

GenAL: Generative Agent for Adaptive Learning

Rui Lv¹, Qi Liu^{1,2*}, Weibo Gao¹, Haotian Zhang¹, Junyu Lu², Linbo Zhu²

¹State Key Laboratory of Cognitive Intelligence, University of Science and Technology of China

²Institute of Artificial Intelligence, Hefei Comprehensive National Science Center

lvru2018@mail.ustc.edu.cn, qiliuql@ustc.edu.cn,

{weibogao,sosweetzhang,lujunyu}@mail.ustc.edu.cn, lbzhu@iai.ustc.edu.cn

Abstract

Adaptive learning, also known as adaptive teaching, relies on learning path recommendations that sequentially suggest personalized learning items (such as lectures and exercises) to meet the unique needs of each learner. Despite the extensive research in this field, previous approaches have primarily modeled the interaction sequences between learners and items using simple indexing, leading to three issues: (1) The utilization of information from both learners and items is not sufficient. For instance, these models are unable to leverage the semantic information contained within the textual content of the items. (2) Models need to be retrained on different datasets separately, which makes it difficult to adapt to the continuously expanding item pool in online educational scenarios. (3) The existing recommendation paradigm based on trained reinforcement learning frameworks, suffers from unstable recommendation performance in sparse learning logs. To address these challenges, we propose a generalized **Generative Agent for Adaptive Learning (GenAL)**, which integrates educational tools with LLMs' semantic understanding to enable effective and generalizable learning path recommendations across diverse data distributions. Specifically, our framework consists of two components: the Global Thinking Agent, which updates the learner profile and reflects on recommendation outcomes based on the learner's historical learning records. The other is the Local Teaching Agent, which recommends items using educational prior knowledge. Leveraging the LLM's robust semantic understanding, our framework does not rely on item indexing but instead extracts relevant information from the textual content. We evaluated our approach on three real-world datasets, and the experimental results demonstrate that our GenAL not only consistently outperforms all baselines but also exhibits strong generalization ability.

Introduction

Learning serves as a cornerstone for human advancement, allowing individuals to acquire knowledge and skills. Unlike traditional methods of education (e.g., classroom courses) that deliver uniform material to all learners, adaptive learning focuses on providing personalized learning items (e.g., exercises) and pathways tailored to each

learner's unique needs (Liu et al. 2019), which is generalized researched as the Learning Path Recommendation (LPR) tasks. Early works employ traditional recommendation algorithms or deep learning-based methods to suggest similar learning paths to comparable users (Elshani and Nuçi 2021; Nabizadeh et al. 2020a), which is usually limited under complex and dynamic learning processes. Recent attempts model the LPR task as a dynamic sequential decision-making problem and learn optimized recommendation strategies using Reinforcement Learning (RL) techniques (Liu et al. 2019; Li et al. 2023b, 2024), which demonstrated remarkable effectiveness. However, the current process of LPR has three aspects of limitations: (1) **Coarse-grained information**: The utilization of information from both learners and items is not sufficient. For instance, these models are unable to leverage the semantic information contained within the textual content of the items. The text of the items contains rich, fine-grained semantic information that implicitly reveals various attributes of the items (e.g., the knowledge concepts covered, difficulty level, and cognitive layers being assessed), which are hard to utilize in existing methods sufficiently. For example, although both exercises assess the knowledge concept of “linear functions”, exercise A: “Write the equation of the linear function” focuses only on the basic concepts, while exercise B: “Given the linear function $y = kx + b$ with its graph passing through points $A(0, 3)$ and $B(-4, 0)$. Determine the distance from the $(0,0)$ to the line AB .” not only examines the concept of “linear functions” but also assesses the deeper knowledge of “distance from a point to a line”. While some existing works utilize knowledge graphs (Liu et al. 2019; Li et al. 2023b; Wu et al. 2024) to assist in recommending learning paths, they are limited by the size of the knowledge graph and the granularity of the knowledge concepts, resulting in less accurate recommendations. (2) **Limited generalization ability**: Existing methods rely on training within a specific scenario (e.g., a standard dataset), resulting in very limited generalization capabilities. For example, when new scenarios are introduced or a dataset shift occurs, these methods require retraining, which limits the generalization of these methods. (3) **Performance instability**: Several works have adopted advanced Reinforcement Learning (RL) methods, which rely on rich interaction sequence for training (Liu et al. 2019; Li et al. 2023b; Kubotani, Fukuhara, and Mor-

*Corresponding author

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

ishima 2021). However, in real-world online education scenarios, interaction data is often sparse, leading to unstable recommendation results with these methods. For example, our experimental results show that while existing methods perform well on datasets with over 90 interact logs per user, they perform poorly on sparse datasets with only 13 interactions per user, even failing to improve the learners' cognitive state at all.

However, an experienced teacher can easily address these issues. With their extensive knowledge and teaching experience, teachers can assess a learner's current knowledge level from a small number of learners learning content and provide suitable learning paths during interactions with learners. Recently, Large Language Models (LLMs), pre-trained on vast amounts of text data, have demonstrated rich knowledge and powerful planning capabilities (Shi et al. 2024; Ouyang et al. 2022; Touvron et al. 2023). Inspired by this, we propose a novel **Generative Agent for Adaptive Learning (GenAL)** framework based on LLMs. Specifically, we leverage the extensive knowledge of LLMs to evaluate learners' learning states from both global and local perspectives and adaptively recommend personalized learning paths accordingly. The framework consists of two main Agents:

- **Global Thinking Agent (GTA):** This module aims to integrate learners' response records and summarize their overall cognitive habits (e.g., assessing the learner's learning abilities and the current stage of learning). It comprises three sub-modules: (1) **Log Memory:** Stores learner records. (2) **Tools:** A knowledge tracing model that provides a reference for learning states. (3) **Reflector:** Generates a learner learning profile by combining current response records with tool outputs and reflects on the effectiveness of recommended exercises.
- **Local Teaching Agent (LTA):** This module utilizes the powerful semantic understanding and analysis capabilities of large language models to analyze items' content and provide learning recommendations. It consists of a large language model-based agent with two modules: (1) **Teaching Tools:** Provide educational prior information, such as knowledge concept graphs, to supplement the knowledge of large language models. (2) **Recommender:** Given the exercise for the next learning steps and provide reasons based on the learner profile from the Global Thinking Agent and the current response.

