PMTutor: A Personalized Mathematics Tutor based on Multi-Agent Collaboration

Anonymous ACL submission

Abstract

Recent advances in LLMs have opened possibilities for dialogue-based intelligent tutoring. However, most existing ITS still adopt fixed, one-size-fits-all strategies, limiting personalization. To address this, we propose a multi-agent framework with a realistic simulated student agent, a pedagogically grounded teacher agent, and a supervisory agent. The student agent models Big Five personality traits, four learning factors, personality-adaptive expressions, and probabilistic errors grounded in educational research. The teacher leverages a curated Strategy Base of 28 theory-driven strategies, while the supervisor ensures student realism and pedagogical quality through bidirectional oversight. Using this framework, we construct the PMTDial dataset and build PMTutor through a two-stage training pipeline that combines SFT for strategy-aware learning with DPO for preference optimization based on students' learning factors. Experiments show that PMTutor outperforms strong baselines in both automatic and human evaluations.

1 Introduction

Recent advances in large language models (LLMs) have enabled their widespread use in education, particularly in mathematics, which plays a key role in developing logical thinking and problem-solving skills. Meanwhile, LLMs have demonstrated strong mathematical capabilities. General models like GPT-4 (Achiam et al., 2023) and GPT-4o, along with specialized ones such as DeepSeek-Math (Shao et al., 2024) and Qwen-Math (Yang et al., 2024), achieve impressive results on benchmarks like GSM8K and AIME. These capabilities provide a solid foundation for LLM-powered math education.

However, most existing LLM-based Intelligent Tutoring Systems (ITS) still rely on fixed, one-size-fits-all strategies, producing generic responses that fail to adapt to individual student needs (Dan et al., 2023; Ding et al., 2024). Although multi-agent frameworks have been explored for personalized tutoring, current approaches use coarse student modeling and lack deep grounding in educational theories. To address these limitations, we propose a multi-agent framework consisting of student, teacher, and supervisor agents. The contrast between general and personalized mathematics tutoring is illustrated in Figure 1.

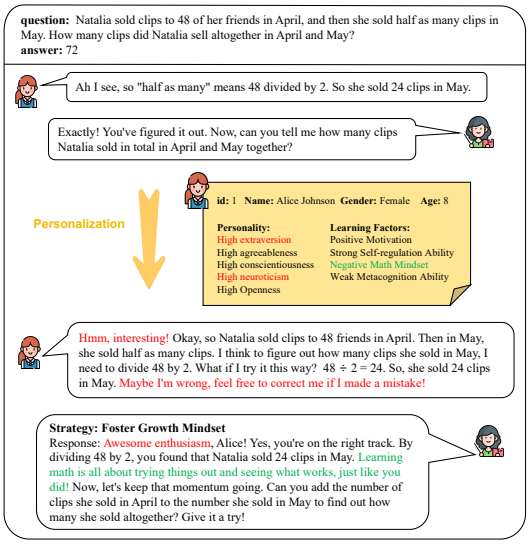

Figure 1: Comparison of General Teaching Tutoring Dialogue and Personalized Teaching Tutoring Dialogue.

The student agent is modeled with Big Five personality traits, four key learning factors (motivation, self-regulation, mathematical mindset, and metacognition), an *Expression Base* for personality-adaptive language, and a probabilistic *Error Base* grounded in educational research. The teacher draws from a theory-backed *Strategy Base* of 28 instructional strategies. A supervisor agent provides bidirectional oversight to ensure student realism and high pedagogical quality of teacher responses. Using this framework, we construct the **PMTDial** dataset and fine-tune Qwen2.5-7B-Instruct (Team, 2024) through a two-stage process

(SFT + DPO) to build **PMTutor**. We also introduce five teaching-oriented metrics: Clarity, Guidance, Empathy, Learning Support, and Thinking Cultivation. Experiments show that PMTutor significantly outperforms strong baselines in both automatic and human evaluations.

Our main contributions are as follows:

- We propose a multi-agent framework that generates the high-quality PMTDial dataset and builds PMTutor, which delivers personalized instructional strategies tailored to students' personalities and learning factors.
- We introduce five fine-grained, teaching-oriented evaluation metrics for assessing mathematics instructional dialogue quality.
- Both automatic and human evaluations show that PMTutor significantly outperforms strong baselines, providing more personalized and effective intelligent tutoring.

2 Related work

2.1 LLM-driven Agent Simulation

LLM-driven agents have shown strong capabilities in reasoning, planning, and simulating human-like behaviors across domains. They can perceive environments, make decisions, and exhibit personalities and social dynamics. In social science, traditional methods like surveys are costly and ethically challenging, while LLM-based multi-agent simulations offer a scalable and controllable alternative. Landmark works such as Generative Agents (Park et al., 2023) demonstrated realistic daily behaviors, and the paradigm has since expanded to medical collaboration (Li et al., 2024; Fan et al., 2024), game exploration (Wang et al., 2023), software development (Qian et al., 2024), and legal negotiation (Yue et al., 2025). In education, multi-agent systems have been used to simulate classroom interactions. Examples include SimClass (Zhang et al., 2024), CGMI (Shi et al., 2023), and MATHVC (Yue et al., 2024), which highlight the potential for immersive educational simulations.

2.2 Intelligent Tutoring Systems

Intelligent Tutoring Systems (ITS) aim to provide adaptive, personalized guidance similar to human teachers, and have been shown to improve learning outcomes (Kulik and Fletcher, 2016). However, traditional ITS rely heavily on manual engineering, resulting in rigid and generic responses (MacLellan and Koedinger, 2022). Recent LLM-based approaches have enabled more natural conversational tutoring (Chen et al., 2024; Liu et al., 2024b; Park et al., 2024) and introduced some personalization mechanisms (Wang et al., 2025). Despite these advances, most systems still use coarse student modeling (e.g., broad personality types or proficiency levels) and limited teaching strategies, lacking fine-grained personalization and robust evaluation grounded in real student feedback. These limitations in student modeling, pedagogical adaptability, and evaluation form the key gaps that our work addresses.

3 Methods

We employ a multi-agent collaborative framework with three core roles, the Student, Teacher, and Supervisor, to generate high-quality teacher-student tutoring dialogue data named PMTDial. We then use a two-stage training process, comprising SFT and DPO, to produce a personalized math tutoring model, PMTutor. The agent role definition and multi-agent framework are illustrated in Figure 2.

