Auto-Coding Collaborative Dialogue from Classrooms with Open LLMs Using Zero-Short Prompting

Wenting Sun[1] and Jiangyue Liu[2]
[1]Humboldt-Universität zu Berlin, Germany
[2]Soochow University, Jiangsu, China

suwentin@hu-berlin.de
ljy@suda.edu.cn

Abstract: Collaborative Problem Solving (CPS) is a vital 21st-century skill that demands nuanced coordination of cognitive, social, and regulatory processes among learners. In face-to-face classrooms, peer dialogue offers rich data for studying CPS, but manual annotation of such unstructured, oral interaction is labour-intensive and difficult to scale. This study investigates whether open-source Large Language Models (LLMs), including Llama and Qwen, can perform inductive qualitative coding on classroom peer dialogues using zero-shot prompting alone—without fine-tuning or training data. We collected over 210,000 words of dialogue transcripts from 38 student dyads across two CPS tasks at university classrooms: an engineering design activity and a GenAI-supported peer assessment of lesson plans. Through a multi-phase process, we iteratively developed three zero-shot prompting strategies (self-prompting, chain-of-thought prompting, and in-context prompting) via GPT-4o interactions and deployed them across different LLMs via API access. Our findings suggest that in-context prompting consistently yields context-sensitive and theoretically coherent CPS constructs. Chain-of-thought prompting facilitates abstract reasoning but may lead to overgeneralization, while self-prompting demonstrates autonomous logic yet lacks consistency. Expert evaluations using a five-dimensional rubric (clarity, concreteness, objectivity, granularity, specificity) show moderate to high alignment between human and LLM-generated codes, although LLMs tend to overrate clarity and coherence. We further analyse discrepancies between LLMs and academic frameworks such as PISA CPS and ATC21S, and highlight challenges such as terminological drift, low recall, and theoretical misalignment. This work contributes a scalable, human-centered workflow for inductive coding of classroom dialogue and provides ready-to-use prompt templates for educational researchers. The aim of this study is to critically examine whether open-source LLMs can inductively code classroom peer dialogue in collaborative problem-solving tasks, while acknowledging both their potential and limitations in educational practice. We conclude with a dual-pathway strategy for combining practice-oriented, behaviourally grounded constructs with theory-aligned coding schemes, and offer design recommendations for future human-AI collaborative tools in learning analytics and classroom assessment.

Keywords: Collaborative problem solving, Prompt engineering, Inductive coding, Zero-Shot prompting, Peer dialogue

1. Introduction

Collaborative Problem Solving (CPS) is increasingly recognised as a critical 21st-century skill, aligned with global educational priorities (OECD, 2017). As artificial intelligence (AI) technologies become more integrated into classrooms, understanding how students engage in CPS is essential for providing timely pedagogical feedback and designing effective learning environments. However, analysing peer dialogue to identify CPS behaviours remains a methodological bottleneck. Manual annotation is costly, labour-intensive, and time-consuming—especially in face-to-face (F2F) classroom settings where conversations are dynamic, unstructured, and contextually embedded.

Recent advances in large language models (LLMs) have shown promise in automating qualitative coding tasks. Prior studies have explored their use in deductive coding (Ramanathan et al., 2025), rubric-based annotation (Vujinović et al., 2024), and structured online interactions (Pinargote et al., 2024; Na & Feng, 2025). However, few studies have examined how LLMs perform inductive coding on raw, oral peer dialogue in F2F classroom settings, particularly using prompt-only zero-shot strategies without fine-tuning.

This study investigates whether open-source LLMs can meaningfully and reliably support inductive CPS coding in such contexts. We build on prompt engineering frameworks such as CLEAR (Lo, 2023) and TELeR (Santu & Feng, 2023), adapting them to real-world educational data. Specifically, we compare three zero-shot prompting strategies—in-context prompting, self-prompting, and chain-of-thought prompting—to evaluate their effectiveness in generating CPS constructs from classroom dialogue. We further explore the practical implications of these methods for supporting teachers in understanding and responding to students' collaborative learning behaviours.

Pitfalls of integrating LLMs in education. While LLMs offer opportunities for scalable and context-sensitive analysis, it is important to acknowledge the pitfalls of EdTech integration. Over-reliance on automated coding tools risks reducing teachers' professional agency and may lead to unintended consequences in practical

classroom use, as algorithms can oversimplify nuanced student interactions (Harvey et al., 2025). Moreover, LLMs often operate as "black boxes," raising concerns about data privacy, system bias, transparency, reproducibility, and accountability in classroom assessment (Pitts et al., 2025). These risks highlight the need for critical reflection when embedding LLM-based tools into classroom practice.

