

CoGrader: Transforming Instructors' Assessment of Project Reports through Collaborative LLM Integration

Zixin Chen

zchendf@connect.ust.hk

The Hong Kong University of Science
and Technology
Hong Kong, China

Jiachen Wang

wangjiachen@zju.edu.cn

Zhejiang University
Hangzhou, Zhejiang, China

Yumeng Li

liyumengcafa@gmail.com

The University of Hong Kong
Hong Kong, China

Haobo Li

hliem@connect.ust.hk

The Hong Kong University of Science
and Technology
Hong Kong, China

Chuhan Shi

chuhanshi@seu.edu.cn

Southeast University
Nanjing, Jiangsu, China

Rong Zhang

rzhangab@connect.ust.hk

The Hong Kong University of Science
and Technology
Hong Kong, China

Huamin Qu

huamin@cse.ust.hk

The Hong Kong University of Science
and Technology
Hong Kong, China

Abstract

Grading project reports is increasingly significant in today's educational landscape, where they serve as key assessments of students' comprehensive problem-solving abilities. However, it remains challenging due to the multifaceted evaluation criteria involved, such as creativity and peer-comparative achievement. Meanwhile, instructors often struggle to maintain fairness throughout the time-consuming grading process. Recent advances in AI, particularly large language models, have demonstrated potential for automating simpler grading tasks, such as assessing quizzes or basic writing quality. However, these tools often fall short when it comes to complex metrics, like design innovation and the practical application of knowledge, that require an instructor's educational insights and contextual understanding of the class. To address this challenge, we conducted a formative study with six instructors and developed CoGrader, which introduces a novel grading workflow combining human-LLM collaborative metrics design, benchmarking, and AI-assisted feedback. CoGrader was found effective in improving grading efficiency and consistency while providing reliable peer-comparative feedback to students. We also discuss design insights and ethical considerations for the development of human-AI collaborative grading systems.

CCS Concepts

- Human-centered computing → Visualization systems and tools;
- Applied computing → E-learning;
- Computing methodologies → Artificial intelligence.

Keywords

project report grading, large language models, human-AI collaboration, llm for education

ACM Reference Format:

Zixin Chen, Jiachen Wang, Yumeng Li, Haobo Li, Chuhan Shi, Rong Zhang, and Huamin Qu. 2025. CoGrader: Transforming Instructors' Assessment of Project Reports through Collaborative LLM Integration. In *The 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25), September 28–October 1, 2025, Busan, Republic of Korea*. ACM, New York, NY, USA, 18 pages. <https://doi.org/10.1145/3746059.3747670>

1 Introduction

With the revolutionary development of generative artificial intelligence (GAI), not only is our day-to-day life undergoing significant changes, but the skills that are valued and emphasized in education are also shifting in a foreseeable way [8, 26]. Recent educational research argues that skills such as critical thinking, creativity, and problem-solving are becoming increasingly important, as they enable individuals to move beyond simple information recall and engage in complex, valued thinking processes [2, 47, 84]. Among various learning activities, project-based learning (PBL) stands out as particularly effective in cultivating these significant thinking skills [3, 11, 49]. By immersing students in real-world challenges, PBL fosters the application of theoretical knowledge, the development of original ideas, and the synthesis of interdisciplinary insights [18, 72].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

UIST '25, September 28–October 1, 2025, Busan, Republic of Korea

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 979-8-4007-2037-6/2025/09
<https://doi.org/10.1145/3746059.3747670>

Despite its educational value, assessing project reports remains one of the most challenging and time-consuming tasks for instructors [4, 82]. Unlike standardized learning activities such as quizzes, project reports require evaluation across a range of complex, multi-faceted, and often customized metrics that rely on an instructor's educational insights, including assessments of originality, practical application of knowledge, and the level of effort and achievement compared to peers [10, 52]. The complexity of these metrics, combined with the need for subjective judgment, necessitates nuanced evaluation and deep domain expertise, making the grading process particularly demanding. Additionally, instructors face the challenge of maintaining consistency and fairness throughout the grading process, especially over extended periods where there is a risk of grading standards drifting [57, 68, 85]. In many cases, instructors must revisit and adjust earlier scores as they develop a clearer sense of overall class performance, resulting in repetitive work and potential inefficiencies [88].

Recent advances in AI, particularly large language models (LLMs) such as ChatGPT, have demonstrated significant promise in automating various educational tasks for instructors [1, 54, 81]. For example, LLMs have successfully been employed to generate multiple-choice questions, provide personalized exercises and feedback to students, and even support the training of novice teachers [53, 58, 64, 78, 83]. Specifically in the context of grading, existing studies highlight LLMs' strengths in efficiently evaluating standardized components such as structural clarity, coherence, and basic language correctness [38, 77]. However, LLMs often struggle when grading complex or subjective criteria, such as assessing originality, evaluating creative problem-solving approaches, or accurately interpreting context-specific insights in project reports. Common issues include misinterpreting nuanced student arguments, providing overly generic feedback, or incorrectly assessing the significance of unique ideas or methods presented by students [12, 45].

Conversely, experienced instructors excel in making nuanced judgments informed by their educational insights and contextual knowledge of the class, yet they can experience challenges in maintaining consistent standards or managing potential biases over extensive grading tasks. Thus, a collaborative approach where human instructors retain primary responsibility for assessment, supported by AI-generated suggestions and preliminary evaluations, provides an effective solution. Human oversight ensures accuracy, fairness, and customized feedback specifically aligned with the unique qualities and learning objectives of each student's project report [7, 23, 55].

As a result, while LLMs have made significant strides in automating educational tasks, its integration into the complex workflow of grading project reports remains largely unexplored. To address this gap, we propose a human-AI collaborative workflow that blends LLMs' computational strengths with human instructors' nuanced expertise. The workflow begins with an LLM-powered project requirement analysis and collaborative metric design, allowing instructors to select and refine relevant report assessment metrics. Next, LLM automatically grades the reports, providing reference scores and comments for each user-selected metric. Instructors then review these initial assessments, using LLM-generated insights to guide the selection of benchmark reports. Once benchmarks are established, LLM re-evaluates each report, offering comments that

compare the performance on each metric against these benchmarks. This not only provides students with individualized feedback but also allows them to learn from peer comparisons by understanding how their work measures up to the benchmark reports. Instructors can review and modify LLM's comments for each criterion, while also freely adding their own annotations and comments on other parts of the report as needed. Finally, LLM consolidates all grades and comments, generating personalized, comprehensive feedback for each report.

Throughout the process, instructors retain full authority over judgments, while the LLM accelerates the grading workflow by providing initial evaluations and speeding up benchmark-driven regrading, all while ensuring detailed feedback for students. This collaborative approach combines AI efficiency with human expertise, maintaining fairness and accuracy with the instructors' judgment at the core. As a result, students receive more timely, peer-comparison-based, and personalized feedback, enabling them to reflect on their work and learn from their peers, fostering growth and deeper understanding in project-based learning. To summarize, our paper makes the following three contributions:

- A formative study that identifies instructors' challenges in grading project reports and explores how integrating LLMs like ChatGPT can address these challenges while keeping instructors in control, laying the foundation for an effective human-AI co-grading framework.
- The design and implementation of CoGrader, a novel human-AI collaborative grading workflow that integrates LLMs for metric design, report evaluation, benchmark comparison, and feedback generation, enhancing grading efficiency and consistency while providing students with valuable, peer-comparative feedback to support their learning and improvement.
- An evaluation of CoGrader demonstrates the effectiveness of the proposed workflow, highlights instructors' strong preference for its use, and underscores the broader potential of human-AI collaboration in tackling complex educational tasks, ultimately enhancing students' learning experiences.

2 Related Work

In this section, we review prior work on the challenges of assessing project-based learning (PBL), AI-powered tools for grading and educational assessment support, and the use of large language models (LLMs) as interfaces to assist instructors in educational contexts.

2.1 PBL and Assessment Challenges

PBL is widely regarded as an effective pedagogical method by engaging students in real-world [39] and interdisciplinary [14] challenges. In PBL, students typically engage in long-term projects, acquiring practical skills through completing artifacts, such as project reports [13]. Research has explored the advantages of PBL in enhancing students' skills, including fostering self-regulated learning [37], cultivating creativity [30], and developing critical thinking and collaboration [50].

Grading project reports is one of the primary method for assessing the learning outcomes of PBL. The traditional grading method

utilizes a comprehensive grading rubric that outlines specific criteria and corresponding levels of achievement for each aspect of the project [15, 31]. While this method aims to standardize evaluations, it has limitations. First, the rubric makes it hard to capture qualitative elements like creativity and originality [20]. Second, rubrics can introduce bias [65] in evaluation when multiple instructors or teaching assistants review reports, as the evaluators' personal preferences can influence the grading process. Third, developing a standard and comprehensive rubric for PBL projects by instructors can be time-consuming, especially when dealing with complex projects or diverse student work [65]. Moreover, rubric-based grading is susceptible to grading drift over time [42], as instructors often find it difficult to maintain consistent and fair evaluations throughout a lengthy grading process.

These limitations pose significant challenges due to the need for instructors to evaluate diverse, complex, and often subjective rubrics and metrics. Our proposed workflow addresses these challenges through a human-AI collaborative approach. By combining collaborative metric design and selection with an iterative, LLM-facilitated grading process, instructors can reduce the burden of grading standardized, repetitive metrics, allowing greater focus on complex, subjective evaluations. Meanwhile, the benchmarking-driven auto regrading process significantly accelerates the grading workflow, promotes fairness and mitigates grading drift, ensuring consistent and reliable assessments.

2.2 AI for Learning Assessment

The increasing complexity of educational assessment has driven the development of AI tools to support grading and feedback. These tools typically either automate the grading process or assist instructors by providing references and insights in a human-AI collaborative manner.

Advancements in artificial intelligence have enabled the development of tools that automate grading and assessment tasks [32]. For example, AI-powered programming assistants such as CodeAid and Learning with Style [36, 71] offer automated, real-time evaluation of code quality and style. Similarly, tools like VoiceCoach and TypeTalker [5, 86] provide immediate grading and comments for students' spoken language assignments. While these tools significantly enhance efficiency and consistency in assessment, they also raise concerns around transparency, fairness, and potential biases in AI-driven evaluation [51, 96], underscoring the importance of maintaining human oversight in educational contexts.

In addition to automated grading tools, a growing body of work explores the use of AI to facilitate feedback generation and communication among students and instructors, recognizing the pedagogical value of peer evaluation and teacher feedback [73, 92]. Tools such as SlideSpecs [87] and chat.codes [59] help aggregate presentation feedback and code comments from student discussions, streamlining the feedback process and enhancing communication efficiency.

Other platforms—such as RichReview++ [90], edX [91], Group-Meter [44], and DeepThinkingMap [46]—employ multimodal AI techniques to support feedback generation through visualizations, annotations, and recommendation systems for both peers and instructors. Additionally, large-scale peer grading and peer-comparison

tools have been developed to enhance fairness and scalability in assessing subjective assignments [21, 28, 67, 94]. While these systems highlight the benefits of feedback in fostering learning and engagement, they often rely heavily on active student participation, which can limit their effectiveness and scalability in real-world classroom settings.

Further work highlights the trade-offs in feedback quality and timing, especially in balancing rapid feedback with depth and personalization. Studies have explored how feedback timing impacts learning outcomes [56, 89] and how students perceive automatically generated feedback [9]. These insights reinforce the importance of designing AI-assisted feedback systems that maintain pedagogical value while providing timely support.

Our work builds on these foundations by introducing an AI-assisted framework that automates large-scale, peer-comparative evaluations under human guidance. The system ensures that instructors retain a central, decision-making role in the grading workflow through clearly defined responsibilities, while leveraging AI to enhance efficiency and consistency. It also promotes transparency and generates concise, personalized feedback for students. This approach addresses key ethical concerns in AI-driven assessment by preserving instructor oversight and reinforcing the educational value of meaningful, actionable feedback.

2.3 LLM-Powered Educational Support Systems

Recent studies have increasingly leveraged large language models (LLMs) to build educational systems that support both student learning and instructor workflows [17, 22, 35, 70, 76]. In this section, we focus on the latter—LLM-powered tools designed to assist instructors with a range of other educational tasks beyond grading.

For example, LLMs have been used to assist with course planning [24, 33], project or assignment design [69], and evaluating students' usage of generative AI tools [95]. Other systems leverage LLMs for interdisciplinary ideation [25] or simulate virtual classmates and teaching assistants to facilitate classroom interactions and collaborative learning [36, 40, 48].

