Towards Explainable Computerized Adaptive Testing with Large Language Model

Cheng Cheng[1], **Guanhao Zhao**[1], **Zhenya Huang**[1,2*], **Yan Zhuang**[1], **Zhaoyuan Pan**[1], **Qi Liu**[1,2], **Xin Li**[2,3], **Enhong Chen**[1,2]

[1]State Key Laboratory of Cognitive Intelligence, University of Science and Technology of China
[2]Institute of Artificial Intelligence Comprehensive National Science Center
[3]iFLYTEK Research

{doublecheng, ghzhao0223, zykb, zhaoyuanpan}@mail.ustc.edu.cn
{huangzhy, qiliuql, leexin, cheneh}@ustc.edu.cn

Abstract

As intelligent education evolves, it will provide students with multiple personalized learning services based on their individual abilities. Computerized adaptive testing (CAT) is designed to accurately measure a student's ability using the least questions, providing an efficient and personalized testing method. However, existing methods mainly focus on minimizing the number of questions required to assess ability, often lacking clear and reliable explanations for the question selection process. Educators and students can hardly trust and accept CAT systems without an understanding of the rationale behind the question selection process. To address this issue, we introduce **L**LM-**A**gent-Based **CAT** (LACAT), a novel agent powered by large language models to enhance CAT with human-like interpretability and explanation capabilities. LACAT consists of three key modules: the Summarizer, which generates interpretable student profiles; the Reasoner, which personalizes questions and provides human-readable explanations; and the Critic, which learns from past choices to optimize future question selection. We conducted extensive experiments on three real-world educational datasets. The results demonstrate that LACAT can perform comparably or superior to traditional CAT methods in accuracy and significantly improve the transparency and acceptability of the testing process. Human evaluations further confirm that LACAT can generate high-quality, understandable explanations, thereby enhancing student trust and satisfaction. [1]

1 Introduction

As intelligent education has evolved significantly, computerized adaptive testing (CAT), which aims to accurately measure a student's ability with as few questions as possible, provides an efficient and personalized testing method for downstream educational services, such as learning resource recommendation, and learning status report. (Van der Linden and Glas, 2000). Generally, CAT selects appropriate questions that align with each student's individual performance, further delivering equivalent testing accuracy levels with fewer test items compared to conventional paper-and-pencil exams (Cheng, 2009).

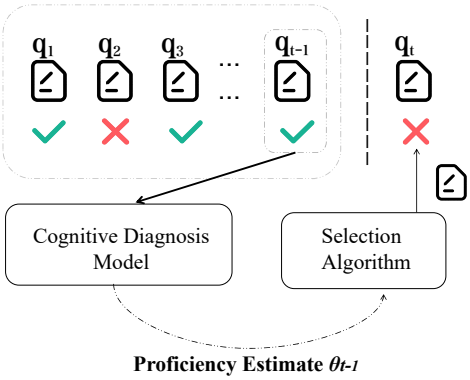


Figure 1: Overview of CAT workflow.

Figure 1 illustrates a toy example of a typical CAT. In practice, CAT involves two core components: the Cognitive Diagnosis Model (CDM), which estimates students' abilities, and a selection strategy that identifies the most appropriate questions based on their past performance (Embretson and Reise, 2013; Zhao et al., 2023). The workflow of the CAT system is as follows. Given a student, in step $t-1$, the selection algorithm selects an exercise based on the CDM's estimate of the student's current abilities, denoted by θ_{t-1}. Subsequently, the students accept and answer the exercise. After receiving the answer, the CDM updates the estimate to θ_t based on the new answer.

Traditionally, heuristic rules such as maximizing Fisher information (Chang et al., 2015), KLI (Chang and Ying, 1996) have been used

*Corresponding author
[1]Code and data are publicly available at https://github.com/doublecheng12/LACAT

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 2655–2672
November 12-16, 2024

for question selection. However, these methods struggle with generalization and modeling complex interactions between students and questions. Recently, the focus has shifted to deep learning methods like NCAT (Zhuang et al., 2022), BOBCAT (Ghosh and Lan, 2021), and GMOCAT (Wang et al., 2023b), which learn selection algorithms from large-scale student response data and have improved test accuracy. Despite their success, these black-box systems lack transparency and fail to provide reasonable explanations (Liu et al., 2024a), leading to potential distrust among teachers and students. Hence, developing interpretable and transparent adaptive testing models is crucial for their widespread adoption and trust (Wainer et al., 2000).

Nowadays, the emergence of large language models (Achiam et al., 2023) offers a new perspective on solving this problem. Recent research has demonstrated their power in textual reasoning (Zhang et al., 2024a), achieving significant results in tasks like text summarization (Luo et al., 2023) and sentiment analysis (Zhang et al., 2023a). Additionally, these models possess powerful language generation capabilities, enabling them to produce human-readable explanatory text for their decision-making processes. However, although large language models excel in language processing tasks, the integration of LLM within the framework of computerized adaptive testing remains highly challenging.

Three primary challenges can be delineated in tackling the explainable computerized adaptive testing task using large language models (LLMs). Firstly, the Cognitive Diagnostic Model (CDM) generally represents the estimation of a student's abilities using a single or multi-dimensional vector. However, relying solely on a vector is insufficient for guiding an LLM in question selection. Secondly, while the primary goal of computerized adaptive testing is to assess students' abilities accurately, the lack of a precise standard for question selection and determining the optimal question poses a significant challenge. This is especially true for LLMs, which must also explain their question selections. Thirdly, the lack of an effective feedback mechanism to optimize question selection is a challenge. In computerized adaptive testing, decision-making is continuous, with past selections and experiences crucial for current decisions. Ensuring past decisions improve future question selection is essential for maintaining accurate assessments.

To address these challenges, we propose an LLM-Agent-Based CAT (LACAT). LACAT incorporates three essential components: (1) The Summarizer, which accurately depicts user profiles from student response data and further enhances the understanding of students' abilities, addresses the challenge of insufficient selection guidance from student's abilities estimation vectors. (2) The Reasoner, which selects the most suitable questions and provides clear, human-readable explanations, tackles the challenge of the lack of a precise standard for question selection and explanation. (3) The Critic, which reflects on past question selections and gathers insights to improve future decisions, addresses the lack of an effective feedback mechanism for optimizing question selection. In summary, our contributions are:

- To the best of our knowledge, our method is the first to integrate large language models (LLMs) into computerized adaptive testing (CAT) for explainable question selection, introducing a new perspective and methodology.
- We introduce the LACAT framework, comprising three modules that leverage LLMs to understand students' cognitive abilities, select the most appropriate questions, and then provide clear and comprehensible explanations.
- Extensive experiments on three real-world public datasets demonstrate the efficacy of LACAT. Human evaluations further confirm the high quality of its generated explanations.

