

Article

CADKT: A Cognitive-Affective Dual-Graph Contrastive Framework for Dynamic Affect-Aware Knowledge Tracing

Hongwei Wang, Liqing Qiu * and Mingxiang He

College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China; wanghongwei011224@163.com (H.W.); hmx_2026@163.com (M.H.)

* Correspondence: liqing_qiu426@163.com

Abstract

Knowledge tracing (KT) predicts students' future performance by modeling their knowledge mastery from historical learning interactions. Although recent deep learning-based KT models have achieved promising results, most of them mainly rely on response sequences or cognitive structures, leaving affect-related learning states insufficiently explored. In real learning scenarios, student performance is shaped not only by cognitive mastery but also by affect-related factors that change during the learning process. To address this issue, this paper proposes a Cognitive-Affective Dual-Graph Knowledge Tracing framework, termed CADKT, which jointly models cognitive structure and behavior-derived affect-related dynamics. Specifically, CADKT first extracts affect-related behavioral factors from learning logs, including correctness discrepancy, time efficiency discrepancy, and learning persistence deviation. These factors are then used to infer interval-level behavior-derived affect-related profiles without relying on explicit emotion annotations, and the inferred profiles are organized by a Temporal Affective Graph to model their transitions along the learning sequence. Meanwhile, a Heterogeneous Knowledge Graph is constructed to represent student–problem interactions and problem–concept associations. To integrate the two views, CADKT introduces cross-view contrastive regularization and a Projected Hybrid Softmax fusion mechanism for next-response prediction. Experiments on four public educational datasets show that CADKT achieves stronger overall performance than representative KT baselines on most evaluation metrics. Further ablation studies, profile analysis, and case visualization demonstrate that CADKT improves prediction performance while providing interpretable evidence for the interaction between knowledge mastery and behavior-derived affect-related dynamics.

Keywords: knowledge tracing; student modeling; educational data mining; graph neural networks; temporal modeling; contrastive learning



Academic Editor: Fernando De la Prieta Pintado

Received: 27 June 2026

Revised: 23 July 2026

Accepted: 28 July 2026

Published: 3 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](#)

[Attribution \(CC BY\)](#) license.

1. Introduction

Knowledge Tracing (KT) is a key technology in Intelligent Tutoring Systems (ITSs) [1,2]. It estimates a student's knowledge state from historical learning records and predicts future performance on problems [3,4]. By tracking students' mastery of specific Knowledge Concepts (KCs), KT provides support for personalized learning path planning [5], real-time instructional intervention, and adaptive content recommendation [6]. For this reason, KT has become an important research topic in both educational data mining and online education practice.

With the development of deep learning, Deep Learning-Based Knowledge Tracing (DLKT) models have shown stronger representation and prediction abilities than traditional methods. Among them, Deep Knowledge Tracing (DKT) [7] is one of the most representative models. However, DKT models learning mainly through exercise-response sequences, and the resulting knowledge state changes may sometimes be too sharp to match the relatively stable and progressive nature of human learning [8]. To improve this limitation, many later studies introduced more advanced architectures into KT. Examples include memory networks such as DKVMN [9] for explicit state storage, attention mechanisms [10] for sequence dependency modeling, and Graph Neural Networks [11,12] for capturing structural relations among students, problems, and KCs. In addition, textual information [13], problem difficulty [14], and multi-scale knowledge states have also been incorporated into DLKT. These studies have improved KT from several aspects. Still, the main research line remains centered on the cognitive side of learning.

However, learning is not only a cognitive process. Non-cognitive factors also play an important role [15,16]. Educational psychology, especially Achievement Emotion Theory and Control-Value Theory (CVT) [17,18], suggests that cognition and affect interact throughout the learning process. As shown in Figure 1, student performance is shaped by both knowledge level and affect-related learning condition. Different affect-related conditions may lead to different outcomes even when students have similar cognitive ability. For example, s1 remains focused and shows performance beyond the current knowledge state. In contrast, s2 shows a drop in performance because adverse learning conditions affect knowledge retrieval. This relation is also bidirectional. For s3, strong mastery supports a positive learning condition and leads to more stable performance. For s4, weak mastery induces adverse learning conditions and further limits learning progress. These cases indicate that KT should not only model knowledge mastery but also account for affect-related changes during learning.

Knowledge state	Student	Exercise and response				Prediction	Exercise	Knowledge concept
	s1						e1	K ₁
	s2						e2	K ₂
	s3						e3	K ₃
	s4						e4	K ₄ , K ₅
							e5	K ₁ K ₃
		t ₁	t ₂	t ₃	t ₄			

Figure 1. Comparison of student performance under different affect-related learning conditions.

Affect-related learning conditions influence knowledge mastery in both indirect and direct ways [17]. On the one hand, they reflect the quality of learning interactions and affect the long-term development of knowledge states. Favorable learning conditions can help students absorb and consolidate knowledge more effectively. On the other hand, affect-related conditions also influence the retrieval and application of existing knowledge during problem solving. Adverse learning conditions may distract attention, increase uncertainty, or weaken the flexible use of mastered knowledge, which directly affects response accuracy. Therefore, effective KT requires a proper treatment of the interaction between cognition and affect-related dynamics.

Nevertheless, incorporating affect-related information into KT remains challenging. Affective states are usually hidden in educational logs, and most standard educational datasets do not provide explicit emotion labels. Traditional affect assessment often relies on manual annotation, questionnaires, physiological signals, facial expressions, or multimodal devices, which makes large-scale application difficult. Therefore, affect-related representations need to be inferred from non-affect-oriented learning logs. Behavioral signals in learning logs, such as correctness, response time, and repeated attempts, can provide useful clues about students' learning conditions. These signals may reflect different aspects of the learning process: low correctness may be related to insufficient mastery or unstable problem solving, longer response time may indicate careful reasoning or hesitation, and repeated attempts may suggest persistence, perceived difficulty, or frustration-related behavior. Therefore, in this study, they are used as behavior-derived affect-related proxies for constructing latent affective profiles, rather than as explicitly observed emotion labels.

In addition, these behavior-derived signals are often discrete and local, whereas students' affect-related learning conditions may fluctuate continuously along the learning process. Treating them as isolated auxiliary features cannot sufficiently describe their temporal evolution or contextual dependence. A single response may not fully characterize a student's learning condition, while a sequence of responses with low correctness, longer response time, or repeated attempts may reveal a more stable behavior-derived pattern. Moreover, cognitive structure and affect-related dynamics are heterogeneous in nature. The relations among students, problems, and KCs provide structural evidence for cognitive modeling, while behavior-derived affective signals provide temporal evidence for learning-condition modeling. How to derive reliable latent affective profiles from ordinary behavioral logs, how to model their temporal variation, and how to integrate them with cognitive structure remain important issues for affect-aware KT.

To address these challenges, this paper proposes a Cognitive-Affective Dual-Graph Knowledge Tracing framework, termed CADKT. The framework models learning as a joint process of cognitive development and affect-related change. First, to derive affect-related signals from non-affect-oriented logs, a behavioral deviation-based feature extraction scheme was designed. It uses correctness discrepancy, time-efficiency discrepancy, and repeated-attempt behaviors to measure the relative deviation of individual learning behavior from group norms. These signals are not treated as direct emotion labels but as behavior-derived proxies for constructing latent affective profiles. Second, a Temporal Affective Graph (TAG) was constructed to model the variation in latent affective profiles across interactions. Unlike methods that treat affect-related information as isolated input features, TAG explicitly represents the contextual dependence and temporal evolution of affect-related learning conditions. Third, a Heterogeneous Knowledge Graph (HKG) was built to describe the relations among students, problems, and KCs. On this basis, a cross-view contrastive learning strategy is introduced to align cognitive and affective representations, and a Projected Hybrid Softmax (PHS) fusion mechanism is further introduced for multimodal fusion. Through this design, cognitive reasoning becomes affect-aware, and affective modeling is grounded in observed learning behavior and learning progress.

The main contributions of this paper are as follows:

1. A dual-graph KT framework, CADKT, is proposed to jointly model cognitive structure and behavior-derived affective dynamics in a unified architecture.
2. A behavioral deviation-based method was designed to derive affect-related proxy signals from non-affect-oriented learning logs, and latent affective profiles were constructed without relying on explicit emotion annotations.

3. A Temporal Affective Graph was constructed to model the temporal variation of latent affective profiles, while a Heterogeneous Knowledge Graph was used to capture the structural relations among students, problems, and KCs.
4. A collaborative mechanism that combines cross-view contrastive learning and Projected Hybrid Softmax fusion was developed to improve the alignment and fusion of cognitive and affective information.
5. Experiments on four real-world educational datasets show that CADKT achieves strong overall performance compared with representative KT baselines, while also providing interpretable evidence for the relation between knowledge mastery and behavior-derived latent affect-related profiles.

2. Related Work

This section reviews related work from three perspectives: knowledge tracing, affective factors and behavior-derived learning profiles, and contrastive learning for knowledge tracing. Since the proposed framework jointly models cognitive structure and behavior-derived temporal dynamics, it is necessary to situate the present study at the intersection of these three research lines.

Beyond intelligent education, recent artificial intelligence research has increasingly emphasized model transferability, structured representation, and robustness. Conditional distribution-guided adversarial transfer learning reduces discrepancies between multiple source domains and a target domain through conditional distribution alignment [19]. Spatial-channel collaborative multi-scale graph interaction deep transfer learning strengthens multi-granularity feature interaction by jointly exploiting spatial, channel, and graph-structured information [20]. Fixed-time adaptive smooth control with relaxed performance constraints has also been developed to maintain stable system behavior under aerodynamic interference and actuator fault conditions [21]. Although these studies address domains different from knowledge tracing, they reflect broader developments toward transferable, structure-aware, and robust artificial intelligence methods.

2.1. Knowledge Tracing

Knowledge Tracing (KT) aims to model the evolution of a student's knowledge state from historical learning interactions. Early approaches, such as Bayesian Knowledge Tracing (BKT) [22], were based on the Hidden Markov framework and relied on a small number of predefined parameters to describe mastery transition, guess, and slip processes. Although such models are interpretable and easy to deploy, their representation capacity is limited when student learning behaviors become highly complex and heterogeneous.

With the rapid development of deep learning, Deep Learning-Based Knowledge Tracing (DLKT) has gradually become the dominant paradigm because of its stronger representation ability and predictive performance. Existing DLKT methods can be broadly divided into three categories.

- (1) **Memory-network-based methods.** These methods introduce external memory structures to explicitly represent concepts and student mastery states. For example, DKVMN uses a static key matrix to represent knowledge concepts and a dynamic value matrix to track mastery updates [9]. Deep-IRT further combines memory networks with Item Response Theory to jointly model student ability and problem difficulty [23].
- (2) **Sequence-modeling-based methods.** Since student learning is naturally sequential, many studies formulate KT as a temporal sequence modeling problem. DKT was the first representative model to use recurrent neural networks to capture the evolution of student states [7]. Later, SAKT [3] and AKT [10] introduced attention mechanisms to model long-range dependencies and forgetting effects. QIKT [24] further

improves performance by modeling the interaction between problem information and knowledge states. More recently, ANT-KT combined Transformer-based sequence modeling with adaptive neural architecture search to balance prediction performance and computational efficiency [25].

- (3) **Graph-based methods.** Pure sequence models may overlook structural relations among students, problems, and knowledge concepts. Graph-based KT methods therefore use graph neural networks to capture concept interactions and other structural dependencies. GKT models the propagation of mastery among concepts through graph neural networks [12]. SKT [26] exploits structured concept dependencies, while DGEKT introduces a dual-graph design to enhance knowledge representation [27]. More recently, HSKT [28] constructs heterogeneous graphs involving students, exercises, and concepts, and integrates multiple learning behaviors into graph-based prediction. DyGAS combines dynamically updated graph representations with interaction sequences [29], whereas GPPKT integrates a programming-domain knowledge graph with personalized answer sequences [30].

These studies demonstrate the value of temporal and structural information for KT. However, most existing models still focus mainly on the cognitive side of learning, relying on correctness sequences, exercise attributes, or concept relations, while temporal variations in students' behavioral learning conditions receive less attention.

2.2. Affective Factors and Behavior-Derived Learning Profiles

Learning is not only a cognitive process. Educational psychology has long suggested that affective factors influence attention allocation, learning efficiency, self-regulation, and performance outcomes. Achievement-related affect has been widely discussed from the perspectives of control-value theory, self-efficacy theory, and other cognitive-affective frameworks [17,31,32]. Positive conditions, such as confidence, concentration, and engagement, may support learning, whereas anxiety, boredom, frustration, or confusion may hinder cognitive processing [33–35].

Affective state recognition has therefore attracted increasing attention in intelligent education. Traditional affect assessment relies on questionnaires, classroom observation, or self-reported scales, whereas recent studies have used physiological signals, facial expressions, speech, text, and multimodal information. Although these methods can provide relatively direct affective evidence, they often require additional devices or manual annotation and are difficult to apply to large-scale educational datasets containing only interaction logs.

To reduce dependence on explicit affect labels, recent studies have attempted to infer affect-related learning conditions from observable behaviors. Correctness, response time, repeated attempts, and response intervals may reflect learning efficiency, persistence, hesitation, or engagement. However, these signals may also be influenced by prior knowledge, item difficulty, interface design, and other contextual factors. They should therefore be interpreted as evidence for latent cognitive-behavioral profiles rather than direct observations of students' true emotional states.

