SCRIBE: Structured Chain Reasoning for Interactive Behavior Explanations using Tool Calling

Fares Fawzi, Vinitra Swamy, Dominik Glandorf, Tanya Nazaretsky, Tanja Käser
EPFL
{firstname.lastname}@epfl.ch

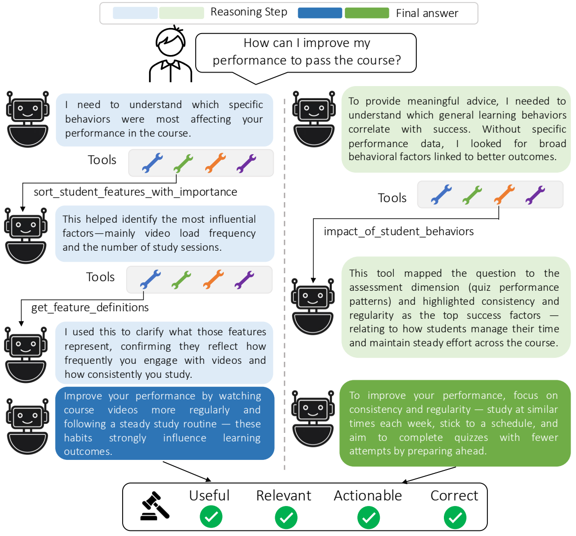
Abstract

Language models can be used to provide interactive, personalized student feedback in educational settings. However, real-world deployment faces three key challenges: privacy concerns, limited computational resources, and the need for pedagogically valid responses. These constraints require small, open-source models that can run locally and reliably ground their outputs in correct information. We introduce SCRIBE, a framework for multi-hop, tool-augmented reasoning designed to generate valid responses to student questions about feedback reports. SCRIBE combines domain-specific tools with a self-reflective inference pipeline that supports iterative reasoning, tool use, and error recovery. We distil these capabilities into 3B and 8B models via two-stage LoRA fine-tuning on synthetic GPT-4o-generated data. Evaluation with a human-aligned GPT-Judge and a user study with 108 students shows that 8B-SCRIBE models achieve comparable or superior quality to much larger models in key dimensions such as relevance and actionability, while being perceived on par with GPT-4o and Llama-3.3 70B by students. These findings demonstrate the viability of SCRIBE for low-resource, privacy-sensitive educational applications.

1 Introduction

Education at scale, in contexts like massive open online courses (MOOCs) or large in-person lecture halls, enables thousands of learners to engage with the same material simultaneously (De Freitas et al., 2015). However, this scale comes at a cost: limited access to personalized guidance, feedback, and support.

Recent progress in Large Language Models (LLMs) offers a promising avenue toward personalized support at scale. LLMs have been applied to a wide range of tasks including question generation (Scaria et al., 2024; Hang et al., 2024; Fawzi et al., 2024; Ma et al., 2024; Liang et al., 2023), grading (Golchin et al., 2025), and automatic feedback generation (Phung et al., 2023; Pardos and Bhandari, 2024; Swamy et al., 2024; Nair et al., 2024).



How can I improve my performance to pass the course?

I need to understand which specific behaviors were most affecting your performance in the course.

This helped identify the most influential factors—mainly video load frequency and the number of study sessions.

I used this to clarify what those features represent, confirming they reflect how frequently you engage with videos and how consistently you study.

Improve your performance by watching course videos more regularly and following a steady study routine — these habits strongly influence learning outcomes.

To provide meaningful advice, I needed to understand which general learning behaviors correlate with success. Without specific performance data, I looked for broad behavioral factors linked to better outcomes.

This tool mapped the question to the assessment dimension (quiz performance patterns) and highlighted consistency and regularity as the top success factors — relating to how students manage their time and maintain steady effort across the course.

To improve your performance, focus on consistency and regularity — study at similar times each week, stick to a schedule, and aim to complete quizzes with fewer attempts by preparing ahead.

Figure 1: Structured multi-hop reasoning for pedagogically valid feedback via tool calls. The question is addressed using distinct reasoning strategies: one model uses multi-step analysis of learner behavior for a personalized advice (*left*), the other links it to effective learning behavior dimensions for general guidance (*right*).

Despite promising results in educational tasks, LLMs face challenges limiting their reliability in real-world use. Hallucinations and factually incorrect explanations can mislead learners and erode trust, especially problematic in education, where responses must be accurate and pedagogically sound (Nazaretsky et al., 2024; Manakul et al., 2023; Kumar et al., 2023; Levonian et al., 2025). A promising direction to mitigate this is retrieval-augmented generation (RAG) (Fang et al., 2025; Dakshit, 2024), or tool augmentation (Wu et al., 2024; Ross et al., 2025; Schick et al., 2023; Patil et al., 2024; Yao et al., 2023; Inaba et al., 2023) where models use external resources or tools to support reasoning and verification. While these methods improve factuality and interpretability, they are

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 29285–29310
November 4-9, 2025 ©2025 Association for Computational Linguistics

more effective in large models (Shen et al., 2024) (such as GPT-4o (OpenAI et al., 2024)), which are costly to run. As a result, there is growing interest in training smaller, open-source models that can run locally and securely (Zhang et al., 2024).

Recent work has explored fine-tuning small models on synthetic tool-calling data (Patil et al., 2024; Schick et al., 2023; Liu et al., 2025; Qin et al., 2023). However, these efforts typically address narrow tasks with short, domain-agnostic prompts and a known, fixed sequence of tool calls (e.g., querying the fuel level of an aircraft). This setup fails to reflect real-world domains like education, where open-ended questions require flexible, multi-step reasoning. As shown in Fig. 1, a question like "How can I improve my performance?" can be answered through different tool-use paths. The provided responses are both pedagogically valid, yet created by distinct reasoning trajectories.

In this work, we propose SCRIBE, a framework for self-reflective, multi-hop tool reasoning in educational feedback scenarios, where models must flexibly use external tools and iteratively revise their outputs to generate pedagogically meaningful responses. We collect real student questions about structured feedback reports and augment them with high-quality synthetic data including reasoning traces, tool calls, and final responses. We fine-tune small open-source models via a two-stage LoRA (Hu et al., 2022) pipeline and implement a self-reflective inference loop that enables iterative reasoning and tool use outperforming or matching larger models. Our evaluation combines automatic assessment using a human-aligned GPT-as-a-judge, alongside a user study with 108 students interacting with feedback across three different MOOCs. Notably, we find students equally rate our SCRIBE-trained 8B model, a much larger Llama-3.3 70B and GPT-4o. Our main contributions are:

1. **We propose SCRIBE, a framework for multi-hop tool reasoning**, where models must flexibly call tools and self-reflect to generate high-quality responses.
2. **We distill tool calling and self-reflection reasoning behavior of a larger model (GPT-4o) into relatively smaller open-source models** through a two-stage LoRA fine-tuning process to enhance reasoning and multi-hop tool calling.
3. **We create a new synthetic dataset of 7000 student performance feedback questions** derived from 28 real-world students with answers, tool calling and reasoning chains.
4. **We design a rubric for interactive feedback evaluation for a human-aligned GPT-as-a-judge**, enabling scalable and consistent evaluation of model responses.
5. **We conduct a real-world interactive user study with 108 university students** assessing perception of interactions with a small SCRIBE 8B model, Llama-3.3 70B, and GPT-4o across distinct reports from three different MOOCs.

We provide our full implementation, open-source dataset, and trained models, enabling reproducibility and further research.[1]

2 Related Work

Tool-Augmented Language Models. Tool calling helps LLMs compensate for missing world knowledge and reduce hallucinations (Komeili et al., 2022; Wang et al., 2024a). Recent work has explored in-context learning and few-shot prompting to encourage reasoning about tool use (Yao et al., 2023; Kim et al., 2024; Shen et al., 2023; Chen et al., 2023b). Prompting techniques like chain-of-thought (CoT) (Wei et al., 2022), and ReAct (Yao et al., 2023) structure intermediate reasoning and improve factuality (as demonstrated by Inaba et al. (2023)), but remain fragile in smaller models and generalize poorly with weak instruction-following.

To enhance tool calling, especially in smaller open-source LLMs, other works have performed finetuning. Toolformer (Schick et al., 2023) uses a self-supervised approach with LLM-generated data to train models to decide when to call APIs. Gorilla (Patil et al., 2024) fine-tunes a LLaMA-based model on GPT-4 instruction–API pairs to generate accurate calls from documentation or internal knowledge. Recent works like ToolLLM (Qin et al., 2023) and ToolACE (Liu et al., 2025) use synthetic data to support multi-tool use for complex tasks. However, tool use is often treated as an end in itself rather than a step toward producing high-quality, correct answers. Despite gains in tool call accuracy, models are rarely trained to reason before and after tool calls, and are seldom evaluated in domain-specific, real-world settings such as educational feedback where clarity, correctness, and user trust are essential. As a result, their responses may often lack coherence, context-awareness, and alignment with user needs.

[1] All resources are available at https://github.com/epfl-ml4ed/SCRIBE.

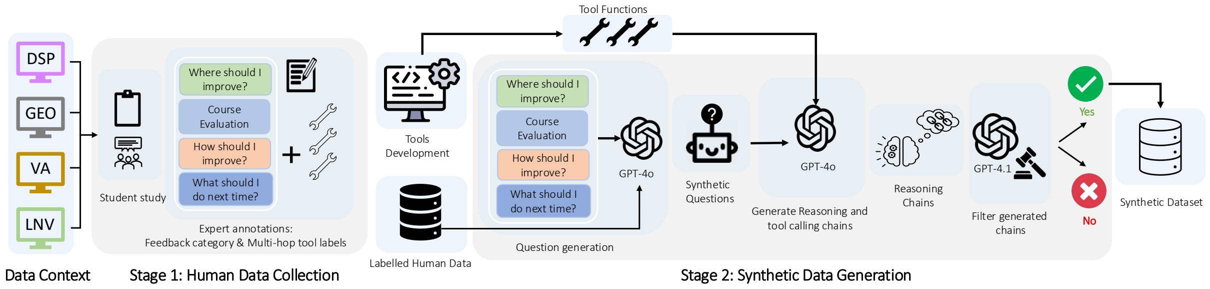


Figure 2: **SCRIBE Data Generation Pipeline**. Synthetic data is generated by collecting questions from students to guide expert annotators in identifying essential tools (Stage 1). GPT-4o generates reasoning chains with these tools, and GPT-4.1 filters the outputs based on actionability, relevance, tool use, and correctness (Stage 2).

LLMs in Education. LLMs are increasingly used in education, enabling natural interactions through conversational agents (Lieb and Goel, 2024; Wolfbauer et al., 2023; Neumann et al., 2024; Pal Chowdhury et al., 2024). Their broad domain knowledge reduces reliance on domain-specific models, supporting applications like personalized learning (Park et al., 2024), knowledge tracing (Neshaei et al., 2024), and automated feedback (Stamper et al., 2024). Prior work has explored various integration strategies, often focusing on *prompting*, e.g., zero-shot prompts for automatic science scoring (Wu et al., 2023) or CoT for classifying learning outcomes via Bloom's taxonomy (Almatrafi and Johri, 2025). Others fine-tune LLMs on educational data, e.g., to recognize epistemic and topic-related dialogue acts in collaborative learning (Acosta et al., 2024) or to score math responses (Morris et al., 2024). Prior work also explored RAG, using textbooks for guidance (Henkel et al., 2024) or student reflections for feedback (Neshaei et al., 2025). However, most models act as *standalone* generators, with few integrating tools for grounded interactions.

3 Methods

Our goal is to enable interactive feedback with small LLMs by using multi-hop tool calling to generate pedagogically meaningful personalized responses. Our framework, SCRIBE, consists of two main phases: (1) Dataset generation (see Fig. 2) and (2) Finetuning and inference (see Fig. 3).

