JOURNAL OF LEARNING ANALYTICS

Early Access Articles, 1–24. https://doi.org/10.18608/jla.2026.9025

Using Large Language Models for Automated Coding of Self-Regulated Learning Think-Aloud Protocol Data

Sirui Ren[1], Ha Nguyen[2], Matthew L. Bernacki[3], Linyu Yu[4] and Jeffrey A. Greene[5]

Abstract

Documenting and understanding self-regulated learning (SRL) processes can inform the design of learning activities and scaffolds to enhance student success in STEM. Think-aloud protocols (TAPs)—prompting students to verbalize thoughts during task performance—reveal real-time, ecologically sensitive verbalizations of students' SRL processes such as planning, monitoring, and strategy use. However, coding TAP data requires substantial resources. We investigated the capabilities of Large Language Models (LLMs) in automating the coding of SRL processes in TAPs across undergraduate STEM courses. To examine how task features, prompt engineering strategies, and SRL codes are linked to LLMs' coding accuracy, we used a factorial design comparing different LLMs (GPT-4o, Claude 3.5 Sonnet, and Gemini 1.5 Pro) across six SRL TAP codes (representing distinct cognitive and metacognitive processes), six prompt conditions (varying few-shots and context levels), and two STEM tasks (mathematics and biology). Analysis of 600 student verbalizations revealed that mathematics tasks yielded significantly higher classification accuracy compared to biology tasks, with GPT-4o and Claude 3.5 Sonnet outperforming Gemini 1.5 Pro. Few-shot prompting showed code-specific effects, with descriptively higher accuracy for monitoring negative judgments of learning but significantly decreased accuracy for subgoal setting. The addition of contextual information showed minimal impact across tasks. Cognitive process TAP codes (e.g., mathematical problem-solving) demonstrated the most consistent cross-task classification accuracy. In contrast, metacognitive monitoring (e.g., judgment of learning) showed substantial task-dependent variations. These findings highlighted both the promise and limitations of LLMs for scaling SRL research. They suggest that theoretical alignment in prompt engineering is essential for effective automated coding of dynamic regulatory processes.

Notes for Practice

- LLMs demonstrate promising accuracy in coding certain self-regulated learning (SRL) processes in Think Aloud Protocols, particularly for cognitive processes in mathematics contexts. Automated coding of SRL can be integrated into learning analytics pipelines for real-time SRL detection, scaffolding, and validation of learning processes.
- Few-shot prompting shows code-specific effects, with a trend toward improved monitoring of negative judgments of learning but significantly decreased accuracy for planning processes like sub-goal setting.
- Metacognitive monitoring shows substantial task-dependent variation in mathematics and biology, requiring task-specific prompt engineering approaches when implementing in different STEM contexts.
- GPT-4o and Claude 3.5 Sonnet outperform Gemini 1.5 Pro for SRL coding tasks and demonstrate more conservative prediction patterns with fewer false positives.

Keywords: Large language models, self-regulated learning, think-aloud protocols, automated coding, STEM education, prompt engineering

Submitted: 01/05/2025 — **Accepted:** 07/04/2026 — **Published:** 25/06/2026

Corresponding author [1]Email: rensirui@unc.edu Address: The University of North Carolina at Chapel Hill, CB 3500 Peabody Hall, Chapel Hill, NC 27599-3500, USA. ORCID iD: https://orcid.org/0009-0008-8480-8459
[2]Email: ha.nguyen@unc.edu Address: The University of North Carolina at Chapel Hill, CB 3500 Peabody Hall, Chapel Hill, NC 27599-3500, USA. ORCID iD: https://orcid.org/0000-0001-7138-1427

[3]Email: mlb@unc.edu Address: The University of North Carolina at Chapel Hill, CB 3500 Peabody Hall, Chapel Hill, NC 27599-3500, USA; Brain Motivation Research Institute, Korea University, 145 Anam-ro, Seongbuk-gu, Seoul 02841, Korea. ORCID iD: https://orcid.org/0000-0003-1279-2829
[4]Email: yu.4610@osu.edu Address: The Ohio State University, 1958 Neil Avenue, Columbus, OH 43210, USA. ORCID iD: https://orcid.org/0000-0002-2021-8811
[5]Email: jagreene@email.unc.edu Address: The University of North Carolina at Chapel Hill, CB 3500 Peabody Hall, Chapel Hill, NC 27599-3500, USA. ORCID iD: https://orcid.org/0000-0003-4145-1847

1. Introduction

The alarming attrition rate of 40–60% among undergraduates pursuing science, technology, engineering, and mathematics (STEM) degrees represents a persistent educational challenge with significant economic implications (Chen, 2013; National Academies of Sciences, Engineering, and Medicine, 2024). Self-regulated learning (SRL)—defined as the proactive cognitive, motivational, and behavioral processes through which learners systematically pursue academic goals using planning, monitoring, and strategic adaptation—is crucial for student persistence (Blackmore et al., 2021) and success in challenging STEM curricula (Greene et al., 2024; Park et al., 2018). What makes SRL particularly complex is its fundamentally dynamic nature, with learners implementing sophisticated IF-THEN decision rules that are influenced by situational demands and then guide selection of appropriate strategies. These decisions occur repeatedly, often on a second-to-second timeframe throughout task completion (Winne & Hadwin, 1998). These conditional relationships manifest differently across tasks, academic domains, and sociocultural settings (Ben-Eliyahu & Bernacki, 2015). Documenting and understanding the dynamic and context-dependent nature of SRL across learning tasks are key to supporting student success in STEM. Such understanding enables educators to develop precise, context-sensitive scaffolding that addresses specific self-regulatory challenges students encounter in rigorous STEM coursework. To better understand the dynamic and context-dependent nature of SRL, researchers have used think-aloud protocols (TAPs; Greene et al., 2018)—prompting learners to verbalize their thoughts during task performance—to collect valid, ecologically sensitive data about SRL processes as they unfold in real time (Binbasaran Tuysuzoglu & Greene, 2015; Greene et al., 2015). Student verbalizations serve as a grounding for insights about learning processes that are associated with outcomes, and learning interventions can promote verbalizations when other learners encounter similar circumstances. For example, TAP studies have identified specific regulatory deficits such as students' failure to adapt strategies after negative judgments of learning (Binbasaran Tuysuzoglu & Greene, 2015) or inadequate task definition and planning that undermines subsequent learning processes (Greene et al., 2012). These findings directly inform when and how to provide metacognitive scaffolding, enabling interventions that target precise moments of regulatory deficiencies.

However, collecting and preparing TAP data requires substantial resources that limit scalability for large-scale research. Coding TAP data involves labor-intensive and time-consuming steps including transcription, segmentation, and classification using theory-aligned schemes (Greene et al., 2013). The only way to continue refining theories that test the complex conditional relationships in SRL is by solving this methodological bottleneck to achieve scaled methods of fine-grained process data collection and coding. Recent methodological investigations in educational research have demonstrated that large language models (LLMs) can substantially reduce manual coding workload and yet maintain high reliability (Tai et al., 2024), positioning these models as promising for addressing the resource challenges of coding TAP data. To optimize LLM performance for specific tasks, researchers have developed diverse prompt engineering strategies including structural reasoning techniques like chain-of-thought prompting (i.e., prompting models to explain their thinking step by step; Kojima et al., 2022), example-based approaches such as few-shot learning (i.e., providing examples; Brown et al., 2020), and role specification through expert persona adoption (Gao, 2023). However, these strategies have primarily emerged from computer science research (e.g., Brown et al., 2020; Kojima et al., 2022) without sufficient incorporation of established learning theories to guide prompt development. To address these limitations, we examined how LLMs can identify self-regulatory processes that manifest at two distinct levels of context (Figure 1). We conceptualize these levels as two concentric rings of contextual influence: the innermost ring represents the fundamental cognitive *task level* that remains consistent across STEM courses, including essential problem-solving processes, analytical reasoning, and core SRL strategies that students employ regardless of learning domain. The outer ring encompasses *course level* factors, combining instructional design elements with content coverage that creates different learning contexts across domains.

Building on this multi-level conceptual framework, we evaluated the capability of theory-driven prompt engineering approaches to accurately classify frequently enacted SRL processes in TAPs across both biology and mathematics contexts. By "theory-driven", we specify that our prompt design and evaluation aligned with SRL theories in accounting for task and course contexts (Efklides, 2011; Zimmerman, 2000), drawing from theoretically grounded codebooks, and incorporating conversational history (i.e., beyond single utterances) to reflect the dynamic nature of SRL. Specifically, to communicate these layered contextual nuances in our LLM prompts, we drew from a comprehensive TAP codebook that has evolved through multiple research iterations across diverse learning contexts and could be applied across many academic tasks (see Bernacki et al., 2025 for the codebook details; codebook was originally developed by Greene & Azevedo, 2009). This codebook has undergone systematic refinement as it was applied to each new domain-specific task, with each implementation contributing

task-specific micro-level SRL codes that complement the core framework. Thus, this codebook aligns well with our interest in both task- and course-level factors. Through this investigation, we addressed three research aims: (1) We examined how core task features implemented within different STEM course designs (i.e., specifically flipped mathematics and high-structure biology) influenced LLM coding accuracy for six key SRL behaviors in student verbalizations, seeking to identify how task-specific elements affected automated classification of SRL processes. (2) We investigated how the effectiveness of different prompt engineering strategies varied across these task implementations (i.e., distinct instructional designs and academic tasks) and across SRL codes, comparing zero-shot versus few-shot approaches and assessing whether including varying levels of contextual information (i.e., no context, task-aligned, and course-aligned) meaningfully impacted classification accuracy. (3) We analyzed how the inherent nature of different SRL processes (i.e., the core cognitive or metacognitive processes that distinguished various self-regulatory behaviors) influenced LLM coding accuracy across tasks, particularly comparing processes with standardized verbal expressions with those more dependent on task-specific language. Through this systematic investigation, we aim to advance methodological approaches for SRL assessment that combine theoretical rigor with computational efficiency, ultimately enhancing the capacity to study SRL at scale across diverse educational contexts.

Automating SRL coding addresses a fundamental learning analytics challenge: scaling fine-grained analysis from laboratory studies to large-scale implementation. Fine-grained analysis of rich data, such as TAPs, is necessary to uncover both the qualities and quantities of SRL processing that occur on a moment-to-moment basis (Bernacki, 2018; Greene & Azevedo, 2009). However, such analysis is not feasible with large samples or in real-time, thus a process is needed to move from informative but resource-intensive TAP coding to scalable but high-inference digital trace data. Achieving this translation requires four steps. First, learning theory must align with learning design so that observable behaviors reflect theoretical constructs (Giannakos & Cukurova, 2023; Lockyer et al., 2013). Second, TAP coding schemas must translate these constructs (e.g., abstract concepts like metacognitive monitoring or strategic planning) into terms learners actually use when verbalizing their thinking (Greene et al., 2018). Third, human and AI coders must reliably apply these codes to student verbalizations (Zhang et al., 2024). Fourth, verbalized SRL processes must consistently co-occur with multimodal data streams to enable large-scale learning analytics in real-world contexts (Bernacki et al., 2025; Fan et al., 2022, 2023). Each step depends on the previous one, and this study addresses the third step, reliable application, by testing whether LLMs can reliably code SRL processes using established, theoretically grounded codebooks. If sufficient reliability can be reached, then LLM coding can be integrated into learning analytics pipelines for real-time SRL detection, scaffolding, and validation of learning processes combining think-aloud and multimodal data (Khalil et al., 2023; Lim et al., 2023).

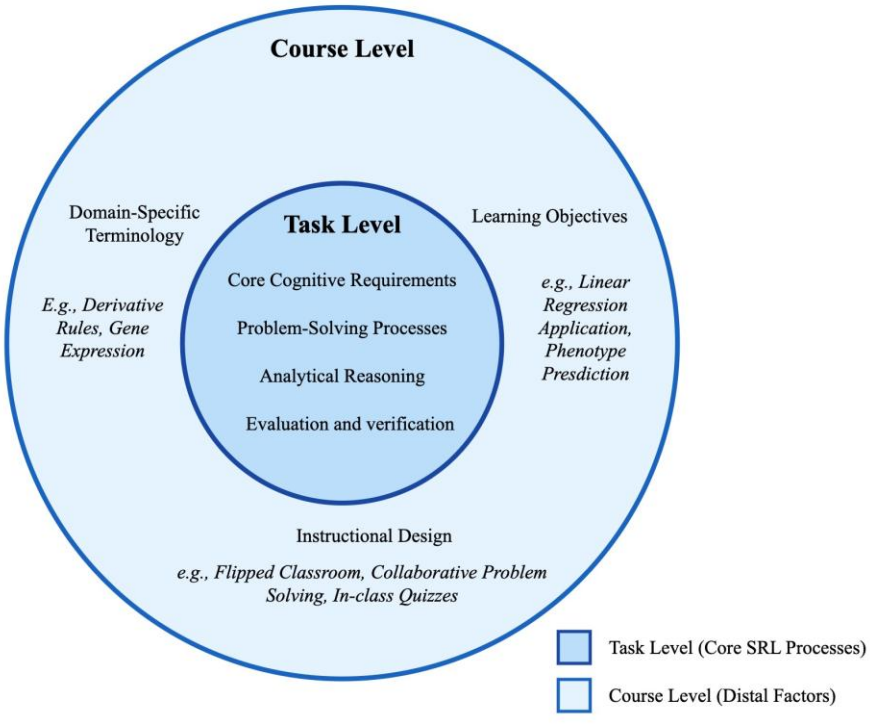

Figure 1. Different Context levels in SRL.

This conceptual model illustrates how SRL operates within nested contextual levels, showing concentric rings representing increasingly broader contexts that influence how students regulate their learning processes.

2. Literature Review

2.1. Self-Regulated Learning (SRL) and Think-Aloud Protocols (TAPs)

2.1.1. Theoretical Foundations of SRL

SRL represents the complex interplay between cognitive, metacognitive, and motivational processes that enable learners to actively manage their learning (Winne & Hadwin, 1998; Zimmerman, 2000). At the core of SRL are cognitive and metacognitive processes that enable learners to effectively direct their learning experiences (Winne, 1995). Cognitive processes involve the actual manipulation of information and the application of learning strategies (elaboration, organization, and rehearsal). Metacognitive processes pertain to thinking about one's own thinking and learning (planning, monitoring, control, evaluation). Meta-analyses consistently showed that effective SRL interventions substantially improve learning outcomes across educational domains and settings (Dignath et al., 2008; Theobald, 2021). SRL enhances students' metacognitive strategy use (Guntur & Purnomo, 2024), improves motivation and self-efficacy, and strengthens academic performance across STEM domains (Dignath et al., 2008).

SRL theory has evolved to recognize regulation of learning as inherently situated within specific contexts and highly sensitive to task demands. The Metacognitive and Affective Model of Self-Regulated Learning (MASRL; Efklides, 2011; Efklides & Schwartz, 2024) provides a detailed account of how regulatory processes operate at multiple levels during task engagement. At the *Person level*, learners bring general knowledge and beliefs about learning strategies that must be adapted to specific task demands. This includes domain knowledge, motivational beliefs, and metacognitive knowledge about strategies that exist prior to engaging with specific tasks. The *Task* × *Person* level captures how cognitive processing, metacognitive experiences, and affect interact during actual task completion. This interaction is critical because it represents the dynamic process through which learners experience subjective feelings (e.g., feeling of difficulty), make judgments (e.g., judgment of learning), and form online task-specific estimates of their performance that directly influence strategy selection and effort allocation (Efklides et al., 2017). The MASRL model (Efklides, 2011; Efklides & Schwartz, 2024) distinction between *Person-level* factors and *Task* × *Person* interactions explained why identical learning behaviors may serve different regulatory functions depending on a learner's current metacognitive experiences and affective states.