Several affect-aware KT methods have been proposed along this direction. Psy-KT incorporates behavior-based psychological features into a multi-task framework [36]. DEKT models the interaction between knowledge states and emotional states and predicts emotional representations from personalized historical states [37]. DASKT derives affect-related factors from non-affect-oriented behavioral data and models their dynamic evolution before integrating them into KT [38]. These studies show that behavior-derived information can provide complementary evidence beyond purely cognitive modeling.

However, existing approaches often treat behavior-derived information as auxiliary input, model its temporal dependence insufficiently, or combine it with cognitive representations through shallow fusion. CADKT instead constructs behavior-derived affect-related profiles, models their temporal variation through a Temporal Affective Graph (TAG), and aligns them with cognitive representations learned from a Heterogeneous Knowledge Graph (HKG). These profiles are treated as latent behavioral representations rather than ground-truth emotional labels.

2.3. Contrastive Learning for Knowledge Tracing

Contrastive Learning (CL) is a self-supervised approach that learns representations by pulling positive pairs closer and pushing negative pairs apart [39]. Graph Contrastive Learning extends this principle to graph-structured data and improves representation quality and robustness [40]. Since educational data involve both sequential interactions and structured concept relations, contrastive learning has also been introduced into KT.

Existing contrastive-learning-based KT studies mainly follow two directions.

- (1) **Intra-sequence contrastive methods.** These methods generate multiple views from the same learning sequence and encourage consistency between them. CL4KT constructs augmented views from students' interaction sequences and aligns them to improve sequence representation learning [41]. More recently, Zhang et al. [42] proposed a question-centric multi-experts contrastive framework that models knowledge acquisition at both question and concept levels.
- (2) **Cross-view or structure-aware contrastive methods.** These methods perform contrastive learning on different graph or structural views. HyperKT constructs adaptive hypergraph views and aligns them with corresponding line-graph views to capture higher-order relations and multi-granularity knowledge states [43]. A recent dual-encoder contrastive KT model learns complementary representations from student interaction data [44]. Related dual-graph contrastive learning has also been explored in knowledge-aware recommendation [45].

Despite these advances, most existing contrastive KT methods focus on augmented sequences or closely related graph views, while explicit alignment between cognitive structures and temporal behavior-derived profiles remains underexplored. CADKT therefore applies cross-view contrastive regularization between HKG-based cognitive representations and TAG-based temporal profiles, encouraging consistency for the same student while preserving discrimination across students.

3. Preliminaries

This section introduces the task formulation and the behavior-derived affective factors used in CADKT. Following the idea of affect simulation from learning behaviors [38], affective states are not assumed to be directly observed emotion labels. Instead, they are inferred from students' historical interaction records and used to describe affect-related learning conditions during problem solving. The main notations used in this paper are summarized in Table 1.

Table 1. Description of the key notations.

Notation	Description
S, P, K	Sets of students, problems, and knowledge concepts.
L_s^t	Learning sequence of student s before predicting the response at time step $t + 1$.
p_t, k_t, r_t	Problem, knowledge concept, and response at time step t .
$\hat{p}_{t+1}$	Predicted probability of a correct response at time step $t + 1$.
$\mathcal{I}_{s,k}^t$	Historical interactions of student s on concept k before prediction.

Table 1. Cont.

Notation	Description
$\mathbf{f}_{s,k}^t$	Behavior-derived affective factor vector composed of CD, TED, and LPD.
a_t	Categorical identifier of the latent behavior-derived affect-related profile at time step t .
L_{Seg}	Length of the causal rolling interval used for profile mining.
$\mathcal{G}^k$	Heterogeneous Knowledge Graph.
$\mathcal{G}^a$	Temporal Affective Graph constructed from behavior-derived profiles.
$\mathbf{H}^K, \mathbf{H}^A$	Cognitive-view and behavior-derived affective-view representations.
$\mathbf{x}_t$	Input representation of a temporal profile node.
$\mathbf{z}_t$	Final fused representation at time step t .
λ, τ	Hyperparameters for contrastive learning.

3.1. Problem Formulation

In an Intelligent Tutoring System, let $S = \{s_1, s_2, \dots, s_n\}$, $P = \{p_1, p_2, \dots, p_m\}$, and $K = \{k_1, k_2, \dots, k_q\}$ denote the sets of students, problems, and knowledge concepts, respectively. The response outcome is denoted by $R = \{0, 1\}$, where $r = 1$ indicates a correct response and $r = 0$ indicates an incorrect response. Each problem may be associated with one or more knowledge concepts. For consistency, the first associated concept is used as the target concept of the current interaction.

For student $s \in S$, the learning sequence before predicting the response at time step $t + 1$ is represented as

$$L_s^t = \{(p_1, k_1, r_1), (p_2, k_2, r_2), \dots, (p_t, k_t, r_t)\}, \quad (1)$$

where $p_i \in P$ is the problem attempted at time step i , $k_i \in K$ is the corresponding knowledge concept, and $r_i \in \{0, 1\}$ is the response label.

Given the historical sequence L_s^t and the next problem p_{t+1} with its associated concept k_{t+1} , the objective of KT is to predict the probability that student s will answer the next problem correctly:

$$\hat{r}_{t+1} = P(r_{t+1} = 1 \mid L_s^t, p_{t+1}, k_{t+1}). \quad (2)$$

In addition to response correctness, learning logs usually contain behavioral information such as response time and attempt count. In CADKT, these behavioral signals are used to infer affect-related learning conditions from historical interactions. When predicting r_{t+1} , only the interaction prefix L_s^t is available. The correctness, response time, and attempt count of the target interaction at time step $t + 1$ are not used in affective factor construction.

3.2. Behavior-Derived Affective Factors

In CADKT, affective information is derived from behavioral deviations rather than explicit emotion annotations. The basic intuition is that students may exhibit different affect-related learning conditions when their behaviors deviate from group-level norms on the same knowledge concept. Therefore, correctness, response time, and learning persistence are used to construct three behavior-derived affective factors.

For student s on concept k before predicting time step $t + 1$, the behavior-derived affective factor vector is defined as

$$\mathbf{f}_{s,k}^t = [\text{CD}_{s,k}^t, \text{TED}_{s,k}^t, \text{LPD}_{s,k}^t]^\top, \quad (3)$$

where CD, TED, and LPD denote Correctness Discrepancy, Time Efficiency Discrepancy, and Learning Persistence Deviation, respectively.

Let

$$\mathcal{I}_{s,k}^t = \{i \mid i \leq t, k_i = k\} \quad (4)$$

denote the set of historical interactions of student s on concept k before the target prediction. Student-level statistics are computed from $\mathcal{I}_{s,k}^t$, while the corresponding group-level statistics serve as behavioral reference norms for the same concept.

Correctness Discrepancy (CD). Correctness discrepancy measures the difference between the student's historical accuracy and the group-level accuracy on the same concept:

$$CD_{s,k}^t = A_{s,k}^t - \bar{A}_k, \quad (5)$$

where

$$A_{s,k}^t = \frac{1}{|\mathcal{I}_{s,k}^t|} \sum_{i \in \mathcal{I}_{s,k}^t} r_i. \quad (6)$$

Here, $A_{s,k}^t$ is the historical accuracy of student s on concept k , and $\bar{A}_k$ is the average accuracy on the same concept. A positive value indicates performance above the group norm, whereas a negative value indicates weaker historical performance.

Time Efficiency Discrepancy (TED). Time efficiency discrepancy describes the deviation between the student's response time and the group-level response time on the same concept:

$$TED_{s,k}^t = T_{s,k}^t - \bar{T}_k, \quad (7)$$

where

$$T_{s,k}^t = \frac{1}{|\mathcal{I}_{s,k}^t|} \sum_{i \in \mathcal{I}_{s,k}^t} \text{time}_{s,i}. \quad (8)$$

Here, $\text{time}_{s,i}$ denotes the response time of student s at interaction i , and $\bar{T}_k$ is the average response time on concept k . A larger TED indicates that the student spends more time than the group norm, which may be associated with careful reasoning, hesitation, or inefficient problem solving.

Learning Persistence Deviation (LPD). Learning persistence deviation is calculated from repeated-attempt behavior:

$$LPD_{s,k}^t = N_{s,k}^t - \bar{N}_k, \quad (9)$$

where

$$N_{s,k}^t = \frac{1}{|\mathcal{I}_{s,k}^t|} \sum_{i \in \mathcal{I}_{s,k}^t} \text{attempt}_{s,i}. \quad (10)$$

Here, $\text{attempt}_{s,i}$ denotes the number of attempts made by student s at interaction i , and $\bar{N}_k$ is the average attempt count on concept k . A larger LPD indicates that the student makes more attempts than the group norm. Such deviation may reflect stronger persistence, higher perceived difficulty, unstable problem solving, or frustration-related learning conditions.

The three factors above describe affect-related learning conditions from different behavioral aspects. They are not manually annotated emotion labels but computational affective factors inferred from learning behaviors. In this way, ordinary interaction logs can provide usable affective evidence for subsequent dynamic affect modeling.

3.3. Interval-Based Behavior-Derived Profile Mining

The behavior-derived affective factors provide local evidence of students' learning conditions. However, these learning conditions are not fixed during the learning process. They may fluctuate as students continuously interact with different problems and knowledge concepts. Therefore, an interval-based profile mining procedure is adopted to transform discrete behavioral factors into a sequence of latent behavior-derived affect-related profiles.

For student s on knowledge concept k at time step t , the behavior-derived factor vector is denoted by $\mathbf{f}_{s,k}^t$, which is composed of CD, TED, and LPD. When predicting the response at time step $t + 1$, all components of $\mathbf{f}_{s,k}^t$ are computed only from the historical interaction prefix available up to time step t . The correctness, response time, and attempt count of the target interaction at time step $t + 1$ are not used in profile construction.

First, the behavior-derived factor vectors are normalized to reduce scale differences among CD, TED, and LPD:

$$\tilde{\mathbf{f}}_{s,k}^t = \frac{\mathbf{f}_{s,k}^t - \boldsymbol{\mu}}{\sigma}, \quad (11)$$

where $\boldsymbol{\mu}$ and σ denote the feature-wise mean and standard deviation used for normalization. To prevent cross-split information leakage, both statistics are estimated only from the training portion of the current fold and remain fixed during validation and testing. The concept-level reference statistics used to calculate CD, TED, and LPD follow the same training-only fitting protocol.

Then, K-means clustering is performed on the normalized training-fold factor vectors to obtain K cluster centroids:

$$\mathcal{C} = \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_K\}. \quad (12)$$

The centroids are estimated by minimizing the within-cluster squared distance:

$$\min_{\{\mathbf{c}_q\}_{q=1}^K} \sum_{\tilde{\mathbf{f}} \in \mathcal{F}_{\text{tr}}} \min_{q \in \{1, \dots, K\}} \|\tilde{\mathbf{f}} - \mathbf{c}_q\|_2^2, \quad (13)$$

where $\mathcal{F}_{\text{tr}}$ denotes the normalized behavior-derived factor vectors obtained exclusively from the training fold. In our implementation, K-means is performed with 20 independent initializations and a maximum of 300 iterations. Each centroid corresponds to a latent behavior-derived profile inferred from behavioral evidence rather than a directly observed emotional category.

A single interaction may contain substantial local behavioral fluctuation. Therefore, the normalized factor vectors are aggregated within a causal rolling interval. For time step t , an interval segment with length L_{Seg} is defined as

$$\mathcal{W}_s^t = \{i \mid \max(1, t - L_{\text{Seg}} + 1) \leq i \leq t\}. \quad (14)$$

The interval contains the current and preceding historical positions only and therefore does not introduce information from future interactions.

The normalized behavior-derived factor vectors within the interval are aggregated as

$$\mathbf{u}_s^t = \text{Pool}\left(\left\{\tilde{\mathbf{f}}_{s,k_i}^i \mid i \in \mathcal{W}_s^t\right\}\right), \quad (15)$$

where $\text{Pool}(\cdot)$ denotes mean pooling. Mean pooling reduces the influence of isolated behavioral fluctuations while retaining the recent behavioral context associated with the current interaction.

The latent behavior-derived profile at time step t is assigned according to the nearest-centroid rule:

$$a_t = \arg \min_{q \in \{1, \dots, K\}} \|\mathbf{u}_s^t - \mathbf{c}_q\|_2^2. \quad (16)$$

During validation and testing, the normalization statistics and cluster centroids learned from the training fold remain fixed. Equation (16) is directly applied to each interval representation without refitting the normalization or clustering model.

No manually specified threshold is used to discretize CD, TED, or LPD. Profile inference is entirely data-driven and is determined by the training-derived normalization

parameters, K-means centroids, and the nearest-centroid assignment rule. Moreover, the profile index a_t is treated as a categorical identifier rather than an ordinal behavioral score, and the numerical order of the cluster labels has no direct semantic meaning.

