Using AI and Natural Language Processing as Data Analysis Tools

Panel Discussion, June 30, 2026

Today, we will explore how natural language processing (NLP) can be used for analyzing large amounts of textual and other qualitative data including considerations of:

- Intercultural team cognition
- Multimodal formative feedback
- Computer programming assignments.

Panelists will tell their story of how they developed or deepened their NLP expertise, including how recent advances in large language models (e.g., GPT) have provided novel approaches for growing researchers' methodological skills. We will then move on to an interactive discussion to explore how NLP can be strategically used in other applications of interest.

Using AI and Natural Language Processing as Data Analysis Tools

Speaker	Relevant ECR:Core or ECR:BCSER Award
Moderator: Denise Wilson (University of Washington Seattle)	#2526864: NLP Assist: Natural Language Processing in STEM Education Research
Alejandra Magana (Purdue University)	#2219271: An AI-Augmented Phenomenographic Approach to Conceptualizing Undergraduate Students Experiences of Intercultural Team Cognition in STEM
Ha Nguyen (University of North Carolina Chapel Hill)	#2422286: Supporting Ambitious and Equitable Science Learning: Leveraging Artificial Intelligence in Developing Multimodal Formative Feedback
Hamid Karimi (Utah State University)	#2321304: Learning Analytics for Process-driven Computer Programming Assignments

Using AI and Natural Language Processing as Data Analysis Tools

Today's Panel Discussion

- Introduction to Speakers (Denise Wilson)
- Exploring the World of NLP and AI Terminology (Hamid)
- Opening Remarks
  - Hamid Karimi: How do students type and edit code in Python programming?
  - Alejandra Magana: How can team cognition be characterized and supported?
  - Ha Nguyen: How can LLMs support formative assessment in science education?
- Interactive Q&A
- Roundtable: Exploring how audience members can use NLP in their own work.

Create Your Own Discussion Topic

TODAY at 5-6 pm (with food/light refreshments provided)

Your name *

Meet-up title *

Meet-up description *

Location *

- Select a numbered table if you anticipate 8 or fewer attendees (most common).
- Select an entire breakout room if you want to facilitate a larger gathering (see bottom of list).
- This list is dynamically updated as tables/rooms are picked by others.

+ Add location

Terminology

Term	Definition
Artificial Intelligence (AI)	Computers performing tasks that usually require human intelligence.
Machine Learning (ML)	AI systems that learn patterns from data.
Deep Learning	ML using layered neural networks to learn complex patterns.
Generative AI	AI that creates new content, such as text, images, or code.
Natural Language Processing (NLP)	AI for understanding and working with human language.
Large Language Model (LLM)	AI trained on large text data to understand and generate language.
Large Multimodal Model (LMM)	AI that works across multiple data types, such as text, images, audio, or video.
Data Mining	Finding useful patterns in large datasets.
Text Mining	Finding useful patterns in text data.
Predictive Modeling	Using data to predict future or unknown outcomes.
Classification	Assigning data items to predefined categories.
Clustering	Grouping similar items without predefined labels.
Sentiment Analysis	Detecting opinion or emotion in text.
Topic Modeling	Finding major themes in a collection of texts.
Embedding	A numerical representation of meaning or similarity.