Winne and Hadwin's (1998) information processing model elaborated on internal SRL processes by emphasizing how learners' regulatory decisions are governed by loosely sequential conditions-operations-products-evaluations-standards (COPES) cycles that are inherently context-dependent. Within this framework, conditions represent task parameters and cognitive resources available to the learner; operations are the cognitive processes and strategies applied; products are the outcomes of operations; evaluations involve comparing products against standards; and standards are criteria for determining success. These COPES elements interact across four distinct phases: task definition, goal setting and planning, enacting strategies, and adaptation. This model specifically distinguished task definition as a separate phase from goal setting; a critical theoretical distinction that highlights how learners' initial interpretation of task requirements fundamentally shapes their subsequent regulatory decisions. The recursive nature of these phases, with monitoring and control processes operating throughout, explained why similar verbalizations may represent different regulatory processes depending on their phase-specific context.

2.1.2. Task-Specific Manifestations of SRL Processes

Research on SRL has consistently demonstrated that although fundamental regulatory mechanisms remain consistent across educational contexts, the specific manifestations of these processes vary substantially based on task requirements and instructional designs (Alexander et al., 2011; Greene et al., 2015). This task-specific nature of SRL has profound implications for both research methodology and educational practice, especially when employing automated analysis tools across different contexts. Early research by Veenman et al. (1997) established that metacognitive skills exhibited both general (task-independent) and specific (task-dependent) components. Their findings indicated that approximately 60% of metacognitive variance could be attributed to a general metacognitive factor, with the remaining 40% explained by task-specific manifestations, documenting these differences across mathematics problem-solving tasks and text-based reading comprehension tasks.

The distinction between the functional purpose of a regulatory process and its surface-level linguistic expression creates a significant challenge for automated coding approaches. Greene and Azevedo (2009) emphasized this distinction in their coding framework, noting that accurate classification of SRL processes required attention to both the underlying regulatory function (which remains consistent across contexts) and the task-specific, micro-level linguistic expressions through which that function is communicated (which varies by learning environment). Beyond shaping linguistic expressions, task features also influence the frequency and type of SRL processes observed during task completion. Veenman and Beishuizen (2004) found that task features such as text difficulty differentially affected the frequency of metacognitive processes, with more complex texts prompting notably greater monitoring and evaluation activity relative to easier texts. Greene et al. (2010) extended this work through their examination of SRL across diverse history tasks, where coders successfully maintained inter-rater reliability

despite linguistic variations by focusing on functional similarities rather than surface expressions. Their study specifically demonstrated how tasks with discrete topics and language could be analyzed using a common codebook, with linguistic differences managed effectively by trained coders. This nuanced relationship between general SRL expression, and their task-specific manifestations becomes particularly important for automated coding with LLMs, to identify comparable regulatory processes across diverse educational settings despite variation in verbal expressions.

2.1.3. TAPs as a Methodology

TAPs have emerged as a particularly valuable method for capturing the dynamic and context-sensitive nature of SRL processes. This methodology involves participants verbalizing their thoughts during task completion, providing researchers with direct access to ongoing cognitive and metacognitive processes that would otherwise remain unobservable (Ericsson, 2017; Fox et al., 2011). Specifically, TAPs capture real-time processing, eliminating the recall biases inherent in retrospective measures (Greene et al., 2018; Winne & Perry, 2000). Furthermore, they provide rich contextual data about how regulatory processes interact with specific task features and environmental conditions. These characteristics make TAPs especially well-suited for examining the dynamic, context-dependent nature of SRL emphasized in theoretical models (Ackerman & Thompson, 2017; Greene et al., 2018).

Despite these strengths, coding TAPs requires significant labor and time. The verbal protocols generated through this methodology typically require labor-intensive processing including transcription, segmentation, and classification using theory-aligned coding schemes (Greene et al., 2013). This process necessitates multiple trained coders who must establish reliability before analyzing the full dataset. The time-consuming nature of traditional TAP coding creates a considerable bottleneck for SRL research, limiting sample sizes and hindering researchers' ability to examine regulatory processes across diverse educational contexts (Azevedo et al., 2010; Greene et al., 2018). These limitations have motivated interest in more efficient approaches to TAP coding that maintain methodological rigor and increase scalability.

2.2. Automated Coding of Qualitative Data Using LLMs

2.2.1. Technical Foundation of LLMs

Modern LLMs such as GPT (OpenAI, 2023), Claude (Anthropic, 2024), and Gemini models (Gemini Team et al., 2023) operate on the Transformer architecture, introduced by Vaswani et al. (2017) in "Attention Is All You Need." This architecture represents a shift from sequential processing to a parallelized attention-based design, enabling the system to focus on relevant information regardless of positional distance (Vaswani et al., 2017). The self-attention mechanism in transformer models processes long-range dependencies effectively (Khan et al., 2022; Vaswani et al., 2017) by calculating relevance weights for all parts of a dialogue when processing any single utterance.

Models with robust transformer architectures (e.g., GPT-4o, Claude 3.5, Gemini 1.5 Pro) have demonstrated good performance in complex reasoning tasks requiring contextual understanding (Anthropic, 2024; Brown et al., 2020; OpenAI, 2023). These models incorporate extensive parameter counts (i.e., typically ranging from tens to hundreds of billions of parameters) and pre-training datasets including scientific literature, educational content, and analytical writing. This pre-training enables pattern recognition in language corresponding to theoretical constructs in educational research, even without specific training on specialized coding schemas (Chowdhery et al., 2023; Wei et al., 2022).

2.2.2. LLMs for Qualitative Analysis: Prompt Engineering

LLMs are changing how researchers approach qualitative analysis by making coding more efficient. Learning analytics and AI in education researchers have explored the effectiveness of applying LLMs for qualitative coding across educational contexts such as analyzing students' written reflections and perspectives (Nagashima et al., 2024; Ramanathan et al., 2025), tutoring dialogues (J. Lin et al., 2024; Scarlatos et al., 2025), focus group transcripts (Bakharia et al., 2025), debugging behaviors (Liu et al., 2025), and SRL strategies in think-aloud verbalizations (Zhang et al., 2024). Zhang et al. (2023) confirmed that LLMs performed effectively in both simulated and real datasets when compared to manual coding. Barany et al. (2024) evaluated codebook development approaches and established that hybrid human-LLM methods produced more reliable results than either fully manual or automated approaches. McClure et al. (2024) validated LLMs' effectiveness in supporting deductive coding when paired with well-designed codebooks, and De Paoli (2024) established a structured six-phase process model enabling researchers to identify and refine inductive codes more efficiently than traditional methods.

The effectiveness of LLMs in qualitative coding depends significantly on prompt engineering: the systematic design of input instructions to optimize performance for specific analytical tasks (White et al., 2023). Several promising approaches align with coding needs in educational research. For example, *chain-of-thought prompting* guides LLMs through step-by-step analytical processes mirroring expert coding decisions, breaking down complex classification tasks into smaller reasoning steps (Kojima et al., 2022; Wei et al., 2022). *Few-shot learning* provides concrete examples of the desired analysis decisions (Brown et al., 2020; Min et al., 2022). *Role specification* through expert persona adoption positions the model to emulate the knowledge and analytical approach of domain specialists (Shanahan et al., 2023; White et al., 2023). These approaches enable

LLMs to address core challenges in qualitative coding: maintaining theoretical consistency, processing contextual information, and handling ambiguous cases that require interpretation within theoretical frameworks.

2.2.3. Application of LLMs in Coding SRL TAP Data

The task-specific nature of SRL processes presents unique challenges and opportunities for LLM-based coding approaches. Traditional manual coding methods required human coders to maintain extensive knowledge of both theoretical SRL frameworks and task-specific contexts to accurately interpret learners' verbalizations (Binbasaran Tuysuzoglu & Greene, 2015). This dual expertise requirement contributed significantly to the labor-intensive nature of SRL research.

LLMs address this challenge through their capacity to process large volumes of contextual information, detect patterns, and apply classification rules. The transformer architecture is particularly applicable to SRL research because of the complex, context-dependent nature of regulatory processes (Greene et al., 2015; Winne & Hadwin, 1998). For example, determining whether a student statement such as "I need to try a different approach" represents metacognitive monitoring, strategy adaptation, or motivation regulation depends critically on surrounding context that may span across multiple utterances in TAP data (Azevedo et al., 2018). A comment about adjusting strategy can only be properly classified as metacognitive regulation by considering earlier expressions of goal setting or task definition (Zhao et al., 2023). These technical capabilities address the core challenge in SRL coding: differentiating between superficially similar statements that serve different regulatory functions based on their broader context.

Recent empirical work by Zhang et al. (2024) illustrated both the potential and limitations of automating SRL detection in educational settings using LLMs. These authors experimented with two specific embedding approaches for coding SRL think-aloud transcripts: the Universal Sentence Encoder (USE) and OpenAI's text-embedding-3-small model. Although these embedding approaches effectively detected general SRL behaviors in student verbalizations from different tasks in chemistry and logic, they encountered difficulties with task-specific vocabulary. Notably, the "Realizing Errors" category showed the strongest transfer effects due to its consistent task-agnostic indicators such as "wrong" or "incorrect." These findings highlighted the importance of prompt engineering approaches that account for the task-specific characteristics of SRL, considering variations in task structures and epistemic demands (Greene et al., 2015; Veenman et al., 2006).

2.3. Purpose and Research Questions

We examine how theory-driven prompt engineering approaches can enhance the accuracy of SRL process coding in TAPs across different learning contexts. Through systematic evaluation of these factors, we aim to develop more effective approaches for automated coding of SRL processes. Our investigation is motivated by three key considerations:

1. **Task Effects**: Learning contexts significantly influence how students verbalize their self-regulatory processes. By comparing flipped mathematics and high-structure biology tasks, we aim to identify how tasks affect automated coding accuracy for SRL behaviors.
2. **Prompt Engineering Optimization**: Various prompt engineering approaches may yield different classification results across educational contexts. Understanding which combinations of strategies (zero-shot vs. few-shot, contextual information levels) perform best for specific SRL codes can improve automated analysis efficiency.
3. **SRL Process Characteristics**: Different self-regulatory processes may exhibit varying degrees of linguistic consistency across tasks. Cognitive processes may demonstrate more standardized verbal patterns (e.g., variable identification) compared to metacognitive processes (e.g., forming conclusions about task-specific concepts), which could require different prompt engineering approaches.

Based on these motivations, we address the following research questions:

RQ1: How do core task features implemented within different STEM course designs influence LLM coding accuracy for SRL processes?

RQ2: How does the effectiveness of different prompt engineering strategies vary across task implementations and SRL codes?

RQ3: How does the nature of different SRL processes (specific micro-level cognitive and metacognitive process) influence LLM coding accuracy across tasks?

3. Methods

3.1. Participants and Data Collection

3.1.1. Research Setting and Data Collection Procedures

Our participants included 49 undergraduate students enrolled in Introductory Biology and 49 in a Pre-calculus course, with age range 17–26, predominantly female representation (77.55% in mathematics, 75% in biology), mostly first-year students (57.14% in mathematics, 85.42% in biology). All procedures were approved and conducted in accordance with the ethical guidelines of the Institutional Review Board (#19-1897). Each participant completed 90-minute learning sessions focusing on

either Biodiversity (Biology) or Ellipses (Mathematics) content in a controlled laboratory setting. The biology task centered on Hardy-Weinberg equilibrium. It required students to calculate allele frequencies, predict genotype distributions, and apply evolutionary concepts to population genetics problems. The mathematics task involved analyzing ellipse equations through completing the square, identifying geometric properties (center, vertices, foci), and graphing conic sections (see Appendix E for task examples). During these sessions, participants verbalized their thoughts and actions when engaging in structured learning activities including video lectures, homework problems, and quizzes. These activities reflected the highly structured instructional designs that faculty implemented in their actual courses (see Figure 2).

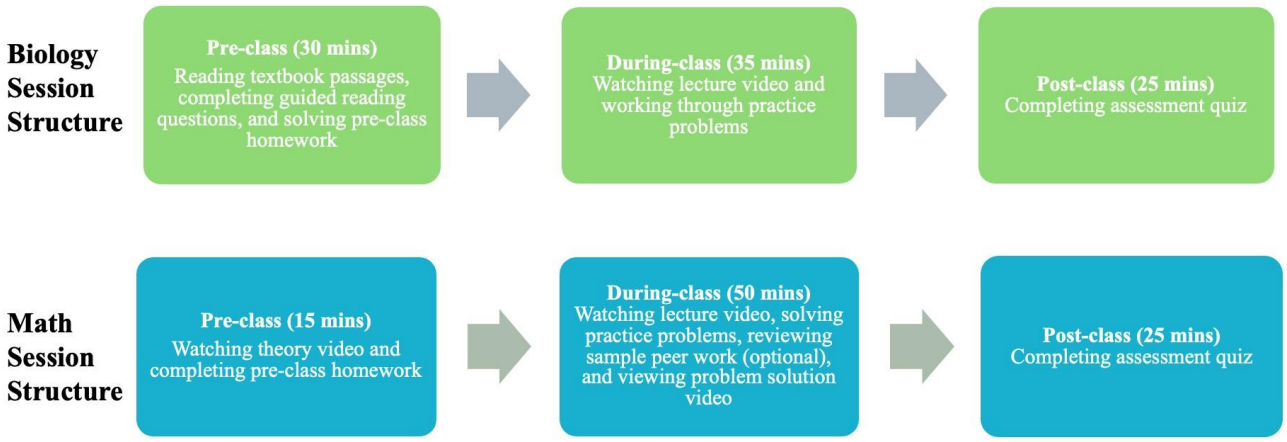

Figure 2. Structured activities for Biodiversity (Biology) and Ellipse (Mathematics) in a laboratory setting.

3.1.2. Codebook Development and Human Coding Procedures

We audio- and video-recorded participants' verbalizations during the sessions (see Figure 3). We used a professional transcription service (Rev.com) to transcribe the audio recordings. Three research assistants then segmented and coded the transcripts in NVivo. During concurrent think-aloud, participants verbalized their thoughts without elaborating on or explaining their mental processes. When a participant remained silent for more than three seconds, the researcher prompted "Please keep talking." Three research assistants independently segmented each transcript into the smallest unit that could be understood out of context. Each segment received only one code; when a segment could plausibly receive two or more codes, the coders either split the segment further, if possible, or assigned "no code." Coders merged contiguous segments carrying the same code into a single segment.

Building on Greene and Azevedo's (2009) hierarchical coding framework, we distinguish between macro-level and micro-level SRL processes. Macro-level processes represent broad categories of regulatory activity that align with phases and functions described in major SRL models (Pintrich, 2000; Winne & Hadwin, 1998; Zimmerman, 2000). Micro-level processes are the specific, observable regulatory behaviors nested within each macro-level category. For example, judgment of learning and feeling of knowing are micro-level processes within the macro-level category of Monitoring, whereas sub-goal setting is a micro-level process within Task Definition and Planning. Our codebook (Bernacki et al., 2025) included over 50 micro-level codes nested within five macrolevel categories: Task Definition and Planning (TASK), Monitoring (MONITOR), Domain-General Strategies (DGS), Domain-Specific Strategies (DSS), and Assessment Strategies (ASSESS). Each code incorporated detailed operational definitions, sample excerpts, and distinguishing features from similar codes. We refined the codebook through several iterations based on coder feedback and adjustments and validated that the codebook can be applied to different academic tasks (Bernacki et al., 2025). Across participants, we coded a total of 6,591 verbalizations for the biology task and 3,800 for the mathematics task.

We assessed coding reliability using Cohen's kappa and percentage agreement, mathematics: Cohen's $\kappa = 0.65$, percentage agreement = 74.0%; biology: Cohen's $\kappa = 0.66$, percentage agreement = 79.2%. Following Cicchetti and Feinstein (1990), we examined positive and negative agreement rates. Both tasks showed near-perfect negative agreement (mathematics: pneg = 0.998; biology: pneg = 0.998) but moderate positive agreement (mathematics: ppos = 0.65; biology: ppos = 0.67). This indicates that disagreements primarily reflected differential sensitivity in behavior detection rather than random coding errors. According to Hallgren (2012), these reliability measures indicated substantial agreement. After establishing reliability, we used the final coding files as the ground truth for evaluating the performance of LLM coding.