The GTA and LTA collaborate and dynamically recommend learning paths as interactions with learners progress, achieving an effective, general, and stable personalized path recommendation framework based on LLMs. To the best of our knowledge, we are the first to introduce the large language model's knowledge into the learning path recommendation task, which demonstrates the potential of LLM to empower the educational applied tasks.

Overall, our contributions are as follows:

- We first introduce a new large-model-based personalized Learning Path Recommendation (LPR) framework GenAL, which can enhance the learning effect and empower the generalization ability of LPR systems by leveraging fine-grained semantic analysis of learning content.

- We developed a collaborative mechanism between the Global Thinking Agent and Local Teaching Agent to ensure stable and effective personalized learning path recommendations in sparse data scenarios, which is hard to solve in existing methods.
- Experimental results on three real-world datasets demonstrate the effectiveness of our proposed GenAL.

Related Work

Adaptive Learning

In online education, Learning Path Recommendation (LPR) stands as a crucial undertaking, which refers to planning and designing a structured learning path for learners, enabling them to systematically and orderly acquire knowledge and skills (Liu et al. 2019; Li et al. 2023b). Researchers have proposed various methods for the LPR task. The existing LPR methods can be summarized into two categories according to the approach of path generation (Chen et al. 2023; Nabizadeh, Mário Jorge, and Paulo Leal 2017): (1) Complete generation, a complete path of a specified length is generated and provided to learners at once. (2) Step-by-step generation, a dynamic path of a varying length is generated in real-time by considering the feedback from learners' interactions at each step with the next item recommended (Liu et al. 2019; Li et al. 2023b). The main disadvantage of complete generation is ignoring users' performance and their cognitive changes during the learning process, which may lead to users wasting time on inappropriate or unmanageable paths (Nabizadeh et al. 2020b). Due to its ability to better consider the dynamic interaction between learners and items, step-by-step based methods are rapidly gaining prominence. In this branch, the early works use traditional recommendation algorithms or deep learning-based methods. Recently, since learning path recommendation can be regarded as a sequential decision-making problem, some work adopt advanced reinforcement learning methods, which demonstrate better techniques. However, these approaches are limited by training the sequences of learner behaviors, achieving only a coarse level of personalization, and suffering from limited generalization ability and instability. Our GenAL, which is based on text content and utilizes the well pre-trained LLMs, provides a effective solution for this problem.

LLM-Empowered Agents and Their Applications in Education

LLM-empowered generative agents demonstrate remarkable abilities in perceiving their environment, making decisions, and taking actions, leading to a surge of research in this area (Wang et al. 2024; Li et al. 2023a). In education, these agents have introduced new possibilities. For example, (Wu et al. 2023) leveraged chat-optimized LLMs as potent agents, orchestrating multi-agent dialogues to collaboratively tackle complex queries, highlighting the potential of agents in addressing general questions. In learner simulation tasks, EduAgent (Xu, Zhang, and Qin 2024) employs LLM-based agents to replicate learners engaging with PowerPoint presentations and videos, assessing learner performance by

predicting quiz results. In educational recommendations, (Li et al. 2024) utilized factual knowledge from LLMs to develop SKarREC, a concept recommendation model that better suggests the next concept for learners. It is important to note that this study focuses on knowledge concept recommendation, which differs from our task of learning exercise path recommendation.

Problem Definition

We focus on step-by-step recommendations for session-based learning paths based on real-time interactions. A learner’s process typically involves two types of items: learning items (e.g., knowledge concepts or skills) and exercise items (e.g., questions). Without loss of generality, we denote the learning item set as $\mathcal{CI} = \{c_1, c_2, \dots, c_k\}$ and the exercises item set as $\mathcal{EI} = \{e_1, e_2, \dots\}$. Instead of only using item index in previous work, the $c_i \in \mathcal{CI}$ and $e_i \in \mathcal{EI}$ consist of item index and item text content. The learner’s goals are denoted as $\mathcal{G} = \{g_1, g_2, \dots, g_m\}$, where $g_i \in \mathcal{CI}$. The learning process is as follows: before starting, educational tools test a learner on their goals to obtain an initial score E_s . Simultaneously, based on Large Language Models (LLMs), the “teacher” assesses the learner’s overall learning ability and preferences from their response records to build a learner profile, denoted as L_t , where t means time steps.