3.1 Agent Role Definition

We employ a three-role multi-agent framework (Student, Teacher, and Supervisor) for generating personalized tutoring dialogues. More details on the agent roles setting are provided in Appendix A.

3.1.1 Student Agent

Following prior work (Park et al., 2024; Van Bragt et al., 2011), we model each student with basic demographics, Big Five personality traits (McCrae and John, 1992) (Extraversion, Agreeableness, Conscientiousness, Neuroticism, Openness), and four key learning factors: motivation, self-regulation ability, mathematical mindset, and metacognition ability.

To produce realistic responses, we construct a personality-driven *Expression Base* inspired by the BF-TC model (Liu et al., 2024), which maps Big Five traits to typical verbal behaviors in mathematical problem-solving contexts. We also build an *Error Base* grounded in the IRIS Center case studies (IRIS Center, 2016), covering factual, procedural, and conceptual errors. Unlike random error injection in previous studies, we adopt a rule-based probabilistic mechanism. The error probability starts at 0.5 and is dynamically adjusted accord-

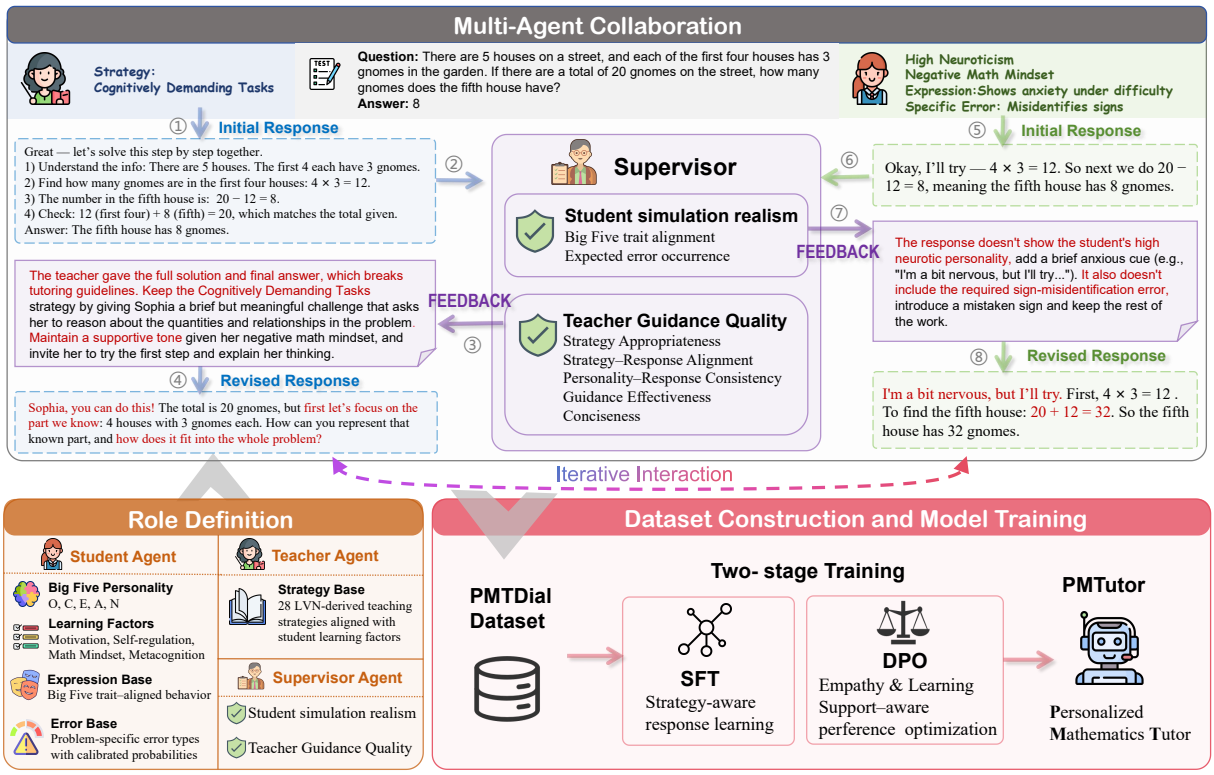

Figure 2: Overview of PMTutor based on multi-agent collaboration.

ing to the student's personality traits and learning factors.

To increase profile diversity, we prompt five different LLMs (ChatGLM3-6B, Qwen-Max, Llama-3.3-70B-Instruct, DeepSeek-V3, and GPT-3.5) to act as students and answer standardized questionnaires on learning factors and personality traits, generating 40 distinct student profiles.

3.1.2 Teacher Agent

The teacher draws from established educational theories and selects strategies from a curated *Strategy Base* containing 28 actionable instructions targeting the four learning factors. These strategies were manually selected by experts based on the Learner Variability Navigator project (Digital Promise, 2022). Strategy recommendation prioritizes the student's weak learning factors.

3.1.3 Supervisor Agent

To ensure the realism of the simulated student and the pedagogical quality of the teacher responses, we introduce a supervisor agent that performs bidirectional oversight on both agents. For the teacher, it evaluates each response on five dimensions: strategy appropriateness, strategy-response consistency, student personality consistency, guidance quality (avoiding direct answers), and conciseness. For the student, it checks consistency with designated Big Five traits and appropriate error occurrences across the dialogue. If needed, the supervisor provides targeted feedback for revision.

3.2 Multi-Agent Collaboration

The three agents collaborate to generate personalized tutoring dialogues. The overall process is summarized in Algorithm 1.

Before each dialogue turn, the system first computes a student-specific error rate and stochastically injects an error sampled from the *Error Base* into the prompt when triggered. Meanwhile, a personality-consistent behavior is randomly selected from the *Expression Base*. For the teacher, a suitable instructional strategy is recommended from the *Strategy Base* according to the student's weak learning factors.