Large Language Models to Understand Student Programming Processes

#2321304

Hamid Karimi

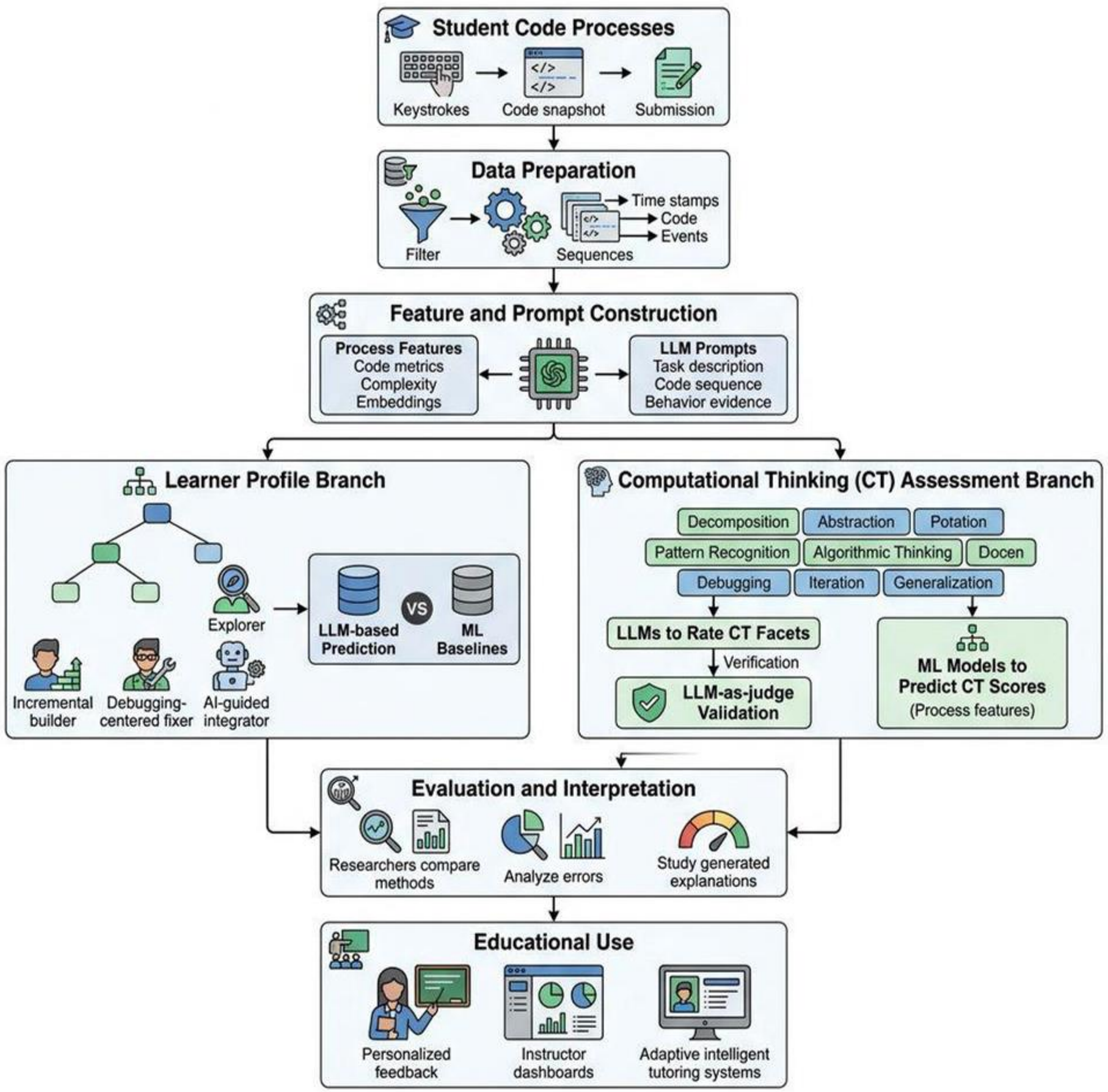
Utah State University

hamid.karimi@usu.edu