Learning is divided into several stages. In each stage, the learner studies a specific learning item c_i , and at each step, they are presented with a practice item e_i^j to comprehend c_i . Feedback $answer_i^j \in \{0, 1\}$ is provided after practice, where 1 indicates mastery of the item and 0 indicates otherwise. All item-feedback pairs construct the historical learning record, denoted as $\mathcal{H} = (e, answer)$. After time t , $(e_t, answer_t)$ is added to the historical record, i.e., $\mathcal{H}_t = \mathcal{H}_{t-1} \cup (e_t, answer_t)$. Finally, a learning path is generated step-by-step as $\mathcal{P} = \{e_1, e_2, \dots, e_p\}$, where $e_i \in \mathcal{EI}$, aligning with previous works. After completing the entire learning path, a final test using educational tools (e.g., knowledge tracing models) is taken on the learning goals to obtain a final score E_e , allowing us to calculate learning effectiveness E_p :

$$E_p = \frac{E_e - E_s}{E_{sup} - E_s}, \quad (1)$$

where E_{sup} is the full score of the examination, equal to the number of learning goals. Our goal is to maximize E_p by providing an effective learning path.

Framework

As shown in Figure 1, our framework consists of two main components: the **Global Thinking Agent (GTA)** and the **Local Teaching Agent (LTA)**. Both intelligences are built based on LLM, aiming to utilize LLM’s powerful semantic understanding to mine learners’ learning and interaction content from a textual perspective, which is hard to do with existing methods. In particular, we have the two intelligences collaborate for comprehensive and effective path recommendations from a global and local perspective, respectively. In this section, we provide a detailed description of our design.

Global Thinking Agent (GTA)

The Global Thinking Agent (GTA) stores learners’ learning records and summarizes their learning abilities and preferences, providing prior knowledge for the teaching process to recommend more efficient learning paths in limited initial information, and addressing the shortcomings of existing methods. Specifically, the GTA consists of three components: Log Memory, Educational Tools, and Reflector. First, the learner’s history response records, including exercise text and knowledge concept content, are stored in the Log Memory. Then, educational tools such as cognitive diagnosis models (Wang et al. 2023; Yao et al. 2023, 2024; Zhang et al. 2024) or knowledge tracing models (Piech et al. 2015; Liu et al. 2021; Yang et al. 2024; Yu et al. 2024b,a) can assess the learner’s mastery of current knowledge based on the sequence of these records, effectively quantifying the learner’s learning abilities. The Reflector, built on a large language model (LLM), summarizes the learner’s profile, including abilities (e.g., proficiency in a specific knowledge concept) and learning preferences (e.g., the learner performs better on real-world context questions or abstract questions) based on the response records and tool outputs, storing this information in the corresponding memory. What’s more, the reflector will rethink the efficiency of the current learning path, which is similar to the “backward update” in the neural network models training process. Next, we introduce the details of each component.

Log Memory. The Log is designed to store the learning history, including \mathcal{H} , and the feedback from the Local Teaching Agent, i.e., the reason for the recommended question and the prediction of the learner’s answer. Specifically, the data in time(step) t of the Log memory \mathcal{M}_t^G can be presented as follow:

$$\mathcal{M}_t^G \leftarrow \mathcal{H}_t \cup (F_{t+1}, A_{t+1}), \quad (2)$$

where the (F_{t+1}, A_{t+1}) note as one of the outputs of the LTA that given the current recommended question’s explanation and the prediction of the answer. The information in \mathcal{M}_t^G is used to generate the profile of the learner in the following Reflector module.

Educational Tools. Accurate learner profiles are crucial for any Adaptive Learning (AL) system. Unlike commodity recommendation profiles, AL profiles focus on tracking learners’ knowledge proficiency, which is not directly observable. To address this, prior works have developed Knowledge tracing (KT), a popular approach, using machine learning to estimate learners’ proficiency by predicting the correctness of their next question. These educational tools provide valuable references for constructing learner profiles. Specifically, our framework integrates pre-trained models of Deep Knowledge Tracing, referred to as DKT, to predict the learner’s mastery of learning goal items. For each knowledge concept c_i , the learner’s mastery of this knowledge point can be represented as follows:

$$E_t^{c_i} = \text{AVG}(\text{DKT}(\mathcal{E}_i, \mathcal{H}_t)). \quad (3)$$

where AVG denotes the average of the predicted values for all practice items in \mathcal{E}_i .

