education*, volume 4, 87. Frontiers Media SA.

Aniche, M.; Mulder, F.; and Hermans, F. 2021. Grading 600+ Students: A Case Study on Peer and Self Grading. In *2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*, 211–220.

Barbara; Hamner, B.; Morgan, J.; lynnvandev; and Shermis, M. 2012. The Hewlett Foundation: Short Answer Scoring. https://kaggle.com/competitions/asap-sas. Accessed: 2025-06-10.

Brookhart, S. M. 2018. Appropriate criteria: Key to effective rubrics. In *Frontiers in education*, volume 3, 22. Frontiers Media SA.

Brookhart, S. M.; and Chen, F. 2015. The quality and effectiveness of descriptive rubrics. *Educational Review*, 67(3): 343–368.

Cain, J.; Medina, M.; Romanelli, F.; and Persky, A. 2022. Deficiencies of traditional grading systems and recommendations for the future. *American Journal of Pharmaceutical Education*, 86(7): 8850.

Chu, S.; Kim, J.; Wong, B.; and Yi, M. 2025. Rationale Behind Essay Scores: Enhancing S-LLM's Multi-Trait Essay Scoring with Rationale Generated by LLMs. arXiv:2410.14202.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 4171–4186.

Dikli, S.; and Bleyle, S. 2014. Automated essay scoring feedback for second language writers: How does it compare to instructor feedback? *Assessing writing*, 22: 1–17.

Dong, F.; Zhang, Y.; and Yang, J. 2017. Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the 21st conference on computational natural language learning (CoNLL 2017)*, 153–162.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. 2024. The llama 3 herd of models. *arXiv e-prints*, arXiv–2407.

Eltanbouly, S.; Albatarni, S.; and Elsayed, T. 2025. TRATES: Trait-Specific Rubric-Assisted Cross-Prompt Essay Scoring. *arXiv preprint arXiv:2505.14577*.

Errica, F.; Siracusano, G.; Sanvito, D.; and Bifulco, R. 2025. What Did I Do Wrong? Quantifying LLMs' Sensitivity and Consistency to Prompt Engineering. arXiv:2406.12334.

Figueras, C.; Farazouli, A.; Cerratto Pargman, T.; McGrath, C.; and Rossitto, C. 2025. Promises and breakages of automated grading systems: a qualitative study in computer science education. *Education Inquiry*, 1–22.

Gunjal, A.; Wang, A.; Lau, E.; Nath, V.; Liu, B.; and Hendryx, S. 2025. Rubrics as Rewards: Reinforcement Learning Beyond Verifiable Domains. arXiv:2507.17746.

Guo, S.; Wang, Y.; Yu, J.; Wu, X.; Ayik, B.; Watts, F. M.; Latif, E.; Liu, N.; Liu, L.; and Zhai, X. 2025. Artificial Intelligence Bias on English Language Learners in Automatic Scoring. In *International Conference on Artificial Intelligence in Education*, 268–275. Springer.

Hamner, B.; Morgan, J.; lynnvandev; Shermis, M.; and Ark, T. V. 2012. The Hewlett Foundation: Automated Essay Scoring. https://kaggle.com/competitions/asap-aes. Accessed: 2025-06-10.

Hashemi, H.; Eisner, J.; Rosset, C.; Van Durme, B.; and Kedzie, C. 2024. LLM-rubric: A multidimensional, calibrated approach to automated evaluation of natural language texts. *arXiv preprint arXiv:2501.00274*.

Hurst, A.; Lerer, A.; Goucher, A. P.; Perelman, A.; Ramesh, A.; Clark, A.; Ostrow, A.; Welihinda, A.; Hayes, A.; Radford, A.; et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Hussein, M. A.; Hassan, H.; and Nassef, M. 2019. Automated language essay scoring systems: A literature review. *PeerJ Computer Science*, 5: e208.

Jonsson, A.; and Svingby, G. 2007. The use of scoring rubrics: Reliability, validity and educational consequences. *Educational research review*, 2(2): 130–144.

Latif, E.; Lee, G.-G.; Neumann, K.; Kastorff, T.; and Zhai, X. 2024. G-sciedbert: A contextualized llm for science assessment tasks in german. *arXiv preprint arXiv:2402.06584*.

Latif, E.; and Zhai, X. 2023. Automatic Scoring of Students' Science Writing Using Hybrid Neural Network. *arXiv preprint arXiv:2312.03752*.

Latif, E.; and Zhai, X. 2024. Fine-tuning ChatGPT for automatic scoring. *Computers and Education: Artificial Intelligence*, 6: 100210.

Lee, G.-G.; Latif, E.; Wu, X.; Liu, N.; and Zhai, X. 2024a. Applying large language models and chain-of-thought for automatic scoring. *Computers and Education: Artificial Intelligence*, 6: 100213.

Lee, S.; Cai, Y.; Meng, D.; Wang, Z.; and Wu, Y. 2024b. Unleashing Large Language Models' Proficiency in Zero-shot Essay Scoring. arXiv:2404.04941.

Liu, Z.; He, X.; Liu, L.; Liu, T.; and Zhai, X. 2023. Context matters: A strategy to pre-train language model for science education. In *International conference on artificial intelligence in education*, 666–674. Springer.