To avoid overly long or meaningless dialogues, we limit the maximum number of turns to 10. In the first turn, the teacher decomposes the problem into smaller steps and applies a customized strategy to guide the student. The student then responds, after which the supervisor checks whether the response reflects the designated personality traits and error patterns. If revision is needed, the student revises based on supervisor feedback. In subse-

Algorithm 1 Multi-agent collaboration algorithm

Require: Student St, Teacher T, Supervisor Sp, Maxturn m, Dialogue history H, Error Base ERB, Expression Base EXB, Strategy Base SB
Input: Student profile p, Question q, Answer a
Output: H

Compute student-specific error rate ϵ based on profile p
$er \leftarrow RandomSelect(ERB, \epsilon)$
$ex \leftarrow RandomSelect(EXB)$
$p \leftarrow p \cup \{er, ex\}$
Initialize student $St \leftarrow (p, q)$
Extract recommended strategy s based on p
Initialize teacher $T \leftarrow (p, q, a, s)$
for $i = 1$ **to** m **do**
  T generate r_i^T
  Sp generate $r_i^{Sp(T)}$ based on r_i^T
  if "No Revision" not in $r_i^{Sp(T)}$ **then**
    T regenerate r_i^T
  end if
  $H \leftarrow H \cup \{r_i^T\}$
  if <end> in r_i^T **then**
    break
  end if
  St generate r_i^{St}
  Sp generate $r_i^{Sp(S_t)}$ based on r_i^{St}
  if "No Revision" not in $r_i^{Sp(S_t)}$ **then**
    St regenerate r_i^{St}
  end if
  $H \leftarrow H \cup \{r_i^{St}\}$
end for
return H

quent turns, the teacher generates the next response conditioned on the dialogue history, student profile, and recommended strategy. Each teacher response explicitly states the selected strategy and the instructional message. The supervisor then evaluates the teacher's response across five dimensions and provides targeted feedback for revision if necessary. The process continues until the teacher outputs the <end> token, at which point the complete dialogue history is returned.

3.3 Data Construction and Analysis

Using the proposed multi-agent framework, we had 40 simulated students each solve 50 math problems from GSM8K (Cobbe et al., 2021), initially generating 2,000 dialogues. After manual quality control, we removed dialogues that revealed full solutions too early, contained unnecessary extensions (e.g., unrelated follow-ups or excessive farewells), or failed to correct student errors. This left 1,964 high-quality dialogues, forming the final PMTDial dataset.

We compare PMTDial with existing math tutoring datasets (see Table 1). SocraTeach (Liu et al., 2024a) uses a large-scale "Dean–Teacher–Student" pipeline but lacks fine-grained student modeling and relies solely on the Socratic method. SocraticMATH (Ding et al., 2024) and PACEdata (Liu et al., 2025b) employ GPT-4-based teacher-student interactions with personality traits, yet their strategies are either limited or randomly generated by LLMs. MATHDIAL (Macina et al., 2023) adopts a semi-manual approach with human pedagogical expertise. Compared to prior work, PMTDial introduces key improvements: holistic student modeling across Big Five personality traits, learning factors, and cognitive profiles; a rich, theory-grounded Strategy Base with 28 instructional strategies; and a supervisor agent that enhances both student realism and teacher response quality.

3.4 Two-stage Training

We train the tutor model in two stages to improve both instructional competence and personalization. First, we perform supervised fine-tuning (SFT) to teach the model to generate strategy-aware tutoring responses. Second, we apply Direct Preference Optimization (DPO) (Rafailov et al., 2023) to further align responses with students' personality traits and learning factors.

3.4.1 Stage 1. Strategy-Aware Response Generation

We convert 1,964 multi-turn dialogues into 7,286 single-turn teacher-response instances and split them into training and validation sets with a 9:1 ratio. For the i-th turn, the input is defined as $X_i = \{P, Q, A, H_i, T\}$, where P is the student profile, Q is the question, A is the answer, $H_i = \{r_0^T, r_0^{St}, \ldots, r_{i-1}^T, r_{i-1}^{St}\}$ is the dialogue history, and T is the task instruction. The model is trained to generate the teacher response r_i^T. We adopt LoRA (Hu et al., 2022) for efficient fine-tuning and optimize the standard cross-entropy loss:

$$\mathcal{L}_{SFT} = -\sum_{j=1}^{|r_i^T|} \log p(r_{i,j}^T \mid X_i, r_{i,1:j-1}^T). \quad (1)$$

The resulting model is denoted as π_{SFT}.

Table 1: Comparison of existing math tutoring dialogue datasets. #Dials. and #Strat. indicate the number of dialogues and teaching strategies, while Pers., LF, and Cog. stand for personality traits, learning preferences, and cognitive factors.

Dataset	#Dials.	Turns	Avg. Length		Student Profile			#Strat.	Settings	Lang
			Stu.	Tea.	Pers.	LF	Cog.			
SocraTeach	35,151	5.28	16.4	30.3	✗	✗	✓	1	3-agent roleplay	EN
MATHDIAL	2,861	4.96	37.38	15.08	✗	✗	✓	4	1:1 semi-synthetic	EN
SocraticMATH	6,846	5	38.82	54.50	✓	✗	✗	4	2-agent roleplay	CH
PACEdata	1,410	10	44.67	54.56	✓	✓	✗	Unfixed	2-agent roleplay	EN
PMTDial (Ours)	1964	6.47	67.33	72.86	✓	✓	✓	28	3-agent roleplay	EN

3.4.2 Stage 2. Preference Optimization via DPO

To enhance personalization, we construct preference pairs based on two profile-sensitive dimensions: empathy and learning support. Specifically, responses from π_{SFT} are evaluated using these two metrics. High-scoring responses are treated as preferred samples y_w, while GPT-4o is prompted to generate lower-quality counterparts y_l; for low-scoring responses, we generate improved counterparts as y_w. Ambiguous cases with medium scores are discarded and regenerated. This yields empathy pairs D_e and learning-support pairs D_l, which are merged into the final preference dataset $D = \{(x, y_w, y_l)^{(i)}\}$.

We initialize both the policy model π_θ and reference model π_{ref} from π_{SFT}, and optimize the DPO loss:

$$\mathcal{L}_{\text{DPO}} = -\,\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\Bigg[\log\sigma\Big(\beta\log\frac{\pi_\theta(y_w \mid x)}{\pi_{\text{ref}}(y_w \mid x)} - \beta\log\frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)}\Big)\Bigg] \quad (2)$$

where β controls the deviation from the reference policy. This objective encourages the model to assign higher likelihood to personalized and pedagogically supportive responses.

4 Experimental setup

4.1 Implementation Detail

For multi-agent simulation, GPT-4 drives the student agent due to its stronger ability to simulate student errors and diverse responses, while GPT-5 mini is used for the teacher and supervisor agents to balance performance and cost. We set the temperature to 0.6 for teacher and student responses to promote diversity, and 0.3 for supervisor responses to ensure rigor.