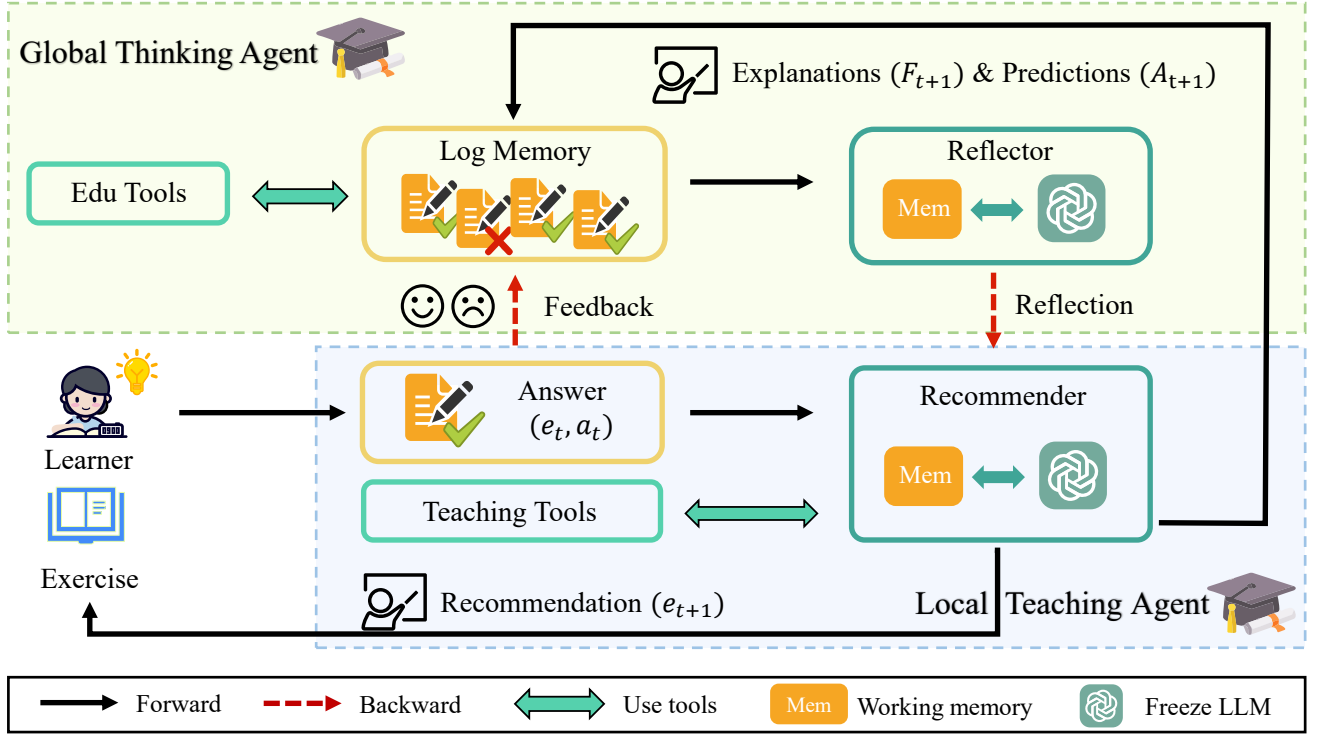


Figure 1: An overview of our GenAL. Our framework consists of two Agents: (1) **Global Thinking Agent**, which aims to update profiles and recommended strategies from learners’ history records. (2) **Local Teaching Agent**, which is used to select the most suitable exercise (e_{t+1}) based on the specific answer records (e_t, a_t) and prior knowledge with the Teaching tools. Additionally, we asked for explanations (F_{t+1}) and predictions of the learner’s response (A_{t+1}).

Reflector. Based on the learner’s response records from the Log Memory and knowledge state analysis provided by the Educational Tools, we designed a Reflector using LLM. The Reflector leverages the rich knowledge embedded in the LLM, acting as an “experienced teacher” to comprehensively evaluate the learner’s learning ability and reflect on and adjust the currently recommended learning path. Specifically, the Reflector comprises two subtasks:

- **Generate** the learner’s learning profile based on the learner’s response records, including learning ability and learning preferences. At time t , the learner’s profile L_t can be note as:

$$L_t \sim \text{LLM}(\mathbf{P}_L, \mathcal{H}_t, E_t^{c_i}), \quad (4)$$

where l is the index number of the current learner, \mathcal{D}_t is the historical questions’ difficulty before t time steps, and \mathbf{P}_L is the prompt to generate the summaries. The memory in Reflector can be updated by:

$$\mathcal{M}_t^R \leftarrow L_t. \quad (5)$$

- **Reflect** on the efficiency and rationality of past recommendations by comparing the learner’s actual responses with the predicted responses, considering the reasons provided during the recommendation of items. This step is similar to the gradient backpropagation process in neural network training, which is used to update the model parameters. At time t , the reflection R_t generation process

can be formulated as:

$$R_t \sim \text{LLM}(\mathbf{P}_R, \mathcal{M}_t^R, \mathcal{M}_t^G, E_t^{c_i}), \quad (6)$$

where \mathbf{P}_R notes the prompt for LLM to generate the reflection. Then, R_t is passed to the Local Teaching Agent to update the recommendation strategy.

Local Teaching Agent (LTA)

While the Global Thinking Agent (GTA) provides recommendations from a macro perspective on the learner’s overall learning ability, the Local Teaching Agent (LTA) leverages the powerful semantic understanding and analysis capabilities of large language models to analyze learners’ specific responses and offer subsequent learning suggestions. Specifically, the LTA consists of two modules: Teaching Tools and Recommender. We introduce details as follows.

Teaching Tools. LTA aims to retrieve the most suitable next question e_{t+1} from the question bank, tailored to the learner’s target knowledge point, by leveraging all existing information from the GTA and LTA. In this task, we need to employ educational prior knowledge to narrow the search space. What’s more, although language models have been shown to have a strong knowledge, LLM also has shortcomings in specific domains and often generates hallucinatory (Zhang et al. 2023). Therefore, it is necessary to introduce

external knowledge to guide and assist LLM in generating decisions (Lewis et al. 2020; Chen et al. 2024). Based on the above analysis, we design the Teaching tools to provide domain knowledge for help. Specifically, we utilize a hierarchical knowledge graph \mathcal{G} to limit the retrieval scope to the current knowledge concept and its immediate predecessor nodes. The exercises associated with this knowledge form the candidate exercises set $\mathcal{S}_t \subseteq \mathcal{EI}$ for recommendation.

Recommender. Recommender is a decision-making module designed to recommend exercises for the learner to more effectively master a specific knowledge concept. Specifically, the Planner is implemented as a frozen LLM instance equipped with a memory repository \mathcal{M}_t^D , which stores recommendation strategies R_t , from the GTA, and relevant external knowledge before time t :

$$\mathcal{M}_t^D \leftarrow (R_t, \mathcal{S}_t). \quad (7)$$

Now, we can utilize LLM to predict the next exercise item $e_{t+1} = (\text{index}, \text{content})$:

$$e_{t+1} \sim \text{LLM}(\mathbf{P}_E, \mathcal{M}_t^D), \quad (8)$$

where \mathbf{P}_E notes the prompt for LLM. While recommending the next question, we expect the “teacher” to provide the rationale behind the recommendation (note as F_{t+1}) and to predict the learner’s response (note as A_{t+1}) to the recommended question based on their current status. This allows for subsequent comparison with the learner’s actual response, enabling reflection and updating of the recommendation method. This process can be formalized as:

$$(F_{t+1}, A_{t+1}) \sim \text{LLM}(\mathbf{P}_F, \mathcal{M}_t^D, e_{t+1}), \quad (9)$$

where \mathbf{P}_F notes the prompt. This information will be sent to GTA to update the Log memory, which is used to reflect whether the current recommendation is correct and efficient.