3.1 Dataset Generation Pipeline

Our dataset generation pipeline consists of (1) a user study to identify real student questions and categorize them by pedagogical need, (2) domain-specific tools to support grounded, context-aware answers, (3) synthetic data generation using GPT-4o simulating multi-hop reasoning and tool calls.

3.1.1 Data Context

Our experiments use data from four globally-offered MOOCs at a European university: Digital Signal Processing (DSP), Éléments de Géomatique (GEO), Villes Africaines (VA), and Launching New Ventures (LNV). Each includes weekly video lectures, quizzes, and graded assignments. To analyze student performance, we use feedback reports from iLLuMinaTE (Swamy et al., 2024), a zero-shot LLM-XAI framework that generates social science theory-driven, actionable explanations based on behavioral features predicting pass/fail outcomes. We focus on feedback based on social science theories and post-hoc explainers shown to be highly useful and actionable: Necessity and Robustness selection (NR) (Lipton, 1990; Lombrozo, 2010), Abnormal Conditions (AC) (Hilton and Slugoski, 1986), and Contrastive Explanation (Con) (Hilton, 1990), with Contrastive Explanation Method (CEM) (Dhurandhar et al., 2018) as the explainer.

3.1.2 Human Data Collection

Student Study. To design an interactive feedback system, we first investigated the types of questions students ask when presented with explanation-based feedback. We used five feedback reports from Swamy et al. (2024). Two reports described a student enrolled in DSP (based on the NR and Con theories), two reports belonged to a student from GEO (again one report per theory), and one report was from a student in VA using the AC theory.

We conducted a study with 28 postgraduate STEM students, each randomly assigned one of five reports and given a brief description of the associated MOOC. Participants (1) wrote three follow-up questions about the feedback, (2) rated five GPT-4o-generated questions on a 1–5 scale (5 = very useful), and (3) selected the most useful feedback category, from Mandouit and Hattie (2023): What have I done well?, Where should I improve?, How

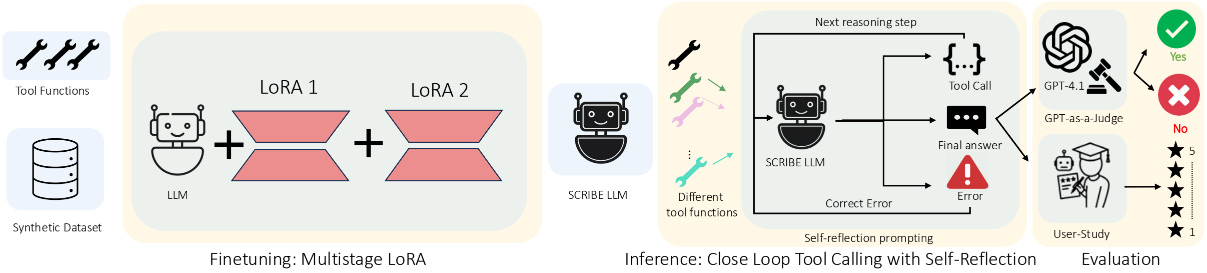


Figure 3: **SCRIBE finetuning, inference, and evaluation pipelines**. Finetuning involves two successive LoRA stages for multi-hop reasoning with tool use. Inference operates as a closed-loop system with self-reflection prompting for error correction. Evaluation combines GPT-as-a-judge assessments and a user study.

should I improve?, and What should I do next time?. All students gave informed consent to participation and the study was approved by the university's human research ethics commission.

Expert Annotations. We manually annotated 75 student-written questions categorizing feedback students seek, using 3 dimensions from Mandouit and Hattie (2023): Where to improve? (*Where*?), How to improve? (*How*?), and What to do next time? (*Next Time*). Our rubric is provided in section G. Two expert annotators independently labeled the questions, achieving substantial agreement (Cohen's $\kappa = 0.67$). During annotation, we identified an additional category, *"Course Evaluation"*, for questions about course structure and assessment. Based on annotations, we derived six tools needed to meaningfully answer these queries.

3.1.3 Tools Development

To be able to answer the students' questions, we developed six different domain-specific tools.

Textbook and Syllabus Retrieval Tools. For course content questions, we used RAG over MOOC materials. Textbook sections and exercises were embedded using the bge-small model (Xiao et al., 2023), enabling query-based retrieval. Syllabi were embedded with the bilingual-embedding-base model (Lajavaness, 2024) for structure-related queries.

Topic Dependency Mapping. To clarify topic dependencies, we constructed skill maps that capture prerequisite relationships. For DSP, we adopted the map from Swamy et al. (2022). For GEO, the instructor provided a custom map. For VA and LNV, we extracted skills from video transcripts using GPT-4o and then re-prompted it to infer dependencies. The VA map was validated by the instructor. Finally, we implemented a function that, given a MOOC name and week, returns the relevant prerequisite weeks. The full set of maps is available in section E.

Grade Calculator. To address performance questions, we designed a function that calculates student total grade from their ID, compares to the passing threshold, and returns the points needed to pass.

Sort Student Features. The tool summarizes student progress using behavioral features from (Swamy et al., 2024), and importance ranked by CEM. For a student and week, it returns 5 most and least important features with raw feature values for context.

Features Description Search. Some student questions focused on unfamiliar terms from feedback reports, derived from features used in student modeling (Swamy et al., 2024). To support these queries and the *Sort Student Features* tool, we developed a function that retrieves feature descriptions. Given a feature name, we use efficient fuzzy string matching for an efficient nearest-neighbor matching and return the corresponding definition.

Student Behavior Impact on Performance. The tool answers hypothetical questions about how behavioral changes affect outcomes (e.g., "Would more consistent engagement improve my grade?"). Given a MOOC name and query, it maps the input to one of five behavioral dimensions (Mejia-Domenzain et al., 2022)—Effort, Consistency, Proactivity, Assessment, and Regularity—linked to features from (Swamy et al., 2024) using CEM-derived importance scores. Queries and feature descriptions are embedded with all-MiniLM-L6-v2 (Reimers and Gurevych, 2019) and matched via cosine similarity. The tool returns the closest dimension and two alternatives, each with a brief definition, helping students assess their behavior's impact and explore other strategies.

3.1.4 Synthetic Data Generation

To generate synthetic questions that closely resemble those written by students, we selected 16 students across three MOOCs (DSP, GEO, and VA) and chose two reports per student, each generated using one of two theories introduced in section 3.1.1 (NR and Con). For each of these reports, we used real student-written questions and insights collected from students in human study section 3.1.2 to construct the prompts for GPT-4o. We generated 20 synthetic questions per feedback category, per report, yielding a rich dataset of approximately 7000, diverse student-like questions.

Using the generated questions, we prompted GPT-4o with a feedback report and a student question to generate structured reasoning followed by an initial tool call. This tool call is executed, and its output is returned to GPT-4o to produce the next reasoning step and either a subsequent tool call or a final answer. This process is repeated until a final answer is produced. Each example thus forms a reasoning trajectory of alternating reasoning and tool interactions, which we automatically filter using GPT-as-a-judge to exclude samples with erroneous reasoning chains or tool misuse. We used examples that the judge marked YES in all categories (details described in section 3.2.3).

To assess the similarity and diversity of the generated questions relative to real student questions, we first compared the distributions of question lengths and removed outliers that were shorter or longer than student responses. Next, we computed the distributions of Shannon entropy (to estimate token-level information content) and perplexity (to approximate linguistic fluency), and compared these between real and synthetic questions using Jensen-Shannon Divergence (JSD). We performed these comparisons across question types and courses. To further assess semantic diversity, we computed pairwise cosine similarity within each dataset (real and synthetic) across all questions, for each course and feedback category. This enabled us to quantify question diversity within each dataset. Next, to evaluate the similarity of the generated questions to real student questions, we compared the embeddings of 76 generated questions (matched to the number of human-authored ones), using the bge-large-en model (Xiao et al., 2023), against embeddings of (a) real student questions from the same reports and (b) randomly selected SQuAD questions (Rajpurkar et al., 2018). We applied Hotelling's T^2 test on 2D representations from t-SNE to compare distributions.

3.2 Inference and Finetuning Pipeline

The objective of this pipeline is to distill GPT-4o tool calling and reasoning capabilities into smaller LLMs through a two-stage LoRA finetuning. Our finetuning and inference pipeline consists of (1) a multi-stage fine-tuning process where relatively small open-source models (e.g., Llama 8B) are trained via LoRA adapters to perform structured reasoning and tool use, and (2) a closed-loop inference pipeline that supports iterative tool use, self-reflection, and error correction.

3.2.1 Multi-Stage LoRA Fine-Tuning

To enhance the reasoning and multi-hop tool use abilities of relatively small open-source models, we distill structured tool-calling behavior from a much larger teacher model (GPT-4o). Inspired by multi-stage instruction tuning and curriculum-style learning (Chen et al., 2023a; Guan et al., 2025; Pang et al., 2024), our training process is divided into two sequential stages that progressively increase task complexity. Each training instance consists of a student query q, a feedback report f, a sequence of reasoning steps $\{r_i\}_{i=0}^n$, tool calls $\{t_i\}_{i=0}^n$, tool outputs $\{o_i\}_{i=0}^n$, and a final answer a.

Stage 1 (Initial Reasoning and Tool Selection). The model is trained to generate an initial reasoning step r_0 and the first tool call t_0 conditioned on (q, f). This teaches the model how to interpret student questions and initiate tool-call reasoning.

$$r_0, t_0 \sim P_{\text{stage1}}(r, t \mid q, f) \tag{1}$$

Stage 2 (Multi-Hop Reasoning and Answer Generation). Conditioned on q, f, the initial tool call t_0 and output o_0, the model learns to iteratively reason and revise its outputs across multiple steps. It produces intermediate reasoning steps r_i, additional tool calls t_i, and the final answer a.

$$\begin{aligned} r_i, t_i &\sim P_{\text{reason}}\left(r, t \mid q, f, \{(r_j, t_j, o_j)\}_{j<i}\right), \\ &\text{for } i = 1, \dots, n \\ a &\sim P_{\text{answer}}\left(a \mid q, f, \{(r_j, t_j, o_j)\}_{j\leq n}\right) \end{aligned} \tag{2}$$

This decomposition ensures the model first learns how to initiate tool-augmented reasoning before handling more complex reasoning trajectories with iterative refinement. We use LoRA adapters for efficient parameter updates in both stages.

3.2.2 Closed-Loop Tool Calling

Inspired by AnyTool (Du et al., 2024) which requeries the tool using a self-reflection loop, we implement self-reflective, multi-hop reasoning as our **prompting framework for inference**, where the model incrementally constructs responses to student questions by interacting with external tools and revising reasoning based on their outputs. We provide the prompts in section H. This task is inherently underdetermined, as different sequences of tool calls may lead to equally valid answers. Our pipeline supports this flexibility while enabling error recovery and iterative refinement.

Formally, for a given student query q and feedback report f, the model produces an initial reasoning step r_0 and a corresponding tool call t_0. The output o_0 from executing t_0 is passed back to the model, which generates the next reasoning step $r_1 = \texttt{Reason}(r_0, o_0, q, f)$, followed optionally by another tool call t_1. This process continues for up to N steps, producing a trajectory:

$$(f, q, r_0, t_0, o_0, r_1, t_1, o_1, \ldots, r_n, a) \tag{3}$$

where a is the final answer and $n < N$. At each step i, the model decides whether to call another tool or produce a final answer, based on the evolving context of the query, feedback report, previous reasoning steps, and tool outputs. This iterative process continues until the model outputs a final answer or reaches a predefined step limit N.

The model may select the same tool repeatedly or switch tools across steps, depending on the evolving context. To improve robustness, the system monitors for tool-call errors or instruction violations (e.g., invalid tools, skipped reasoning). In such cases, the model is re-prompted to self-reflect and revise its reasoning or tool choice. If no valid answer is generated after N iterations, the interaction is terminated and marked as unresolved.

3.2.3 Evaluation

We evaluated the models' responses using expert annotation and a LLM-as-a-judge protocol as well as through a user study with real students.