Through this procedure, the original behavioral records are converted into a temporally ordered profile sequence:

$$\mathcal{A}_s = (a_1, a_2, \dots, a_T). \quad (17)$$

The interval length L_{seg} controls the trade-off between local robustness and temporal resolution. A short interval preserves rapid behavioral changes but is more sensitive to isolated fluctuations, whereas an excessively long interval may over-smooth short-term transitions. Accordingly, $L_{\text{seg}} = 5$ is selected according to validation performance because it provides sufficient local smoothing while retaining short-term behavioral variation. The number of profiles is set to $K = 4$ using the same validation-based selection protocol rather than test-set performance. The resulting profile sequence provides the basis for dynamic modeling in the Temporal Affective Graph.

4. Model

This section presents the Cognitive-Affective Dual-Graph Knowledge Tracing (CADKT) framework. As shown in Figure 2, CADKT represents the learning process from two complementary views. The cognitive view describes the structural relations among students, problems, and knowledge concepts through a Heterogeneous Knowledge Graph (HKG), while the affective view models the temporal changes of latent affective states through a Temporal Affective Graph (TAG). The representations obtained from the two views are further integrated by a Projected Hybrid Softmax (PHS) fusion module and used for next-response prediction. A cross-view contrastive regularization term is introduced to encourage the cognitive-view and affective-view representations to provide consistent information for student modeling.

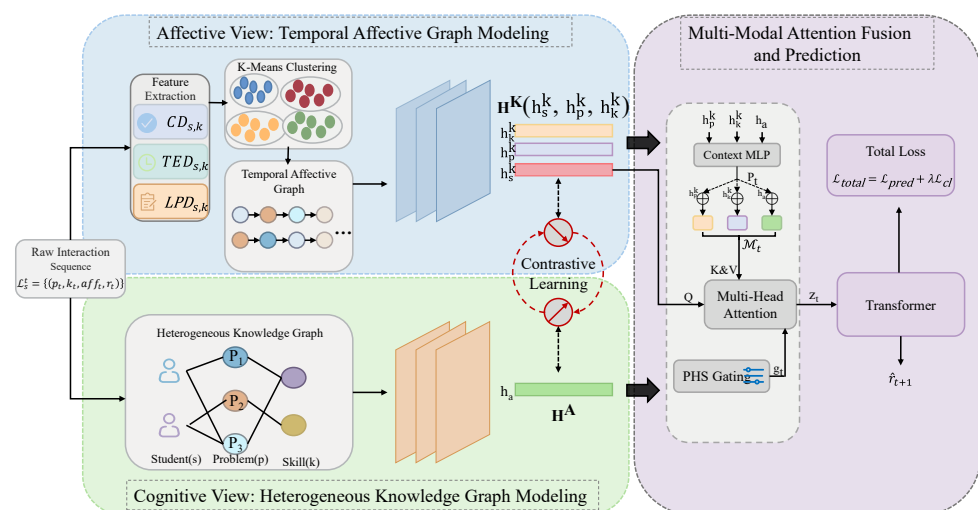


Figure 2. Overview of the proposed CADKT architecture. The framework encodes cognitive relations and behavior-derived affect-related profile transitions through two graph views, integrates the resulting representations through PHS fusion, and predicts the probability of the next response. The ellipsis in the Temporal Affective Graph indicates omitted subsequent interaction nodes in a longer temporal sequence. Colors are used only to visually distinguish different views and components of the framework.

4.1. Cognitive View: Heterogeneous Knowledge Graph Modeling

The cognitive view aims to represent the relational information contained in KT data. A student's interaction sequence records the order of problem-solving activities, but the

sequence itself does not explicitly organize the associations among students, problems, and knowledge concepts. To provide such structural information, CADKT constructs an HKG based on observed student–problem interactions and problem–concept mappings. The HKG is used to obtain cognitive-view representations of students, problems, and concepts.

4.1.1. HKG Construction

The HKG is denoted as

$$\mathcal{G}^k = (\mathcal{V}^k, \mathcal{E}^k, \mathcal{R}^k), \quad (18)$$

where $\mathcal{V}^k$ is the set of heterogeneous nodes, $\mathcal{E}^k$ is the set of observed edges, and $\mathcal{R}^k$ is the set of relation types.

The node set is composed of three types of educational entities and is written as

$$\mathcal{V}^k = \mathcal{V}_s \cup \mathcal{V}_p \cup \mathcal{V}_c. \quad (19)$$

Here, $\mathcal{V}_s$, $\mathcal{V}_p$, and $\mathcal{V}_c$ denote the sets of student nodes, problem nodes, and knowledge concept nodes, respectively. This heterogeneous formulation keeps different entity types in the same graph while allowing their relations to be encoded separately.

The relation types are determined by the available records in KT data. The relation-type set is expressed as

$$\mathcal{R}^k = \{r_{sp}, r_{pc}\}. \quad (20)$$

The relation r_{sp} denotes prefix-observed student–problem interactions. For the prediction at time step $t + 1$, an edge $(s, p_i) \in r_{sp}$ is added only when problem p_i appears in the historical prefix L_s^t . Thus, the target interaction and future interactions are excluded from HKG construction. The relation r_{pc} denotes problem–concept associations. An edge $(p_i, k_i) \in r_{pc}$ indicates that problem p_i is associated with concept k_i according to the problem–concept mapping.

In this graph, student–problem edges provide information about observed learning records, and problem–concept edges provide the connection between exercises and the concepts they assess. The HKG therefore supplies a structural cognitive context for the subsequent representation learning process.

4.1.2. HKG Encoder

A relation-aware graph attention encoder is applied to the HKG to learn node representations. Problem and concept nodes are initialized with learnable embeddings. For student nodes, a shared student-type embedding is used as the initial representation, and individualized student representations are obtained through message passing over prefix-observed student–problem edges. In this way, the student representation used in prediction reflects the observed learning history before the target interaction rather than a static student identity embedding. For a relation type $r \in \mathcal{R}^k$, the embeddings of target node v and its neighbor $u \in \mathcal{N}_r(v)$ are first projected into a relation-specific latent space:

$$\tilde{\mathbf{h}}_{v,r}^{(l-1)} = \mathbf{W}_r^{(l)} \mathbf{h}_v^{(l-1)}, \quad \tilde{\mathbf{h}}_{u,r}^{(l-1)} = \mathbf{W}_r^{(l)} \mathbf{h}_u^{(l-1)}, \quad (21)$$

where $\mathbf{W}_r^{(l)}$ is the learnable transformation matrix for relation r in the l -th layer.

The attention score between node v and node u for relation r is calculated by

$$e_{v,u}^{r,(l)} = \mathbf{a}_r^{(l)\top} \text{LeakyReLU}(\tilde{\mathbf{h}}_{v,r}^{(l-1)} \parallel \tilde{\mathbf{h}}_{u,r}^{(l-1)}), \quad (22)$$

where $\mathbf{a}_r^{(l)}$ is a relation-specific attention vector and $\parallel$ denotes vector concatenation.

The attention coefficient is then obtained by applying softmax over the neighbors connected with the same relation:

$$\alpha_{v,u}^{r,(l)} = \frac{\exp(e_{v,u}^{r,(l)})}{\sum_{j \in \mathcal{N}_r(v)} \exp(e_{v,j}^{r,(l)})}. \quad (23)$$

Based on the attention coefficients, the relation-specific representation of node v is updated as

$$\mathbf{h}_{v,r}^{(l)} = \sum_{u \in \mathcal{N}_r(v)} \alpha_{v,u}^{r,(l)} \tilde{\mathbf{h}}_{u,r}^{(l-1)}. \quad (24)$$

When node v is connected by multiple relation types, the relation-specific representations are combined to obtain the layer-wise node representation:

$$\mathbf{h}_v^{(l)} = \text{ReLU} \left(\frac{1}{|\mathcal{R}_v|} \sum_{r \in \mathcal{R}_v} \mathbf{h}_{v,r}^{(l)} \right), \quad (25)$$

where $\mathcal{R}_v$ denotes the set of relation types associated with node v .

After L graph attention layers, the HKG encoder produces the cognitive-view representations

$$\mathbf{H}^K = \{\mathbf{H}_s^K, \mathbf{H}_p^K, \mathbf{H}_c^K\}. \quad (26)$$

Here, $\mathbf{H}_p^K$ and $\mathbf{H}_c^K$ denote the representations of problems and knowledge concepts, respectively. $\mathbf{H}_s^K$ denotes the prefix-conditioned student representations obtained by aggregating the observed historical interactions before the target prediction. These representations are used in the following cross-view alignment, fusion, and prediction modules.

4.2. Affective View: Temporal Affective Graph Modeling

The affective view is designed to model the temporal variation of the latent behavior-derived profiles obtained in Section 3.3. Different from the HKG, which represents structural relations among educational entities, the Temporal Affective Graph (TAG) focuses on profile changes along each student's chronological interaction sequence.

The profiles used by TAG are inferred from CD, TED, and LPD rather than directly obtained from manually annotated emotion labels. Accordingly, TAG should be interpreted as modeling temporal dependencies among behavior-derived affect-related profiles rather than recognizing ground-truth emotional states. The resulting affective-view representations are subsequently integrated with the cognitive-view representations for next-response prediction.

4.2.1. Behavior-Derived Profile Trajectory and TAG Construction

For a learning sequence with T observed interactions, the TAG is written as

$$\mathcal{G}^a = (\mathcal{V}^a, \mathcal{E}^a), \quad (27)$$

where $\mathcal{V}^a$ denotes the temporal profile node set and $\mathcal{E}^a$ denotes the temporal edge set.

Each historical interaction corresponds to one temporal profile node. The node set is expressed as

$$\mathcal{V}^a = \{v_i \mid i = 1, 2, \dots, T\}, \quad (28)$$

where v_i denotes the node associated with the i -th historical interaction. Therefore, the chronological order of the nodes is consistent with the order of the student's interaction sequence.

The input feature of node v_i is constructed by combining the problem embedding and the embedding of the corresponding behavior-derived profile:

$$\mathbf{x}_i = [\mathbf{e}_{p_i} \parallel \mathbf{e}_{a_i}^a], \quad (29)$$

where $\mathbf{e}_{p_i}$ denotes the learnable embedding of problem p_i , a_i denotes the profile assigned to the i -th interaction, $\mathbf{e}_{a_i}^a$ denotes the learnable embedding of profile a_i , and $\parallel$ denotes vector concatenation. Combining the problem context with the profile embedding enables the same behavior-derived profile to obtain different contextual representations in different problem-solving situations.

The profile index a_i is treated as a categorical identifier rather than a continuous or ordinal behavioral score. Therefore, its numerical value is not directly used to define profile distance, edge existence, or edge weight. Instead, the learnable profile embedding maps each categorical profile into a continuous latent representation suitable for graph encoding.

The temporal edges follow the chronological order of the interaction sequence:

$$\mathcal{E}^a = \{(v_i, v_{i+1}) \mid i = 1, 2, \dots, T-1\}. \quad (30)$$

The ordered pair (v_i, v_{i+1}) represents a directed edge from an earlier interaction to its immediately subsequent interaction. Therefore, information is propagated only along the chronological direction, and reverse future-to-past edges are not introduced.

When predicting the response at time step $t+1$, the representation of node v_t can receive messages only from nodes within the historical prefix $\{v_1, v_2, \dots, v_t\}$. Although the complete sequence graph may be constructed in parallel for efficient computation, the directed edge structure prevents nodes after time step t from propagating information back to v_t . Consequently, neither the target response at time step $t+1$ nor any subsequent interaction can influence the representation used for the current prediction.

The existence of a temporal edge is determined by chronological adjacency rather than by a manually specified profile-similarity threshold. No hand-crafted transition weight is assigned to the edges. All adjacent temporal edges are treated equally during graph construction, whereas their relative contributions are adaptively learned through the attention coefficients of the TAG encoder. This design captures local behavioral continuity and short-term profile transitions without introducing manually defined transition rules.

4.2.2. TAG Encoder

A temporal graph attention encoder is applied to the TAG to obtain affective-view representations. The initial representation of node v_i is given by

$$\mathbf{h}_i^{(0)} = \mathbf{x}_i. \quad (31)$$

For node v_i in the l -th layer, its causal temporal neighborhood is denoted by $\mathcal{N}(v_i)$. With explicit self-loops,

$$\mathcal{N}(v_1) = \{v_1\}, \quad \mathcal{N}(v_i) = \{v_{i-1}, v_i\}, \quad i > 1. \quad (32)$$

Thus, in one TAG layer, node v_i aggregates its current problem–profile context and information from its immediate predecessor. After stacking multiple TAG layers, information from earlier historical interactions can be propagated indirectly through successive causal edges.

For a target node v_i and its temporal neighbor $v_j \in \mathcal{N}(v_i)$, their projected representations in the l -th layer are computed as

$$\tilde{\mathbf{h}}_i^{(l-1)} = \mathbf{W}^{(l)} \mathbf{h}_i^{(l-1)}, \quad \tilde{\mathbf{h}}_j^{(l-1)} = \mathbf{W}^{(l)} \mathbf{h}_j^{(l-1)}, \quad (33)$$

where $\mathbf{W}^{(l)}$ is the learnable transformation matrix of the l -th layer.

The attention score between v_i and v_j is calculated by

$$e_{i,j}^{(l)} = \mathbf{a}^{(l)\top} \text{LeakyReLU}(\tilde{\mathbf{h}}_i^{(l-1)} \parallel \tilde{\mathbf{h}}_j^{(l-1)}), \quad (34)$$

where $\mathbf{a}^{(l)}$ is the learnable attention vector.

