The impact of prompting strategies on LLM-assisted assessment: Evaluative alignment and boundary conditions for student performance

Kai Liang
Luoyang Institute of Science and Technology, China
https://orcid.org/0000-0003-4571-187X

Chuanfeng Li
Luoyang Institute of Science and Technology, China
https://orcid.org/0000-0002-0286-6444

Large language models (LLMs) are increasingly applied in automated code assessment, yet their integration remains constrained by evaluative inconsistency and deficits in instructional specificity. To examine these challenges, this study employed a quasi-experimental design involving 93 novice programmers to investigate how prompting strategies relate to evaluative alignment, pedagogical characteristics and student feedback adoption. Analysis indicated that the few-shot prompting condition was associated with significantly higher scoring alignment with human instructors (90.5% versus 81.4%) and with generated feedback exhibiting greater criteria anchoring and actionability. Mixed-effects models revealed specific boundary conditions: the positive association of few-shot feedback with assignment scores and submission probability was most pronounced for students with moderate prior competence engaging in tasks of moderate complexity. Furthermore, mediation analysis indicated that perceived usefulness, rather than perceived ease of use, functioned as a significant correlate of substantive revision behaviours and subsequent code quality. These findings suggest that deploying LLMs in formative assessment requires context-specific prompt engineering and human-in-the-loop oversight to manage cognitive load, particularly for novice learners.

Implications for practice or policy:

- Educators can enhance artificial intelligence feedback quality by embedding explicit rubric criteria and concrete input-output exemplars into prompts.
- Instructors should mediate LLM outputs for novice programmers to prevent automation bias and manage high cognitive load.
- Institutions should mandate human-in-the-loop oversight protocols for automated grading systems to ensure evaluative fairness and accuracy.
- Assessors must demonstrate the instrumental value of artificial intelligence feedback to students to drive substantive code revision behaviours.

Keywords: evaluative alignment, LLM-based assessment, GPT-4, prompt optimisation, student engagement, case study

Introduction

Timely feedback on coursework assignments has been shown to enhance student motivation and academic performance by reinforcing learning and clarifying misconceptions (Hattie & Timperley, 2007). Furthermore, Nicol and Macfarlane-Dick (2006) emphasised that formative feedback is key to facilitating self-regulated learning (SRL; Zimmerman, 2002), as it encourages students to reflect on their thought processes and make strategic adjustments in their studies. However, delivering such immediate and individualised formative feedback imposes a considerable workload on instructors, a practical challenge that is prevalent in the large-scale higher education classrooms frequently observed across the Asia-Pacific region and globally (Agostini et al., 2024; Hahn et al., 2021). To address this workload constraint while remaining aligned with these foundational assessment principles, the development of automated

evaluation systems has been increasingly adopted (Hahn et al., 2021), as such systems can identify errors, provide targeted advice for improvement and produce provisional grades in an efficient manner (Hahn et al., 2021). Recently, large language models (LLMs) have been increasingly applied within automated code assessment (Pirzado et al., 2024; Savelka et al., 2023). They demonstrate the capability to process programming tasks, generate detailed feedback and adapt to diverse programming languages and contexts without exhaustive retraining. Models such as GPT-4 and LLaMA, which rely on extensive corpora of human-written code and natural language, show comparably high accuracy in evaluating code (Savelka et al., 2023).

However, the integration of LLMs into assessment presents specific challenges (Farazouli et al., 2024). Despite their potential, LLMs demonstrate limitations in accurately assessing advanced tasks due to constrained contextual understanding (Moazzez et al., 2024). Furthermore, their operational opacity raises concerns regarding reliability and fairness in assessment contexts (McGraw et al., 2024). Recent studies have cautioned against the uncritical adoption of LLMs in assessment (Li et al., 2024; Moazzez et al., 2024). Specifically, reviews of LLM behaviour have documented variability in model outputs, indicating that slight variations in prompt formulation may lead to inconsistent differences in evaluation results (Gallegos et al., 2024; Pirzado et al., 2024).

Furthermore, stakeholder perspectives have revealed the multifaceted challenges of implementation (Agostini et al., 2024; Celik et al., 2024). Within higher education, recent reviews have indicated that while instructors acknowledge efficiency gains from artificial intelligence (AI; Li et al., 2024), they have also reported persistent concerns regarding grading accuracy, the consistency of automated criteria and the pedagogical validity of the generated feedback (Nikolic et al., 2024; Wilkinson et al., 2024). From the student perspective, studies on automated feedback systems have revealed that learners' acceptance is contingent upon the perceived pedagogical usefulness of the tool (Amoozadeh et al., 2024; Kim et al., 2024; Zhang et al., 2024). Students' acceptance of machine-generated evaluations remains conditional, necessitating explicit, actionable guidance and human oversight to validate AI feedback and manage cognitive demands (Li & Kim, 2024; Soffer & Cohen, 2024).

These concerns align with a broader line of work in educational technology focusing on technology-enhanced assessment (Hahn et al., 2021; Li et al., 2024). Recent research has conceptualised analytics-enabled frameworks to support formative assessment by combining multimodal data with structured feedback processes (Arthars et al., 2024) and has demonstrated how AI can analyse large volumes of student data to inform pedagogical decision-making (Dann et al., 2024). Furthermore, innovations such as LLMs and automated grading simultaneously challenge established evaluation paradigms (Farazouli et al., 2024) and enable novel assessment practices. Despite the potential of these models, educators and institutions have encountered operational barriers to integrating them into pedagogical workflows (Nikolic et al., 2024; Wilkinson et al., 2024). While research has emphasised the importance of aligning AI tools with specific instructional designs (Agostini et al., 2024; Celik et al., 2024), mitigating the technical and pedagogical risks of unconstrained models remains a primary concern (Gallegos et al., 2024; McGraw et al., 2024). Specifically, there is a need to better understand how explicit design strategies, such as embedding detailed rating scale descriptors into prompts, can address the opacity and inconsistency of automated code evaluation (Li et al., 2024; Pirzado et al., 2024). At the same time, determining how these automated assessments interact with students' prior competence and technology acceptance to shape their engagement presents an ongoing pedagogical challenge (Kim et al., 2024; Soffer & Cohen, 2024).

In response to these challenges, this study aimed to examine the integration conditions of LLM-based automated feedback in programming education. By employing a quasi-experimental design, this research investigated how explicit prompt optimisation relates to *evaluative alignment*, instructional boundary conditions and feedback adoption mechanisms, drawing upon SRL theory (Zimmerman, 2002) and the technology acceptance model (TAM; Davis, 1989) to elucidate the underlying pedagogical and behavioural mechanisms. The goal was to determine the specific learner and task conditions for automated feedback and to understand the feedback adoption mechanisms that translate machine-generated suggestions into substantive code revisions.

This study makes the following key contributions: (a) It contributes to the empirical validation of LLM-driven assessment by examining how rubric-embedded prompt configurations are associated with evaluative alignment and perceived *feedback clarity*; (b) It provides empirical evidence delineating the instructional boundary conditions under which generative feedback systems demonstrate utility, with particular attention to learner prior competence and task complexity; and (c) It offers empirical insights into feedback adoption mechanisms in programming education, clarifying how perceived usefulness relates to substantive revision behaviours and assignment submission.