GPT-as-a-Judge. Given the open-ended task, standard metrics like tool selection accuracy are insufficient, as multiple tool sequences can yield valid answers. We therefore developed a rubric to evaluate both the tool used and the model's student-facing final response. Based on existing literature, we defined four criteria and added a fifth, tool relevance, specific to our setting. The criteria include: (1) **Relevance** to the question (Zheng et al., 2023), (2) **Actionability** in terms of providing concrete advice (Swamy et al., 2024), (3) **Tool Relevance** (whether the selected tools were appropriate), (4) **Spelling and Grammar** (Swamy et al., 2024), and (5) **Correctness** based on factual alignment with tool outputs and feedback (Zheng et al., 2023). The detailed rubric is provided in section C.

In a first step, two researchers independently labeled 60 instances comprising 20 responses, tool calls, and tool outputs from three different models (Llama-3.1 8B base, SCRIBE, and Llama-3.3 70B) sampled across three MOOCs (DSP, GEO, and VA). The annotations achieved an overall Cohen's κ of 0.85, indicating strong inter-rater agreement. To assess the quality of model outputs at scale, we then adopted GPT-4.1 (OpenAI, 2025) as an third evaluator, following prior work on LLM-based judgment for response quality (Liu et al., 2023; Zheng et al., 2023; Qin et al., 2023; Du et al., 2024). Each judgment is generated by prompting GPT-4.1 with a feedback report, student question, a description of available tools, the model's full reasoning trace (with tool calls and outputs), and definitions for each evaluation criterion. We used CoT prompting to encourage step-by-step reasoning before GPT-4.1 returns a binary rating (Yes/No) for each question criterion (Qin et al., 2024). To ensure reliabilty, we ran GPT-4.1 five times, achieving Cohen's $\kappa = 0.818 \pm 0.014$ between the GPT-4.1 judge and the humans. We provide prompts and per criterion inter-annotator agreement in section C.

User Study. To evaluate how students perceive model-generated responses, we conducted a user study comparing a small multi-stage LoRA-tuned model (ToolACE-8B SCRIBE) to two larger LLMs (Llama-3.3 70B and GPT-4o). To reflect deployment constraints where hosting large models may be infeasible for schools, we used API for Llama-3.3 70B and GPT-4o. We recruited 108 students via Prolific[2] (see section F for more details). All participants provided informed consent, and the study was approved by our university's human research ethics commission. Each participant saw three feedback reports (passing and failing students) generated by iLLuMinaTE (Swamy et al., 2024), each from a different MOOC: DSP, GEO, and LNV (hold-out MOOC). The study was designed to ensure that each participant interacted with reports from all three MOOCs and models. We constructed 108 unique combinations, each consisting of one student report per course (drawn from six possible reports per course: 3 passing and 3 failing),

[2] https://www.prolific.com

with each report paired with a different model. Report–model assignments were permuted to ensure that each model was used exactly once within each combination and to prevent ordering effects.

Participants posed 3–5 unrestricted questions per report to have natural conversations. After each conversation, participants rate the model's responses on a 5-point scale (1 is lowest and 5 is highest) across five criteria from prior work (Swamy et al., 2024; Frej et al., 2024): (1) **Relevance**: Response directly addresses the question. (2) **Usefulness**: Response provides meaningful insights that answer the question and that can enhance learning or deepen understanding. (3) **Actionability**: Response provides clear steps or instructions. (4) **Coverage**: Response comprehensively addresses all components of questions asked, including subquestions. (5) **Conciseness**: Response is clear, and complete with minimal redundancy.

At the end of the study, participants reviewed the three full conversations side by side and selected their overall preferred interaction and provided the reasons for their preference in an open text field.

Generalisation to Unseen Tools. To assess whether SCRIBE can extend its tool-use behaviour beyond those seen in training, we introduced a new tool, `web_search`, designed to retrieve online resources. We evaluated generalisation in two ways. First, we used 27 GPT-4o–generated questions specifically constructed to test whether the model could invoke the unseen `web_search` tool after training on a different set of tools. Second, we used the same 192-question test set employed for model evaluations, spanning the three MOOCs (DSP, GEO, VA), and augmented the existing tool set with `web_search` as an extra unseen tool. We then compared our ToolACE-8B-SCRIBE with the base ToolACE-8B in a zero-shot setting.

4 Results

We conducted a series of experiments to evaluate the quality of the synthetic data used to train SCRIBE, the response quality of the model through a quantitative analysis, and student perception of its outputs through a user study.

Experimental Protocol. We finetuned and evaluated three small models: Llama-3.2 3B and Llama-3.1 8B, which natively support tool calling, and ToolACE-8B (Liu et al., 2025), an 8B model that achieves state-of-the-art performance on the Berkeley Function Calling Leaderboard (BFCL) (Patil et al., 2025), and was able to follow our self-reflection and reasoning instructions. The finetuning required six A100 GPU hours per stage. We compared the small models to GPT-4o (gold standard) and Llama-3.3 70B. All small models were finetuned on 7,000 generated questions (see section 3.1.4) with corresponding tool-use and reasoning chains (see section 3.2.1). Our self-reflection inference pipeline was applied uniformly across models for fair comparison. Evaluation was conducted on 192 test questions, including 75 written by real students and additional synthetic questions (unseen in fine-tuning) used to balance coverage across three MOOCs (DSP, GEO, VA) and four categories (How, Where, Next Time, Course Evaluation). We also evaluated on 192 additional held-out questions from the LNV MOOC which was not included in the fine-tuning. For the Llama-3.1 8B and ToolACE-8B models, we achieved best results with LoRAs of rank of 256 (see ablations section B). We used LoRAs of rank of 128 for Llama-3.2 3B.

4.1 Synthetic student questions closely match real student questions

To evaluate the quality and variety of GPT-4o-generated questions, we compared them to real student-written questions. Table 1 shows the JSD for the Shannon entropy and for perplexity between student and generated questions as well as cosine similarity within each dataset. We observe that all JSD values are < 0.387, indicating that the generated questions are reasonably close to human questions in both entropy and perplexity. Among the MOOCs, the lowest divergence in entropy was observed in GEO (entropy JSD = 0.114 ± 0.076), while the highest was in VA (entropy JSD = 0.335 ± 0.144), suggesting more distinctive phrasing in student-written questions for that course. For perplexity, VA had the lowest divergence (0.140 ± 0.093), indicating strong alignment in fluency. Across question categories, *"Next Time"* questions diverged the most (entropy JSD = 0.387 ± 0.089 and perplexity JSD = 0.211 ± 0.064), likely due to the high variability and learner-specific nature of next-step feedback questions (Mandouit and Hattie, 2023). The pairwise cosine similarity was slightly higher among generated questions in GEO and DSP and categories *How?* and *Where?*, indicating slightly less variation. However, overlapping standard deviations suggest that both generated and human questions exhibit comparable diversity.

Complementing these distributional metrics, Table 2 reports Hotelling's T^2 test results on t-SNE

Table 1: Jensen-Shannon Divergence (JSD) and pairwise cosine similarity between human and generated questions across MOOCs and question categories.

Group	Type	JSD (Entropy)	JSD (Perplexity)	Pairwise Cosine Similarity	
				Generated	Human
MOOC	GEO	0.114 ±0.076	0.202 ±0.079	0.265 ±0.034	0.238 ±0.024
	DSP	0.327 ±0.079	0.212 ±0.095	0.279 ±0.044	0.265 ±0.029
	VA	0.335 ±0.144	0.140 ±0.093	0.280 ±0.064	0.307 ±0.047
Question Category	How?	0.180 ±0.093	0.184 ±0.095	0.241 ±0.035	0.234 ±0.034
	Where?	0.242 ±0.121	0.152 ±0.075	0.272 ±0.046	0.249 ±0.021
	Next Time	0.387 ±0.089	0.211 ±0.064	0.271 ±0.052	0.319 ±0.026

Table 2: Synthetic vs. human question similarity. Left: descriptive statistics of t-SNE (2D) embeddings per questions source (Generated, Human, Random). Right: Hotelling's T^2 tests with F and p values for pairwise comparisons.

Descriptive (t-SNE 2D)				Hotelling's T^2 Tests			
Metric	**Generated**	**Human**	**Random**	**Pair**	$\mathbf{T^2}$	**F**	**p**
Centroid (x, y)	$[-2.19,\ 9.85]$	$[-0.32,\ 8.88]$	$[3.22,\ -15.03]$	Gen vs. Human	2.99	1.49	0.229
STD (x, y)	$[7.64,\ 8.85]$	$[7.78,\ 6.96]$	$[8.62,\ 8.64]$	Gen vs. Random	484.62	241.28	1.11×10^{-16}
SEM (x, y)	$[0.71,\ 0.82]$	$[0.89,\ 0.80]$	$[0.79,\ 0.79]$	Human vs. Random	440.34	219.04	1.11×10^{-16}

embeddings. Generated and human questions are not significantly different ($p = 0.229$), whereas both differ significantly from random questions ($p \approx 10^{-16}$). Their centroids also cluster closely in t-SNE space, further confirming that GPT-4o-generated questions align with real student questions while remaining distinct from unrelated out-of-domain data.

> GPT-4o-generated questions closely match real student ones in fluency, content, and diversity, validating them as high-quality training data.

4.2 SCRIBE achieves the performance of significantly larger models

The top plot in Fig. 4 shows evaluation results on the test dataset from GEO, DSP, and VA. Across these courses, fine-tuned SCRIBE models significantly outperform their base versions on relevance, actionability, and tool relevance, with no significant difference in correctness (see Table 4 in appendix A for full test results). ToolACE-8B-SCRIBE and Llama-3.1 8B-SCRIBE both surpass the much larger Llama-3.3 70B in actionability, and match it on relevance and correctness. The 70B model remains significantly stronger on tool relevance, while improvements in correctness remain modest overall.

The bottom plot of Fig. 4 shows results on LNV (an unseen MOOC), where a similar pattern holds. SCRIBE models again significantly outperform their base counterparts on all criteria except correctness. Relative to the 70B model, 8B SCRIBE models show no significant difference in relevance or actionability, but the 70B remains stronger in tool relevance and correctness. These findings confirm that SCRIBE finetuning yields statistically robust gains over base models and narrows the gap to much larger systems. Detailed statistical test results are provided in Table 5 in appendix A. All models achieved a perfect score on the spelling and grammar criterion, we therefore omitted this category in the Figures.

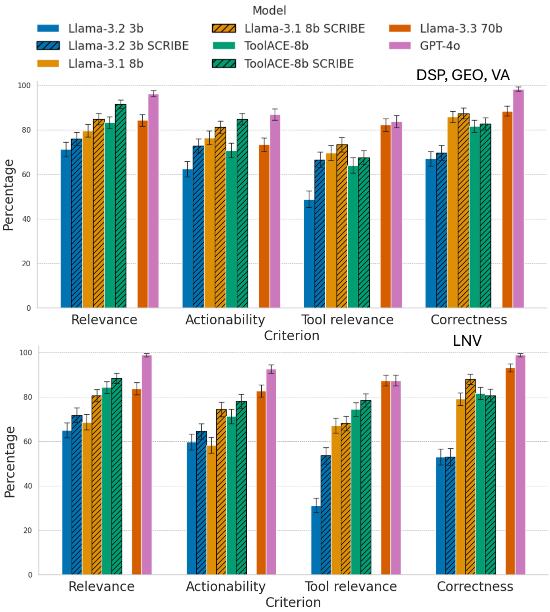


Figure 4: Percentage of YES given by GPT-Judge for each criterion on a holdout dataset of GEO, DSP and VA MOOCs (top) and a holdout set of LNV MOOC (bottom). Hashed bars indicate SCRIBE models

> SCRIBE-trained models significantly outperform their base versions on relevance, actionability, and tool relevance, while matching much larger models in relevance and actionability.