Our contributions are threefold: We construct a domain-informed, multi-turn prompt chaining structure for inductive coding of CPS behaviours; We compare multiple LLMs in terms of accuracy, interpretability, and construct complexity in educational behaviour recognition tasks; We explore the potential of LLMs as pedagogical expert annotators, capable of generating context-sensitive scaffolding suggestions from raw classroom dialogue.

2. Related Work and Research Questions

This section reviews prior work in four key areas relevant to our study: (1) established frameworks for collaborative problem solving, (2) the use of large language models (LLMs) in educational text annotation and dialogue analysis, (3) prompting strategies for qualitative coding, and (4) human-AI collaboration in educational assessment.

To interpret CPS behaviours, several well-established frameworks have been developed, including the indicator-based mapping by Sun et al. (2022), the PISA CPS framework (OECD, 2017), the ATC21S framework (Hesse et al., 2015), and the CPS Ontology (Andrews-Todd et al., 2023). These frameworks typically distinguish between social and cognitive dimensions, covering facets such as shared understanding, task regulation, negotiation, and communication. While they offer strong theoretical grounding and are widely used in standardized assessments, their abstract constructs and formal terminology often pose challenges when applied to spontaneous, real-time classroom dialogue. In particular, they may lack the linguistic naturalism and contextual adaptability needed to capture the nuanced nature of peer interactions in face-to-face learning environments.

Recent studies have explored the potential of LLMs to support educational annotation tasks. Vujinović et al. (2024) demonstrated that ChatGPT can serve as a pedagogy expert annotator, while Ramanathan et al. (2025) proposed a grounded prompt engineering (GROPROE) approach to translate literature-based codebooks into effective prompts for deductive coding. In collaborative learning contexts, Pinargote et al. (2024) used GPT-3.5 to classify utterances into collaboration metrics, and Na and Feng (2025) enhanced LLMs' contextual understanding by modelling interdependencies between different communicative components in dialogue. Snyder et al. (2024) further applied LLMs to analyse CPS behaviour patterns across performance levels. However, few studies have examined how LLMs perform inductive coding on unstructured, oral peer dialogue in face-to-face classroom settings using prompt-only strategies.

Prompt engineering plays a critical role in shaping LLM outputs, especially in zero-shot or few-shot scenarios. While frameworks such as CLEAR (Lo, 2023) and TELeR (Santu & Feng, 2023) offer abstract guidance, they often require domain-specific adaptation. Ramanathan et al. (2025) iteratively refined chain-of-thought prompts based on prior frameworks, and Snyder et al. (2024) experimented with pattern-based prompting strategies. Na and Feng (2025) leveraged in-context learning to embed human knowledge into prompts. Despite these advances, there is limited empirical evidence comparing different zero-shot prompting strategies for inductive coding of CPS behaviours in authentic classroom dialogue.

Finally, research on human-AI collaboration in educational assessment highlights both the potential and limitations of LLMs. Ramanathan et al. (2025) noted that while LLMs can explain their coding decisions, they may exhibit sycophantic bias in interactive settings. Parfenova et al. (2025) found that human coders outperform LLMs on complex sentences, whereas LLMs perform better on simpler ones. These findings suggest complementary strengths and underscore the need to examine alignment patterns and interpretive gaps between human and machine evaluations. Our study builds on this line of work by focusing on inductive coding from raw, unstructured peer dialogues using prompt-only strategies without fine-tuning, and by comparing the performance of three zero-shot prompting methods—self-prompting, in-context prompting, and chain-of-thought prompting—in real-time classroom settings. We further explore the practical implications of these methods for supporting teachers in understanding and responding to students' collaborative learning behaviours.