2 Related Works

2.1 Computerzied Adaptive Testing

In classical Computerized Adaptive Testing (CAT) systems, Item Response Theory (IRT) is typically used as the Cognitive Diagnostic Model (CDM) to predict student responses to questions. Recently, neural network-based CDMs, such as NCDM (Wang et al., 2020), ICDM (Liu et al., 2024b) have also emerged. The selection algorithm plays a crucial role in CAT systems. Initially, selection algorithms were based on information metrics like MFI (Chang et al., 2015) and KLI (Chang and Ying, 1996). Subsequently, rule-based methods, such as MAAT (Bi et al., 2020) using active learning, were proposed. Recent works have focused on data-driven approaches. For instance, BOBCAT (Ghosh and Lan, 2021) and NCAT (Zhuang

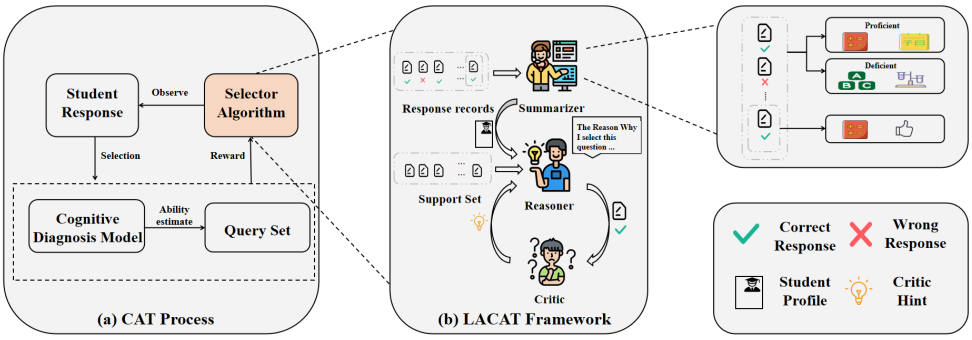


Figure 2: Illustration of (a) a typical process of CAT (Left), and (b) the overflow of LACAT(Right), which integrates three components, i.e., Summarizer, Reasoner, and Critic. The Support Set and Query Set are randomly selected from the pool of candidate questions.

et al., 2022) treat CAT as a reinforcement learning (RL) (Kaelbling et al., 1996) problem and train selection algorithms directly from large-scale student response data. Additionally, to achieve a more general and scalable framework, researchers have proposed the bounded estimation CAT framework (BECAT) (Zhuang et al., 2024), which redefines the selection problem using a data summary approach. However, these methods primarily aim to reduce test length and do not address the interpretability of CAT selections. To our knowledge, no current work focuses on the transparency and explainability of CAT selection algorithms.

2.2 LLM-Based Agent

With the rapid development of large language models, increasing efforts have been made to utilize these models as agents for decision-making (Yang et al., 2024; Zhao et al., 2024a). These models often possess reflective (Wang et al., 2023c; Pan et al., 2023; Li et al., 2023), memorization (Zhong et al., 2024; Sun et al., 2023; Li and Qiu, 2023), and tool-using capabilities (Zhao et al., 2024b) and have been applied in diverse fields such as game decision-making (Ma et al., 2023; Zhu et al., 2023; Wang et al., 2023a), finance (Li et al., 2024b; Koa et al., 2024), and recommendation systems (Shi et al., 2024; Zhou et al., 2024; Zhang et al., 2023b). In the field of education, LLM-based agents have been applied to student state modeling tasks, such as knowledge tracking (Li et al., 2024a) and simulating student behavior (Xu et al., 2024). However, existing research typically addresses individual aspects of intelligent education, such as student ability modeling, without integrating proper modules to accomplish complex multi-step tasks. Thus, they are not suitable to tackle the explanation problem in the CAT task. In this paper, We propose a multi-module framework, LACAT (LLM-Agent-Based CAT), which enhances the interpretability and transparency of multi-step decision-making in Computerized Adaptive Testing (CAT) through the collaborative work of the Summarizer, Reasoner, and Critic modules.

3 PRELIMINARIES

3.1 Computerized Adaptive Testing (CAT)

In Computerized Adaptive Testing (CAT), as illustrated in Figure 2(a), two core components, the cognitive diagnostic model and the selection algorithm, work in tandem through a series of steps $t \in [1, T]$. Initially, the candidate questions pool is randomly split into the support and query set. The selection algorithm identifies the most appropriate question from the support set for the student to respond to. Post-response, the cognitive diagnostic model estimates the student's true ability θ_t based on their response record, denoted as (q_t, a_t, c_t), where q_t is the question answered by the student at step t, a_t is the response (1 if correct, 0 otherwise), and c_t is the text content of the question. When combined with θ_t, this query set helps compute feedback to assess the accuracy of the ability estimation, as illustrated in Figure 2(a). This feedback is then used to guide the selection of the next question, refining the process to match the student's demonstrated abilities better.

3.2 Problem Statement

As mentioned above, Computerized Adaptive Testing System involves selecting a question q from the support set for student u_i at each step t based on the student's historical records and feedback from the Cognitive Diagnostic Model (CDM). After receiving the student's response a to question q, the CDM updates and estimates the new ability level θ_t. This process is repeated T times to accurately

estimate the student's ability, aiming for θ_T to approach θ_0, where θ_0 represents the true (usually unknown) ability of student u_i.

Furthermore, we aim to integrate Large Language Models (LLMs) into the question selection process of Computerized Adaptive Testing (CAT) to enhance user modeling and improve the transparency, interpretability, and efficiency of ability estimation. This integration will enable the generation of clear and transparent explanations for question selection, thus making the process more understandable and accessible for students and educators. By leveraging LLMs, we can provide more personalized and accurate assessments, ultimately contributing to a more effective and insightful evaluation of student ability.

4 Method

4.1 Overall Workflow

As shown in Figure 2(b), the Summarizer is responsible for constructing the student profile, consisting of a long-term profile from all of the student's response records and a short-term profile from their recent response records. This profile serves as the basis for subsequent question selection, addressing the challenge of insufficient selection guidance typically provided by traditional ability estimation vectors. Then, the Reasoner selects the most suitable question from the support set based on the detailed user profile created by the Summarizer and insights from the Critic's evaluation of the previous round's question selection. Importantly, the Reasoner provides a clear and reasonable explanation for each selection, ensuring transparency and interpretability. The Critic's evaluation is based on the reward provided by the environment. If the reward is low, the Critic analyzes the issues in the previous question selection to avoid repeating the same mistakes in future selections, thereby optimizing the quality of the question selection and achieving a more precise capability assessment.

Specifically, given that a new student has no response, a representative item from the support set, such as one with high distinction and medium difficulty, will first be selected for them. Then, the Summarizer will obtain the student's record and analyze their profile. The Reasoner selects the most suitable exercise for the student based on their profile and provides reasons for the selection. If the quality of the exercise selection is not satisfactory, the Critic will analyze the reasons to prevent the same error and provide hints for the next selection round. These three components work in a cycle until the end of the test.

4.2 Summarizer

The Summarizer module comprises both long-term and short-term profile systems. As illustrated in Figure 2(b), the long-term profile repository meticulously analyzes and catalogs student-specific features extracted from their entire response history. It systematically identifies and archives areas of strength, weakness, and persistent challenges over time, providing a comprehensive view of each student's unique educational journey. Concurrently, the short-term profile system focuses on the immediate evaluation of the student's most recent response, conducting a real-time analysis to synthesize an informed guiding thought. This cognitive process is crucial for the adaptive selection of subsequent questions, ensuring their appropriateness at the challenge level and alignment with the student's immediate learning needs.

4.2.1 Long-term Profile

Students may exhibit consistent ability in exam performance over time. For instance, a student lacking in spatial visualization typically performs inadequately on solid geometry problems (Chen et al., 2023b; Zhao et al., 2021). Therefore, we propose to learn the long-term profile of students to capture this consistent ability. Specifically, for a student u_i at time step t, the profile utilizes the historical response records and the current ability estimate θ_t, which is derived from the parameters of the abstract CDM. This estimate takes the response records from all previous steps as input.