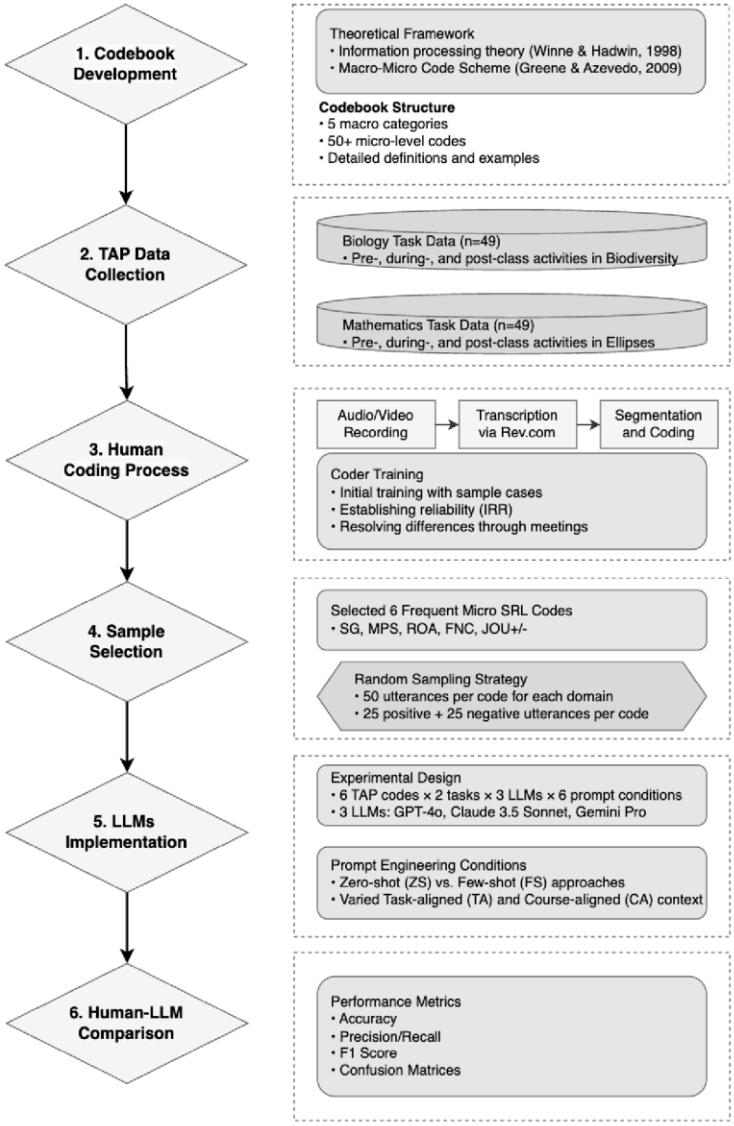

Figure 3. Research Design and Procedures.

3.2. Research Design

3.2.1. Experimental Design and Variables

We employed a factorial design to test how different prompt engineering approaches affect automated coding of TAPs data. Specifically, we examined 6 TAP codes × 2 tasks × 3 LLMs × 6 prompt conditions (see Table 1 and Table 2). This design enabled systematic evaluation of how different prompt engineering strategies affected coding accuracy across different SRL processes and educational contexts.

The six micro-level codes represented distinct cognitive and metacognitive processes essential to SRL theory and can be applied to both mathematics and biology (see Table 1). We selected these codes based on their theoretical significance and prevalence in our dataset (see Appendix C for details). At the macro level of Task Definition and Planning, the micro-level *Setting Goals for Sub-target* (SG) code identified verbalizations where students articulated specific targets for their learning process. Within the macro category of Strategies, we coded for two distinct types of cognitive processes: *Mathematical Problem Solving* (MPS), which captured instances where students applied mathematical operations or procedures to work toward solutions. MPS captured procedural problem-solving rather than domain-specific content knowledge and was therefore applicable to both mathematics and biology tasks. We coded for *Forming New Conclusions* (FNC), which identified where students drew inferences or generated new understanding from available information. The macro category of Monitoring contained micro-level codes that distinguished between *Positive Judgments of Learning* (JOU+), indicating students' perceived understanding, and *Negative Judgments of Learning* (JOU-), signaling perceived confusion or knowledge gaps. These verbalizations provided evidence of students' real-time assessment of their comprehension. Finally, within the macro category

of Assessment, the *Ruling Out Answers* (ROA) micro-code identified instances where students systematically eliminated incorrect solutions, representing a critical evaluation process in problem-solving activities.

For each code (6 TAP codes) and task (mathematics, biology), we randomly sampled 50 utterances (25 positive examples of the code, 25 negative examples) from the human-coded TAPs dataset, yielding 600 utterances from the two tasks (biology and mathematics) for evaluating binary classification accuracy (i.e., returning 1 if a TAP code is present and 0 if the code is absent). This sampling approach ensured balanced representation of each code and enabled fair comparisons across the different experimental conditions (see Figure 3).

Table 1. Selected TAP codes definition and examples

Macro TAP code	Micro TAP code (*N*)	Definition
Task Definition/ Planning	SG (*N* = 50)	Sub-Goal: target (SG): Learner articulates a specific sub-goal that is relevant to the task. Must immediately carry out some action relevant to the sub-goal (i.e., can't drop the goal immediately after verbalizing).
Domain-specific Strategy	MPS (*N* = 50)	Mathematical Problem-Solving (MPS): Student actively working through a mathematical problem, showing steps or calculations.
Assessment	ROA (*N* = 50)	Ruling Out Answers (ROA): Reviewing answer choices on a multiple-choice question and systematically ruling out answer choices in order to narrow down to the answer they will select.
Domain General Strategy	FNC (*N* = 50)	Forming New Conclusion (FNC): Putting together two pieces of information and drawing a new conclusion that extends beyond what is presented in the learning environment.
Monitoring	JOU+ (*N* = 50)	Judgment of Understanding (JOU): Learner recognizes that they do (JOU+) or do not (JOU-) understand content related to the learning task.
Monitoring	JOU- (*N* = 50)	Judgment of Understanding (JOU): Learner recognizes that they do (JOU+) or do not (JOU-) understand content related to the learning task.

Table 2. Prompt conditions and components

	Chain of thought (CoT)	Rubric (CR)	Few-shot (FS)	Task Aligned (TA)	Course-Aligned (CA)
ZS-CoT-CR: Basic prompt including chain-of-thought reasoning and coding criteria.	x	x			
ZS-CoT-CR-TA: Enhanced the basic prompt with task-aligned contextual information.	x	x		x	
ZS-CoT-CR-CA: Further enhanced with both task and course design with broader instructional features.	x	x		x	x
FS-CoT-CR: Basic prompt supplemented with one positive and one negative example.	x	x	x		
FS-CoT-CR-TA: Few-shot prompt enhanced with task-aligned contextual information.	x	x	x	x	
FS-CoT-CR-CA: Comprehensive prompt including examples and both task and course design with broader instructional features	x	x	x	x	x

3.2.2. LLM Implementation and Prompt Engineering Conditions

We investigated three contemporary LLMs released in 2024: OpenAI GPT-4o, Anthropic Claude 3.5 Sonnet, and Google Gemini 1.5 Pro. We selected these models based on their demonstrated capabilities in natural language understanding and generation tasks (OpenAI, 2023; Anthropic, 2024; Gemini et al., 2023). To implement deductive coding of TAPs data, we developed a systematic prompt engineering approach grounded in recent instruction techniques (Brown et al., 2020;

Wei et al., 2022). Each prompt followed a standardized structure that included explicit role definition, detailed task framing, and structured reasoning steps (Liu et al., 2025). Code definition and rubrics (CR) was part of the base prompt and included code explanations (Table 1) and defining components (e.g., "problem elements" and "active engagement" for MPS) for each TAP subcode. The base prompt (illustrated in Figure 4) established the model's function as an educational psychology expert and provided specific TAP code definitions. We then incorporated a five-step reasoning framework (Wei et al., 2022) that guided the coding process: 1. Identifying key utterance elements; 2. Assessing alignment with code definitions and examples (Hou et al., 2024); 3. Considering previous utterances and task contextual information (Rao et al., 2023); 4. Evaluating classification confidence; and 5. Making final binary classifications.

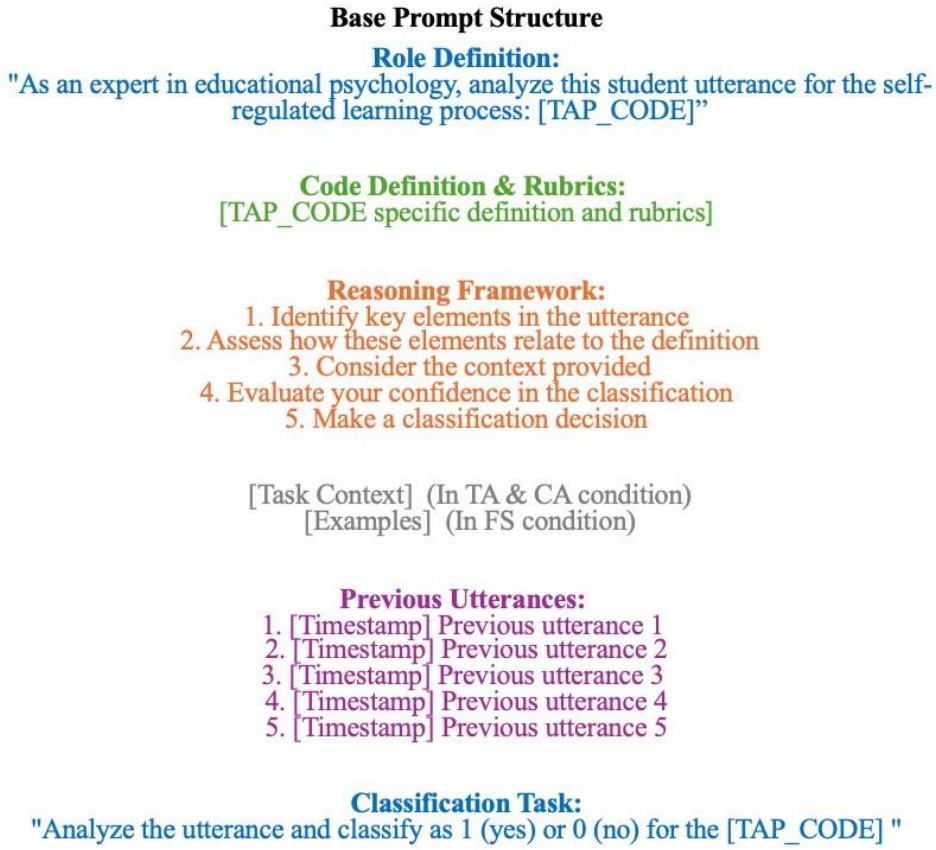

Figure 4. Base Prompt Structure. All prompts begin with a core structure containing the structured elements.

We implemented six prompt conditions (see Table 2) that varied component inclusion based on prompt engineering principles. The zero-shot with chain-of-thought reasoning and coding rubric (*ZS-CoT-CR*) condition included only basic reasoning structure and specific coding criteria. The zero-shot with task-aligned context (*ZS-CoT-CR-TA*) added information about the specific learning task, such as Biodiversity for biology or Ellipse for mathematics. The zero-shot with course-aligned context (*ZS-CoT-CR-CA*) incorporated both task information and course design elements, including instructional approaches like high-structure active learning for biology and flipped classroom for mathematics (see Appendix F for TA and CA example). Additionally, the few-shot with basic structure (*FS-CoT-CR*) introduced one positive and one negative example for the targeted SRL TAP code. Moreover, the few-shot with task-aligned context (*FS-CoT-CR-TA*) combined examples with task-aligned contextual information. Finally, the few-shot with course-aligned context (*FS-CoT-CR-CA*) included examples, chain-of-thought reasoning, coding criteria, and both contextual information types. Appendix A contains an example prompt for *FS-CoT-CR-CA*.

These carefully designed prompt conditions reflect different levels of contextual integration in SRL assessment. Our prompt engineering approach delineated two contextual levels in the conceptual framework. The fundamental level, task-aligned context, refers to core cognitive requirements across tasks including formula application, variable identification, and calculations. At this level, mathematics and biology students demonstrated comparable cognitive processes such as systematic computational steps. Task-aligned prompts contained information about these abstract problem-solving processes without disciplinary terminology.

The broader course-aligned context encompassed discipline-specific applications integrated with instructional design elements (Van Alten et al., 2020). At this level, mathematics problems presented direct disciplinary applications and flip-class design. In comparison, biology problems required translation between disciplinary concepts (like heredity) and mathematical representations, introducing heightened linguistic complexity (similar to findings from Greene et al., 2015). Course-aligned prompts incorporated both task-level information and these broader contextual elements. This two-level approach corresponds to the concentric rings model described in Figure 1, wherein the task level aligns inside the course level; and the course level represents the combination of instructional design features and content coverage that shape the overall learning environment and support self-regulatory processes.

Our preliminary analyses (Appendix D) demonstrated that a five-utterance contextual window achieved optimal performance (accuracy in coding TAP codes, compared to human coding) when compared with alternative window sizes (0, 1, 3, 5, and 10 utterances). This indicated that providing the LLMs with the five previous TAP utterances from the students produced the most accurate coding results at the same time balancing efficiency and context retention.

3.2.3. Analytical Strategies

For all RQs, we used utterance-level accuracy as the dependent variable. We computed accuracy by comparing each LLM's label to the human-coded label: accuracy equaled 1 when the labels matched and 0 otherwise.

Each utterance appeared in 18 experimental conditions (3 LLMs × 6 prompt conditions), creating non-independent observations. To account for this clustered structure, we fit generalized linear mixed models (GLMM) with a binomial distribution and logit link using the lme4 package in R (Bates et al., 2015). We included utterance as a random intercept to model within-utterance correlations. The intraclass correlation coefficient (ICC = .78) confirmed substantial dependence among repeated measurements of each utterance, supporting the use of mixed-effects modeling. We set statistical significance at $\alpha = .05$ and report odds ratios (OR) with 95% confidence intervals for effect size interpretation.

For **RQ1** (task effects), we modeled accuracy as a function of Task, LLM, and TAP Code as fixed effects. For **RQ2** (prompt engineering effects), we first tested a model including Few-shot prompting, Context, Task, LLM, and the Task × Few-shot interaction. To examine whether prompt effects varied by TAP code, we fit a separate model including the Few-shot × TAP Code interaction, controlling for Task, LLM, and Context. We estimated post-hoc pairwise comparisons (few-shot vs. zero-shot within each code) using the emmeans package (Lenth, 2023), with Bonferroni correction for six comparisons. For **RQ3** (code-specific patterns), we modeled accuracy as a function of the Task × TAP Code interaction and LLM. All models included a random intercept for utterance ID to account for repeated classifications of the same utterance across conditions. We also report descriptive accuracy rates and confusion matrices (precision, recall, and F1 scores; see Appendix B) to complement the inferential analyses.

4. Results

4.1. Overall Model Performance Metrics

The performance of LLMs in classifying SRL behaviors across 600 utterances exhibited moderate to high accuracy across conditions, ranging from .49 to .90 accuracy. Analysis of the classification performance revealed several notable patterns in accuracy across different conditions and tasks within mathematics and biology.

4.2. Tasks Effect, TAPs and LLMs Effect (RQ1)

We fit a GLMM predicting accuracy from Task, LLM, and TAP Code with utterance as a random intercept (300 utterances in mathematics, 300 in biology; 10,800 total observations across 6 prompts × 3 LLMs).

