



MAS-KCL: knowledge component graph structure learning with large language model-based agentic workflow

Yuan-Hao Jiang^{1,2,3,4} · Kezong Tang⁵ · Zi-Wei Chen⁵ · Yang Wei^{1,2,3,6} · Tian-Yi Liu⁷ · Jiayi Wu^{3,8}

Accepted: 21 April 2025

© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2025

Abstract

Knowledge components (KCs) are the fundamental units of knowledge in the field of education. A KC graph illustrates the relationships and dependencies between KCs. An accurate KC graph can assist educators in identifying the root causes of learners' poor performance on specific KCs, thereby enabling targeted instructional interventions. To achieve this, we have developed a KC graph structure learning algorithm, named MAS-KCL, which employs a multi-agent system driven by large language models for adaptive modification and optimization of the KC graph. Additionally, a bidirectional feedback mechanism is integrated into the algorithm, where AI agents leverage this mechanism to assess the value of edges within the KC graph and adjust the distribution of generation probabilities for different edges, thereby accelerating the efficiency of structure learning. We applied the proposed algorithm to 5 synthetic datasets and 4 real-world educational datasets, and experimental results validate its effectiveness in learning path recognition. By accurately identifying learners' learning paths, teachers are able to design more comprehensive learning plans, enabling learners to achieve their educational goals more effectively, thus promoting the sustainable development of education.

Keywords Graph structure learning · Knowledge component · Large language model · Multi-agent system · AI agent

Yuan-Hao Jiang, Kezong Tang and Zi-Wei Chen contributed equally to this work

✉ Kezong Tang
tangkezong@jci.edu.cn

✉ Zi-Wei Chen
ziwei.cs@foxmail.com

¹ Shanghai Institute of Artificial Intelligence for Education, East China Normal University, Shanghai, China

² School of Computer Science and Technology, East China Normal University, Shanghai, China

³ Lab of Artificial Intelligence for Education, East China Normal University, Shanghai, China

⁴ Graduate School, Shanghai Jiao Tong University, Shanghai, China

⁵ School of Information Engineering, Jingdezhen Ceramic University, Jingdezhen, China

⁶ School of Computing, National University of Singapore, Singapore, Singapore

⁷ School of Computer, Jiangsu University of Science and Technology, Zhenjiang, China

⁸ Department of Educational Information Technology, East China Normal University, Shanghai, China

1 Introduction

The increasing global emphasis on sustainable development has brought quality education to the forefront, as highlighted by the United Nations Educational, Scientific and Cultural Organization (UNESCO) and its 17 Sustainable Development Goals (SDGs), which include goals, such as equitable education, sustainable communities, and reduced inequalities. Among the key pillars of educational development are inclusivity, resource accessibility, educational technology, and sustainability. Educational technology, in particular, leverages digital tools and information systems to enhance interactivity and broaden access to learning. Its integration has transformed teaching and learning processes [2, 44]. However, traditional digital education often lacks sufficient personalization and coherent instructional design, leading to cognitive overload. Addressing these limitations requires a deeper understanding of the structural relationships among knowledge components (KCs) commonly referred to as learning paths [20].

Learning paths capture the dependencies among knowledge components and play a crucial role in guiding learners' progression. Consequently, the completeness and hierarchi-

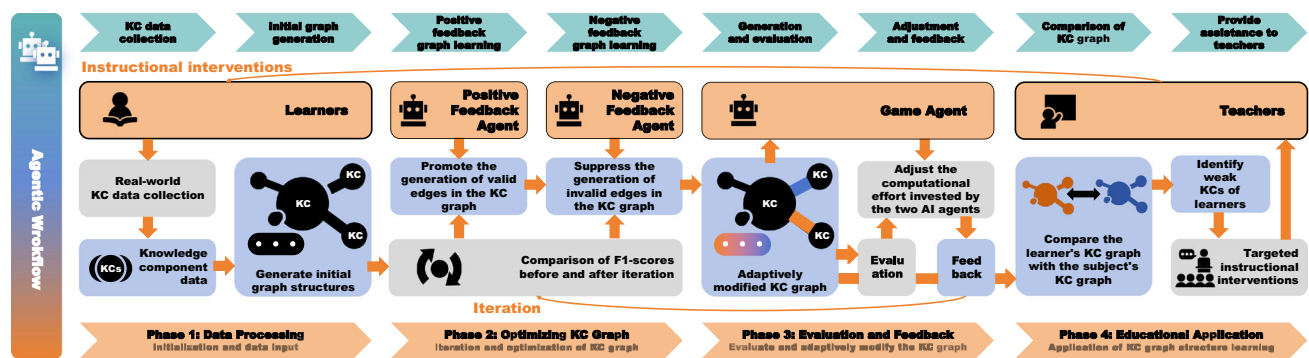


Fig. 1 The proposed agentic workflow for KC graph structure learning. This framework injects the emergent intelligence of large language models into the designed multi-agent system, enabling it to extract KCs from real-world learning data and construct KC graphs. The agentic

workflow facilitates deep collaboration among learners, teachers, and AI agents, assisting teachers in identifying the root causes of learners' poor performance at the KC level, thereby providing support for the development of targeted instructional interventions

cal organization of knowledge component structures are widely recognized as key indicators of learners' proficiency [31, 35]. Recent studies have employed clustering methods to classify learners based on their knowledge structures, uncovering links between structure patterns and learning outcomes [10]. Graph structures [22, 23], particularly directed acyclic graphs [21], offer an effective means to encode such educational data. In these graphs, nodes denote individual knowledge components and edges define directed dependencies, thereby visualizing the logical flow of learning.

To address the inherent complexity of real-world knowledge structures, we propose a multi-agent system for graph structure learning specifically tailored for learning path recognition. This system leverages a differential evolution (DE) algorithm augmented with dynamic population control, which is guided by artificial intelligence (AI) agents through a bidirectional feedback mechanism. Experimental evaluations demonstrate that our approach significantly improves the accuracy of graph structure learning. The main contributions are as follows:

- We propose a Multi-Agent System for Knowledge Component graph structure Learning (MAS-KCL), as illustrated in Fig. 1. The effectiveness of the proposed method is validated using real-world educational datasets.
- We develop a large language model (LLM)-powered agentic workflow for MAS-KCL, where the integration of LLMs effectively facilitates the structural learning of KC graphs.
- The KC graphs generated by MAS-KCL capture causal relationships among KCs within a specific subject, offering data-driven support for instructional interventions. For example, if a learner struggles with multiplication, the root cause may lie in an insufficient understanding of addition-related KCs. The proposed method also provides valuable insights for educational practice.

2 Related work

2.1 Causal graph

When recognizing learning paths, it is essential to determine the relationships among KCs within a KC graph [20]. Most KCs exhibit causal dependencies, which can be effectively represented using a causal graph—a directed acyclic graph (DAG) that models the connections between phenomena and their underlying causes. A method for constructing causal knowledge networks based on Bayesian networks (BNs) has been proposed, leveraging causal relationships to build these structures [38].

The application of causal graphs in educational research has been widely acknowledged, offering key advantages for causal inference and facilitating empirical investigation [7]. Their value lies in enabling explicit causal reasoning, supported by tools that assist researchers in uncovering causal links, as demonstrated in educational technology studies [39]. It is important to clarify that the concept of a causal graph is closely related to the KC graph. A causal graph is defined as a graph composed of a set of nodes and edges, where the edges indicate causal relationships between elements in the node set. The KC graph further constrains the node set to represent knowledge components specifically.