The attention coefficient is obtained by applying softmax over the causal temporal neighborhood:

$$\alpha_{i,j}^{(l)} = \frac{\exp(e_{i,j}^{(l)})}{\sum_{q \in \mathcal{N}(v_i)} \exp(e_{i,q}^{(l)})}. \quad (35)$$

The coefficient $\alpha_{i,j}^{(l)}$ determines the contribution of node v_j to the updated representation of node v_i .

With the learned attention coefficients, the hidden representation of node v_i is updated through weighted aggregation:

$$\mathbf{h}_i^{(l)} = \sum_{j \in \mathcal{N}(v_i)} \alpha_{i,j}^{(l)} \tilde{\mathbf{h}}_j^{(l-1)}. \quad (36)$$

For clarity, Equations (34)–(36) present the single-head formulation. In the implementation, multiple attention heads are used in the intermediate TAG layers, and their outputs are concatenated. The final TAG layer uses a single non-concatenated head to produce a d -dimensional output representation.

After stacking L temporal graph attention layers, the TAG encoder produces the affective-view representation sequence

$$\mathbf{H}^A = [\mathbf{h}_1^{(L)}, \mathbf{h}_2^{(L)}, \dots, \mathbf{h}_T^{(L)}] \in \mathbb{R}^{T \times d}. \quad (37)$$

Layer normalization and dropout are applied to the output representations in implementation. The resulting $\mathbf{H}^A$ serves as the affective-view input to the subsequent fusion and prediction modules. Specifically, at time step t , $\mathbf{h}_t^{(L)}$ is combined through the PHS module with the problem, knowledge-concept, and prefix-conditioned student representations obtained from the HKG. The fused representation is then processed by the causal sequence encoder to predict the response at time step $t + 1$.

Therefore, the interval length and clustering configuration affect the final prediction by determining the inferred profile sequence, while the TAG edge structure affects prediction through attention-weighted propagation of historical profile information before cognitive–affective fusion.

4.3. Multimodal Fusion and Prediction

This subsection describes how the cognitive-view and affective-view representations are combined for next-response prediction. At time step t , the problem representation, concept representation, and student representation are obtained from the HKG encoder, while the affective representation is obtained from the TAG encoder. These representations correspond to different sources of information and may have different scales and distributions. Therefore, they are first projected into a shared latent space before fusion.

4.3.1. Projected Hybrid Softmax Fusion

Let $\mathbf{h}_{p_t}^K$, $\mathbf{h}_{c_t}^K$, $\mathbf{h}_{s,t}^K$, and $\mathbf{h}_t^A$ denote the problem, knowledge concept, prefix-conditioned student, and affective representations used at time step t , respectively. They are projected into the same d -dimensional space by

$$\tilde{\mathbf{h}}_p = \mathbf{W}_p \mathbf{h}_{p_t}^K + \mathbf{b}_p, \quad \tilde{\mathbf{h}}_c = \mathbf{W}_c \mathbf{h}_{c_t}^K + \mathbf{b}_c, \quad (38)$$

$$\tilde{\mathbf{h}}_s = \mathbf{W}_s \mathbf{h}_{s,t}^K + \mathbf{b}_s, \quad \tilde{\mathbf{h}}_a = \mathbf{W}_a \mathbf{h}_t^A + \mathbf{b}_a. \quad (39)$$

Here, $\tilde{\mathbf{h}}_p$, $\tilde{\mathbf{h}}_c$, $\tilde{\mathbf{h}}_s$, and $\tilde{\mathbf{h}}_a$ are the projected representations of the problem, concept, student, and affective modalities.

The projected representations are concatenated to form a context vector

$$\mathbf{m}_t = [\tilde{\mathbf{h}}_p \| \tilde{\mathbf{h}}_c \| \tilde{\mathbf{h}}_s \| \tilde{\mathbf{h}}_a]. \quad (40)$$

Based on this context, PHS computes a modality score vector and a context-dependent correction vector. The modality score vector is obtained by

$$\mathbf{s}_t = \mathbf{W}_r \text{ReLU}(\mathbf{W}_q \mathbf{m}_t + \mathbf{b}_q) + \mathbf{b}_r, \quad (41)$$

where $\mathbf{s}_t \in \mathbb{R}^4$ corresponds to the four projected modalities. The correction vector is computed as

$$\mathbf{g}_t = \mathbf{W}_g \mathbf{m}_t + \mathbf{b}_g. \quad (42)$$

The final modality weights are calculated by applying softmax to the sum of the modality scores and the correction vector:

$$\boldsymbol{\alpha}_t = \text{Softmax}(\mathbf{s}_t + \mathbf{g}_t). \quad (43)$$

The weighted-sum branch is then obtained by

$$\mathbf{z}_t^{attn} = \alpha_{t,p} \tilde{\mathbf{h}}_p + \alpha_{t,c} \tilde{\mathbf{h}}_c + \alpha_{t,a} \tilde{\mathbf{h}}_a + \alpha_{t,s} \tilde{\mathbf{h}}_s, \quad (44)$$

where $\alpha_{t,p}$, $\alpha_{t,c}$, $\alpha_{t,a}$, and $\alpha_{t,s}$ are the weights assigned to the problem, concept, affective, and student representations.

In parallel, the projection branch maps the concatenated multimodal context into the same latent space:

$$\mathbf{z}_t^{proj} = \mathbf{W}_z \mathbf{m}_t + \mathbf{b}_z. \quad (45)$$

The final fused representation is calculated as

$$\mathbf{z}_t = \text{Dropout}\left(\text{ReLU}\left(\mathbf{z}_t^{proj} + \mathbf{z}_t^{attn}\right)\right). \quad (46)$$

The PHS module combines a weighted-sum branch and a projection branch. The weighted-sum branch assigns different weights to the four modalities in the current context, while the projection branch retains the joint information from their concatenated representation. The fused representation $\mathbf{z}_t$ is then used as the input of the sequence prediction module.

4.3.2. Sequence Modeling and Prediction

At time step t , the fused representation $\mathbf{z}_t$ is combined with the observed response label r_t . The sequence input is defined as

$$\mathbf{x}_t^{seq} = \mathbf{W}_{seq} [\mathbf{z}_t \| r_t] + \mathbf{b}_{seq}. \quad (47)$$

Here, r_t is the response observed at the current interaction. It is used to update the learning sequence representation for predicting the next response, and it is not the target label to be predicted.

The sequence input is fed into a sequence encoder to model the temporal dependency in the learning trajectory. In the current implementation, a Transformer encoder is used as the backbone:

$$\mathbf{o}_{1:T} = \text{Transformer}(\mathbf{x}_{1:T}^{seq}). \quad (48)$$

Let $\mathbf{o}_t$ denote the encoder output at time step t . The predicted probability of a correct response at time step $t + 1$ is computed by

$$\hat{r}_{t+1} = \sigma(\mathbf{W}_{pred}\mathbf{o}_t + \mathbf{b}_{pred}), \quad (49)$$

where $\sigma(\cdot)$ denotes the Sigmoid activation function. This formulation follows the next-step KT setting, where the representation after observing the interaction at time step t is used to predict the response at time step $t + 1$.

4.4. Optimization

The training objective of CADKT consists of the primary response prediction loss and an auxiliary cross-view contrastive regularization term. The prediction loss supervises next-response prediction, while the contrastive term constrains the cognitive-view and affective-view representations in a shared latent space.

4.4.1. Cross-View Contrastive Regularization

The cognitive view and the affective view describe the same learning process from different perspectives. To reduce the representation gap between them, a symmetric contrastive objective is adopted. For the i -th learning instance in a mini-batch, the cognitive representation $\mathbf{h}_i^K$ is obtained from the HKG-based student representation, and the affective representation $\mathbf{h}_i^A$ is obtained by pooling the valid TAG outputs along the temporal dimension:

$$\mathbf{h}_i^A = \text{Pool}(\mathbf{H}_i^A). \quad (50)$$

Both representations are projected into the same latent space before contrastive learning:

$$\mathbf{q}_i^K = \text{MLP}_K(\mathbf{h}_i^K), \quad \mathbf{q}_i^A = \text{MLP}_A(\mathbf{h}_i^A). \quad (51)$$

Following the InfoNCE formulation [39], the cognitive and affective representations of the same learning instance are treated as a positive pair, while unmatched pairs in the same mini-batch are treated as negative pairs. The symmetric contrastive loss is defined as

$$\mathcal{L}_{cl} = \frac{1}{2}(\mathcal{L}_{K \rightarrow A} + \mathcal{L}_{A \rightarrow K}). \quad (52)$$

The loss from the cognitive view to the affective view is given by

$$\mathcal{L}_{K \rightarrow A} = -\frac{1}{N} \sum_{i=1}^N \log \frac{\exp(\text{sim}(\mathbf{q}_i^K, \mathbf{q}_i^A)/\tau)}{\sum_{j=1}^N \exp(\text{sim}(\mathbf{q}_i^K, \mathbf{q}_j^A)/\tau)}, \quad (53)$$

and the reverse-direction loss is defined as

$$\mathcal{L}_{A \rightarrow K} = -\frac{1}{N} \sum_{i=1}^N \log \frac{\exp(\text{sim}(\mathbf{q}_i^A, \mathbf{q}_i^K)/\tau)}{\sum_{j=1}^N \exp(\text{sim}(\mathbf{q}_i^A, \mathbf{q}_j^K)/\tau)}. \quad (54)$$

Here, N is the batch size, $\text{sim}(\cdot)$ denotes cosine similarity, and τ is the temperature hyperparameter.

This regularization does not replace the prediction objective. It only serves as an auxiliary constraint that encourages the two views of the same learning instance to be close in the shared space while separating unmatched instances in the mini-batch.

4.4.2. Objective Function

The final training objective is defined as

$$\mathcal{L}_{total} = \mathcal{L}_{pred} + \lambda \mathcal{L}_{cl}, \quad (55)$$

where λ controls the contribution of the contrastive regularization term.

The primary prediction loss is the binary cross-entropy over valid next-step predictions:

$$\mathcal{L}_{pred} = -\frac{1}{|\mathcal{M}|} \sum_{t \in \mathcal{M}} [r_{t+1} \log(\hat{r}_{t+1}) + (1 - r_{t+1}) \log(1 - \hat{r}_{t+1})], \quad (56)$$

where $\mathcal{M}$ denotes the set of valid time steps used for next-response prediction. In this loss, the label r_{t+1} supervises the prediction generated from the sequence representation at time step t , which is consistent with the temporal order of KT.

The complete training procedure is summarized in Algorithm 1.

Algorithm 1 Training procedure of CADKT

Require: Interaction logs $\mathcal{L}$, HKG $\mathcal{G}^k$, latent affective states $\{a_t\}$, hyperparameters λ and τ

Ensure: Optimized parameters Θ

```

1: Initialize model parameters  $\Theta$ 
2: repeat
3:   for each mini-batch  $\mathcal{B}$  sampled from  $\mathcal{L}$  do
4:     Encode  $\mathcal{G}^k$  to obtain cognitive-view representations  $\mathbf{H}^K$ 
5:     Construct TAGs from  $\{a_t\}$  and obtain affective-view representations  $\mathbf{H}^A$ 
6:     Fuse problem, concept, student, and affective representations through PHS
7:     Encode the fused sequence and predict  $\hat{r}_{t+1}$ 
8:     Compute prediction loss  $\mathcal{L}_{pred}$ 
9:     Compute cross-view contrastive loss  $\mathcal{L}_{cl}$ 
10:    Update  $\Theta$  by minimizing  $\mathcal{L}_{pred} + \lambda \mathcal{L}_{cl}$ 
11:   end for
12: until convergence or the maximum number of epochs is reached
13: return  $\Theta$ 

```

5. Experiments

This section evaluates the proposed CADKT model on four public educational datasets, including ASSIST09, ASSIST12, ASSISTchall and Junyi Academy. All datasets contain student response logs, exercise metadata and knowledge concept annotations and supply the necessary temporal behavioral fields required to construct CD, TED and LPD. The experiments examine the predictive performance of CADKT against representative baselines, the contribution of each core component, the effectiveness of interval-based behavior-derived profile mining, and the influence of key hyperparameters.

5.1. Datasets

Experiments were conducted on four widely used real-world educational datasets, namely, ASSIST09, ASSIST12, ASSISTchall, and Junyi Academy. These datasets were collected from online learning platforms and contain student response records, exercise information, and knowledge concept annotations. They differ in platform, scale, interaction

density, and average sequence length, which makes them suitable for evaluating the model in different learning scenarios. The behavioral fields used for affective factor construction, including response time and attempt count, are also available in these datasets. Table 2 reports the main statistics after preprocessing.

Table 2. Statistics of the four public educational datasets used in the experiments.

Statistic	ASSIST09	ASSIST12	ASSISTchall	Junyi
# Students	4151	29,018	1709	19,882
# Exercises	17,751	53,091	3162	816
# Knowledge Concepts	123	265	102	40
# Response Records	325,637	2,711,813	942,816	3,973,041
Avg. Interactions per Student	78.4	93.5	551.7	199.8

ASSIST09. (<https://sites.google.com/site/assistmentsdata/home/2009-2010-assistment-data> (accessed on 27 July 2026)) ASSIST09 was collected from the ASSISTments platform during the 2009–2010 academic year and mainly contains middle-school mathematics exercises. It includes 325,637 response records from 4151 students and 123 knowledge concepts. Since the average number of interactions per student is relatively small, this dataset is used to evaluate model performance in relatively sparse interaction settings.