Task. Mathematics tasks yielded significantly higher classification accuracy ($M = .78$, $SD = .41$) than biology tasks ($M = .56$, $SD = .50$), $\beta = 2.39$, $SE = 0.30$, OR = 10.93, 95% CI [6.04, 19.78], $p < .001$ (see Figure 5). The odds of correct classification were approximately 11 times higher for mathematics utterances. This difference suggests that SRL behaviors may be more distinctly identifiable in mathematical tasks, possibly due to the more structured nature of mathematical reasoning processes.

TAP Code. Accuracy varied across TAP codes (see Figure 6). Compared to FNC (reference category), ROA showed significantly higher accuracy ($\beta = 1.56$, OR = 4.78, 95% CI [1.63, 14.00], $p = .004$), as did MPS ($\beta = 1.40$, OR = 4.05, 95% CI [1.41, 11.64], $p = .009$). JOU- showed a trend toward higher accuracy, although this was not significant ($\beta = 0.93$, OR = 2.53, $p = .081$). JOU+ and SG did not differ significantly from FNC. Descriptively, MPS showed the highest accuracy ($M = .73$), followed by ROA ($M = .72$), JOU- ($M = .69$), JOU+ ($M = .66$), FNC ($M = .62$), and SG ($M = .61$).

LLM. Claude 3.5 Sonnet achieved the highest accuracy ($M = .71$). GPT-4o ($M = .69$) and Gemini 1.5 Pro showed significantly lower accuracy than Claude (GPT-4o-Claude: $\beta = -0.21$, OR = 0.81, 95% CI [0.68, 0.96], $p = .016$; Gemini-Claude: $\beta = -0.94$, OR = 0.39, 95% CI [0.33, 0.46], $p < .001$; see Figure 7).

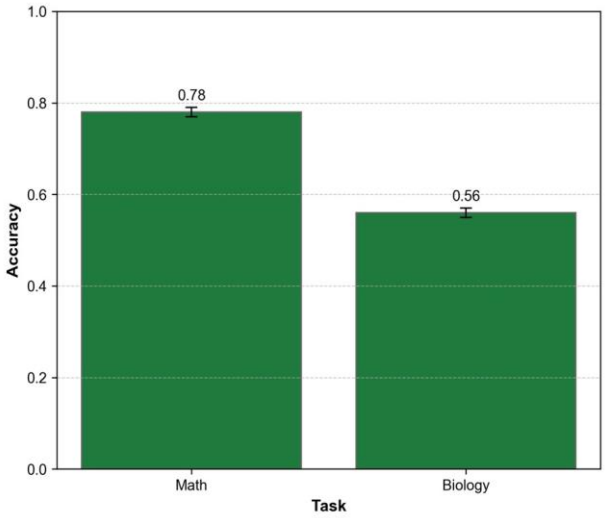

Figure 5. Main effect of tasks on LLMs' coding accuracy.

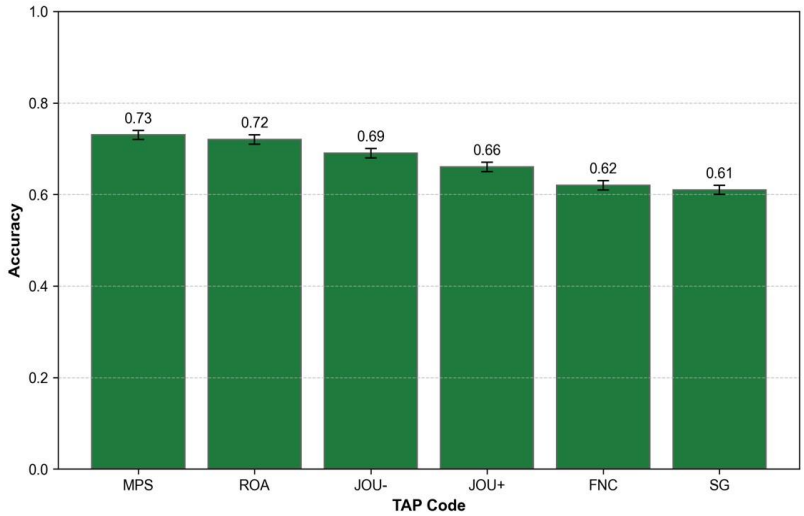

Figure 6. Main effect of TAP code types on LLMs' coding accuracy (ranked by accuracy of the TAP code).

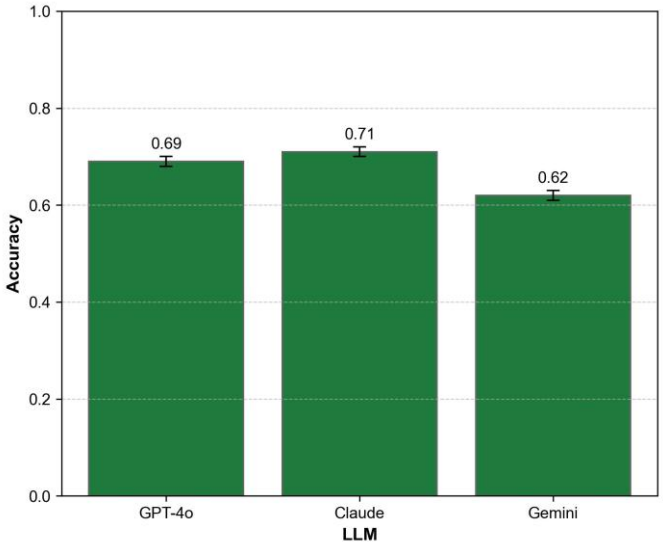

Figure 7. LLMs main effect in coding accuracy.

Confusion matrices revealed distinct classification patterns (see Figure 8). GPT-4o and Claude 3.5 Sonnet showed conservative prediction patterns with higher true negative rates (40.5% and 41.9%) than Gemini 1.5 Pro (32.3%). Gemini 1.5 Pro produced a notably higher false positive rate (17.7%) compared to GPT-4o (9.5%) and Claude 3.5 (8.1%).

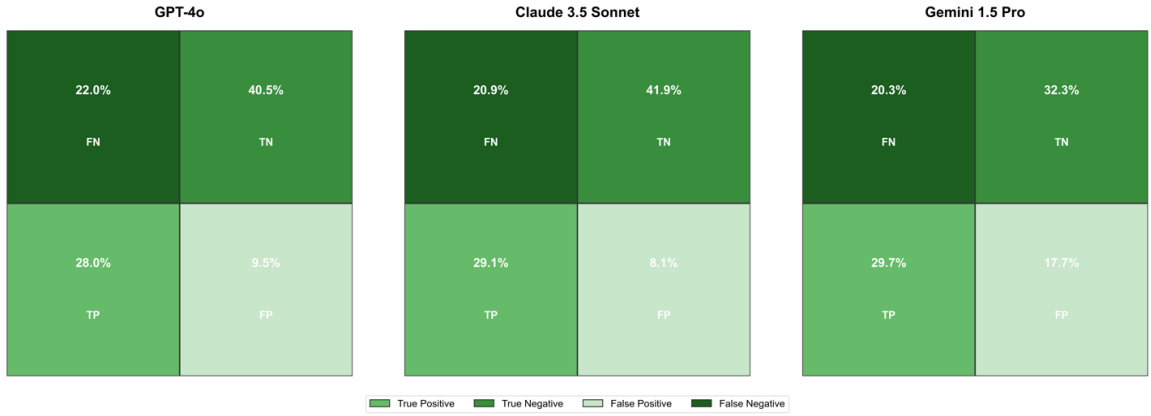

Figure 8. Confusion matrix in LLMs coding performance.

4.3. Effects of Prompt Engineering Strategies (RQ2)

4.3.1. Zero-Shot vs Few-Shot Performance Across Tasks

We fit a GLMM predicting accuracy from Task, LLM, Few-shot (zero-shot vs. few-shot), Context (none, task-aligned, course-aligned), and the Task × Few-shot interaction, with utterance as a random intercept.

The overall effect of prompting approach was not significant, β = -0.10, *SE* = 0.07, OR = 0.90, 95% CI [0.79, 1.02], $p = .111$. The Task × Few-shot interaction was also non-significant, β = -0.17, *SE* = 0.13, OR = 0.84, 95% CI [0.65, 1.09], $p = .189$, indicating that the relative effectiveness of few-shot prompting did not differ significantly between mathematics and biology tasks. However, post-hoc contrasts revealed heterogeneous effects across TAP codes.

To examine code-specific effects, we fit a model including the Few-shot × TAP Code interaction (controlling for Task, LLM, and Context). Post-hoc contrasts revealed heterogeneous effects. For JOU-, few-shot prompting yielded descriptively higher accuracy (estimated marginal mean: 90.1% vs. 85.9% for zero-shot), although this difference was not significant after Bonferroni correction (OR = 1.48, 95% CI [1.10, 1.99], $p\_adj = .054$). Conversely, zero-shot prompting significantly outperformed few-shot for SG (81.6% vs. 71.2%, OR = 0.56, 95% CI [0.41, 0.75], $p\_adj < .001$). The remaining codes (FNC, JOU+, MPS, ROA) showed no significant differences after Bonferroni correction (all $p\_adj \geq .05$).

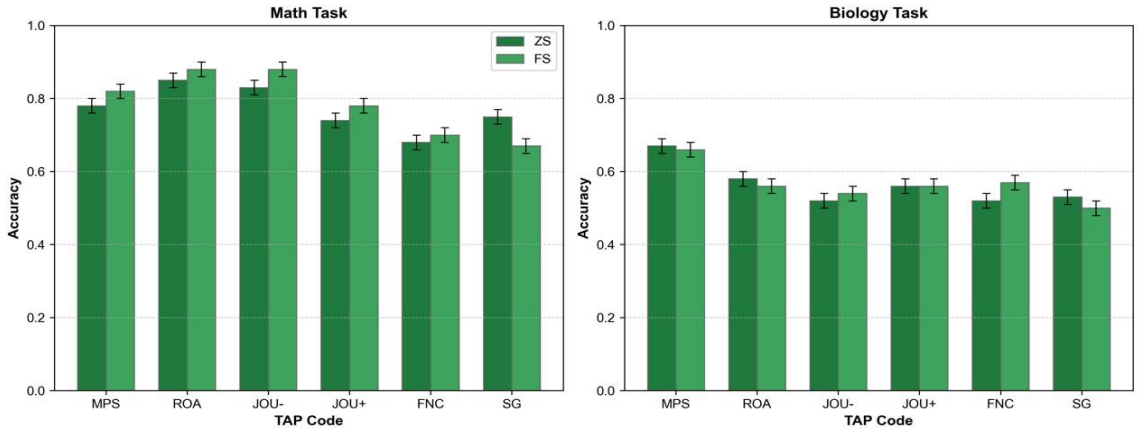

Figure 9. FS vs ZS prompt conditions across TAPs code and tasks in LLMs coding performance.

Note. Figure 9 compares zero-shot (ZS) and few-shot (FS) prompting performance across all TAP codes and both task types.

4.3.2. Context Effects across Tasks

Neither context condition differed significantly from the course-aligned reference (no-context: β = -0.03, OR = 0.97, $p = .719$; task-aligned: β = -0.13, OR = 0.88, $p = .102$). Descriptively, accuracy remained stable across all three context levels for both tasks and all TAP codes (see Figure 10). Adding task-specific or course-level contextual information to prompts did not improve coding accuracy for any SRL process in either academic task.

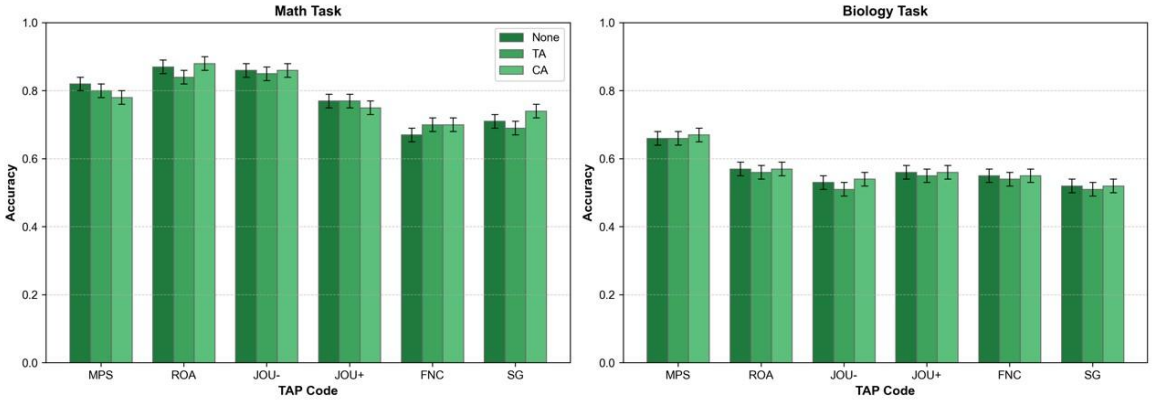

Figure 10. Different context level prompting across TAPs code and tasks in LLMs coding performance.

Note. Figure 10 represent accuracy of LLMs in three different context levels in the prompting approaches: None (no additional context), TA (Task-Aligned context that includes task-specific details), and CA (Course-Aligned context that includes both task and broader course-level information).

4.4. TAP Code-Specific Performance (RQ3)

4.4.1. Task and TAP Code Interactions

We fit a GLMM predicting accuracy from the Task × TAP Code interaction and LLM, with utterance as a random intercept.

The Task × TAP Code interaction revealed differential task effects across SRL processes (see Figure 11). Compared to FNC (reference), JOU- showed a statistically significantly larger task effect (β = 2.25, *SE* = 1.05, OR = 9.49, 95% CI [1.21, 74.62], p = .033), indicating that the mathematics advantage was especially pronounced for negative judgments of learning. The ROA interaction was not significant (β = 2.00, *SE* = 1.08, OR = 7.39, 95% CI [0.90, 60.95], p = .063). The interactions for JOU+ (β = 1.23, p = .187), SG (β = 0.89, p = .395), and MPS (β = 0.34, p = .748) were also not statistically significant.

Descriptively, mathematics yielded higher accuracy than biology across all TAP codes. JOU- showed the largest task difference (Δ = .33), followed by ROA (Δ = .30), JOU+ (Δ = .21), SG (Δ = .20), FNC (Δ = .14), and MPS (Δ = .13). MPS showed the smallest cross-task gap, suggesting that procedural problem-solving processes maintained relatively consistent linguistic patterns across domains.

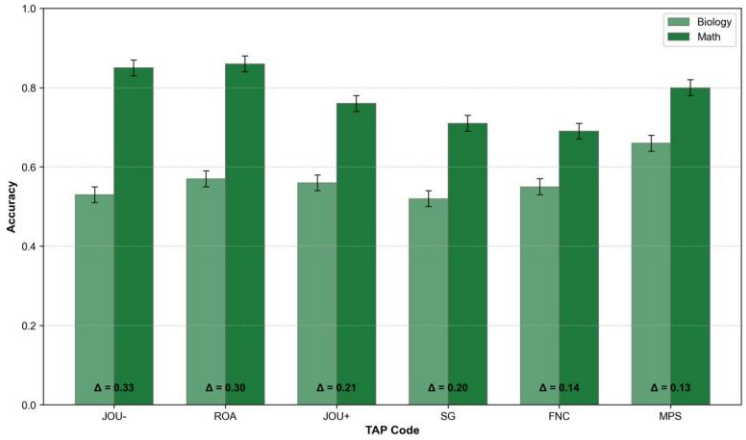

Figure 11. Task differences across TAP codes in LLMs coding performance.

4.4.2. Prompt, Task, and TAP Code Interactions

Descriptive analysis of individual TAP codes revealed distinct classification patterns across prompt conditions and tasks overall (Figure 12) and for each LLM (Figures 13-15). In mathematics, ROA demonstrated consistently high performance across all conditions (ranging from .83 to .90), with peak accuracy under FS-CA prompting (.90). MPS also showed strong and stable performance in mathematics (.74-.83), particularly excelling with zero-shot prompts. Notably, FNC remained the most challenging code in mathematics (.66-.71), showing minimal variation across prompt conditions.