2.2 Knowledge component graph structure learning

KC graph structure learning has become a central topic in intelligent tutoring systems (ITS). Early studies relied on expert-defined prerequisite structures, typically modeled using BNs, where domain knowledge is encoded as probabilistic dependencies [3, 5]. To enhance flexibility and reduce reliance on expert priors, later work proposed data-driven approaches that learn conditional probabilities directly from learners' data [6]. These methods infer KC dependencies by analyzing the temporal evolution of knowledge mas-

tery, comparing patterns predicted by learner models [6, 26]. Dynamic Bayesian networks (DBNs) extended this line by embedding causal relations between KCs [15], though they suffer from parameter explosion when modeling multiple prerequisites. To improve scalability and interpretability, the embedding prerequisite relationships in student modeling (E-PRISM) framework introduces interpretable parameters and infers causal dependencies from learning trajectories [1], while more recent research applies interventional and counterfactual reasoning to extract explainable KC relationships [38].

Moreover, data-driven approaches have also adopted reinforcement learning (RL) [18, 32] and graph neural networks (GNNs) [27, 41] for KC graph learning. GNNs are effective at capturing structural patterns but lack interpretability and causal representation, and are often limited to static graphs. RL-based methods treat graph construction as a sequential decision process, aligning with the dynamics of learning, but require careful reward design and struggle with sparse feedback. To address these limitations, we propose MAS-KCL, a multi-agent system driven by LLMs. MAS-KCL employs an agentic workflow where LLM-based agents iteratively generate and adaptively modify KC graphs through interpretable interactions. A bidirectional feedback mechanism enables agents to assess edge importance and dynamically adjust generation probabilities, leading to efficient, interpretable, and causally grounded KC structure learning.

2.3 AI agent for education

The Transformer architecture [4, 12, 36], driven by the attention mechanism [45], has underpinned the rapid evolution of LLMs [28], which are increasingly applied in educational contexts. LLMs can generate personalized learning plans tailored to learners' habits and backgrounds [19, 37], and support external tools through iterative dialogue [46]. Frameworks such as LangChain have demonstrated the capability to automatically generate contextualized mathematical multiple-choice questions with improved accuracy [17]. Recent studies have explored how LLMs and adaptive learning models serve as interfaces for human-computer interaction (HCI) in personalized education [40], and how multimodal large language models (MLLMs) can provide synchronized visual content to enhance mathematics learning experiences in HCI contexts [13].

Parallel to this development, LLMs-based AI agents have emerged as versatile tools in educational systems. By integrating web services and modeling learner characteristics, agents can infer learning domains and facilitate intelligent information retrieval [16]. Multi-agent systems (MASs), including those based on von Neumann architectures, have shown effectiveness in enhancing learners' learning and supporting teaching processes [14]. While MASs cannot replace

human roles entirely, they promote multi-agent collaboration (MAC) and learner interaction, improving learning outcomes [29]. In this work, we adopt a MAS framework to optimize the learning of KC graph structures.

3 MAS-KCL for KC graph structure learning

3.1 LLM-based multi-agent system

This section introduces the multi-agent system composed of AI agents, which dynamically control parameters to deliver decision outputs during the algorithm's iterative process. These agents interact with MAS-KCL by influencing the optimization process through decisions continuously updated via LLM calls. The overall interaction between agents and the MAS-KCL-controlled population is illustrated in Fig. 2. In addition, the pseudocode of the proposed method is provided in Algorithm 1.

Specifically, after each iteration of the algorithm, the Game Agent is invoked using prompt engineering. It is asked to determine whether to adjust the Ambient Pressure (AP) parameter in MAS-KCL-based on the magnitude of *loss* change observed in the current iteration-and to what extent. The decision is returned in a structured JSON format, which includes both the decision outcome and its reasoning. Before the next iteration begins, the value of AP is updated accordingly, allowing the algorithm to remain dynamically aligned with the intermediate decisions made by the Game Agent, as described in Algorithm 1. The value of the AP parameter determines the proportion of elite individuals retained for the next iteration. It reflects the dynamic allocation of computational effort between exploration and exploitation.

Similarly, the Positive Feedback Agent and Negative Feedback Agent are invoked at the end of each iteration. Based on the current *loss* change and parameter status, they make decisions regarding the activation and adjustment of the positive and negative feedback mechanisms, respectively. These decisions take effect in the subsequent iteration and are passed to MAS-KCL through dynamic parameter control, as described in Lines 20-21 of Algorithm 1. The Positive Feedback Agent focuses on identifying edges that contribute to increased *loss* and aims to eliminate these low-quality edges broadly across the population. In contrast, the Negative Feedback Agent is responsible for identifying edges associated with decreased *loss* and promotes the widespread addition of these valuable edges throughout the population. Together, these agents collaborate to enable efficient learning of the KC graph structure, thereby supporting teachers in making informed instructional interventions, as illustrated in Fig. 1. Further details on the interaction between the multi-agent system and the MAS-KCL framework are provided in the following Sect. 3.2.

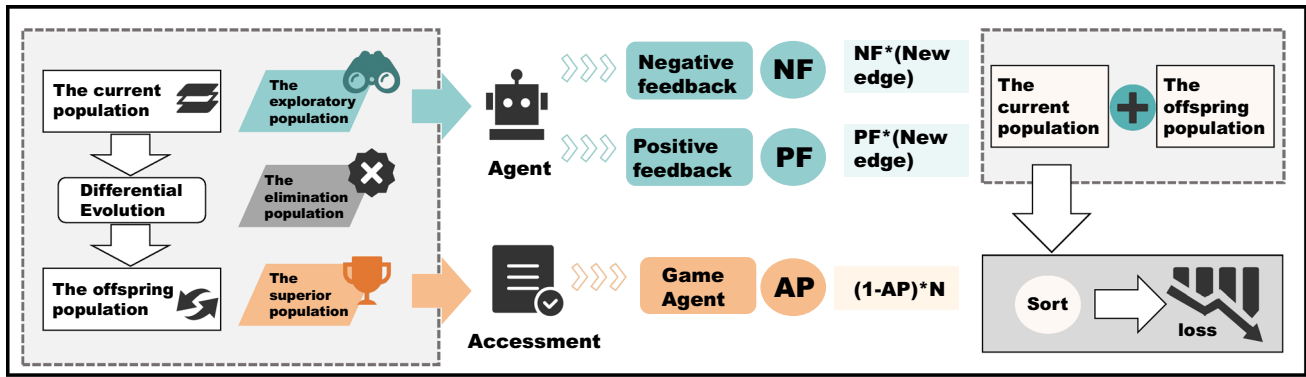


Fig. 2 Game agent and feedback agents. The diagram displays the interaction between the Game Agent and the population, as well as the probability of adding new edges in the bidirectional feedback mechanism generated by the Feedback Agents

3.2 The structure of the proposed MAS-KCL

To improve the balance between exploration and exploitation, the proposed MAS-KCL adopts a multi-sub-population strategy that partitions the population into superior, exploratory, and elimination sub-populations. This structure allows certain individuals to exploit known promising regions while others explore new areas, enhancing global search capacity and avoiding premature convergence. The algorithm evaluates each individual using a fitness score F , with the *loss* defined as $1 - F$ [20]. A higher F indicates better structural alignment and, consequently, a lower *loss*, reflecting a more effective solution.