Additionally, several systems have integrated LLMs into learning analytics pipelines and formative feedback workflows, enabling personalized support based on students' behavior and performance [19, 34, 79, 80, 93]. These works highlight the growing role of LLM-powered interfaces in enhancing various aspects of instructional design, classroom facilitation, and teaching support.

However, recent studies have raised concerns that LLM-powered systems may inadvertently reduce instructor engagement and cognitive effort over time [41, 43]. Additionally, emerging research has shown that LLMs may exhibit “self-favoritism” biases, rating their own outputs more favorably than human-written content [66], posing risks to fairness in grading contexts where originality is critical. Our system mitigates these concerns by positioning AI-generated comments primarily as evidential support rather than as final judgment. Key procedures, such as benchmark selection and AI comment revision, remain fully instructor-driven to encourage critical review of AI outputs and ensure fairness through sustained human oversight.

Building on these efforts, our work addresses a more complex and pedagogically significant challenge: grading open-ended reports in project-based learning. It advances human-AI collaboration by providing a scalable solution that supports instructors in managing grading and helps students receive meaningful, improvement-oriented feedback.

3 Formative Study

To gain a deeper understanding of the challenges and needs instructors face when grading students' project reports, we conducted semi-structured interviews with six university instructors, including two assistant professors, three lecturers, and one postdoctoral researcher. All participants had at least three years of experience in designing and grading project reports.

Each session lasted approximately 70 minutes and included two parts: a 30-minute discussion focused on current project report grading workflows and their pain points, followed by a 40-minute brainstorming session to identify which evaluation aspects might be appropriate for AI automation and which should remain under human control. This structure helped surface both practical challenges and design opportunities for collaborative AI-assisted grading systems.

With participant consent, three co-authors independently reviewed the interview notes and recordings. Using open coding, they extracted key concepts and clustered them into emerging themes. Through two rounds of discussion, discrepancies were resolved, and consensus was reached on the core themes. Below, we summarize the key findings from these discussions.

3.1 Workflow and Challenges in Grading Project Reports

Instructors generally follow a structured workflow when grading project reports, though they encounter several recurring challenges at each stage. Insights from our interviews reveal the following workflow stages and associated challenges:

S1: Appropriate Grading Metrics Design: The grading process begins with designing metrics that align closely with course learning outcomes and project requirements. A primary challenge at this stage is ensuring that these metrics remain consistently aligned with the objectives shared with students, providing clear expectations while accommodating diverse approaches in student submissions.

S2: Quick Overview and Benchmark Selection: Instructors often start the grading process by skimming through all submissions to gain a high-level understanding of the overall quality and performance distribution within the class. During this phase, they also identify a few representative submissions to serve as benchmarks to guide more consistent evaluation later. However, this step is always cognitively demanding, especially when the number of submissions is large or when the quality differences are subtle. Instructors often find it challenging to quickly identify suitable benchmark reports while ensuring their initial impressions remain unbiased and representative of the overall submission quality.

S3: Detailed Evaluation, Benchmark Adjustments, and Retroactive Grading: After the initial overview and selection of benchmark reports, instructors proceed with the detailed evaluation

of each report according to the grading metrics, often comparing them to the selected benchmarks. During this phase, instructors frequently update benchmarks and revisit previously graded reports to ensure consistency, making necessary adjustments based on new insights. This iterative process of retroactive grading is time-consuming and poses significant challenges, particularly when grading open-ended reports. Instructors highlighted the challenge of maintaining fairness in subjective judgment throughout the grading process, especially when evaluating factors like innovation and peer-comparative design quality.

S4: Comprehensive Feedback and Synthesizing Class-Level Insights:

After evaluating each report, instructors provide feedback, though they often find it challenging to deliver detailed, cohesive comments that address all critical points. Writing comprehensive feedback is time-intensive, leading some instructors to focus on limited key points and encourage students to seek additional guidance in office hours. Instructors also recognize the importance of comparing individual reports against overall class performance to highlight improvement areas. However, synthesizing these insights across all reports and effectively communicating them to the class presents another significant challenge, requiring substantial time and effort to ensure feedback is both meaningful and actionable.

3.2 Perceptions of AI in Grading Assistance

In the second phase of our formative study, we first asked instructors to share their overall impressions of integrating AI tools like ChatGPT into the grading process. All instructors expressed an open-minded and positive attitude toward using AI to support the evaluation of project reports. At the same time, most emphasized the necessity of maintaining a certain level of human control over key parts of the grading workflow. To gain deeper insights into the appropriate roles of AI in the system, we further posed two guiding questions and encouraged instructors to openly brainstorming:

(Q1) *Which parts of the grading process could be effectively managed by AI tools alone such as ChatGPT or automated algorithms?*

(Q2) *Which aspects of grading would remain challenging or unreliable for AI alone, where human expertise is indispensable?*

Based on their reflections, we identified specific tasks corresponding to each question. In the following section, we first summarize the tasks that instructors viewed as well-suited for AI automation (Q1), followed by tasks where AI could play a supporting role but human judgment remains essential (Q2).

3.2.1 Tasks Suitable for AI and Automated Algorithms Alone:

(1) Automated Metrics Generation and Recommendation:

AI can support instructors by analyzing project requirements to suggest relevant assessment metrics, acting as a brainstorming tool to recommend valuable criteria for consideration in grading design.

(2) Standardized Report Evaluation:

Instructors noted AI's potential to handle basic assessments of submissions, including generating summary statistics, preliminary scores, and comments based on standardized criteria. These automated checks could cover areas like formatting, grammar, and other objective elements.

(3) Content Summarization and Key Point Extraction:

AI can assist by efficiently summarizing reports and extracting

key points, providing instructors with a quick and clear overview of each submission's core ideas.

3.2.2 Tasks Where AI Assists but Human Judgment is Essential:

- (1) **Benchmark Selection and Retroactive Grading:** Instructors noted that while AI can streamline the benchmarking process by suggesting rankings, classifications, and comparisons among student reports, it may lack the nuanced understanding needed for complex assessments. AI can assist by automating score updates during retroactive grading, reducing the manual effort required for revisiting prior evaluations. However, instructors emphasized that human judgment is essential to interpret context-specific elements and ensure that benchmarks align with course objectives and standards, particularly when evaluating complex or open-ended submissions.
- (2) **Evaluating Subjective Aspects:** While AI can aid in evaluating subjective aspects—such as creativity, novelty, and aesthetic quality—final judgment should remain with instructors. AI could provide supportive comments and, where possible, generate quantitative insights to complement instructor evaluations, such as assessing workload intensity, identifying unique insights, or highlighting innovative ideas within reports.
- (3) **Feedback and Insights Generation:** AI can assist in generating detailed, individualized feedback by organizing and expanding upon instructors' notes and evaluations, rather than generating feedback independently. This approach enables AI to provide students with clear, actionable points and compile class-wide insights to identify common strengths and areas for improvement. However, because AI lacks the contextual understanding and sensitivity required for nuanced feedback, human input remains essential to ensure the feedback aligns with learning objectives and addresses each student's unique approach and progress.

4 System Design

In this section, we first outline the core design requirements of the system, followed by an overview of CoGrader through a representative user example. We then describe the system's frontend interface and backend components in detail.

4.1 Design Requirements for the Grading Tool

Based on the insights gained from the formative study, we identified several key requirements for the development of an AI-assisted grading tool that facilitates a human-AI collaborative process in grading students' project reports:

R1: Automatic Extraction of Core Metrics from Project Requirements. The system should be able to analyze and interpret instructors' project descriptions to automatically extract and summarize key grading criteria. This includes outlining essential evaluation metrics directly derived from the provided requirements, ensuring that foundational assessment standards are identified early. Automating this step helps streamline the initial grading setup and reduces manual workload, allowing instructors to concentrate on more complex and subjective aspects of evaluation.

R2: Enriching Metrics through Collaborative, Multi-Source Generation.

Building on the core metrics, the system should support a human-AI collaborative process to expand and refine the metric set. This includes AI-suggested metrics beyond project requirements, integration of external standards or rubrics, and instructor-provided additions or modifications. The system should also support co-designing new metrics tailored to specific learning goals, ensuring that the final assessment criteria are comprehensive, context-aware, and aligned with instructional intent.

R3: Automated Preliminary Analysis and AI-Driven Insights. The system should provide automated preliminary assessments of all submissions, including summary statistics of evaluation scores and AI-suggested comments or recommendations for instructor review. These insights help instructors quickly grasp overall class performance, identify potential benchmark submissions, and enter the grading process with a clearer understanding of report quality distribution—ultimately supporting a more informed and efficient evaluation workflow.

R4: AI-Assisted Report Benchmarking, Comparison, and Retroactive Grading. The system should support AI-assisted benchmarking and report comparison, allowing instructors to select, review, and update benchmark submissions with AI recommendations throughout the grading process. These benchmarks serve as reference points to maintain grading consistency. The tool should also enable automated retroactive grading, where previously scored reports are re-evaluated based on updated benchmarks. This AI-instructor collaboration reduces manual effort while ensuring fairness and coherence across all submissions.

R5: AI-Assisted, Instructor-Centered Feedback and Insight Generation. The system should support instructors in composing clear, structured feedback by synthesizing their annotations, comments, and grading decisions across the evaluation process. AI-assisted summarization and phrasing can reduce the time and effort required to generate detailed feedback. Crucially, the final feedback should prioritize instructor-authored content, with AI-generated suggestions used to supplement areas not covered—ensuring clarity, consistency, and pedagogical integrity.

R6: Ensure Instructor Authority and Balance AI Reliance. The system must maintain instructors' full authority over the grading process by clearly positioning AI outputs as supportive references rather than final decisions. AI should primarily assist with routine, low-judgment tasks—such as grammar checks and initial comment drafting—to reduce cognitive load and allow instructors to focus on nuanced, expertise-driven evaluation. All critical actions, including selecting criteria, identifying benchmark reports, and assigning final grades, remain under instructor control. To encourage instructor active review, AI-generated comments must include direct excerpts from student reports as evidence, discouraging passive acceptance of AI suggestions.

Together, these design requirements respond to the key challenges surfaced in our formative study, highlighting opportunities where AI can meaningfully support and streamline the grading process, while ensuring that instructors retain full control over final decisions.

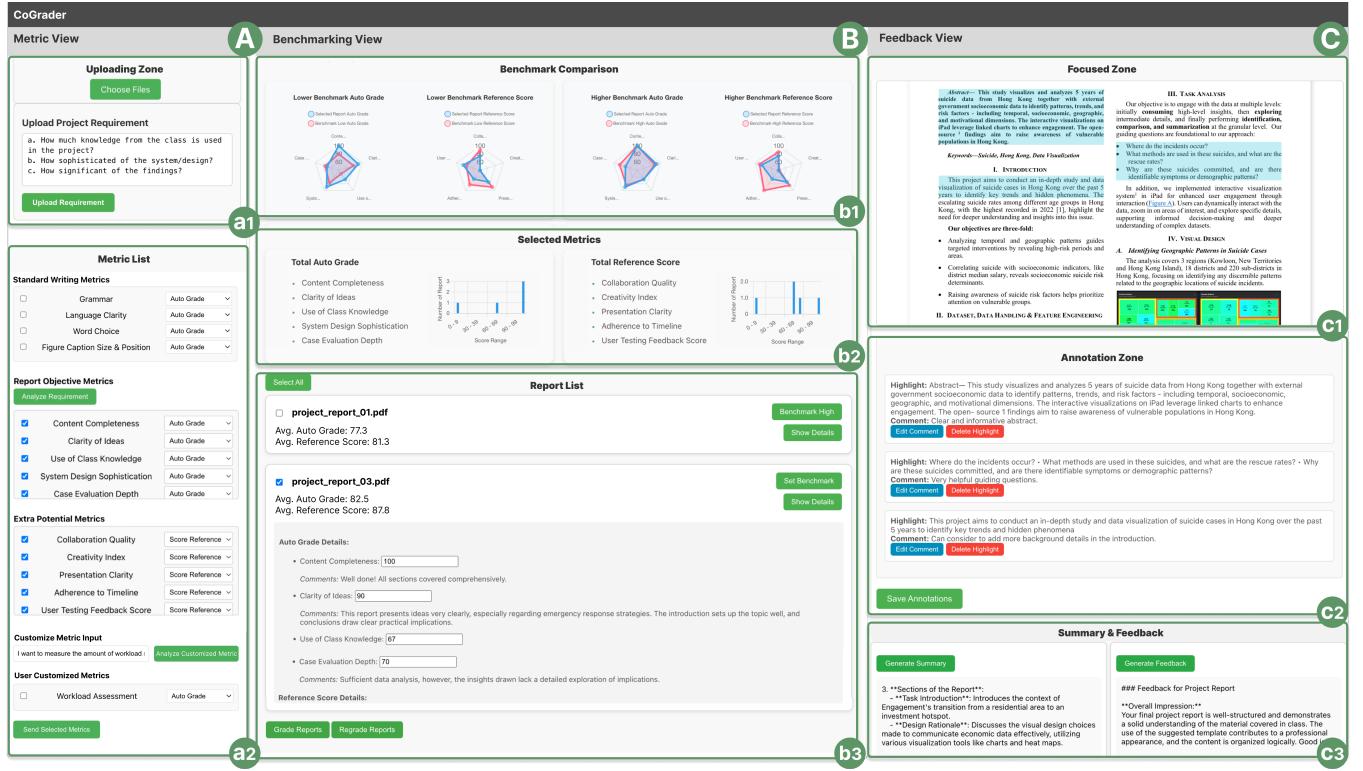


Figure 1: System interface of CoGrader. The *Metric View* (A) provides functions for uploading reports and project requirements (a1), along with co-design tools for generating multi-perspective project assessment metrics (a2). The *Benchmarking View* (B) includes a Benchmark Comparison zone (b1) where users can compare metric scores between a focused report and user-selected benchmarks. The Selected Metrics zone (b2) below offers an overview of metric score distribution across all reports. The Report List zone (b3) displays uploaded reports, allowing users to trigger automatic report evaluation, review and edit AI-generated scores and comments for each metric, or select benchmarks for AI-assisted iterative regrading. The *Feedback View* (C) shows the contents of the focused report (c1), enabling users to review and annotate (c2). Users can also access the AI-generated summary and create the final feedback for the focused report (c3).