Note that the input for the long-term profile includes the response records from all previous steps. Thus, for each previous step $t' \in [1, t-1]$, the triple $(q_{t'}, a_{t'}, c_{t'})$ can be represented by $e_{t'}$. The student's current ability estimate θ_t is calculated using these previous responses.

$$L_t = \text{LLM}_{\text{long-term}}\left([e_1, e_2, \ldots, e_{t-1}], \theta_{t-1}, \text{prompt}_{\text{long-term}}\right). \quad (1)$$

The long-term profile primarily captures the student's state in each type of knowledge and serves as a valuable resource for selecting customized questions tailored to her individual needs.

Table 1: Example of Critic.

Critic Case
The selected trigonometric function problem was too difficult for the student, resulting in less-than-ideal outcomes. This mismatch may have caused the student frustration and hindered their progress. To improve this situation, we should adjust the difficulty of the problems to match the student's current skill level. By providing questions appropriate to the student's abilities, we can help them gradually build confidence and effectively promote learning.

4.2.2 Short-term Profile

Besides the long-term ability, when constructing students'profiles, their recent academic performance must also be considered. For instance, if students correctly answer the trigonometric function knowledge in the previous question, the difficulty level of the questions related to this knowledge point should be increased to gain a more precise assessment of their capabilities. We represent the short-term profile as a thought that guides the selection of the next question based on the student's long-term profile and their recent response history. Specifically, for a student u_i at time step t, the Summarizer utilizes the recent response record e_{t-1} as input to generate the short-term profile.

$$S_t = \text{LLM}_{\text{short-term}}\left([e_{t-1}], L_t, \text{prompt}_{\text{short-term}}\right). \tag{2}$$

In light of these comprehensive profiling mechanisms, the student profile is thus composed of both long-term and short-term components. Consequently, the Reasoner leverages this multifaceted student profile to make informed decisions in selecting the most appropriate subsequent question.

4.3 Critic

In the computerized adaptive testing (CAT) selection process, decision-making is continuous, with past selections and experiences playing a key role in current selections (Zhao et al., 2024a; Li et al., 2024c; He et al., 2024). With this in mind, we design the Critic module to extract guiding principles from past selections. When the quality of the selected questions is low, the Critic module analyzes the cause and provides guidance for the next selection. The quality of a question is defined by its ability to predict students' true abilities accurately. Although students' true abilities are usually unknown, we can use their query set (Zhuang et al., 2022) to measure the error of the ability estimate θ_t. During the test phase t, we use θ_t to calculate the prediction accuracy of the query set, denoted as $\text{ACC}(\theta_t)$. The higher the $\text{ACC}(\theta_t)$, the closer the estimated θ_t is to the true ability. Therefore, the quality of a question can be formally defined as follows:

$$r_{\text{qua}} = \begin{cases} 1, & \text{if } \text{ACC}(\theta_t) > \text{ACC}(\theta_{t-1}) \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

The detailed calculation process of $\text{ACC}(\theta_t)$ is described in Appendix C.

Questions are deemed high quality when they enhance the accuracy of ability assessments. The Critic module aims to simulate this quality assessment process and maximize r_{qua} over time. If the selected questions fall short of improving accuracy, the Critic assesses output quality and offers strategic direction through the following formulation:

$$C_t = \text{LLM}_{\text{critic}}\left([e_1, \ldots, e_{t-1}], \theta_{t-1}, \text{prompt}_{\text{critic}}\right). \tag{4}$$

To provide a clearer understanding, we present an example in Table 1. In this instance, the poor quality of question selection is due to a mismatch between the selected question and the student's ability. Specifically, a question on trigonometric functions was chosen that is far beyond the student's current capability. Therefore, the following selections should consider reducing the difficulty of questions in this knowledge point. Obviously, the Critic module helps the agent fully perceive past student ability and adapt their strategies to maintain high-quality selections.

4.4 Reasoner

The Reasoner aims to select the most appropriate questions for students of varying abilities based on their historical response records and to provide reasonable, transparent, and human-readable explanations. To achieve this, the Reasoner considers both students' long-term and short-term performance, combining these insights in the question selection process. Guided by the Critic, the Reasoner can promptly adjust the selection strategy. Additionally, we employ the chain of thought prompt (Wei et al., 2022) to enhance the effectiveness and explainability of question selection. Specifically, the Reasoner first identifies the characteristics of problems that should be selected for each student, paying particular attention to how each problem affects the student's ability profile. This approach ensures that the reasoning behind each choice is clear and easy to understand, thereby enhancing the transparency and credibility of the process.

Table 2: Example of Reasoner.

Reasoner Case
I selected question 1 because it involves solving a quadratic equation, an area where the student has recently shown significant improvement. This will help consolidate their algebra skills and build confidence. The moderate difficulty level can effectively assess their understanding and differentiate their ability in algebra. Additionally, focusing on algebra avoids their weaknesses in geometry, allowing them to concentrate on their strengths and enhance their overall learning experience.

For example, Table 2 shows a student who has significantly improved in solving quadratic equations, demonstrating a strength in this area, while still struggling with geometry, which remains a weakness. Based on this information, the Reasoner selects a question involving quadratic equations from the list of candidate questions to reinforce the student's algebra skills and build confidence. This selection leverages the student's strengths and provides an appropriate challenge to assess their understanding accurately. Avoiding geometry questions at this stage allows the student to focus on areas where they can perform well, enhancing the learning experience.

For test stage t, the final process can be formulated as:

$$(r_t, q_t) = \text{LLM}_{\text{Reasoner}}\left(L_{t-1}, S_{t-1},\ C_{t-1},\ [\text{support set}],\ \text{prompt}_{\text{Reasoner}}\right). \tag{5}$$

Here, r_t represents the reason for selecting q_t, and q_t is the selected question.

5 Experiments

In this section, we conduct extensive experiments on three real-world datasets to evaluate the performance of LACAT. Our experiments focus on answering the following research questions:

- **RQ1:** How does LACAT compare to existing question selection algorithms under the general CAT setting?
- **RQ2:** What is the quality of the Explanation of the different components (i.e., Summarizer, reasoner) generated by LACAT?
- **RQ3:** How does LACAT perform on Question Exposure and Context Overlap?
- **RQ4:** How do different components (i.e., Summarizer, Critic) influence our LACAT?

Furthermore, we conduct several quantitative experiments to further demonstrate the effectiveness of LACAT. More details can be found in Appendix B.

5.1 Experimental Settings

5.1.1 Data Partition and Experiment Process

We evaluate our LACAT method using three real-world educational datasets: Junyi, MOOCRadar, and EXAM. The Junyi dataset (Chang et al., 2015) consists of question logs from the 2018-2019 academic year on the online learning platform Junyi Academy. MOOCRadar (Yu et al., 2023) includes learning records from students in various MOOCs. The EXAM dataset, provided by Zhixue.com, contains records of junior high school students' mathematical exam performances. For all datasets, we remove students with fewer than 30 test records. Then, we allocate 80%, 10%, and 10% of the students for training, validation, and testing, respectively. Additionally, we divide the samples containing student's test records into a support question set, which comprises 80% of the samples, and a query question set, which comprises the remaining 20%. We use AUC (Area Under the Curve) and ACC (Accuracy) to evaluate the performance of each CAT method.