The main process of the algorithm is detailed in Algorithm 1. The procedure begins with the initialization of a random population and the setting of the Function Evaluation (FE) counter to zero, as shown in Lines 1–2 of Algorithm 1. In each iteration, DE is employed to generate an offspring population, which is then sorted according to a predefined optimization direction, as described in Lines 4–5. The offspring are ranked in ascending order of *loss* and divided into three sub-populations, as outlined in Lines 6–7. The proportion of the superior sub-population is controlled by the ambient pressure parameter AP , where a higher AP preserves more elite individuals, and a lower value encourages exploration. The exploratory sub-population, sized as $(1 - AP) \cdot N$, is further refined through a bidirectional feedback mechanism, as implemented in Lines 7–15 of Algorithm 1.

Within this feedback mechanism, individuals are updated based on their sub-population assignments: P_{PFO} (high-fitness exploratory individuals) and P_{NFO} (remaining individuals) influence the addition or deletion of edges through positive and negative factors (PF , NF). These structural changes are quantified using the metric $count_{ones}$. The modified exploratory sub-population is then merged with the superior sub-population and passed to the next generation, as shown in Line 17 of Algorithm 1. Following this, both parent and offspring populations are re-ranked, and the least

fit individuals are eliminated to maintain selection pressure and solution quality, as described in Lines 16–17. Then, the best individuals are retained based on AP , and the remaining ones are refined through the feedback mechanism for the subsequent iteration, as described in Lines 18–21 of Algorithm 1. This iterative process continues until FE reaches the maximum Function Evaluation ($\max FE$), at which point the algorithm terminates and outputs the optimal individual, as shown in Lines 22–23. The structure learning process for the KC graph is illustrated in Fig. 3, and the fitness definition aligns with prior work [20].

3.3 Time and Space Complexity Analysis

The MAS-KCL algorithm involves iterative operations including individual generation, structure evaluation, ranking, and LLM-based feedback. Its computational cost mainly depends on the number of knowledge components n . Each iteration involves $D = n \cdot (n - 1) / 2 = (n^2 - n) / 2$ structure comparisons, along with fixed overheads and three LLM API calls, leading to a per-iteration cost of $O(n^2 - n + 3c_{API} + c_t)$. With a total of $\max FE / N$ iterations, the overall time complexity is:

$$\begin{aligned} T(n) &= O\left(\frac{\max FE}{N} \cdot (n^2 - n + 3c_{API} + c_t)\right) \\ &= O(n^2 - n + a) \end{aligned}$$

In terms of space, the algorithm maintains structural encodings of all individuals, each of size $D = (n^2 - n) / 2$. Including auxiliary storage, the space complexity is $S(n) = O(n^2 - n + c_s)$. Here, $\max FE$ denotes the maximum number of fitness evaluations, N is a constant representing the population size, and n denotes the number of KCs. c_{API} represents the waiting time for LLM responses through API calls, while c_t , c_s , and a are all constants. These results show that MAS-KCL exhibits quadratic time and space growth with respect to the KC graph size.

Algorithm 1: Multi-Agent System for Knowledge Component Graph Structure Learning**Input:** AP (Ambient Pressure), OD (Optimization Direction)**Output:** best individual and its $loss$

```

1 Initialize the population randomly;
2  $EF \leftarrow 0$ ;
3 while  $EF < maxEF$  do
4   Generate offspring population by DE operator;
5    $OffPop \leftarrow sort([OffPop, OD])$ ;
   // Sort individuals in the offspring
   // population according to  $OD$ .
6    $OffPop \leftarrow \{(AP) * N[P_{sup}], (1 - AP) * N[P_{EX}], N[P_{EL}]\}$ ;
   // Divide the offspring population into
   // superior and exploratory, and
   // elimination sub-populations.
7   Establish bidirectional feedback mechanism;
8    $P_{EX} \leftarrow sort([P_{PFO}, P_{NFO}])$ ;
   // Sort exploratory sub-populations.
9   foreach  $i \in \{P_{PFO}, P_{NFO}\}$  do
10     $NewEdges \leftarrow OffPop(1, i).dec - Pop(1, i).dec$ ;
11     $count\_ones \leftarrow count\_ones + NewEdges$ ;
    // Record the new edges of the
    // exploratory sub-population for the
    // current population.
12  Multiply the vector of the newly added edge by the Positive
    // Factor;
13  Multiply the vector of the newly added edge by the Negative
    // Factor;
14  foreach  $i \in \{P_{PFO}, P_{NFO}\}$  do
15     $OffPop(1, i).dec$  is modified by adding or deleting
    // edges based on  $count\_ones$ ;
16   $sort\{Fitness(Pop), Fitness(OffPop)\}$ ;
   // The fitness of the offspring
   // population was compared with the
   // current one.
17   $Pop(replace) \leftarrow OffPop(replace)$ ;
   // Sub-population with lowest fitness is
   // eliminated, and the remaining
   // sub-populations are retained.
18   $Pop([P_{SUP}], [P_{EX}]) \leftarrow OffPop\{(AP) * N, (1 - ap) * N\}$ ;
   // A portion of superior population will
   // enter the next iteration. The
   // exploratory population will undergo a
   // bidirectional feedback mechanism
   // operation.
19   $AP \leftarrow$  Game Agent decision;
20   $PF \leftarrow$  Positive Feedback Agent decision;
21   $NF \leftarrow$  Negative Feedback Agent decision;
22   $FE \leftarrow FE + N$ ;
23 return best individual and its  $loss$ ;

```

4 Results

4.1 Experimental setup

4.1.1 Educational datasets

The experiments in this study are mainly conducted using datasets from the NeurIPS CausalML Challenge and the Xue-

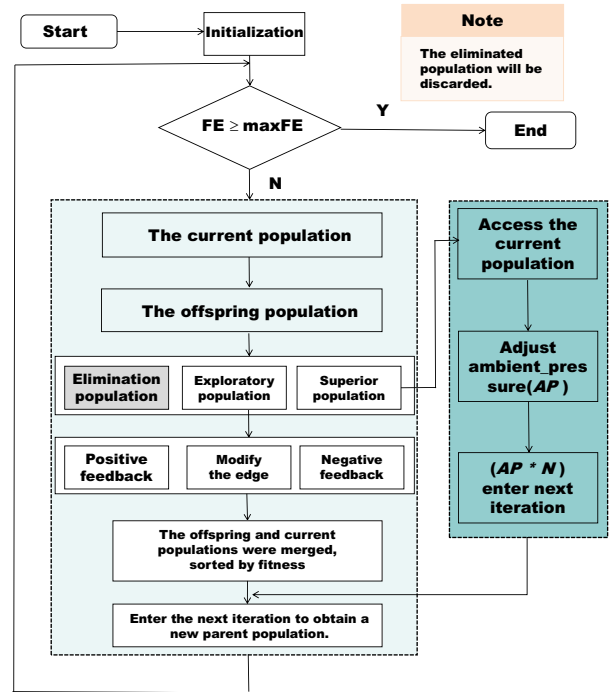


Fig. 3 KC graph structure learning search process. Figure 1 shows the search mechanism for the graph KC Structure based on multi-sub-population collaboration

tangX platform, both of which are suitable for the task of knowledge component graph structure learning. A total of 9 datasets are used in this research, including 5 synthetic datasets and 4 real-world datasets. Detailed descriptions of the datasets are provided below according to their respective sources.