Literature review

Evaluative alignment and feedback clarity in LLM-based assessment

Recent developments in LLMs such as GPT have enabled advanced natural language understanding and generation across educational contexts, including tutoring and assessment (Savelka et al., 2023). In programming education, Savelka et al. showed that GPT-4 can achieve over 90% accuracy in grading multi-stage programming tasks by analysing both code logic and accompanying comments, demonstrating advantages over traditional tools in semantic understanding. As a result, LLMs are increasingly considered for automated assessment because they can process complex textual and code inputs, potentially bridging the semantic gap left by earlier rule-based systems (Farazouli et al., 2024).

However, limitations remain. While LLMs demonstrate proficiency in interpreting open-ended solutions, their evaluative inconsistency and instructional specificity deficits in high-stakes assessments remain subjects of debate. Recent work by Moazzez et al. (2024) revealed that LLMs exhibit inconsistency in their evaluation results, showing a 15–20% variance in scoring identical submissions across trials. Unlike rule-based systems, LLMs operate with inherent opacity (McGraw et al., 2024), offering limited transparency into their internal decision processes. This opacity has raised concerns about evaluative alignment, defined as the scoring consistency between LLMs and human instructors, and feedback clarity, referring to the extent to which generated guidance is clear and actionable (Pirzado et al., 2024). Although aligning AI evaluations with explicit criteria has been proposed as an approach (Li et al., 2024), empirical evidence remains limited regarding how specific prompt optimisation strategies mitigate these dual challenges to improve evaluative alignment and feedback clarity.

Instructional scaffolding and boundary conditions of automated feedback

Despite these technical concerns regarding reliability, empirical studies have begun to explore how generative models can be integrated into concrete teaching and assessment designs (Agostini et al., 2024; Celik et al., 2024). Research in higher education has reported that ChatGPT-based scaffolding can support project performance, self-efficacy, motivation and creative thinking when embedded into a structured flipped learning approach (Wu & Yu, 2024). Other work has shown that ChatGPT can act as a marking assistant that grades both coding and reflective assignments against rubrics, achieving human-comparable accuracy for higher quality submissions while remaining sensitive to prompt formulation and rubric specification (Li et al., 2024).

At the same time, a recent study by Kim et al. (2024) has drawn attention to the variability of automated feedback's pedagogical impact across different learners, demonstrating that while AI feedback exhibits utility for lower-level cognitive skills, its effect on higher-order skills is more context dependent. Furthermore, research on automated systems has illustrated how learners use automated comments to identify weaknesses, but it stressed the need for teacher guidance to foster critical engagement rather than dependence (Li & Kim, 2024). In programming education, addressing these variations requires instructional scaffolding that calibrates feedback to task complexity and learner characteristics to promote knowledge construction (Vygotsky, 1978). These findings indicate that in LLM-based automated assessment, achieving feedback calibrated to student ability and task constraints remains a significant

challenge. However, systematic research analysing how students' prior competence and task complexity jointly shape the utility of automated feedback remains scarce.

Student perceptions and feedback adoption

The adoption of automated assessment is influenced in part by student perceptions (Soffer & Cohen, 2024). Grounded in the TAM, early research has suggested that while automatic grading systems can enhance learning through immediate feedback (Hahn et al., 2021), students' willingness to engage with such systems is primarily associated with their perceived usefulness and perceived ease of use (Davis, 1989).

The deployment of LLMs, however, has introduced additional perceptual considerations. Amoozadeh et al. (2024) reported that many computer science students expressed distrust towards LLM-generated materials unless these were validated by instructors, reflecting concerns about the alignment between AI-generated evaluations and instructional expectations. At the same time, a tension exists: although LLMs enable personalised feedback at scale, students may experience cognitive overload when confronted with overly detailed or unstructured AI-generated critiques (Zhang et al., 2024). Such cognitive strain, together with concerns about feedback clarity, may be negatively associated with students' perceptions of ease of use (Zhang et al., 2024).

Despite increasing attention to student acceptance of AI-supported systems (Amoozadeh et al., 2024; Zhang et al., 2024), limited empirical research has explicitly examined how students' perceptions of automated feedback translate into observable learning actions. Specifically, it remains unclear how perceived usefulness and perceived ease of use function as predictive variables for students' substantive code revision behaviours, and how these engagement patterns subsequently relate to assignment submission and overall code quality.

The present study

This study addressed the dual challenges of evaluative inconsistency and instructional specificity deficits in automated code assessment through three interconnected dimensions: technical validation, pedagogical mechanisms and behavioural prediction. First, we examined how prompt optimisation related to evaluative alignment (model-human scoring consistency) and students' perceived feedback clarity. Second, drawing on SRL theory (Zimmerman, 2002), we investigated how prior competence and task complexity moderate the pedagogical utility of automated feedback. Third, applying Davis' (1989) TAM, we analysed how perceived usefulness and ease of use predicted substantive revision behaviours, and how these revisions were associated with assignment submission and code quality.

The study sequentially investigated prompt optimisation, instructional boundary conditions and the perceptual drivers of student revision behaviours through three research questions (RQs):

- RQ1: How do prompt optimisation strategies affect the evaluative alignment of LLM-generated scores with human grading and students' perceived feedback clarity?
- RQ2: Under what learner and task conditions does LLM-generated feedback most significantly impact assignment scores and submission probabilities?
- RQ3: How do perceived usefulness and ease of use predict students' revision behaviours, and how do these revisions relate to assignment submission and code quality?

Methodology

Research context and participants

This empirical study was situated in a regional university of engineering, within the School of Computer and Information Engineering. Participants were first-year students taking a 64-hour advanced C

programming course that covered core C syntax, functions, pointers and basic file processing. Students completed 15 programming assignments of varying complexity that linked course concepts to practical problem-solving and introductory software design.

Participants comprised three class sections of software engineering majors (N = 93; 69% male, 31% female). The students were randomly assigned to these sections upon admission by the academic administration, stratified by gender. This ensured a balanced gender ratio within each section. The target course is a compulsory requirement for this major. The participants represent a typical cohort of novice engineering students with comparable academic backgrounds. Their high school programming completion exam scores were relatively homogeneous, with a maximum-minimum range of approximately 15 points. This homogeneity minimised baseline variances in cognitive ability, ensuring that observed differences could be more reliably attributed to the intervention conditions.

The instructional team consisted of one lead lecturer and three teaching assistants. Theoretical instruction was delivered in a combined class format, with the lead lecturer responsible for all theoretical teaching. The three assistants managed the laboratory sessions and assignment grading. All of them underwent calibration training on the assessment rubric to ensure grading standardisation.

Ethical considerations and data privacy

Following departmental ethics approval, all students provided informed consent. Students could opt out of research data collection at any time without academic penalty; students who chose to opt out completed standard assignments with instructor evaluations, but their data were excluded from analysis. To ensure confidentiality, identities were pseudonymised. Only anonymised code submissions and assignment prompts were transmitted via the LLM API, strictly excluding personally identifiable information. All data were stored on encrypted university servers with access restricted to the core research team.