Different from prior work, the *aim of this study is to critically examine whether open-source LLMs can inductively code classroom peer dialogue in collaborative problem-solving tasks, while acknowledging both their potential and limitations in educational practice*. This approach emphasizes transparency, scalability, and adaptability for

educational use. Specifically, we investigate the following research questions using data from university classrooms:

RQ1: *How do performances of three different zero-shot prompting strategies (in-context, self-prompting, and chain-of-thought) differ in inductive coding of CPS behaviours in peer oral dialogue from face-to-face classrooms?*

RQ2: *What are the relative advantages and disadvantages of automatic inductive coding methods for improving teachers' practice?*

3. Methods

This study adopts a multi-phase approach to explore the feasibility and effectiveness of using open large language models (LLMs) for inductive qualitative coding of collaborative problem-solving (CPS) behaviours in oral peer dialogues. The overall strategy involves prompt development using small-scale samples, followed by large-scale automated coding using refined prompts, and expert-based evaluation of the generated constructs.

3.1 Data Collection

We collected peer dialogue transcripts from 38 dyad groups across two face-to-face CPS tasks from two university classrooms:

Task A: An engineering-based design challenge (ICMP task)

Task B: A GenAI-supported peer assessment of lesson plans

All dialogues were audio-recorded and manually transcribed, resulting in a total word count of 214,696 words (134,256 from Task A and 80,440 from Task B).

3.2 Prompt Development and Coding Strategy

We adopted a prompt-only zero-shot strategy using open LLMs (e.g., Llama 3.1/3.3/4, Qwen3) via API access. The coding process followed these steps:

Prompt Iteration: Initial prompts were developed using small samples from the dataset. Drawing on frameworks such as CLEAR (Lo, 2023) and TELeR (Santu & Feng, 2023), we iteratively refined the prompts through multi-turn interactions with GPT-4o. This involved adjusting instructions, formatting, and output structure to improve clarity and alignment with inductive coding principles.

Prompt Types: Referring to the prompt strategies used in the study by Liu Y. et al. (2024), we compare three zero-shot coding methods: self-prompt, chain-of-thought, in-context prompt. We compared three prompting strategies to generate the zero-shot prompting. Inspiring by the experience from the article by Dai et al. (2025), several round prompt iterations was conducted.

Scaling Up: Finalized prompts were used to code the full dataset. Due to token window limitations, longer transcripts were segmented before processing. The temperature was set to 0.8 to encourage diverse and exploratory outputs, consistent with prior studies (Zhang et al., 2025; Parfenova et al. 2025).

3.3 Construct Evaluation and Framework Comparison

After coding, CPS constructs were synthesized by GPT-4o and verified by human experts. The evaluation followed the five-dimension rubric from Liu X. et al. (2024), assessing: Clarity, Concreteness, Objectivity, Granularity, Specificity. Each construct was rated on a 0–2 scale with justifications. The resulting codebooks were then compared with established CPS frameworks including Sun et al. (2022), the PISA CPS framework (OECD, 2017), the ATC21S framework (Hesse et al., 2015), and the CPS Ontology (Andrews-Todd et al., 2023) to identify overlaps. A simplified workflow is illustrated in Figure 1.

The code solution to call LLMs API in multi-turn dialogue transcription analysis, 3 prompt templates used in this code solution, one example of the coding results (Llama4+prompt1+ICMP task), one example of the GPT 4o analysis of the coding results (Llama4+prompt1+ICMP task) are available via OSF: https://osf.io/sm5zp/?view_only=1f71b048431a45b9b7118c8b21fe3bdc

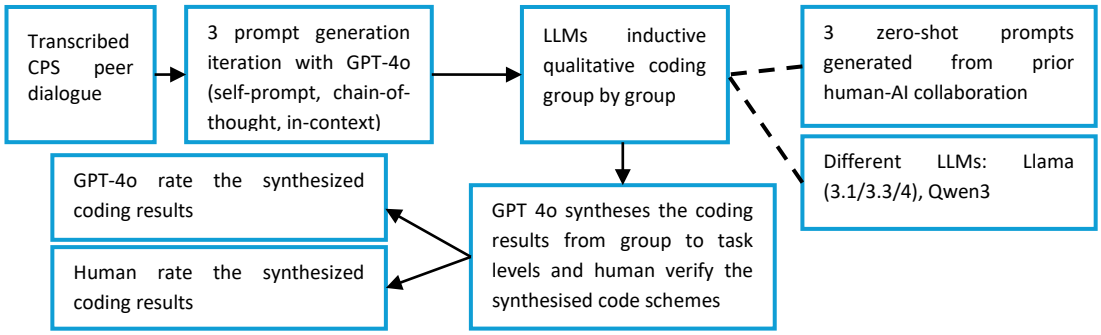

Figure 1: Analysis method

4. Results

For RQ1: *How do performances of three different zero-shot prompting strategies (in-context, self-prompting, and chain-of-thought) differ in inductive coding of CPS behaviours in peer oral dialogue from face-to-face classrooms?*

This section presents a concise synthesis of the outputs generated by large language models (LLMs) across various collaborative problem-solving (CPS) tasks. Rather than providing an in-depth interpretive analysis, the focus here is on summarizing the types and structures of CPS skills identified through inductive coding.