4.2 CoGrader System Design

Building on the design requirements outlined above, we developed CoGrader, a system that provides AI-driven support for instructors in grading project-based reports. CoGrader is designed to seamlessly integrate into existing grading workflows, combining the efficiency of AI with the nuanced judgment of human instructors to enhance grading consistency and reduce workload. The system automates the analysis of project requirements, supports collaborative metric generation, and streamlines benchmarking and feedback processes. This section begins with a walkthrough of a typical user experience with CoGrader, followed by a detailed explanation of each system component and how it addresses the identified design requirements.

4.2.1 Example User Walkthrough. Jason teaches a data visualization course at a local university. As a hands-on course, he assigns a term project midway through the semester, which he plans to use as the primary assessment of his students' learning outcomes. The project is open-ended, allowing students to either choose from a set of sample topics or select their own datasets and perform

self-directed analytics tasks. Along with the project, Jason provides students with a set of project requirements, which include key details such as the report style (e.g., academic writing), report contents (e.g., Introduction, Related Works, ...), and general grading metrics (e.g., system sophistication, design quality, ...).

At the end of the semester, Jason collects over 20 project reports, each approximately 5 pages long, and faces the daunting task of grading them. Although he has the project requirements, they are broad and abstract, making it challenging to grade efficiently while maintaining high-quality grading standards. To assist with this, Jason uses CoGrader. Jason begins by uploading all the student reports into CoGrader, along with the original project requirements that were shared with the students at the start of the term. He then clicks the “Analyze Requirement” button in the Metric List area, allowing the AI to assist in generating assessment metrics based on both the project requirements and AI suggestions. After reviewing the AI’s analysis, Jason selects the recommended metrics: “Content Coverage”, “Technical Depth”, and “Innovation and Creativity Index”. Recognizing that “Content Coverage” is more objective, he sets it as an “Auto Grade” metric. In contrast, he designates the

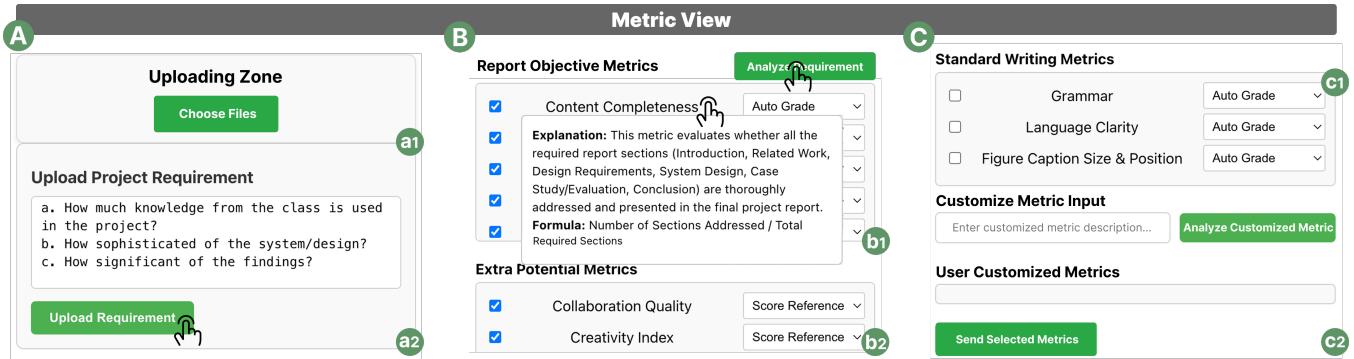


Figure 2: The components in Metric View: (A) The Uploading Zone of CoGrader, where users can upload project requirements and report files. (B) Automatic metric analysis, generating Report Objective Metrics and Extra Potential Metrics based on the uploaded project requirements. (C) The predefined Standard Writing Metrics and User Customized Metric design area which allow users to create metrics using natural language descriptions.

other two as “Score Reference” metrics to reflect their more subjective nature and to remind himself to later review the AI-generated comments for those aspects.

With the metrics defined, Jason moves to the **Benchmarking View**, where he allows the AI to automatically grade all the reports based on the selected metrics and generate reference scores and comments. This provides him with an overview of the AI-recommended score distribution and the corresponding report rankings in the Report List below. By clicking on a report card, Jason can review its details in the right-hand **Feedback View**, where the Focused Zone allows for in-depth examination and reflection. He reviews the AI-generated scores and comments alongside the original report content, making annotations in the Annotation Zone and adjusting the scores and comments for each metric as needed. After reviewing about one-third of the reports, Jason identifies one report as a good example (higher benchmark) and another as a bad example (lower benchmark). He designates these as benchmarks back in the Report List, spending additional time to refine the grading and comments for these two reports. To expedite the grading of the remaining reports, Jason selects them all and clicks the “Regrade Reports” button at the bottom of the Report List area, allowing the AI to regrade each report based on the previously defined metrics and the selected benchmarks. With the updated scores in hand, Jason conducts an iterative review, comparing each report’s metric scores against the two benchmarks using the radar charts in the above Benchmark Comparison area. Once satisfied with the scores, Jason moves to the Summary & Feedback zone in the bottom-right corner of the **Feedback View**, where he uses the AI to automatically generate personalized feedback based on the grading results, his annotations, and comments. Finally, Jason exports all the feedback and sends it to his students for their review.

4.3 CoGrader Frontend Components

In the following section, we introduce the CoGrader frontend user interface and the core components.

4.3.1 Metric View. CoGrader is meticulously designed to assist instructors in the project report grading process while ensuring

they retain control throughout. The grading process begins with the instructor uploading the project report files (fig. 2-a1) and the project requirements (fig. 2-a2) via the Uploading Zone (fig. 2-A).

After the project requirements are uploaded, CoGrader automatically analyzes these requirements to support the instructor’s grading metrics generation (**R1**). Instructors can initiate this process by clicking the “Analyze Requirement” button (fig. 2-b1). The system generates two categories of metrics: **Report Objective Metrics** (fig. 2-b1) and **Extra Potential Metrics** (fig. 2-b2). The **Report Objective Metrics** are derived directly from the uploaded requirements, ensuring alignment with the instructor’s expectations. In contrast, the **Extra Potential Metrics** are suggested by the AI based on its understanding of valuable aspects that might not be explicitly mentioned in the requirements but could be important.

Each metric is presented in a list format, where hovering over the metric name provides a detailed explanation and suggested formula (fig. 2-b1). Instructors can select metrics for further analysis by checking the box next to each metric. They can also categorize each metric into two types: *Auto Grade* or *Score Reference*. *Auto Grade* metrics are those where instructors can rely fully on AI-generated results, while *Score Reference* metrics are more subjective, requiring AI suggestions but leaving the final judgment to the instructor. This design is in line with the user control principle outlined in **R6**, ensuring that from the metric selection phase onward, instructors have primary control while AI provides sufficient support.

Additionally, CoGrader offers predefined **Standard Writing Metrics**, such as grammar and language clarity checks (fig. 2-c1), for instructors to select as needed. To provide even greater flexibility, the system includes a **Customized Metric Design** feature (fig. 2-c2), allowing instructors to describe additional metrics in natural language. The AI then assists by generating customized metrics and their corresponding formulas. It also cross-references the instructor’s customized metrics with existing ones to avoid redundancy, highlighting any overlaps in red. This collaborative approach to metric design ensures that the generated metrics are comprehensive and reliable, aligning with the design requirement **R2** while still preserving user control as emphasized in **R6**.

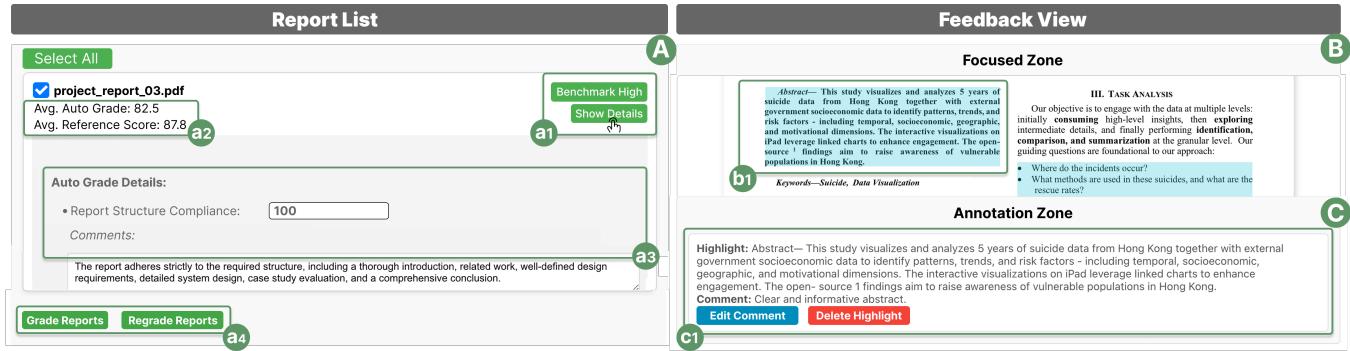


Figure 3: (A) The Report List in the *Benchmarking View*. Each uploaded report is presented as a report card. Users can view average scores for each metric group (a2), review and edit detailed scores and comments for selected metrics (a3), and select benchmark reports (a1). Additionally, users can initiate automatic grading and benchmark-driven regrading with a click (a4). **(B & C):** The focused report details in the *Feedback View*. (B) Users can view the focused report file by clicking on each report card in the report list (A). Users can freely highlight sections (b1) and add comments (c1) to the highlighted content. (C) These annotations are saved for use in generating report feedback.

4.3.2 Benchmarking View & Feedback View. After co-designing and selecting the desired metrics, users can send the metrics to the **Benchmarking View** (fig. 1-B) for further report grading through a collaborative benchmarking process (R4). The **Benchmark Comparison** component (fig. 1-b1), the **Selected Metrics** component (fig. 1-b2) work together with the **Focused & Annotation & Summary Zone** (fig. 1-c1,c2,c3) in the **Feedback View** to assist instructors in benchmarking and reports grading.

In the **Benchmarking View**, user-uploaded reports are displayed in a list format (fig. 3-A). Initially, by clicking the “Grade Reports” button (fig. 3-a4) at the bottom of, backend AI components will quickly grade all reports based on the user-selected metrics, and return reference scores along with AI-generated comments to provide the automated preliminary analysis (R3). The average scores recommended by AI are displayed in the Report List (fig. 3-a2). Meanwhile, instructors can click the “Show Details” button to view detailed scores for each metric along with AI’s comments (fig. 3-a1). These AI-recommended scores serve as a reference, and instructors are free to modify and update the scores (fig. 3-a3) for each metric (R6). To start a in-depth review of each report, instructors can click on each report card to focus and view the original report in the **Focused Zone** (fig. 3-B). Instructors are able to highlight, annotate, and edit freely within the **Annotation Zone** (fig. 3-C) during the grading process. All the metric scores, together with instructors’ highlights and annotations will all be saved for the further feedback generation in the **Summary and Feedback Zone** (fig. 5).