5.1.2 Compared Methods

In our experiment, we use Item Response Theory (IRT) (Embretson and Reise, 2013), which is the most popular Cognitive Diagnostic Model (CDM) in Computerized Adaptive Testing (CAT). To measure the effectiveness of LACAT, We first compare it with traditional CAT methods that do not require training, including **MFI** (Chang et al., 2015), **KLI** (Chang and Ying, 1996), **MAAT** (Bi et al., 2020), and **BECAT** (Zhuang et al., 2024). Furthermore, to evaluate the superiority of LACAT, we compare it with other data-driven CAT algorithms, specifically **BOBCAT** (Ghosh and Lan, 2021) and **NCAT** (Zhuang et al., 2022). Detailed implementations of the baseline methods are presented in Appendix A.

5.1.3 Implementation Details

We used GPT-3.5-turbo-16k provided by OpenAI API as the LLM backbone for its strong long context modeling ability, and the temperature of the model is set to 0. Moreover, We reproduced all baseline models using the public EduCAT library [2]. We use the recommended hyperparameters from the original papers.

[2] https://github.com/bigdata-ustc/EduCAT

Table 3: The performance of different methods on Student Score Prediction with ACC and AUC metrics. The boldfaced indicates the statistically significant improvements (p-value < 0.05) over the best baseline.

Dataset	MoocRadar			Junyi			Exam		
Metric	AUC			AUC			AUC		
Step	1	5	10	1	5	10	1	5	10
MFI	65.03	70.12	74.67	68.14	68.57	69.03	74.12	74.50	74.59
KLI	65.11	69.59	73.95	68.12	68.48	69.01	74.14	74.47	74.56
MAAT	65.45	69.71	74.17	68.18	68.70	69.54	74.17	74.53	74.64
BECAT	65.72	70.54	75.13	68.17	68.91	69.66	74.29	74.61	74.74
BOBCAT	65.97	71.34	75.78	68.23	69.07	69.73	**74.49**	74.73	74.97
NCAT	65.93	71.42	76.01	68.19	68.95	69.69	74.45	**74.85**	**75.08**
LACAT (Zero-shot)	**66.08**	**71.66**	**76.32**	**68.37**	**69.15**	**69.79**	74.22	74.60	74.71
Metric	ACC			ACC			ACC		
Step	1	5	10	1	5	10	1	5	10
MFI	74.67	83.14	86.67	71.20	71.93	72.08	64.44	65.35	66.47
KLI	74.76	83.28	86.66	71.23	71.99	72.13	64.39	65.27	66.40
MAAT	75.19	83.42	86.92	71.26	72.05	72.17	64.48	65.46	66.52
BECAT	75.36	83.54	87.06	71.39	72.07	72.20	64.55	65.50	66.61
BOBCAT	75.49	83.58	87.14	71.51	72.09	72.39	64.77	65.96	66.84
NCAT	75.58	83.97	87.36	71.43	72.07	72.22	**64.83**	**66.12**	**67.10**
LACAT (Zero-shot)	**75.70**	**84.35**	**87.48**	**71.75**	**72.13**	**72.42**	64.48	65.44	66.50

5.2 Student Performance Prediction (RQ1)

To verify the validity of the selection algorithm, we tested it on the student score prediction task of CAT. The evaluation methodology is detailed in Appendix D. As shown in Table 3, the AUC and ACC of students' responses to questions in the query set of the test set were used as evaluation metrics. We present the results for 1, 5, and 10 test steps. Our observations are as follows:

- LACAT demonstrates superior performance across both the Junyi and MoocRadar datasets, surpassing all baseline models. Remarkably, despite requiring no data training, LACAT outperforms data-driven approaches, validating the efficacy of our framework. This achievement underscores LACAT's capacity to effectively model the intricate relationships between students and exercises, highlighting its potential in educational assessment.

- LACAT exhibits suboptimal performance on the Exam dataset. We attribute this to the difficulty large language models have in recognizing the attributes of mathematical problems requiring complex reasoning. Despite being trained on many datasets, large language models still struggle with complex reasoning. Compared to Junyi's simple primary school math problems and the quiz questions in MoocRadar, large language models may make more errors on questions requiring complex reasoning.

5.3 Evaluation Results For Explainability (RQ2)

Since there was no existing reference on the explainability of CAT, inspired by previous experiments on text quality assessment (Yang et al., 2023; Chen et al., 2023a), we designed four indexes to evaluate the text generated by LACAT, including the student profile by the Summarizer and the selection reason by the Reasoner. The indexes are: 1) Fluency: the fluency and readability of the text. 2) Comprehensiveness: whether the generated text covers all the students' issues. 3) Relevancy: whether the generated text is relevant to the students' problem-solving situation. 4) Overall: the general effectiveness of the generated explanation.

Each aspect was rated on a scale from 0 to 2, with higher ratings reflecting more satisfactory performance. We extracted 50 samples each from the MoocRadar and Exam data sets, totaling 100 samples. These samples were distributed to 5 evaluators, including teachers and graduate students in the field of education. Specifically, we provided the evaluators with the record of the student's question responses, the student profile summarized by the Summarizer, and the Reasoner's explanations for selecting each question. Additionally, the emergence of large language models (LLMs) had sparked interest in their potential applications for evaluating natural language generation (NLG) tasks. To leverage these advancements, we utilized state-of-the-art models, specifically GPT-4 and Deepseek-v2, as automated evaluation methods to assess the quality of the generated text. Details of

Table 4: The Fleiss' Kappa statistics and average scores of human evaluations on LACAT explanations

Module	Fleiss' Kappa Statistics				Average Scores			
	Fluency	Relevance	Comprehensiveness	Overall	Fluency	Relevance	Comprehensiveness	Overall
Summarizer	0.94	0.51	0.46	0.66	1.92	1.63	1.72	1.78
Reasoner	0.98	0.63	0.59	0.60	1.97	1.57	1.60	1.52

the evaluation criteria are described in Appendix E.

5.3.1 Human Evaluation

First, we assess the quality of the annotations in the human evaluation results by calculating the inter-evaluator agreement using Fleiss' Kappa statistic (Falotico and Quatto, 2015) for each aspect. The Fleiss' Kappa results are presented on the left of Table 4. The evaluators' opinions were highly consistent, surpassing the standard for moderate agreement (>0.41) according to Fleiss' Kappa statistics. Particularly for the fluency index, they almost reached a complete agreement, reflecting the high quality of the manual annotations.

As shown in the right of Table 4, in terms of fluency, both the user portraits generated by the Summarizer and the questions selected by the Reasoner are human-readable and highly fluent, closely resembling human-written text, with an average score of almost 2.0. The average text scores generated by both the Summarizer and Reasoner in relevancy and completeness are over 1.5, indicating a clear and comprehensive understanding of students' abilities. However, the Reasoner performs slightly worse than the Summarizer in both indicators, possibly because question selection requires a deeper understanding of the student's state, making it a more complex task. Overall, both modules averaged over 1.5 points, demonstrating that LACAT can generate human-readable, clear explanations for correct classifications regarding fluency, relevance, and completeness, achieving the goal of making the CAT process transparent.

5.3.2 Automatic Evaluation

We asked the LLMs to choose among the following options: 1) Comprehensive (correct and thorough); 2) Partially good (reasonable but not comprehensive); 3) Invalid. Table 5 presents the automatic evaluation results of the LACAT explanations. The result is the average of the evaluations from the two models. The interpretations generated by the two modules show good performance, with over 90% of the content being valid. This reflects the high quality of the explanations.