Experimental tools

The automated feedback generation was powered by the OpenAI API. The specific model version, decoding parameters and system-level safeguards utilised during the experiment are detailed in Table 1. These configurations were maintained across all experimental conditions to prioritise deterministic and pedagogically focused outputs. The complete system prompt and exact API payload template are provided in Section S1 of the supplementary material (https://github.com/rockylk/LLM-Prompting-Assessmen/raw/main/Supplementary_Material.pdf).

Table 1
Technical specifications and API configurations for LLM-generated feedback

Parameter category	Configuration details
Model version	GPT4o-2024-05-13
Decoding parameters	Temperature = 0.3; top-p = 0.9; max. output tokens = 3,500
Reproducibility controls	Frequency penalty = 0.0; presence penalty = 0.0; seed = 42 (to ensure output consistency); *n* = 1 (single choice generated)

The system integrated prompt management, scoring logic and dual dashboards: a teacher interface for configuring prompts and a student interface for submitting code and accessing feedback. Each submission triggered a single API call, generating exactly one feedback output and eliminating the need for automated multi-generation filtering. Prior to delivery, a human-in-the-loop verification mechanism enabled instructors to review and manually revise this AI-generated feedback to correct any pedagogical misalignments or inaccuracies.

Instrument development and content validation

The automated assessment rubric served as the primary instrument for evaluating code quality. To ensure content validity, the rubric was constructed through a rigorous modified Delphi technique (Cuhls, 2023) involving a panel of 20 experts, including senior software engineering faculty and industry practitioners. The development process consisted of two iterative rounds. In the first round, experts evaluated an initial pool of seven dimensions derived from international software testing standards (Institute of Electrical and Electronics Engineers, 2008).

To rigorously quantify expert agreement, each dimension was rated on a 4-point relevance scale (from 1 = *not relevant* to 4 = *highly relevant*). The item-level content validity index was computed for each dimension by dividing the number of experts giving a rating of 3 or 4 by the total number of experts. Based on established psychometric guidelines (Polit & Beck, 2006), a strict decision threshold of item-level content validity index ≥ 0.80 was applied. Two dimensions failing to meet this threshold were removed due to their subjectivity and high measurement error. In the second round, the expert panel operationalised the five retained dimensions (correctness, code efficiency, code structure, error handling and innovation) to ensure reliable measurement. Quantitative thresholds (C1, C2) were benchmarked against predefined test cases and instructor reference solutions, while qualitative dimensions (C3–C5) were anchored using expert-approved historical exemplars for each proficiency tier. Prior to formal grading, human evaluators underwent calibration training using these expert-approved anchors to align their interpretative standards. Detailed assessment criteria are provided in Section S2 of the supplementary material.

In addition to the rubric, the survey instruments used to measure perceived clarity and TAM constructs were adapted from established scales. The clarity survey was developed specifically for this study, while the TAM items were adapted from Davis (1989) to fit the programming context. Prior to full deployment, a pilot test was conducted with a small cohort (N = 15) to verify item readability and refine the instrument structure.

Research design

This study employed a quasi-experimental design. Three pre-existing software engineering cohorts were randomly assigned to three distinct assessment conditions. In the zero-shot condition (Group A), prompts included task descriptions, evaluation criteria and formatting rules, but provided no exemplar demonstrations, relying entirely on the model's pre-trained knowledge (Savelka et al., 2023). The few-shot condition (Group B) encompassed all prompt elements of Group A, but following established prompting paradigms (Savelka et al., 2023), added concrete input-output exemplars paired with explicit rationale explaining their alignment with the evaluation rubric. Group C received traditional instructor grading to maintain standard course administration; subsequent analyses focused exclusively on the LLM conditions (Groups A and B) to examine AI feedback mechanics. To validate reliability, instructors reassessed a random 50% subset of submissions from the LLM experimental groups via a unified learning management system. To explicitly illustrate the operational distinctions between the LLM conditions, comprehensive prompt templates featuring actual student code samples are provided in Sections S3 and S4 of the supplementary material.

Data collection

This study employed a convergent parallel mixed-methods design (Plano Clark, 2019) to triangulate objective performance metrics with subjective perceptions. Comprehensive data, including system logs, evaluation scores, questionnaires and qualitative interviews, were collected to address the RQs. A detailed breakdown of all data sources and precise operational definitions by research question is provided in Table 2, with a comprehensive data dictionary available in Section S5 of the supplementary material.

Table 2
Summary of primary quantitative constructs and their operational definitions

Construct or variable	Associated RQ	Operational definition
Scoring alignment	RQ1	Agreement between LLM-generated scores and human instructors' baseline across the five rubric dimensions.
Subjective feedback clarity	RQ1	Self-reported assessment of the generated feedback's clarity, actionability and comprehensibility.
Prior competence	RQ2	Students' standardised high school programming completion exam scores.
Task complexity	RQ2	Cognitive and technical demand of the programming assignment, categorised as low, medium or high.
Assignment-level performance	RQ2, RQ3	Final rubric-based code quality score of the submitted assignment. (Note: Aggregated to student-level mean score for RQ3.)
Assignment submission status	RQ2, RQ3	Binary indicator of successful pre-deadline submission (1 = *yes*, 0 = *no*). (Note: Aggregated to student-level submission rate for RQ3.)
Perceived usefulness (PU)	RQ3	Belief that utilising LLM feedback enhances learning effectiveness and course performance.
Perceived ease of use (PEU)	RQ3	Belief that interpreting and applying LLM feedback requires minimal cognitive and technical effort.
Revision frequency	RQ3	Precise count of substantive code resubmissions made after viewing the AI feedback.

Data coding and reliability

Reliability checks were conducted across objective metrics, subjective surveys and qualitative transcripts. For scoring alignment, a stratified random sample of 50% of the assignment submissions was double-coded by two independent human evaluators against the five rubric dimensions. Inter-rater reliability was assessed using Cohen's kappa, yielding a coefficient of 0.84, indicating high agreement between raters.

For objective content analysis comparing pedagogical features across prompting conditions, a stratified random sample of 60 LLM-generated feedback logs (30 zero-shot, 30 few-shot) was extracted. Two independent researchers coded these logs based on four binary (presence or absence) pedagogical criteria: (a) *criteria anchoring*, defined as explicit reference to the assessment rubric dimensions; (b) actionability, which involves providing specific, step-by-step corrective guidance rather than merely identifying errors; (c) *localisation*, referring to the precise identification of the issue by code line number or specific function name; and (d) *structural clarity*, characterised by organisation into clear sections such as strengths, weaknesses and suggestions. For the complete codebook and detailed scoring examples, see Section S6 of the supplementary material. The inter-rater reliability for this textual coding process was 0.88 (Cohen's Kappa), indicating high agreement, and any initial disagreements were resolved through consensus discussion.

To assess the internal consistency of the subjective measures, reliability analyses were conducted on the post-task clarity survey and the end-of-semester TAM questionnaire. Cronbach's alpha for all subscales exceeded the standard 0.70 threshold, indicating acceptable internal consistency for the self-reported constructs.