We simulate 40 students solving 50 GSM8K elementary math problems each, resulting in 1,964 filtered multi-turn dialogues and 7,286 teacher responses. The data is split 9:1 for SFT training and validation. Starting from Qwen2.5-7B-Instruct, we fine-tune the model for 3 epochs with LoRA, using rank 8, α = 32, batch size 2, learning rate 2e-4, a cosine scheduler, and a warm-up ratio of 0.1, obtaining π_{SFT}.

We then apply DPO to π_{SFT} using 14,338 preference pairs for 5 epochs, with batch size 2, learning rate 1e-5, gradient accumulation of 8 steps, β = 0.1, and LoRA settings r = 32 and α = 16. All training is conducted on a single NVIDIA GeForce RTX 4090 GPU with 48 GB memory, yielding π_{DPO}, our final PMTutor model.

4.2 Metrics

To evaluate teacher-generated responses while ensuring the realism of simulated students, we design metrics from both teacher and student perspectives. For student responses, we focus on linguistic diversity and human-likeness. The indicators are as follows:

- **Word Count:** Measures response length, reflecting student engagement and elaboration.
- **Sentence Count:** Counts the number of sentences to assess structural completeness.
- **Distinct-n:** Evaluates lexical diversity at different n-gram levels, indicating the richness of student language.
- **Human-Like Win Rate:** Measures how often student responses are judged as more human-like than baselines, capturing naturalness and authenticity.

For teacher-generated responses, we follow prior studies (Liu et al., 2024a, 2025b) and incorporate refinements from experienced educators to define five key indicators:

- **Clarity:** Assesses whether the response is clear, concise, and free of redundancy.
- **Guidance:** Measures whether the teacher encourages independent thinking instead of giving answers directly.
- **Empathy:** Evaluates whether the teacher considers the student's personality or emotions.
- **Learning Support:** Determines whether the response supports motivation, self-regulation, math mindset, or metacognition.
- **Thinking Cultivation:** Assesses whether the response promotes higher-order thinking, such as critical thinking, problem-solving, or reflection.

Each metric is scored on a 1–10 scale, with criteria provided at two-point intervals to ensure evaluation reliability.

4.3 Baselines

We compare our model PMTutor with a range of baseline methods in two categories: (i) closed-source models, including `GPT-4o` and `DeepSeek-V3`, and (ii) open-source models, including the base model `Qwen2.5-7B-Instruct` and the intermediate model π_{SFT} from the first stage of supervised fine-tuning, to clearly illustrate the effects of our two-stage training framework.

5 Results

5.1 Main Results

Our experiments mainly focus on assessing large language models as teaching assistants from two perspectives: single-turn response quality given specific context, and overall effectiveness as independent ITS systems guiding students without prior dialogue history.

5.1.1 Context-Given Evaluation

We first evaluated model performance under given contexts, including student profile, question, answer, and dialogue history. We tasked 40 simulated students with solving ten novel math problems, generating 400 interactive dialogues (1,403 teacher turns) as the test set. Models were prompted to generate responses based on the same contexts, and GPT-5 mini scored them across five key dimensions: Clarity, Guidance, Empathy, Learning Support, and Thinking Cultivation. Results are summarized in Table 2 (upper part).

It is indicated that the model fine-tuned in the initial SFT stage produced the most concise and clear responses, achieving high scores in Clarity, Guidance, and Thinking Cultivation. PMTutor further improved upon these by significantly enhancing Empathy and Learning Support, resulting in the highest overall performance among all models. Notably, while the closed-source model GPT-4o also showed strong performance in Empathy and Learning Support, it lagged behind in Guidance and Thinking Cultivation.

5.1.2 Context-Free Evaluation

We further conducted a context-free evaluation, where each model acts as an independent LLM-driven ITS and guides students from scratch without prior dialogue history. Following the same setup as the Context-Given evaluation, 40 simulated students solved 10 unseen problems.

As shown in the lower part of Table 2, all models show performance drops compared to the context-given setting, with Qwen2.5-7B-Instruct declining the most. This highlights the value of high-quality contextual information from our multi-agent framework, which helps compensate for base-model limitations and improves performance. Moreover, the two-stage SFT+DPO training brings significant gains over the base model, ranging from 9.45% to 17.96%, demonstrating the effectiveness of our training approach.

We also analyzed the average number of dialogue turns per problem, as shown in Figure 3. Overly short interactions often lead to suboptimal outcomes: GPT-4o and DeepSeek-V3 tend to use fewer turns by giving answers too quickly, potentially limiting students' independent thinking. In contrast, PMTutor produces longer multi-turn dialogues with richer guidance, better supporting step-by-step reasoning and deeper cognitive engagement.

5.2 Ablation Study

To systematically evaluate the contribution of the supervisor agent in our multi-agent framework, we conduct ablation studies on the test set from two perspectives: teacher supervision and student supervision.

5.2.1 Ablation on Teacher Supervision

We first remove teacher supervision (denoted as w/o SP) and compare it with the full model (w/ SP). As shown in Table 3, introducing supervision leads

Table 2: Results of Context-Given and Context-Free Evaluation. The best performance is shown in bold, and the second-best is underlined. LS = Learning Support; TC = Thinking Cultivation.

	Clarity	Guidance	Empathy	LS	TC	Average
Context-Given Evaluation						
GPT-4o	9.569	7.637	8.972	8.332	6.795	8.261
DeepSeek-V3	9.409	8.102	8.746	8.141	6.902	8.260
Qwen2.5-7B-Instruct	9.401	7.902	8.110	7.885	6.711	8.002
+SFT (π_{SFT})	**9.579**	8.243	8.568	8.010	7.491	8.378
+SFT & DPO (**PMTutor**)	9.480	**8.606**	**9.174**	**8.764**	**7.764**	**8.758**
Context-Free Evaluation						
GPT-4o	9.362	7.464	8.463	8.295	6.866	8.039
DeepSeek-V3	9.320	7.404	8.425	7.880	6.948	7.995
Qwen2.5-7B-Instruct	8.293	7.379	7.332	7.206	6.174	7.277
+SFT (π_{SFT})	8.741	**8.577**	7.894	7.640	6.886	7.790
+SFT & DPO (**PMTutor**)	**9.666**	8.489	**8.909**	**8.438**	**7.417**	**8.584**

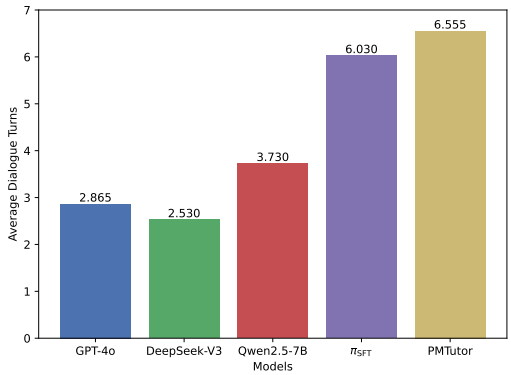

Figure 3: Comparison of the average number of teacher–student dialogue turns per model under the Context-Free Evaluation setting.