During the reviewing process, once instructors identify reports suitable to serve as benchmarks, they can go back to the Report List and click the “Set Benchmark” button inside each report card (fig. 3-a1) to designate the current report as either a *Benchmark High* or *Benchmark Low* for further grading reference.

After instructors decide the benchmarks and select the report to focus on, their metrics scores will be updated in the **Benchmark Comparison Zone**, where users can easily compare the focused report with the selected benchmarks under each metric for a further grading adjustments (fig. 4-A). The **Benchmark Comparison**

Zone features four radar charts, with each vertex representing a unique metric. The first two radar charts compare the focused report with the *Lower benchmark* (fig. 4-a1), and the last two compare it with the *Higher benchmark* (fig. 4-a2). A consistent color coding scheme is used, with red representing the benchmarks and blue representing the focused report. Instructors can hover over the labels in the radar charts to view the full name of each metric (fig. 4-a3). The radar charts provide a clear comparison for the focused report, helping instructors to quickly review the AI’s grading for each metric. Additionally, an overview of the metric scores distribution among all reports is provided below, clustered into two categories based on the user’s original selection in **Metric View** (fig. 4-B). While reviewing individual metric values, users can sort the report cards in the Report List below (fig. 3-A) by clicking on each metric name to arrange them in descending order of scores (fig. 4-b1). These functions streamlines the benchmarking process, aligning with the proposed design requirement (R4) in a convenient and efficient manner.

Throughout the AI-assisted grading process, instructors often perform multiple iterations of benchmark selection and regrading—a task that can be time-consuming and cognitively demanding when aiming to ensure fairness and consistency. To support this process, CoGrader provides a regrade function aligned with our design goals (R4) and safeguards the balance of reliance on AI (R6). Once benchmarks are selected, instructors can trigger automated regrading by clicking the “Regrade Reports” button (fig. 3-a4), prompting the system to re-evaluate each report against the selected metrics, benchmarks, and comments. Crucially, these AI-generated updates also serve as supportive suggestions only and can be freely discarded or revised by instructors. Instructors are encouraged to critically review these outputs and repeat the regrading process as needed to ensure the outcomes align with their pedagogical standards (R6).

Finally, once the benchmarking-driven grading process is complete, instructors can generate summarized feedback for each report to share with students. By clicking the “Generate Feedback”

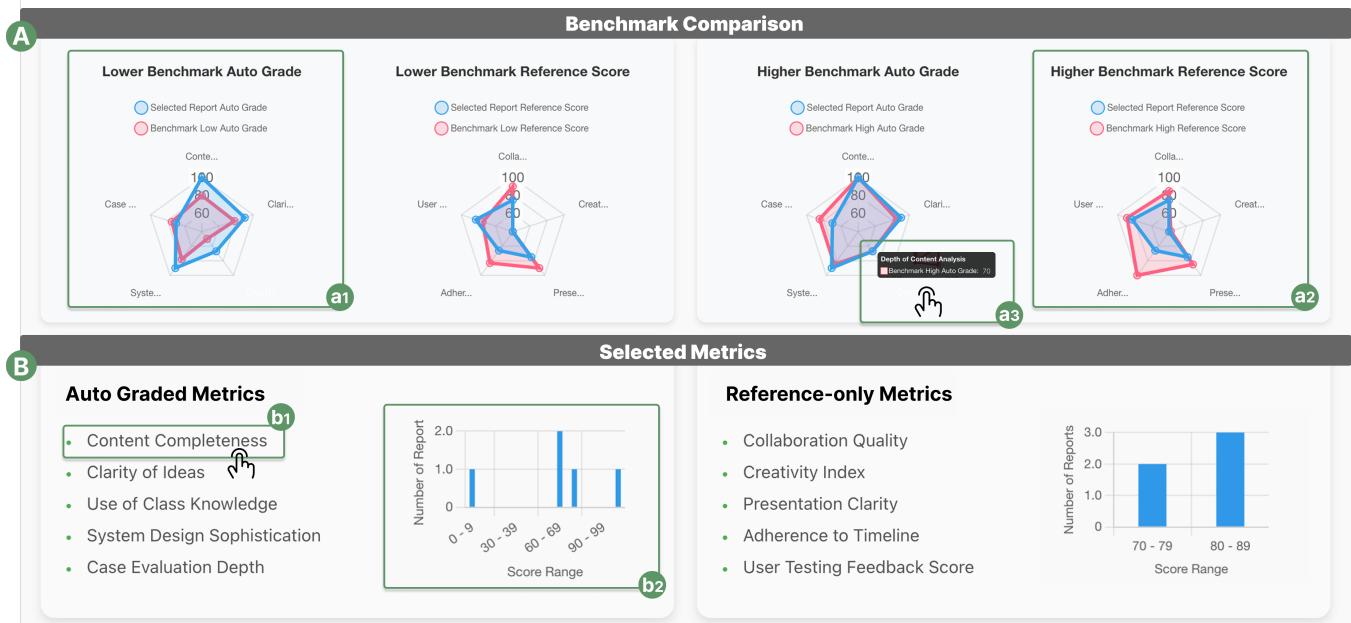


Figure 4: (A) The Benchmark Comparison component. Radar charts (a1 & a2) comparing the focused report with both high and low benchmarks. Each vertex (a3) represents a user-selected metric, categorized based on user labels (auto grade or reference score) from the Metric View (fig. 2-B & C). (B) The Selected Metrics component. Users can check the selected metrics list and the corresponding score distribution charts. Users can click on each metric (b1) to view its score distribution (b2) across all reports, with the Report List (fig. 3-A) automatically sorted in descending order based on the selected metric.

button in the **Summary & Feedback** zone (fig. 5-A,B), AI will automatically compile all metric grades and comments, along with the instructor's highlights and annotations from (fig. 3-A,B,C), to generate personalized summary and feedback for each report (fig. 5-A,B). The feedback is structured to prioritize instructor-authored content, followed by instructor-edited AI suggestions, with pure AI-generated comments used only to fill remaining gaps (R5). Instructors remain free to review and revise the output before exporting it to students (R6). These features support consistent and personalized feedback, enhancing the student learning experience while reducing the instructor's workload (R5 & R6).

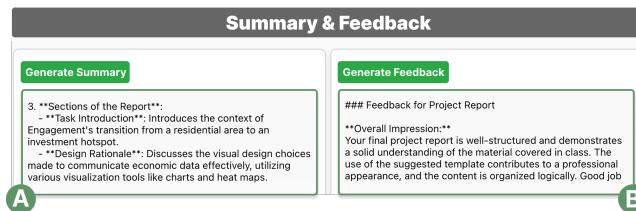


Figure 5: The Summary and Feedback zone, where users can view an AI-generated summary of the selected report (A). Users can also generate feedback for the report (B), which integrates highlights and comments from Focused Zone and Annotation Zone (fig. 3-B&C), along with detailed metric scores and comments from Report List (fig. 3-A), to produce a comprehensive report.

4.4 CoGrader Backend LLM Development

In this section, we introduce the backend structure of CoGrader, focusing on the design and integration of LLMs.

The core AI functionality of CoGrader is built using OpenAI's ChatGPT-4o API [63], which offers a lightweight yet efficient solution, striking a balance between fast response times and model performance. The backend architecture aligns with the frontend system design and is divided into three main components: *Collaborative Metric Design* (fig. 6-A), *Benchmarking-driven Grading* (fig. 6-B), and *Feedback Generation* (fig. 6-C). Each of these components leverages the *Assistant API* to build AI agents that support report grading, file search, and conversation context management—ensuring a smooth and cohesive grading experience [61].

In the *Collaborative Metric Design* component, the LLM-based AI agent processes users' project requirement inputs and any extra requirement descriptions (fig. 6-A). To ensure structured and consistent output, the *Structured Response* function is employed [62]. The first round of requirement analysis generates two sets of metrics: *Report Objective Metrics* and *Extra Potential Metrics*. While users are free to specify additional metrics using natural language descriptions, all *Customized Metrics* are passed through a *Redundancy Check* to prevent duplication or overlap with existing metrics. After this process, the system generates three categories of metrics (with explanations and recommended formulas), allowing users to review and select appropriate ones for the grading process.

In the *Benchmarking-driven Grading* component, another LLM-based AI agent processes the uploaded project reports and applies

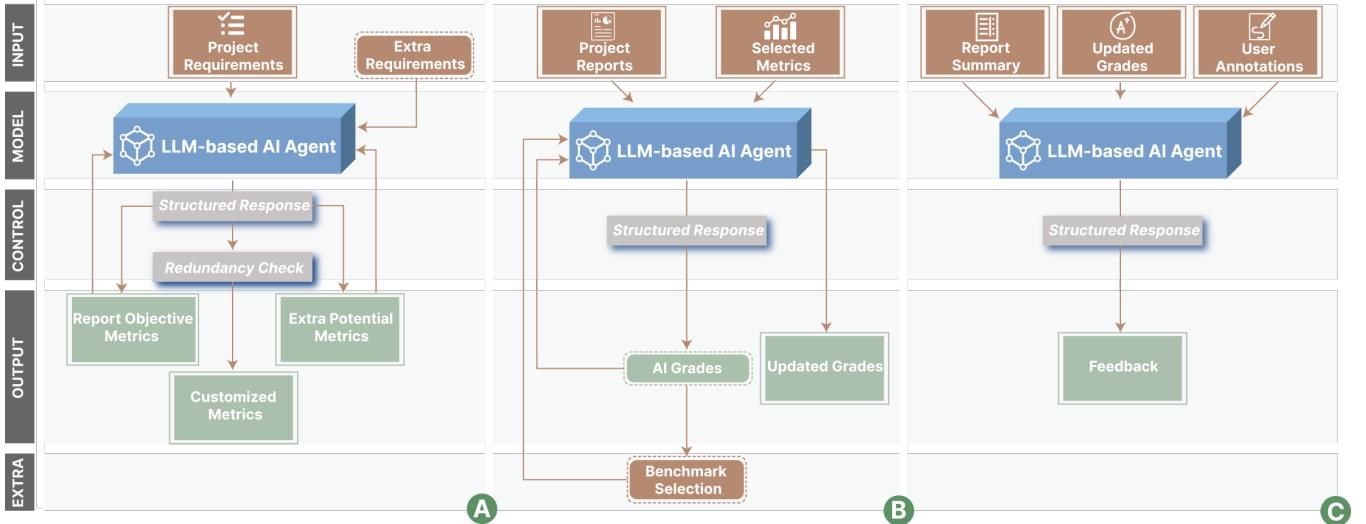


Figure 6: (A): In the *Collaborative Metric Design* component, uploaded project requirements are automatically analyzed by an LLM-based AI agent built using the Assistant API. The output is structured into *Report Objective Metrics* and *Extra Potential Metrics*. Users can also input additional requirements, which the AI agent will check for redundancy before generating *Customized Metrics*. (B): In the *Benchmarking-driven Grading* component, both project report files and user-selected metrics are sent to another AI agent for initial grading. After users review and edit the first-round grades and select benchmark reports, the updated grades and benchmarks are sent back to the AI agent for regrading the reports. (C): In the *Feedback Generation* component, The AI-generated report summary, updated grades, and user annotations are sent to the extra AI agent to generate structured feedback for students.

the selected metrics (fig. 6-B). The system uses the *File Search* tool provided by the Assistant API [60], a built-in retrieval-augmented generation (RAG) tool that enables efficient file search and content analysis [16]. For the initial grading, we prompt the AI to “think” and “reason” about each report based on the selected metrics. It searches for relevant evidence within the report, assigns initial scores, and generates reference comments with supporting excerpts. After the instructor updates the AI-generated scores and selects benchmark reports based on their expertise, the system re-grades each report by comparing them against the benchmarks and adjusting the scores accordingly. During the benchmarking-based regrading stage, we prompt the AI to “compare” the target report and its current evaluations (scores and comments for each metric) against the selected benchmarks to refine its reasoning and update the evaluation accordingly. Once the grading process is completed, the *Updated Grades* and comments are sent back to the frontend for review.

In the *Feedback Generation* component, an additional LLM-based AI agent synthesizes user-updated scores, comments, annotations, and report summaries to produce personalized feedback for each student (fig. 6-C). To maintain consistent feedback style, the output is structured through the *Structured Response* feature. This ensures uniformity and clarity in the feedback provided to students. Detailed prompt templates for each components are available in the supplementary materials.