- **Learning Path Recognition-Generate Datasets (LPR-GDs)** [9]: The LPR-GDs series datasets are provided by the *NeurIPS CausalML Challenge: Causal Insights for Learning Paths in Education*. This series includes five generated datasets-LPR-GD1 to LPR-GD5-which contain data on the causal relationships among KCs in the domain of mathematics. Although there exists a gap between synthetic and real-world data, the LPR-GDs datasets are well-suited for the preliminary validation and rapid testing of KC graph structure learning methods.
- **Learning Path Recognition-Real World Datasets (LPR-RWDs)** [8]: The LPR-RWDs series datasets provided by *Microsoft Research Cambridge* and *Eedi*. The LPR-RWDs include three real-world datasets of varying scales: LPR-RWD, LPR-RWD1, and LPR-RWD2. These datasets are constructed from real-world educational data provided by Eedi, an online education company registered in England and Wales. The LPR-GDs series datasets represent learning data collected from real learners engaged in mathematics education, making them

well-suited for thoroughly evaluating the generalizability of KC graph structure learning methods.

- **MOOCCubeX-Math** [42]: XuetangX is one of the largest MOOC platforms in China. The MOOCCubeX dataset is collected on the XuetangX platform, which provides partially annotated and algorithmically predicted prerequisite relationships between concepts [43]. By treating the concepts defined in MOOCCubeX as KCs, we can obtain real-world data suitable for the task of KC graph structure learning. Specifically, we extracted MOOCCubeX-Math from the 331,202 mathematics-related JSON records provided by MOOCCubeX and performed data cleaning based on the methodology in [30], in order to further evaluate the generalizability of the proposed approach.

4.1.2 Baseline algorithms

In this study, we compare our proposed algorithm, MAS-KCL, against four representative baseline algorithms: Multi-Stage multi-objective Evolutionary Algorithm (MSEA) [34], Golden Eagle Optimizer (GEO) [24], Elite Evolution Strategy-based Harris Hawk Optimization algorithm (EESHHO) [11], and Adaptive Geometry Estimation-based Many-Objective Evolutionary Algorithm II (AGE-MOEA-II) [25]. The selected methods exhibit diverse characteristics, making them suitable for a comprehensive comparison and evaluation of the proposed MAS-KCL from multiple perspectives. Our selection was guided by the recommendations in [20], from which MSEA, GEO, and EESHHO were adopted. Additionally, we included AGE-MOEA-II, a recently proposed algorithm with strong theoretical foundations, to further validate the effectiveness of the proposed LLM-driven approach.

It is worth noting that the parameter settings for all comparison algorithms were either derived from their original papers or based on the default configurations provided by the Platform for Evolutionary Multi-objective Optimization (PlatEMO) [33]. The hyperparameters of the comparison algorithms are listed in Table 1. It should be noted that AGE-MOEA-II [25] is not included in Table 1 because it does not contain adjustable parameters.

4.1.3 Implementation details

The experiments were primarily conducted on a machine equipped with an Intel i5-13600KF @ 3.50GHz dual-core processor, 32GB of RAM, and an NVIDIA RTX 4070 GPU. The software environment included MATLAB 2022b and the PlatEMO platform [33]. Additionally, the online tool *Chplot.online* was used for part of the data processing and visualization in this study. Following previous work, we employed the *loss* metric to evaluate the performance [20].

4.2 Comparison experiments on real-world datasets

This subsection compares the MAS-KCL with four algorithms on real-world datasets. To reduce the impact of random factors, each algorithm was run at least three times for each dataset. The results are shown in Table 2, where lower *loss* indicates better performance.

Table 2 presents the results on four real-world datasets: MOOCCubeX-Math, LPR-RWD, LPR-RWD1, and LPR-RWD2, with problem sizes D of 210, 6670, 1225, and 1225, respectively. The results for each algorithm are reported as the average *loss* values obtained through multiple independent runs. As shown in this table, LPR-RWD exhibits the highest *loss* among the three LPR-RWDs series datasets due to its largest problem size, making optimization more challenging. Meanwhile, the MOOCCubeX-Math dataset is relatively sparse, which leads to poorer performance across most algorithms compared to the other datasets. Overall, the EESHHO algorithm consistently delivers second-best performance across all datasets, while the proposed MAS-KCL achieves the best results in every case, demonstrating its effectiveness in the task of KC graph structure learning.

4.3 Generalization comparison on generated datasets

To further evaluate the generalization capability of the proposed method, we conducted additional comparison experiments on the generated datasets. Specifically, MAS-KCL was compared against the best-performing baseline algorithm identified in Sect. 4.2. The results on five generated datasets-LPR-GD1 to LPR-GD5-are presented in Table 3. Consistent with the results observed on real-world datasets, MAS-KCL achieves the best performance across all generated datasets. On average, MAS-KCL reduces the *loss* by 5.51%, with the greatest improvement observed on LPR-GD3, where the reduction reaches 7.71%. These results confirm the strong generalization ability of the MAS-KCL on both real-world and generated datasets.

4.4 Generalization sensitivity of MAS-KCL to LLMs

Since MAS-KCL integrates LLMs to enhance its decision-making process, we conducted a comparative study to analyze its generalization performance under different LLM versions. As shown in Table 4, we compared two GPT models developed by OpenAI (GPT-4.0 and GPT-3.5) with two alternative LLMs (LLaMA-70B and Claude-3-7-Sonnet-20250219) on four real-world datasets. The results show that the GPT models consistently outperform the other LLMs across all datasets. Notably, GPT-3.5 performs better on smaller-scale datasets such as MOOCCubeX-Math, while GPT-4.0 exhibits superior performance as the problem size

Table 1 Parameter settings of all algorithms

Algorithm	Parameters
GEO [24]	$AP_{min} = 0.5, AP_{max} = 2.0, CP_{min} = 1.0, CP_{max} = 0.5$
EESHHO [11]	$U_b = 1, L_b = 0$
MSEA [34]	$F_{max} = 1, F_{min} = 0$
MAS-KCL	$PF = 0.6, NF = 0.4, AP = 0.4$
This table presents the algorithm parameters used in the simulation experiments	

Table 2 Comparison of *loss* (Mean \pm SD) for five algorithms on four real-world datasets

Real-world dataset	Algorithm					
Dataset	D	MSEA	GEO	EESHHO	AGE-MOEA-II	MAS-KCL
MOOCubeX-Math	210	7.37e-1 (5.35e-3)[-]	6.46e-1 (4.08e-2)[-]	2.57e-1 (4.95e-2)[-]	7.28e-1 (6.15e-3) [-]	2.51e-1 (7.61e-2)
LPR-RWD	6670	4.84e-1 (1.78e-3) [-]	3.44e-1 (3.95e-5)[-]	3.43e-1 (2.76e-4)[-]	4.81e-1 (2.72e-3) [-]	3.31e-1 (1.30e-3)
LPR-RWD1	1225	4.13e-1 (4.36e-3)[-]	2.72e-1 (2.06e-3)[-]	2.71e-1 (1.59e-3)[-]	4.11e-1 (4.16e-3) [-]	2.23e-1 (4.23e-3)
LPR-RWD2	1225	4.43e-1 (8.90e-3)[-]	3.34e-1 (2.15e-4)[-]	3.21e-1 (1.12e-2)[-]	4.42e-1 (1.04e-2) [-]	2.65e-1 (1.40e-2)
+/-/-	-	0/0/4	0/0/4	0/0/4	0/0/4	-