Mansour, W.; Albatarni, S.; Eltanbouly, S.; and Elsayed, T. 2024. Can large language models automatically score proficiency of written essays?. arXiv. org.

Mayfield, E.; and Black, A. W. 2020. Should you fine-tune BERT for automated essay scoring? In *Proceedings of the fifteenth workshop on innovative use of NLP for building educational applications*, 151–162.

Minaee, S.; Mikolov, T.; Nikzad, N.; Chenaghlu, M.; Socher, R.; Amatriain, X.; and Gao, J. 2025. Large Language Models: A Survey. arXiv:2402.06196.

Misgna, H.; On, B.-W.; Lee, I.; and Choi, G. S. 2024. A survey on deep learning-based automated essay scoring and feedback generation. *Artificial Intelligence Review*, 58(2): 36.

Mohler, M.; Bunescu, R.; and Mihalcea, R. 2011. Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, 752–762.

Nordquist, P. 2007. Providing accurate and timely feedback by automatically grading student programming labs. *Journal of Computing Sciences in Colleges*, 23(2): 16–23.

Panadero, E.; Broadbent, J.; Boud, D.; and Lodge, J. M. 2019. Using formative assessment to influence self-and co-regulated learning: the role of evaluative judgement. *European Journal of Psychology of Education*, 34(3): 535–557.

Reddy, Y. M.; and Andrade, H. 2010. A review of rubric use in higher education. *Assessment & evaluation in higher education*, 35(4): 435–448.

Ridley, R.; He, L.; Dai, X.; Huang, S.; and Chen, J. 2020. Prompt Agnostic Essay Scorer: A Domain Generalization Approach to Cross-prompt Automated Essay Scoring. arXiv:2008.01441.

Riordan, B.; Horbach, A.; Cahill, A.; Zesch, T.; and Lee, C. 2017. Investigating neural architectures for short answer scoring. In *Proceedings of the 12th workshop on innovative use of NLP for building educational applications*, 159–168.

Scarlatos, A.; Smith, D.; Woodhead, S.; and Lan, A. 2024. *Improving the Validity of Automatically Generated Feedback via Reinforcement Learning*, 280–294. Springer Nature Switzerland. ISBN 9783031643026.

Seßler, K.; Fürstenberg, M.; Bühler, B.; and Kasneci, E. 2025. Can AI grade your essays? A comparative analysis of large language models and teacher ratings in multidimensional essay scoring. In *LAK 25*, 462–472.

Shermis, M. D.; and Burstein, J. 2013. Handbook of automated essay evaluation. *NY: Routledge*.

Shi, L.; Ma, C.; Liang, W.; Diao, X.; Ma, W.; and Vosoughi, S. 2025. Judging the Judges: A Systematic Study of Position Bias in LLM-as-a-Judge. arXiv:2406.07791.

Taghipour, K.; and Ng, H. T. 2016. A neural approach to automated essay scoring. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, 1882–1891.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Trirat, P.; Jeong, W.; and Hwang, S. J. 2024. Automl-agent: A multi-agent llm framework for full-pipeline automl. *arXiv preprint arXiv:2410.02958*.

Vo, Y.; Rickels, H.; Welch, C.; and Dunbar, S. 2023. Human scoring versus automated scoring for English learners in a statewide evidence-based writing assessment. *Assessing Writing*, 56: 100719.

Williamson, D. M.; Xi, X.; and Breyer, F. J. 2012. A framework for evaluation and use of automated scoring. *Educational measurement: issues and practice*, 31(1): 2–13.

Wu, X.; He, X.; Liu, T.; Liu, N.; and Zhai, X. 2023. Matching exemplar as next sentence prediction (mensp): Zero-shot prompt learning for automatic scoring in science education. In *International conference on artificial intelligence in education*, 401–413. Springer.

Wu, X.; Saraf, P. P.; Lee, G.; Latif, E.; Liu, N.; and Zhai, X. 2025. Unveiling scoring processes: Dissecting the differences between llms and human graders in automatic scoring. *Technology, Knowledge and Learning*, 1–16.

Zhai, X.; and Pellegrino, J. W. 2023. Large-scale assessment in science education. In *Handbook of research on science education*, 1045–1097. Routledge.

Zhai, X.; Shi, L.; and Nehm, R. H. 2021. A meta-analysis of machine learning-based science assessments: Factors impacting machine-human score agreements. *Journal of Science Education and Technology*, 30(3): 361–379.

Zhai, X.; Yin, Y.; Pellegrino, J. W.; Haudek, K. C.; and Shi, L. 2020. Applying machine learning in science assessment: a systematic review. *Studies in Science Education*, 56(1): 111–151.

Zhang, H.; and Litman, D. 2019. Co-attention based neural network for source-dependent essay scoring. *arXiv preprint arXiv:1908.01993*.

Zhu, X.; Wu, H.; and Zhang, L. 2022. Automatic short-answer grading via BERT-based deep neural networks. *IEEE Transactions on Learning Technologies*, 15(3): 364–375.

Zhu, Y.; Li, R.; Wang, D.; Haehn, D.; and Liang, X. 2025. Focus Directions Make Your Language Models Pay More Attention to Relevant Contexts. arXiv:2503.23306.