Table 3: Ablation study on teacher supervision.

	Clarity	Guidance	Empathy	LS	TC
w/o SP	9.667	7.622	8.259	8.119	6.944
w/ SP	9.561	8.888	8.764	8.386	7.682

to consistent improvements across most metrics. In particular, Guidance increases significantly from 7.622 to 8.888, and Thinking Cultivation rises from 6.944 to 7.682. These gains suggest that the supervised teacher delivers more structured, step-by-step guidance and better fosters students' reasoning ability.

The improvements mainly stem from the supervisor enforcing key pedagogical constraints: ensuring strategy adherence, maintaining consistency between selected strategy and response, discouraging direct solution revelation, and promoting incremental, student-adaptive support.

Table 4: Ablation study on student supervision.

	#Words	#Sents	Dist-1	Dist-3	Dist-5
w/o SP	42.32	4.75	0.807	0.979	0.980
w/ SP	69.91	6.48	0.755	0.984	0.995

5.2.2 Ablation on Student Supervision

We further analyze the impact of supervision on the student agent. As shown in Table 4, incorporating student supervision significantly improves response richness and realism. The average word count per response increases from 42.32 to 69.91, and the average sentence count rises from 4.75 to 6.48. These changes indicate that the supervised student produces more detailed, coherent, and expressive utterances, better simulating real student behavior.

Regarding lexical diversity, the model without supervision (w/o SP) shows a higher Distinct-1 score (0.807 vs. 0.755), mainly because its shorter responses have fewer repetitions. However, the supervised model (w/ SP) achieves better performance on Distinct-3 and Distinct-5, suggesting it generates longer and more structurally complex expressions. Overall, student supervision effectively leads to richer, more natural, and linguistically diverse student responses.

5.3 Human Evaluation

To validate the real-world applicability of our findings, we conducted multi-faceted human evaluations.

5.3.1 Student Simulation Realism

First, we assessed the authenticity of simulated student responses using the Human-Like Win Rate

metric through both automated (GPT-5 mini) and human evaluations. As shown in Figure 4, responses with student supervision achieve win rates of 56.70% (automated) and 68.40% (human), compared to 43.30% and 31.60% without supervision, corresponding to absolute improvements of 13.40 and 36.80 percentage points. This confirms that our supervision strategy, which enforces Big Five personality expressions and common mathematical errors, produces significantly more human-like student behaviors.

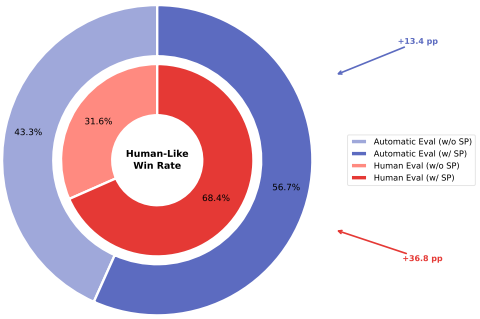

Figure 4: Human-Like Win Rate of student responses with and without supervisor guidance.

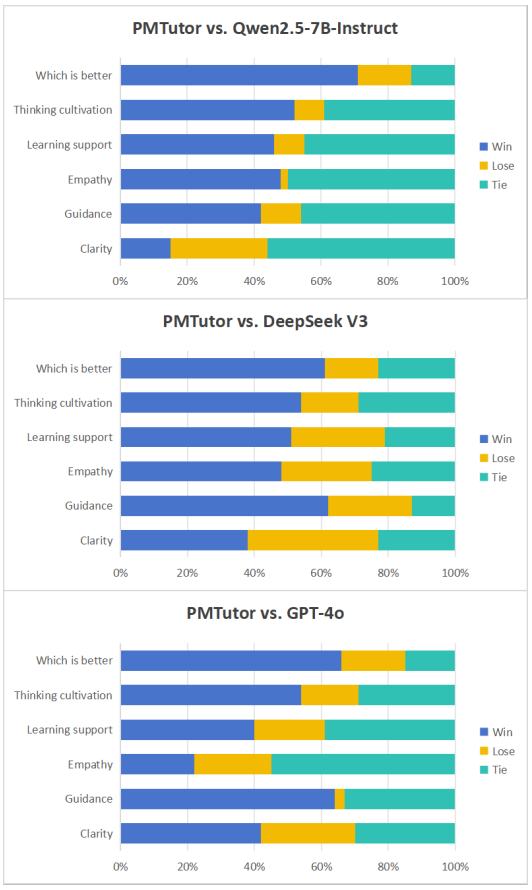

Figure 5: Comparison between PMTutor and baseline models on win–lose–tie preferences.

5.3.2 Expert Pairwise Preference

To complement automatic evaluation, we conducted human pairwise comparisons on the Context-Given test set. We sampled 50 instances for each of three settings: PMTutor vs. Qwen2.5-7B-Instruct, PMTutor vs. GPT-4o, and PMTutor vs. DeepSeek-V3. Six experienced annotators were organized into three pairs, with each pair evaluating one comparison. As shown in Figure 5, PMTutor substantially outperforms its base model and achieves over 60% win rate against both GPT-4o and DeepSeek-V3. These results are highly consistent with automatic evaluations, with strong inter-annotator agreement ($\kappa > 0.65$).

5.3.3 User Experience

Finally, we assessed real students' preferences for different models as standalone ITS under context-free conditions by recruiting 20 Grade-7 students. Each student evaluated complete tutoring dialogues from five models on one of five selected problems and chose their top two preferred tutors. As shown in Figures 7 and 6, PMTutor exhibits a dominant and stable preference distribution. Students particularly liked its positive encouragement, confidence building, appropriate strategies, clear reasoning, and warm, human-like interaction style. See Appendix B for more detailed analysis.