Table 5: Automatic evaluation results for LACAT explanations

Methods	Quality	Summarizer	Reasoner
LACAT	Comprehensive	72.5%	64.5%
	Partially good	18.5%	25.5%
	Invalid	9.0%	10.0%

5.4 Question Exposure and Context Overlap (RQ3)

To gain a better insight into LACAT, we closely examine the characteristics of questions selected by the algorithm. We evaluated the novelty of different algorithms using two main metrics: question exposure rate and text overlap rate. Question exposure rate refers to how often students encounter the same questions in a single exam, while text overlap rate indicates the similarity between two or more tests. Lower exposure and overlap rates indicate a better algorithm. We adopted the methods proposed by Chang (Chang and Ying, 1999) and Chen (Chen et al., 2003) to measure these rates. As shown in Table 6, which lists the exposure rate (Exp.) and mean overlap rate (Over.) at step 10 using the MoocRadar dataset, LACAT demonstrates low question exposure rate and text overlap rate.

5.5 Ablation Study (RQ4)

In this subsection, we conduct ablation studies to investigate each module's contribution to LACAT further. We test AUC metrics at steps 3, 5, and 10 on IRT with the MoocRadar dataset. The settings are discussed as follows:

- **LACAT-S:** Removing the Summarizer module means we only use the student ability estimate provided by the IRT model, ignoring the student profile constructed from the student response sequence.
- **LACAT-C:** Removing the Critic module means that there is no mechanism to identify and rectify flaws in the reasoning process.

The results are presented in Table 7. We observe that removing any module leads to a decrease in LACAT performance. Our analysis is as follows:

Table 6: Exposure rate (Exp.) and mean overlap rate (Over.) at test step 10 with MoocRadar dataset. The best results are given in bold.

Dataset	**MoocRadar**	
Metric	Exp.%(mid)	Over.%
MFI	**0.13**	8.73
KLI	0.19	8.27
MAAT	0.20	11.71
BOBCAT	0.65	16.32
BECAT	0.21	6.09
NCAT	0.17	5.67
LACAT	0.18	**3.69**

Table 7: The results of ablation studies. We test on the AUC Metric at time step t = 3, 5, and 10 with the MoocRadar dataset. The best results are given in bold.

Metric	AUC@3	AUC@5	AUC@10
LACAT	**69.24**	**71.66**	**76.32**
LACAT-S	68.19	70.17	74.20
LACAT-C	68.73	70.88	75.17

- LACAT-S does not use the student portrait provided by the Summarizer module, resulting in a lack of crucial student information and subsequent performance deterioration. This indicates the importance of extracting information from students' response sequences to construct accurate student portraits.
- Without the Critic module, LACAT-C fails to correct errors in a timely manner, leading to performance deterioration. This highlights the critical role of timely error correction.

6 Conclusion

We introduce LACAT, a framework integrating large language models with computerized adaptive testing. It comprises the Summarizer for student profiling, the Reasoner for question selection and explanation, and the Critic for process optimization, enhancing adaptive testing in real educational settings. Compared to traditional rule-based and data-driven methods, LACAT can generate clearer, more transparent, and more reasonable questions. This integration of large language models with computerized adaptive testing demonstrates a promising avenue for enhancing educational assessment.

Limitations

Our experiment is based solely on a type of IRT (Item Response Theory) that provides unidimensional ability estimation in the CDM (Cognitive Diagnostic Model). While many other CDM models evaluate student ability using complex vectors (Wang et al., 2020; Gao et al., 2021; Zhang et al., 2024b), translating these vectors into text for large language models remains challenging. Additionally, as this paper is the first to explore CAT explainability, further research is needed to better define and enhance evaluation indicators.

Acknowledgement

This research was supported by grants from the National Science and Technology Major Project (No. 2022ZD0117103), National Natural Science Foundation of China (Grants No. 62337001, 62106246), the Key Technologies R&D Program of Anhui Province (No. 202423k09020039), and the University Synergy Innovation Program of Anhui Province (GXXT-2022-042).

References

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

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Haoyang Bi, Haiping Ma, Zhenya Huang, Yu Yin, Qi Liu, Enhong Chen, Yu Su, and Shijin Wang. 2020. Quality meets diversity: A model-agnostic framework for computerized adaptive testing. In *2020 IEEE International Conference on Data Mining (ICDM)*, pages 42–51. IEEE.

Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. 2024. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*.

Haw-Shiuan Chang, Hwai-Jung Hsu, Kuan-Ta Chen, et al. 2015. Modeling exercise relationships in e-learning: A unified approach. In *EDM*, pages 532–535.

Hua-Hua Chang and Zhiliang Ying. 1996. A global information approach to computerized adaptive testing. *Applied Psychological Measurement*, 20(3):213–229.

Hua-Hua Chang and Zhiliang Ying. 1999. A-stratified multistage computerized adaptive testing. *Applied psychological measurement*, 23(3):211–222.

Shu-Ying Chen, Robert D Ankenmann, and Judith A Spray. 2003. The relationship between item exposure and test overlap in computerized adaptive testing. *Journal of Educational Measurement*, 40(2):129–145.

Zhiyu Chen, Yujie Lu, and William Yang Wang. 2023a. Empowering psychotherapy with large language models: Cognitive distortion detection through diagnosis of thought prompting. *arXiv preprint arXiv:2310.07146*.

Zihang Chen, Mengxiao Zhu, Fei Wang, Shuanghong Shen, Zhenya Huang, and Qi Liu. 2023b. Long-term and short-term perception in knowledge tracing. In *CCF Conference on Big Data*, pages 1–15. Springer.

Ying Cheng. 2009. When cognitive diagnosis meets computerized adaptive testing: Cd-cat. *Psychometrika*, 74:619–632.

Susan E Embretson and Steven P Reise. 2013. *Item response theory*. Psychology Press.

Rosa Falotico and Piero Quatto. 2015. Fleiss' kappa statistic without paradoxes. *Quality & Quantity*, 49:463–470.

Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. 2018. Bilevel programming for hyperparameter optimization and meta-learning. In *International conference on machine learning*, pages 1568–1577. PMLR.

Weibo Gao, Qi Liu, Zhenya Huang, Yu Yin, Haoyang Bi, Mu-Chun Wang, Jianhui Ma, Shijin Wang, and Yu Su. 2021. Rcd: Relation map driven cognitive diagnosis for intelligent education systems. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 501–510.

Aritra Ghosh and Andrew Lan. 2021. Bobcat: Bilevel optimization-based computerized adaptive testing. *arXiv preprint arXiv:2108.07386*.

Liyang He, Zhenya Huang, Jiayu Liu, Enhong Chen, Fei Wang, Jing Sha, and Shijin Wang. 2024. Bitmask robust contrastive knowledge distillation for unsupervised semantic hashing. In *Proceedings of the ACM on Web Conference 2024*, pages 1395–1406.

Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. 1996. Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4:237–285.

Kelvin JL Koa, Yunshan Ma, Ritchie Ng, and Tat-Seng Chua. 2024. Learning to generate explainable stock predictions using self-reflective large language models. In *Proceedings of the ACM on Web Conference 2024*, pages 4304–4315.

Haoxuan Li, Jifan Yu, Yuanxin Ouyang, Zhuang Liu, Wenge Rong, Juanzi Li, and Zhang Xiong. 2024a. Explainable few-shot knowledge tracing. *arXiv preprint arXiv:2405.14391*.

Nian Li, Chen Gao, Mingyu Li, Yong Li, and Qingmin Liao. 2024b. Econagent: Large language model-empowered agents for simulating macroeconomic activities. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*.