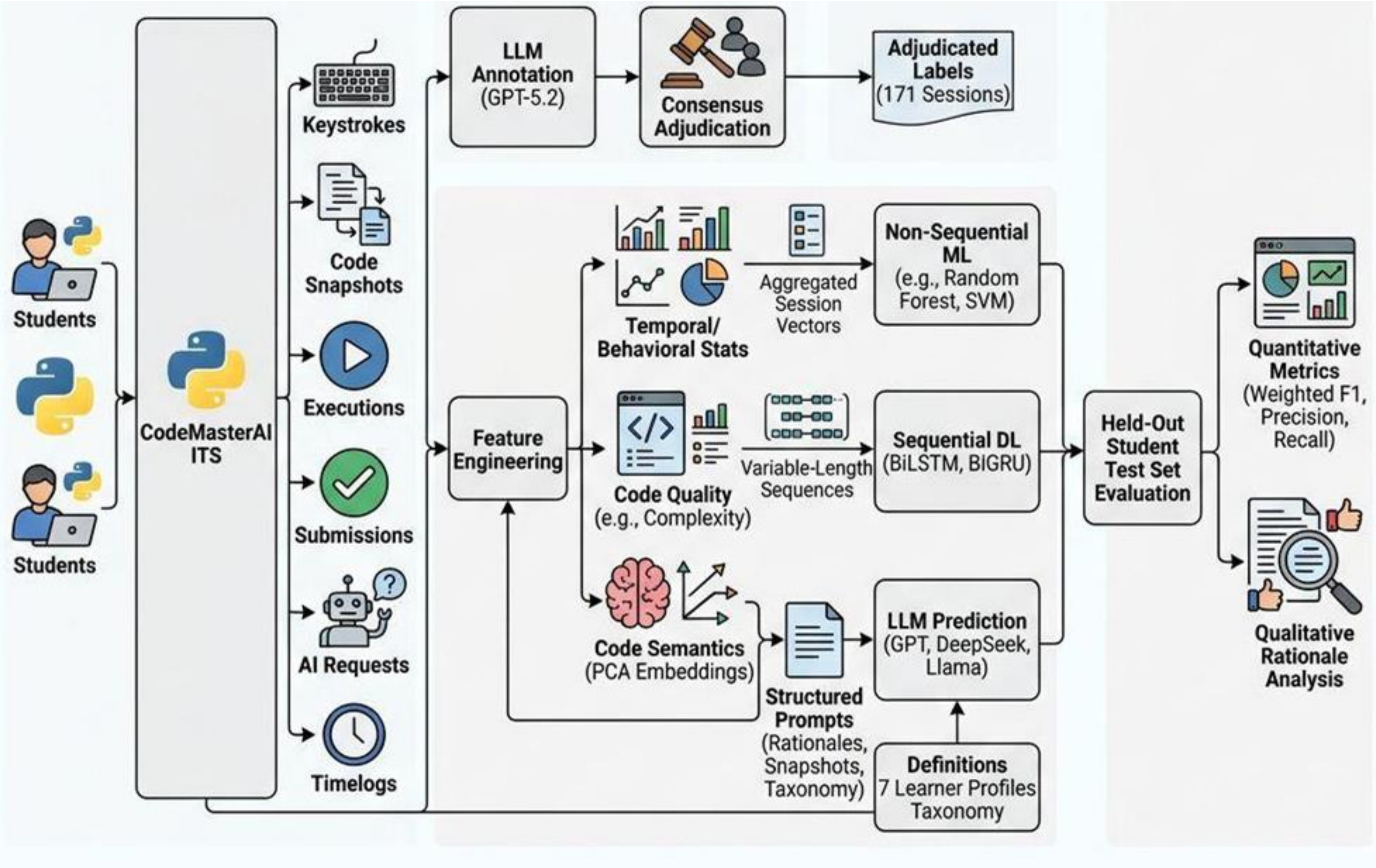
Research Problem

- **Problem:** how LLMs can interpret students' programming processes from keystrokes and code snapshots?