6 Conclusion

In this paper, we present a novel multi-agent collaborative framework for personalized mathematics tutoring. By integrating fine-grained student modeling (Big Five personality traits, four key learning factors, personality-adaptive expressions, and probabilistically injected errors grounded in educational research), theory-driven instructional strategies, and a bidirectional supervisor agent, we construct PMTDial, a high-quality personalized tutoring dialogue dataset. Building upon this dataset, we develop PMTutor via a two-stage SFT+DPO fine-tuning process. Extensive automatic and human evaluations demonstrate that PMTutor consistently outperforms strong baselines, validating the effectiveness of our approach in delivering personalized and pedagogically sound tutoring.

Limitations

Despite the encouraging results, this work still has several limitations.

First, PMTDial is constructed based on elementary-level mathematical problems from GSM8K, and our experiments mainly focus on arithmetic reasoning and word-problem tutoring. While this setting provides a useful starting point for studying personalized mathematical tutoring, the effectiveness of PMTutor in broader mathematical domains, such as geometry, algebra, proof-based reasoning, and more advanced STEM subjects, remains to be further investigated. We plan to extend the framework to more diverse topics, grade levels, and problem types to better examine its generalizability.

In addition, PMTutor is intended to serve as a supportive tool for personalized mathematical guidance rather than a replacement for human teachers. In practical educational scenarios, students may differ substantially in background knowledge, learning needs, emotional states, and interaction preferences, which calls for more careful adaptation and monitoring. We will further explore appropriate human oversight, safety mechanisms, and uncertainty-aware response detection to ensure more reliable and responsible tutoring support.

Ethics Statements

The math problems used in this work are from publicly available datasets. For human evaluation, we recruited experienced annotators and Grade-7 students to assess model-generated tutoring dialogues. Participants were informed of the study purpose and evaluation procedure before participation. For student participants, consent was obtained from their guardians or school representatives where required. All collected responses were anonymized and used only for research analysis. No personally identifiable information was collected or released. The evaluation involved only reading anonymized tutoring dialogues and selecting preferred responses, posing minimal risk to participants.

Use of AI Assistants

We primarily used AI tools to assist with writing polishing and language refinement.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Yulin Chen, Ning Ding, Hai-Tao Zheng, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2024. Empowering private tutoring by chaining large language models. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 354–364.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Yuhao Dan, Zhikai Lei, Yiyang Gu, Yong Li, Jianghao Yin, Jiaju Lin, Linhao Ye, Zhiyan Tie, Yougen Zhou, Yilei Wang, and 1 others. 2023. Educhat: A large-scale language model-based chatbot system for intelligent education. *arXiv preprint arXiv:2308.02773*.

Digital Promise. 2022. The learner variability navigator. https://lvp.digitalpromiseglobal.org/.

Yuyang Ding, Hanglei Hu, Jie Zhou, Qin Chen, Bo Jiang, and Liang He. 2024. Boosting large language models with socratic method for conversational mathematics teaching. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 3730–3735.

Zhihao Fan, Jialong Tang, Wei Chen, Siyuan Wang, Zhongyu Wei, Jun Xi, Fei Huang, and Jingren Zhou. 2024. Ai hospital: Interactive evaluation and collaboration of llms as intern doctors for clinical diagnosis. *arXiv preprint arXiv:2402.09742*.

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.

IRIS Center. 2016. Mathematics: Identifying and addressing student errors. https://iris.peabody.vanderbilt.edu/case-study/mathematics-identifying-and-addressing-student-errors/. Vanderbilt University.

James A Kulik and John D Fletcher. 2016. Effectiveness of intelligent tutoring systems: a meta-analytic review. *Review of educational research*, 86(1):42–78.

Junkai Li, Siyu Wang, Meng Zhang, Weitao Li, Yunghwei Lai, Xinhui Kang, Weizhi Ma, and Yang Liu. 2024. Agent hospital: A simulacrum of hospital with evolvable medical agents. *arXiv preprint arXiv:2405.02957*.

Ben Liu, Jihai Zhang, Fangquan Lin, Xu Jia, and Min Peng. 2025b. One size doesn't fit all: A personalized conversational tutoring agent for mathematics instruction. In *Companion Proceedings of the ACM on Web Conference 2025*, pages 2401–2410.

Jiayu Liu, Zhenya Huang, Tong Xiao, Jing Sha, Jinze Wu, Qi Liu, Shijin Wang, and Enhong Chen. 2024a. Socraticlm: Exploring socratic personalized teaching with large language models. *Advances in Neural Information Processing Systems*, 37:85693–85721.

Zhengyuan Liu, Stella Xin Yin, Geyu Lin, and Nancy F Chen. 2024. Personality-aware student simulation for conversational intelligent tutoring systems. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 626–642.

Zhengyuan Liu, Stella Xin Yin, Geyu Lin, and Nancy F Chen. 2024b. Personality-aware student simulation for conversational intelligent tutoring systems. *arXiv preprint arXiv:2404.06762*.

Jakub Macina, Nico Daheim, Sankalan Pal Chowdhury, Tanmay Sinha, Manu Kapur, Iryna Gurevych, and Mrinmaya Sachan. 2023. Mathdial: A dialogue tutoring dataset with rich pedagogical properties grounded in math reasoning problems. *arXiv preprint arXiv:2305.14536*.

Christopher J MacLellan and Kenneth R Koedinger. 2022. Domain-general tutor authoring with apprentice learner models. *International Journal of Artificial Intelligence in Education*, 32(1):76–117.

Robert R McCrae and Oliver P John. 1992. An introduction to the five-factor model and its applications. *Journal of personality*, 60(2):175–215.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pages 1–22.

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.

Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, and 1 others. 2024. Chatdev: Communicative agents for software development. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15174–15186.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Jinxin Shi, Jiabao Zhao, Yilei Wang, Xingjiao Wu, Jiawen Li, and Liang He. 2023. Cgmi: configurable general multi-agent interaction framework. *CoRR*.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Cyrille AC Van Bragt, Anouke WEA Bakx, Theo CM Bergen, and Marcel A Croon. 2011. Looking for students' personal characteristics predicting study outcome. *Higher Education*, 61(1):59–75.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. 2023. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*.

Tianfu Wang, Yi Zhan, Jianxun Lian, Zhengyu Hu, Nicholas Jing Yuan, Qi Zhang, Xing Xie, and Hui Xiong. 2025. Llm-powered multi-agent framework for goal-oriented learning in intelligent tutoring system. In *Companion Proceedings of the ACM on Web Conference 2025*, pages 510–519.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, and 1 others. 2024. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*.