In Biology tasks, MPS emerged as the most reliably classified code (.65-.68), maintaining relatively stable performance across all prompt conditions. SG proved most challenging in biology (.49-.55), with particularly low performance under FS conditions.

Further analysis revealed that task differences between mathematics and biology tasks persisted across various prompt conditions, although the magnitude varied by TAP code and condition. The most pronounced task differences were consistently observed in JOU- and ROA codes, regardless of prompt engineering approaches. Notably, the MPS code demonstrated the highest overall performance in biology tasks (M = .66), at the same time maintaining strong performance in mathematics (M = .80). In contrast, SG showed the lowest performance in biology (M = .52). These findings suggest that the identification of certain TAP codes is inherently more robust in mathematical problem-solving contexts compared to biological reasoning tasks. This is possibly due to the more structured nature of mathematical thinking processes or the more explicit manifestation of specific TAP behaviors within mathematical reasoning. The consistent pattern across all codes indicates a substantive task effect that transcends specific prompt engineering strategies.

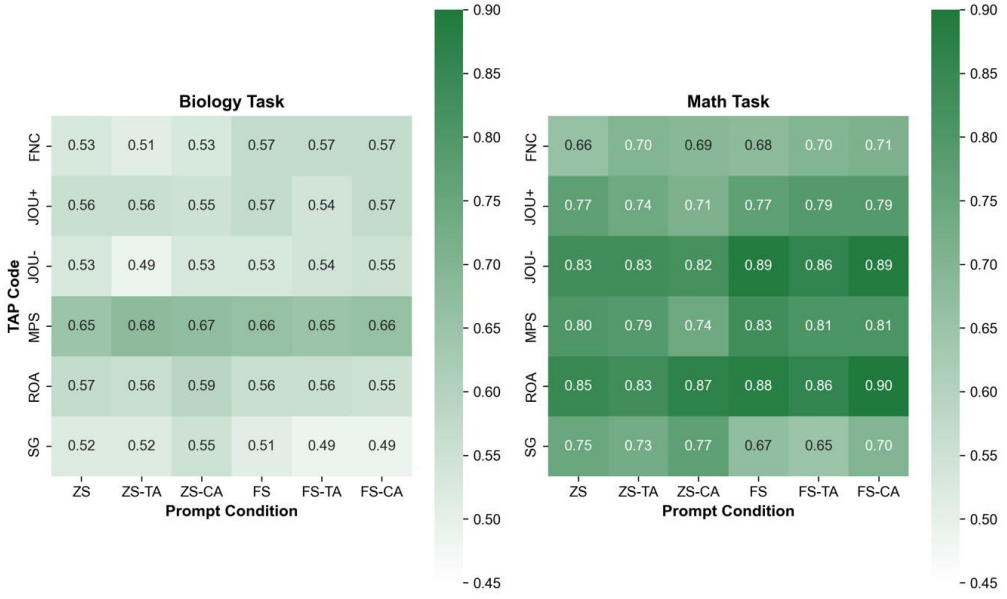

Figure 12. LLMs coding accuracy across TAP codes, prompt condition, and tasks.

Note. Heatmaps show mean accuracy by TAP code × prompt × task. Each heatmap cell includes N=150 observations (50 utterances × 3 LLMs); values are rounded to two decimals.

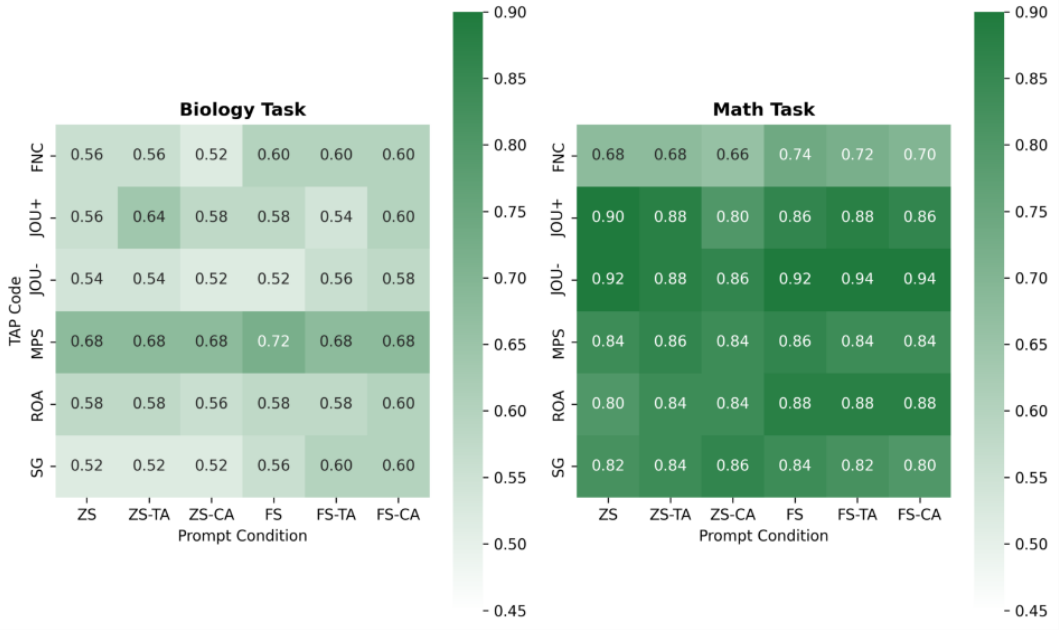

Figure 13. Claude coding accuracy across TAP codes, prompt condition, and tasks.

Note. Heatmaps show mean accuracy by TAP code × prompt × task for Claude. Each cell uses N = 50 utterances; values are rounded to two decimals.

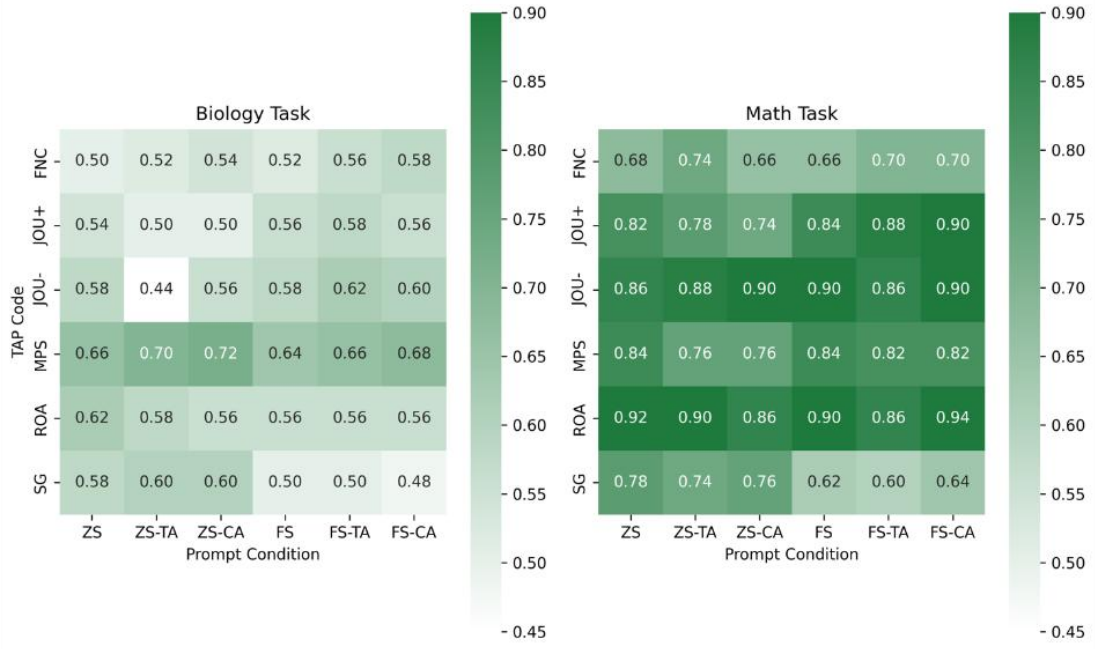

Figure 14. GPT-4o coding accuracy across TAP codes, prompt condition, and tasks.

Note. Heatmaps show mean accuracy by TAP code × prompt × task for GPT-4o. Each cell uses $N = 50$ utterances; values are rounded to two decimals.

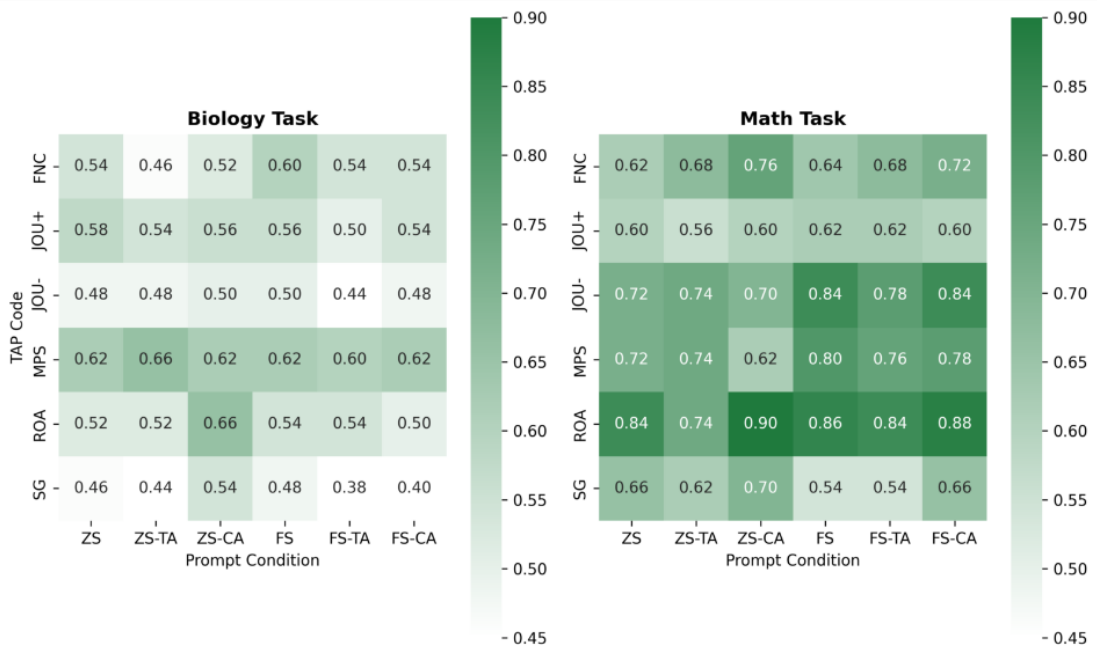

Figure 15. Gemini 1.5 Pro coding accuracy across TAP codes, prompt condition, and tasks.

Note. Heatmaps show mean accuracy by TAP code × prompt × task for Gemini 1.5 Pro. Each cell uses $N = 50$ utterances; values are rounded to two decimals.

5. Discussion

The pragmatic aim of this study was to pursue LLM coding to augment or replace human coding, to scale the use of TAPs in service of refining SRL theory and expanding TAP application to authentic learning tasks. Our analysis of LLMs' performance in coding SRL processes in TAPs revealed several significant patterns. First, in examining task effect across all SRL codes

(RQ1), we observed that mathematical problem-solving tasks yielded higher classification accuracy (M = .78) compared to biology tasks (M = .56). Second, there was differential performance among the LLMs. Claude 3.5 Sonnet demonstrated the highest accuracy, statistically outperforming GPT-4o (OR = 0.81, p = .016) and Gemini 1.5 Pro (OR = 0.39, p < .001). Third, prompt engineering strategies showed code-specific rather than uniform effects (RQ2). The overall effect of few-shot prompting was not statistically significant, but post-hoc analyses revealed that few-shot prompting yielded descriptively higher JOU- classification accuracy (although not statistically significant, p_adj = .054) but significantly impaired SG classification. Finally, we compared accuracy across SRL codes (RQ3). Despite the overall performance differences between tasks, MPS showed the smallest cross-task performance gap (ΔM = .13) among all codes analyzed. This stability contrasted sharply with metacognitive codes like JOU- that showed dramatic task-dependent variations (ΔM = .33). We discuss these findings in relation to prior research on SRL and LLMs to extend current understanding of automated coding of dynamic SRL processes in educational contexts.

5.1. Task Effects

Our analysis revealed distinct patterns in how cognitive versus metacognitive processes were classified across learning tasks. The findings demonstrated specific strengths and challenges of using LLMs to code SRL behaviors in mathematics versus biology contexts.

In mathematics tasks, where problem-solving follows more structured procedures (Nathan & Koedinger, 2000), LLMs demonstrated robust classification performance across multiple SRL behaviors. Such performance may be because problem-solving discourse (MPS) tends to follow standardized patterns with precise procedural language (e.g., "So C and D are the only ones that work"), creating verbalizations that are particularly amenable to LLMs (Liu et al., 2025). Furthermore, Liu et al. (2025) found that LLM coding performance varied depending on construct properties such as clarity, concreteness, and objectivity, suggesting that well-defined, observable behaviors are more amenable to automated classification. This may help explain why LLMs performed better with mathematics problem-solving discourse, where verbalizations tend to follow more concrete and procedural patterns, than with less structured biological reasoning.

In contrast, biology tasks presented greater challenges for automated classification, as students frequently combined multiple cognitive processes within single statements. Utterances like "No this doesn't seem right because natural selection would take longer" integrated problem-solving assessment with conceptual application, creating multi-layered verbalizations. Such verbalizations reflect Greene and Azevedo's (2009) observation that TAP data in concept-rich contexts can reflect multiple regulatory processes. Consequently, more research is needed to determine whether LLMs can be trained to identify such intricate verbalizations and subsequently infer the underlying cognitive processes they indicate.

The most pronounced task-dependent performance differences emerged in metacognitive monitoring processes, particularly in JOU-. LLM classification achieved notably higher accuracy in mathematics (M = .85) compared to biology (M = .53). This substantial difference (ΔM = .33) suggests limitations in current LLM capabilities for coding complex metacognitive verbalizations across different academic contexts. Moreover, the linguistic structure of biology verbalizations, characterized by task-specific terminology and multi-layered reasoning within single utterances, creates particular challenges for automated classification tasks. These findings align with recent research on natural language processing across scientific tasks, which demonstrates that LLMs frequently exhibit varying performance based on the linguistic patterns and structural complexity inherent in different tasks (Yang et al., 2024).

Analysis of confusion matrices reveals the underlying patterns driving these differences. Take the Claude 3.5 Sonnet model with the *FS-CoT-CR* condition as an example. Here, mathematics classification showed balanced outcomes (True Positive: 24, True Negative: 22) with minimal errors (False Negative: 1, False Positive: 3). Students' mathematics utterances contained explicit phrases that signaled uncertainty about computational steps, such as "I'm confused," "I have no idea," "I don't know," or "Not sure." However, in biology classifications, we observed substantial asymmetry (True Positive: 5, True Negative: 21, False Negative: 20, False Positive: 4). This imbalanced pattern indicated a systematic difficulty in identifying JOU- in biology contexts. More specifically, students' monitoring processes in biology tasks manifested through nuanced expressions that combined task-aligned knowledge with varying degrees of uncertainty. As a result, utterances like "I'm guessing from the example they have here, that P is the dominant allele" intertwined tentative reasoning with task-specific terminology, thereby requiring simultaneous analysis of both conceptual coherence and metacognitive stance.

An unexpected and particularly insightful finding emerged in the relative stability of mathematical problem-solving strategy (MPS) classification across both contexts. Despite the overall performance differences between tasks, MPS showed the smallest cross-task performance gap (ΔM = .13) among all codes analyzed. The data suggests that people verbalize certain core reasoning processes similarly across contexts, creating linguistic consistency that aids LLM classification accuracy. Additionally, this linguistic consistency appeared in students' verbalization patterns when employing computational reasoning, regardless of the subject matter. For example, in mathematics, students verbalized calculations with explicit steps: "I'm going to go ahead and pull the nine out. So, y squared plus two y equals negative nine." Similarly, in biology, students used parallel

computational structures despite different content: "P is .6 and Q is .4. So, P-squared is .6 square plus 2 times .6 times .4 plus .4 squared equals one."