For qualitative coding reliability, interview transcripts from six purposively sampled participants (representing divergent PU scores) were analysed via thematic analysis. The unit of coding was defined

as a distinct sentence or conversational turn expressing a specific feedback utilisation strategy (Miles et al., 2014). Two independent coders (the trained teaching assistants) first coded 20% of the transcripts (N = 96) and then developed a codebook comprising three behavioural themes: error-driven fixing (κ = 0.89), structural refactoring (κ = 0.82) and selective revision (κ = 0.83). Overall inter-rater reliability indicated strong agreement (κ = 0.85). The remaining transcripts were double-coded, and discrepancies were resolved through discussion until consensus. A detailed excerpt of the qualitative codebook is provided in Section S7 of the supplementary material.

Data analysis

Quantitative procedures

Data were analysed using SPSS version 26.0. The analysis proceeded in three stages. For RQ1, independent samples t tests compared rubric alignment scores between prompting conditions. Subjective clarity ratings were analysed using repeated-measures analysis of variance. Chi square tests compared the frequencies of pedagogical features from the content analysis. For RQ2, prompting strategy served as an experimental manipulation to alter feedback pedagogical features (Kim & Steiner, 2016). Linear mixed-effects models and generalised linear mixed-effects models predicted continuous assignment scores and binary submission status. Both models incorporated fixed effects for feedback condition, prior competence, task complexity and their two-way interactions. Random intercepts for students and assignments accounted for nested data. Significant interactions were probed via simple slopes analysis (Aiken & West, 1991). Supplementary correlations and one-way analyses of variance explored secondary relationships involving engagement, review duration and cognitive load.

For RQ3, the PROCESS macro (Model 4) tested mediation effects among PU, PEU and revision frequency (Hayes, 2018). Outcome variables were aggregated to the student level (N = 62): overall submission rate and mean code quality score. Indirect effects were estimated using 5,000 bootstrap resamples for 95% confidence intervals. Effect sizes followed conventional benchmarks (Cohen, 2013).

Qualitative procedures

To contextualise quantitative findings, interview data underwent a six-phase thematic analysis (Braun & Clarke, 2006) and qualitative data reduction (Miles et al., 2014) to identify feedback adoption strategies.

Mixed-methods integration

Methodological triangulation (Plano Clark, 2019) integrated both data sources. Statistical patterns from performance scores and system logs were compared with qualitative interview data to examine convergences and divergences. For instance, the statistical association between PU and revision frequency was cross-verified by students' narratives. This mitigated single-method bias and offered a comprehensive perspective.

Findings

RQ1: Evaluative alignment, perceived clarity and pedagogical characteristics

Alignment scores, representing the agreement between LLM-generated and human grades, were calculated for a subset of human-evaluated assignments (N = 466). The few-shot condition was associated with higher evaluative accuracy, with overall alignment increasing from 81.4% to 90.5% ($t(464) = 11.3$, $p < .001$).

A repeated measures ANOVA revealed significant main effects for prompting strategy ($F(1, 61) = 16.71$, $p < .001$, $\eta^2 = .215$) and evaluator origin ($F(1, 61) = 4.23$, $p = .044$, $\eta^2 = .065$), with no significant interaction ($F = 2.05$, $p = .157$, $\eta^2 = .033$). The few-shot condition was associated with higher mean clarity scores (from 3.59, SD = 0.61, to 4.17, SD = 0.47). Synthesising these metrics, few-shot prompting was associated with higher evaluative alignment and perceived clarity than the zero-shot baseline within the present instructional setting.

Qualitatively, review logs indicated human instructors emphasised conceptual frameworks, whereas LLM-generated feedback allocated more comments to granular debugging (see Section S8 of the supplementary material). In addition, LLM-generated outputs included occasional inaccuracies, mostly coded as hallucinations when processing atypical solutions.

Content analysis of 60 sampled feedback logs (Table 3) indicated that few-shot prompts were associated with significantly higher frequencies of pedagogical features across all measured dimensions: criteria anchoring, actionability, localisation and structural clarity.

Table 3
Comparison of pedagogical features in LLM-generated feedback by prompting strategy

Pedagogical feature	Zero-shot (*n* = 30)	Few-shot (*n* = 30)	χ^2	*p* value
Criteria anchoring	7 (23.3%)	28 (93.3%)	30.13	<.001
Actionability	12 (40.0%)	26 (86.7%)	14.02	<.001
Localisation	10 (33.3%)	24 (80.0%)	13.25	<.001
Structural clarity	5 (16.7%)	30 (100%)	42.86	<.001

These content analysis results suggest that few-shot prompting generated feedback with higher actionability, clearer structure and stronger pedagogical alignment. This observed difference in feedback quality provides the operational basis for examining its relationship with assignment scores and submission probability across varying learner and task conditions.

RQ2: Conditional relationships with assignment scores and submission behaviours

Baseline programming proficiency did not differ significantly among the three cohorts ($F(2, 90) = 0.89$, $p = .41$, $\eta^2 = .019$), indicating baseline equivalence (see Figure 1 and Table 4).

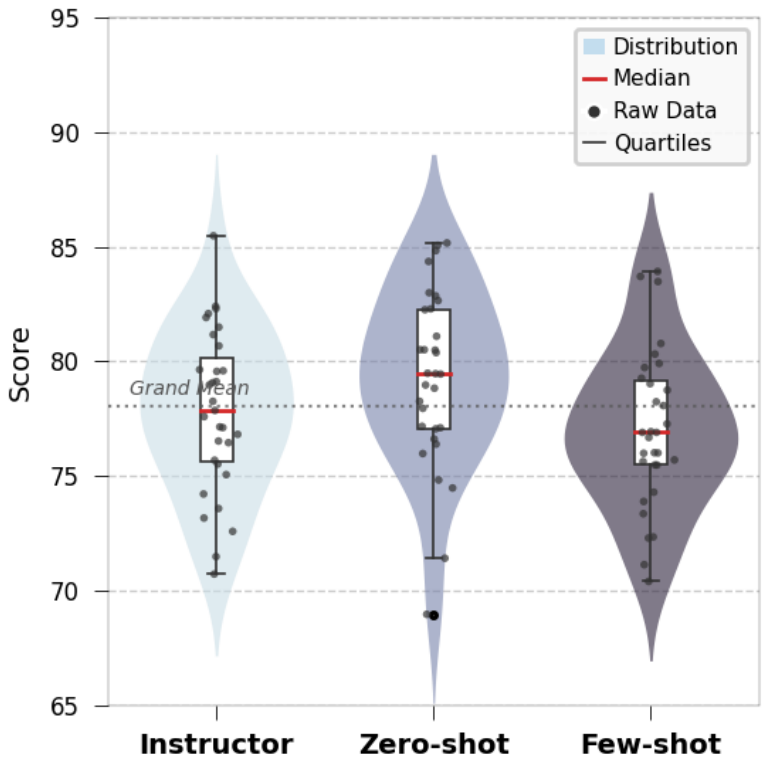


Figure 1. Baseline prior competence by group

Table 4
Analysis of variance for prior programming proficiency

Source of variation	Sum of squares	*df*	*MS*	*F*	*p* value
Between groups	26.70	2	13.35	0.89	.40
Error (within groups)	1350.00	90	15.00		
Total	1376.70	92			

Main and moderation effects: A linear mixed-effects model (LMM) with random intercepts for students and assignments examined relationships among feedback condition, prior competence, task complexity and assignment scores (Figure 2). The model accounted for substantial student-level variance (intraclass correlation coefficient = .34). A significant interaction between feedback condition and prior competence emerged (β = 0.31, *SE* = 0.10, *t* = 3.10, *p* = .002). The positive association of few-shot feedback with assignment scores was pronounced for intermediate students (β = 0.34, *p* = .001) but non-significant for advanced (β = 0.12, *p* = .18) and novice students (β = 0.11, *p* = .25). In addition, the interaction between feedback condition and task complexity was significant (β = 0.28, *SE* = 0.09, *t* = 3.02, *p* = .003). Few-shot feedback corresponded to higher scores for medium-complexity assignments (Δ = 0.45, *p* < .001), with minimal differences for low-complexity (Δ = 0.15, *p* = .07) and high-complexity tasks (Δ = 0.18, *p* = .04).