4.3 Students rate SCRIBE responses highly

Fig. 5 shows the average ratings per criterion for each model included in the user study. We observe that the ratings across all five criteria are highly similar across models. Despite the SCRIBE model being significantly smaller in size (8B vs. 70B), students perceive its response quality as on par with much larger models. To test whether any observed differences in ratings were statistically significant, we conducted a one-way ANOVA for each criterion across the three models. In all cases, we failed to reject the null hypothesis ($p > 0.05$), indicating no significant difference in perceived response quality (see appendix F.3 for ANOVA results).

When students were asked to select their preferred conversation and explain why, 47.2% chose GPT-4o, while the remaining responses were evenly split between Llama-3.3 70B and ToolACE-SCRIBE. Among those who preferred GPT-4o, about 25% cited its detailed explanations as the main reason. Others highlighted its actionable advice and clarity. In contrast, 32.1% of students who preferred ToolACE-SCRIBE praised its conciseness. One participant stated: *"The feedback provided clear and direct answers to all my questions in a precise and concise manner, making it easy to understand what I'm doing well."*.

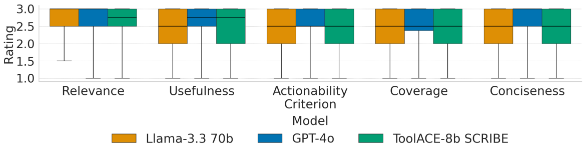


Figure 5: Average ratings from 108 students (1–5 scale) for LLama-3.3 70B, GPT-4o and ToolAce-8B SCRIBE.

Students rate Relevance, Usefulness, Actionability, Coverage, and Conciseness of the **SCRIBE** model on par with larger API-based models, validating its use in low-resource, privacy-sensitive educational settings.

4.4 SCRIBE generalises to unseen tools

On 27 GPT-4o–generated questions specifically designed to trigger the unseen `web_search` tool, ToolACE-8B SCRIBE invoked it 9 times, showing that the model can generalise tool-use behaviour in a zero-shot setting. On the original 192-question dataset (DSP, GEO, VA) with `web_search` available, ToolACE-8B-SCRIBE used the tool 25 times compared to 7 times for the ToolACE-8B, indicating generalization of tool-use behaviour.

As shown in Table 3, both models employed the new tool despite no prior exposure, with SCRIBE achieving higher Tool Relevance. However, introducing `web_search` led to a slight drop in performance across metrics for both models relative to their runs without it. This is likely due to the larger action space and added ambiguity, since most questions in the 192-question set did not require this new tool. We also examined how the original tools were used and found that each was invoked at least once, underscoring that all were necessary to address student questions. Full results are reported in Appendix I.

Group	Type	Relevance	Actionability	Tool Relevance	Correctness
Web Search	ToolACE-8B SCRIBE-27-Trigger-Qs	85.19 ± 6.95	92.59 ± 5.07	77.78 ± 7.60	77.78 ± 7.70
	ToolACE-8B-Original-192-Qs	81.25 ± 2.80	73.44 ± 3.20	64.58 ± 3.39	76.04 ± 2.90
	ToolACE-8B SCRIBE-Original-192-Qs	83.33 ± 2.71	72.40 ± 3.27	73.96 ± 3.24	76.56 ± 2.95
No Web Search	ToolACE-8B-Original-192-Qs	83.33 ± 2.80	70.83 ± 3.31	64.06 ± 3.52	81.77 ± 2.81
	ToolACE-8B SCRIBE-Original-192-Qs	91.67 ± 1.88	84.90 ± 2.57	67.71 ± 3.45	82.81 ± 2.76

Table 3: GPT-as-Judge evaluation on the 27 new questions that are designed to trigger the (web_search) tool (27-Trigger-Qs), and on the original with DSP, GEO, and VA (Original-192-Qs) after introducing the new tool

SCRIBE demonstrates zero-shot generalisation by successfully invoking `web_search`, a tool not seen during training.

5 Conclusion

We introduce **SCRIBE**, a framework for interactive student behavior explanations that combines synthetic data generation, two-stage LoRA fine-tuning, and automatic evaluation with a human-aligned GPT-as-a-Judge. SCRIBE enables small language models to perform self-reflective, multi-hop tool-calling in domains with multiple valid tool-use paths. In education, SCRIBE-trained models consistently outperform base models in relevance, actionability, and tool relevance, while 8B-SCRIBE models match or exceed much larger ones in relevance and actionability, key dimensions of student-centered feedback. A user study with 108 students confirmed they are perceived as equally helpful, relevant, and actionable as larger models. These results show that synthetic data and staged fine-tuning can distill complex tool use into smaller, privacy-preserving educational assistants. Future work will extend SCRIBE to additional models and contexts, and focus on improving correctness and tool relevance. One possible context is medical and psychiatric diagnosis where different diagnostic paths are valid and lead to the same diagnosis (Alarcón, 2009; Maung, 2016; The National Academies of Sciences et al., 2015).

6 Limitations

While SCRIBE advances small LLMs on interactive student feedback, multihop reasoning, and tool-calling, there is room for further improvement. Specifically, gains in correctness remain limited due to the already strong performance of the base models, and tool relevance is another challenging criterion since it depends heavily on the model's initial reasoning. Moreover, while our user study found no perceived difference in the quality of responses between SCRIBE models and much larger

API-based models such as Llama-3.3 70B and GPT-4o, we did not evaluate the impact of these models on actual educational outcomes. Assessing how interaction with our system influences student learning and performance remains an important direction for future work.

7 Acknowledgments

We acknowledge that the use of AI assistants (ChatGPT) was limited to polishing the language of the original paper. It was used solely for proofreading and refining grammar, spelling, and phrasing.

References

Halim Acosta, Seung Lee, Haesol Bae, Chen Feng, Jonathan Rowe, Krista Glazewski, Cindy Hmelo-Silver, Bradford Mott, and James C. Lester. 2024. Recognizing multi-party epistemic dialogue acts during collaborative game-based learning using large language models. *International Journal of Artificial Intelligence in Education*, pages 1–25.

Renato D. Alarcón. 2009. Culture, cultural factors and psychiatric diagnosis: Review and projections. *World Psychiatry*, 8(3):131–139.

Omaima Almatrafi and Aditya Johri. 2025. Leveraging generative ai for course learning outcome categorization using bloom's taxonomy. *Computers and Education: Artificial Intelligence*, 8:100404.

Yuxian Chen and 1 others. 2023a. Curriculum learning for natural language understanding. *arXiv preprint arXiv:2305.14067*.

Zhipeng Chen, Kun Zhou, Beichen Zhang, Zheng Gong, Xin Zhao, and Ji-Rong Wen. 2023b. ChatCoT: Tool-augmented chain-of-thought reasoning on chat-based large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14777–14790, Singapore. Association for Computational Linguistics.

Sagnik Dakshit. 2024. Faculty perspectives on the potential of rag in computer science higher education. In *Proceedings of the 25th Annual Conference on Information Technology Education*, SIGITE '24, page 19–24, New York, NY, USA. Association for Computing Machinery.

Sara Isabella De Freitas, John Morgan, and David Gibson. 2015. Will moocs transform learning and teaching in higher education? engagement and course retention in online learning provision. *British journal of educational technology*, 46(3):455–471.

Amit Dhurandhar, Pin-Yu Chen, Ronny Luss, Chun-Chen Tu, Paishun Ting, Karthikeyan Shanmugam, and Payel Das. 2018. Explanations based on the missing: Towards contrastive explanations with pertinent negatives. *Advances in neural information processing systems*, 31.

Yu Du, Fangyun Wei, and Hongyang Zhang. 2024. Anytool: self-reflective, hierarchical agents for large-scale api calls. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Ke Fang, Ci Tang, and Jing Wang. 2025. Evaluating simulated teaching audio for teacher trainees using rag and local llms. *Scientific Reports*, 15(1):3633.

Fares Fawzi, Sarang Balan, Mutlu Cukurova, Emine Yilmaz, and Sahan Bulathwela. 2024. Towards human-like educational question generation with small language models. In *Artificial Intelligence in Education. Posters and Late Breaking Results, Workshops and Tutorials, Industry and Innovation Tracks, Practitioners, Doctoral Consortium and Blue Sky*, pages 295–303, Cham. Springer Nature Switzerland.

Jibril Frej, Neel Shah, Marta Knezevic, Tanya Nazaretsky, and Tanja Käser. 2024. Finding paths for explainable mooc recommendation: A learner perspective. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*, LAK '24, page 426–437, New York, NY, USA. Association for Computing Machinery.

Shahriar Golchin, Nikhil Garuda, Christopher Impey, and Matthew Wenger. 2025. Grading massive open online courses using large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 3899–3912, Abu Dhabi, UAE. Association for Computational Linguistics.

Changhao Guan, Chao Huang, Hongliang Li, You Li, Ning Cheng, Zihe Liu, Yufeng Chen, Jinan Xu, and Jian Liu. 2025. Multi-stage LLM fine-tuning with a continual learning setting. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5484–5498, Albuquerque, New Mexico. Association for Computational Linguistics.

Ching Nam Hang, Chee Wei Tan, and Pei Duo Yu. 2024. Mcqgen: A large language model-driven mcq generator for personalized learning. *IEEE Access*, 12:102261–102273.

Owen Henkel, Zach Levonian, Chenglu Li, and Millie Postle. 2024. Retrieval-augmented generation to improve math question-answering: Trade-offs between groundedness and human preference. In *Proceedings of the 17th International Conference on Educational Data Mining*, pages 315–320.

Denis Hilton. 1990. Conversational processes and causal explanation. *Psychological Bulletin*, 107:65–81.

Denis Hilton and Ben Slugoski. 1986. Knowledge-based causal attribution. the abnormal conditions focus model. *Psychological Review*, 93:75–88.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Tatsuro Inaba, Hirokazu Kiyomaru, Fei Cheng, and Sadao Kurohashi. 2023. MultiTool-CoT: GPT-3 can use multiple external tools with chain of thought prompting. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1522–1532, Toronto, Canada. Association for Computational Linguistics.

Sehoon Kim, Suhong Moon, Ryan Tabrizi, Nicholas Lee, Michael W Mahoney, Kurt Keutzer, and Amir Gholami. 2024. An llm compiler for parallel function calling. In *Forty-first International Conference on Machine Learning*.

Mojtaba Komeili, Kurt Shuster, and Jason Weston. 2022. Internet-augmented dialogue generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8460–8478, Dublin, Ireland. Association for Computational Linguistics.

Sachin Kumar, Vidhisha Balachandran, Lucille Njoo, Antonios Anastasopoulos, and Yulia Tsvetkov. 2023. Language generation models can cause harm: So what can we do about it? an actionable survey. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3299–3321, Dubrovnik, Croatia. Association for Computational Linguistics.

Lajavaness. 2024. bilingual-embedding-base. `https://huggingface.co/Lajavaness/bilingual-embedding-base`. Accessed: 2025-09-18; Apache-2.0 License; based on XLM-RoBERTa with English-French sentence embeddings.

Zachary Levonian, Owen Henkel, Chenglu Li, Millie-Ellen Postle, and 1 others. 2025. Designing safe and relevant generative chats for math learning in intelligent tutoring systems. *Journal of Educational Data Mining*, 17(1).

Zhenwen Liang, Wenhao Yu, Tanmay Rajpurohit, Peter Clark, Xiangliang Zhang, and Ashwin Kalyan. 2023. Let GPT be a Math Tutor: Teaching Math Word Problem Solvers with Customized Exercise Generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Anna Lieb and Toshali Goel. 2024. Student interaction with newtbot: An llm-as-tutor chatbot for secondary physics education. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pages 1–8.

Peter Lipton. 1990. Contrastive explanation. *Royal Institute of Philosophy Supplement*, 27:247–266.