- **Main constructs being inferred:**
  - Learner behavior profiles
  - Computational thinking facets

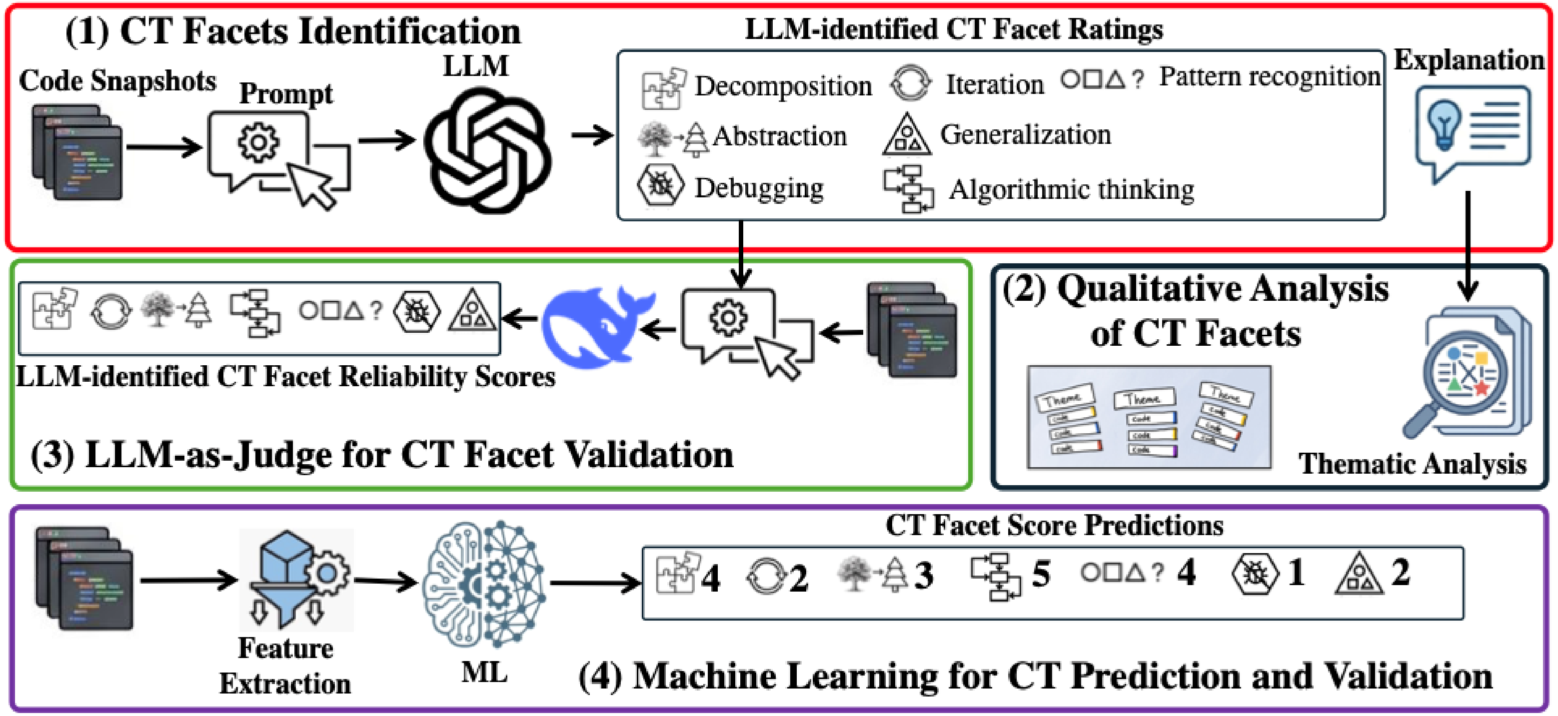


Study 1: Learner Profile Identification from Code Snapshots using LLMs



- LLMs substantially outperformed traditional ML
- LLM rationales captured useful behavioral patterns

Study 2: Computational Thinking Assessment from Code Processes with LLMs



- ➢LLMs produced broadly consistent computational thinking facet profiles
- ➢LLM explanations relied on observable cues such as revision trajectories
- ➢Cross-LLM judging showed moderate to high agreement

Characterizing Teamwork Engagement in STEM

#2219271

Alejandra J. Magana

Purdue University

admagana@purdue.edu

Research Problem and Context

- **Problem:** To promote online (and in-person) **productive** teamwork engagement in the context of large lectures (100-150).