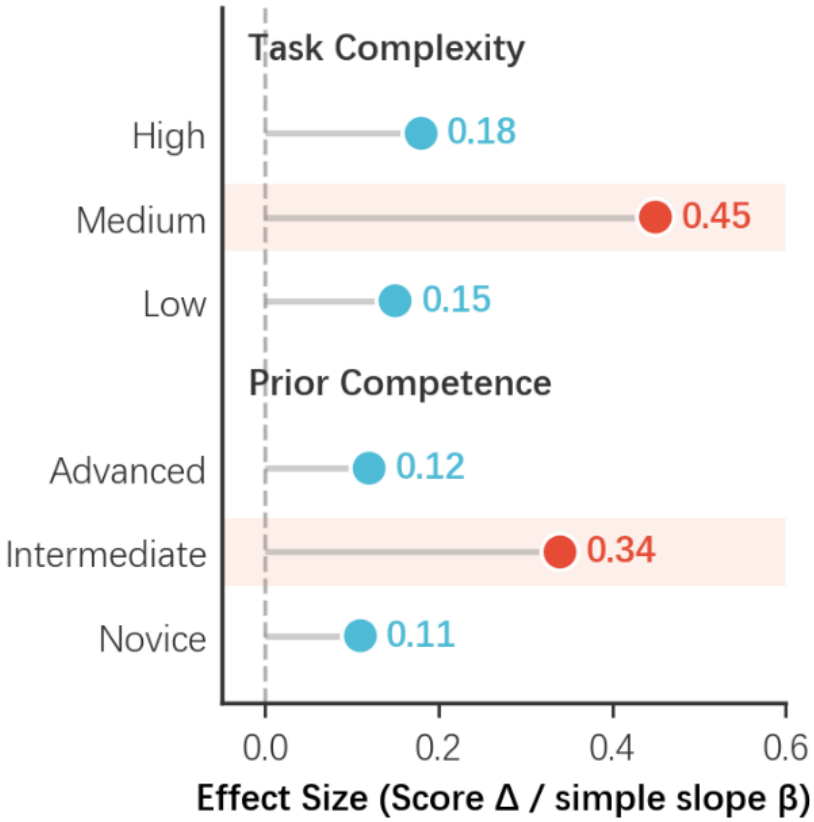


Figure 2. Estimated marginal means of assignment scores across feedback conditions, prior competence and task complexity

Submission and interaction behaviours: The GLMM analysis indicated a significant three-way interaction among feedback condition, prior competence, and task complexity in predicting submission probability (β = 0.75, *p* = .007, odds ratio = 2.12; see Section S9 of the supplementary material). Few-shot feedback was associated with higher submission probabilities for intermediate learners on medium-complexity tasks. Aggregated at the student level, mean assignment scores positively associated with submission rates (*r* = .43, *p* < .001) and interaction frequencies (*r* = .38, *p* < .001), indicating an alignment between assignment outcomes and sustained engagement.

Exploratory sub-analyses: High-ability students typically skimmed feedback, spending less time (*M* = 3m21s, *SD* = 0.45) and reporting lower cognitive load (*M* = 2.5, *SD* = 0.6). In contrast, low-ability students remained on the feedback interface longer (*M* = 7m12s, *SD* = 1.13) and reported higher cognitive load (*M* = 4.1, *SD* = 0.9). These differences were statistically significant ($p < .05$), suggesting novice learners experience greater difficulty in digesting complex LLM suggestions.

Taken together, few-shot feedback was associated with the highest assignment scores and submission probabilities specifically for intermediate learners engaging with medium-complexity tasks. Exploratory sub-analyses indicated these contextual advantages appeared to diminish at ability extremes, likely reflecting informational redundancy for advanced learners or cognitive overload for novices.

RQ3: Mediation of revision behaviours between student perceptions and assignment outcomes

Table 5 illustrates descriptive data concerning revision frequency and code quality, stratified by participants' perception levels regarding PU.

Table 5
Stratification of revision frequency and code quality based on PU levels

PU level	Revision count (*M* ± *SD*)	Code quality score (*M* ± *SD*)
High	3.9 ± 1.1	4.5 ± 0.6
Medium	2.7 ± 0.9	3.7 ± 0.5
Low	1.8 ± 0.9	3.2 ± 0.8

As outlined in Table 6, PU indicated significant positive correlations with both revision frequency and code quality, whereas PEU displayed negligible associations. Mediation analyses further confirmed that substantive revisions formed a significant indirect pathway between PU and both assignment outcomes, whereas PEU demonstrated no significant mediation effects.

Table 6
Summary of bivariate correlations and mediation effects regarding student perceptions and performance

Pathway	Statistical metric	Observed value
Correlation: PU → Revision frequency	*r*	.63 ($p < .001$)
Correlation: PU → Code quality	*r*	.48 ($p < .001$)
Correlation: PEU → Revision frequency	*r*	.19 ($p = .09$)
Correlation: PEU → Code quality	*r*	.13 ($p = .20$)
Mediation: PU → Revisions → Submission	Indirect effect	0.22 (95% CI [0.13, 0.32])
	η^2	.11
Mediation: PU → Revisions → Code quality	Indirect effect	0.15 (95% CI [0.08, 0.23])
	η^2	.08
Mediation: PEU → Revisions → submission	Indirect effect	0.02 (95% CI [-0.05, 0.08])
	η^2	<.01
Mediation: PEU → Revisions → Code quality	Indirect effect	0.01 (95% CI [-0.09, 0.11])
	η^2	<.01

Qualitative evidence from six purposively sampled students further elucidated these statistical patterns. Participants perceiving high PU described an error-driven fixing strategy, observing that the system identified specific logical deficits. Furthermore, many engaged in structural refactoring to enhance modularity, a behaviour aligning with elevated code quality scores (*M* = 4.52, *SD* = 0.49). Students reporting low PU were primarily characterised by selective revision, prioritising superficial formatting over deep corrections due to scepticism regarding AI reliability. These thematic insights appear to corroborate the quantitative mediation results, suggesting that valuing feedback is associated with substantive revisions that correlate with higher output quality within the observed assignments.