Symbols "+", "=", and "-" indicate performance relative to the proposed method; the best and second-best results are shown in bold and italic, respectively

Table 3 Comparison results on the generated datasets

Algorithm	Dataset					Statistic	
	LPR-GD1	LPR-GD2	LPR-GD3	LPR-GD4	LPR-GD5	Mean	SD
MAS-KCL	26.99	26.13	27.30	29.19	28.85	27.69	1.29
Baseline	32.81	31.00	35.01	33.87	33.30	33.20	1.48
$\Delta loss$	$\downarrow 5.82$	$\downarrow 4.87$	$\downarrow 7.71$	$\downarrow 4.68$	$\downarrow 4.45$	$\downarrow 5.51$	$\downarrow 0.19$

We compare the baseline with the proposed MAS-KCL on five datasets (LPR-GD1 to LPR-GD5) in terms of *loss* (%). The symbol \downarrow indicates the percentage reduction in *loss* achieved by MAS-KCL compared to the corresponding baseline. For each method, the mean and standard deviation are reported

increases, as seen in the LPR-RWD dataset. On average, the use of GPT models reduces the *loss* by 5.79% compared to other LLMs, demonstrating their strong compatibility with the MAS-KCL framework.

Based on these observations, we suggest the following: GPT-3.5 should be prioritized for small-scale datasets; GPT-4.0 is more suitable for large-scale scenarios to achieve optimal results. When GPT models are not available, LLaMA serves as a viable alternative, offering the lowest standard

deviation among all models, which indicates better stability-albeit at the cost of some performance. These findings offer valuable insights into selecting LLMs for integration with MAS-KCL in different application contexts.

4.5 Ablation experiments

To assess the contribution of the LLM-based multi-agent system to algorithm performance, we conducted an abla-

Table 4 The *loss* results of MAS-KCL integrated with different versions of LLMs on real-world datasets

LLM	Dataset				Statistic	
	MOOCubeX-Math	LPR-RWD	LPR-RWD1	LPR-RWD2	Mean	SD
GPT-4.0	25.12	33.08	22.31	26.47	26.75	4.57
GPT-3.5	19.58	33.55	23.14	26.23	25.63	5.94
Claude	48.49	33.91	25.44	30.16	34.50	9.95
LLaMA	27.99	33.93	25.88	30.04	29.46	3.43
$\Delta loss$	$\downarrow 15.89$	$\downarrow 0.61$	$\downarrow 2.94$	$\downarrow 3.75$	$\downarrow 5.79$	$\downarrow 1.21$

For each dataset, the best-performing and second-best methods are highlighted in bold and italics, respectively. The table also reports the mean and standard deviation for each method, with consistent formatting applied. The symbol \downarrow indicates the average *loss* reduction achieved by GPT models (GPT-4.0 and GPT-3.5) compared to other variants

Table 5 Ablation results of MAS-KCL with respect to the entire multi-agent system and each individual agent

Real-world dataset		Algorithm								
Dataset	D	MAS-KCL	(-GA)	MAS-KCL	(-PFA)	MAS-KCL	(-NFA)	MAS-KCL	(-MAS)	MAS-KCL
LPR-RWD	6670	<i>3.39e-1</i>		3.42e-1		3.40e-1		4.43e-1		3.31e-1
LPR-RWD1	1225	2.49e-1		<i>2.46e-1</i>		2.55e-1		3.35e-1		2.23e-1
LPR-RWD2	1225	<i>3.00e-1</i>		3.15e-1		<i>3.15e-1</i>		3.60e-1		2.65e-1
Mean	–	2.96e-1		3.01e-1		3.03e-1		3.79e-1		2.73e-1
$\Delta loss$	–	\uparrow 2.30%		\uparrow 2.80%		\uparrow 3.03%		\uparrow 10.63%		–

Here, (-GA), (-PFA), (-NFA), and (-MAS) denote the removal of the Game Agent, Positive Feedback Agent, Negative Feedback Agent, and the entire LLM-based multi-agent system, respectively. For each dataset, the best-performing and second-best methods are highlighted in bold and italics, respectively. ↑ indicates an increase in *loss* compared to the standard MAS-KCL

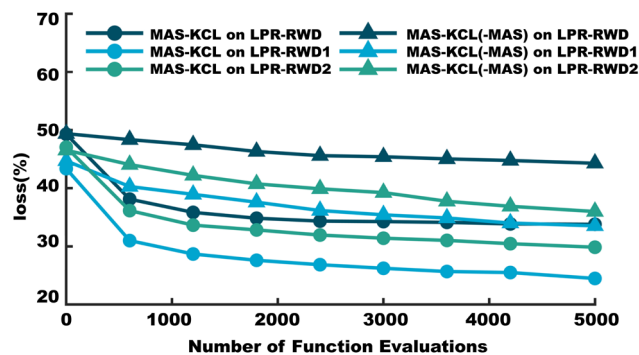


Fig. 4 Results of ablation experiments on the multi-agent system component in MAS-KCL. These results illustrate the variation in *loss* across different real-world educational datasets. Specifically, the comparison algorithm without the multi-agent system component is referred to as MAS-KCL (-MAS), represented by triangles, while the standard MAS-KCL is depicted by circles

tion study by removing the MAS component and evaluating the results on the real-world LPR-RWDs series datasets, as shown in Fig. 4. Through iterative optimization, the standard MAS-KCL consistently achieved significantly lower *loss* across all datasets compared to MAS-KCL(-MAS), which excludes the multi-agent system component. These findings demonstrate that integrating the multi-agent system component improves the efficiency of KC graph structure learning.

To further investigate the impact of each agent, we conducted a more fine-grained ablation study, and the results are presented in Table 5. We observed that the removal of any single agent led to an increase in *loss*, indicating that each agent contributes positively to the KC graph structure learning task. Specifically, removing the entire multi-agent system resulted in a 10.63% increase in *loss*, while removing any single agent caused a rise of approximately 2%–3%. Among all agents, the removal of the Negative Feedback