Characterize → Detect → Monitor → Intervene

Constructs	Data Sources
Teamwork engagement	Online recorded teamwork sessions
Teamwork processes	Surveys, self and peer evaluations, contribution charts
Intercultural orientations	Individual reflections and team retrospectives

Use of NLP and AI in Research Workflows

- **Characterizing and Detecting Teamwork Engagement**
  - Qualitative analysis of video data
  - Multimodal AI algorithms (FC-CNN, DenseNet-201, FastText)
- **Identifying Intercultural Orientations**
  - Qualitative Analysis of reflection and retrospective data
  - NLP methods for large dataset analysis (Topic Modeling, RAG, GPT-4.0)
  - Clustering analysis to identify patterns (K-Means)
- **Analyzing Engagement Dynamics Over Time**
  - Survival Analysis (Kaplan-Meier Estimator) on AI-based detection data
  - Qualitative data triangulation for rich descriptions of engagement
- **Promoting Conflict Resolution Training**
  - Simulation-based learning via LLM interaction
  - Qualitative analysis of conversational data

Lessons Learned (special issue in JEE)

- **Maintain human oversight** throughout the research process ("human in the loop").
- **Validate AI outputs** using established reliability and validity procedures.
- **Document AI use transparently**, including models, settings, and applications.
- Use **advanced prompt engineering** and report prompts for reproducibility.
- **Iteratively refine prompts** and document limitations and solutions.
- **Mitigate bias** through explainability techniques and ongoing validation.
- **Protect privacy and confidentiality** through de-identification and secure models.
- **Consider local or open-source LLMs** for greater data control and security.

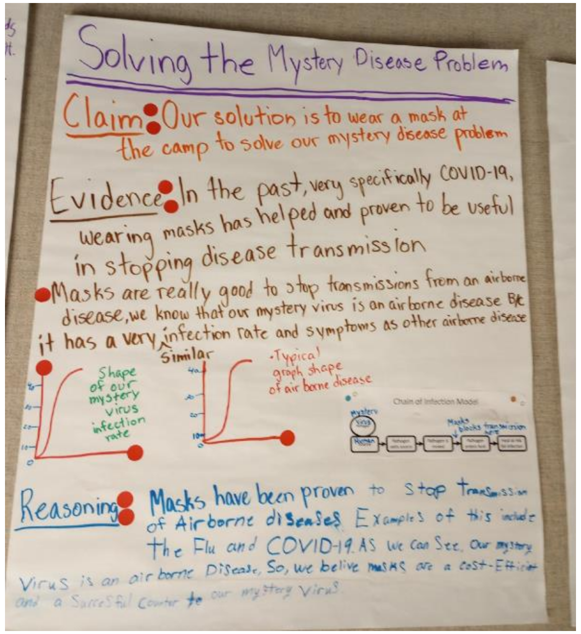
Large Language Models to Analyze Evidence of Science Learning