Taken together, these findings suggest that PU, rather than PEU, likely emerges as the primary correlate of the metacognitive behaviours associated with assignment submission and code refinement within the present instructional context.

Discussion

Outcomes of prompt refinement

Regarding RQ1, content analysis indicated that the few-shot condition exhibited more pronounced pedagogical characteristics, providing actionable and localised guidance. Our results extend the recent work of Li et al. (2024), who noted that while ChatGPT can function as a marking assistant, its accuracy often fluctuates based on prompt formulation. Our findings suggest that few-shot prompting may help attenuate some of this fluctuation and bring AI output closer to expert rubrics. However, acknowledging the inherent limitations of generative models, maintaining human-in-the-loop oversight remains a standard practical necessity. In this sense, prompt optimisation provides a partial methodological response to reliability concerns, but it does not remove the need for instructor mediation. Furthermore, this pattern is consistent with the framework proposed by Arthars et al. (2024), who argued that analytics-enabled assessment must be carefully designed to support formative goals. We provide empirical support for their conceptual framework by indicating that when LLM feedback is structurally aligned with rubrics, it is more likely to function as an instructional scaffold rather than solely as an error-checking tool within the context of these programming assignments.

From a theoretical perspective, the integration of specific pedagogical features in the few-shot condition supports the potential role of refined prompts within a sociocognitive framework. Our findings suggest that, rather than acting as an uninterpretable evaluator, explicit prompt engineering can clarify the underlying evaluation rationale. By embedding grading criteria into the LLM's generation process, the few-shot prompts appeared to reduce the ambiguity of the reference level (Sadler, 2014). This explicitness may facilitate learners' ability to compare their actual performance with the expected standard, which remains a foundational element of effective formative assessment (Nicol & Macfarlane-Dick, 2006). In this sense, the present findings extend the body of research on automated feedback systems by illustrating how prompt tuning can foreground specific instructional objectives, thereby bridging the gap between computational outputs and pedagogical goals within specific instructional settings (Agostini et al., 2024).

Practically, this study underscores the utility of embedding refined prompting strategies within instructional frameworks, particularly in resource-constrained environments characterised by large-scale classes (Agostini et al., 2024). To generate feasible and contextually beneficial feedback for structured technical assignments, instructors should explicitly embed input-output exemplars and evaluation rationales into their prompt designs. This targeted prompt engineering can mitigate critical oversights and provide more actionable, localised guidance for learners in introductory computer programming courses (Pirzado et al., 2024).

Interplay among prompting strategies, learner proficiency and task complexity

The results of RQ2 offer a nuanced counter narrative to earlier studies suggesting that high-ability learners naturally benefit most from automated systems (Hahn et al., 2021; Orji & Vassileva, 2022). Instead, we observed a conditional pattern where the contextual benefits of prompt optimisation appeared to diminish for both novice and advanced learners. This finding resonates with the interaction patterns observed by Kim et al. (2024), who found that students' interactions with AI tools is heavily dependent on their existing skill levels and attitudes. Our behavioural sub-analyses help clarify the specific boundaries of this interaction: for high-ability students, the detailed few-shot feedback may be informationally redundant, whereas for novices, interpreting complex LLM suggestions may induce cognitive load that exceeds their current processing capacity without additional human mediation.

This pattern is consistent with the perspective on affordances discussed by Celik et al. (2024), who emphasised that the potential of AI tools depends on the user's capacity to leverage them. Our study extends this by identifying the specific boundary conditions of moderate complexity where the utility of refined prompts is most apparent. Moreover, the observed difficulties among novice learners underscore the argument by Nikolic et al. (2024), who posited that the role of academics is shifting from primary evaluators to validators of AI generation. Our exploratory sub-analyses indicated that novice learners experienced significantly higher cognitive load when processing LLM feedback. Human validation may be particularly crucial for these learners, who lack the domain expertise to filter AI hallucinations or overly complex architectural suggestions. Thus, the integration of generative AI feedback should be accompanied by the thoughtful integration strategies advocated by Dann et al. (2024) to ensure it scaffolds rather than overwhelms the learner.

From a theoretical perspective, these interaction effects can be interpreted through the lens of Vygotsky's (1978) zone of proximal development within the context of human-AI interaction. Our results suggest that the detailed feedback generated via few-shot prompts appears to function most appropriately as a scaffold when the gap between the learner's current capability and the task demands lies within their zone of proximal development. Thus, when generative AI feedback is appropriately calibrated through prompt optimisation and aligned with task complexity and student ability, it may help establish conditions where the level of challenge is better matched to learners' capabilities within the observed instructional setting.

Associations between user perceptions and revision behaviours

The findings of RQ3 offer an empirical refinement to the TAM in the context of generative AI. While traditional studies (Davis, 1989; Venkatesh et al., 2003) have often emphasised PEU, our results align with the pedagogical trade-offs observed by Soffer and Cohen (2024). Specifically, students appeared to tolerate the cognitive friction of interpreting complex feedback because they valued its instrumental utility for grade improvement.

This distinction also reflects the critical usage patterns identified by Li and Kim (2024). They observed that self-regulating learners treat automated systems as a critical friend rather than an oracle. Our qualitative data support this. Students with high PU scores engaged in error-driven fixing, which mirrors the critical engagement described by Li and Kim. In contrast, the superficial revisions observed in the low-PU group align with the concerns raised by Wilkinson et al. (2024) regarding academic integrity and the potential for passive reliance on AI. By indicating that PU, rather than PEU, is the primary correlate of substantive engagement, this study suggests that future implementations should prioritise evidencing the tool's learning benefits over further interface simplification within introductory programming contexts.

Interpreting these findings through the lens of SRL theory offers a plausible theoretical explanation for the dominance of PU over PEU. Within the framework of the forethought phase described by Zimmerman (2002), PU functions as a potential proxy for task value and outcome expectations. This theoretical alignment suggests that students may exert volitional control to engage in the cognitively demanding work of revision primarily when they perceive high instrumental value in the feedback. As Panadero (2017) posited, learners are willing to tolerate higher cognitive friction in complex learning tasks provided the pedagogical utility supports their self-regulation goals. Consequently, usefulness rather than ease of use serves as the principal factor associated with meaningful revisions and helps contextualise the motivational processes underpinning AI-assisted formative assessment (Li & Kim, 2024). While minimal trust in AI capabilities was associated with selective and superficial edits, the perception of direct performance benefits appeared to encourage students to allocate significant cognitive resources towards error-driven fixes and structural refactoring. This pattern reinforces the importance of SRL principles and underscores how AI feedback may support learner agency when users recognise its immediate practical advantages within the observed instructional setting (Dignath & Büttner, 2008).

Critical reflections: Balancing efficiency with pedagogical values

Despite the contextual benefits observed in feedback quality and student engagement, our findings must be interpreted in light of broader critical debates regarding AI in education. First, high-fidelity, rubric-aligned feedback may inadvertently encourage automation bias. Novices may accept AI suggestions uncritically rather than engaging in deep reflection (Li & Kim, 2024), risking critical thinking development.