ASSIST12. (<https://sites.google.com/site/assistmentsdata/datasets/2012-13-school-data-with-affect> (accessed on 27 July 2026)) ASSIST12 was collected from the 2012–2013 ASSISTments platform and contains a larger number of students, exercises, and response records. It includes 2,711,813 interactions from 29,018 students and 265 knowledge concepts. Compared with ASSIST09, this dataset provides a larger-scale setting for evaluating the predictive ability of KT models.

ASSISTchall. (<https://sites.google.com/view/assistmentsdatamining/dataset> (accessed on 27 July 2026)) ASSISTchall was released for the 2017 Educational Data Mining challenge. It contains 942,816 response records from 1709 students and 102 knowledge concepts. The average number of interactions per student is much larger than that of ASSIST09 and ASSIST12, making it suitable for evaluating model behavior on longer learning sequences.

Junyi Academy. (<https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=1198> (accessed on 27 July 2026)) Junyi Academy contains mathematics learning records collected from an independent online education platform. After preprocessing, it includes 19,882 students, 816 exercises, 40 knowledge concepts, and 3,973,041 chronologically ordered interactions. Its correctness, response-time, attempt-count, and timestamp fields support the construction of CD, TED, and LPD under the same training-only preprocessing protocol used for the ASSISTments datasets. This dataset provides an additional cross-platform setting for evaluating the robustness of CADKT.

5.2. Baselines and Evaluation Metrics

CADKT was compared with twelve representative KT models. These baselines cover classical probabilistic; sequence- and memory-based; attention- and temporal-dynamics-based; graph-based, behavior- or affect-related; and contrastive-learning-based methods. This selection provides a broad comparison with the methodological directions most relevant to CADKT.

The compared methods are BKT [22], DKT [7], DKVMN [9], SAKT [3], GKT [12], AKT [10], QIKT [24], DGEKT [27], Psy-KT [36], DEKT [37], HSKT [28], and Q-MCKT [42]. Among them, BKT is a classical probabilistic KT method based on a Hidden Markov Model. DKT and DKVMN are classical sequence- and memory-based baselines. SAKT, AKT, and QIKT represent attention- or temporal-dynamics-based KT methods, whereas GKT, DGEKT, and HSKT introduce graph structures into knowledge tracing. Psy-KT and

DEKT incorporate behavior- or affect-related information, while Q-MCKT is included as a recent contrastive-learning-based KT model.

BKT [22] models concept mastery as a binary latent state within a Hidden Markov Model and updates the mastery probability according to observed student responses.

DKT [7] uses recurrent neural networks to model the temporal evolution of students' knowledge states.

DKVMN [9] introduces a memory-augmented architecture with a static key matrix and a dynamic value matrix to represent concepts and student mastery.

SAKT [3] applies self-attention to identify historical interactions that are relevant to the target prediction.

GKT [12] models knowledge concepts as graph nodes and propagates information among concepts through graph neural networks.

AKT [10] combines attention mechanisms with temporal decay to model contextualized knowledge states.

QIKT [24] improves KT by modeling interactions between question-level and concept-level information.

DGEKT [27] uses a dual-graph structure to represent static concept relations and dynamic learning transitions.

Psy-KT [36] incorporates behavior-derived psychological features into the knowledge tracing process.

DEKT [37] models knowledge tracing with a double-effect mechanism that considers both cognitive and affective influences.

HSKT [28] constructs a heterogeneous graph involving students, exercises, and concepts to enhance student representation learning.

Q-MCKT [42] employs question-centric multi-expert representations and contrastive learning to model knowledge acquisition at both the question and concept levels.

Following common practice in KT, Area Under the Curve (AUC) and Accuracy (ACC) were used as the main evaluation metrics. AUC evaluates the ranking quality of predicted response probabilities, whereas ACC measures the proportion of correctly classified responses. Higher AUC and ACC values indicate better predictive performance. The final results are reported as mean $\pm$ standard deviation over independent runs.

5.3. Implementation Details

CADKT was implemented in PyTorch 2.4.1 and trained on an NVIDIA RTX 4090 GPU. Records with missing problem identifiers or concept labels were removed. For multi-concept problems, the first concept was retained. Response time and attempt count were used to construct behavior-derived affect-related factors.

All models are evaluated using fixed student-level 5-fold cross-validation. The same fold assignments, preprocessing procedure, maximum sequence length, and evaluation code were used for all methods and random seeds. Within each fold, a validation subset was sampled from the training students for hyperparameter selection, early stopping, and checkpoint selection, while the held-out test fold was used only for final evaluation. Model-specific hyperparameters were selected according to validation AUC.

To prevent cross-split and temporal leakage, concept-level statistics, normalization parameters, and K-means centroids were estimated only from the training portion of each fold and then fixed for validation and testing. When predicting the $(t + 1)$ -th response, CD, TED, LPD, TAGs, and affect-related profiles were constructed solely from L_s^t , without using the correctness, response time, or attempt count of the target interaction. Problem–concept associations in the HKG were obtained from static metadata and contained no student

response information. Validation and test target responses were not used to estimate training statistics or construct behavioral input factors.

The complete 5-fold procedure was repeated using random seeds 1025, 2024, and 3407 while retaining the same fold assignments. For each seed, AUC and ACC were averaged over the five test folds. The final results are reported as the mean $\pm$ standard deviation of the three seed-level averages.

For CADKT, the cognitive embedding dimension, affect-related profile embedding dimension, and sequence hidden dimension were set to 128, 32, and 256, respectively. The HKG and TAG encoders contained two and three graph attention layers, respectively, while the Transformer contained two layers with four attention heads. CADKT was optimized using Adam with a learning rate of 10^{-3} , batch size 64, dropout rate 0.1, and a maximum of 100 epochs. Early stopping was triggered after 10 epochs without improvement in validation AUC. The temperature parameter τ was set to 0.5, and λ was selected using validation performance.

5.4. Experimental Results and Analysis

This subsection reports the experimental results of CADKT and the compared baseline models. The analysis focuses on the overall prediction performance, the effectiveness of the proposed cognitive and behavior-derived affect-related dual-view design, and the consistency between the experimental results and the model architecture.

5.4.1. Overall Performance Comparison

Tables 3 and 4 report the AUC and ACC results of CADKT and twelve representative baseline models under the unified evaluation protocol. For each random seed, the results were first averaged over the five student-level test folds, and the reported mean and standard deviation were then calculated from the three seed-level cross-validated estimates. Bold and underlined values indicate the best and second-best mean performance, respectively. The symbol “–” denotes an unavailable result.

Table 3. Performance comparison on ASSIST09 and ASSIST12. The results are reported as mean $\pm$ standard deviation. The best results are shown in bold, and the second-best results are underlined.

KT Method	ASSIST09		ASSIST12	
	AUC	ACC	AUC	ACC
BKT	0.733 $\pm$ 0.024	0.658 $\pm$ 0.027	0.718 $\pm$ 0.022	0.728 $\pm$ 0.025
DKT	0.746 $\pm$ 0.021	0.669 $\pm$ 0.024	0.729 $\pm$ 0.020	0.736 $\pm$ 0.023
DKVMN	0.769 $\pm$ 0.018	0.725 $\pm$ 0.021	0.722 $\pm$ 0.023	0.733 $\pm$ 0.020
SAKT	0.777 $\pm$ 0.017	0.742 $\pm$ 0.019	0.723 $\pm$ 0.021	0.733 $\pm$ 0.022
GKT	0.773 $\pm$ 0.016	0.735 $\pm$ 0.018	0.735 $\pm$ 0.019	0.713 $\pm$ 0.021
AKT	0.786 $\pm$ 0.015	0.768 $\pm$ 0.017	0.770 $\pm$ 0.014	0.754 $\pm$ 0.016
QIKT	0.787 $\pm$ 0.014	<u>0.781 $\pm$ 0.016</u>	0.748 $\pm$ 0.018	0.741 $\pm$ 0.017
DGEKT	0.766 $\pm$ 0.019	0.720 $\pm$ 0.020	0.765 $\pm$ 0.015	<u>0.758 $\pm$ 0.016</u>
Psy-KT	<u>0.807 $\pm$ 0.012</u>	0.776 $\pm$ 0.014	0.780 $\pm$ 0.013	0.755 $\pm$ 0.015
DEKT	–	–	<u>0.789 $\pm$ 0.012</u>	0.746 $\pm$ 0.017
HSKT	0.798 $\pm$ 0.013	0.780 $\pm$ 0.014	0.778 $\pm$ 0.013	0.758 $\pm$ 0.014
Q-MCKT	0.789 $\pm$ 0.014	0.738 $\pm$ 0.019	0.781 $\pm$ 0.012	0.753 $\pm$ 0.015
CADKT	0.828 $\pm$ 0.010	0.789 $\pm$ 0.011	0.800 $\pm$ 0.009	0.777 $\pm$ 0.010

Note: Rankings are determined using the original unrounded values.

As shown in Tables 3 and 4 and Figure 3, CADKT achieves the highest mean AUC on all four datasets and the highest ACC on ASSIST09, ASSIST12, and Junyi while ranking second on ASSISTchall. Compared with the strongest baseline, its AUC improves by 0.021, 0.011, 0.008, and 0.023 on ASSIST09, ASSIST12, ASSISTchall, and Junyi, respectively. The highest AUC but second-best ACC on ASSISTchall suggests that CADKT provides

stronger probability ranking, whereas fixed-threshold accuracy is more sensitive to class distribution and probability calibration.

Table 4. Performance comparison on ASSISTchall and Junyi. The results are reported as mean $\pm$ standard deviation. The best results are shown in bold, and the second-best results are underlined.

KT Method	ASSISTchall		Junyi	
	AUC	ACC	AUC	ACC
BKT	0.713 $\pm$ 0.026	0.683 $\pm$ 0.028	0.728 $\pm$ 0.023	0.816 $\pm$ 0.021
DKT	0.721 $\pm$ 0.024	0.692 $\pm$ 0.025	0.735 $\pm$ 0.021	0.824 $\pm$ 0.020
DKVMN	0.710 $\pm$ 0.025	0.684 $\pm$ 0.026	0.740 $\pm$ 0.020	0.831 $\pm$ 0.019
SAKT	0.661 $\pm$ 0.028	0.668 $\pm$ 0.027	0.752 $\pm$ 0.018	0.840 $\pm$ 0.017
GKT	0.756 $\pm$ 0.019	0.713 $\pm$ 0.022	0.743 $\pm$ 0.019	0.837 $\pm$ 0.018
AKT	0.749 $\pm$ 0.017	0.707 $\pm$ 0.019	0.746 $\pm$ 0.017	0.837 $\pm$ 0.016
QIKT	0.736 $\pm$ 0.020	0.725 $\pm$ 0.021	0.766 $\pm$ 0.014	0.841 $\pm$ 0.015
DGEKT	0.776 $\pm$ 0.015	0.721 $\pm$ 0.018	0.766 $\pm$ 0.013	0.841 $\pm$ 0.014
Psy-KT	0.794 $\pm$ 0.012	0.743 $\pm$ 0.015	<u>0.779 $\pm$ 0.012</u>	<u>0.851 $\pm$ 0.013</u>
DEKT	<u>0.801 $\pm$ 0.011</u>	0.744 $\pm$ 0.014	0.776 $\pm$ 0.012	0.849 $\pm$ 0.013
HSKT	0.793 $\pm$ 0.012	0.768 $\pm$ 0.013	0.772 $\pm$ 0.013	0.845 $\pm$ 0.014
Q-MCKT	0.795 $\pm$ 0.011	0.746 $\pm$ 0.014	0.777 $\pm$ 0.011	0.850 $\pm$ 0.012
CADKT	0.809 $\pm$ 0.009	<u>0.759 $\pm$ 0.010</u>	0.802 $\pm$ 0.009	0.855 $\pm$ 0.010

Note: Rankings are determined using the original unrounded values.

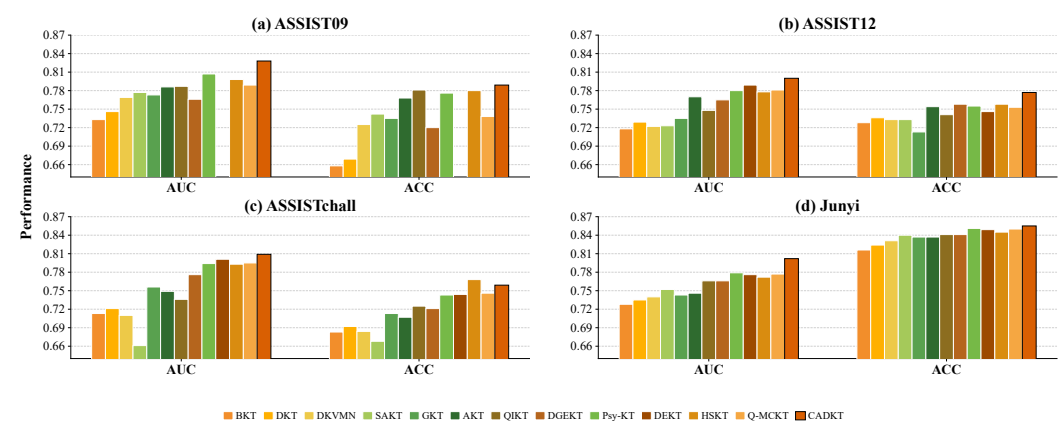


Figure 3. Performance comparison of knowledge tracing models in terms of AUC and ACC. Bars represent the mean of the three seed-level cross-validated estimates, and error bars represent the corresponding standard deviations.