Experiment

Dataset

Our experiments are performed on three real-world datasets: Junyi¹ and ASSIST09² and we collect a dataset with question text content details from the real-world scenarios, note as TextLog. All datasets contain learners’ learning log data and the knowledge concept name for all exercises. For the Junyi dataset, we use the “topic” field as learning items, which are commonly used in education and we use the “name (Exercise name)” as the text content of exercises. For the ASSIST09 dataset, the “skill name” field is used to represent learning items. Due to ASSIST09 not having direct information about exercises, we use the information from some fields of provided data, such as “the response time” and “original (Main/Scaffolding Problem)” etc. to represent the exercises. In addition, the TextLog dataset contains the text content for each practice item. We construct the knowledge transition graph following (Gao et al. 2021) for all three datasets. The statistics of datasets can be found in Table 1.

¹<https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=1198>

²<https://sites.google.com/site/assistmentsdata/home/>

Statistics	Junyi	ASSIST09	TextLog
Knowledge Concepts	36	97	698
Exercises	711	16,836	8021
Learners	245,511	4,092	127,610
Response records	25,367,573	397,235	1,680,886
Records / Learners	1034	97	13

Table 1: Statistics of datasets.

Simulators

A critical challenge in evaluation is that existing realistic datasets only provide static information, making it difficult to assess whether practice items not presented in a sequence can be answered correctly (Huang et al. 2019). As a result, these datasets are not suitable for evaluating learning paths. To address this, we follow prior works (Liu et al. 2019; Chen et al. 2023) and employ a Knowledge Evolution-based Simulator (KES) as introduced in (Liu et al. 2019). KES is a data-driven system that utilizes the DKT model (Piech et al. 2015) to simulate the dynamic changes in learners’ knowledge states. Initial logs from these datasets are used to simulate the learner’s starting state (Li et al. 2023b).

Baselines

In the Learning Path Recommendation (LPR) task, we compare our approach against existing methods as baselines. It is important to note that all these baseline models only use the sequence of question(or knowledge concept) IDs by learners while ignoring the textual information of the questions and knowledge concepts. Consistent with prior studies (Li et al. 2023b), we use the improvement E_p (Eq. 1) provided by the simulators to evaluate following methods:

- KNN: KNN (Cover and Hart 1967) identifies similar learners based on their learning paths and determines the next learning item for a new learner by referencing the paths of the closest identified learners.
- GRU4Rec: GRU4Rec (Hidasi et al. 2015) is a well-known model that takes the session sequence as input and generates a probability distribution predicting the learning items most likely to appear next.
- DQN: DQN (Chen et al. 2018) uses a neural network to evaluate action values and recommends the action with the highest value.
- Actor-Critic: This approach uses a GRU encoder within a standard actor-critic framework (Konda and Tsitsiklis 1999) to provide recommendations.
- CB: Contextual Bandits (Intayoad, Kamyod, and Temdee 2020) frames the learning path recommendation process as a contextual bandit problem.
- RLTutor: RLTutor (Kubotani, Fukuhara, and Morishima 2021) is an adaptive tutoring system that integrates a model-based reinforcement learning approach with DAS3H (Dwivedi, Kant, and Bharadwaj 2018) for recommending learning items.

	Junyi	ASSIST09	TextLog
KNN	0.1343	-0.0932	0.0085
GRU4Rec	0.0993	-0.1344	-0.0002
DQN	0.1536	-0.0267	-
Actor-Critic	0.1916	0.0676	-
CB	0.2098	0.0038	-
RLTutor	-0.1034	0.0784	-
CSEAL	0.2505	0.1009	-
GEHRL	<u>0.4206</u>	<u>0.1971</u>	-
GenAL	0.5692	0.3665	0.3804

Table 2: Performance comparison for learning path recommendation methods at 20 learning steps. Existing state-of-the-art results are underlined and the best results are **bold**. Our GenAL is compared with the SOTA GEHRL. It should be noted that “-” in the table indicates that the method cannot achieve absolute promotion and meet the learning goals.

- **CSEAL:** CSEAL (Liu et al. 2019) utilizes an actor-critic framework with cognitive navigation for learning path recommendation.
- **GEHRL:** GEHRL (Li et al. 2023b) enables efficient goal planning and achievement through Hierarchical Reinforcement Learning, with the GEHRL-EB variant used for comparison due to its superior performance.

Implementation Details

In our framework, we employ three LLM-based models for testing: Llama2-7B, Llama3-8B, and the GPT-3.5-turbo provided by OpenAI. The temperature parameter is set to 0.9. The dataset divided method is following (Liu et al. 2019). In particular, our GenAL uses the training dataset to train the simulator and initial the learner’s profile. Then we use the test dataset to achieve inference. Our code is available at <https://github.com/karin0018/GenAL>.

Main Results

Table 2 presents the average E_p values of all models across the three datasets, and the learning step is 20. The results reveal several important insights.