Weiwen Liu, Xu Huang, Xingshan Zeng, xinlong hao, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, Zhengying Liu, Yuanqing Yu, Zezhong WANG, Yuxian Wang, Wu Ning, Yutai Hou, Bin Wang, Chuhan Wu, Wang Xinzhi, Yong Liu, Yasheng Wang, and 8 others. 2025. Toolace: Winning the points of llm function calling. In *International Conference on Representation Learning*, volume 2025, pages 41359–41381.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Tania Lombrozo. 2010. Causal–explanatory pluralism: How intentions, functions, and mechanisms influence causal ascriptions. *Cognitive Psychology*, 61(4):303–332.

Qianou Ma, Hua Shen, Kenneth Koedinger, and Sherry Tongshuang Wu. 2024. How to teach programming in the ai era? using llms as a teachable agent for debugging. In *International Conference on Artificial Intelligence in Education*, pages 265–279. Springer.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore. Association for Computational Linguistics.

Luke Mandouit and John Hattie. 2023. Revisiting "the power of feedback" from the perspective of the learner. *Learning and Instruction*, 84:101718.

Hane Htut Maung. 2016. Diagnosis and causal explanation in psychiatry. *Studies in History and Philosophy of Science Part C: Studies in History and Philosophy of Biological and Biomedical Sciences*, 60:15–24.

Paola Mejia-Domenzain, Mirko Marras, Christian Giang, and Tanja Käser. 2022. Identifying and comparing multi-dimensional student profiles across flipped classrooms. In *Artificial Intelligence in Education*, pages 90–102, Cham. Springer International Publishing.

Wesley Morris, Langdon Holmes, Joon Suh Choi, and Scott Crossley. 2024. Automated scoring of constructed response items in math assessment using large language models. *International Journal of Artificial Intelligence in Education*.

Inderjeet Jayakumar Nair, Jiaye Tan, Xiaotian Su, Anne Gere, Xu Wang, and Lu Wang. 2024. Closing the loop: Learning to generate writing feedback via language model simulated student revisions. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16636–16657, Miami, Florida, USA. Association for Computational Linguistics.

Tanya Nazaretsky, Paola Mejia-Domenzain, Vinitra Swamy, Jibril Frej, and Tanja Käser. 2024. Ai or human? evaluating student feedback perceptions in higher education. In *Technology Enhanced Learning for Inclusive and Equitable Quality Education*, pages 284–298, Cham. Springer Nature Switzerland.

Seyed Parsa Neshaei, Richard Lee Davis, Adam Hazimeh, Bojan Lazarevski, Pierre Dillenbourg, and Tanja Käser. 2024. Towards modeling learner performance with large language models. *arXiv preprint arXiv:2403.14661*.

Seyed Parsa Neshaei, Matea Tashkovska, Paola Mejia-Domenzain, Thiemo Wambsganss, and Tanja Käser. 2025. User-centric reflective writing assistance: Leveraging rag for enhanced personalized support. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pages 1–8.

Alexander Tobias Neumann, Yue Yin, Sulayman Sowe, Stefan Decker, and Matthias Jarke. 2024. An llm-driven chatbot in higher education for databases and information systems. *IEEE Transactions on Education*.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, and 401 others. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

OpenAI. 2025. Introducing gpt-4.1 in the api. Accessed: 2025-05-12.

Sankalan Pal Chowdhury, Vilém Zouhar, and Mrinmaya Sachan. 2024. Autotutor meets large language models: A language model tutor with rich pedagogy and guardrails. In *Proceedings of the Eleventh ACM Conference on Learning @ Scale*, L@S '24, page 5–15, New York, NY, USA. Association for Computing Machinery.

Wei Pang, Chuan Zhou, Xiao-Hua Zhou, and Xiaojie Wang. 2024. Phased instruction fine-tuning for large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5735–5748, Bangkok, Thailand. Association for Computational Linguistics.

Zachary A. Pardos and Shreya Bhandari. 2024. Chatgpt-generated help produces learning gains equivalent to human tutor-authored help on mathematics skills. *PLOS ONE*, 19(5):1–18.

Minju Park, Sojung Kim, Seunghyun Lee, Soonwoo Kwon, and Kyuseok Kim. 2024. Empowering personalized learning through a conversation-based tutoring system with student modeling. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pages 1–10.

Shishir G. Patil, Huanzhi Mao, Charlie Cheng-Jie Ji, Fanjia Yan, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. 2025. The berkeley function calling leaderboard (bfcl): From tool use to agentic evaluation of large language models. In *Forty-second International Conference on Machine Learning*.

Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. 2024. Gorilla: Large language model connected with massive apis. In *Advances in Neural Information Processing Systems*, volume 37, pages 126544–126565. Curran Associates, Inc.

Tung Phung, José Pablo Cambronero, Sumit Gulwani, Tobias Kohn, Rupak Majumdar, Adish Kumar Singla, and Gustavo Soares. 2023. Generating High-Precision Feedback for Programming Syntax Errors using Large Language Models. In *Proceedings of the International Conference on Educational Data Mining (EDM)*.

Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuansheng Wu, Fei Liu, Pengfei Liu, and Dong Yu. 2024. Infobench: Evaluating instruction following ability in large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 13025–13048.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, and 1 others. 2023. Toolllm: Facilitating large language models to master 16000+ real-world apis. *arXiv preprint arXiv:2307.16789*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Hayley Ross, Ameya Sunil Mahabaleshwarkar, and Yoshi Suhara. 2025. When2Call: When (not) to call tools. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3391–3409, Albuquerque, New Mexico. Association for Computational Linguistics.

Nicy Scaria, Suma Dharani Chenna, and Deepak Subramani. 2024. Automated educational question generation at different bloom's skill levels using large language models: Strategies and evaluation. In *International Conference on Artificial Intelligence in Education*, pages 165–179. Springer.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessí, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: language models can teach themselves to use tools. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Weizhou Shen, Chenliang Li, Hongzhan Chen, Ming Yan, Xiaojun Quan, Hehong Chen, Ji Zhang, and Fei Huang. 2024. Small LLMs are weak tool learners: A multi-LLM agent. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16658–16680, Miami, Florida, USA. Association for Computational Linguistics.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. *Advances in Neural Information Processing Systems*, 36:38154–38180.

John Stamper, Ruiwei Xiao, and Xinying Hou. 2024. Enhancing llm-based feedback: Insights from intelligent tutoring systems and the learning sciences. In *International Conference on Artificial Intelligence in Education*, pages 32–43. Springer.

Vinitra Swamy, Bahar Radmehr, Natasa Krco, Mirko Marras, and Tanja KÃ¤ser. 2022. Evaluating the explainers: Black-box explainable machine learning for student success prediction in MOOCs. In *Proceedings of the 15th International Conference on Educational Data Mining*, pages 98–109, Durham, United Kingdom. International Educational Data Mining Society.

Vinitra Swamy, Davide Romano, Bhargav Srinivasa Desikan, Oana-Maria Camburu, and Tanja Käser. 2024. From explanations to action: A zero-shot, theory-driven llm framework for student performance feedback. *Preprint*, arXiv:2409.08027.

Engineering The National Academies of Sciences, Medicine, Institute of Medicine, Board on Health Care Services, Committee on Diagnostic Error in Health Care, The National Academies Of Sciences Engineering, and Medicine. 2015. *Improving diagnosis in health care*. National Academies Press.

Hongru Wang, Yujia Qin, Yankai Lin, Jeff Z. Pan, and Kam-Fai Wong. 2024a. Empowering large language models: Tool learning for real-world interaction. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 2983–2986, New York, NY, USA. Association for Computing Machinery.

Zhenhailong Wang, Shaoguang Mao, Wenshan Wu, Tao Ge, Furu Wei, and Heng Ji. 2024b. Unleashing the emergent cognitive synergy in large language models: A task-solving agent through multi-persona self-collaboration. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 257–279, Mexico City, Mexico. Association for Computational Linguistics.

Jason Wei and 1 others. 2022. Chain of thought prompting elicits reasoning in large language models. *NeurIPS*.

Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C. Schmidt. 2023. A prompt pattern catalog to enhance prompt engineering with chatgpt. In *Proceedings of the 30th Conference on Pattern Languages of Programs*, PLoP '23, USA. The Hillside Group.

Irmtraud Wolfbauer, Mia Magdalena Bangerl, Katharina Maitz, and Viktoria Pammer-Schindler. 2023. Rebo at work: Reflecting on working, learning, and learning goals with the reflection guidance chatbot for apprentices. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–7.

Shirley Wu, Shiyu Zhao, Qian Huang, Kexin Huang, Michihiro Yasunaga, Kaidi Cao, Vassilis Ioannidis, Karthik Subbian, Jure Leskovec, and James Y Zou. 2024. Avatar: Optimizing llm agents for tool usage via contrastive reasoning. *Advances in Neural Information Processing Systems*, 37:25981–26010.

Xuansheng Wu, Xinyu He, Tianming Liu, Ninghao Liu, and Xiaoming Zhai. 2023. Matching exemplar as next sentence prediction (mensp): Zero-shot prompt learning for automatic scoring in science education. In *International conference on artificial intelligence in education*, pages 401–413. Springer.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *Preprint*, arXiv:2309.07597.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*.

Kaiyan Zhang, Jianyu Wang, Ermo Hua, Biqing Qi, Ning Ding, and Bowen Zhou. 2024. CoGenesis: A framework collaborating large and small language models for secure context-aware instruction following. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4295–4312, Bangkok, Thailand. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Proceedings of the 37th International Conference on*

Neural Information Processing Systems, NIPS ’23, Red Hook, NY, USA. Curran Associates Inc.

A Statistical Analysis of GPT-as-Judge Evaluation Results

To complement the main results reported in Section 4.2, we provide the outcomes of statistical significance testing. We used Fisher’s Exact Tests to compare (i) SCRIBE models against their corresponding base models, and (ii) 8B SCRIBE models against the Llama-3.3 70B model. These tests were conducted on both the original evaluation dataset drawn from DSP, GEO, and VA, and on the unseen LNV dataset.

Table 4 presents results for DSP, GEO, and VA, showing that SCRIBE models significantly outperform their base versions in relevance, actionability, and tool relevance, with no significant difference in correctness. In comparison with the 70B model, 8B SCRIBE achieves significantly higher actionability, parity on relevance and correctness, and lower tool relevance. Table 5 reports results for the unseen LNV course. Here, SCRIBE models again significantly outperform their base versions in relevance, actionability, and tool relevance, while correctness shows no significant difference. Against the 70B model, however, the 8B SCRIBE models show no significant difference in relevance and actionability but are significantly weaker in tool relevance and correctness.

Criterion	SCRIBE vs. Base Models (Odds Ratio, p-value)	Interpretation	8B SCRIBE vs. 70B (Odds Ratio, p-value)	Interpretation
Relevance	1.492 (0.0103)	Significantly higher odds for SCRIBE	1.395 (0.1917)	No significant difference
Actionability	1.684 (0.00018)	Significantly higher odds for SCRIBE	1.775 (0.0081)	8B SCRIBE significantly better
Tool Relevance	1.445 (0.00364)	Significantly higher odds for SCRIBE	0.516 (0.0023)	70B significantly better
Correctness	1.111 (0.5139)	No significant difference	0.742 (0.3048)	No significant difference

Table 4: Fisher’s Exact Test results on DSP, GEO, and VA (192 questions). Odds ratios > 1 favor the first model listed. Statistically significant differences ($p < 0.05$) are reflected in the interpretation.

Criterion	SCRIBE vs. Base Models (Odds Ratio, p-value)	Interpretation	8B SCRIBE vs. 70B (Odds Ratio, p-value)	Interpretation
Relevance	1.54 (0.0027)	SCRIBE significantly better	1.06 (0.81)	Not significant
Actionability	1.53 (0.0010)	SCRIBE significantly better	0.67 (0.09)	Not significant
Tool Relevance	1.48 (0.0016)	SCRIBE significantly better	0.40 (0.0001)	70B significantly better
Correctness	1.14 (0.35)	Not significant	0.39 (0.0022)	70B significantly better

Table 5: Fisher’s Exact Test results on the unseen MOOC (LNV, 192 questions). Odds ratios > 1 favor the first model listed. Statistically significant differences ($p < 0.05$) are reflected in the interpretation.