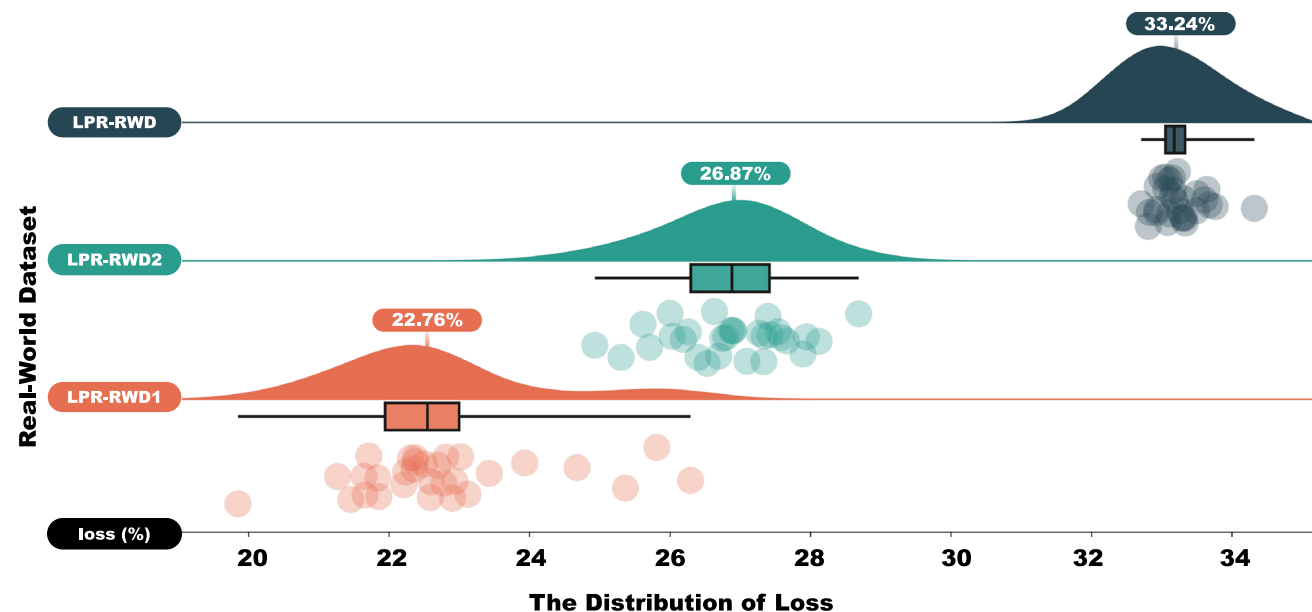


Fig. 5 Convergence analysis, which shows the convergence behavior and loss distribution of MAS-KCL. The centralized distribution of these results validates the strong convergence capability of the proposed method

Agent (NFA) led to the largest increase in *loss*, suggesting that it plays the most critical role in algorithm performance. This is likely due to its ability to identify and eliminate low-quality edges between KCs. The Positive Feedback Agent (PFA) showed the second most significant impact, as it promotes the generation of high-quality edges through positive reinforcement. These findings provide strong evidence for the effectiveness of both the MAS as a whole and each of its individual components.

4.6 Convergence experiment of MAS-KCL

This subsection analyzes the convergence of the proposed MAS-KCL algorithm, which is a key measure of an algorithm's stability and effectiveness. In practical applications, convergence ensures that, after sufficient computation or iteration, the algorithm yields stable and consistent results. The datasets used in the experiments include LPR-RWD, LPR-RWD1, and LPR-RWD2. The results are presented in Fig. 5, which shows the convergence behavior and *loss* distribution of MAS-KCL. These results were derived from 30 independent runs for each dataset. The boxplots below the x-axis illustrate the distribution of the 30 runs, while the median *loss* for each problem is indicated as the central value. The ranking of *loss* from highest to lowest is consistent with earlier results: LPR-RWD, LPR-RWD2, and LPR-RWD1.

For the LPR-RWD, the *loss* distribution is dense, despite the dataset's high dimensionality contributing to its relatively high *loss*. The clear distribution patterns in Fig. 5 demonstrate the stability and convergence of MAS-KCL across different datasets.

5 Conclusion

To improve the efficiency and accuracy of KC graph structure learning and support learners in achieving their learning goals, we propose the Multi-Agent System for Knowledge Component graph structure Learning (MAS-KCL). MAS-KCL incorporates multi-agent collaboration and a bidirectional feedback system to optimize KC graph during iterations by dividing the population based on *loss* values. Experiments demonstrated the proposed algorithm's effectiveness on real datasets. By enabling rapid and accurate KC graph structure learning path, the MAS-KCL algorithm optimizes teaching strategies, enhances learning outcomes, improves the quality of educational development, and ensures sustainable progress in education.

Acknowledgements This work was carried out while Yuan-Hao Jiang was participating in an inter-university joint student program at Shanghai Jiao Tong University. The authors would like to express their special thanks to Xiaobao Shao and Hanglei Hu from East China Normal University for their valuable assistance. It is important to note that the

proposed MAS-KCL is an algorithm enhanced with large language models (LLMs) to improve decision-making capabilities. In this study, four different LLMs were used in the experiments: OpenAI GPT-4.0, OpenAI GPT-3.5, LLaMA-70B, and Claude-3-7-sonnet-20250219. No images in this study were generated using LLMs.

Author Contributions Yuan-Hao Jiang contributed to conceptualization, methodology, visualization, writing-original draft, and project administration. Yang Wei contributed to conceptualization and writing-original draft. Tian-Yi Liu contributed to conceptualization, software, data curation, and writing-original draft. Zi-Wei Chen contributed to methodology, data curation, visualization, and writing-review and editing. Jiayi Wu contributed to data curation and validation. Kezong Tang contributed to project administration, supervision, and writing-review and editing.

Funding This work was supported in part by the Special Foundation for Interdisciplinary Talent Training in "AI Empowered Psychology / Education" of the School of Computer Science and Technology, East China Normal University (2024JCRC-03), and the Doctoral Research and Innovation Foundation of the School of Computer Science and Technology, East China Normal University (2023KYCX-03), and the National Innovative Training Program for College Students of Jingdezhen Ceramic University (202410408028), and the Jiangxi Provincial Innovative Training Program for College Students of Jingdezhen Ceramic University (S202410408033).

Data Availability All datasets used in this paper are publicly available and can be accessed through open channels upon request. Detailed descriptions of all datasets and their access links can be found in the Educational Datasets Section of this paper.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

- Allègre, O., Yessad, A., Luengo, V.: Discovering prerequisite relationships between knowledge components from an interpretable learner model. In: Proceedings of the 16th International Conference on Educational Data Mining, pp. 490–496 (2023)
- Berrett, B., Murphy, J., Sullivan, J.: Administrator insights and reflections: technology integration in schools. *Qual. Rep.* **17**(1), 200–221 (2012)
- Carmona, C., Millán, E., Pérez-de-la Cruz, J.L., Trella, M., Conejo, R.: Introducing prerequisite relations in a multi-layered bayesian student model. In: User Modeling 2005: 10th International Conference, UM 2005, Edinburgh, Scotland, UK, July 24–29, 2005. Proceedings 10, pp. 347–356. Springer (2005)
- Chen, X., Yao, Y., Guan, Z., Li, C., Guan, J., Pu, J., Liu, R., Sheng, B., Yin, S., Qin, Y.: Dsts-gf: a dual-stream temporal-spatial transformer with gated fusion for the classification of obstructive sleep apnea. *The Visual Computer* pp. 1–13 (2025)
- Conati, C., Gertner, A., Vanlehn, K.: Using Bayesian networks to manage uncertainty in student modeling. *User Model. User-Adap. Inter.* **12**, 371–417 (2002)
- Desmarais, M.C., Meshkinfam, P., Gagnon, M.: Learned student models with item to item knowledge structures. *User Model. User-Adap. Inter.* **16**, 403–434 (2006)