4.5 Implementation

CoGrader is developed as a full-stack web application, featuring a React.js [27] frontend for an intuitive and responsive user interface. The backend, built with Python using Flask API [29], hosts and manages the LLM modules. The frontend and backend are connected via RESTful APIs, allowing smooth communication and integration between the user interface and the LLM-powered grading modules.

5 Evaluation

We conducted an evaluation study to assess the effectiveness of the proposed human-AI collaborative workflow in CoGrader, as well as the performance of each component within the system. Our primary goal was to evaluate how well the workflow enhances grading efficiency, reliance, and consistency through LLM assistance. To further evaluate concerns about over-reliance on AI, we also examined participants’ actual engagement behaviors with AI-generated outputs. To explore these objectives, we addressed the following research questions:

RQ1: How effective is the metric generation step in helping instructors define and select assessment metrics?

RQ2: How well does the LLM’s grading and comments align with instructors’ assessments, and how do instructors perceive the efficiency, reliance and quality of LLM-generated scores and comments?

RQ3: How does the benchmark-driven grading process impact grading consistency and efficiency?

RQ4: How do instructors perceive the reliance of the LLM-generated feedback, and to what extent does it help streamline the feedback process?

RQ5: What are instructors' overall attitudes toward the human-AI collaborative approach in CoGrader, and how do they evaluate its impact on improving the grading workflow?

RQ6: How closely do the instructors' grading results with CoGrader align with their original, purely human-assigned report grades?

RQ7: To what extent do instructors actively verify, adjust, or override AI-generated outputs during the grading process?

5.1 Procedure of User Study

Participants were recruited from the authors' institution via email invitation during the summer of 2024. This study received approval from the Institutional Review Board (IRB) of the authors' institution. After obtaining participants' consent, demographic data were collected anonymously, focusing on their experience as instructors, their teaching roles, length of experience, and previous attitudes toward using LLMs like ChatGPT for grading.

Each user study session lasted approximately 80 minutes and consisted of three stages: (1) a 50-minute system exploration and grading experiment, (2) a 15-minute questionnaire to collect participants' perceptions of each step in the workflow, and (3) a 15-minute post-interview to gather qualitative feedback on user experience and system design.

Before beginning the grading experiment, participants were provided with a brief background introduction and a system demo. They were shown how to use the proposed workflow, covering metric generation, benchmark-driven grading, and feedback generation steps. Participants were provided with a sample project requirement and five sample project reports from a real-world master-level data visualization course conducted in the previous semester at the authors' institution. These materials were collected with IRB approval and consent from both the instructors and the student authors. Participants were required to upload both the project requirements and reports to the system and begin the grading process. Their primary task was to provide a final ranking, assign reference scores (out of 100), and determine reference letter grades (from F to A) for each report. A score of 85 and above corresponds to an A, with every 5-point interval representing a subgrade (e.g., 80-85 is A-, 75-80 is B+, and so on). To support the analysis of instructors' engagement with AI-generated outputs, we logged participants' interactions with the system, including the number of selected metrics, the frequency of manual comment additions or edits, and the proportion of AI-generated content that was adjusted or overridden. The experimental results and logs were later used in several quantitative analyses to evaluate the system's effectiveness (**R6**) and instructors' level of engagement with AI assistance (**R7**).

After completing the grading experiment, participants were given a 7-point Likert-scale questionnaire consisting of multiple rating questions to collect their opinions on each step of the workflow, following the research questions (RQ1–RQ5). A grading scale of 1 to 7 was used to represent participants' negative to positive attitudes toward each feature. Finally, after completing the questionnaire, participants were invited to a brief interview to share

their thoughts and comments on each step of the workflow and the system itself.

5.2 Project Report Data Collection

The project report requirements, submissions, as well as the ground truth final scores and letter grades, were collected from a master-level data visualization course at the authors' institution during the spring semester of 2024. This project served as the main assessment for the course, in which each group of students was asked to choose an open dataset, identify specific research questions of interest, and apply the knowledge they gained throughout the course to analyze the dataset using visual analytics techniques. The students reported their methods and findings in a formal written report. The final scores were determined by taking a weighted average of evaluations from one course instructor and two senior teaching assistants. We obtained consent from both the course instructors and students to use their reports for our research at the end of the semester, and we also received IRB approval for the data collection. For the 5 sample reports used in the user experiments, we selected reports with letter grades ranging from B- to A, aiming to reflect the real distribution of students' work. All student personal information was anonymized before being used in the experiments to ensure privacy and confidentiality.

5.3 Participants

We recruited 12 participants (P1-P12, 8 male and 4 female) through email invitations to instructors and teaching assistants at our institution who had experience teaching this data visualization course. Of the participants, four (P1-P4) were university professors or lecturers, and eight (P5-P12) were senior PhD students with teaching assistant experience in this course. On average, participants had 3.25 years of experience in designing or grading project reports. Among the 12 participants, seven had experience using LLMs to assist with grading, while five did not. Each participant spent approximately one hour on the study and received \$30 in compensation.

5.4 User Study Results

In this section, we discuss our findings regarding instructors' experiences with CoGrader, focusing on their perceptions of each step in the proposed workflow: collaborative criteria generation, the benchmark-driven grading process, and feedback generation.

5.4.1 RQ1: Instructors find the collaborative metric generation step efficient, effective, and acceptably reliable. To evaluate instructors' experiences with the collaborative metric design step, we asked participants to rate the efficiency and reliability of various metric generation methods on a 7-point scale, with 4 representing neutrality. Overall, participants rated the efficiency of the collaborative metric generation process very positively (**Q4:** mean = 6.5, stdev = 0.522, fig. 7). For the reliability of each metric design component, the *Report Objective Metrics* feature (automatic analysis of report requirements) received positive ratings (**Q1:** mean = 5.91, stdev = 0.793, fig. 8), highlighting the strong ability of LLM to convert the requirements of the report into concrete quantitative metrics. Similarly, the *Extra Potential Metrics* feature (automatic metric recommendation beyond report requirement) was highly rated (**Q2:** mean = 6.16, stdev = 0.937, fig. 8), with participants appreciating



Figure 7: Summary of user study results on participants' ratings for each system component's Efficiency and Satisfaction. Each question was rated by participants on a 7-point scale, where dark brown represents the strongest negative response, dark green indicates the strongest positive response, and neutral beige marks the midpoint. The number inside each colored bar reflects the count of participants who selected that specific rating.

its ability to brainstorm additional useful metrics with clear explanations. On the other hand, the *Custom Metric Design* feature (metric co-design through user inputs) also scored well for its reliability (Q3: mean = 5.83, stdev = 1.115, fig. 8). Participants overall expressed satisfaction with the collaborative metric design step. However, participant P4 noted occasional misunderstandings by the LLM when using the *Custom Metric Design* function and suggested allowing users to freely edit the metric explanations and formulas to improve the process. Participants also recommended adding features like semantic grouping and metric weighting for further customization.

In general, participants felt that the collaborative metric generation step aligned well with their workflow, significantly enhancing both grading efficiency and reliability.

5.4.2 RQ2: Instructors find the LLM's grading and comments efficient, with generally acceptable reliability and alignment with their assessments. Regarding the efficiency of LLM-generated scores and comments, participants expressed strong positive attitudes (Q9: mean = 5.92, stdev = 1.083, fig. 8) towards the LLM's initial grading, and even more favorable attitudes (Q10: mean = 6.42, stdev = 0.515, fig. 8) towards the benchmark regrading. Participants noted the system's ability to handle multiple metrics and compare reports with quick response times as particularly valuable.

In terms of the reliability of LLM-generated scores and comments, participants rated the benchmark-regrading process highly, with benchmark-regraded scores (Q7) receiving a mean of 6.08 (stdev = 0.900, fig. 8) and comments (Q8) receiving a mean of 5.83 (stdev = 1.11, fig. 8). However, the reliability of the LLM's initial scores (Q5: mean = 4.91, stdev = 1.083, fig. 8) and initial comments (Q6: mean = 4.83, stdev = 1.030, fig. 8) were rated slightly lower. One participant (P11) noted that the comments were sometimes too generic, applying to multiple reports, which reduced their specificity. P8 and P11 also mentioned that certain selected metrics were

occasionally missed in the initial grading and suggested adding an automatic checking function to help refine the LLM's initial output. The user editing function was considered crucial for refining the initial grading by allowing adjustments to the LLM's outputs.

From the interviews, the majority of participants found the scores and comments to be reliable and contextually appropriate. Participants particularly appreciated the benchmark-regrading process, with P12 stating, “*The regrading process specifically compares each report with the benchmarks I selected. I can clearly see from the comments why one report performed better or worse than the benchmarks under each metric, which is very helpful and significant. I also found that most of the LLM comments aligned with my assessment.*”

5.4.3 RQ3: Instructors find the benchmark-driven grading process significantly improves grading consistency and efficiency. Participants gave highly positive ratings for the benchmark-driven regrading process, praising both its efficiency (Q10: mean = 6.42, stdev = 0.515, fig. 7) and reliability (Q11: mean = 6.00, stdev = 0.853, fig. 8). They found that this process significantly reduced their workload and maintained grading consistency. As P1 remarked, “*The system excels in the benchmarking process. AI makes grading faster, and the comparison charts simplify the process. It reduced my effort significantly, allowing me to focus more on content quality and ensure my judgment is more unbiased*”. Participants (P1, P2, P5 & P10) also appreciated that most comments included specific comparisons with the benchmark reports after the LLM regrading. Beyond the LLM's role, the clear radar charts also provided visual comparisons and references, making it easier for participants to make grading decisions. On the other hand, participants valued the ability to

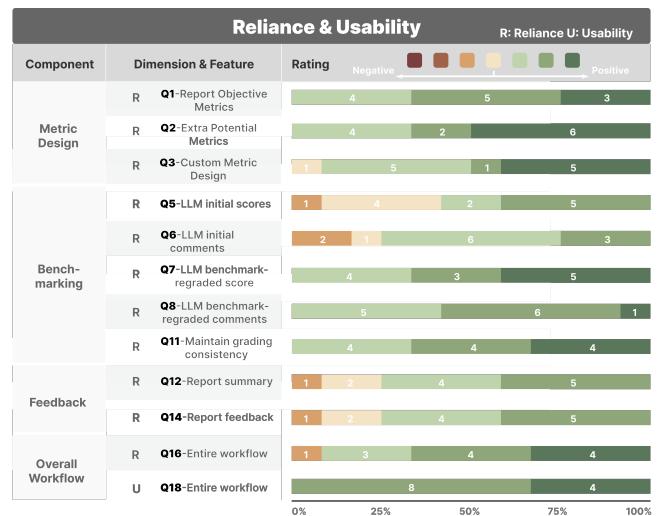


Figure 8: Summary of user study results on participants' ratings for each system component's Reliance and Usability. Each question was rated by participants on a 7-point scale, where dark brown represents the strongest negative response, dark green indicates the strongest positive response, and neutral beige marks the midpoint. The number inside each colored bar reflects the count of participants who selected that specific rating.

select and edit benchmarks. As P9 noted, “*For the second round of grading, it performed much better. I think the benchmarks I set made a difference. The metric scores and comments I edited in the benchmarks were definitely important. It's good to have control, especially over key aspects like benchmark references.*”

Overall, participants expressed trust and satisfaction with the LLM’s performance in the benchmarking process, particularly in re-grading and comparing reports. The combination of LLM-powered analysis and instructor-driven benchmark selection proved to be a powerful tool for improving grading consistency and efficiency.

5.4.4 RQ4: Instructors find the quality of the AI-generated summary and feedback helpful in streamlining the feedback process. Participants gave strong positive ratings for the efficiency of the AI-generated summary (**Q13**: mean = 6.33, stdev = 0.778, fig. 7) and feedback (**Q15**: mean = 6.00, stdev = 1.279, fig. 7), along with positive ratings for reliability (**Q12 & Q14**: mean = 5.08, stdev = 0.996, fig. 8) for both processes. Participants appreciated how this simple yet effective feature provided a quick summary of the students’ reports, allowing them to quickly grasp the topic and content. They also found the feedback function particularly useful. As P10 mentioned, “*The feedback function is very helpful, as it not only collects the metrics, scores, and comments I edited but also includes the annotations and highlights I made. In my past experience, I often felt exhausted after writing feedback at the end of a long grading process, especially when I had many bullet points to cover.*” However, P3 noted that while the AI-generated feedback is a good reference, it should not be used directly when providing feedback to students, as it could appear too generic and overly “AI-styled”, making students feel their hard work wasn’t given proper attention.