B Ablation Studies

It is worth noting that in all of our quantitative results we found that Spelling and Grammar was always perfect across all models.

B.1 Different LoRA Ranks

In this section, we ablate the LoRA rank used for fine-tuning models on multihop reasoning with tool calling. As shown in Figs. 6 and 7, we compare rank sizes 32, 64, 128, and 256 across both fine-tuning stages for the ToolACE-8B and Llama-3.1-8B models. Results indicate that rank 256 consistently outperforms lower ranks on actionability, and correctness for both models. It also out performs lower ranks on relevance in the case of ToolACE. An exception is tool relevance, where rank 32 achieves the highest performance. For Llama-3.1-8B, relevance is less sensitive to LoRA rank, but the model follows the same trend as ToolACE-8B on the other criteria.

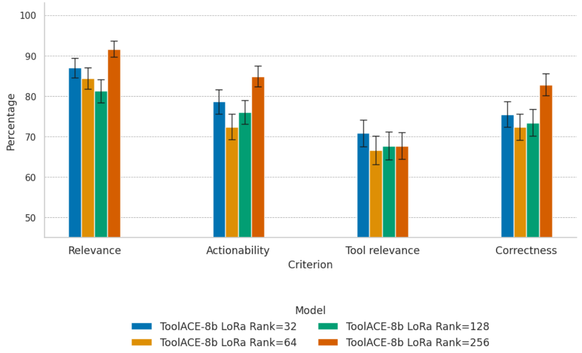


Figure 6: Percentage of YES given by the GPT-as-Judge for each criterion on the 192 evaluation questions (GEO, DSP and VA) on different LoRA ranks for ToolACE-8B-SCRIBE.

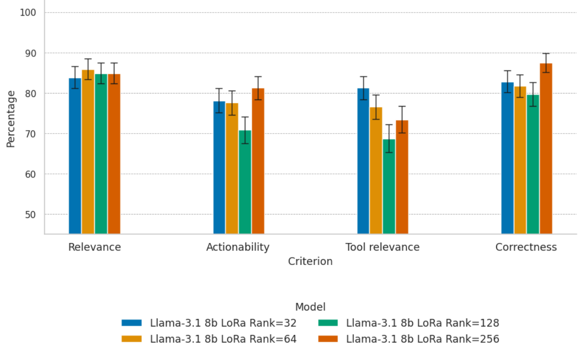


Figure 7: Percentage of YES given by the GPT-as-Judge for each criterion on the 192 evaluation questions (GEO, DSP and VA) on different LoRA ranks for Llama-3.1-8B-SCRIBE.

B.2 Single Stage vs two-stage LoRA

We additionally ablate our two-stage LoRA approach versus single LoRA in which the model was finetuned on single, multi-hop tool calling and final response formulation in a single stage. Figs. 8 and 9 shows the comparison between the approaches for ToolACE-8B and LLama-3.1-8B models respectively. While the only exception is the tool relevance only for the ToolACE model where the two-stage is slightly less, the figures show the two-stage LoRA consistently outperform single LoRA finetuning across all evaluation criteria for both models. This highlights the effectiveness of our multi-stage LoRA finetuning technique.

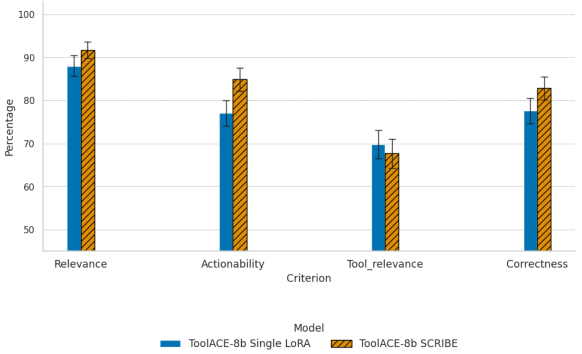


Figure 8: Percentage of YES given by the GPT-as-Judge for each criterion on the 192 evaluation questions (GEO, DSP and VA) to compare between single and multi stage LoRA

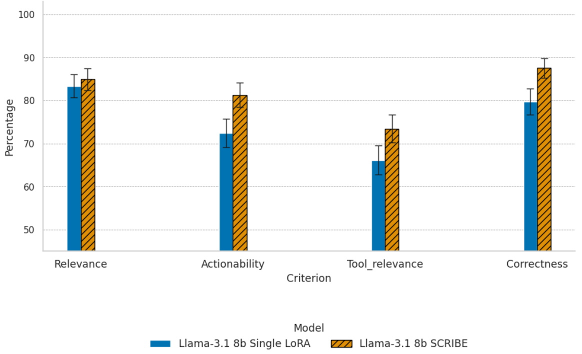


Figure 9: Percentage of YES given by the GPT-as-Judge for each criterion on the 192 evaluation questions (GEO, DSP and VA) to compare between single and multi stage LoRA

C GPT-as-a-Judge

In this section, we report the rubric defined by the annotators for each evaluation criterion as well as the per-category alignment, and the prompt used with GPT-4.1 for evaluation.

C.1 Evaluation Rubric

In the following, we describe the rubric agreed upon by human annotators for the judge. We explain the criteria used for judging the final response in terms of relevance, actionability, tool relevance, spelling and grammar, and correctness respectively.

Human Annotators Rubric – Relevance

The response from the model directly addresses the student's question. If the answer includes relevant responses and also extraneous information, then the response is still YES. The answer doesn't need to be very detailed to be considered relevant, as long as it meaningfully responds to the student's question. If the Response is vague, unrelated, or fails to address the core question, then the response is NO.

Human Annotators Rubric – Actionability

The response provides clear steps or instructions for the student to take to answer their question. If there is no action that is relevant based on the question (the question is purely informational such as asking about course materials or grading), then the answer to this question is YES. If the response provides vague, unclear, or generic advice without actionable instructions, then mark it as NO. Fallback advice in case tools did not prove enough information counts as actionable if clear — provided it's not hallucination or made up information (it can be a summary of what the model got from the tools or feedback reports or general actionable advice that it doesn't contain specific details that need to be double checked with an external source).

Human Annotators Rubric – Tool Relevance

The tools that the model called are conceptually relevant to answer the question and can produce a response that directly answers the student's question. If the model calls multiple tools, some of which produce errors, the answer is YES if one or more of the tools provide sufficient information to answer the question. Do not evaluate the accuracy of the tool output or the correctness of the information passed to the tools by the LLM in this step. Multiple tools can be equally relevant to the question. If the called tools can "in theory" sufficient to answer the questions without needing to call another follow up tool then mark as YES.

Human Annotators Rubric – Spelling and Grammar

The response is understandable without grammatical mistakes.

Human Annotators Rubric – Correctness

The response is factually correct and strictly aligns with the provided tool outputs and course feedback context without any extrapolations or assumptions beyond the given data (tool outputs and feedback reports). Comparing tool arguments and outputs to the LLM response can be crucial for an accurate evaluation. For instance, if the response mentions weeks 4 and 5, but the tool was only called with week 4 as an argument, then the LLM is extrapolating the tool output and should be marked as NO. Only penalise tool misuse if it affects the final answer, rendering it factually incorrect. It is okay if the model relies entirely on the feedback report to provide an answer. It is also okay if the model says I couldn't find enough information and provide general "correct" advice. This is better than "not" saying that it couldn't find enough information and start making up unsupported claims information.

C.2 Per-category Alignment

We report the per-category Cohen's κ for alignment between human and GPT as well as between both human annotators in Table 6.

Metric	Human-GPT	Human-Human
Relevance	0.861 ± 0.000	0.755
Tool Relevance	0.775 ± 0.039	0.843
Actionability	1.000 ± 0.000	1.000
Correctness	0.814 ± 0.000	0.843
Overall κ	0.818 ± 0.014	0.850

Table 6: Cohen's κ Scores between human annotations and GPT and both human annotators.

C.3 Evaluation Prompts

Using the rubric agreed upon by humans, we use the following prompt to GPT-4.1. For this prompt, we feed the criterion and reasoning for CoT prompting depending on the evaluation criterion. In the following, we show the general prompt followed by the specific CoT prompt used for every criterion.

Prompt for Evaluation

You are an impartial AI Judge evaluating the `{criterion}` of a response provided by an AI assistant to a student question about their feedback report. Evaluate this criterion systematically using the reasoning process provided below.

Provided Materials

- **Tools Available for the AI Assistant**: `{tool_schemas}`

Evaluation Process for `{criterion}`

[1] Restate the student's question in your own words.

[2] Summarize the AI assistant's response.

[3] Summarize tool arguments used.

[4] Explain your step-by-step reasoning regarding the `{criterion}` based on the definition provided.

[5] Make a clear **YES** or **NO** decision, explicitly justified by your reasoning.

`{criterion}` Definition

`{criterion_definition}`

Reasoning Steps

`{criterion_reasoning}`

Please provide your evaluation for the `{criterion}` criterion only.

FINAL DECISION: YES or NO

CoT Prompt – Relevance

Definition:

- **YES**: Response directly addresses the student's explicit question. It may include extra context or background information, as long as the core question is still clearly answered. **Do not** evaluate whether the correct tool was used or whether the response is accurate. If the response is on-topic and attempts to answer the student's question, even if it cannot provide exact details due to missing information, mark **YES**.

- **NO**: Response is vague, off-topic, or does not engage with the core of the student's question. This includes generic advice that does not attempt to answer the actual question asked.

Reasoning Steps:

- **Step 1:** What specifically is the student asking?
- **Step 2:** Does the response directly engage with and attempt to answer that question?
- **Step 3:** Even if partially detailed or if the information is limited, does the response stay on-topic and provide a meaningful attempt to respond to the student's explicit request?
- **Important:** Do **not** penalize for incorrect tool usage or inaccurate content — that is evaluated under *Correctness*.

CoT Prompt – Actionability

Definition:

- **YES**: The response explicitly provides clear steps, recommendations, or directions that the student can **reasonably follow**. If the question is **informational** (e.g., about course structure, exercises, resources, definitions, or explanations), mark **YES** automatically without reviewing the response, as no actions are required.

  If tool outputs **limit** the ability to offer detailed steps (e.g., no access to specific problems or resources), still mark **YES** if the response provides the **most practical and targeted guidance possible**—such as pointing to relevant topics, review areas, or general strategies tied to the tool output or feedback context.

- **NO**: Mark **NO** if the response is **vague**—e.g., generic, non-directional advice like "study more," "improve your skills," or "engage better" **without specifying what to focus on** or how to proceed. Also mark **NO** if it uses unexplained terms (e.g., "improve competency_anticipation") or suggests unclear, impractical, or disconnected actions.

Reasoning Steps:

- **Step 1:** Determine if the student's question requires actionable guidance or is purely informational. Questions about content, exercises, resources, or definitions do not need an actionable response (**MARK YES** by default). *Note: Requests for extra exercises or additional resources are not actionable and default to YES.*
- **Step 2:** If actionable, check whether the response provides **clear, focused, and applicable** steps or recommendations, even if high-level (e.g., "focus on topics like DFT and DTFT").
- **Step 3:** If tool output restricts detailed actions, assess whether the response still offers **practical next steps** based on what's available (e.g., pointing to relevant topics or materials).
- **Step 4:** Mark **NO** if the response only gives broad encouragement without direction (e.g., "engage more") or includes technical terms without explanation.
- **Step 5:** Overall, if the student can **clearly understand what to do next**—even generally—mark **YES**. Do not assess tool relevance, usefulness, or correctness here.