These results support the complementary design of CADKT. Sequential models primarily capture temporal dependencies, graph-based models emphasize structural relations, and behavior-aware models introduce auxiliary learning conditions; however, they generally model these sources separately. CADKT instead combines heterogeneous student–exercise–concept relations through HKG with the temporal evolution of behavior-derived profiles through TAG, while Projected Hybrid Softmax and cross-view contrastive learning coordinate the two representations. Its advantage over Q-MCKT indicates that cognitive–behavioral cross-view alignment provides additional predictive information beyond contrastive learning confined to the cognitive view. On the independent Junyi platform, CADKT obtains an AUC of 0.802 and an ACC of 0.855, and its standard deviations remain between 0.009 and 0.011, supporting its cross-platform robustness and stable optimization across random seeds.

5.4.2. Ablation Study

To isolate the contribution of each component, we conducted controlled component-level, temporal-modeling, fusion-level, and behavioral-factor ablations on the four datasets.

TAG was replaced with an HMM to directly compare the proposed temporal graph encoder with a probabilistic state-transition model, while PHS was replaced with standard Multi-Head Attention. Table 5 reports the corresponding results.

Table 5. Ablation results of CADKT. The reported values were averaged over five fixed student-level test folds using random seed 2024.

Model	ASSIST09		ASSIST12		ASSISTchall		Junyi	
	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC
CADKT (Full)	0.828	0.789	0.800	0.777	0.809	0.759	0.802	0.855
CADKT (TAG → HMM)	0.812	0.775	0.786	0.765	0.792	0.742	0.788	0.840
CADKT (PHS → MHA)	0.816	0.778	0.790	0.769	0.796	0.746	0.791	0.844
CADKT w/o HKG	0.795	0.764	0.765	0.748	0.771	0.721	0.769	0.826
CADKT w/o CD	0.818	0.780	0.791	0.769	0.797	0.748	0.793	0.845
CADKT w/o TED	0.819	0.782	0.793	0.771	0.800	0.751	0.795	0.847
CADKT w/o LPD	0.821	0.784	0.795	0.773	0.802	0.754	0.797	0.849
CADKT w/o CL	0.815	0.779	0.789	0.768	0.793	0.743	0.790	0.842

As shown in Table 5, all ablation and replacement variants underperform the full CADKT model in terms of both AUC and ACC. Figure 4 further visualizes the AUC changes relative to the full model, showing consistent trends across the four datasets.

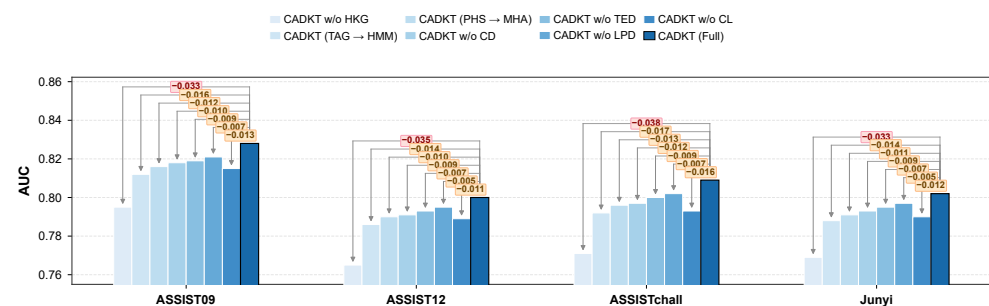


Figure 4. AUC results of the component, behavioral-factor, temporal-modeling, and fusion ablations on four datasets. The annotated values indicate the AUC changes relative to the full model. Red annotations highlight the largest AUC reduction on each dataset, whereas the other reductions are shown in light orange.

Removing HKG causes the largest degradation, with AUC reductions of 0.033–0.038 and ACC reductions of 0.025–0.038. This result highlights the importance of heterogeneous student–exercise–concept relations for knowledge-state estimation. Replacing TAG with an HMM reduces AUC by 0.014–0.017 and ACC by 0.012–0.017, indicating that TAG is more effective than the evaluated HMM variant in modeling the temporal evolution of behavior-derived profiles.

The factor-level results in Figure 4 show a consistent contribution order: CD produces the largest AUC decrease when removed, followed by TED and LPD. This suggests that CD provides the strongest individual signal, while TED and LPD offer complementary information regarding response efficiency and repeated attempts.

Replacing PHS with Multi-Head Attention or removing contrastive learning also leads to consistent performance losses. These results support the effectiveness of PHS for heterogeneous representation fusion and of contrastive learning for coordinating the cognitive and behavior-derived views. Overall, the table provides the complete AUC and ACC results, whereas Figure 4 highlights the relative magnitude and cross-dataset consistency of the AUC changes.

5.4.3. Analysis of Behavior-Derived Affective Profiles

To examine whether the interval-based profile mining process can provide meaningful auxiliary signals for CADKT, the centroid distributions of the learned behavior-derived affect-related profiles were analyzed. Since CADKT is designed for non-affect-oriented educational logs, it does not rely on manually annotated emotion labels. The learned clusters are therefore not regarded as ground-truth emotional categories. Instead, they are interpreted as behavior-derived learning-condition profiles inferred from CD, TED, and LPD.

Consistent with the interval-based profile mining procedure introduced in Section 3.3, the three behavioral factors were normalized before clustering so that CD, TED, and LPD could be compared in the same feature space. In this study, the number of clusters was set to $K = 4$ according to validation performance, corresponding to four behavior-derived affect-related profiles used in the subsequent visualization. The influence of the number of profiles is further examined in the sensitivity analysis in Section 5.4.4. Table 6 reports the centroid values, sample size, average response accuracy, and behavioral interpretation of each cluster. The profile descriptions used in the following analysis serve only as interpretive references for the observed behavioral patterns rather than as directly observed emotional labels.

Table 6. Centroids and characteristics of behavior-derived affect-related profiles.

Cluster	CD (Mean)	TED (Mean)	LPD (Mean)	N (Samples)	ACC	Behavior-Derived Profile
0	−0.82	−0.65	1.12	293,665	0.28	Repeated-attempt low-performance pattern
1	0.95	2.18	−0.73	58,720	0.62	Deliberate successful pattern
2	0.93	−0.41	−0.92	62,174	0.59	Efficient low-effort success pattern
3	−0.78	−0.52	0.89	347,449	0.31	Unstable low-performance pattern

As shown in Table 6, the four clusters present distinguishable behavior-derived affect-related profiles. These profiles are inferred from behavioral deviations rather than manually annotated emotion labels and should therefore be interpreted according to their observable response characteristics.

Cluster 0 has negative CD and the highest LPD, indicating relatively low response performance together with more repeated attempts. This profile is described as a repeated-attempt low-performance pattern. Students assigned to this profile tend to make additional attempts while still achieving comparatively low accuracy, reflecting persistent but unsuccessful problem-solving behavior. Cluster 3 also shows negative CD, but its LPD is lower than that of Cluster 0 and its TED remains negative. This profile therefore represents an unstable low-performance condition characterized by relatively short response time and moderately elevated repeated-attempt behavior.

Clusters 1 and 2 correspond to more favorable response outcomes. Cluster 1 has positive CD and the highest TED, suggesting that students spend more time while obtaining better performance. This profile is described as a deliberate successful pattern because longer response time is accompanied by higher correctness and fewer repeated attempts. Cluster 2 also has positive CD and negative LPD, but its TED is lower than that of Cluster 1. This combination indicates comparatively efficient successful responses with fewer attempts. Therefore, it is described as an efficient low-effort success pattern and is interpreted as a mastery-related behavioral profile rather than as evidence of a specific emotional state.

The sample distribution also provides useful descriptive information. Clusters 0 and 3 contain more samples than Clusters 1 and 2, suggesting that the corresponding low-performance or unstable learning-condition patterns occur frequently in the interaction

records. Meanwhile, the average response accuracy clearly separates the successful profiles from the low-performance profiles. Clusters 1 and 2 achieve average accuracies of 0.62 and 0.59, whereas Clusters 0 and 3 obtain 0.28 and 0.31, respectively. Because correctness-related information contributes to the construction of CD, these accuracy differences should be regarded as descriptive consistency rather than independent validation of the learned profiles.

Overall, CD, TED, and LPD characterize different combinations of response quality, response time, and learning persistence. The learned profiles consequently exhibit descriptive consistency with students' observable response behavior and provide structured auxiliary signals for subsequent temporal modeling. Based on these profiles, TAG further captures how students transition among different learning-condition patterns along the interaction sequence, providing the basis for the subsequent trajectory visualization and case analysis. The predictive contribution of the profile construction is further evaluated through the ablation and sensitivity analyses.

5.4.4. Parameter Sensitivity Analysis

To evaluate the influence of key hyperparameters in CADKT, three representative parameters were selected for sensitivity analysis: the contrastive learning weight λ , the interval length L_{Seg} , and the number of behavior-derived profiles K . These parameters control the cross-view regularization strength, temporal aggregation granularity, and profile clustering granularity, respectively. Other hyperparameters were fixed according to the validation setting described in Section 5.3. The experiments were conducted on ASSIST09, and the results are shown in Figure 5.

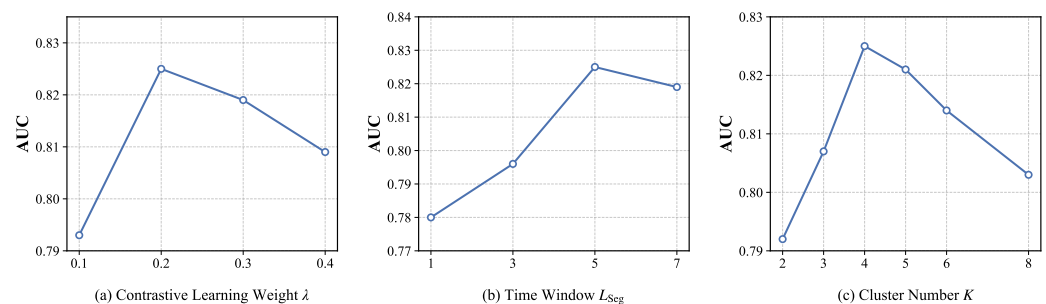


Figure 5. Parameter sensitivity analysis of CADKT. (a) Performance with different contrastive learning weights λ . (b) Performance with different interval lengths L_{Seg} . (c) Performance with different numbers of behavior-derived profiles K .

As shown in Figure 5a, the AUC first increases and then decreases as λ increases, reaching the highest value of 0.825 at $\lambda = 0.2$. A small λ provides insufficient regularization to coordinate the cognitive and affective views, whereas an excessively large value may over-constrain the representation space and interfere with optimization of the primary prediction objective. Therefore, $\lambda = 0.2$ provides an appropriate balance between cross-view consistency and predictive performance.

Figure 5b shows that the AUC increases from 0.780 at $L_{\text{Seg}} = 1$ to 0.825 at $L_{\text{Seg}} = 5$, and then decreases to 0.819 at $L_{\text{Seg}} = 7$. A short interval is more sensitive to isolated behavioral fluctuations, whereas a long interval may over-smooth short-term profile transitions. Since L_{Seg} determines the range of the causal rolling window $\mathcal{W}_s^t$ in Equation (14), the result supports $L_{\text{Seg}} = 5$ as a suitable balance between temporal resolution and local robustness.

As shown in Figure 5c, increasing K from 2 to 4 improves the AUC from 0.792 to 0.825, whereas further increasing K gradually reduces performance. A small K may merge distinct behavior-derived patterns into the same profile, limiting their discriminative capacity. In contrast, an excessively large K may divide the interaction samples into overly fine-grained clusters and reduce the stability of profile assignment. Because the inferred

profiles are subsequently used as categorical inputs to TAG, $K = 4$ provides an appropriate profile granularity for temporal graph modeling.

Overall, the three parameters exhibit intermediate optima, indicating that CADKT benefits from balanced cross-view regularization, temporal aggregation, and profile granularity rather than from maximizing any single modeling component. Therefore, $\lambda = 0.2$, $L_{\text{Seg}} = 5$, and $K = 4$ were adopted in the final model according to validation performance.

5.4.5. Case Study and Interpretability Analysis

To further examine the interpretability of CADKT, a case study was conducted on a representative student trajectory from ASSISTchall. The analysis proceeded from two aspects. First, the mastery trajectories estimated by CADKT and DKT were compared to examine how different models describe the evolution of student knowledge states. Second, the behavior-derived affective profiles inferred by CADKT were analyzed along the same interaction sequence so as to reveal how changes in mastery estimation are associated with behavioral-state transitions.