5.4.5 RQ5: Instructors overall have a very positive impression of the human-AI collaborative workflow in CoGrader, finding it greatly supports the grading process. Overall, participants expressed strong positive attitudes toward the workflow and system usability. Specifically, the workflow’s efficiency was well-regarded (**Q17 & Q19**: mean = 5.92, stdev = 0.996, figs. 7 and 8), with good ratings for workflow reliability (**Q16**: mean = 5.83, stdev = 1.193, fig. 8). The system was also rated as very easy to use (**Q18**: mean = 6.33, stdev = 0.492, fig. 8), and overall satisfaction with the system was high (**Q20**: mean = 6.00, stdev = 0.739, fig. 7). In summary, our user study and analysis suggest that the proposed collaborative workflow and system are helpful, effective, and reliable in most cases. However, some challenges, particularly in LLM stability and ethical considerations, remain to be addressed. In the Discussion section, we will delve deeper into instructors’ perspectives on this human-AI collaborative grading process, potential ethical issues, and design considerations for similar tools.

5.5 Quantitative Analysis of Grading Consistency

To evaluate the internal consistency of instructor grading when using CoGrader, we analyzed the alignment between participants’ grading score distribution and the ground truth grades provided by the course instructor. Ensuring internal consistency is crucial for maintaining fairness and reducing biases introduced by the system—concerns highlighted by instructors during formative studies.

To this end, we employed three widely used statistical metrics: Kendall’s Tau [6], Spearman’s Rank Correlation [75], and Pearson Correlation [74]. These metrics collectively assess consistency from both ranking and scoring perspectives, providing a robust evaluation of whether the distribution of participant-assigned grades aligns with the ground truth. The results for Kendall’s Tau, Spearman’s Rank Correlation, and Pearson Correlation were 0.799, 0.899, and 0.900, respectively. These strong correlations indicate that the distribution of grades assigned by each participant closely aligns with the ground truth, confirming low variation and strong internal consistency within individual graders.

5.5.1 Consistency in Letter Grades. In addition to evaluating internal consistency within each grader, we also examined inter-grader consistency, as instructors highlighted fairness and reducing inter-grader bias also as key priorities. Specifically, we analyzed the consistency of letter grades assigned by different participants against the ground truth, as letter grades are often the formal outcomes communicated to students and central to academic decisions. Assessing their consistency provides practical validation of the system’s reliance. For example, in the case of report 01, the 12 participants assigned 9 A’s, 2 A-’s, and 1 B+, closely reflecting the ground truth score of “A”. A detailed comparison between the letter grades assigned by different participants and the ground truth is shown in Table 1. This consistent alignment across letter grades further demonstrates the system’s robustness and reliability in supporting fair assessment of diverse project reports.

The findings from our evaluation study suggest that CoGrader not only aligns well with traditional grading methods but also paves the way for more efficient, scalable, and reliable grading practices in PBL environments.

5.6 Quantitative Analysis of Instructor Engagement with AI-Generated Outputs

To examine the extent to which instructors critically engaged with AI-generated content (**RQ7**), we conducted a quantitative analysis of participants’ interaction logs during the grading process. Specifically, we analyzed three key behaviors: metric selection, manual comment contributions, and modification of AI-generated feedback.

Participants selected an average of 4.50 Automatic Metrics and 5.83 Reference Metrics per report, indicating a balanced use of system-generated assessment criteria. In addition, they manually added an average of 5.25 personalized comments per report, demonstrating that instructors actively contributed their own evaluations rather than relying solely on AI suggestions.

In terms of verification behavior, 9 out of 12 participants explicitly reported manually checking most AI-referenced facts in the Automatic Metrics against the original student reports. The remaining three manually verified high-impact criteria such as report structure or alignment with project requirements, while skipping low-stakes aspects like grammar or formatting.

When using Reference Metrics, instructors adjusted or overrode an average of 64.09% of the AI-generated scores and comments. All participants reported that they reviewed nearly every AI-suggested comment before making a decision to accept, modify, or discard it.

These findings confirm that instructors engaged with AI-generated outputs in a deliberate and selective manner. Rather than relying on

Statistics	Report ID				
	R01	R02	R03	R04	R05
Avg.score (Ps)	90.9	75.0	74.5	84.1	83.3
Ground truth score	89.1	75.6	71.5	79.3	82.9
Assigned Grades (Ps)	9A 2A- 1B+	1A- 1B+ 5B 5B-	1A- 2B+ 3B 4B- 2C	3A 2A- 5B+ 2B	2A 2A- 7B+ 1B-
Ground truth grade	A	B	B-	B+	A-

Table 1: The summary of grading consistency analysis: (1) Comparison of participants' average scores for each report with the ground truth scores. (2) Comparison of participants' assigned letter grades for each report with the curved ground truth grades.

AI passively, participants exercised consistent judgment in validating and revising AI assistance—highlighting the effectiveness of our system design in promoting instructor authority and supporting balanced use.

6 Discussion

In this paper, we explored the potential of LLMs to assist instructors in grading project reports and developed a tool aimed at supporting this critical learning assessment process. In the following sections, we will discuss how our study contributes to existing research on human-AI collaboration in educational assessment tools. Specifically, we will explore instructors' perspectives, ethical concerns, and the design considerations and implications that can guide the future development of such systems. We will conclude by addressing the limitations of our study and outlining potential directions for future research.

6.1 Instructor Perspectives on Leveraging LLMs for Grading

Our study revealed that instructors generally appreciated the integration of LLMs to streamline the grading process. Participants, such as P6 and P8, emphasized how the use of LLMs helped them brainstorm potential metrics, gain external perspectives on grading, reduce the back-and-forth effort during the grading process, and organize their thoughts to provide more structured feedback. Overall, most participants expressed a strong positive attitude toward using LLMs to assist in the grading process.

However, concerns were raised about the reliability of certain parts of the LLM-generated outputs. For instance, P6, P9, and P11 observed that LLMs sometimes overlooked specific grading metrics or neglected certain report sections. Due to these inherent limitations, it is challenging to guarantee that AI outputs are consistently thorough and comprehensive. Introducing safeguards or providing more flexibility for human calibration could address these shortcomings. As P9 suggested, enabling iterative refinement through a dialogue with the LLM could allow instructors to guide the system in regrading specific sections without a complete reevaluation, improving both accuracy and usability. Ultimately, such enhancements could help foster a more balanced and reliable integration of AI in the grading process.

6.2 Social Risks and Pedagogical Implications of LLM-Assisted Grading

While LLMs offer clear potential to support grading workflows, their integration also raises important social and pedagogical concerns—particularly the risk of eroding personalized feedback, diminishing instructor engagement, and reducing student trust. These concerns reflect broader challenges in maintaining human presence and pedagogical authenticity in AI-assisted grading, and point to the importance of understanding students' perceptions.

A key insight from our study is that AI-supported grading systems should be explicitly designed to enhance—rather than diminish—personalized and actionable feedback. As participant P11 noted, one of the CoGrader's most valuable features was the benchmark-driven workflow, which supported instructors in crafting more thoughtful and targeted feedback. By enabling comparisons with exemplar reports on selected metrics—supplemented by AI-generated scores, comments, and contextual evidence—CoGrader provided a structured yet flexible framework for evaluating student work. This design allowed instructors to offload routine tasks and focus more deeply on higher-order, expertise-intensive evaluation—leveraging AI to strengthen, rather than compromise, the quality of feedback and instruction.

Another important design consideration is the structure and transparency of AI-assisted feedback. To maintain student trust and pedagogical authenticity, instructors emphasized that human-authored comments should remain central and be clearly distinguishable from AI-generated content. In our study, participants responded positively to CoGrader's feedback generation, which prioritized instructor-written comments, followed by instructor-edited AI suggestions, and finally, pure AI-generated comments used primarily to fill remaining gaps. Future systems may benefit from incorporating more explicit visual or textual cues (e.g., labels or icons) to indicate the origin of each comment—especially in high-stakes or formative assessment contexts where transparency and trust are essential.

Students' perspectives further reinforced the importance of thoughtful feedback design in AI-assisted grading. In our informal discussions with students from the course that provided the project report data, many expressed openness to the use of AI, noting its potential to reduce overlooked content, mitigate instructor bias toward certain topics, and accelerate feedback turnaround. At the same time, several students mentioned they could often “tell when the feedback wasn't really written by the teacher”, particularly when it

felt overly generic or formulaic. These insights highlight both the perceived benefits and the importance of addressing the social risks of AI involvement in assessment design—particularly the need to preserve authenticity and transparency to foster trust and support constructive engagement with feedback.

6.3 Design Considerations for calibration of LLM's grading accuracy

A recurring theme in the interviews was the need for more nuanced calibration of LLM grading accuracy. Participants such as P2, P5, and P11 observed that, while benchmarking generally aligned with their grading habits, the system occasionally showed inconsistencies or missed certain grading elements. To address this, participants proposed a multi-level calibration process involving human oversight at the metric, section, and overall report levels.

At the metric level, a targeted regrade function could allow instructors to select specific metrics for re-evaluation. Before initiating regrading, instructors could interact with the LLM, providing comments, restrictions, or instructions to refine its focus on particular metrics.

For section-level calibration, enabling granular grading within specific report sections would offer instructors greater control. As suggested by P11, this approach could improve consistency across submissions by allowing instructors to fine-tune the grading for distinct report sections.

At the overall report level, incorporating a chatbot into the system could facilitate real-time interaction, allowing instructors to pose questions or clarify discrepancies in the LLM's grading approach. This "discussion" feature would be especially valuable during benchmarking, helping to reconcile differences in judgment and reduce personal biases, ultimately supporting greater fairness and accuracy in grading.

6.4 Design Implications for Instructor-AI Collaborative Grading System

Several design implications emerged from our study on how to enhance instructor-AI collaboration in grading. One key takeaway was the importance of integrating multiple functional components into a single, interconnected interface. Instructors appreciated this seamless experience, as highlighted by P10, who valued the integration of comments, PDF annotation, AI grading references, and AI-generated feedback within a unified interface. This streamlined workflow allowed her to avoid toggling between different platforms, enhancing efficiency and reducing cognitive load.

Additionally, P1 and P9 emphasized the importance of allowing instructors to first review the report summary before interacting with LLM-generated scores and comments. This step enables instructors to form their own independent impression of the report, helping to mitigate potential bias introduced by the AI assessments. Such an approach would be instrumental in preserving instructor autonomy and ensuring that the LLM outputs are used as a reference, rather than dictating the grading process. P9 further suggested leveraging the design of benchmarking radar charts to provide more informative and personalized feedback to students. By including personalized comparisons against "role model" reports across multiple metrics, students could gain clearer insights

into how their work aligns with established standards. Integrating visual aids like radar charts into feedback would offer a transparent and easily interpretable representation of performance relative to benchmarks, making feedback more instructive and actionable.

Beyond ensuring trustworthy grading accuracy, CoGrader provides substantial benefits in grading efficiency, a feature that was highly appreciated by the instructors. By automating the generation of quantitative metrics, handling initial evaluations, and enabling benchmark-driven regrading, the system reduces the cognitive effort required from instructors. This allows them to focus more on qualitative assessments that necessitate human expertise. It also frees up time for instructors to engage in more personalized instruction and intervention.

Moreover, the involvement of AI in grading can help mitigate personal biases, ensuring a more consistent and fair evaluation process. This not only addresses the issue of limited instructor and teaching assistant availability but also improves the fairness of grading. As educational demands continue to grow, AI-driven tools like CoGrader have the potential to play a pivotal role in enhancing assessment workflows, making grading more efficient and equitable.

6.5 Limitations and Future Work

While our system and workflow demonstrate strong potential, several limitations emerged during the study. One notable issue, as raised by P11, is the inconsistency of LLM-generated grading—some variations in scores were observed across different attempts. This underscores the need for further research to enhance the stability of LLM outputs, particularly in subjective or complex assessment contexts. Future work could explore more robust human-AI collaboration strategies, such as iterative feedback loops or multi-step calibration processes, to reduce such inconsistencies and improve grading reliability. Secondly, as P9 pointed out, the connection between the original report content and LLM-generated comments could be more explicit, enabling automatic scrolling to the relevant text excerpts for easier feedback verification. Strengthening this link would not only improve transparency but also increase instructors' confidence in the system's outputs. Thirdly, improving the system's ability to handle non-textual components, such as visual elements—is also helpful, particularly in project-based learning contexts where these components are integral to the evaluation.