Murong Yue, Wenhan Lyu, Wijdane Mifdal, Jennifer Suh, Yixuan Zhang, and Ziyu Yao. 2024. Mathvc: An llm-simulated multi-character virtual classroom for mathematics education. *arXiv preprint arXiv:2404.06711*.

Shengbin Yue, Ting Huang, Zheng Jia, Siyuan Wang, Shujun Liu, Yun Song, Xuanjing Huang, and Zhongyu Wei. 2025. Multi-agent simulator drives language models for legal intensive interaction. *arXiv preprint arXiv:2502.06882*.

Zheyuan Zhang, Daniel Zhang-Li, Jifan Yu, Linlu Gong, Jinchang Zhou, Zhiyuan Liu, Lei Hou, and Juanzi Li. 2024. Simulating classroom education with llm-empowered agents. *arXiv preprint arXiv:2406.19226*.

A Details on Agent Role Setting

A.1 Student Agent

A.1.1 Expression Base

Inspired by the BF-TC model (Liu et al., 2024) originally proposed for language learning, we develop a Big Five-based ***Expression Base*** tailored to mathematical problem-solving contexts (a shortened version is presented in Table 5). This design captures how different personality traits shape students' behavioral tendencies and verbal expressions during reasoning.

A.1.2 Error Base

To enhance the realism of the simulated student, we construct an ***Error Base*** grounded in the IRIS Center case studies. The Error Base categorizes common mathematical errors into three types: factual, procedural, and conceptual. Representative examples are shown in Table 6.

A.1.3 Rule-based Error Injection Mechanism

We implement a rule-based probabilistic mechanism for error injection. The base error probability is initialized to 0.5 and then dynamically adjusted based on the student's personality traits and learning factors as follows:

- **Conscientiousness**: High decreases the probability by 0.2; low increases it by 0.2.
- **Neuroticism**: High increases the probability by 0.1; low decreases it by 0.1.
- **Self-regulation Ability**: Strong decreases the probability by 0.1; weak increases it by 0.1.
- **Motivation**: Positive decreases the probability by 0.1; negative increases it by 0.1.
- **Metacognition Ability**: Strong decreases the probability by 0.2; weak increases it by 0.2.

The resulting probability is clamped to the range $[0.1, 0.9]$. This mechanism ensures that error occurrence correlates naturally with the student's profile rather than being injected randomly.

Table 5: Student Behaviors by Big Five Personality Traits

Trait	High	Low
Openness	Suggests alternatives; tries new methods; asks about concepts.	Resists alternatives; reluctant to try new methods; shows little curiosity.
Conscientiousness	Explains steps clearly; checks reasoning; revises based on feedback.	Gives vague or incomplete steps; ignores errors; appears distracted or disengaged.
Extraversion	Speaks actively; expresses ideas clearly; keeps the conversation going.	Speaks very little; hesitates to answer; lets the teacher lead.
Agreeableness	Responds supportively; accepts correction cooperatively; speaks politely.	Shows little interest in guidance; responds passively; appears uncooperative.
Neuroticism	Shows anxiety; seeks reassurance often; has emotional fluctuations.	Remains calm; shows confidence; has little emotional reaction.

Table 6: Error Base: Common mathematical errors used in student simulation.

Error Category	Specific Error	Examples
Common Factual Errors	Has not mastered basic number facts and makes errors in single-digit operations.	$3 + 2 = 7$, $2 \times 3 = 7$
	Misidentifies signs.	$2 \times 3 = 5$ (treats $\times$ as $+$)
	Misidentifies digits.	Identifies 5 as 2
	Makes counting errors.	1, 2, 3, 4, 5, 7, 8, 9 (skips 6)
	Does not know mathematical terms.	Does not understand "numerator", "denominator", "greatest common factor", etc.
	Does not know mathematical formulas.	Does not know the formula for the area of a circle
Common Procedural Errors	Performs the incorrect operation.	Adds instead of subtracts ($234 - 45 = 279$), multiplies instead of adds ($3 + 2 = 6$)
Common Conceptual Errors	Misunderstands place value.	writes "nine hundred seventy-four" as 90074
	Overgeneralizes.	Always subtracts the smaller from the larger ($321 - 245 = 124$)
	Overspecializes.	Associates right triangle only with a specific orientation

A.2 Teacher Agent

A.2.1 Strategy Base

An experienced teacher should select appropriate instructional strategies based on students' individual characteristics, rather than applying a one-size-fits-all approach. The Learner Variability Navigator (LVN) project (Digital Promise, 2022) exemplifies this by identifying diverse learning factors, such as learner background, social and emotional learning, and cognition, and provides instructional strategies aimed at enhancing each factor, which help guide teachers in tailoring personalized instruction. Building on this foundation, We invited experts to manually select strategies aligned with four key learning factors in our student profiles: *Motivation, Self-regulation Ability, Math Mindset, and Metacognitive Ability*, then constructed a ***Strategy Base*** containing 28 strategies in total, as shown in Table 7.

Table 7: Teaching strategies grouped by student learning factors.

Learning Factors	Strategies
Motivation	Building Trusting Relationships Direct Instruction: Patterning Discussing Emotions Providing Support and Guidance Model Positive Connections to Math Real-world Math Providing Resources Strengths-based Approach Wait Time
Self-regulation Ability	Concrete-Representational-Abstract (CRA) Goal Setting & Monitoring Model Assignment or Skill Self-monitoring Writing to Learn Math
Math Mindset	Cognitively Demanding Tasks Error Analysis Foster Growth Mindset Guided Inquiry Response Devices Student-generated Problems Worked Examples Growth Mindset Feedback
Metacognition Ability	Direct Instruction: Problem Structures Direct Instruction: Problem-solving Strategies Explaining Their Thinking Reflect on Learning Encourage Student Self-advocacy Self-instructions

A.3 Supervisor Agent

To ensure the quality of generated dialogues, the supervisor agent evaluates and revises the outputs of the teacher and student agents using the following prompts.