For RQ2: *What are the relative advantages and disadvantages of automatic inductive coding methods for improving teachers' practice?*

Issues related to human verification, theoretical alignment, and interpretive challenges are discussed in detail in the subsequent Discussion section.

Overall, the use of LLM APIs for large-scale inductive coding proved to be both practical and efficient. Across different tasks, the models generated context-sensitive CPS skill constructs, with nuanced variation depending on the prompt design and model architecture. Specifically:

In-context prompting yielded nuanced and context-aware outputs;

Self-prompting showed potential for autonomous reasoning but struggled with consistency;

Chain-of-thought prompting excelled at logical sequencing and abstraction, though it occasionally led to overgeneralization.

Different prompts applied to different LLMs revealed distinct patterns in CPS skill labelling, suggesting that prompt design plays a critical role in shaping the granularity and relevance of the generated constructs.

To facilitate cross-model comparison, frequent CPS skills identified across LLMs were synthesized by human coders into a unified table. This synthesis used less task-specific terminology to enhance generalizability and support subsequent alignment with academic frameworks. To illustrate the variation in CPS constructs generated by large language models (LLMs) under different prompting strategies and task contexts, we initially generated six tables. For brevity, we present two representative examples in the main text: Table 1 (in-context prompting in the ICMP task) and Table 2 (chain-of-thought prompting in the lesson plan task). The remaining tables are available on the OSF link: https://osf.io/sm5zp/?view_only=1f71b048431a45b9b7118c8b21fe3bdc

Table 1: CPS construct induced by LLMs using in-context prompts in ICMP engineering operation task

Dimensions	LLMs generated CPS skills	Academic terms
cognitive	Trial and error + experimental adjustment → Problem-solving ability	Task regulation (ATC21S), Planning and executing (PISA)
	Clarifying issues and division of roles	Task regulation (ATC21S), Planning and executing (PISA)
	Tool usage → Technological proficiency	Task regulation (ATC21S), Planning and executing (PISA), Exploring and understanding (CPS Ontology)

Dimensions	LLMs generated CPS skills	Academic terms
social	Expression and response → Communication competence	Maintaining communication (CPS Ontology), Responding to others' ideas (Sun et al., 2022)
	Knowledge sharing → Collaborative support capability	Establishing shared understanding (CPS Ontology), Participation+Social regulation (ATC21S), Establishing and maintaining shared understanding (PISA)
	Role allocation → Coordination and team management	Establishing and maintaining team organization (PISA), Participation +Social regulation (ATC21S)
	Emotional encouragement → Socio-emotional regulation	Compliments or encourages others (Sun et al., 2022), Participation +Social regulation (ATC21S)
metacognition	Process monitoring → Strategic adjustment skills	Monitoring and reflecting (PISA), Monitoring (CPS Ontology)
	Experience reflection → Self-assessment ability	Monitoring and reflecting (PISA), Task regulation (ATC21S)
	Perseverance/adaptation → Adaptability	Suggests appropriate ideas (Sun et al., 2022), Perspective taking (ATC21S)
	Seeking assistance → Remedial strategy capability	Social regulation (ATC21S), Asking for suggestions (Sun et al., 2022)

Table 2: CPS construct induced by LLMs using Chain-of-thought prompts in GenAI-supported peer assessment of lesson plans

Dimensions	LLMs generated CPS skills	Academic terms
Cognitive	Structured Task Planning	Planning and executing (PISA), Task regulation (ATC21S), Planning (CPS Ontology)
	Technical & Adaptive Integration	Task regulation (ATC21S), Planning and executing (PISA), Exploring and understanding (CPS Ontology)
Social	Dialogue-Based Knowledge Co-Construction	Discusses results (Sun et al., 2022), Participation (ATC21S), Establishing and maintaining shared understanding (PISA)
Metacognitive	Collaborative Goal Regulation	Monitoring and reflecting (PISA), Monitoring (CPS Ontology)

Evaluating the Consistency of Human and LLM Coding

This subsection examines the alignment between human raters and LLMs in evaluating CPS constructs generated through inductive coding. Overall, the results indicate moderate to high alignment between human and machine ratings across evaluated dimensions. However, notable differences were observed in scoring tendencies:

- LLMs tended to assign higher scores to CPS constructs, possibly due to their reliance on surface-level coherence and internal consistency;
- Human raters, by contrast, often drew on their familiarity with the dataset and existing theoretical frameworks, leading to more conservative evaluations—particularly when constructs lacked coverage or theoretical grounding.