Rui Li, Liyang He, Qi Liu, Yuze Zhao, Zheng Zhang, Zhenya Huang, Yu Su, and Shijin Wang. 2024c. Consider: Commonalities and specialties driven multilingual code retrieval framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 8679–8687.

Tao Li, Gang Li, Zhiwei Deng, Bryan Wang, and Yang Li. 2023. A zero-shot language agent for computer control with structured reflection. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11261–11274.

Xiaonan Li and Xipeng Qiu. 2023. Mot: Memory-of-thought enables chatgpt to self-improve. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6354–6374.

Qi Liu, Yan Zhuang, Haoyang Bi, Zhenya Huang, Weizhe Huang, Jiatong Li, Junhao Yu, Zirui Liu, Zirui Hu, Yuting Hong, et al. 2024a. Survey of computerized adaptive testing: A machine learning perspective. *arXiv preprint arXiv:2404.00712*.

Shuo Liu, Junhao Shen, Hong Qian, and Aimin Zhou. 2024b. Inductive cognitive diagnosis for fast student learning in web-based intelligent education systems. In *Proceedings of the ACM on Web Conference 2024*, pages 4260–4271.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2023. Chatgpt as a factual inconsistency evaluator for abstractive text summarization. *arXiv preprint arXiv:2303.15621*.

Weiyu Ma, Qirui Mi, Xue Yan, Yuqiao Wu, Runji Lin, Haifeng Zhang, and Jun Wang. 2023. Large language models play starcraft ii: Benchmarks and a chain of summarization approach. *arXiv preprint arXiv:2312.11865*.

Liangming Pan, Alon Albalak, Xinyi Wang, and William Wang. 2023. Logic-lm: Empowering large language models with symbolic solvers for faithful logical reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3806–3824.

Wentao Shi, Xiangnan He, Yang Zhang, Chongming Gao, Xinyue Li, Jizhi Zhang, Qifan Wang, and Fuli Feng. 2024. Enhancing long-term recommendation with bi-level learnable large language model planning. *arXiv preprint arXiv:2403.00843*.

Wangtao Sun, Xuanqing Yu, Shizhu He, Jun Zhao, and Kang Liu. 2023. Expnote: Black-box large language models are better task solvers with experience notebook. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15470–15481.

Wim J Van der Linden and Cees AW Glas. 2000. *Computerized adaptive testing: Theory and practice*. Springer.

Hado Van Hasselt, Arthur Guez, and David Silver. 2016. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30.

Howard Wainer, Neil J Dorans, Ronald Flaugher, Bert F Green, and Robert J Mislevy. 2000. *Computerized adaptive testing: A primer*. Routledge.

Fei Wang, Qi Liu, Enhong Chen, Zhenya Huang, Yuying Chen, Yu Yin, Zai Huang, and Shijin Wang. 2020. Neural cognitive diagnosis for intelligent education systems. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 6153–6161.

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

Hangyu Wang, Ting Long, Liang Yin, Weinan Zhang, Wei Xia, Qichen Hong, Dingyin Xia, Ruiming Tang, and Yong Yu. 2023b. Gmocat: A graph-enhanced multi-objective method for computerized adaptive testing. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2279–2289.

Yu Wang, Zhiwei Liu, Jianguo Zhang, Weiran Yao, Shelby Heinecke, and Philip S Yu. 2023c. Drdt: Dynamic reflection with divergent thinking for llm-based sequential recommendation. *arXiv preprint arXiv:2312.11336*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Songlin Xu, Xinyu Zhang, and Lianhui Qin. 2024. Eduagent: Generative student agents in learning. *arXiv preprint arXiv:2404.07963*.

Kailai Yang, Shaoxiong Ji, Tianlin Zhang, Qianqian Xie, Ziyan Kuang, and Sophia Ananiadou. 2023. Towards interpretable mental health analysis with large language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Qisen Yang, Zekun Wang, Honghui Chen, Shenzhi Wang, Yifan Pu, Xin Gao, Wenhao Huang, Shiji Song, and Gao Huang. 2024. Llm agents for psychology: A study on gamified assessments. *arXiv preprint arXiv:2402.12326*.

Jifan Yu, Mengying Lu, Qingyang Zhong, Zijun Yao, Shangqing Tu, Zhengshan Liao, Xiaoya Li, Manli Li, Lei Hou, Hai-Tao Zheng, et al. 2023. Moocradar: A fine-grained and multi-aspect knowledge repository for improving cognitive student modeling in moocs. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2924–2934.

Jinghan Zhang, Xiting Wang, Weijieying Ren, Lu Jiang, Dongjie Wang, and Kunpeng Liu. 2024a. Ratt: Athought structure for coherent and correct llmreasoning. *arXiv preprint arXiv:2406.02746*.

Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Jialin Pan, and Lidong Bing. 2023a. Sentiment analysis in the era of large language models: A reality check. *arXiv preprint arXiv:2305.15005*.

Yuting Zhang, Ying Sun, Fuzhen Zhuang, Yongchun Zhu, Zhulin An, and Yongjun Xu. 2023b. Triple dual learning for opinion-based explainable recommendation. *ACM Transactions on Information Systems*, 42(3):1–27.

Zheng Zhang, Qi Liu, Zirui Hu, Yi Zhan, Zhenya Huang, Weibo Gao, and Qingyang Mao. 2024b. Enhancing fairness in meta-learned user modeling via adaptive sampling. In *Proceedings of the ACM on Web Conference 2024*, pages 3241–3252.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. 2024a. Expel: Llm agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 17, pages 19632–19642.

Guanhao Zhao, Zhenya Huang, Yan Zhuang, Jiayu Liu, Qi Liu, Zhiding Liu, Jinze Wu, and Enhong Chen. 2023. Simulating student interactions with two-stage imitation learning for intelligent educational systems. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 3423–3432.

Hongke Zhao, Xinpeng Wu, Chuang Zhao, Lei Zhang, Haiping Ma, and Fan Cheng. 2021. Coea: A cooperative–competitive evolutionary algorithm for bidirectional recommendations. *IEEE Transactions on Evolutionary Computation*, 26(1):28–42.

Yuyue Zhao, Jiancan Wu, Xiang Wang, Wei Tang, Dingxian Wang, and Maarten de Rijke. 2024b. Let me do it for you: Towards llm empowered recommendation via tool learning. *arXiv preprint arXiv:2405.15114*.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2024. Memorybank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19724–19731.

Yujia Zhou, Qiannan Zhu, Jiajie Jin, and Zhicheng Dou. 2024. Cognitive personalized search integrating large language models with an efficient memory mechanism. In *Proceedings of the ACM on Web Conference 2024*, pages 1464–1473.

Xizhou Zhu, Yuntao Chen, Hao Tian, Chenxin Tao, Weijie Su, Chenyu Yang, Gao Huang, Bin Li, Lewei Lu, Xiaogang Wang, et al. 2023. Ghost in the minecraft: Generally capable agents for open-world enviroments via large language models with text-based knowledge and memory. *arXiv preprint arXiv:2305.17144*.

Yan Zhuang, Qi Liu, Zhenya Huang, Zhi Li, Shuanghong Shen, and Haiping Ma. 2022. Fully adaptive framework: Neural computerized adaptive testing for online education. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 4734–4742.

Yan Zhuang, Qi Liu, GuanHao Zhao, Zhenya Huang, Weizhe Huang, Zachary Pardos, Enhong Chen, Jinze Wu, and Xin Li. 2024. A bounded ability estimation for computerized adaptive testing. *Advances in Neural Information Processing Systems*, 36.