7. Feng, Y.: Introduction to causal graphs for education researchers. *Asia Pac. Educ. Rev.* **25**(3), 595–609 (2024). <https://doi.org/10.1007/s12564-024-09980-3>
8. Gong, W., Pawlowski, N., Smith, D., Wang, Z., Jennings, J., Barton, C., Woodhead, S., Zhang, C.: Instructions and guide: Causal insights for learning paths in education (2022). <https://eedi.com/projects/neurips-2022>
9. Gong, W., Pawlowski, N., Smith, D., Wang, Z., Jennings, J., Barton, C., Woodhead, S., Zhang, C.: Neurips 2022 causalml challenge: causal insights for learning paths in education (2022). <https://codalab.lisn.upsaclay.fr/competitions/5626>
10. He, X., Fang, J., Cheng, H.N.H., Men, Q., Li, Y.: Investigating online learners' knowledge structure patterns by concept maps: a clustering analysis approach. *Educ. Inf. Technol.* **28**(9), 11401–11422 (2023). <https://doi.org/10.1007/s10639-023-11633-8>
11. Heidari, A.A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., Chen, H.: Harris hawks optimization: algorithm and applications. *Futur. Gener. Comput. Syst.* **97**, 849–872 (2019). <https://doi.org/10.1016/j.future.2019.02.028>
12. Huang, S., Liu, X., Tan, T., Hu, M., Wei, X., Chen, T., Sheng, B.: Transmrsr: transformer-based self-distilled generative prior for brain MRI super-resolution. *Vis. Comput.* **39**(8), 3647–3659 (2023)
13. Jiang, Y.H., Li, R., Wei, Y., Jia, R., Shao, X., Hu, H., Jiang, B.: Synchronizing Verbal Responses and Board Writing for Multimodal Math Instruction with LLMs. In: The 4th Workshop on Mathematical Reasoning and AI at NeurIPS'24, pp. 46–59. Neural Information Processing Systems Foundation, Vancouver, Canada (2024). <https://openreview.net/forum?id=esblrV8N12>
14. Jiang, Y.H., Li, R., Zhou, Y., Qi, C., Hu, H., Wei, Y., Jiang, B., Wu, Y.: AI agent for education: von Neumann multi-agent system framework. Global Chinese Conference on Computers in Education, Chongqing, China (2024). <https://doi.org/10.48550/arXiv.2501.00083>
15. Käser, T., Klingler, S., Schwing, A.G., Gross, M.: Dynamic Bayesian networks for student modeling. *IEEE Trans. Learn. Technol.* **10**(4), 450–462 (2017)
16. Li, R.: An artificial intelligence agent technology based web distance education system. *J. Intell. Fuzzy Syst.* **40**(2), 3289–3299 (2021). <https://doi.org/10.3233/JIFS-189369>
17. Li, R., Jiang, Y.H., Wang, Y., Hu, H., Jiang, B.: A Large Language Model-Enabled Solution for the Automatic Generation of Situated Multiple-Choice Math Questions. In: Conference Proceedings of the 28th Global Chinese Conference on Computers in Education (GCCCE 2024), pp. 130–136. Chongqing, China (2024). http://gccce2024.swu.edu.cn/GCCCE2024_gongzuofanglunwenji2024-06-23A.pdf#page=148
18. Li, S., Wang, H., Pan, R., Mao, M.: Memorypath: a deep reinforcement learning framework for incorporating memory component into knowledge graph reasoning. *Neurocomputing* **419**, 273–286 (2021)
19. Li, X., Guo, S., Wu, J., Zheng, C.: An interpretable polytomous cognitive diagnosis framework for predicting examinee performance. *Inf. Process. Manag.* **62**(1), 103913 (2025). <https://doi.org/10.1016/j.ipm.2024.103913>
20. Liu, T.Y., Jiang, Y.H., Wei, Y., Wang, X., Huang, S., Dai, L.: Educational practices and algorithmic framework for promoting sustainable development in education by identifying real-world learning paths. *Sustainability* (2024). <https://doi.org/10.3390/su16166871>
21. Meng, L., Shao, Y., Yuan, L., Lai, L., Cheng, P., Li, X., Yu, W., Zhang, W., Lin, X., Zhou, J.: A survey of distributed graph algorithms on massive graphs. *ACM Comput. Surv.* **57**(2), 1–39 (2024)
22. Meng, L., Yuan, L., Chen, Z., Lin, X., Yang, S.: Index-based Structural Clustering on Directed Graphs. In: 2022 IEEE 38th International Conference on Data Engineering (ICDE), pp. 2831–2844 (2022). <https://doi.org/10.1109/ICDE53745.2022.00257>
23. Meng, L., Yuan, L., Lin, X., Li, C., Wang, K., Zhang, W.: Counting Butterflies over Streaming Bipartite Graphs with Duplicate Edges (2024). <https://doi.org/10.48550/arXiv.2412.11488>
24. Mohammadi-Balani, A., Dehghan Nayeri, M., Azar, A., Taghizadeh-Yazdi, M.: Golden eagle optimizer: a nature-inspired metaheuristic algorithm. *Comput. Ind. Eng.* **152**, 107050 (2021). <https://doi.org/10.1016/j.cie.2020.107050>
25. Panichella, A.: An improved pareto front modeling algorithm for large-scale many-objective optimization. In: Proceedings of the Genetic and Evolutionary Computation Conference, pp. 565–573. ACM (2022). <https://doi.org/10.1145/3512290.3528732>
26. Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L.J., Sohl-Dickstein, J.: Deep knowledge tracing. *Advances in neural information processing systems* **28** (2015)
27. Senior, H., Slabaugh, G., Yuan, S., Rossi, L.: Graph neural networks in vision-language image understanding: a survey. *Vis. Comput.* **41**(1), 491–516 (2025)
28. Sheng, B., Keane, P.A., Tham, Y.C., Wong, T.Y.: Synthetic data boosts medical foundation models. *Nat. Biomed. Eng.* **9**(4), 443–444 (2025)
29. Shi, J., Zhao, J., Wu, X., Xu, R., Jiang, Y.H., He, L.: Mitigating reasoning hallucination through multi-agent collaborative filtering. *Expert Syst. Appl.* **263**(2025), 125723 (2025). <https://doi.org/10.1016/j.eswa.2024.125723>
30. Shokouhinejad, H., Razavi-Far, R., Higgins, G., Ghorbani, A.A.: Node-centric pruning: a novel graph reduction approach. *Mach. Learn. Knowl. Extract.* **6**(4), 2722–2737 (2024)
31. Spector, J.M.: A methodology for assessing learning in complex and ill-Structured task domains. *Innov. Educ. Teach. Int.* **43**(2), 109–120 (2024). <https://doi.org/10.1080/14703290600650368>
32. Sun, L., Yan, J., Qiu, Y., Qin, W.: The crowd cooperation approach for formation maintenance and collision avoidance using multi-agent deep reinforcement learning. *Vis. Comput.* **41**(6), 4081–4095 (2025)
33. Tian, Y., Cheng, R., Zhang, X., Jin, Y.: Platemo: a matlab platform for evolutionary multi-objective optimization [educational forum]. *IEEE Comput. Intell. Mag.* **12**(4), 73–87 (2017). <https://doi.org/10.1109/MCI.2017.2742868>
34. Tian, Y., He, C., Cheng, R., Zhang, X.: A multistage evolutionary algorithm for better diversity preservation in multiobjective optimization. *IEEE Trans. Syst. Man Cybernet. Syst.* **51**(9), 5880–5894 (2021). <https://doi.org/10.1109/TSMC.2019.2956288>
35. Wadoud, J., Liu, N., Sandmann, A., Neuhaus, B.J.: The effect of knowledge linking levels in biology lessons upon students' knowledge structure. *Int. J. Sci. Math. Educ.* **12**(1), 25–47 (2014). <https://doi.org/10.1007/s10763-012-9390-8>
36. Wang, J., Wang, J., Li, H., Huang, X., Xia, J., Li, Z., Wu, W., Sheng, B.: Temporal goal-aware transformer assisted visual reinforcement learning for virtual table tennis agent. *Vis. Comput.* pp. 1–15 (2025)
37. Wang, W., Tu, Y., Song, L., Zheng, J., Wang, T.: An adaptive design for item parameter online estimation and Q-matrix online calibration in CD-CAT. *Front. Psychol.* (2021). <https://doi.org/10.3389/fpsyg.2021.710497>
38. Wei, Y., Zhou, Y., Jiang, Y.H., Jiang, B.: Enhancing explainability of knowledge learning paths: causal knowledge networks. In: Joint Proceedings of the Human-Centric eXplainable AI in Education and the Leveraging Large Language Models for Next Generation Educational Technologies Workshops (HEXED-L3MNGT 2024) co-located with 17th International Conference on Educational Data Mining (EDM 2024), vol. 3840, pp. 9–17. International Educational Data Mining Society, Atlanta, Georgia, USA (2024). https://ceur-ws.org/Vol-3840/HEXED24_paper2.pdf
39. Weidlich, J., Hicks, B., Drachler, H.: Causal reasoning with causal graphs in educational technology research. *Educ. Tech. Res. Dev.* **72**(5), 2499–2517 (2024). <https://doi.org/10.1007/s11423-023-10241-0>