- **Promotion Comparison:** Our proposed method outperforms all existing baselines across the three datasets. This demonstrates the importance and necessity of analyzing data text in the task of learning path recommendation. Additionally, it highlights the potential of large language models in learning path recommendation tasks. Most methods perform best on the Junyi dataset because it involves fewer knowledge concepts and a smaller number of exercises, which reduces the model’s selection space.
- **Generalization Performance:** Our method does not rely on training with a specific dataset, allowing it to generalize to new data distributions after a small amount of initial data is used for setup. For example, while existing methods can only be trained and tested on the same dataset, our

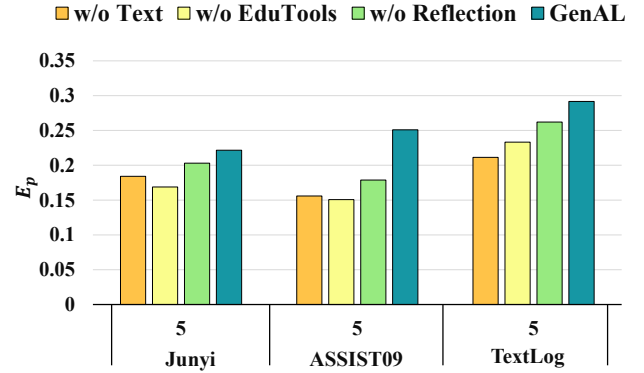


Figure 2: Results of ablation experiments on all datasets.

approach requires no retraining and can be transferred to different datasets while maintaining effective recommendation performance. It should be clear that negative values in Junyi and ASSIST09 are due to the presence of a large number of items in these simulators, some of which never occurred in the training data. This can lead to unstable predictions by the KT model, resulting in negative rewards (Li et al. 2023b).

- **Stability Analysis:** Although reinforcement learning-based methods like CSEAL and GEHRL perform well in scenarios with abundant interaction data—thanks to interactive feedback and sufficient long-term cumulative rewards—they fail when interaction data is sparse, such as in the TextLog dataset, where students have an average of only 13 records. These methods struggle to train effectively and cannot recommend learning paths during testing. Our proposed GenAL, a content-based recommendation framework, is unaffected by sparse data. GenAL leverages the rich subject knowledge and reasoning abilities of large language models to model students’ learning abilities and content from minimal initial data, enabling adaptive learning path generation.

Ablation Study

Impact of different modules. We conducted ablation experiments on the key modules of GenAL, and the results are shown in Figure 2, which presents the E_p scores across three datasets when the learning steps are set to 5, and the LLM is using GPT-3.5-turbo. “w/o Text” indicates that we only use the item ID during recommendation process, instead of text content for all items (including knowledge concepts and exercises); “w/o EduTools” notes we drop the Educational Tools module, which means we are unable to obtain prior knowledge through additional educational tools, such as the relationships between knowledge concepts. “w/o Reflection” indicates that we do not update the recommendation strategy or the student’s profile during the learning path recommendation process. The experimental results show a significant performance decrease when text information is not utilized in our framework, highlighting the importance of textual content. Additionally, the prior knowledge pro-

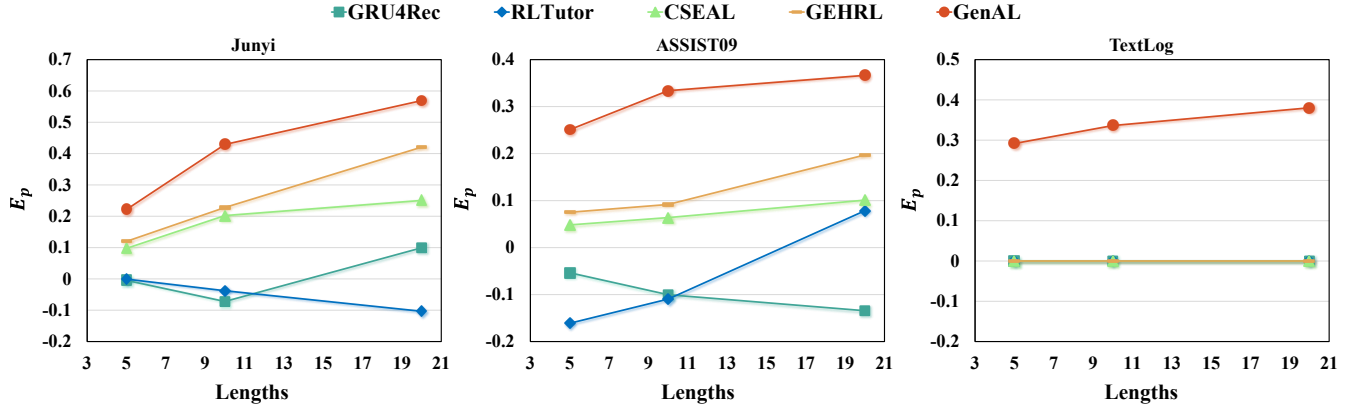


Figure 3: Impact of different lengths on three datasets with representative baseline models and our GenAL. It should be noted that the baseline models failed to achieve the learning goals in TextLog, so we present their results as 0.00.

Steps	5 Steps	10 Steps	20 Steps
Llama2-7b	0.2235±0.046	0.2517±0.057	0.3591±0.007
Llama3-8b	0.2720±0.047	0.3262±0.028	0.3731±0.009
GPT-3.5-turbo	0.2917±0.012	0.3363±0.017	0.3804±0.011

Table 3: Robustness estimation across different LLMs for TextLog dataset. \pm means standard deviation.

vided by Educational Tools is also crucial, demonstrating that relying solely on the internal knowledge of the large language model is insufficient; supplementary knowledge is necessary to assist the model in making decisions. Furthermore, since learners’ abilities are constantly evolving, the recommendation effectiveness declines when the model no longer dynamically updates the recommendation strategy.