A single student trajectory with 39 interaction steps was selected for visualization. Figures 6 and 7 show the estimated mastery trajectories of five representative knowledge concepts for CADKT and DKT, respectively. The horizontal axis denotes the interaction sequence, and the vertical axis denotes the selected knowledge concepts. The numerical values and color intensity in each cell indicate the estimated mastery probability.

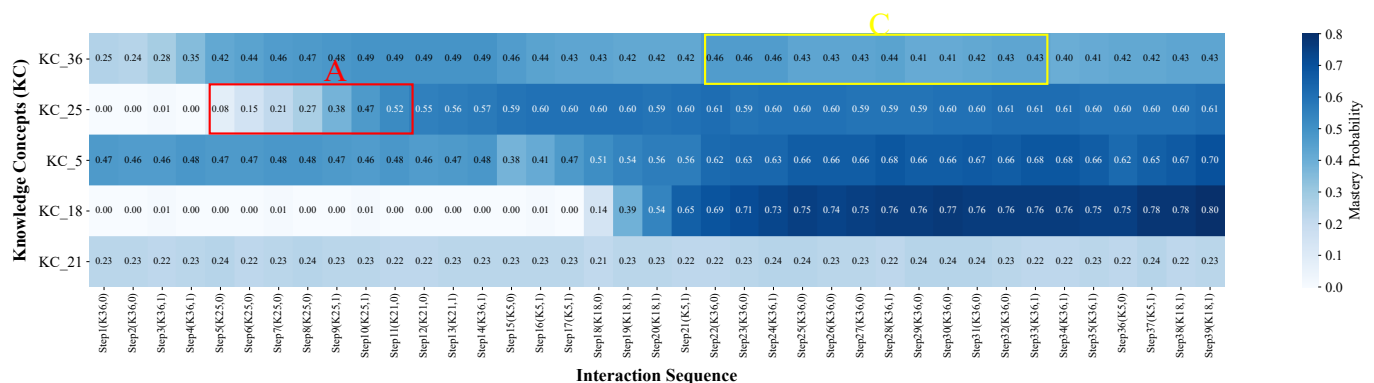


Figure 6. Knowledge-state trajectory estimated by CADKT for a representative student. Regions A and C highlight representative areas with gradual and coherent changes in the estimated mastery states.

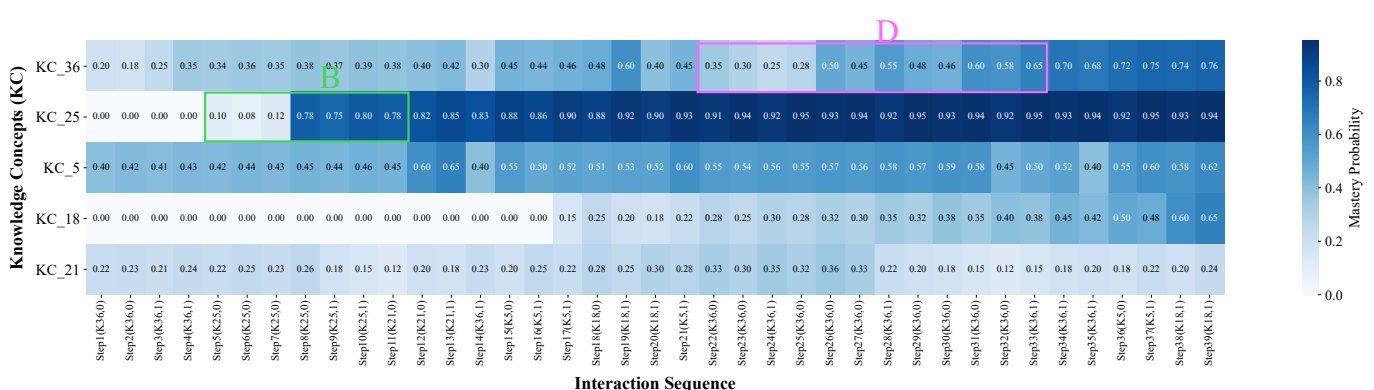


Figure 7. Knowledge-state trajectory estimated by DKT for the same student. Regions B and D highlight representative areas with sharper local fluctuations in the estimated mastery states.

(1) Case study of knowledge state evolution. As shown in Figures 6 and 7, CADKT and DKT present different mastery estimation patterns on the same student trajectory. CADKT produces smoother and more coherent changes across several knowledge concepts.

In Figure 6, the highlighted regions A and C show gradual transitions, indicating that the estimated mastery changes progressively along the interaction sequence. This pattern is consistent with the cumulative nature of knowledge acquisition, where mastery usually develops through continuous learning rather than abrupt changes.

By contrast, DKT shows sharper local fluctuations in regions B and D of Figure 7. These changes suggest that DKT is more sensitive to individual response correctness, and its estimated mastery may fluctuate when isolated responses change. The comparison indicates that CADKT can provide more stable mastery estimation by incorporating cognitive structural information from HKG and behavior-derived state information from TAG. Therefore, the visualization supports the interpretability of CADKT from the perspective of knowledge-state evolution.

(2) Behavior-derived affect-related profiles along the trajectory. To further illustrate the behavioral-state modeling process, Figure 8 visualizes the same 39-step trajectory together with the dominant behavior-derived affect-related profiles inferred by CADKT. The background labels in the figure, including Confusion, Concentration, Frustration, and Boredom, are used as shorthand interpretive references following affect-aware KT studies. They do not represent manually annotated emotion labels or ground-truth emotional categories.

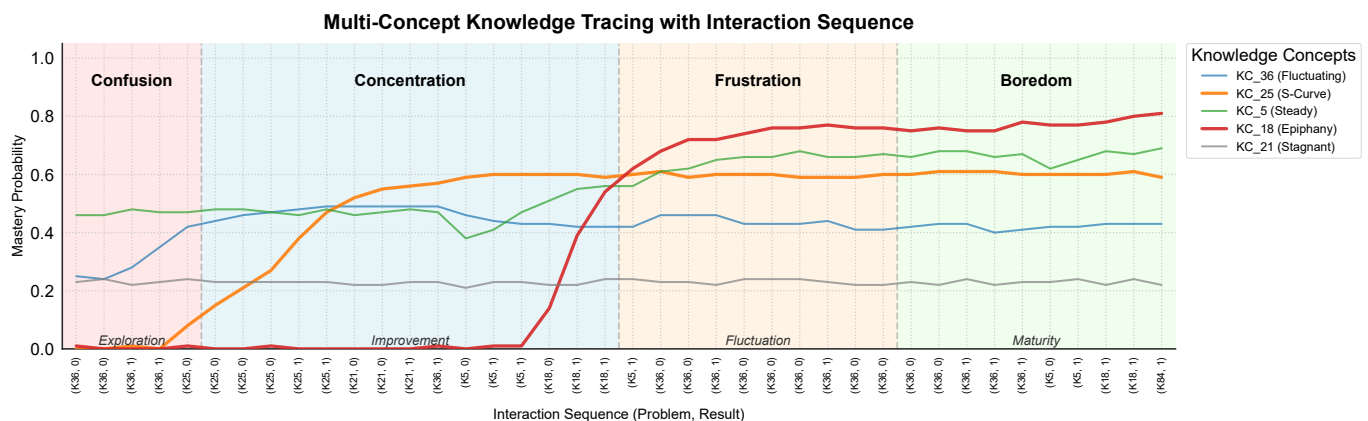


Figure 8. Visualization of mastery trajectories and behavior-derived affect-related profiles across 39 interaction steps, where the colored background regions indicate the dominant behavior-derived learning-condition profiles inferred by CADKT and the affective labels serve only as shorthand interpretive references rather than ground-truth emotion labels.

As shown in Figure 8, the trajectory can be divided into four behavior-derived learning-condition stages. In the region labeled as Confusion, the student is assigned to an unstable low-performance profile. During this period, the mastery estimates are generally low or unstable, suggesting that the student has not yet formed stable mastery of the related knowledge concepts. In the region labeled as Concentration, several mastery trajectories increase gradually, indicating that the student begins to make more effective progress. This stage is consistent with the behavior-derived profile characterized by longer response time and improved performance.

In the region labeled as Frustration, the trajectory shows more noticeable fluctuation. Some concepts continue to improve, while others remain unstable or increase slowly. This pattern is consistent with repeated attempts and unstable learning outcomes, and may be associated with frustration-related learning behavior. In the region labeled as Boredom, the mastery estimates become relatively stable. Since this profile is characterized by relatively high correctness, fewer repeated attempts, and lower behavioral effort, it is more

conservatively interpreted as an efficient low-effort success pattern or mastery-related behavioral profile, rather than as a directly observed boredom state.

Overall, the case study shows that CADKT provides interpretable cognitive-behavioral trajectories. The comparison with DKT indicates that CADKT yields smoother and more coherent mastery estimation, while the behavior-derived affect-related profiles show how learning-state changes are accompanied by behavioral-state transitions. These results further support the role of TAG in modeling behavior-derived affect-related dynamics and explain how CADKT connects cognitive structure with behavioral information in knowledge tracing.

5.4.6. Computational Efficiency Analysis

To evaluate the practical computational cost of CADKT, we measured training latency, inference latency, and peak GPU memory at interaction-history lengths of 50, 100, 200, 300, and 500. All experiments were conducted on an NVIDIA RTX 4090 using FP32 and a fixed batch size of 32. Each setting was evaluated after five warm-up iterations, with training and inference statistics averaged over 20 optimization steps and 50 forward passes, respectively.

As shown in Figure 9, when the history length increases from 50 to 500, training latency rises from 0.641 to 5.621 s/batch and inference latency from 0.309 to 2.970 s/batch. Peak training memory increases from 1.770 to 2.356 GiB, whereas peak inference memory remains approximately 0.491 GiB because no backpropagation activations are retained. These results show predictable resource growth and confirm that CADKT remains executable at a history length of 500 on a single RTX 4090.

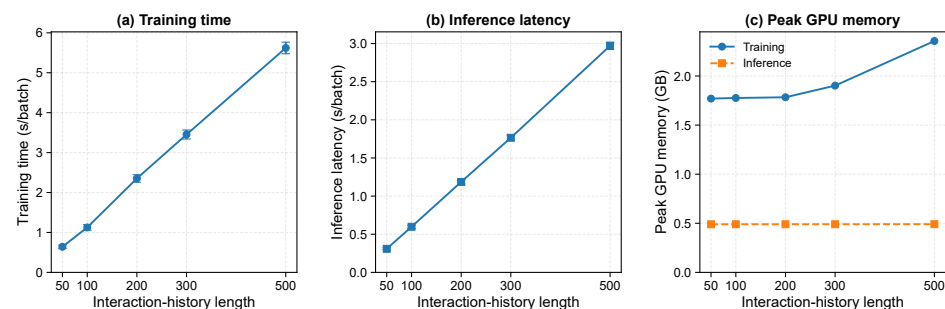


Figure 9. Computational costs of CADKT for different interaction-history lengths, including (a) training latency, (b) inference latency, and (c) peak GPU memory, with error bars denoting standard deviations over repeated measurements.

To further quantify the overhead of dual-graph modeling, Table 7 compares CADKT with representative sequential, attention-based, and graph-based KT models under the same measurement protocol.

Table 7. Computational overhead comparison at history length $L = 500$ with a batch size of 32 on an NVIDIA RTX 4090.

Model	Params	Train (s/Batch)	Infer (s/Batch)	Train Mem. (GiB)	Infer Mem. (GiB)
DKT	742,631	4.326	2.105	1.720	0.410
AKT	826,412	4.813	2.461	2.040	0.450
DGEKT	893,674	5.136	2.615	2.170	0.470
CADKT	970,858	5.621	2.970	2.356	0.491

Compared with DGEKT, CADKT increases the parameter count by 8.64%, training latency by 9.44%, inference latency by 13.58%, peak training memory by 8.57%, and peak inference memory by 4.47%. In return, CADKT improves AUC by 0.0625, 0.0344, and 0.0331

on ASSIST09, ASSIST12, and ASSISTchall, respectively. This indicates that the additional TAG branch provides improved predictive performance with a moderate and measurable computational overhead rather than relying on excessive parameter expansion.

6. Conclusions

In this paper, a Cognitive-Affective Dual-Graph Knowledge Tracing framework, named CADKT, is proposed to improve student performance prediction by jointly modeling cognitive structure and behavior-derived affect-related dynamics. Different from conventional KT methods that mainly rely on response sequences, CADKT constructs a Heterogeneous Knowledge Graph to represent student–problem interactions and problem–concept associations, and further builds a Temporal Affective Graph to model the temporal transitions of behavior-derived affect-related profiles inferred from learning logs. Based on these two views, the Projected Hybrid Softmax fusion mechanism adaptively integrates problem, concept, student, and behavioral-state representations, while cross-view contrastive regularization promotes consistency between cognitive and behavioral representations.

Experiments on four public educational datasets demonstrate the effectiveness of CADKT. The proposed model achieves stronger overall performance than representative sequence-based, graph-based, and affect-aware baselines on most evaluation metrics. The ablation study confirms the contribution of each component, showing that HKG provides cognitive structural evidence, TAG supplements behavior-derived temporal information, and PHS together with contrastive regularization improves cross-view interaction. In addition, the analysis of behavior-derived affect-related profiles and the case study show that CADKT can produce more interpretable learning trajectories by connecting mastery evolution with behavioral-state changes.