A.3.1 Teacher Supervision Prompt

```
Given the dialogue history and the teacher's latest response,
you need to determine whether the teacher's latest response
need to be revised:

1. The response must strictly follow the required format:
Strategy:
Response:
2. Determine whether a strategy is needed for this turn.
- If no suitable strategy applies, the teacher may set the
strategy to None.
- If a strategy is selected, ensure it is chosen from the
Reference Strategies and is appropriate; otherwise, remind the
teacher to reselect.
3. When a strategy is applied, ensure the response aligns with
it; otherwise, revise the response accordingly.
4. The response should be tailored to the student's
personality.
5. Teachers should focus on guidance and avoid directly
providing solution approaches, formulas, or answers unless the
student asks.
6. Teachers should guide the student step by step and avoid
providing multiple steps in a single reply.
7. The response should be concise and clear.
8. When the student has fully solved the current problem,
instruct the teacher to output <end> and end the dialogue.

Reference Strategies:
{teacher.strategy}

Student Personality:
{student.profile}

Dialogue History:
{history}
Teacher's Latest Response:
{to_revise}

Return the result as:
Check Result: 0/1
<Revision Suggestion>: ...

Where:
- 1 means the teacher's response meets the requirements and
needs no changes.
- 0 means the teacher's response does not meet the requirements
and needs revision.

Keep your suggestions as concise as possible.
```

A.3.2 Student Supervision Prompt

```
Given the dialogue history and the student's latest response,
your task is to check whether the student's response in the
current round contains both personality expression and errors.

The student's profile is:
{student.profile}
The Big Five personality expression is:
{student.bf_exp}
The possible error is:
{student.error}.

Dialogue History:
{history}

Student's Latest Response:
{to_revise}

Return the checking result as a binary tuple (x, y),
where:
- x = 1 if the response contains the required personality
expression, otherwise 0;
- y = 1 if the response contains the specified error, otherwise
0.

Ideally, both x and y should be 1. If either of them is 0, you
should provide a concise revision suggestion to guide the
student to modify the response accordingly.

Output format:
Check Result: (x, y)
<Revision Suggestion>: ...

## Output Examples:
```

```
Example 1:
Check Result: (1, 1)
<Revision Suggestion>: <No Revision>

Example 2:
Check Result: (1, 0)
<Revision Suggestion>: The response does not include the
specified error. Please revise the response to incorporate the
required error.
```

B User Experience Analysis

In addition, our experiment examined whether PMTutor, as an independent Intelligent Tutoring System (ITS), aligns with the actual preferences of real students. To this end, we randomly selected 5 problems from the test set used in the Context-Free Evaluation. For each problem, we prepared complete dialogue transcripts in which five different models (GPT-4o, DeepSeek-V3, Qwen2.5-7B-Instruct, π_{SFT}, and PMTutor) independently tutored a student from start to finish. We recruited 20 first-year junior high school students (Grade 7) and asked them to read each set of dialogues, imagine themselves as the student in the conversation, and decide which tutor they would prefer to learn from. They were then asked to select their top two preferred teacher responses out of the five and explain their reasons for choosing them. Considering the cognitive and comprehension levels of junior high students, and to ensure the quality of the collected responses, each student was asked to evaluate only one problem.

The results of the clustering heatmap (as shown in Figure 7) reveal that Model A (PMTutor) exhibits a dominant and distinctive preference distribution, branching off early from the other four models. It was selected as one of the top two choices with a significantly higher probability than the others, followed by π_{SFT} (Model D). The remaining three models (Models B, C, and E) form a subcluster characterized by relatively lower and more consistent preference rates. We further analyze the students' open-ended explanations and find that PMTutor stands out for its high emotional value: it provides positive and encouraging feedback, builds students' confidence, employs appropriate strategies, demonstrates clear logical reasoning, and interacts in a warm, friendly, and human-like manner. In contrast, π_{SFT} is particularly appreciated for its strong scaffolding ability, as it effectively breaks down complex problems into smaller, manageable parts and guides students step by step in a patient and systematic manner, and this concise and clearly structured style is preferred by students.

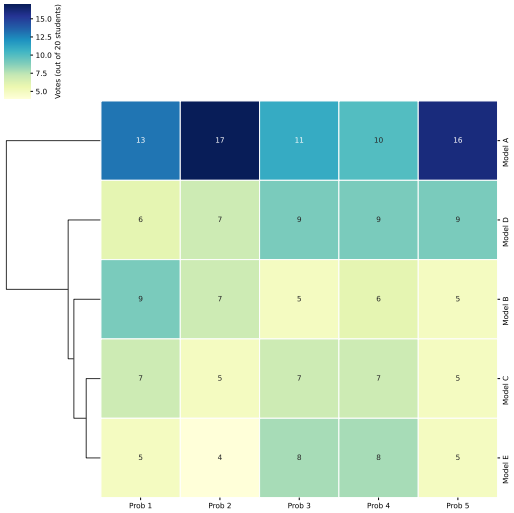

Figure 7: Global Preference Heatmap with Hierarchical Clustering (N=20 junior high students).

We further performed a Preference Shift Analysis to evaluate the robustness and task-sensitivity of each model. As shown in Figure 6, the length of the "dumbbell" visually represents the degree of performance fluctuation each model exhibits across different problems. Notably, Model A (PMTutor) maintained a dominant lead with high stability across contexts. Model D (π_{SFT}) demonstrated a significant positive shift in certain problems, substantially narrowing the gap with the leading model and revealing strong potential under specific contexts. In contrast, Models B, C, and E exhibited lower preference votes with relatively stable patterns, con-

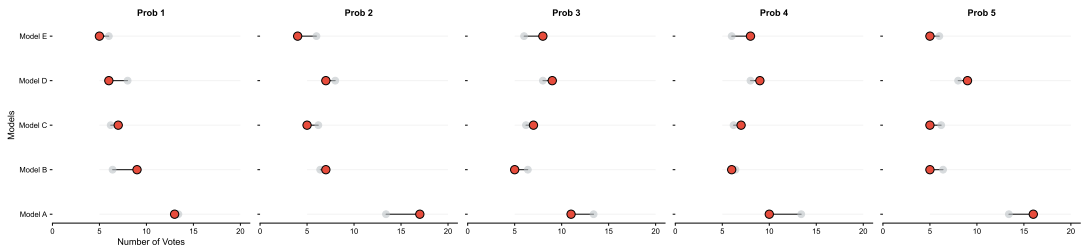

Figure 6: Preference Shift Analysis between problem-specific and global average votes.

sistently forming a sub-cluster of less-preferred models. Overall, these findings underscore PMTutor's superior robustness and consistent appeal to students, while highlighting the context-dependent strengths of π_{SFT}.