The average scores across key dimensions (e.g., clarity, theoretical alignment, transferability) are summarized in **Table 3**.

Table 3: Average scores of evaluated dimensions for each construct

Rater+LLMs	Prompt	Sum	Clarity	Concreteness	Objectivity	Granularity	Specificity
Human+	1	6.9	1.5	1.35	1.4	1.15	1.5
Llama4	2	6.85	1.5	1.5	1.35	1	1.5
	3	7.15	1.4	1.45	1.55	1.25	1.5
GPT-4o+	1	8.2	1.915	1.55	1.55	1.45	1.735
Llama4	2	7.55	1.815	1.385	1.615	1	1.735

Rater+LLMs	Prompt	Sum	Clarity	Concreteness	Objectivity	Granularity	Specificity
	3	7.535	1.79	1.535	1.69	1	1.52
Human+ Llama3.3	1	6.25	1.25	1.25	1.25	1.25	1.25
	2	6.875	1.375	1.375	1.375	1.375	1.375
	3	6.25	1.25	1.25	1.25	1.25	1.25
GPT-4o+ Llama3.3	1	8.96	1.875	1.875	1.875	1.46	1.875
	2	8.975	1.855	1.93	1.93	1.475	1.785
	3	8.395	1.9	1.59	1.715	1.45	1.74
Human+ Llama3.1	1	7.125	1.375	1.375	1.375	1.5	1.5
	2	7.25	1.375	1.375	1.5	1.5	1.5
	3	7.5	1.5	1.5	1.5	1.5	1.5
GPT-4o+ Llama3.1	1	7.625	1.875	1.585	1.29	1.165	1.71
	2	8.75	2	1.835	1.665	1.415	1.835
	3	8.43	1.9	1.735	1.615	1.465	1.715
Human+ Qwen3	1	6.9	1.4	1.4	1.3	1.3	1.5
	2	7.35	1.5	1.5	1.5	1.35	1.5
	3	7.5	1.5	1.5	1.5	1.5	1.5
GPT-4o+ Qwen3	1	8.165	2	1.5	1.5	1.33	1.835
	2	9.17	1.835	1.835	1.835	1.665	2
	3	9.165	2	1.75	1.835	1.665	1.915

Note: the LLMs in the first column refer to the API during the scale inductive coding. Prompt 1 is the final prompt after in-context prompting strategy between human-GPT-4o; Prompt 2 is the final prompt after self-prompting strategy between human-GPT-4o; Prompt 3 is the final prompt after Chain-of-thought prompting strategy between human-GPT-4o.

5. Discussion

While this study identifies several limitations in the use of LLMs for inductive qualitative coding—such as inconsistency, lack of transparency, and weak theoretical grounding—it also reveals important strengths that highlight the practical value of LLMs in educational research and practice.

5.1 Strengths

First, the locally generated "behavior → skill" constructs, grounded in authentic student dialogue (e.g., "you revise the structure, I'll add an example"), offer clarity, concreteness, and immediate usability for instructional design, classroom observation, and formative assessment. Compared to academic CPS frameworks such as PISA (OECD, 2017), ATC21S (Hesse et al., 2015), and the CPS Ontology (Andrews-Todd et al., 2023), these LLM-derived constructs are more accessible to teachers and students, easier to observe and operationalize, and better suited for real-world discourse analysis.

Second, inductive coding using LLM APIs at scale proves to be both efficient and feasible, especially for large datasets. The models are capable of generating context-sensitive CPS skills that reflect the nuances of different tasks. Notably, in-context prompting yields outputs that are linguistically natural and context-aware, particularly in tasks with limited prior information. Chain-of-thought prompting supports logical sequencing and abstraction, though it may occasionally overgeneralize. Self-prompting shows potential for autonomous reasoning but remains inconsistent.