A Baseline Methods

- **MFI (Chang et al., 2015):** It selects the question based on the maximum Fisher information.
- **KLI (Chang and Ying, 1996):** It selects the question with the maximum moving average of Kullback-Leibler information.
- **MAAT (Bi et al., 2020):** It proposes an active learning-based method that measures the informativeness of questions by calculating the Expected Model Change (EMC) caused by each question.
- **BECAT (Zhuang et al., 2024):** It proposes an effective expected gradient-based selection algorithm that minimizes the estimation error term in CAT systems.
- **BOBCAT (Ghosh and Lan, 2021):** It is the first bilevel optimization (Franceschi et al., 2018) framework for CAT that adopts an approximate gradient estimation method to learn a data-driven selection algorithm.
- **NCAT (Zhuang et al., 2022):** It's a reinforcement learning-based method that designs an attention-based DQN (Van Hasselt et al., 2016).

B Additional Results

Cost and Efficiency Analysis Experiment We conducted comprehensive cost and efficiency experiments on the MoocRadar and Exam datasets using four large language models: GPT-3.5-turbo-16k, DeepSeek, GPT-4, and GPT-4o-mini. Table 8 presents the average cost and time for 100 students to complete a 10-step Computerized Adaptive Test (CAT) exam. The results demonstrate that the cost is relatively affordable; even the most expensive model (GPT-3.5-turbo-16k) does not exceed $0.2 per test round, while DeepSeek and GPT-4o-mini remain under $0.01 per test.

However, efficiency posed a challenge, with the slowest model, DeepSeek, taking an average of approximately eight seconds to select a problem. To address this issue, we devised a strategy to reduce time consumption: as students progress through the exercises, problem selection occurs concurrently based on their correct and incorrect responses. This approach allows for immediate recommendation

Table 8: Cost and time per CAT test with 10 steps for 100 students on MoocRadar and Exam datasets. The best results are given in bold.

Dataset	**MoocRadar**		**Exam**	
Metric	Cost (USD)	Time (s)	Cost (USD)	Time (s)
GPT-3.5-turbo-16k	0.1334	65.62	0.0923	68.13
DeepSeek	**0.0076**	79.96	**0.0056**	80.58
GPT-4	0.1873	49.28	0.1617	55.77
GPT-4o-mini	0.0083	**34.51**	0.0051	**36.88**

of subsequent exercises upon result submission, significantly reducing overall time expenditure.

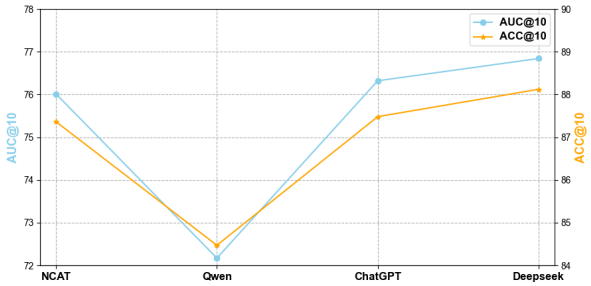
While the current cost structure is reasonably economical, our future work will explore methods to further reduce computational expenses. This two-pronged approach—optimizing both time efficiency and cost-effectiveness—aims to enhance the viability and scalability of large language model applications in adaptive testing scenarios.

LLM impact on student score prediction performance Following the experimental setup described in Section 5.2, we replace the base LLM with two alternative LLMs: Qwen-1.5-7B (Bai et al., 2023), developed by Alibaba, and Deepseek-v2 (Bi et al., 2024), developed by Magic Square, are both open-source models. These models perform well in certain aspects, with Deepseek-v2 even surpassing GPT-3.5 in performance. As As revealed in Figure 3, the following observations can be made:

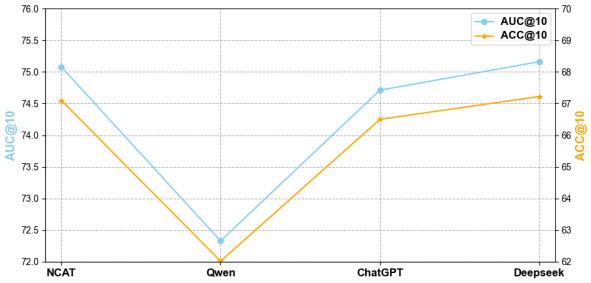
- Firstly, Qwen-1.5-7B performs poorly in predicting student performance on the MoocRadar and Exam datasets. Despite using the same prompt templates as ChatGPT, its outputs often fail to meet the specified performance metrics, even after multiple attempts. This issue may be attributed to the model's handling of extensive context, leading to a random selection of exercises by default.

- In contrast, Deepseek-v2 demonstrates superior performance, likely due to its advanced reasoning capabilities. Specifically, on the MoocRadar dataset, Deepseek-v2 outperforms NCAT, although ChatGPT achieves the highest results. Furthermore, on the Exam dataset, where ChatGPT performs poorly, Deepseek-v2 still surpasses the best baseline performance.

These findings suggest that the framework's performance is closely related to the reasoning abilities of the selected model. Therefore, choosing an appropriate LLM as the core component of the framework is crucial for achieving optimal performance.



(a) MoocRadar.



(b) Exam.

Figure 3: Student Score Prediction performance across different LLMs.

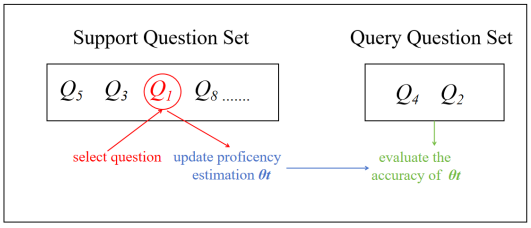


Figure 4: The calculation process of ACC(θ_t).

C The calculation process of ACC(θ_t)

As shown in Figure 4, at step t, the selection algorithm selects questions from the support set and uses the CDM to estimate the student's ability θ_t. It then estimates the student's performance on the query set based on the estimated θ_t and compares this with the actual performance to calculate accuracy (Wang et al., 2023b).

D Detailed Evaluation Method

The primary objective of Computerized Adaptive Testing (CAT) is to accurately assess a student's true ability using the minimum number of test items. However, true ability is an abstract concept that cannot be directly measured. To evaluate the effectiveness of a CAT system's capability es-

timation, we employ an indirect approach. This method utilizes the scores (correct or incorrect) from the questions answered in the retained response data. By analyzing these scores, we can infer the strengths and weaknesses of the CAT system's ability to estimate student capability. This indirect evaluation serves as a proxy for assessing the overall performance and accuracy of the CAT system in achieving its goal of efficient and precise ability assessment.

E Human Evaluation Criteria

Evaluators will be given explanations generated by Summarizer and Reasoner. Evaluators will need to score and annotate the generated explanations from the following aspects:

Fluency Fluency evaluates the coherence and readability of the explanation. Evaluators should assess if the generated explanation is well-structured, easy to read, and free of grammatical or syntax errors.