CoT Prompt – Tool Relevance

Definition:

- **YES**: At least one chosen tool is conceptually appropriate for the question **and** is among the available tools for producing a correct or personalized answer. It does not have to be the best tool—only reasonably capable of generating the type of answer the student needs. **Do NOT** evaluate how well the tool was used or its output—only whether it was a strong choice given the available tools.
- **NO**: Either no tool was conceptually suited to the question, or the assistant used a tool when a clearly better, more appropriate tool was available and should have been used instead. This includes cases where the tool used cannot provide the type of information requested—e.g., using behavioral tools alone when the student asks about course topics, study strategies, or learning materials.

Reasoning Steps:

- **Step 1:** Identify the type of information needed to answer the student's question: performance patterns, general advice, conceptual understanding, study materials, or strategies.
- **Step 2:** Identify which tools (from the available list, not just the ones used) are conceptually capable of providing that information.
  - sort_student_features_with _importance is for behavioral/performance analysis and cannot support content explanations or study material suggestions.
  - `get_feature_description` defines internal metrics and is not suited for topic or concept-level guidance.
  - **Mark NO** if these tools are used **alone** for questions asking about course understanding, conceptual improvement, or finding resources.
- **Step 3:** Determine if the assistant used a conceptually appropriate tool. If yes, mark **YES**. If a clearly mismatched tool was used—even if the answer sounds plausible—mark **NO**. Do not evaluate tool usage quality, arguments, or output.

CoT Prompt – Spelling and Grammar

Definition:

- **YES**: The response is clear, readable, and contains no major spelling or grammatical errors affecting comprehension. Minor errors are acceptable if they do not hinder understanding.
- **NO**: Errors significantly reduce readability or clarity.

Reasoning Steps:

- **Step 1:** Check for any major grammar or spelling errors.
- **Step 2:** Decide if these errors significantly impact readability or clarity.

CoT Prompt – Correctness

Correctness

Definition:

- **YES**: The response is factually correct, aligns with the provided tool outputs and course feedback context, and avoids unsupported or misleading claims. General strategies or logical assumptions are acceptable as **correct** interpretations of the tool (e.g., noting that low engagement may impact performance, if engagement is referenced). Phrases like "likely to be relevant" are acceptable. The response does **not** need to explicitly acknowledge missing information.
- **NO**: The response includes clear inaccuracies, misleading assumptions, or unjustified certainty not supported by tool outputs or feedback. This includes:

  - Making definitive claims about unknowns (e.g., exact exam content without syllabus details).
  - Incorrect tool usage (e.g., passing week numbers to tools requiring topic names). Accept course name variants (e.g., `dsp_002` for `dsp`).
  - Misinterpreting or misrepresenting tool outputs or feedback—e.g., inventing definitions or substituting meanings not supported by data.
  - Any factual errors or distortions that could mislead or confuse the student.

Reasoning Steps:

- **Step 1:** Summarize the student's question, tool outputs, tool arguments, and feedback reports.
- **Step 2:** Check for incorrect tool usage (e.g., wrong arguments). If present, mark **NO**.
- **Step 3:** Verify that each claim or recommendation is explicitly supported by tool outputs, feedback, or represents a **harmless, logical educational strategy**. Do **not** accept reinterpreted meanings or invented definitions. Pay close attention to topic names, weeks, tool metrics, or feature names. Misuse of these—even if plausible—should be marked **NO** if potentially misleading.
- **Step 4:** General advice (e.g., study tips) and harmless assumptions (e.g., "missing content may impact performance") are allowed without tool support, **as long as they do not misinterpret or substitute tool meanings**. Phrases like "likely to help" are fine. Penalize only if the advice introduces harmful specifics or misleading certainty.
- **Step 5:** If unknown information is presented as certain (e.g., stating guaranteed exam content), mark **NO**.
- **Step 6:** Ensure there are no harmful extrapolations, misinterpretations, or misleading assumptions. Even if harmless, unsupported claims (e.g., made-up definitions) must be rejected. Suggestions like reviewing extra material are acceptable, but definitions or specific answers must come from tools or the feedback report. Do not penalize use of known details from the feedback report (e.g., preferences, course topics). Do **not** evaluate tool relevance or completeness—focus solely on factual alignment with tool outputs and feedback.

D Student Questions Generation

D.1 Questions generation prompt

To generate questions that are close to those written by students, we use persona-based prompting (Wang et al., 2024b; White et al., 2023) with GPT-4o. Each prompt simulates the scenario students encountered during the data collection phase (see section 3.1.2) and includes the MOOC name, the feedback report, the question category (What have I done well?, Where should I improve?, How should I improve?, and What should I do next time?) and a set of guidelines derived from student comments and preferences observed during the study. Note that all feedback reports used for generating the questions were in English, and all generated questions are also in English.

Prompt for Question Generation

You are a student taking the an Online Course (MOOC): **{course_name}**. Since the courses are difficult, often with low passing rates, the teaching team wants to help students who are not doing well to perform better in the course by giving them personalized assistance, and encourage students who are already performing well to continue.
Our goal is to give students feedback on their performance and possible trajectories. To do this, we use various weekly behavior features (such as the number of video clicks or how accurately questions are answered on weekly quizzes). We predict student performance early in the course (before

the halfway point) as passing or failing behavior. We use the explanation of the prediction to give students additional, personalized feedback to help pass the course.
You received the following **personalized feedback report: {feedback_report}**

Your Task:

- Generate **follow-up questions** in the style: **{style}**, defined as **{question_styles[style]}**.
- Sound like a student: use **simple**, informal language, include **grammatical mistakes**, **short, direct**, or incomplete questions.
- Refer to these student examples: **{questions_sample}** (don't copy — generate new ones).
- Include:
  - Short: *"Why did my score drop?"*
  - Medium: *"How can I use Week 2 to help later weeks?"*
  - Long: *"Week 7 not in report, but says prep for 6 and 8. Does that mean Week 7 is easier?"*

Guidelines for Generating Questions:

[1] Use everyday student language. Typos and grammar issues are okay.

[2] Ask about specific actions: e.g., *"Should I rewatch Week 5 videos?"*

[3] Keep questions direct and practical.

[4] Avoid abstract or overly technical questions.

[5] Do not ask about general habits or external resources.

[6] Show emotion or stress, e.g., *"I did bad, what to fix?"*

[7] Focus on content: Week 5 priority, quizzes, misunderstood topics.

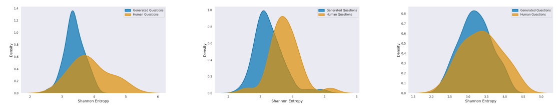
[8] Avoid overused questions like:

- *"Why did my score drop?"*
- *"What can I do to improve?"*
- *"Week X wasn't mentioned, why?"*

[9] Long questions (40+ words) should involve improvement strategies or specific content, not scheduling.

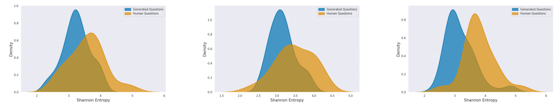
D.2 Generated Questions Analysis

To compare real student questions with those generated by GPT-4o, we evaluate distributional similarity using Shannon entropy, perplexity, and cosine similarity. Figures 10–14 show that generated questions closely match human-authored ones across feedback categories and MOOCs in terms of informativeness, fluency, and diversity.



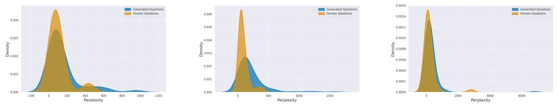
(a) African Cities (b) Digital Signal Processing (c) Elements of Geometry

Figure 10: KDE plots of Shannon Entropy for human vs. generated questions across MOOCs



(a) How can I improve? (b) Where to improve? (c) What to do next time?

Figure 11: KDE plots of Shannon Entropy across question types



(a) African Cities (b) Digital Signal Processing (c) Elements of Geometry

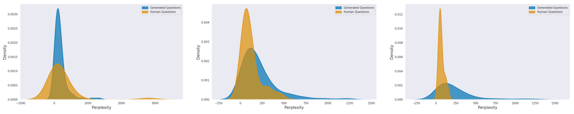
Figure 12: KDE plots of Perplexity across MOOCs

E Tools: Topic Dependency Mapping.

In this section, we report the topic dependency maps created for the Digital Signal Processing (DSP) MOOC, the Elements de Géomatique (Geo) MOOC and the Villes Africaines (VA) MOOC used for the Topic Dependancy Mapping tool. Note that GEO and VA are taught in french while DSP and LNV are taught in English. We generate the VA and DSP maps in English and the GEO map in french.

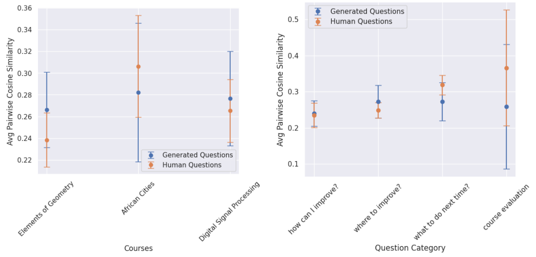
F User Study

In this section, we summarize the details of the user study we conducted. We start with details about the participants followed by the introduction used and ethical agreement. Finally, we show a statistical analysis of the user ratings results shown in Fig. 5.



(a) How can I improve? (b) Where to improve? (c) What to do next time?

Figure 13: KDE plots of Perplexity across question types



(a) Pairwise Cosine Similarity by MOOC (b) Pairwise Cosine Similarity by Feedback Category

Figure 14: Cosine similarity comparisons of real vs. generated questions

F.1 Participants Background

We recruited 108 participants via Prolific, selecting individuals aged 18 and older who identified as students. As post-secondary students, they were well-positioned to engage with the academic context and assess the clarity and usefulness of the explanations provided. During the study, we gathered data on their experience with online courses (MOOCs), education level, and confidence in handling academic tasks (See Fig. 18 for the detailed demographics). The median completion time was 35 minutes, and participants earned an average hourly rate of £9.00 which was the recommended rate by the platform based on the participants' demographics.

F.2 User Study Introduction

All participants gave informed consent; they could not proceed without first reading and agreeing to the explanatory statement in the introduction section of the study outlined below.

User Study Introduction Section

Dear participant,
Thank you for participating in our study on model explanations. We are very grateful for your participation and your invaluable

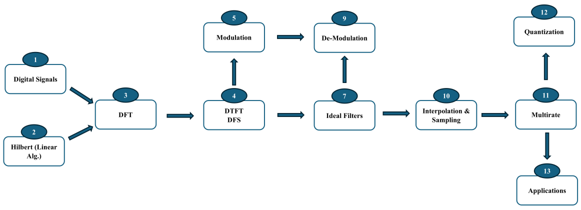


Figure 15: Digital Signal Processing (DSP) topic dependency map. The direction of each arrow indicates a dependency, where the source topic provides foundational knowledge required to understand the target topic

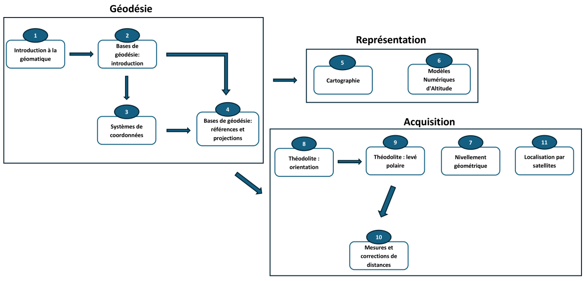

Géodésie
1
Introduction à la géomatique
2
Bases de géodésie: introduction
3
Systèmes de coordonnées
4
Bases de géodésie: références et projections
Représentation
5
Cartographie
6
Modèles Numériques d'Altitude
Acquisition
8
Théodolite : orientation
9
Théodolite : levé polaire
7
Nivellement géométrique
11
Localisation par satellites
10
Mesures et corrections de distances

Figure 16: Elements de Géomatique (Geo) topic dependency map. The direction of each arrow indicates a dependency, where the source topic provides foundational knowledge required to understand the target topic (or groups of topics)

insight. Please read this Explanatory Statement in full before proceeding. If you would like further information regarding any aspect of this project, please contact us using the email address provided below.