These findings align with Anderson's (1982) Theory of Cognitive Architecture (ACT*) theory of skill acquisition, which distinguished between procedural knowledge (i.e., how to execute cognitive operations) and declarative knowledge (i.e., factual information specific to different fields). The relatively consistent coding performance for MPS across both tasks suggests that LLMs can reliably identify procedural knowledge patterns even when the surface-level content vocabulary differs between biology and mathematics contexts. The procedural aspects of problem-solving appear to maintain consistent linguistic patterns that LLMs can detect regardless of the specific declarative knowledge being applied.

5.2. Evaluation of LLMs & Prompt Engineering Strategies

Claude 3.5 Sonnet achieved the highest accuracy (M = .71), significantly outperforming GPT-4o (M = .69, OR = 0.81, p = .016), with both substantially outperforming Gemini 1.5 Pro (M = .62, OR = 0.39, p < .001). This finding is consistent with findings by Huang et al. (2024) on complex reasoning tasks. The confusion matrix analysis revealed distinctive error patterns, contributing to research on LLM applications in qualitative coding process (cf. Chang et al., 2024; Tai et al., 2024; Ziems et al., 2024). GPT-4o and Claude 3.5 Sonnet exhibited more conservative prediction tendencies, contrasting with Gemini 1.5 Pro's higher false positive rate. This indicates that architectural differences affect not only accuracy but classification error mechanisms. The finding presents a crucial consideration for researchers selecting LLMs for educational coding tasks.

The observed differences in model architecture and training methodologies explained these classification patterns. Technical analyses by Brown et al. (2020) and Z. Lin et al. (2024) documented similar error patterns across LLM architectures. GPT-4o and Claude 3.5 Sonnet's conservative prediction patterns likely stem from training data differences and instruction tuning approaches affecting precision-recall tradeoffs in classification tasks (Bommasani et al., 2022; Katz et al., 2023; Zhao et al., 2023). These architectural and training variations produce distinct error patterns despite similar overall accuracy, as documented in LLM evaluation studies (Schaeffer et al., 2023; Singhal et al., 2023).

Prompt engineering strategies demonstrated varying effectiveness across tasks, supporting research by Liu et al. (2025), Wei et al. (2022), and Liévin et al. (2024) on LLM performance context-sensitivity. Few-shot prompting produced code-specific effects: JOU- showed descriptively higher accuracy under few-shot conditions but significantly impaired SG classification (OR = 0.56, p_adj < .001). This pattern corresponds with Brown et al.'s (2020) findings that few-shot learning benefits depend on structural alignment between the provided examples and target tasks. The addition of contextual information (i.e., task-aligned and course-aligned) produced no significant impact on classification accuracy across both tasks, suggesting distal context outside the immediate task context may not help in LLM detection of SRL behavior.

Our findings suggest several practical strategies for educational researchers implementing LLMs for SRL coding. First, MPS classification's consistent performance across tasks demonstrates that cognitive process codes with clear language markers (such as procedure knowledge) may be particularly suitable for automated coding approaches, even without extensive task-aligned contextualization. Second, the substantial task differences in metacognitive process classification highlight the necessity for specialized prompt strategies for processes manifesting differently across learning contexts. Third, few-shot prompting produced code-specific effects, improving JOU- but impairing SG classification. This suggests that examples help when target verbalizations share consistent markers but may constrain models when expressions are more diverse.

5.3. Advancing Learning Analytics Through Validated SRL Process Detection

Our findings have important implications for advancing learning analytics research on self-regulated learning. Although most learning analytics studies have relied on more general SRL indicators such as engagement metrics or clickstream patterns (Saint et al., 2020; Winne, 2017), our approach demonstrates the feasibility of capturing nuanced, theory-informed SRL microprocesses through automated coding of think-aloud protocols. By applying Greene and Azevedo's (2009) comprehensive micro-level framework rather than macro-level indicators of more general SRL processes (e.g., channels of SRL processes including cognition, metacognition, motivation, affect; Azevedo et al., 2022), we can detect specific and nuanced regulatory processes. In addition, TAPs provide more reliable evidence of actual SRL events compared to retrospective self-report measures that suffer from memory limits and social desirability bias (Winne & Perry, 2000). The successful automation of TAP coding using LLMs paves the way for scalable learning analytics applications including real-time SRL detection and adaptive scaffolding in digital learning environments (Khalil et al., 2023; Lim et al., 2023), automated scaffolding of SRL support based on detected regulatory deficits (Pérez-Álvarez et al., 2020), and personalized feedback systems that target individual SRL trajectory (Ingkavara et al., 2022; Taub et al., 2021; Wiedbusch et al., 2021).

Furthermore, future multimodal research combining validated TAP coding with other data sources, such as video recordings, text reflections, physiological data, eye-tracking, and log files, could establish whether specific digital patterns reliably indicate planning deficits, monitoring gaps, or strategy use difficulties (Azevedo & Gašević, 2019; Fan et al., 2022, 2023; Molenaar et al., 2023). This multimodal triangulation in SRL process would enable the development of comprehensive learner models, ultimately supporting more precise and effective SRL support at scale (Khalil et al., 2023; Lim et al., 2023;

Panadero et al., 2016). They further afford opportunities to scale observation of the discrete SRL event and event sequences theorized to occur under temporal, contingent, and contextual conditions (Ben-Eliyahu & Bernacki, 2015; Winne & Hadwin, 1998). If automatically coded verbal data that represent SRL processes can be made more broadly available, corpora of data may accrue in time to afford powered testing of these nuanced assumptions within SRL frameworks, leading to their testing and refinement.

6. Limitations and Future Directions

We identified key limitations and promising future directions. A significant limitation is current LLMs' struggle with long-range dependencies and lack of explicit training on cognitive and metacognitive processes. To address this, we propose future research exploring a hierarchical prompting framework that aligns with SRL theoretical models. This approach would first identify macro-level regulatory categories (e.g., monitoring, strategy use, planning) before progressing to specific micro codes like JOU+/JOU-, MPS, potentially improving classification accuracy by managing contextual information more effectively. This hierarchical approach aligns with SRL theoretical frameworks by Winne and Hadwin (1998) and Greene and Azevedo (2009), which conceptualize SRL as structured, nested processes. After initial categorization, subsequent prompting stages would refine classifications using theoretical framework-aligned decision trees. This approach integrates linguistic analysis with theoretical constraints to address the challenge of identifying equivalent regulatory processes across different tasks. Future research should also explore multimodal approaches that integrate verbal protocols with other data streams. As suggested by Järvelä et al. (2019) and Molenaar et al. (2023), combining TAPs with eye-tracking, physiological measures, and log data could provide complementary indicators of SRL processes that are less sensitive to verbalization differences across tasks. These multimodal approaches could potentially address some of the task-specific challenges identified in our analysis, particularly for metacognitive processes that show substantial cross-task variability. Additionally, collecting TAP data requires training participants and using controlled laboratory settings, which limits direct scalability to authentic classroom environments. Future researchers should explore less intrusive methods such as prompted self-explanation or embedded reflection prompts that maintain ecological validity (Wolters & Won, 2018).

7. Conclusion

Our findings demonstrated that LLM-based approaches can reliably code certain SRL processes (particularly MPS) across different STEM tasks and have potential as scalable tools for coding TAPs. However, the varying performance across different SRL codes highlighted the need for continued refinement of prompting strategies. These strategies should account for both the theoretical structure of SRL and the contextual nature of its manifestation.

Data Availability Statement

The research received IRB approval (#19-1897). The codebook and data are accessible via [https://osf.io/7jeru/?view_only=6c233acfbc9c4a8881f1983127c6062d]. The prompts and codebook are described in Figure 4 and Appendix C.

Declaration of Conflicting Interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

This research was supported by the U.S. National Science Foundation Award DRL 1920756 and an AI Provost Fellowship from the University of North Carolina at Chapel Hill. The opinions, findings, and conclusions or recommendations expressed are those of the authors and do not necessarily reflect the views of the U.S. National Science Foundation.

References

Ackerman, R., & Thompson, V. A. (2017). Meta-reasoning: Monitoring and control of thinking and reasoning. *Trends in Cognitive Sciences, 21*(8), 607-617. https://doi.org/10.1016/j.tics.2017.05.004

Alexander, P. A., Dinsmore, D. L., Parkinson, M. M., & Winters, F. I. (2011). Self-regulated learning in academic domains. In B. J. Zimmerman & D. H. Schunk (Eds.), *Handbook of self-regulation of learning and performance* (pp. 393-407). Routledge. https://doi.org/10.4324/9780203839010.ch25

Anderson, J. R. (1982). Acquisition of cognitive skill. *Psychological Review, 89*(4), 369. https://doi.org/10.1037/0033-295X.89.4.369

Anthropic. (2024). *Claude 3.5: More capable, correct, and comprehensive*. https://www.anthropic.com/news/claude-3-5-sonnet

Azevedo, R., Bouchet, F., Duffy, M., Harley, J., Taub, M., Trevors, G., ... & Cerezo, R. (2022). Lessons learned and future directions of MetaTutor: Leveraging multichannel data to scaffold self-regulated learning with an intelligent tutoring system. *Frontiers in Psychology*, *13*, 813632. https://doi.org/10.3389/fpsyg.2022.813632

Azevedo, R., & Gašević, D. (2019). Analyzing multimodal multichannel data about self-regulated learning with advanced learning technologies: Issues and challenges. *Computers in human behavior*, *96*, 207-210. https://doi.org/10.1016/j.chb.2019.03.025

Azevedo, R., Moos, D. C., Johnson, A. M., & Chauncey, A. D. (2010). Measuring cognitive and metacognitive regulatory processes during hypermedia learning: Issues and challenges. *Educational Psychologist, 45*(4), 210-223. https://doi.org/10.1080/00461520.2010.515934

Azevedo, R., Taub, M., & Mudrick, N. V. (2018). Understanding and reasoning about real-time cognitive, affective, and metacognitive processes to foster self-regulation with advanced learning technologies. In D. H. Schunk & J. A. Greene (Eds.), *Handbook of self-regulation of learning and performance* (2nd ed., pp. 254-270). Routledge. https://doi.org/10.4324/9781315697048-17

Bakharia, A., Shibani, A., Lim, L. A., McCluskey, T., & Shum, S. B. (2025). From transcripts to themes: A trustworthy workflow for qualitative analysis using large language models. In *CEUR Proceedings of the Workshop 'From Data to Discovery: LLMs for Qualitative Analysis in Education', 15th International Learning Analytics & Knowledge Conference (LAK '25).* https://hdl.handle.net/10779/DRO/DU:30473783

Barany, A., Nasiar, N., Porter, C., Zambrano, A. F., Andres, J. M. A. L., Bright, D., Choi, J., Gao, S., Giordano, C., Liu, X., Mehta, S., Shah, M., Zhang, J., & Baker, R. S. (2024). ChatGPT for education research: Exploring the potential of large language models for qualitative codebook development. In *Proceedings of the 25th International Conference on Artificial Intelligence in Education.* https://doi.org/10.1007/978-3-031-64299-9_10

Bates, D., Mächler, M., Bolker, B., & Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, *67*(1), 1–48. https://doi.org/10.18637/jss.v067.i01

Ben-Eliyahu, A., & Bernacki, M. L. (2015). Addressing complexities in self-regulated learning: A focus on contextual factors, contingencies, and dynamic relations. *Metacognition and Learning, 10*(1), 1-13. https://doi.org/10.1007/s11409-015-9134-6

Bernacki, M. L. (2018). Examining the cyclical, loosely sequenced, and contingent features of self-regulated learning: Trace data and their analysis. In D. H. Schunk & J. A. Greene (Eds.), *Handbook of self-regulation of learning and performance* (2nd ed., pp. 370–387). Routledge/Taylor & Francis Group. https://doi.org/10.4324/9781315697048-24

Bernacki, M. L., Yu, L., Kuhlmann, S. L., Plumley, R. D., Greene, J. A., Duke, R. F., Freed, R., Hollander-Blackmon, C., & Hogan, K. A. (2025). Using multimodal learning analytics to validate digital traces of self-regulated learning in a laboratory study and predict performance in undergraduate courses. *Journal of Educational Psychology, 117*(2), 176–205. https://doi.org/10.1037/edu0000890

Binbasaran Tuysuzoglu, B., & Greene, J. A. (2015). An investigation of the role of contingent metacognitive behavior in self-regulated learning. *Metacognition and Learning, 10*(1), 77-98. https://doi.org/10.1007/s11409-014-9126-y

Blackmore, C., Vitali, J., Ainscough, L., Langfield, T., & Colthorpe, K. (2021). A review of self-regulated learning and self-efficacy: The key to tertiary transition in science, technology, engineering and mathematics (STEM). *International Journal of Higher Education, 10*(3), 169-177. https://doi.org/10.5430/ijhe.v10n3p169

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., ... Liang, P. (2022). *On the opportunities and risks of foundation models*. arXiv. https://arxiv.org/abs/2108.07258

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems, 33*, 1877-1901. https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html

Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., ... & Xie, X. (2024). A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology, 15*(3), 1-45. https://doi.org/10.1145/3641289

Chen, X. (2013). STEM attrition: College students' paths into and out of STEM field. NCES 2014-001. *National Center for Education Statistics*. https://nces.ed.gov/pubs2014/2014001rev.pdf

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., ... & Fiedel, N. (2023). Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research, 24*(240), 1-113. https://www.jmlr.org/papers/volume24/22-1144/22-1144.pdf

Cicchetti, D. V., & Feinstein, A. R. (1990). High agreement but low kappa: II. Resolving the paradoxes. *Journal of clinical epidemiology*, *43*(6), 551-558. https://doi.org/10.1016/0895-4356(90)90159-M

De Paoli, S. (2024). Performing an inductive thematic analysis of semi-structured interviews with a large language model: An exploration and provocation on the limits of the approach. *Social Science Computer Review*, *42*(4), 997-1019. https://doi.org/10.1177/08944393231220483

Dignath, C., Buettner, G., & Langfeldt, H. P. (2008). How can primary school students learn self-regulated learning strategies most effectively? A meta-analysis on self-regulation training programmes. *Educational Research Review*, *3*(2), 101-129. https://doi.org/10.1016/j.edurev.2008.02.003

Efklides, A. (2011). Interactions of metacognition with motivation and affect in self-regulated learning: The MASRL model. *Educational Psychologist*, *46*(1), 6-25. https://doi.org/10.1080/00461520.2011.538645

Efklides, A., & Schwartz, B. L. (2024). Revisiting the metacognitive and affective model of self-regulated learning: Origins, development, and future directions. *Educational Psychology Review*, *36*(2), Article 61. https://doi.org/10.1007/s10648-024-09896-9

Efklides, A., Schwartz, B. L., & Brown, V. (2017). Motivation and affect in self-regulated learning: does metacognition play a role? In *Handbook of self-regulation of learning and performance* (pp. 64-82). Routledge. https://doi.org/10.4324/9781315697048-5

Ericsson, K. A. (2017). Protocol analysis. In W. Bechtel & G. Graham (Eds.), *A companion to cognitive science* (pp. 425-432). Wiley. https://doi.org/10.1002/9781405164535.ch33

Fan, Y., Lim, L., van der Graaf, J., Kilgour, J., Raković, M., Moore, J., Molenaar, I., Bannert, M., & Gašević, D. (2022). Improving the measurement of self-regulated learning using multi-channel data. *Metacognition and Learning*, *17*(3), 1025-1055. https://doi.org/10.1007/s11409-022-09304-z

Fan, Y., Raković, M., van der Graaf, J., Lim, L., Singh, S., Moore, J., Molenaar, I., Bannert, M., & Gašević, D. (2023). Towards a fuller picture: Triangulation and integration of the measurement of self-regulated learning based on trace and think aloud data. *Journal of Computer Assisted Learning*, *39*(4), 1303-1324. https://doi.org/10.1111/jcal.12801