- 0: Incoherent, difficult to read, and contains numerous errors
- 1: Mostly fluent, easy to read, with few minor errors
- 2: Completely fluent, coherent, and error-free

Relevance Relevance measures how well the generated topic selection reasons or user profiles align with the student's knowledge state and the appropriateness of the chosen exercises. Evaluators should assess whether the explanations are pertinent, coherent, and adequately reflect the student's learning needs. Main criteria to check (sorted by criticality):

- 0: Irrelevant information or completely misaligned with the student's knowledge state
- 1: Somewhat relevant information with several misalignments or off-topic elements
- 2: Mostly relevant information with minor misalignments, generally aligned with the student's knowledge state and learning needs

Comprehensiveness Comprehensiveness measures how thoroughly the generated explanations cover the student's knowledge state and the appropriateness of the chosen exercises. Evaluators should assess whether the explanations fully address all relevant aspects of the student's learning needs. Main criteria to check (sorted by criticality):

- 0: Incomplete information, missing many critical aspects of the student's knowledge state
- 1: Partially complete information, covering some but not all critical aspects
- 2: Comprehensive information, thoroughly covering the student's knowledge state and learning needs

Overall Score Overall Score evaluates the quality of the generated explanations by assessing fluency, relevance, and comprehensiveness. Evaluators should consider how clearly the explanations are articulated, how well they align with the student's knowledge state, and how thoroughly they address their learning needs. Main criteria to check (sorted by criticality):

- 0: The explanations are poorly articulated, largely irrelevant, and do not adequately cover the student's knowledge state.
- 1: The explanations are somewhat clear, moderately relevant, and cover some aspects of the student's knowledge state.
- 2: The explanations are well-articulated, highly relevant, and provide a thorough understanding of the student's knowledge state and learning needs.

F Prompt Templates

In this section, we present some prompt templates used by LACAT.

Prompt Template (Summarizer, Long-term Profile).

You are an expert in educational assessment. You are analyzing the performance of a student based on their correct and incorrect answers to a set of questions and ability estimation generated by IRT. Your goal is to provide a comprehensive analysis of the student's strengths and weaknesses to help them improve.

Correct: {right response records}

Incorrect: {wrong response records}

Ability Estimation: {ability estimation}

Please carefully analyze the student's performance step by step. Your reasoning process should be thorough and detailed. First, summarize the students' strengths based on the correct response records, focusing on what they do well and the skills or knowledge areas they have mastered. Then, summarize the student's weaknesses based on the Incorrect response records, highlighting areas where the student struggles or needs improvement. Ensure your summary is concise and clear and does not simply repeat existing information.

To help you with this analysis, here are some key points to consider:

Patterns in Correct Answers:
– Look for recurring themes or skills in the correct answers.
– Identify any specific questions or question types where the student consistently performs well.

Patterns in Incorrect Answers:
– Analyze the incorrect answers to find common mistakes or misunderstandings.
– Determine if there are specific questions or question types where the student frequently struggles.

Comparison and Contrast:
– Compare the correct and incorrect answers to understand the student's overall performance.
– Highlight any discrepancies between what the student knows well and where they need improvement.

Organize your output by strictly following the format below:

Strength: <summary of strengths based on correct answers>

Weakness: <summary of weaknesses based on incorrect answers>

Prompt Template (Summarizer, Short-term Profile).

You are an expert in educational assessment. You are analyzing the recent performance of a student based on their profile and their most recent response records in a computer adaptive test. Your goal is to understand the student's needs and provide a comprehensive analysis of their performance on the most recent question.

Profile: {student profile}

Last record: {last response record}

Please understand the student's needs based on their recent performance and analyze their performance on the most recent question. Your analysis should be concise and clear, focusing on identifying strengths and weaknesses demonstrated in the last answer.

To help you with this analysis, here are some key points to consider:

Understanding the Student's Profile:

Identify the student's strengths and areas for improvement based on their profile.
Consider their learning style, preferences, and any specific goals or challenges mentioned in the profile.
Analyzing the Last response record:

Review the student's performance on the most recent question to determine their current understanding.
Note any patterns or specific errors that may indicate areas needing further practice.
Assess the approach the student took to solve the question and identify any misconceptions.
Providing Feedback:

Offer specific, actionable feedback based on the student's performance.
Suggest strategies or resources that can help the student address their weaknesses.
Organize your output by strictly following the format below:

Thought: <the analysis of the student's performance on the most recent question>

Prompt Template (Critic).

You are an expert in educational assessment. Your role is to inspect the recommendations given to a student to ensure they are appropriate and free from basic errors. Below are some typical error types and corresponding examples. If you encounter errors that do not fall into common categories, please infer the type of error and provide hints accordingly.

Typical Errors:

The recommended exercise is too difficult, exceeding the student's ability.

Example: Recommending advanced calculus problems to a student struggling with basic algebra.
The recommended exercise is too easy, below the student's actual ability.

Example: Recommending basic addition problems to a student ability in algebra.
The recommended exercise is knowledge−saturated, covering material the student has already mastered.

Example: Recommending the same set of geometry problems repeatedly to a student who has already demonstrated ability in that area.
Here is the student recommendation that you need to check:

student_profile: {profile}

last_recommended_exercise: {exercise}

Please check if the recommendation has made any of the above errors and provide your judgment. Your judgment should be clear and concise, pointing out any errors and the type of error identified.

To help you with this inspection, here are some key points to consider:

Understanding the Student's Profile:

Review the student's mastery of various knowledge points.
Identify the students strengths and weaknesses based on recent performance and assessments.
Evaluating the Recommended Exercise:

Assess whether the exercise is appropriately challenging for the student.
Determine if the exercise introduces new concepts or reinforces recently taught material.
Ensure the exercise is not redundant with previously mastered content.
Making a Judgment:

Clearly state whether the exercise is appropriate for the student.
If an error is found, specify the type of error and provide a concise explanation.
Offer a brief suggestion or adjustment if necessary.
Organize your output by strictly following the format below:

Hint: <some short reminders you would like to give>

Prompt Template (Reasoner).

By thinking step by step about the student's strengths and weaknesses, select the most suitable question from the candidate questions, considering difficulty, discrimination, and knowledge coverage. Use the principles of computerized adaptive testing so that after the student answers this question, their ability can be measured more accurately. In your reasoning process, you need to think carefully.

Long−term profile: {strength}

candidate questions: {candidate_questions}

Short−term profile: {thought}

hint: {hint}

To begin the selection process, carefully analyze the student's long−term and short−term performance:

Analyze Long−term Performance:

Identify the student's strengths and weaknesses over an extended period to understand which knowledge points they have mastered and which require improvement.
Review Short−term Performance:

Examine recent response records to determine the student's current understanding and ability, reflecting their immediate learning needs.
Based on this analysis, consider the following characteristics when evaluating candidate questions:

Difficulty:

Ensure the question is appropriately challenging, providing a balance between too easy and too difficult.
Discrimination:

Choose questions that effectively differentiate between varying levels of student ability.
Knowledge Coverage:

Select questions that cover key knowledge areas necessary for the student to reinforce or master.
Incorporate the principles of computerized adaptive testing (CAT):

Dynamic Adjustment:

Adjust the difficulty of questions based on the student's real−time performance to assess their ability accurately.
Precise Measurement:

Ensure each question helps in narrowing the estimation of the student's ability, increasing the accuracy of the assessment.
Emphasize the reasoning behind your selection, providing a clear and logical explanation:

Explain the Selection:
Clearly state why you selected the particular question from the candidate list.
Detail how this question addresses the student's current strengths and weaknesses.
Explain how the question will help in detecting and measuring the student's cognitive state and ability.
Now start selecting and organizing your output by strictly following the output format below:

Reason: Explain why you select the question, how to detect cognitive state with the question

Selected question with index: the selected question here with the index, following the output format like <the index>, only one index here