5.2 Challenges

This study reveals several critical limitations and inconsistencies in the use of large language models (LLMs) for qualitative coding in collaborative problem-solving tasks, particularly when applied to educational contexts such as lesson planning and instructional design. Our findings highlight three major challenges: instability across models, lack of transparency in inductive reasoning, and insufficient abstraction and theoretical grounding.

Instability Across Models and Prompts. A key observation is the inconsistency in code schemes generated by different LLMs (e.g., LLama4 vs. Qwen3) when given the same prompt. For instance, under Prompt 1, LLama4 identified cognitive dimensions such as *Problem Identification & Structured Analysis*, *Critical Thinking*, *Lesson Structuring & Instructional Design*, and *Content Integration & Application*. In contrast, Qwen3 produced a different set of dimensions, including *Critical Thinking*, *Cognitive Restructuring*, *Adaptive Problem Solving*, and *Technical Tool Application*. These discrepancies suggest that LLMs lack a stable internal representation of CPS constructs, leading to divergent interpretations even under controlled conditions.

Lack of Transparent Inductive Processes. Despite the use of Chain-of-thought prompting, the inductive reasoning process behind LLM-generated codes remains opaque. It is unclear whether the models rely on semantic similarity, frequency of phrase occurrence, or other heuristics. This lack of transparency undermines the interpretability and replicability of the coding process. This phenomenon also has been reported by the studies by Liu X. et al. (2024) in deductive coding tasks. They claimed that the API usage is less effective at explaining its decisions, identifying ambiguities in definitions compared to Open AI ChatGPT interface.

Insufficient abstraction and theoretical grounding. The generated code schemes often exhibit low abstraction and weak theoretical coherence. Unlike academic CPS frameworks, which feature mutually exclusive and hierarchically structured skills, LLM-generated schemes tend to be fragmented and overlapping. For example, when applying a code scheme derived from the ICMP task to lesson plan tasks, many skills lacked corresponding behavioural samples. Some behaviours specific to lesson planning were not captured in the original CPS framework, indicating poor generalizability and limited transferability of inductively generated codes.

5.3 Pitfalls of Integrating LLMs in Education

Beyond these methodological issues, our findings also underscore broader pitfalls of integrating EdTech in educational settings. The instability and opacity of LLM outputs highlight risks for teachers who might over-rely on machine-generated codes, potentially overlooking the contextual richness of student dialogue. Furthermore, terminological drift and incomplete tagging raise concerns about fairness and validity when such tools are used for high-stakes assessment. These challenges echo earlier critiques of AI in education (e.g., Liu X. et al., 2024; Ramanathan et al., 2025), which caution against assuming that efficiency gains necessarily translate into pedagogical value. A more balanced view requires recognizing both the scalability of LLM-based coding and its limitations in terms of transparency, teacher agency, and ethical responsibility (Harvey et al., 2025; Pitts et al., 2025). In this study, at the operation level, these pitfalls were demonstrated in the following ways:

Terminological Inconsistency and Conceptual Drift. This issue is related to the inconsistency in terminology across tasks and prompts. Constructs such as *Goal Awareness & Task Orientation* appeared in earlier outputs but were later replaced or omitted, with only loosely related constructs like *Goal-oriented Strategic Thinking* emerging in subsequent analyses. This terminological drift leads to the proliferation of similar but non-identical items, complicating efforts to synthesize or compare results across tasks.

Low Recall and Incomplete Tagging in Open Coding. In tasks requiring fine-grained classification (e.g., Prompt 2's segmentation into cognitive, procedural, and affective talk), LLMs often failed to assign tags to each utterance. Instead, they provided summaries without detailing the abstraction process from behaviour to skill. This aligns with prior findings (e.g., Liu X. et al., 2024), which reported that zero-shot LLMs tend to miss many relevant cases, resulting in lower recall compared to human coders or few-shot approaches.

Overlapping Codes and Dimensional Ambiguity in Axial Coding. During axial coding, LLMs frequently generated overlapping items across dimensions. For instance, in the GenAI-assisted lesson plan assessment task, both *AI Evaluation* and *AI-assisted Usage* were categorized under *Technological Mediation Collaboration*, despite representing distinct cognitive operations. This suggests that LLMs struggle to maintain dimensional clarity and often conflate related but separate constructs. This is aligned with the expression about the open coding phase by Parfenova et al. (2025) that categories and clustering codes from open coding phase into higher-order categories in axial coding stage by grouped hierarchically into more abstract categories remains a separate task requiring distinct experiment and evaluation.