Another current limitation is that the system does not yet incorporate external materials, such as code files, into its evaluation process. This limits its ability to verify certain types of factual content that rely on supplementary materials beyond the report itself. Incorporating external materials is part of our planned future work, which will expand the system's capacity to support more comprehensive and accurate assessments, particularly in technical domains where code, datasets, and other artifacts play a central role.

Instructors also pointed to other research opportunities, noting that the tool's success in data visualization courses indicates strong potential for adaptation to other disciplines—particularly those with unique grading rubrics or subjective evaluation criteria, such as business presentations or design projects. Future work could further refine the system to accommodate a wider range of assessment

types, enhancing its versatility and applicability across diverse educational contexts.

7 Conclusion

This paper tackled the challenges of grading project reports by integrating large language models (LLMs) into the assessment process. Based on insights from our formative study, we developed a novel workflow combining instructor-AI metric co-design, benchmarking, and AI-assisted feedback. The CoGrader tool implements this workflow to improve grading efficiency and consistency, while ensuring instructors retain control over all judgments. Our evaluation demonstrated that this approach effectively streamlines the grading process, provides valuable support to instructors, and offers students peer-comparative feedback. Future work should explore more fine-grained human calibration of LLM outputs, while maintaining an ethical balance where AI enhances—rather than replaces—human judgment. Our work contributes to advancing human-AI collaboration in educational assessment, offering a scalable approach to the growing complexity of project-based learning, and finally supporting students' learning and improvement.

Acknowledgments

This work is supported by RGC GRF grant 16218724.

References

- [1] Alaa Abd-Alrazaq, Rawan AlSaad, Dari Alhuwail, Arfan Ahmed, Padraig Mark Healy, Syed Latifi, Sarah Aziz, Rafat Damseh, Sadam Alabed Alrazaq, Javaid Sheikh, et al. 2023. Large language models in medical education: opportunities, challenges, and future directions. *JMIR Medical Education* 9, 1 (2023), 148–162.
- [2] Eman A Alasadi and Carlos R Baiz. 2023. Generative AI in education and research: Opportunities, concerns, and solutions. *Journal of Chemical Education* 100, 8 (2023), 2965–2971.
- [3] Sharifah Balkish Albar and Jane Elizabeth Southcott. 2021. Problem and project-based learning through an investigation lesson: Significant gains in creative thinking behaviour within the Australian foundation (preparatory) classroom. *Thinking Skills and Creativity* 41 (2021), 100–118.
- [4] Anabela C Alves, Francisco Moreira, Rui M Sousa, and Rui M Lima. 2009. Teachers' workload in a project-led engineering education approach. In *International Symposium on Innovation and Assessment of Engineering Curricula*, Vol. 14.
- [5] Ian Arawjo, Dongwook Yoon, and François Guimbretière. 2017. Typetalker: A speech synthesis-based multi-modal commenting system. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*. 1970–1981.
- [6] Stephan Arndt, Carolyn Turvey, and Nancy C Andreasen. 1999. Correlating and predicting psychiatric symptom ratings: Spearmans r versus Kendalls tau correlation. *Journal of psychiatric research* 33, 2 (1999), 97–104.
- [7] Stephen Atlas. 2023. ChatGPT for higher education and professional development: A guide to conversational AI. (2023).
- [8] Zied Bahroun, Chiraz Anane, Vian Ahmed, and Andrew Zanca. 2023. Transforming education: A comprehensive review of generative artificial intelligence in educational settings through bibliometric and content analysis. *Sustainability* 15, 17 (2023), 129–143.
- [9] Leopold Bayerlein. 2014. Students' feedback preferences: how do students react to timely and automatically generated assessment feedback? *Assessment & Evaluation in Higher Education* 39, 8 (2014), 916–931.
- [10] Gulbahar Beckett. 2002. Teacher and student evaluations of project-based instruction. *TESL Canada journal* (2002), 52–66.
- [11] William N Bender. 2012. *Project-based learning: Differentiating instruction for the 21st century*. Corwin Press.
- [12] Margherita Bernabei, Silvia Colabianchi, Andrea Falegnami, and Francesco Costantino. 2023. Students' use of large language models in engineering education: A case study on technology acceptance, perceptions, efficacy, and detection chances. *Computers and Education: Artificial Intelligence* 5 (2023), 100172.
- [13] Phyllis C. Blumenfeld, Elliot Soloway, Ronald W. Marx, Joseph S. Krajcik, Mark Guzdial, and Annemarie Palincsar. 1991. Motivating Project-Based Learning: Sustaining the Doing, Supporting the Learning. *Educational Psychologist* 26, 3–4 (1991), 369–398.
- [14] Mirjam Brassler and Jan Dettmers. 2017. How to Enhance Interdisciplinary Competence—Interdisciplinary Problem-Based Learning versus Interdisciplinary Project-Based Learning. *Interdisciplinary Journal of Problem-Based Learning* 11, 2 (2017), 369–398.
- [15] Lyn Brodie and Peter Gibbons. 2009. Comparison of PBL assessment rubrics. In *Proceedings of the Research in Engineering Education Symposium (REES 2009)*.
- [16] Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2024. Benchmarking large language models in retrieval-augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 17754–17762.
- [17] John Chen, Xi Lu, Yuzhou Du, Michael Rejtig, Ruth Bagley, Mike Horn, and Uri Wilensky. 2024. Learning agent-based modeling with LLM companions: Experiences of novices and experts using ChatGPT & NetLogo chat. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–18.
- [18] Shih-Yeh Chen, Chin-Feng Lai, Ying-Hsun Lai, and Yu-Sheng Su. 2022. Effect of project-based learning on development of students' creative thinking. *The International Journal of Electrical Engineering & Education* 59, 3 (2022), 232–250.
- [19] Zixin Chen, Jiachen Wang, Meng Xia, Kent Shigyo, Dingdong Liu, Rong Zhang, and Huamin Qu. 2025. StuGPTViz: A Visual Analytics Approach to Understand Student-ChatGPT Interactions. *IEEE Transactions on Visualization and Computer Graphics* 31, 1 (2025), 908–918.
- [20] Renee M. Clary, Robert F. Brzuszek, and C. Taze Fulford. 2011. Measuring Creativity: A Case Study Probing Rubric Effectiveness for Evaluation of Project-Based Learning Solutions. *Creative Education* 02, 04 (2011), 333–340.
- [21] Derrick Coetzee, Seongtaek Lim, Armando Fox, Bjorn Hartmann, and Marti A Hearst. 2015. Structuring interactions for large-scale synchronous peer learning. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. 1139–1152.
- [22] Taufiq Daryanto, Xiaohan Ding, Lance T Wilhelm, Sophia Stil, Kirk McInnis Knutson, and Eugenia H Rho. 2025. Conversate: Supporting Reflective Learning in Interview Practice Through Interactive Simulation and Dialogic Feedback. *Proceedings of the ACM on Human-Computer Interaction* 9, 1 (2025), 1–32.
- [23] Joost C. F. de Winter, Dimitra Dodou, and Arno H. A. Stienen. 2023. ChatGPT in Education: Empowering Educators through Methods for Recognition and Assessment. *Informatics* 10, 4, 87.
- [24] Haoxiang Fan, Guanzheng Chen, Xingbo Wang, and Zhenhui Peng. 2024. Lesson-Planner: Assisting novice teachers to prepare pedagogy-driven lesson plans with large language models. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 1–20.
- [25] Haoxiang Fan, Changshuang Zhou, Hao Yu, Xueyang Wu, Jiangyu Gu, and Zhenhui Peng. 2025. LitLinker: Supporting the Ideation of Interdisciplinary Contexts with Large Language Models for Teaching Literature in Elementary Schools. In *Proceedings of the 2025 CHI conference on human factors in computing systems*.
- [26] Tom Farrelly and Nick Baker. 2023. Generative Artificial Intelligence: Implications and Considerations for Higher Education Practice. *Education Sciences* 13, 11 (2023), 2227–7102.
- [27] Artemij Fedosejev. 2015. *React.js essentials*. Packt Publishing Ltd.
- [28] Elena L Glassman, Aaron Lin, Carrie J Cai, and Robert C Miller. 2016. Learnersourcing personalized hints. In *Proceedings of the 19th ACM conference on computer-supported cooperative work & social computing*. 1626–1636.
- [29] Miguel Grinberg. 2018. *Flask web development: developing web applications with python*. " O'Reilly Media, Inc.".
- [30] Pengyue Guo, Nadira Saab, Lysanne S Post, and Wilfried Admiraal. 2020. A review of project-based learning in higher education: Student outcomes and measures. *International journal of educational research* 102 (2020), 101586.
- [31] Helia Hashemi, Jason Eisner, Corby Rosset, Benjamin Van Durme, and Chris Kedzie. 2024. LLM-Rubric: A Multidimensional, Calibrated Approach to Automated Evaluation of Natural Language Texts. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 13806–13834.
- [32] Steffan Hooper, Burkhard C Wünsche, Andrew Luxton-Reilly, Paul Denny, and Tony Haorun Feng. 2024. Advancing Automated Assessment Tools—Opportunities for Innovations in Upper-level Computing Courses: A Position Paper. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education* V. 1. 519–525.
- [33] Bihao Hu, Longwei Zheng, Jiayi Zhu, Lishan Ding, Yilei Wang, and Xiaoqing Gu. 2024. Teaching Plan Generation and Evaluation With GPT-4: Unleashing the Potential of LLM in Instructional Design. *IEEE Transactions on Learning Technologies* 17 (2024), 1445–1459.
- [34] Qinjin Jia, Jialin Cui, Haoze Du, Parvez Rashid, Ruijie Xi, Ruochi Li, and Edward Gehringer. 2024. LLM-generated Feedback in Real Classes and Beyond: Perspectives from Students and Instructors. In *Proceedings of the 17th International Conference on Educational Data Mining*. 862–867.
- [35] Hyoungwook Jin, Seonghe Lee, Hyungyu Shin, and Juho Kim. 2024. Teach ai how to code: Using large language models as teachable agents for programming education. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–28.