Impact of different path lengths. Figure 3 shows the performance of paths of various lengths generated by different models across three datasets. We compare several representative baseline models with our GenAL. The results demonstrate that our method consistently outperforms the baselines in various scenarios, further validating its effectiveness. As the number of learning steps increases, most methods show improved outcomes with longer paths, which aligns with educational intuition. In datasets with sufficient training data (e.g., Junyi and ASSIST09), reinforcement learning-based methods like CSEAL and GEHRL perform competitively, while RLtutor starts to show instability. Probabilistic methods like GRU4Rec prove less effective. The RL-based methods rely on abundant data and fail to converge on the sparse TextLog dataset, making it impossible to reach learning objectives during testing (i.e., they fail to run), so we report their results as 0.00. In contrast, our GenAL consistently performs well on TextLog.

Effects of Different LLMs

To validate the robustness of the GenAL framework across various base models, we conduct additional experiments with other different LLM backbones: “Llama-2-7B” and

“Llama-3-8B”. The results are presented in Table 3, which shows that our GenAL framework performs well across different LLMs, demonstrating its robustness. Additionally, we repeated the experiments three times under the setting of temperature = 0.9 and calculated their standard variance. The results indicate that the performance of LLMs is stable within our framework. The recommendation effectiveness of the “Llama-3-8B” and “GPT-3.5-turbo” models outperform the “Llama-2-7B”, indicating a positive correlation between our framework’s performance and the knowledge embedded in the large language models.

Conclusion

In this paper, we presented a focused study on adaptive learning, which aimed to recommend personalized learning items to meet the unique needs of each learner. Building upon the powerful semantic understanding of LLMs, we introduced a generalized Generative Agent for Adaptive Learning (GenAL), which integrated educational tools to facilitate effective and adaptable learning path recommendations across diverse data distributions. Specifically, our GenAL comprised two components: the Global Thinking Agent and the Local Teaching Agent. The former was designed to update the learner profile and evaluate recommendation outcomes based on the learner’s historical learning, while the latter was tasked with suggesting items using prior educational knowledge.

Limitation and Futureworks: Our work is an initial attempt to apply large language models in adaptive learning. The main limitations lie in the inference cost and inefficient use of external knowledge, and the hallucinations still exist in our framework, even with external knowledge, which also is a common issue with LLMs. Nevertheless, our work demonstrates the potential of LLMs in addressing complex problems in the educational domain. In the future, we plan to explore further applications, such as using LLMs to help students overcome specific challenges encountered during their learning process dynamically.

Acknowledgments

This research was partially supported by grants from the National Natural Science Foundation of China (Grants No. 62337001), the Key Technologies R & D Program of Anhui Province (No. 202423k09020039), and the Fundamental Research Funds for the Central Universities.

References

- Chen, J.; Lin, H.; Han, X.; and Sun, L. 2024. Benchmarking large language models in retrieval-augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 17754–17762.
- Chen, X.; Shen, J.; Xia, W.; Jin, J.; Song, Y.; Zhang, W.; Liu, W.; Zhu, M.; Tang, R.; Dong, K.; et al. 2023. Set-to-sequence ranking-based concept-aware learning path recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 5027–5035.
- Chen, Y.; Li, X.; Liu, J.; and Ying, Z. 2018. Recommendation system for adaptive learning. *Applied psychological measurement*, 42(1): 24–41.
- Cover, T.; and Hart, P. 1967. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1): 21–27.
- Dwivedi, P.; Kant, V.; and Bharadwaj, K. K. 2018. Learning path recommendation based on modified variable length genetic algorithm. *Education and information technologies*, 23: 819–836.
- Elshani, L.; and Nuçi, K. P. 2021. Constructing a personalized learning path using genetic algorithms approach. *arXiv preprint arXiv:2104.11276*.
- Gao, W.; Liu, Q.; Huang, Z.; Yin, Y.; Bi, H.; Wang, M.-C.; Ma, J.; Wang, S.; and Su, Y. 2021. RCD: Relation Map Driven Cognitive Diagnosis for Intelligent Education Systems. *SIGIR '21*, 501–510. Association for Computing Machinery.
- Hidasi, B.; Karatzoglou, A.; Baltrunas, L.; and Tikk, D. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*.
- Huang, Z.; Liu, Q.; Zhai, C.; Yin, Y.; Chen, E.; Gao, W.; and Hu, G. 2019. Exploring multi-objective exercise recommendations in online education systems. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 1261–1270.
- Intayoad, W.; Kamyod, C.; and Temdee, P. 2020. Reinforcement learning based on contextual bandits for personalized online learning recommendation systems. *Wireless Personal Communications*, 115(4): 2917–2932.
- Konda, V.; and Tsitsiklis, J. 1999. Actor-critic algorithms. *Advances in neural information processing systems*, 12.
- Kubotani, Y.; Fukuhara, Y.; and Morishima, S. 2021. RL-Tutor: Reinforcement Learning Based Adaptive Tutoring System by Modeling Virtual Student with Fewer Interactions. *arXiv preprint arXiv:2108.00268*.
- Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33: 9459–9474.
- Li, Q.; Fu, L.; Zhang, W.; Chen, X.; Yu, J.; Xia, W.; Zhang, W.; Tang, R.; and Yu, Y. 2023a. Adapting large language models for education: Foundational capabilities, potentials, and challenges. *arXiv preprint arXiv:2401.08664*.
- Li, Q.; Xia, W.; Du, K.; Zhang, Q.; Zhang, W.; Tang, R.; and Yu, Y. 2024. Learning Structure and Knowledge Aware Representation with Large Language Models for Concept Recommendation. *arXiv preprint arXiv:2405.12442*.
- Li, Q.; Xia, W.; Yin, L.; Shen, J.; Rui, R.; Zhang, W.; Chen, X.; Tang, R.; and Yu, Y. 2023b. Graph Enhanced Hierarchical Reinforcement Learning for Goal-Oriented Learning Path Recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 1318–1327.
- Liu, Q.; Huang, Z.; Yin, Y.; Chen, E.; Xiong, H.; Su, Y.; and Hu, G. 2021. EKT: Exercise-Aware Knowledge Tracing for Student Performance Prediction. *IEEE Transactions on Knowledge and Data Engineering*, 33(1): 100–115.
- Liu, Q.; Tong, S.; Liu, C.; Zhao, H.; Chen, E.; Ma, H.; and Wang, S. 2019. Exploiting Cognitive Structure for Adaptive Learning. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '19*, 627–635. Association for Computing Machinery.
- Nabizadeh, A. H.; Goncalves, D.; Gama, S.; Jorge, J.; and Rafsanjani, H. N. 2020a. Adaptive learning path recommender approach using auxiliary learning objects. *Computers & Education*, 147: 103777.
- Nabizadeh, A. H.; Leal, J. P.; Rafsanjani, H. N.; and Shah, R. R. 2020b. Learning path personalization and recommendation methods: A survey of the state-of-the-art. *Expert Systems with Applications*, 159: 113596.
- Nabizadeh, A. H.; Mário Jorge, A.; and Paulo Leal, J. 2017. Rutico: Recommending successful learning paths under time constraints. In *Adjunct publication of the 25th conference on user modeling, adaptation and personalization*, 153–158.
- Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744.
- Piech, C.; Bassen, J.; Huang, J.; Ganguli, S.; Sahami, M.; Guibas, L. J.; and Sohl-Dickstein, J. 2015. Deep knowledge tracing. *Advances in neural information processing systems*, 28.
- Shi, W.; He, X.; Zhang, Y.; Gao, C.; Li, X.; Zhang, J.; Wang, Q.; and Feng, F. 2024. Large Language Models are Learnable Planners for Long-Term Recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '24*, 1893–1903. Association for Computing Machinery.
- Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale,