Limited Alignment with Established Frameworks. LLMs are unable to autonomously align their outputs with established academic frameworks unless explicitly instructed. Even when provided with a CPS framework and specific task requirements, the alignment results are partial—offering only examples rather than a comprehensive mapping of all generated skills to existing constructs. This partial alignment limits the utility of LLMs in theory-driven qualitative research.

Human-Machine Alignment in Evaluation. Interestingly, human raters tend to assign lower scores than LLMs in evaluating CPS skills. This may be due to human raters' broader familiarity with the dataset and their awareness of missing behaviours or underrepresented skills from authoritative frameworks. This phenomenon echoes findings by Parfenova et al. (2025), who noted that during the open coding phase, human coders tend to overinterpret the data and add unnecessary complexity to straightforward sentences while LLM lake the interpretative depth necessary for complex qualitative analysis.

6. Conclusion and Future Directions

This study provides a comprehensive evaluation of large language models (LLMs) in inductive qualitative coding for collaborative problem-solving (CPS) tasks in educational contexts. While LLMs exhibit notable limitations—including inconsistency across models, lack of transparency in inductive reasoning, and weak alignment with academic frameworks—they also demonstrate significant strengths in scalability, contextual sensitivity, and practical usability.

LLM-generated CPS constructs, grounded in authentic student discourse, offer high clarity and linguistic naturalism, making them immediately applicable for instructional design, classroom observation, and formative assessment. These constructs are particularly accessible to teachers and students, and well-suited for analysing real-world educational interactions. However, they often lack the layered abstraction and definitional precision found in validated CPS frameworks such as PISA, ATC21S, and CPS Ontology.

To address these trade-offs, we propose a **dual-pathway strategy**:

- **Practice-oriented pathway**: Leverage LLMs to generate context-rich, behaviourally grounded constructs that support classroom-based interventions and teacher-student communication.
- **Theory-aligned pathway**: Align LLM outputs with academically validated frameworks to enhance standardization, interoperability, and theoretical rigor in research and assessment.

Future Directions Building on these findings, future research can explore the following areas:

Model-Specific Behaviour Analysis. Investigate how differences in model architecture, training data, and interface (API vs. Chat) influence the generation of CPS constructs, and develop guidelines for model selection based on task type.

Prompt Engineering for Educational Coding. Systematically compare prompting strategies (e.g., in-context, self-prompting, chain-of-thought) to identify optimal configurations for generating reliable and interpretable codes in educational settings.

Human-AI Collaborative Coding Workflows. Design hybrid workflows where human coders guide LLMs through structured alignment with academic frameworks, improving both coverage and conceptual fidelity.

Tool Design for Teachers and Researchers. Develop educational technology tools that embed LLMs with prompt templates, alignment protocols, and visualization features to support teacher-led coding and instructional decision-making.

Cross-Linguistic and Cultural Adaptability. Examine how LLMs perform in multilingual and culturally diverse educational contexts and assess the transferability of locally generated constructs across languages and regions.

By integrating the efficiency and contextual relevance of LLMs with the conceptual depth of academic frameworks, future work can advance both the theory and practice of qualitative analysis in education. At the same time, educators and researchers should remain cautious of potential pitfalls, including over-reliance on automated analysis, diminished teacher agency, and ethical issues surrounding data privacy and fairness. Addressing these risks is essential for developing responsible human-AI collaborative workflows in education.

Limitation. While this study provides valuable insights into the use of large language models (LLMs) for inductive qualitative coding in educational CPS tasks, several limitations should be noted. The analysis was based on a limited set of models and tasks, which may constrain the generalizability of the findings. Human synthesis and interpretation were involved in evaluating LLM outputs, introducing potential subjectivity. Moreover, the study focused on two specific educational contexts, and the applicability of results to other domains remains to be tested. Finally, while the practical utility of LLM-generated constructs is evident, their integration into classroom practice raises broader questions about transparency, teacher agency, and ethical use, which warrant further exploration.

Ethics Declaration: Ethical clearance was not required for this study.

AI Declaration: AI tools (specifically ChatGPT by OpenAI) were used during the writing process to assist with grammar refinement and language polishing. All research design, data analysis, and interpretation were conducted solely by the authors.

References

Andrews-Todd, J., Jiang, Y., Steinberg, J., Pugh, S. L. & D'Mello, S. K. (2023) 'Investigating collaborative problem solving skills and outcomes across computer-based tasks', Computers & Education, 207, 104928.