Second, the observation that contextual benefits appeared to diminish for novice learners suggests that uncritical scaling could exacerbate the digital divide. If AI tools primarily benefit students within their zone of proximal development, they may inadvertently disadvantage struggling learners. Moreover, residual hallucinations and grading inconsistencies raise broader fairness concerns in technology-mediated assessment (Gallegos et al., 2024).

Finally, the required human-in-the-loop model introduces new forms of invisible labour. As noted by Nikolic et al. (2024), the expectation for instructors to monitor and explain machine-generated judgments shifts their role from primary evaluators to auditors of AI outputs. Future implementations must balance automation efficiency with professional judgement, ensuring that AI serves as a complement to the pedagogical relationship.

Limitations and future research

This study has several limitations. First, the sample was restricted to first-year students in a single C programming course, which limited control over instructor effects. Future research should employ randomised controlled trials across diverse cohorts and languages to verify broader applicability.

Second, prompt engineering did not entirely eliminate AI hallucinations and grading inconsistencies, which may compromise the perceived fairness of automated assessments. Future work should explore robust verification mechanisms, such as multi-model consensus checking or hybrid scoring pipelines.

Third, the intervention exclusively utilised GPT-4. Future studies should compare alternative proprietary and open-source models to identify cost-effective solutions for educational integration.

Fourth, the system delivered uniform feedback regardless of prior competence. Since novices experienced higher cognitive load, the current design lacks sufficient scaffolding for this subgroup. Future studies should examine adaptive feedback mechanisms and teacher-mediated supports tailored to diverse learner profiles.

Finally, assessing student perceptions cross-sectionally at the semester's end limits causal inference and cannot rule out reverse causality. Future research should employ longitudinal designs to establish temporal precedence and capture real-time perceptual shifts.

Conclusions

This study investigated the pedagogical integration of LLM-generated feedback in programming education. First, few-shot prompting demonstrated closer evaluative alignment with human grading and higher perceived clarity. Second, the utility of optimised prompts peaked for moderate-ability learners engaging with moderately complex tasks. Third, PU emerged as the primary correlate of substantive revision behaviours and assignment outcomes.

Theoretically, these findings position context-specific prompt engineering as a crucial variable in computer-assisted formative assessment. Practically, instructional designs must explicitly embed rubric components and exemplars to anchor the AI's evaluation rationale. To mitigate automation bias among novice learners, institutions must integrate complementary scaffolding, such as teacher-mediated debriefing. Pre-intervention AI-literacy training is essential to equip students to critically validate

machine-generated suggestions. These targeted pedagogical designs align generative AI capabilities with formative goals while maintaining essential human oversight.

Supplementary material

The supplementary materials supporting this study, including detailed assessment rubrics, prompt configurations and qualitative codebooks, are available in the GitHub repository LLM-Prompting-Assessment (https://github.com/rockylk/LLM-Prompting-Assessmen/raw/main/Supplementary_Material.pdf).

Author contributions

Kai Liang: Conceptualization, Methodology, Investigation, Software, Writing – original draft, Writing – review & editing; **Chuanfeng Li:** Formal analysis, Validation, Supervision, Writing – review & editing.

Acknowledgements

The authors would like to extend their sincere gratitude to all participating students for their cooperation and dedication during the study. This study was funded by the Henan Provincial Key Research and Development Program (No. 251111211800), the Henan Provincial Science and Technology Research Program (No. 262102211092), and the Henan Provincial Higher Education Teaching Reform Research and Practice Key Project (No. 2026SJGLX0199).

References

Agostini, D., Lazareva, A., & Picasso, F. (2024). Advancements in technology-enhanced assessment in tertiary education. *Australasian Journal of Educational Technology, 40*(4), 1–7. https://doi.org/10.14742/ajet.10122

Aiken, L. S., & West, S. G. (1991). *Multiple regression: Testing and interpreting interactions*. SAGE Publications.

Amoozadeh, M., Daniels, D., Nam, D., Kumar, A., Chen, S., Hilton, M., Srinivasa Ragavan, S., & Alipour, M. A. (2024). Trust in generative AI among students: An exploratory study. In B. Stephenson, J. A. Stone, L. Battestilli, S. A. Rebelsky, & L. Shoop (Chairs), *Proceedings of the 55th ACM Technical Symposium on Computer Science Education* (Vol. 1, pp. 67–73). Association for Computing Machinery. https://doi.org/10.1145/3626252.3630842

Arthars, N., Thompson, K., Huijser, H., Kickbusch, S., Cunningham, S., Winter, G., Cook, R., & Lockyer, L. (2024). Formative assessment of group work skills: An analytics-enabled conceptual framework. *Australasian Journal of Educational Technology, 40*(4), 139–154. https://doi.org/10.14742/ajet.9444

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology, 3*(2), 77–101. https://doi.org/10.1191/1478088706qp063oa

Celik, I., Gedrimiene, E., Siklander, S., & Muukkonen, H. (2024). The affordances of artificial intelligence-based tools for supporting 21st-century skills: A systematic review of empirical research in higher education. *Australasian Journal of Educational Technology, 40*(3), 19–38. https://doi.org/10.14742/ajet.9069

Cohen, J. (2013). *Statistical power analysis for the behavioral sciences* (2nd ed.). Routledge. https://doi.org/10.4324/9780203771587

Cuhls, K. (2023). The Delphi method: An introduction. In M. Niederberger & O. Renn (Eds.), Delphi methods in the social and health sciences: Concepts, applications and case studies (pp. 3–27). Springer. https://doi.org/10.1007/978-3-658-38862-1_1

Dann, C., Redmond, P., Fanshawe, M., Brown, A., Getenet, S., Shaik, T., Tao, X., Galligan, L., & Li, Y. (2024). Making sense of student feedback and engagement using artificial intelligence. *Australasian Journal of Educational Technology, 40*(3), 58–76. https://doi.org/10.14742/ajet.8903

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly, 13*(3), 319–340. https://doi.org/10.2307/249008

Dignath, C., & Büttner, G. (2008). Components of fostering self-regulated learning among students. A meta-analysis on intervention studies at primary and secondary school level. *Metacognition and Learning, 3*, 231–264. https://doi.org/10.1007/s11409-008-9029-x

Farazouli, A., Cerratto-Pargman, T., Bolander-Laksov, K., & McGrath, C. (2024). Hello GPT! Goodbye home examination? An exploratory study of AI chatbots impact on university teachers' assessment practices. *Assessment & Evaluation in Higher Education, 49*(3), 363–375. https://doi.org/10.1080/02602938.2023.2241676

Gallegos, I. O., Rossi, R. A., Barrow, J., Tanjim, M. M., Kim, S., Dernoncourt, F., Yu, T., Zhang, R. Y., & Ahmed, N. K. (2024). Bias and fairness in large language models: A survey. *Computational Linguistics, 50*(3), 1097–1179. https://doi.org/10.1162/coli_a_00524

Hahn, M. G., Navarro, S. M. B., Valentín, L. D. L. F., & Burgos, D. (2021). A systematic review of the effects of automatic scoring and automatic feedback in educational settings. *IEEE Access*, 9, 108190–108198. https://doi.org/10.1109/ACCESS.2021.3100890

Hattie, J., & Timperley, H. (2007). The power of feedback. *Review of Educational Research*, 77(1), 81–112. https://doi.org/10.3102/003465430298487

Hayes, A. F. (2018). *Introduction to mediation, moderation, and conditional process analysis: A regression-based approach* (2nd ed.). Guilford Publications.