Fox, M. C., Ericsson, K. A., & Best, R. (2011). Do procedures for verbal reporting of thinking have to be reactive? A meta-analysis and recommendations for best reporting methods. *Psychological Bulletin*, *137*(2), 316-344. https://doi.org/10.1037/a0021663

Gao, A. (2023). Prompt engineering for large language models. *Available at SSRN 4504303*. https://doi.org/10.2139/ssrn.4504303

Gemini Team, Anil, R., Borgeaud, S., Alayrac, J. B., Yu, J., Soricut, R., ... & Blanco, L. (2023). *Gemini: A family of highly capable multimodal models*. arXiv. https://doi.org/10.48550/arXiv.2312.11805

Giannakos, M., & Cukurova, M. (2023). The role of learning theory in multimodal learning analytics. *British Journal of Educational Technology*, *54*(5), 1246–1267. https://doi.org/10.1111/bjet.13320

Greene, J. A., & Azevedo, R. (2009). A macro-level analysis of SRL processes and their relations to the acquisition of a sophisticated mental model of a complex system. *Contemporary Educational Psychology*, *34*(1), 18-29. https://doi.org/10.1016/j.cedpsych.2008.05.006

Greene, J. A., Bernacki, M. L., & Hadwin, A. F. (2024). Self-regulation. In P. A. Schutz & K. R. Muis (Eds.). *Handbook of Educational Psychology* (*4th Edition*) (pp. 314-334). Routledge. https://doi.org/10.4324/9780429433726-17

Greene, J. A., Bolick, C. M., Caprino, A. M., Deekens, V. M., McVea, M., Yu, S., & Jackson, W. P. (2015). Fostering high-school students' self-regulated learning online and across academic domains. *The High School Journal*, *99*(1), 88-106. https://doi.org/10.1353/hsj.2015.0019

Greene, J. A., Bolick, C. M., & Robertson, J. (2010). Fostering historical knowledge and thinking skills using hypermedia learning environments: The role of self-regulated learning. *Computers & Education*, *54*(1), 230-243. https://doi.org/10.1016/j.compedu.2009.08.006

Greene, J. A., Deekens, V. M., Copeland, D. Z., & Yu, S. (2018). Capturing and modeling self-regulated learning using think-aloud protocols. In D. H. Schunk & J. A. Greene (Eds.), *Handbook of self-regulation of learning and performance* (2nd ed., pp. 323–337). Routledge/Taylor & Francis Group. https://doi.org/10.4324/9781315697048-21

Greene, J. A., Dellinger, K. R., Tüysüzoğlu, B. B., & Costa, L. J. (2013). A two-tiered approach to analyzing self-regulated learning data to inform the design of hypermedia learning environments. In R. Azevedo & V. Aleven (Eds.), *International Handbook of Metacognition and Learning Technologies*, 117-128. https://doi.org/10.1007/978-1-4419-5546-3_8

Greene, J. A., Hutchison, L. A., Costa, L., & Crompton, H. (2012). Investigating how college students' task definitions and plans relate to self-regulated learning processing and understanding of a complex science topic. *Contemporary Educational Psychology*, *37*, 307-320. https://doi.org/10.1016/j.cedpsych.2012.02.002

Guntur, M., & Purnomo, Y. W. (2024). A meta-analysis of self-regulated learning interventions studies on learning outcomes in online and blended environments. *Online Learning*, *28*(3), 563-584. https://doi.org/10.24059/olj.v28i3.4025

Hallgren, K. A. (2012). Computing inter-rater reliability for observational data: an overview and tutorial. *Tutorials in quantitative methods for psychology*, *8*(1), 23. https://doi.org/10.20982/tqmp.08.1.p023

Hou, Y., Zhang, J., Lin, Z., Lu, H., Xie, R., McAuley, J., & Zhao, W. X. (2024, March). Large language models are zero-shot rankers for recommender systems. In *European Conference on Information Retrieval* (pp. 364-381). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-56060-6_24

Huang, Z., Wang, Z., Xia, S., & Liu, P. (2024). *Olympic arena medal ranks: Who is the most intelligent AI so far*? arXiv. https://arxiv.org/abs/2406.16772

Ingkavara, T., Panjaburee, P., Srisawasdi, N., & Sajjapanroj, S. (2022). The use of a personalized learning approach to implementing self-regulated online learning. *Computers and Education: Artificial Intelligence*. *3*(1), 100086. https://doi.org/10.1016/j.caeai.2022.100086

Järvelä, S., Järvenoja, H., & Malmberg, J. (2019). Capturing the dynamic and cyclical nature of regulation: Methodological Progress in understanding socially shared regulation in learning. *International Journal of Computer-Supported Collaborative Learning*, *14*, 425-441. https://doi.org/10.1007/s11412-019-09313-2

Katz, A., Shakir, U., & Chambers, B. (2023). *The utility of large language models and generative AI for education research*. arXiv. https://arxiv.org/abs/2305.18125

Khalil, M., Prinsloo, P., & Slade, S. (2023). The use and application of learning theory in learning analytics: A scoping review. *Journal of Computing in Higher Education, 35*(3), 573–594. https://doi.org/10.1007/s12528-022-09340-3

Khan, S., Naseer, M., Hayat, M., Zamir, S. W., Khan, F. S., & Shah, M. (2022). Transformers in vision: A survey. *ACM computing surveys (CSUR)*, *54*(10s), 1-41. https://doi.org/10.1145/3505244

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., & Iwasawa, Y. (2022). Large language models are zero-shot reasoners. *Advances in Neural Information Processing Systems*, *35*, 22199-22213. https://doi.org/10.48550/arXiv.2205.11916

Lenth, R. (2023). emmeans: Estimated Marginal Means, aka Least-Squares Means. https://CRAN.R-project.org/package=emmeans

Liévin, V., Hother, C. E., Motzfeldt, A. G., & Winther, O. (2024). Can large language models reason about medical questions? *Patterns*, *5*(3). https://doi.org/10.1016/j.patter.2024.100943

Lim, T., Gottipati, S., Cheong, M., Ng, J. W., & Pang, C. (2023). Analytics-enabled authentic assessment design approach for digital education. *Education and Information Technologies*, *28*(7), 9025-9048. https://doi.org/10.1007/s10639-022-11525-3

Lin, J., Chen, E., Han, Z., Gurung, A., Thomas, D.R., Tan, W., Nguyen, N.D., & Koedinger, K. (2024). *How can i improve? Using GPT to highlight the desired and undesired parts of open-ended responses*. arXiv. https://doi.org/10.48550/arXiv.2405.00291

Lin, Z., Guan, S., Zhang, W., Zhang, H., Li, Y., & Zhang, H. (2024). Towards trustworthy LLMs: a review on debiasing and dehallucinating in large language models. *Artificial Intelligence Review*, *57*(9), 243. https://doi.org/10.1007/s10462-024-10896-y

Liu, X., Zambrano, A. F., Baker, R. S., Barany, A., Ocumpaugh, J., Zhang, J., Pankiewicz, M., Nasiar, N., & Wei, Z. (2025). Qualitative coding with GPT-4: Where it works better. *Journal of Learning Analytics*, *12*(1), 169-185. https://doi.org/10.18608/jla.2025.8575

Lockyer, L., Heathcote, E., & Dawson, S. (2013). Informing pedagogical action: Aligning learning analytics with learning design. *American Behavioral Scientist, 57*(10), 1439–1459. https://doi.org/10.1177/0002764213479367

McClure, J., Smyslova, D., Hall, A., & Jiang, S. (2024). Deductive coding's role in AI vs. human performance. In *Proceedings of the 17th International Conference on Educational Data Mining* (pp. 809-813). https://doi.org/10.5281/zenodo.12729958

Min, S., Lewis, M., Zettlemoyer, L., & Hajishirzi, H. (2022). *MetaICL: Learning to learn in context*. arXiv. https://arxiv.org/abs/2110.15943

Molenaar, I., de Mooij, S., Azevedo, R., Bannert, M., Järvelä, S., & Gašević, D. (2023). Measuring self-regulated learning and the role of AI: Five years of research using multimodal multichannel data. *Computers in Human Behavior*, *139*, 107540. https://doi.org/10.1016/j.chb.2022.107540

Nagashima, T., Vincoli, M., Scholz, N., & Su, M. (2024). Understanding students' nuanced views on AI-supported classroom learning through perspective taking. In *Proceedings of the 25th International Conference on Artificial Intelligence in Education*. Springer. https://doi.org/10.1007/978-3-031-98414-3_2

Nathan, M. J., & Koedinger, K. R. (2000). An investigation of teachers' beliefs of students' algebra development. *Cognition and Instruction, 18*(2), 209-237. https://doi.org/10.1207/S1532690XCI1802_03

National Academies of Sciences, Engineering, and Medicine. (2024). *Roundtable on systemic change in undergraduate STEM education*. National Academies Press. https://doi.org/10.17226/26901

OpenAI. (2023). *GPT-4 technical report*. arXiv. https://arxiv.org/pdf/2303.08774

Panadero, E., Klug, J., & Järvelä, S. (2016). Third wave of measurement in the self-regulated learning field: When measurement and intervention come hand in hand. *Scandinavian Journal of Educational Research*, *60*(6), 723–735. https://doi.org/10.1080/00313831.2015.1066436

Park, J., Yu, R., Rodriguez, F., Baker, R., Smyth, P., & Warschauer, M. (2018). Understanding student procrastination via mixture models. In *Proceedings of the 11th International Conference on Educational Data Mining*. https://eric.ed.gov/?id=ED593094

Pérez-Álvarez, R. A., Maldonado-Mahauad, J., Sharma, K., Sapunar-Opazo, D., & Pérez-Sanagustín, M. (2020). Characterizing learners' engagement in MOOCs: An observational case study using the NoteMyProgress tool for supporting self-regulation. *IEEE transactions on Learning Technologies*, *13*(4), 676-688. https://doi.org/10.1109/TLT.2020.3003220

Pintrich, P. R. (2000). The role of goal orientation in self-regulated learning. In *Handbook of self-regulation* (pp. 451-502). Academic Press. https://doi.org/10.1016/B978-012109890-2/50043-3

Ramanathan, S., Lim, L.-A., Mottaghi, N. R., & Buckingham Shum, S. (2025). When the prompt becomes the codebook: Grounded prompt engineering (GROPROE) and its application to belonging analytics. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference* (pp. 713-725). ACM Press. https://doi.org/10.1145/3706468.3706564

Rao, A., Pang, M., Kim, J., Kamineni, M., Lie, W., Prasad, A. K., ... & Succi, M. D. (2023). Assessing the utility of ChatGPT throughout the entire clinical workflow: development and usability study. *Journal of Medical Internet Research*, *25*(1). https://preprints.jmir.org/preprint/48659

Saint, J., Whitelock-Wainwright, A., Gasevic, D., & Pardo, A. (2020). Trace-SRL: A framework for analysis of microlevel processes of self-regulated learning from trace data. *IEEE Transactions on Learning Technologies*, *13*, 861-877. https://doi.org/10.1109/TLT.2020.3027496

Scarlatos, A., Baker, R. S., & Lan, A. (2025). Exploring knowledge tracing in tutor-student dialogues using LLMs. In *Proceedings of the 15th International Learning Analytics and Knowledge Conference* (pp. 249-259). ACM Press. https://doi.org/10.1145/3706468.3706501

Schaeffer, R., Miranda, B., & Koyejo, S. (2023). Are emergent abilities of large language models a mirage? *Advances in Neural Information Processing Systems*, *36*, 55565-55581. https://doi.org/10.48550/arXiv.2304.15004

Shanahan, M., McDonell, K., & Reynolds, L. (2023). Role play with large language models. *Nature*, *623*(7987), 493-498. https://doi.org/10.1038/s41586-023-06647-8

Singhal, K., Azizi, S., Tu, T., Mahdavi, S. S., Wei, J., Chung, H. W., ... & Natarajan, V. (2023). Large language models encode clinical knowledge. *Nature*, *620*(7972), 172-180. https://doi.org/10.1038/s41586-023-06291-2

Tai, R. H., Bentley, L. R., Xia, X., Sitt, J. M., Fankhauser, S. C., Chicas-Mosier, A. M., & Monteith, B. G. (2024). An examination of the use of large language models to aid analysis of textual data. *International Journal of Qualitative Methods*, *23*, 16094069241231168. https://doi.org/10.1177/16094069241231168

Taub, M., Azevedo, R., Rajendran, R., Cloude, E. B., Biswas, G., & Price, M. J. (2021). How are students' emotions related to the accuracy of cognitive and metacognitive processes during learning with an intelligent tutoring system?. *Learning and Instruction*, *72*, 101200. https://doi.org/10.1016/j.learninstruc.2019.04.001

Theobald, M. (2021). Self-regulated learning training programs enhance university students' academic performance, self-regulated learning strategies, and motivation: A meta-analysis. *Contemporary Educational Psychology, 66*, 101976. https://doi.org/10.1016/j.cedpsych.2021.101976

Van Alten, D. C., Phielix, C., Janssen, J., & Kester, L. (2020). Self-regulated learning support in flipped learning videos enhances learning outcomes. *Computers & Education*, *158*, 104000. https://doi.org/10.1016/j.compedu.2020.104000

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*. https://doi.org/10.48550/arXiv.1706.03762

Veenman, M. V. J., & Beishuizen, J. J. (2004). Intellectual and metacognitive skills of novices while studying texts under conditions of text difficulty and time constraint. *Learning and Instruction*, *14*(6), 621-640. https://doi.org/10.1016/j.learninstruc.2004.09.004

Veenman, M. V. J., Elshout, J. J., & Meijer, J. (1997). The generality vs domain-specificity of metacognitive skills in novice learning across domains. *Learning and Instruction, 7*(2), 187-209. https://doi.org/10.1016/S0959-4752(96)00025-4

Veenman, M. V. J., Van Hout-Wolters, B. H. A. M., & Afflerbach, P. (2006). Metacognition and learning: Conceptual and methodological considerations. *Metacognition and Learning, 1*(1), 3-14. https://doi.org/10.1007/s11409-006-6893-0

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., ... & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35, 24824-24837. https://doi.org/10.48550/arXiv.2201.11903

White, J., Fu, Q., Hays, S., Sandborn, M., Olea, C., Gilbert, H., ... & Schmidt, D. C. (2023). *A prompt pattern catalog to enhance prompt engineering with chatgpt*. arXiv. https://doi.org/10.48550/arXiv.2302.11382

Wiedbusch, M. D., Kite, V., Yang, X., Park, S., Chi, M., Taub, M., & Azevedo, R. (2021, February). A theoretical and evidence-based conceptual design of metadash: An intelligent teacher dashboard to support teachers' decision making and students' self-regulated learning. In *Frontiers in education* (Vol. 6, p. 570229). Frontiers Media SA. https://doi.org/10.3389/feduc.2021.570229

Winne, P. H. (1995). Inherent details in self-regulated learning. *Educational Psychologist, 30*(4), 173-187. https://doi.org/10.1207/s15326985ep3004_2

Winne, P. H. (2017). Learning analytics for self-regulated learning. In C. Lang, G. Siemens, A. Wise, & D. Gašević (Eds.), *Handbook of learning analytics* (pp. 241-249). Society for Learning Analytics Research. https://doi.org/10.18608/hla17.021

Winne, P. H., & Hadwin, A. F. (1998). Studying as self-regulated learning. In D. J. Hacker, J. Dunlosky, & A. C. Graesser (Eds.), *Metacognition in Educational Theory and Practice* (pp. 277-304). Routledge. https://doi.org/10.4324/9781410602350-12

Winne, P. H., & Perry, N. E. (2000). Measuring self-regulated learning. In M. Boekaerts, P. R. Pintrich, & M. Zeidner (Eds.), *Handbook of Self-Regulation* (pp. 531–566). Academic Press. https://doi.org/10.1016/B978-012109890-2/50045-7

Wolters, C. A., & Won, S. (2018). Validity and the use of self-report questionnaires to assess self-regulated learning. In D. H. Schunk & J. A. Greene (Eds.), *Handbook of self-regulation of learning and performance* (2nd ed., pp. 307–322). Routledge/Taylor & Francis Group. https://doi.org/10.4324/9781315697048-20

Yang, J., Jin, H., Tang, R., Han, X., Feng, Q., Jiang, H., ... & Hu, X. (2024). Harnessing the power of LLMs in practice: A survey on chatgpt and beyond. *ACM Transactions on Knowledge Discovery from Data*, *18*(6), 1-32. https://doi.org/10.48550/arXiv.2304.13712

Zhang, H., Wu, C., Xie, J., Kim, C., & Carroll, J. M. (2023). *QualiGPT: GPT as an easy-to-use tool for qualitative coding*. arXiv. https://doi.org/10.48550/arXiv.2310.07061

Zhang, J., Borchers, C., Aleven, V., & Baker, R. S. (2024). Using large language models to detect self-regulated learning in think-aloud protocols. In *Proceedings of the 17th International Conference on Educational Data Mining* (pp. 157-168). https://doi.org/10.5281/zenodo.12729790

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., ... & Wen, J. R. (2023). *A survey of large language models*. arXiv. https://doi.org/10.48550/arXiv.2303.18223

Ziems, C., Held, W., Shaikh, O., Chen, J., Zhang, Z., & Yang, D. (2024). Can large language models transform computational social science? *Computational Linguistics*, *50*(1), 237-291. https://doi.org/10.1162/coli_a_00502

Zimmerman, B. J. (2000). Attaining self-regulation: A social cognitive perspective. In M. Boekaerts, P. R. Pintrich, & M. Zeidner (Eds.), *Handbook of self-regulation* (pp. 13–39). Academic Press. https://doi.org/10.1016/B978-012109890-2/50031-7

Appendix A

Prompt example (MPS: *FS-CoT-CR-CA*)

As an expert in educational psychology, analyze this student utterance for the self-regulated learning process: MPS.