40. Wen, Q., Liang, J., Sierra, C., Luckin, R., Tong, R., Liu, Z., Cui, P., Tang, J.: AI for education (AI4edu): Advancing personalized education with LLM and adaptive learning. In: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '24, pp. 6743–6744. Association for Computing Machinery (2024). <https://doi.org/10.1145/3637528.3671498>
41. Yin, Y.H., Shen, L.C., Jiang, Y.H., Gao, S., Song, J., Yu, D.J.: Improving the prediction of DNA-protein binding by integrating multi-scale dense convolutional network with fault-tolerant coding. *Anal. Biochem.* **656**, 114878 (2022). <https://doi.org/10.1016/j.ab.2022.114878>
42. Yu, J., Wang, Y., Zhong, Q., Luo, G., Mao, Y., Sun, K., Feng, W., Xu, W., Cao, S., Zeng, K.: MooccubeX (2021). <https://ifs.aminer.cn/misc/moocdata/data/mooccube2/prerequisites/math.json>
43. Yu, J., Wang, Y., Zhong, Q., Luo, G., Mao, Y., Sun, K., Feng, W., Xu, W., Cao, S., Zeng, K.: MOOCcubeX: a large knowledge-centered repository for adaptive learning in MOOCs. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management, pp. 4643–4652 (2021)
44. Yildiz, E.P., Çengel, M., Alkan, A.: Current trends in education technologies research worldwide: Meta-analysis of studies between 2015–2020. *World J. Educ. Technol.* (2020). <http://acikerisim.subu.edu.tr/xmlui/handle/20.500.14002/486>
45. Zhou, Y., Chen, X., Li, T., Lin, S., Sheng, B., Liu, R., Dai, R.: Gamnet: a gated attention mechanism network for grading myopic traction maculopathy in oct images. *Vis. Comput.* **41**(2), 1097–1108 (2025)
46. Zhuang, X., Wu, H., Shen, X., Yu, P., Yi, G., Chen, X., Hu, T., Chen, Y., Ren, Y., Zhang, Y., Song, Y., Liu, B., Lan, M.: TOREE: Evaluating Topic Relevance of Student Essays for Chinese Primary and Middle School Education. In: L.W. Ku, A. Martins, V. Srikumar (eds.) Findings of the Association for Computational Linguistics ACL 2024, pp. 5749–5765. Association for Computational Linguistics, Bangkok, Thailand and virtual meeting (2024). <https://aclanthology.org/2024.findings-acl.342>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

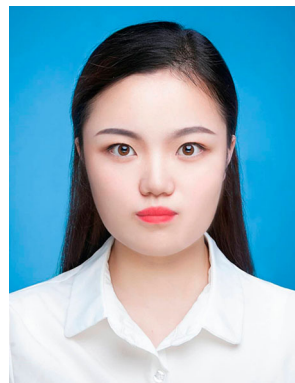


Yuan-Hao Jiang is currently a Ph.D. candidate at the Shanghai Institute of Artificial Intelligence for Education, East China Normal University, Shanghai, China. He received his Master's degree in 2023 from Jiangsu University of Science and Technology, Zhenjiang, China. He is a member of the Association for Computing Machinery (ACM), SIGCHI. He also served as an editor for the monograph *Enhancing Educational Practices: Strategies for Assessing and Improving Learning Outcomes*. His current research interests include AI for education, agentic workflow, and large language model.



chapters and over 30 technical papers. His research interests include genetic algorithms, ant colony system, pattern recognition and image processing.

Kezong Tang received the Ph.D. degree in control theory and control engineering from the Nanjing University of Science and Technology, in 2012. From 2014 to 2016, he was a Postdoctoral Fellow with the School of Computer Science and Engineering, Southeast University, Nanjing, China. Since 2017, he has been with the Jingdezhen Ceramic University, Jingdezhen, China, where he is currently a Professor in the Department of Computer Science. He is the author of two research book



Zi-Wei Chen is currently pursuing her M.S. degree in Statistics at the School of Information Engineering, Jingdezhen Ceramic University, Jingdezhen, China. She has authored and co-authored several papers in international academic conferences and journals, and has contributed to academic book chapters and software copyrights. Her research interests include evolutionary algorithms and their engineering applications.



Yuang Wei will soon graduate with a Ph.D. from the College of Computer Science and Technology at East China Normal University (ECNU) in Shanghai, China, with a specialization in AI for Education. Under the supervision of Professor Bo Jiang, his research focuses on AI for Education and Explainable AI (XAI). To date, he has published over 15 academic papers. He also serves as a reviewer for journals such as *Education and Information Technologies*, *Information Processing and Management*, and *Applied Intelligence*, among others, as well as for major conferences including NeurIPS, AAAI, KDD, ICASSP, AIED, and EDM.

Yuang Wei will soon graduate with a Ph.D. from the College of Computer Science and Technology at East China Normal University (ECNU) in Shanghai, China, with a specialization in AI for Education. Under the supervision of Professor Bo Jiang, his research focuses on AI for Education and Explainable AI (XAI). To date, he has published over 15 academic papers. He also serves as a reviewer for journals such as *Education and Information Technologies*, *Information Processing and Management*, and *Applied Intelligence*, among others, as well as for major conferences including NeurIPS, AAAI, KDD, ICASSP, AIED, and EDM.



Tian-Yi Liu is pursuing his M.S. degree at Jiangsu University of Science and Technology. He received his B.S. degree from Zhejiang International Studies University in 2023. He is leading a Postgraduate Research and Practice Innovation Program funded by Jiangsu Province. He has published several academic papers indexed by SCI, SSCI, and EI. His research interests include evolutionary algorithms, machine learning, and their engineering applications.



Jiayi Wu is a Master's student in the Department of Educational Information Technology at East China Normal University. Her research focuses on artificial intelligence education for K-12 students, with an emphasis on AI literacy assessment and the design of human-computer interaction in AI-supported learning environments. She is committed to exploring effective strategies for integrating AI into school education, enhancing students' AI competencies, and developing measurable indicators for AI literacy. Her work aims to bridge the gap between AI technology and educational practice, contributing to the advancement of innovative AI-based learning tools and assessment frameworks.