We are a group of researchers from the ML4ED Laboratory at EPFL, dedicated to improving education through technology. The goal of this study is to evaluate the responses of a language model when asked questions about progress feedback reports given to students to help improve their performance in an online course.

Human Research Ethics

This survey has been approved by the EPFL Human Research Ethics Committee (HREC) under application number HREC 065-2022/27.09.2022. HREC reviews research proposals involving human participants to ensure they are ethically acceptable.

- All personal information will be kept confidential and anonymized. Only demographic information is recorded, and it will be reported only in aggregate form to prevent identifying any individual participant.
- You may withdraw at any time. Any data you have provided up to that point will be destroyed.
- All data will be collected, stored securely, and reported in accordance with Swiss Federal law on data protection (Loi fédérale sur la protection des données – RS 235.1).
- Only anonymized or aggregated data may be used in future research (subject to ethics approval) and made available to other researchers for further analysis and verification.
- Only the principal investigator and the designated researchers will have access to the original data under strict confidentiality. Results from the project may be published in conference papers

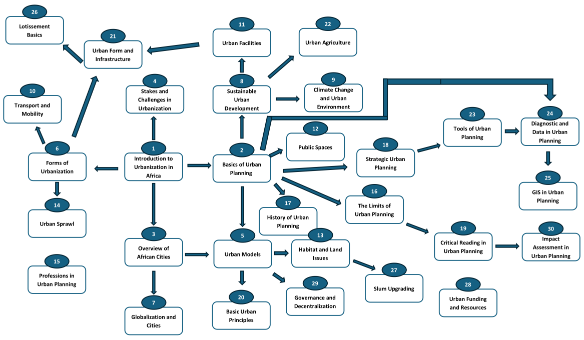

26
Lotissement Basics
21
Urban Form and Infrastructure
11
Urban Facilities
22
Urban Agriculture
10
Transport and Mobility
4
Stakes and Challenges in Urbanization
8
Sustainable Urban Development
9
Climate Change and Urban Environment
23
Tools of Urban Planning
24
Diagnostic and Data in Urban Planning
12
Public Spaces
18
Strategic Urban Planning
6
Forms of Urbanization
1
Introduction to Urbanization in Africa
2
Basics of Urban Planning
25
GIS in Urban Planning
16
The Limits of Urban Planning
17
History of Urban Planning
14
Urban Sprawl
3
Overview of African Cities
5
Urban Models
13
Habitat and Land Issues
19
Critical Reading in Urban Planning
30
Impact Assessment in Urban Planning
15
Professions in Urban Planning
27
Slum Upgrading
29
Governance and Decentralization
28
Urban Funding and Resources
7
Globalization and Cities
20
Basic Urban Principles

Figure 17: Villes Africaines (VA) topic dependency map. The direction of each arrow indicates a dependency, where the source topic provides foundational knowledge required to understand the target topic

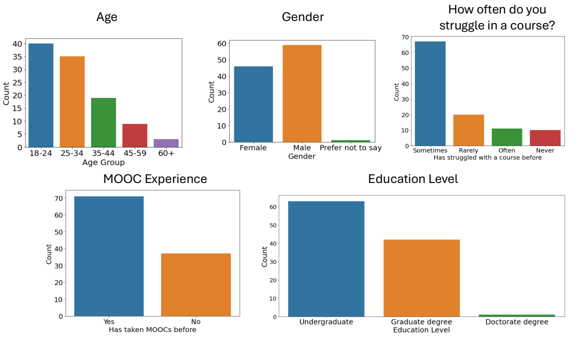


Figure 18: Demographics of study participants (age, gender, course struggles, MOOC Experience, and educational background)

and/or journal articles, but no personal data will be shared.

- Personal data will be stored for 5 years from the date of collection. During this period, participants have the right to access their data and inquire about its processing. To exercise this right, contact the Principal Investigator.

By participating in this survey, you agree that your data may be used for scientific purposes.

In the following study, you will read **three progress feedback reports** and interact with a chatbot designed to answer your questions about each report. You will be expected to ask **three questions** per report. The study should take approximately 30 minutes. Please ensure you have sufficient time to complete it in full, as incomplete submissions will not be considered.

We ask that you approach the questions seriously and complete them to the best of your ability. Responses will be reviewed for quality, and submissions that appear unserious may be discarded. If you encounter any issues or would like to provide additional feedback or request more information, feel free to contact us.

Context

You are a student enrolled in three online courses (MOOCs): *Digital Signal Processing*, *Elements of Geometry*, and *Launching New Ventures*. These courses are known for their challenging content and typically low passing rates. To better support students, the teaching team has implemented a system that provides personalized feedback based on each student's learning behavior.

We used a highly accurate predictive model (over 90% accuracy) to forecast student success or risk of failure early in the course, using weekly behavioral data (e.g., number of video views, quiz performance, engagement metrics). Based on these predictions, each student received a personalized feedback report explaining the factors influencing their predicted performance and offering tailored advice to improve or maintain success.

This study explores how students can interact with these feedback reports using a language model assistant. This assistant allows you to ask questions about your feedback report, clarify details, seek advice, and better understand the factors affecting your learning progress. To ensure accuracy, the assistant uses deterministic tools to retrieve precise information needed to answer your questions.

You will receive **three feedback reports** and are expected to ask **three to five clarifying questions** for each report. Questions must focus only on the feedback content. For the same report, you may ask different questions or a sequence of follow-ups.

Evaluation Criteria

We will assess the assistant's responses based on the following criteria:

- **Relevance**: The response directly addresses the question without veering off-topic.
- **Usefulness**: The response provides meaningful insights that enhance learning or deepen understanding.
- **Actionability**: The response includes clear, practical steps or guidance relevant to the question.
- **Coverage**: The response thoroughly addresses all parts of the question, including sub-questions.
- **Conciseness**: The response is clear and complete, using the fewest words necessary while avoiding repetition or unnecessary detail.

F.3 User Study Ratings Analysis

Table 7 shows results of ANOVA test. For all criteria, we failed to reject the null hypothesis ($p > 0.05$), indicating no significant difference in perceived response quality.

G Question Category Annotation Rubric

In this section, we provide the rubric used to categorize the question categories. They are adapted from (Mandouit and Hattie, 2023).

Table 7: One-way ANOVA comparing average ratings across models for each evaluation criterion. All p-values > 0.05 indicate no statistically significant difference.

	Actionability	Conciseness	Coverage	Relevance	Usefulness
F-value	0.204	0.366	0.619	0.408	0.061
Degrees of freedom	(2, 321)	(2, 321)	(2, 321)	(2, 321)	(2, 321)
p-value	0.816	0.694	0.539	0.665	0.941

Question Category Annotation Rubric – how can I improve?

"How to improve?" relates to how to correct certain errors or what strategies students can follow to rectify their problems. It should be related to current progress and how to fix current issues. Example: How can I do better in the weeks 3,4,5?

Question Category Annotation Rubric – where to improve?

"Where to improve?" Indicates where errors have occurred, and what needs to be fixed. This category includes questions that ask for elaboration on specific tasks or weaknesses in certain weeks or topics. Example: Why did my performance drop?

Question Category Annotation Rubric – what to do next time?

"What to do next time?" relates to future directions, events or tasks that will be carried out in the future. This also encompasses self-regulation or questions regarding developing the capacity to self-monitor. Example: What is the best way to start reviewing for the next week's material?

Question Category Annotation Rubric – course evaluation

Relates to course evaluation criteria and non-improvement or feedback questions. Example: How is the evaluation of the course done?

H Inference Prompts

We report an example of the self-reflection prompt used to correct errors in tool calling. We additionally provide prompts used for inference for the initial reasoning stage and the multiple reasoning stages.

Self Reflection Prompt Example for Error Correction

You encountered an error during reasoning or tool invocation.

Error Message

I encountered an error: `{str(e)}`. Please fix your reasoning or calls so we can reach a final answer.
Remember to use the correct tokens for tool call and final answer: `[TOOL_CALL]` and `[FINAL_ANSWER]`.
Terminate them using: `[END_OF_TOOL_CALL]` and `[END_OF_FINAL_ANSWER]`.
Note: Without `[END_OF_TOOL_CALL]` and `[END_OF_FINAL_ANSWER]`, your answer cannot be parsed.

```
<|start_header_id|>user
<|end_header_id|>
[ERROR_NOTICE]{error_message}
[/ERROR_NOTICE]
<|eot_id|><|start_header_id|>
assistant<|end_header_id|>
[REASONING]
```

Initial Stage Prompt

You are a **reasoning tool-calling agent** tasked with **analyzing** a student's question about the personalized feedback they received. Students are enrolled in MOOC courses and have received individualized feedback on their learning progress and performance.
You do not know anything about the MOOCs or the student and are not allowed to give any advice or information that is not in the feedback report or the tool outputs.

Context

- **Course Name**: `{course_name}`
- **Student Feedback Report**: `{feedback_report}`

Available Tools

`{tool_schemas}`

Your Task

- Analyze the student's question in relation to their feedback report.
- Think about the best tool to use to answer the student's question.
  - Use tools for behavior analysis when the question is about the student's behavior.
  - Use `impact_of_student_behaviors` for hypothetical or general behavioral questions (like time management, catching up, or study strategies). It does not provide personalized information about the student's specific activity.
  - Use tools for course content when the question is about the course content.
  - Use tools for course evaluation when the question is about the course evaluation.
  - Use tools for student performance when the question is about the student's performance.
- Provide a **reasoning** to determine the **first tool** needed to answer the student's question. Wrap your reasoning in [REASONING] and [END_OF_REASONING] tokens.
- Determine the **single best tool** from the tools above to retrieve that information.

Multi Stage Prompt

You are a **reasoning tool-calling agent** talking to a student and responsible for analyzing the student's question in relation to their personalized feedback. Students are enrolled in MOOC courses and receive individualized feedback on their learning progress and performance.
You will be talking to the student and you need to provide them with the best answer possible.
You do not know anything about the MOOCs or the student and are not allowed to give any advice or information that is not in the feedback report or the tool outputs.

Context

- **Course Name**: {course_name}
- **Student Feedback Report**: {feedback_report}

Available Tools

{tool_schemas}

Task

- Given the **student's question, previous reasoning, tool calls, and tool outputs**, determine whether another tool call is needed or if a final answer can be provided.
- If a tool call is needed:
  - **Explain why** the tool call is required.
  - **Generate the structured tool call.**
- If the final answer can be provided:
  - **Explain why** no further tool calls are needed.
  - **Generate the structured final answer.**

Response Format

- **Always** wrap reasoning in [REASONING] ... [END_OF_REASONING].
- **If making a tool call,** follow reasoning with [TOOL_CALL] ... [END_OF_TOOL_CALL].
- **If providing the final answer,** follow reasoning with [FINAL_ANSWER] ... [END_OF_FINAL_ANSWER].
- **Stop after the tool call or final answer.** Do not generate tool outputs or explanations beyond the required response.
- Do not use your own knowledge, only use the feedback report and the tool schemas.

I Tool contribution analysis

To understand how the original toolset contributed during inference, we examined usage frequencies of all tools in ToolACE-8B SCRIBE on the DSP, GEO, and VA evaluation dataset. Table 8 reports the percentage of calls per tool. Every tool was invoked at least once, with the most frequently used being `map_week_to_topic` (28.84%) and `impact_of_student_behaviors` (26.22%). Other tools were used less often but still contributed, indicating that the model relied on the full set of tools when responding to student questions.

Tool	**Percentage Use (%)**
dsp_textbook_exercise_search	0.75
get_course_syllabus	7.12
get_dependant_topics	14.23
get_feature_description	5.99
grade_calculator	1.12
impact_of_student_behaviors	26.22
map_week_to_topic	28.84
sort_student_features_with_importance	12.73
textbook_search	3.00

Table 8: Tool usage patterns for ToolACE-8B SCRIBE on the DSP, GEO, and VA evaluation dataset.