MPS Definition:
Mathematical Problem-Solving (MPS): Student actively working through a mathematical problem, showing steps or calculations.
Rubrics:
Components for MPS:
A. Problem Elements: Presence of relevant mathematical concepts, formulas, or calculations.
B. Active Engagement: Evidence of student actively solving or attempting to solve a mathematical problem.

Scoring for MPS:
- Satisfied (1): BOTH components A AND B are present.
- Non_Satisfied (0): ONE OR NONE of the components are present.

Reasoning steps:
1. Identify key elements in the utterance.
2. Assess how these elements relate to the definition of the TAP code.
3. Consider the context provided and make reasonable inferences.
4. Evaluate your confidence in the classification.
5. Make a classification decision.

Context:
1. [2024-10-31 18:05:00] I know that's Q-squared
2. [2024-10-31 18:05:07] What proportion is born homozygous recessive
3. [2024-10-31 18:05:13] That's going to be very small
4. [2024-10-31 18:05:19] P + Q add up to one
5. [2024-10-31 18:05:50] so that's going to be long tails 280 plus 180

Analyze the following utterance:
"180 long tail, these also have a long tail plus 240, so total is 420. Out of 420 plus 80 out of 500," Provide a brief analysis and classify as 1 (presence of the TAP code) or 0 (absence of the TAP code).

End your response with:
Confidence Level: [not sure/confident]
Classification: [1 or 0]

Appendix B

Performance Metrics by Model, TAP Code, and Domain

Model	TAP Code	Domain	Accuracy	Precision	Recall	F1	*n*
Claude-3.5	SG	Biology	0.553	0.593	0.34	0.432	300
		Math	0.83	0.837	0.82	0.828	300
	ROA	Biology	0.58	1	0.16	0.276	300
		Math	0.853	1	0.707	0.828	300
	JOU+	Biology	0.583	0.617	0.44	0.514	300
		Math	0.863	0.819	0.933	0.872	300
	JOU-	Biology	0.543	0.576	0.327	0.417	300
		Math	0.91	0.855	0.987	0.916	300
	MPS	Biology	0.687	0.792	0.507	0.618	300
		Math	0.847	0.813	0.9	0.854	300
	FNC	Biology	0.573	0.662	0.3	0.413	300
		Math	0.697	0.766	0.567	0.651	300
GPT-4o	SG	Biology	0.543	0.587	0.293	0.391	300
		Math	0.69	0.761	0.553	0.641	300
	ROA	Biology	0.573	1	0.147	0.256	300
		Math	0.897	1	0.793	0.885	300
	JOU+	Biology	0.54	0.539	0.547	0.543	300
		Math	0.827	0.761	0.953	0.846	300
	JOU-	Biology	0.563	0.58	0.46	0.513	300
		Math	0.883	0.825	0.973	0.893	300
	MPS	Biology	0.677	0.798	0.473	0.594	300
		Math	0.807	0.791	0.833	0.812	300
	FNC	Biology	0.537	0.641	0.167	0.265	300
		Math	0.69	0.777	0.533	0.632	300
Gemini 1.5 Pro	SG	Biology	0.45	0.426	0.287	0.343	300
		Math	0.62	0.667	0.48	0.558	300
	ROA	Biology	0.547	0.621	0.24	0.346	300
		Math	0.843	0.926	0.747	0.827	300
	JOU+	Biology	0.547	0.534	0.727	0.616	300
		Math	0.6	0.565	0.867	0.684	300
	JOU-	Biology	0.48	0.484	0.593	0.533	300
		Math	0.77	0.706	0.927	0.801	300
	MPS	Biology	0.623	0.683	0.46	0.55	300
		Math	0.737	0.703	0.82	0.757	300
	FNC	Biology	0.533	0.556	0.333	0.417	300
		Math	0.683	0.701	0.64	0.669	300

Note. We presented model performance using four key metrics: accuracy (i.e., total correct predictions divided by all predictions), precision (i.e., correctly identified positives out of all positive predictions), recall (i.e., successfully detected positive cases among all actual positives), F1 score (i.e., balanced metric combining precision and recall), and sample size (i.e., total utterances analyzed per experimental condition).

Appendix C

TAP code definition, rubrics, and examples updated from codebook

TAP code	Definition	Rubrics	Positive example	Negative example
MPS	Mathematical Problem-Solving (MPS): Student actively working through a mathematical problem, showing steps or calculations.	Components for MPS: A. Problem Elements: presence of relevant mathematical concepts, formulas, or calculations. B. Active Engagement: evidence of actively solving or attempting to solve a problem. Scoring: 1 if A and B; 0 otherwise.	Math (MPS positive) : "81−27=… what's 81−27? 81−27=54. √54 …" Biology (MPS positive) : "2% of the population has the recessive allele… calculate carriers. Okay so I'm…" (MPS)	Math (MPS negative) : "It doesn't make any sense… I really don't understand…" (JOU−) Biology (MPS negative) : "In a fictional population… which is aa… which i…" (JOU−)
FNC	Forming New Conclusion (FNC): Putting together two pieces of information and drawing a new conclusion that extends beyond what is presented.	Components for FNC: A. Integration: combines at least two pieces of information. B. Novel Conclusion: draws a conclusion beyond what is explicit. Scoring: 1 if A and B; 0 otherwise.	"My guess is taxes would probably go up in order for the government to pay for all of the healthcare." Explanation: integrates information and proposes a new inference (A and B satisfied).	"The text says that DNA is the genetic material in cells." Explanation: recalls presented information; no integration or novel conclusion (fails B).
JOU+	Judgment of Understanding (JOU): Learner recognizes that they do (JOU+) understand content related to the learning task.	Components for JOU: A. Expression of understanding. B. Content relevance to the task. Scoring: JOU+ = 1 if A and B; JOU- = 1 if lack of understanding is expressed with relevance; Non_JOU = 0 otherwise.	"That makes sense." Explanation: explicit statement of understanding tied to task content (A and B satisfied).	"I think I'll be able to recall this information during the test next week." Explanation: a Judgment of Learning (future recall), not current understanding (does not meet JOU criteria).
JOU-	Judgment of Understanding (JOU): Learner recognizes that they do not (JOU-) understand content related to the learning task.	Components for JOU: A. Expression of (lack of) understanding. B. Content relevance to the task. Scoring as above.	"I don't really understand that one that much…" Explanation: clearly expresses lack of understanding about task content (A and B satisfied).	"I probably won't remember this for the test." Explanation: a Judgment of Learning about future memory, not present understanding (does not meet JOU criteria).
SG:target	Sub-Goal: target (SG:target): Learner articulates a specific sub-goal relevant to the task and immediately carries out a related action.	Components for SG:target: A. Articulation: specific sub-goal. B. Task Relevance. C. Immediate Action. D. Target Identification. Scoring: 1 if all A–D; 0 otherwise.	"I'm going to go back to the e-text, scroll down, and make sure that "Q" would equal the recessive allele." Explanation: states a specific sub-goal with target and immediate action (A–D satisfied).	"I'm going to download the GRQs so I can type my answers." Explanation: administrative step; not a learning-related sub-goal with a specific target/action (fails D and possibly C).
Ruling Out Answers	Ruling Out Answers: Reviewing answer choices on a multiple-choice question and systematically ruling out options.	Components: A. Multiple-choice context. B. Review of options. C. Elimination process. D. Reasoning for elimination. Scoring: 1 if all A–D; 0 otherwise.	"Well, I can rule out A and D right away…" Explanation: explicitly eliminates options with implied reasoning in an MC context (A–D satisfied).	"My guess is taxes would probably go up in order for the government to pay for all of the healthcare." Explanation: not in an MC elimination context; lacks elimination reasoning (fails A, B, C, D).

Appendix D

Table D1. Robust check results for the optimal utterances window
(Based on results from Claude 3.5 in ZS-CT-COT condition)

TAP Code	Utterance Window	Count	Accuracy	Std
FNC	0	20	0.45	0.51
FNC	1	20	0.6	0.50
FNC	3	20	0.6	0.50
FNC	5	20	0.65	0.49
FNC	10	20	0.5	0.51
MPS	0	20	0.6	0.50
MPS	1	20	0.75	0.44
MPS	3	20	0.55	0.51
MPS	5	20	0.8	0.41
MPS	10	20	0.8	0.41

Appendix E

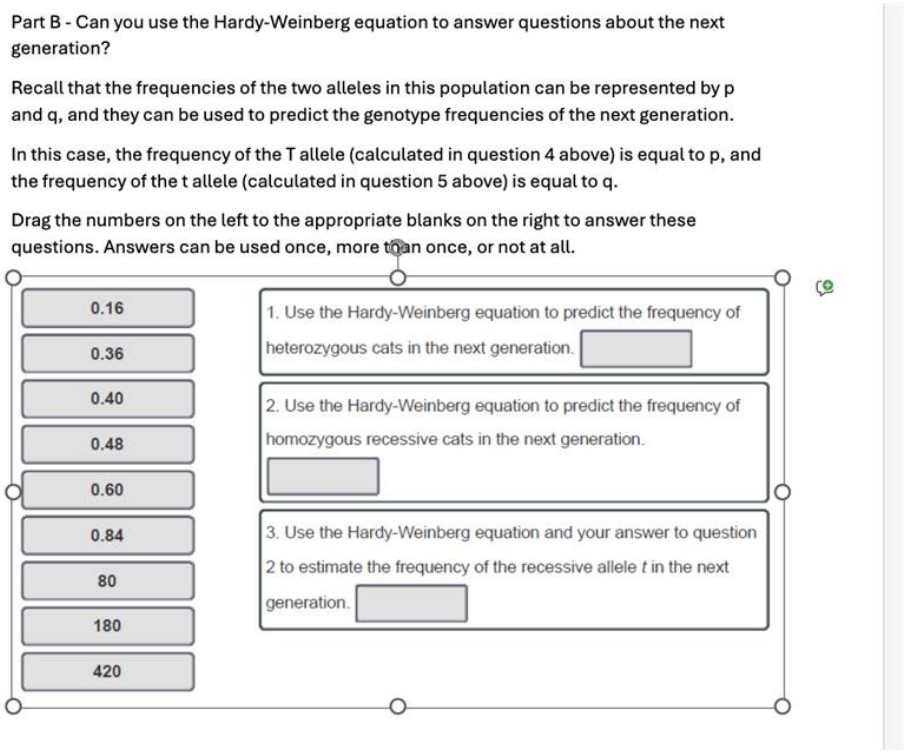

Figure E1. Sample Task Materials in Biology Task: Hardy-Weinberg Equilibrium Problem

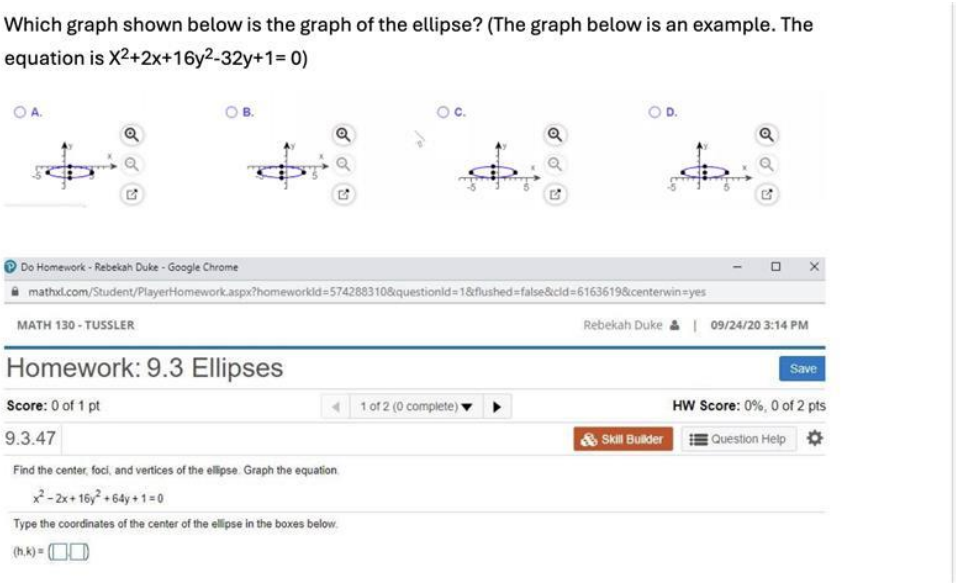

Figure E2. Sample Task Materials in Mathematics Task: Ellipse Identification and Graphing

Appendix F

Table F1. Task-aligned (TA) and Course-aligned (CA) Prompt Examples for Biology Tasks

Field	TA value	CA value
Content	Population Genetics and Hardy-Weinberg Equilibrium	Population Genetics and Hardy-Weinberg Equilibrium
Objectives	Understand equilibrium, apply equations, interpret evolutionary forces	Understand equilibrium, apply equations, interpret evolutionary forces
Tasks	Read textbook, solve problems, analyze scenarios	Read textbook, solve problems, analyze scenarios
Lesson	N/A	Introductory Biology 101 (undergraduate)
Environment	N/A	Laboratory
Procedures	N/A	Participate in Learning Catalytics activities, engage in problem-solving discussions

Table F2. Task-aligned (TA) and Course-aligned (CA) Prompt Examples for Mathematics Tasks

Field	TA value	CA value
Content	Ellipses in Analytic Geometry	Ellipses in Analytic Geometry
Objectives	Understand properties, apply equations, graph ellipses, analyze peer solutions	Understand properties, apply equations, graph ellipses, analyze peer solutions
Tasks	Watch videos, solve problems, review peer work	Watch videos, solve problems, review peer work
Lesson	N/A	Math 130 (undergraduate)
Environment	N/A	Online via Zoom, using mock Sakai LMS
Procedures	N/A	Watch lecture video, solve practice problem, review peer examples, watch review video

Note. Task- and course-aligned context. TA includes the Task Information block only; CA includes both Task Information and Lesson Information. The exact texts are summarized in the tables below (TA shows "N/A" where Lesson Information is not included).