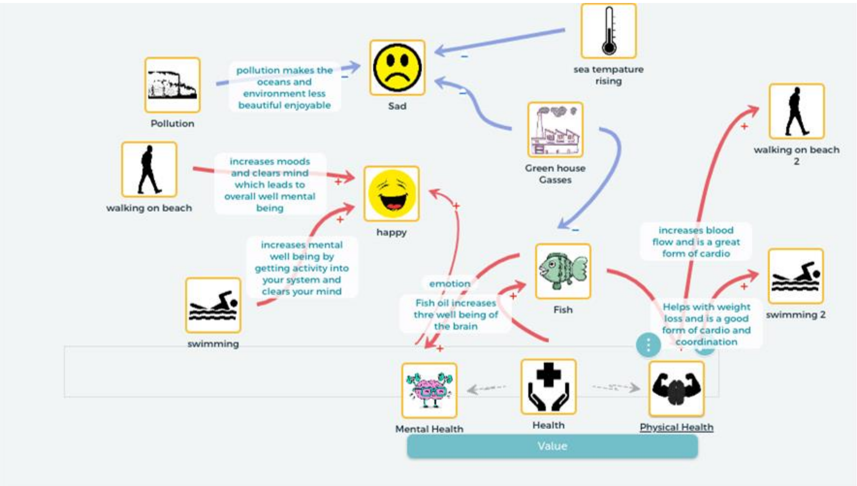
#2422286

Ha Nguyen
University of North Carolina, Chapel Hill
ha.nguyen@unc.edu

Research Problem and Context

- **Problem:** To analyze “found data” in science classrooms, such as analog worksheets, scientific models, and learning behaviors.
- **Rationale:** Learning is not just text-based, but **multimodal**.

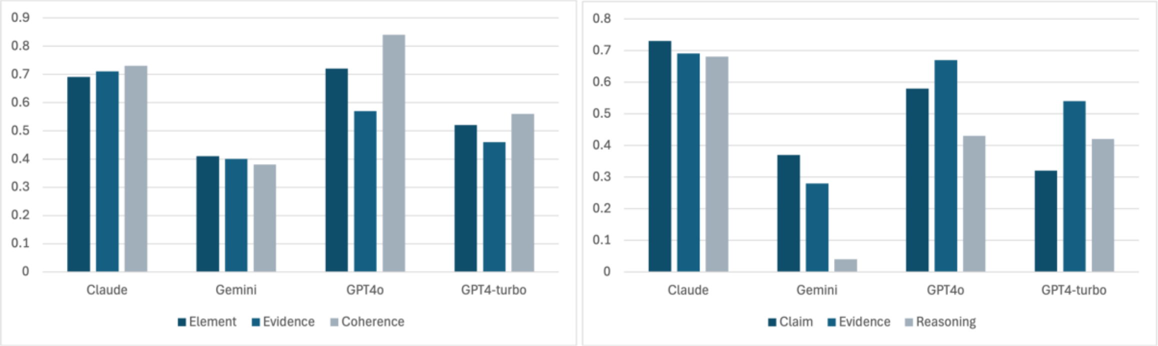




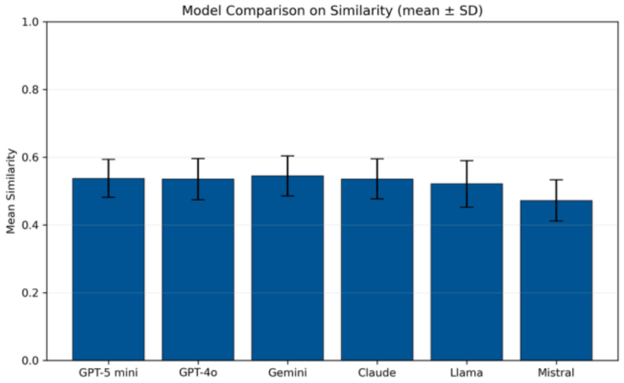
Using LLMs/LMMs to **Analyze Science Learning** and **Inform Science Assessments**

Large Language Models (LLMs) and Large Multimodal Models (LMMs) can:

- **Provide feedback** and **simulate student thinking** in science learning artifacts
- **Analyze** multimodal video data
- **Critique** science assessments in alignment with NGSS (Nguyen & Cao, 2026; Nguyen & Hayward, 2025; Nguyen & Park, 2025; Zhou et al., 2026a, 2026b).