Institute of Electrical and Electronics Engineers. (2008). *IEEE standard for software and system test documentation* (IEEE Std 829-2008). https://doi.org/10.1109/IEEESTD.2008.4578383

Kim, J., Ham, Y., & Lee, S. S. (2024). Differences in student-AI interaction process on a drawing task: Focusing on students' attitude towards AI and the level of drawing skills. *Australasian Journal of Educational Technology, 40*(1), 19–41. https://doi.org/10.14742/ajet.8859

Kim, Y., & Steiner, P. (2016). Quasi-experimental designs for causal inference. *Educational Psychologist*, *51*(3–4), 395–405. https://doi.org/10.1080/00461520.2016.1207177

Li, J., Jangamreddy, N., Bhansali, R., Hisamoto, R., Zaphir, L., Dyda, A., & Glencross, M. (2024). AI-assisted marking: Functionality and limitations of ChatGPT in written assessment evaluation. *Australasian Journal of Educational Technology, 40*(4), 56–72. https://doi.org/10.14742/ajet.9463

Li, L., & Kim, M. (2024). It is like a friend to me: Critical usage of automated feedback systems by self-regulating English learners in higher education. *Australasian Journal of Educational Technology, 40(*1), 1–18. https://doi.org/10.14742/ajet.8821

McGraw, G., Bonett, R., Figueroa, H., & McMahon, K. (2024). 23 security risks in black-box large language model foundation models. *Computer, 57*(4), 160–164. https://doi.org/10.1109/MC.2024.3363250

Miles, M. B., Huberman, A. M., & Saldaña, J. (2014). *Qualitative data analysis: A methods sourcebook* (3rd ed.). SAGE Publications.

Moazzez, A., Barman, A., Liang, S., Katikaneni, V., Nuli, A., Kandala, V., Kamat, K., & Boicu, M. (2024). Assessing the consistency of open-source large language models for algorithm evaluation. In L. A. Balachandran & J. Wang (Eds.), *Proceedings of the 2024 IEEE MIT Undergraduate Research Technology Conference* (pp. 1–5). IEEE. https://doi.org/10.1109/URTC65039.2024.10937513

Nicol, D. J., & Macfarlane-Dick, D. (2006). Formative assessment and self-regulated learning: A model and seven principles of good feedback practice. *Studies in Higher Education*, *31*(2), 199–218. https://doi.org/10.1080/03075070600572090

Nikolic, S., Wentworth, I., Sheridan, L., Moss, S., Duursma, E., Jones, R. A., Ros, M., & Middleton, R. (2024). A systematic literature review of attitudes, intentions and behaviours of teaching academics pertaining to AI and generative AI (GenAI) in higher education: An analysis of GenAI adoption using the UTAUT framework. *Australasian Journal of Educational Technology, 40*(6), 56–75. https://doi.org/10.14742/ajet.9643

Orji, F. A., & Vassileva, J. (2022). Automatic modeling of student characteristics with interaction and physiological data using machine learning: A review. *Frontiers in Artificial Intelligence, 5*, Article 1015660. https://doi.org/10.3389/frai.2022.1015660

Panadero, E. (2017). A review of self-regulated learning: Six models and four directions for research. *Frontiers in Psychology*, *8*, Article 422. https://doi.org/10.3389/fpsyg.2017.00422

Pirzado, F. A., Ahmed, A., Mendoza-Urdiales, R. A., & Terashima-Marin, H. (2024). Navigating the pitfalls: Analyzing the behavior of LLMs as a coding assistant for computer science students—A systematic review of the literature. *IEEE Access*, *12*, 112605–112625. https://doi.org/10.1109/ACCESS.2024.3443621

Plano Clark, V. L. (2019). Meaningful integration within mixed methods studies: Identifying why, what, when, and how. *Contemporary Educational Psychology,* 57, 106–111. https://doi.org/10.1016/j.cedpsych.2019.01.007

Polit, D. F., & Beck, C. T. (2006). The content validity index: Are you sure you know what's being reported? Critique and recommendations. *Research in Nursing & Health, 29*(5), 489–497. https://doi.org/10.1002/nur.20147

Sadler, D. R. (2014). Beyond feedback: Developing student capability in complex appraisal. In S. Hatzipanagos & R. Rochon (Eds.), *Approaches to assessment that enhance learning in higher education* (pp. 45–60). Routledge. https://doi.org/10.4324/9781315872322

Savelka, J., Agarwal, A., An, M., Bogart, C., & Sakr, M. (2023). Thrilled by your progress! Large language models (GPT-4) no longer struggle to pass assessments in higher education programming courses. In K. Fisler, P. Denny, D. Franklin, & M. Hamilton (Eds.), *Proceedings of the 2023 ACM Conference on International Computing Education Research* (Vol. 1, pp. 78–92). Association for Computing Machinery. https://doi.org/10.1145/3568813.3600142

Soffer, T., & Cohen, A. (2024). Privacy versus pedagogy – students' perceptions of using learning analytics in higher education. *Australasian Journal of Educational Technology, 40*(5), 14–30. https://doi.org/10.14742/ajet.9130

Venkatesh, V., Morris, M. G., Davis, G. B., & Davis, F. D. (2003). User acceptance of information technology: Toward a unified view. *MIS Quarterly*, 27(3), 425–478. https://doi.org/10.2307/30036540

Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Harvard University Press.

Wilkinson, C., Oppert, M., & Owen, M. (2024). Investigating academics' attitudes towards ChatGPT: A qualitative study. *Australasian Journal of Educational Technology*, *40*(4), 105–120. https://doi.org/10.14742/ajet.9456

Wu, R., & Yu, Z. (2024). Do AI chatbots improve students learning outcomes? Evidence from a meta-analysis. *British Journal of Educational Technology*, *55*(1), 10–33. https://doi.org/10.1111/bjet.13334

Zhang, Z., Dong, Z., Shi, Y., Price, T., Matsuda, N., & Xu, D. (2024). Students' perceptions and preferences of generative artificial intelligence feedback for programming. *Proceedings of the AAAI Conference on Artificial Intelligence*, *38*(21), 23250–23258. https://doi.org/10.1609/aaai.v38i21.30372

Zimmerman, B. J. (2002). Becoming a self-regulated learner: An overview. Theory into Practice, 41(2), 64–70. https://doi.org/10.1207/s15430421tip4102_2

Corresponding author: Chuanfeng Li, lcf@lit.edu.cn

Copyright: Articles published in the *Australasian Journal of Educational Technology* (AJET) are available under Creative Commons Attribution Non-Commercial No Derivatives Licence (CC BY-NC-ND 4.0). Authors retain copyright in their work and grant AJET right of first publication under CC BY-NC-ND4.0.

Please cite as: Liang, K., & Li, C. (2026). The impact of prompting strategies on LLM-assisted assessment: Evaluative alignment and boundary conditions for student performance. *Australasian Journal of Educational Technology, 42*(5), 42–58. https://doi.org/10.14742/ajet.10733