Future work may further enrich behavior-derived state representations and explore trajectory-level contrastive alignment between cognitive and behavioral learning processes. More efficient graph construction and fusion strategies can also be investigated to improve the scalability of CADKT in large-scale intelligent tutoring systems.

Author Contributions: Conceptualization, L.Q. and M.H.; methodology, H.W.; software, H.W.; validation, H.W., L.Q. and M.H.; formal analysis, H.W.; investigation, H.W.; resources, L.Q. and M.H.; data curation, H.W.; writing—original draft preparation, H.W.; writing—review and editing, L.Q. and M.H.; visualization, H.W.; supervision, L.Q. and M.H.; project administration, L.Q.; funding acquisition, L.Q. and M.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (Grant No. 72572095) and the Shandong Provincial Natural Science Foundation (Grant No. ZR2023MF070).

Institutional Review Board Statement: Ethical review and approval were waived for this study because it used publicly available and anonymized educational datasets and did not involve direct interaction with human participants or the collection of new personal data.

Informed Consent Statement: Informed consent was waived for this secondary analysis because all datasets used in this study were publicly available and anonymized.

Data Availability Statement: The datasets analyzed in this study are publicly available from the ASSISTments data repositories cited in the manuscript. Additional implementation details and processed data are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Castro-Schez, J.J.; Glez-Morcillo, C.; Albusac, J.; Vallejo, D. An Intelligent Tutoring System for Supporting Active Learning: A Case Study on Predictive Parsing Learning. *Inf. Sci.* **2021**, *544*, 446–468. [\[CrossRef\]](#) [\[PubMed\]](#)
2. Yang, H.; Hu, S.; Geng, J.; Huang, T.; Hu, J.; Zhang, H.; Zhu, Q. Heterogeneous Graph-Based Knowledge Tracing with Spatiotemporal Evolution. *Expert Syst. Appl.* **2024**, *238*, 122249. [\[CrossRef\]](#)
3. Pandey, S.; Karypis, G. A Self-Attentive Model for Knowledge Tracing. *arXiv* **2019**, arXiv:1907.06837. [\[CrossRef\]](#)
4. Jiang, B.; Wei, Y.; Zhang, T.; Zhang, W. Improving the Performance and Explainability of Knowledge Tracing via Markov Blanket. *Inf. Process. Manag.* **2024**, *61*, 103620. [\[CrossRef\]](#)
5. Zhang, F.; Feng, X.; Wang, Y. Personalized Process-Type Learning Path Recommendation Based on Process Mining and Deep Knowledge Tracing. *Knowl.-Based Syst.* **2024**, *303*, 112431. [\[CrossRef\]](#)
6. Wu, Z.; Li, M.; Tang, Y.; Liang, Q. Exercise Recommendation Based on Knowledge Concept Prediction. *Knowl.-Based Syst.* **2020**, *210*, 106481. [\[CrossRef\]](#)
7. Piech, C.; Bassen, J.; Huang, J.; Ganguli, S.; Sahami, M.; Guibas, L.J.; Sohl-Dickstein, J. Deep Knowledge Tracing. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 505–513.
8. Yeung, C.K.; Yeung, D.Y. Addressing Two Problems in Deep Knowledge Tracing via Prediction-Consistent Regularization. In Proceedings of the Fifth Annual ACM Conference on Learning at Scale, London, UK, 26–28 June 2018; pp. 1–10. [\[CrossRef\]](#)
9. Zhang, J.; Shi, X.; King, I.; Yeung, D.Y. Dynamic Key-Value Memory Networks for Knowledge Tracing. In Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 3–7 April 2017; pp. 765–774. [\[CrossRef\]](#)
10. Ghosh, A.; Heffernan, N.; Lan, A.S. Context-Aware Attentive Knowledge Tracing. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Virtual Event, 6–10 July 2020; pp. 2330–2339. [\[CrossRef\]](#)
11. Scarselli, F.; Gori, M.; Tsoi, A.C.; Hagenbuchner, M.; Monfardini, G. The Graph Neural Network Model. *IEEE Trans. Neural Netw.* **2008**, *20*, 61–80. [\[CrossRef\]](#) [\[PubMed\]](#)
12. Nakagawa, H.; Iwasawa, Y.; Matsuo, Y. Graph-Based Knowledge Tracing: Modeling Student Proficiency Using Graph Neural Network. In Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, Thessaloniki, Greece, 14–17 October 2019; pp. 156–163.
13. Liu, Q.; Huang, Z.; Yin, Y.; Chen, E.; Xiong, H.; Su, Y.; Hu, G. EKT: Exercise-Aware Knowledge Tracing for Student Performance Prediction. *IEEE Trans. Knowl. Data Eng.* **2019**, *33*, 100–115. [\[CrossRef\]](#)
14. Shen, S.; Huang, Z.; Liu, Q.; Su, Y.; Wang, S.; Chen, E. Assessing Student’s Dynamic Knowledge State by Exploring the Question Difficulty Effect. In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, 11–15 July 2022; pp. 427–437. [\[CrossRef\]](#)
15. Dwivedi, P.; Kant, V.; Bharadwaj, K.K. Learning Path Recommendation Based on Modified Variable Length Genetic Algorithm. *Educ. Inf. Technol.* **2018**, *23*, 819–836. [\[CrossRef\]](#)
16. Abdelrahman, G.; Wang, Q.; Nunes, B. Knowledge Tracing: A Survey. *ACM Comput. Surv.* **2023**, *55*, 224. [\[CrossRef\]](#)
17. Pekrun, R. The Control-Value Theory of Achievement Emotions: Assumptions, Corollaries, and Implications for Educational Research and Practice. *Educ. Psychol. Rev.* **2006**, *18*, 315–341. [\[CrossRef\]](#)
18. Slavin, R.E.; Madden, N.A.; Ross, S.M. *Educational Psychology: Theory and Practice*; Pearson Education: London, UK, 2025.
19. Wu, Z.; Jiang, H.; Liu, S.; Liu, Y.; Yang, W. Conditional Distribution-Guided Adversarial Transfer Learning Network with Multi-Source Domains for Rolling Bearing Fault Diagnosis. *Adv. Eng. Inform.* **2023**, *56*, 101993. [\[CrossRef\]](#)
20. Wang, X.; Jiang, H.; Dong, Y.; Mu, M. Spatial-Channel Collaborative Multi-Scale Graph Interaction Deep Transfer Learning for Unsupervised Rotating Machinery Fault Diagnosis. *Eng. Appl. Artif. Intell.* **2026**, *176*, 114691. [\[CrossRef\]](#)
21. Zhao, H.; Wang, B.; Fu, Y.; Zhou, M.; Li, N.; Gao, Z. Fixed-Time Adaptive Smooth Control with Relaxed Performance Constraints for a Canard Rotor/Wing UAV Subject to Aerodynamic Interferences and Actuator Faults. *Aerosp. Sci. Technol.* **2026**, *178*, 112959. [\[CrossRef\]](#)
22. Corbett, A.T.; Anderson, J.R. Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. *User Model. User-Adapt. Interact.* **1994**, *4*, 253–278.
23. Yeung, C.K. Deep-IRT: Make Deep Learning Based Knowledge Tracing Explainable Using Item Response Theory. *arXiv* **2019**, arXiv:1904.11738. [\[CrossRef\]](#)
24. Chen, J.; Liu, Z.; Huang, S.; Liu, Q.; Luo, W. Improving Interpretability of Deep Sequential Knowledge Tracing Models with Question-Centric Cognitive Representations. *Proc. AAAI Conf. Artif. Intell.* **2023**, *37*, 14196–14204. [\[CrossRef\]](#)
25. Yao, S.; Song, Y.; Liu, Y.; Chen, J.; Zhao, D.; Wang, X. ANT-KT: Adaptive NAS Transformers for Knowledge Tracing. *Electronics* **2025**, *14*, 4148. [\[CrossRef\]](#)
26. Tong, S.; Liu, Q.; Huang, W.; Huang, Z.; Chen, E.; Liu, C.; Ma, H.; Wang, S. Structure-Based Knowledge Tracing: An Influence Propagation View. In Proceedings of the 2020 IEEE International Conference on Data Mining (ICDM); IEEE: New York, NY, USA, 2020; pp. 541–550. [\[CrossRef\]](#)

27. Cui, C.; Yao, Y.; Zhang, C.; Ma, H.; Ma, Y.; Ren, Z.; Zhang, C.; Ko, J. DGEKT: A Dual Graph Ensemble Learning Method for Knowledge Tracing. *ACM Trans. Inf. Syst.* **2024**, *42*, 78. [[CrossRef](#)]
28. Gao, J.; Ma, L.; Qiu, L. Enhancing Heterogeneous Information Networks Through Student Interactions for Knowledge Tracing. *Inf. Sci.* **2025**, *735*, 123062. [[CrossRef](#)]
29. Li, X.; Yan, Z.; Gu, Y.; Zhou, S.; Yang, S. DyGAS: Dynamic Graph-Augmented Sequence Modeling for Knowledge Tracing. *Appl. Sci.* **2025**, *15*, 12767. [[CrossRef](#)]
30. Pan, J.; Dong, Z.; Yan, L.; Cai, X. Knowledge Graph and Personalized Answer Sequences for Programming Knowledge Tracing. *Appl. Sci.* **2024**, *14*, 7952. [[CrossRef](#)]
31. Bandura, A. *Self-Efficacy: The Exercise of Control*; W. H. Freeman and Company: New York, NY, USA, 1997.
32. Lersbach, A.; Jinks, J. Self-Efficacy Theory and Learning Environment Research. *Learn. Environ. Res.* **1999**, *2*, 157–167. [[CrossRef](#)]
33. Wang, Y.; Derakhshan, A.; Zhang, L.J. Researching and Practicing Positive Psychology in Second/Foreign Language Learning and Teaching: The Past, Current Status and Future Directions. *Front. Psychol.* **2021**, *12*, 731721. [[CrossRef](#)] [[PubMed](#)]
34. Spachtholz, P.; Kuhbandner, C.; Pekrun, R. Negative Affect Improves the Quality of Memories: Trading Capacity for Precision in Sensory and Working Memory. *J. Exp. Psychol. Gen.* **2014**, *143*, 1450. [[CrossRef](#)] [[PubMed](#)]
35. Xie, W.; Ye, C.; Zhang, W. Negative Emotion Reduces Visual Working Memory Recall Variability: A Meta-Analytical Review. *Emotion* **2023**, *23*, 859. [[CrossRef](#)] [[PubMed](#)]
36. Wang, Z.; Wu, W.; Zeng, C.; Luo, H.; Sun, J. Psychological Factors Enhanced Heterogeneous Learning Interactive Graph Knowledge Tracing for Understanding the Learning Process. *Front. Psychol.* **2024**, *15*, 1359199. [[CrossRef](#)] [[PubMed](#)]
37. Wang, S.; Yuan, F.; Wang, K.; Yang, X.; Zhang, X.; Wang, M. Dual-State Personalized Knowledge Tracing with Emotional Incorporation. *IEEE Trans. Knowl. Data Eng.* **2025**, *37*, 2440–2455. [[CrossRef](#)]
38. Sun, X.; Zhang, K.; Liu, Q.; Shen, S.; Wang, F.; Guo, Y.; Chen, E. DASKT: A Dynamic Affect Simulation Method for Knowledge Tracing. *IEEE Trans. Knowl. Data Eng.* **2025**, *37*, 1714–1727. [[CrossRef](#)]
39. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A Simple Framework for Contrastive Learning of Visual Representations. In *Proceedings of the International Conference on Machine Learning*; PMLR: New York, NY, USA, 2020; pp. 1597–1607.
40. Hassani, K.; Khasahmadi, A.H. Contrastive Multi-View Representation Learning on Graphs. In *Proceedings of the International Conference on Machine Learning*; PMLR: New York, NY, USA, 2020; pp. 4116–4126.
41. Lee, W.; Chun, J.; Lee, Y.; Park, K.; Park, S. Contrastive Learning for Knowledge Tracing. In *Proceedings of the ACM Web Conference 2022, Virtual Event, 25–29 April 2022*; pp. 2330–2338. [[CrossRef](#)]
42. Zhang, H.; Liu, Z.; Shang, C.; Li, D.; Jiang, Y. A question-centric multi-experts contrastive learning framework for improving the accuracy and interpretability of deep sequential knowledge tracing models. *ACM Trans. Knowl. Discov. Data* **2025**, *19*, 41. [[CrossRef](#)]
43. Li, J.; Deng, Y.; Qin, Y.; Mao, S.; Jiang, Y. Dual-Channel Adaptive Scale Hypergraph Encoders with Cross-View Contrastive Learning for Knowledge Tracing. *IEEE Trans. Neural Netw. Learn. Syst.* **2025**, *36*, 6752–6766. [[CrossRef](#)] [[PubMed](#)]
44. Bai, Y.; Wu, X.; Wei, T.; He, L. A Dual-Encoder Contrastive Learning Model for Knowledge Tracing. *Entropy* **2025**, *27*, 685. [[CrossRef](#)] [[PubMed](#)]
45. Huang, J.; Xie, Z.; Zhang, H.; Yang, B.; Di, C.; Huang, R. Enhancing Knowledge-Aware Recommendation with Dual-Graph Contrastive Learning. *Information* **2024**, *15*, 534. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.