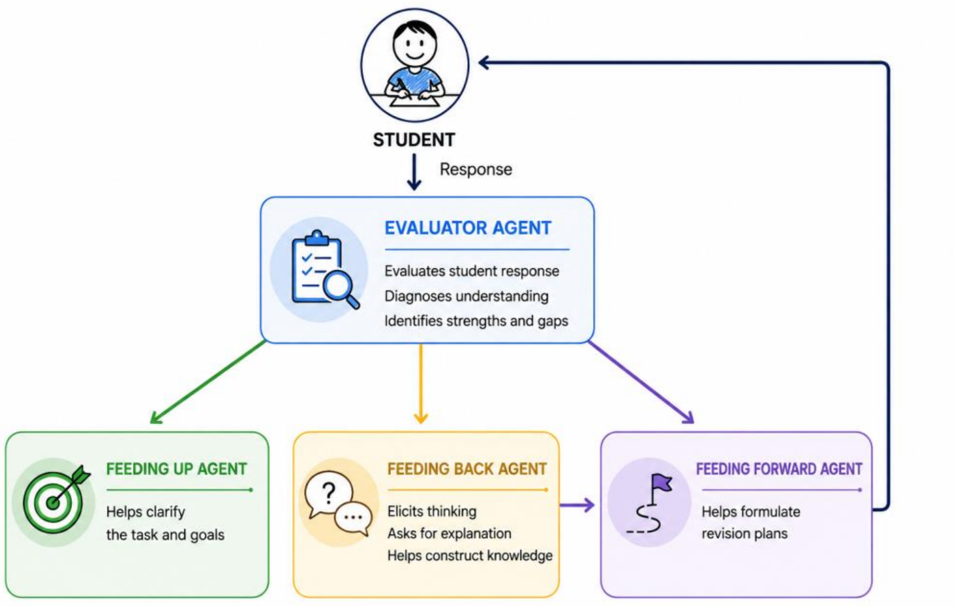
LMMs show **substantial inter-rater agreement** with human coders in scoring science assessments.



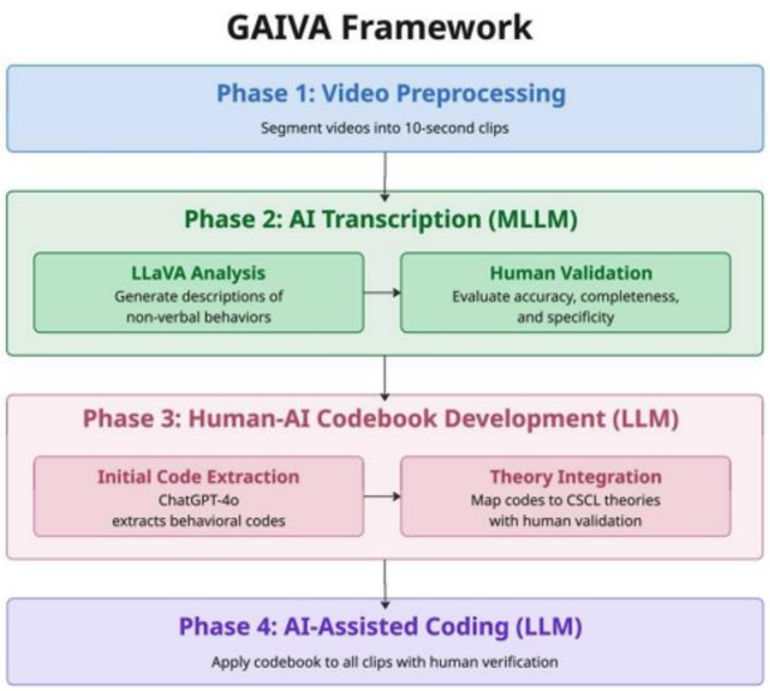
LLMs show **high overlap** with lesson ideas generated by students and teachers.

Lessons Learned

- Consider modularized workflows, systematically test performance across multiple LLMs, and validate with human input.
- Designing these systems prompts us to be more precise about our learning assumptions.



Designing a student-facing feedback system



Using LMMs for video analysis

*Human-in-the-loop.
* Theory alignment.

Lessons Learned

- ❑ **Student code processes are more informative than final submissions alone.**
- ❑ **LLMs can serve as diagnostic models, not just feedback generators.**
- ❑ **Traditional feature-based ML has limits but remains useful for validation.**
- ❑ **LLM rationales are interpretable but require caution.**
    - circular reasoning, facet boundary blurring, correctness halos, construct drift, and over-interpreting visible edits.