S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Wang, F.; Liu, Q.; Chen, E.; Huang, Z.; Yin, Y.; Wang, S.; and Su, Y. 2023. NeuralCD: A General Framework for Cognitive Diagnosis. *IEEE Transactions on Knowledge and Data Engineering*, 35(8): 8312–8327.

Wang, L.; Ma, C.; Feng, X.; Zhang, Z.; Yang, H.; Zhang, J.; Chen, Z.; Tang, J.; Chen, X.; Lin, Y.; et al. 2024. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6): 186345.

Wu, J.; Zhang, H.; Huang, Z.; Ding, L.; Liu, Q.; Sha, J.; Chen, E.; and Wang, S. 2024. Graph-based Student Knowledge Profile for Online Intelligent Education. In *Proceedings of the 2024 SIAM International Conference on Data Mining (SDM)*, 379–387. SIAM.

Wu, Q.; Bansal, G.; Zhang, J.; Wu, Y.; Zhang, S.; Zhu, E.; Li, B.; Jiang, L.; Zhang, X.; and Wang, C. 2023. Autogen: Enabling next-gen llm applications via multi-agent conversation framework. *arXiv preprint arXiv:2308.08155*.

Xu, S.; Zhang, X.; and Qin, L. 2024. EduAgent: Generative Student Agents in Learning. *arXiv preprint arXiv:2404.07963*.

Yang, S.; Yu, X.; Tian, Y.; Yan, X.; Ma, H.; and Zhang, X. 2024. Evolutionary neural architecture search for transformer in knowledge tracing. *Advances in Neural Information Processing Systems*, 36.

Yao, F.; Liu, Q.; Hou, M.; Tong, S.; Huang, Z.; Chen, E.; Sha, J.; and Wang, S. 2023. Exploiting non-interactive exercises in cognitive diagnosis. *Interaction*, 100(200): 300.

Yao, F.; Liu, Q.; Yue, L.; Gao, W.; Li, J.; Li, X.; and He, Y. 2024. Adard: An adaptive response denoising framework for robust learner modeling. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 3886–3895.

Yu, X.; Qin, C.; Shen, D.; Ma, H.; Zhang, L.; Zhang, X.; Zhu, H.; and Xiong, H. 2024a. Rdgt: enhancing group cognitive diagnosis with relation-guided dual-side graph transformer. *IEEE Transactions on Knowledge and Data Engineering*.

Yu, X.; Qin, C.; Shen, D.; Yang, S.; Ma, H.; Zhu, H.; and Zhang, X. 2024b. Rigl: A unified reciprocal approach for tracing the independent and group learning processes. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 4047–4058.

Zhang, Y.; Li, Y.; Cui, L.; Cai, D.; Liu, L.; Fu, T.; Huang, X.; Zhao, E.; Zhang, Y.; Chen, Y.; Wang, L.; Luu, A. T.; Bi, W.; Shi, F.; and Shi, S. 2023. Siren’s Song in the AI Ocean: A Survey on Hallucination in Large Language Models. *arXiv:2309.01219*.

Zhang, Z.; Song, W.; Liu, Q.; Mao, Q.; Wang, Y.; Gao, W.; Huang, Z.; Wang, S.; and Chen, E. 2024. Towards Accurate and Fair Cognitive Diagnosis via Monotonic Data Augmentation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.