Harvey, E., Koenecke, A., & Kizilcec, R. F. (2025, April). '" Don't Forget the Teachers": Towards an Educator-Centered Understanding of Harms from Large Language Models in Education'. Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (pp. 1-19).

Hesse, F., Care, E., Buder, J., Sassenberg, K. & Griffin, P. (2015) 'A framework for teachable collaborative problem solving skills', in Griffin, P. & Care, E. (eds.) Assessment and teaching of 21st century skills. Dordrecht: Springer, pp. 37–56. https://doi.org/10.1007/978-94-017-9395-7_2

Liu, X., Zhang, J., Barany, A., Pankiewicz, M. & Baker, R. S. (2024, November) 'Assessing the Potential and Limits of Large Language Models in Qualitative Coding', International Conference on Quantitative Ethnography, pp. 89–103. Cham: Springer Nature Switzerland.

Liu, Y., Tao, S., Meng, W., Wang, J., Ma, W., Chen, Y. et al. (2024, April) 'Interpretable online log analysis using large language models with prompt strategies', Proceedings of the 32nd IEEE/ACM International Conference on Program Comprehension, pp. 35–46.

Lo, L. S. (2023) 'The CLEAR path: A framework for enhancing information literacy through prompt engineering', The Journal of Academic Librarianship, 49(4), 102720.

Na, Y. & Feng, S. (2025) 'LLM-Assisted Automated Deductive Coding of Dialogue Data: Leveraging Dialogue-Specific Characteristics to Enhance Contextual Understanding', arXiv preprint, arXiv:2504.19734.

OECD. (2017) PISA 2015 collaborative problem-solving framework. Available at: https://www.oecd.org/content/dam/oecd/en/publications/reports/2017/08/pisa-2015-assessment-and-analytical-framework_g1g81b0f/9789264281820-en.pdf

Parfenova, A., Marfurt, A., Denzler, A. & Pfeffer, J. (2025) 'Text annotation via inductive coding: Comparing human experts to LLMs in qualitative data analysis', Proceedings of the 2025 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), pp. 6471–6484. https://aclanthology.org/2025.findings-naacl.361/

Pinargote, A., Calderón, E., Cevallos, K., Carrillo, G., Chiluiza, K. & Echeverria, V. (2024) 'Automating data narratives in learning analytics dashboards using GenAI', Data Storytelling and Learning Analytics Workshop 2024, pp. 150–161. CEUR-WS.

Pitts, G., Marcus, V., & Motamedi, S. (2025). Student Perspectives on the Benefits and Risks of AI in Education, arXiv preprint, arXiv:2505.02198.

Ramanathan, S., Lim, L. A., Mottaghi, N. R. & Buckingham Shum, S. (2025, March) 'When the prompt becomes the codebook: Grounded Prompt Engineering (GROPROE) and its application to belonging analytics', Proceedings of the 15th International Learning Analytics and Knowledge Conference, pp. 713–725.

Santu, S. K. K. & Feng, D. (2023) 'Teler: A general taxonomy of LLM prompts for benchmarking complex tasks', arXiv preprint, arXiv:2305.11430.

Snyder, C., Hutchins, N. M., Cohn, C., Fonteles, J. H. & Biswas, G. (2024, March) 'Analyzing students collaborative problem-solving behaviors in synergistic STEM+C learning', Proceedings of the 14th Learning Analytics and Knowledge Conference, pp. 540–550.

Sun, C., Shute, V. J., Stewart, A. E., Beck-White, Q., Reinhardt, C. R., Zhou, G. et al. (2022) 'The relationship between collaborative problem solving behaviors and solution outcomes in a game-based learning environment', Computers in Human Behavior, 128, 107120.

Vujinović, A., Luburić, N., Slivka, J. & Kovačević, A. (2024) 'Using ChatGPT to annotate a dataset: A case study in intelligent tutoring systems', Machine Learning with Applications, 16, 100557.

Zhang, H., Wu, C., Xie, J., Rubino, F., Graver, S., Cai, J., Kim, C. & Carroll, J. (2025) 'Exploring inductive and deductive qualitative coding with AI: Investigating inter-rater reliability between large language model and human coders', in Ahram, T. Z., Karwowski, W. & Rau, P.-L. (eds.) Human-Computer Interaction & Emerging Technologies, Vol. 195. AHFE International. https://doi.org/10.54941/ahfe1006232