- [36] Majeed Kazemitaab, Runlong Ye, Xiaoning Wang, Austin Zachary Henley, Paul Denny, Michelle Craig, and Tovi Grossman. 2024. Codeaid: Evaluating a classroom deployment of an llm-based programming assistant that balances student and educator needs. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–20.
- [37] Dimitra Kokotsaki, Victoria Menzies, and Andy Wiggins. 2016. Project-based learning: A review of the literature. *Improving Schools* 19, 3 (2016), 267–277. <https://doi.org/10.1177/1365480216659733> arXiv:<https://doi.org/10.1177/1365480216659733>
- [38] Milan Kostic, Hans Friedrich Witschel, Knut Hinkelmann, and Maja Spahic-Bogdanovic. 2024. LLMs in Automated Essay Evaluation: A Case Study. In *Proceedings of the AAAI Symposium Series*, Vol. 3. 143–147.
- [39] Joseph S Krajcik and Phyllis C Blumenfeld. 2006. *Project-based learning*. na.
- [40] Harsh Kumar, Ilya Musabirov, Mohi Reza, Jiakai Shi, Xinyuan Wang, Joseph Jay Williams, Anastasia Kuzminykh, and Michael Liut. 2024. Guiding Students in Using LLMs in Supported Learning Environments: Effects on Interaction Dynamics, Learner Performance, Confidence, and Trust. *Proceedings of the ACM on Human-Computer Interaction* 8, CSCW2 (2024), 1–30.
- [41] Harsh Kumar, Jonathan Vincentius, Ewan Jordan, and Ashton Anderson. 2025. Human creativity in the age of llms: Randomized experiments on divergent and convergent thinking. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. 1–18.
- [42] George Leckie and Jo-Anne Baird. 2011. Rater effects on essay scoring: A multi-level analysis of severity drift, central tendency, and rater experience. *Journal of Educational Measurement* 48, 4 (2011), 399–418.
- [43] Hao-Ping Lee, Advait Sarkar, Lev Tankelevitch, Ian Drosos, Sean Rintel, Richard Banks, and Nicholas Wilson. 2025. The impact of generative AI on critical thinking: Self-reported reductions in cognitive effort and confidence effects from a survey of knowledge workers. In *Proceedings of the 2025 CHI conference on human factors in computing systems*. 1–22.
- [44] Gilly Leshed, Diego Perez, Jeffrey T Hancock, Dan Cosley, Jeremy Birnholtz, Soyoungh Lee, Poppy L McLeod, and Geri Gay. 2009. Visualizing real-time language-based feedback on teamwork behavior in computer-mediated groups. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 537–546.
- [45] Qingyao Li, Lingyue Fu, Weiming Zhang, Xianyu Chen, Jingwei Yu, Wei Xia, Weinan Zhang, Ruiming Tang, and Yong Yu. 2023. Adapting large language models for education: Foundational capabilities, potentials, and challenges. *arXiv preprint arXiv:2401.08664* (2023).
- [46] Jingxian Liao, Mrinalini Singh, and Hao-Chuan Wang. 2023. Deepthinkingmap: Collaborative video reflection system with graph-based summarizing and commenting. In *Companion Publication of the 2023 Conference on Computer Supported Cooperative Work and Social Computing*. 369–371.
- [47] Ming Liu, Yiling Ren, Lucy Michael Nyagoga, Francis Stonier, Zhongming Wu, and Liang Yu. 2023. Future of education in the era of generative artificial intelligence: Consensus among Chinese scholars on applications of ChatGPT in schools. *Future in Educational Research* 1, 1 (2023), 72–101.
- [48] Ziyi Liu, Zhengze Zhu, Lijun Zhu, Enze Jiang, Xiyun Hu, Kylie A Peppler, and Karthik Ramani. 2024. Classmeta: Designing interactive virtual classmate to promote VR classroom participation. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–17.
- [49] Carl A Maida. 2011. Project-based learning: A critical pedagogy for the twenty-first century. *Policy Futures in Education* 9, 6 (2011), 759–768.
- [50] Alias Masek and Sulaiman Yamin. 2011. The effect of problem based learning on critical thinking ability: a theoretical and empirical review. *International Review of Social Sciences and Humanities* 2, 1 (2011), 215–221.
- [51] Kelly McConvey, Shion Guha, and Anastasia Kuzminykh. 2023. A human-centered review of algorithms in decision-making in higher education. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–15.
- [52] Victoria Menzies, Catherine Hewitt, Dimitra Kokotsaki, Clare Collyer, and Andy Wiggins. 2016. Project Based Learning: evaluation report and executive summary. (2016).
- [53] Jennifer Meyer, Thorben Jansen, Ronja Schiller, Lucas W Liebenow, Marlene Steinbach, Andre Horbach, and Johanna Fleckenstein. 2024. Using LLMs to bring evidence-based feedback into the classroom: AI-generated feedback increases secondary students' text revision, motivation, and positive emotions. *Computers and Education: Artificial Intelligence* 6 (2024), 100199.
- [54] Jesse G Meyer, Ryan J Urbanowicz, Patrick CN Martin, Karen O'Connor, Ruowang Li, Pei-Chen Peng, Tiffany J Bright, Nicholas Tattonetti, Kyoung Jae Won, Graciela Gonzalez-Hernandez, et al. 2023. ChatGPT and large language models in academia: opportunities and challenges. *BioData Mining* 16, 1 (2023), 20.
- [55] Steven Moore, Richard Tong, Anjali Singh, Zitao Liu, Xiangen Hu, Yu Lu, Joleen Liang, Chen Cao, Hassan Khosravi, Paul Denny, et al. 2023. Empowering education with llms-the next-gen interface and content generation. In *International Conference on Artificial Intelligence in Education*. Springer, 32–37.
- [56] Tricia J Ngoon, C Alie Fraser, Ariel S Weingarten, Mira Dontcheva, and Scott Klemmer. 2018. Interactive guidance techniques for improving creative feedback. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–11.
- [57] Linda B Nilson and Claudia J Stanny. 2015. *Specifications grading: Restoring rigor, motivating students, and saving faculty time*. Routledge.
- [58] Andrew M Olney. 2023. Generating Multiple Choice Questions from a Textbook: LLMs Match Human Performance on Most Metrics. In *LLM@ AIED*. 111–128.
- [59] Steve Oney, Christopher Brooks, and Paul Resnick. 2018. Creating guided code explanations with chat. codes. *Proceedings of the ACM on Human-Computer Interaction* 2, CSCW (2018), 1–20.
- [60] OpenAI. 2024. Assistants API: File Search. <https://platform.openai.com/docs/assistants/tools/file-search>. [Online].
- [61] OpenAI. 2024. Assistants API Overview. <https://platform.openai.com/docs/assistants/overview>. [Online].
- [62] OpenAI. 2024. CAPABILITIES: Structured Outputs. <https://platform.openai.com/docs/guides/structured-outputs>. [Online].
- [63] OpenAI. 2024. GPT-4o System Card. <https://openai.com/index/gpt-4o-system-card/>; journal-GPT-4o system card | openai. [Online; Accessed 08-August-2024].
- [64] Sitong Pan, Robin Schmucker, Bernardo Garcia Bulla Bueno, Salome Aguilar Llanes, Fernanda Albo Alarcón, Hangxiao Zhu, Adam Teo, and Meng Xia. 2025. TutorUp: What If Your Students Were Simulated? Training Tutors to Address Engagement Challenges in Online Learning. In *Proceedings of the 2025 CHI conference on human factors in computing systems*.
- [65] Ernesto Panadero and Anders Jonsson. 2020. A critical review of the arguments against the use of rubrics. *Educational Research Review* 30 (2020), 100329.
- [66] Arjun Panickssery, Samuel Bowman, and Shi Feng. 2024. Llm evaluators recognize and favor their own generations. *Advances in Neural Information Processing Systems* 37 (2024), 68772–68802.
- [67] Jungkook Park, Yeong Hoon Park, Suin Kim, and Alice Oh. 2017. Eliph: Effective visualization of code history for peer assessment in programming education. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*. 458–467.
- [68] Amirhossein Rasooli, Hamed Zandi, and Christopher DeLuca. 2018. Reconceptualizing classroom assessment fairness: A systematic meta-ethnography of assessment literature and beyond. *Studies in Educational Evaluation* 56 (2018), 164–181.
- [69] Preerna Ravi, John Masla, Gisella Kakoti, Grace Lin, Emma Anderson, Matt Taylor, Anastasia Ostrowski, Cynthia Breazeal, Eric Klopfer, and Hal Abelson. 2025. Co-designing Large Language Model Tools for Project-Based Learning with K12 Educators. In *Proceedings of the 2025 CHI conference on human factors in computing systems*.
- [70] Steven I Ross, Fernando Martinez, Stephanie Houde, Michael Muller, and Justin D Weisz. 2023. The programmer's assistant: Conversational interaction with a large language model for software development. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 491–514.
- [71] Liam Saliba, Elisa Shioji, Eduardo Oliveira, Shaanan Cohney, and Jianzhong Qi. 2024. Learning with Style: Improving Student Code-Style Through Better Automated Feedback. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V*. 1. 1175–1181.
- [72] Irit Sasson, Itamar Yehuda, and Noam Malkinson. 2018. Fostering the skills of critical thinking and question-posing in a project-based learning environment. *Thinking Skills and Creativity* 29 (2018), 203–212.
- [73] Patricia Schank and Lawrence Hamel. 2004. Collaborative modeling: hiding UML and promoting data examples in NEMO. In *Proceedings of the 2004 ACM conference on Computer supported cooperative work*. 574–577.
- [74] Philip Sedgwick. 2012. Pearson's correlation coefficient. *Bmj* 345 (2012).
- [75] Philip Sedgwick. 2014. Spearman's rank correlation coefficient. *Bmj* 349 (2014).
- [76] Zekai Shao, Siyu Yuan, Lin Gao, Yixuan He, Deqing Yang, and Siming Chen. 2025. Unlocking Scientific Concepts: How Effective Are LLM-Generated Analogies for Student Understanding and Classroom Practice?. In *Proceedings of the 2025 CHI conference on human factors in computing systems*.
- [77] Alexander M Sidorkin. 2024. *Embracing chatbots in higher education: the use of artificial intelligence in teaching, administration, and scholarship*. Taylor & Francis.
- [78] Yishen Song, Qianta Zhu, Huabo Wang, and Qinhuai Zheng. 2024. Automated Essay Scoring and Revising Based on Open-Source Large Language Models. *IEEE Transactions on Learning Technologies* (2024).
- [79] John Stamper, Ruiwei Xiao, and Xinying Hou. 2024. Enhancing llm-based feedback: Insights from intelligent tutoring systems and the learning sciences. In *International Conference on Artificial Intelligence in Education*. Springer, 32–43.
- [80] Xiaohang Tang, Sam Wong, Kevin Pu, Xi Chen, Yalong Yang, and Yan Chen. 2024. VizGroup: An AI-assisted Event-driven System for Collaborative Programming Learning Analytics. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 1–22.
- [81] Shubo Tian, Qiao Jin, Lana Yeganova, Po-Ting Lai, Qingqing Zhu, Xiuying Chen, Yifan Yang, Qingyu Chen, Won Kim, Donald C Comeau, et al. 2024. Opportunities and challenges for ChatGPT and large language models in biomedicine and health. *Briefings in Bioinformatics* 25, 1 (2024), bbad493.
- [82] Sue Timmis, Patricia Broadfoot, Rosamund Sutherland, and Alison Oldfield. 2016. Rethinking assessment in a digital age: Opportunities, challenges and risks. *British*

- Educational Research Journal* 42, 3 (2016), 454–476.
- [83] Andrew Tran, Kenneth Angelikas, Egi Rama, Chiku Okechukwu, David H Smith, and Stephen MacNeil. 2023. Generating multiple choice questions for computing courses using large language models. In *2023 IEEE Frontiers in Education Conference (FIE)*. IEEE, 1–8.
- [84] Meng-Lin Tsai, Chong Wei Ong, and Cheng-Liang Chen. 2023. Exploring the use of large language models (LLMs) in chemical engineering education: Building core course problem models with Chat-GPT. *Education for Chemical Engineers* 44 (2023), 71–95.
- [85] Barbara E Walvoord and Virginia Johnson Anderson. 2011. *Effective grading: A tool for learning and assessment in college*. John Wiley & Sons.
- [86] Xingbo Wang, Haipeng Zeng, Yong Wang, Aoyu Wu, Zhida Sun, Xiaojuan Ma, and Huamin Qu. 2020. Voicecoach: Interactive evidence-based training for voice modulation skills in public speaking. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–12.
- [87] Jeremy Warner, Amy Pavel, Tonya Nguyen, Maneesh Agrawala, and Bjoern Hartmann. 2023. Slidespecs: Automatic and interactive presentation feedback collation. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 695–709.
- [88] Rick Wormeli. 2023. *Fair isn't always equal: Assessment & Grading in the Differentiated Classroom*. Routledge.
- [89] Yu-Chun Grace Yen, Isabelle Yan Pan, Grace Lin, Mingyi Li, Hyoungwook Jin, Mengyi Chen, Haijun Xia, Steven P Dow, et al. 2024. When to Give Feedback: Exploring Tradeoffs in the Timing of Design Feedback. (2024).
- [90] Dongwook Yoon, Nicholas Chen, Bernie Randles, Amy Cheatle, Corinna E Löckenhoff, Steven J Jackson, Abigail Sellen, and François Guimbretière. 2016. Richreview++ deployment of a collaborative multi-modal annotation system for instructor feedback and peer discussion. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*. 195–205.
- [91] Dongwook Yoon and Piotr Mitros. 2015. Multi-modal peer discussion with RichReviews on edX. In *Adjunct Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. 55–56.
- [92] Alvin Yuan, Kurt Luther, Markus Krause, Sophie Isabel Vennix, Steven P Dow, and Bjorn Hartmann. 2016. Almost an expert: The effects of rubrics and expertise on perceived value of crowdsourced design critiques. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*. 1005–1017.
- [93] Gefei Zhang, Shenming Ji, Yiciao Li, Jingwei Tang, Jihong Ding, Meng Xia, Guodao Sun, and Ronghua Liang. 2025. CPVis: Evidence-based Multimodal Learning Analytics for Evaluation in Collaborative Programming. In *Proceedings of the 2025 CHI conference on human factors in computing systems*.
- [94] Liang Zhang, Tianyi Chen, Yue Zong, and Xiaopeng Gao. 2024. A Peer Grading Approach for Open-ended Programming Projects Based on Binary System and Swiss System. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education*. V. 1. 1484–1490.
- [95] Chengbo Zheng, Zeyu Huang, Shuai Ma, and Xiaojuan Ma. 2024. SelfGauge: An Intelligent Tool to Support Student Self-assessment in GenAI-enhanced Project-based Learning. In *Adjunct Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 1–3.
- [96] Chengbo Zheng, Yuheng Wu, Chuhan Shi, Shuai Ma, Jiehui Luo, and Xiaojuan Ma. 2023. Competent but Rigid: Identifying the Gap in Empowering AI to Participate Equally in Group Decision-Making. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–19.