Endowing Interpretability for Neural Cognitive Diagnosis by Efficient Kolmogorov-Arnold Networks

Shangshang Yang[1], **Linrui Qin**[2], **Xiaoshan Yu**[2], **Ziwen Wang**[1], **Xueming Yan**[3], **Haiping Ma**[4,5*] and **Ye Tian**[1]

[1]School of Computer Science and Technology, Anhui University
[2]School of Artificial Intelligence, Anhui University
[3]Guangdong University of Foreign Studies
[4]Institutes of Physical Science and Information Technology, Anhui University
[5]Information Materials and Intelligent Sensing Laboratory of Anhui Province, Anhui University
{yangshang0308, stonewallqr, yxsleo, wzw12sir, field910921}@gmail.com,
xueming126@126.com , hpma@ahu.edu.cn

Abstract

Cognitive diagnosis is crucial for intelligent education because of its ability to reveal students' proficiency in knowledge concepts. Although neural network-based neural cognitive diagnosis models (CDMs) have exhibited significantly better performance than traditional models, neural cognitive diagnosis is criticized for the poor model interpretability due to the multi-layer perceptron(MLP) employed, even with the monotonicity assumption. Therefore, this paper proposes to empower the interpretability of neural cognitive diagnosis models through efficient Kolmogorov-Arnold networks (KANs), named KAN2CD, where KANs are used to enhance interpretability in two manners. Specifically, in the first manner, KANs are directly used to replace the used MLPs in existing neural CDMs; while in the second manner, the student embedding, exercise embedding, and concept embedding are directly processed by several KANs, and then their outputs are further combined and learned in a unified KAN to get final predictions. Besides, the implementation of original KANs is modified without affecting the interpretability to overcome the problem of training KANs slowly. Extensive experiments show KAN2CD outperforms traditional CDMs and slightly surpasses existing neural CDMs, and its learned structures ensure interpretability on par with traditional CDMs and better than neural CDMs. The datasets, associated code, and more experimental results are available at https://github.com/null233QAQ/KAN2CD.

1 Introduction

In intelligent education, cognitive diagnosis (CD)[Anderson *et al.*, 2014] identifies students' proficiency in knowledge concepts by analyzing their historical exercise records. As illustrated by the example in Fig. 1, the CD process begins with a student's response records, such as (e_1, e_3, e_4) and (e_1, e_2, e_3). To interpret these responses, the process relies on an exercise-concept relational matrix (Q-matrix) that maps each exercise to a set of underlying knowledge concepts. By integrating the information from both the response records and the Q-matrix, a CD model can be built to infer the student's mastery level for each concept. With this diagnosis, online education platforms can support tasks like remedial instruction, learning path recommendations [Nabizadeh *et al.*, 2020], targeted training [Stojanoski *et al.*, 2018], and exercise/course assembly more effectively[Yang *et al.*, 2023; Beck, 2007].

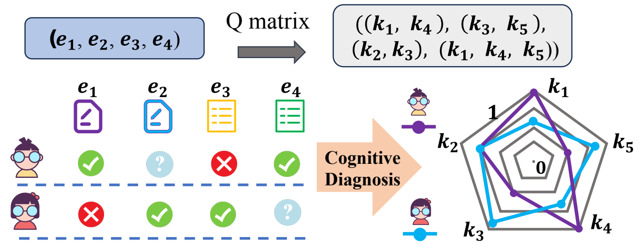

Figure 1: Illustration of the cognitive diagnosis process.

To develop convincing cognitive diagnosis models (CDMs) for meeting the demands of online education platforms (e.g., ASSISTments [Patikorn *et al.*, 2018], PTA [Hu *et al.*, 2023]), massive efforts have been devoted by researchers , mainly from two research perspectives [Wang *et al.*, 2024]. The first perspective is to design completely interpretable CDMs whose components and operations are drawn from educational psychology, so that the users can understand how the diagnosis results are obtained, trusting the results. The representatives include IRT [Embretson and Reise, 2013], DINA [Torre and J., 2009], MIRT [Reckase, 2009], and MF [Koren *et al.*, 2009]. The second perspective is to leverage neural networks (NNs) to model the student's response prediction process, aiming to improve the prediction accuracy and provide more accurate diagnosis results for making subsequent tasks like

[*]Corresponding author

recommendations more convincing [Urdaneta-Ponte *et al.*, 2021]. This type of CDMs is called neural cognitive diagnosis models, and the representatives contain NCD [Wang *et al.*, 2020], KaNCD [Wang *et al.*, 2022], RCD [Gao *et al.*, 2021], and so on [Ma *et al.*, 2022; Yu *et al.*, 2024b; Qian *et al.*, 2024].

Compared to traditional CDMs in educational psychology, neural CDMs achieve better performance, effectively supporting subsequent tasks [Wu *et al.*, 2024; Yu *et al.*, 2024a; Shen *et al.*, 2024]. However, their interpretability is weaker, as they rely on multi-layer perceptrons (MLPs) or fully connected (FC) layers [Fan *et al.*, 2021]. Understanding how these layers process inputs and produce predictions remains challenging [Zhang *et al.*, 2021], even under the monotonicity assumption [Samek *et al.*, 2016], which ensures only positive weights and monotonic output increases [Wang *et al.*, 2020; Zhang *et al.*, ; Zhang *et al.*, 2023]. This limited interpretability undermines their ability to engage users fully.

To this end, this paper aims to build more convincing CDMs by enhancing the interpretability of neural cognitive diagnosis models without sacrificing the accuracy of the diagnosis results. Thus, this paper proposes leveraging efficient Kolmogorov-Arnold networks for neural cognitive diagnosis models (KAN2CD) to empower the model's interpretability and maintain the accuracy of diagnosis. Specifically, our main contributions include:

- This paper is the first to utilize Kolmogorov-Arnold networks (KANs) to enhance the interpretability of neural CDMs while maintaining high diagnostic accuracy. We propose two approaches: replacing MLPs in neural CDMs with KANs for direct interpretability improvement, and designing a novel aggregation framework composed entirely of KANs without relying on neural CDMs.
- In the new aggregation framework, two levels of KANs are used for input processing and prediction. Lower-level KANs handle student, exercise, and concept embeddings separately, while an upper-level two-layer KAN integrates their outputs for final predictions. To address high runtime in the original KAN implementation, modifications were made to accelerate training.
- In experiments, we compared it with representative CDMs on four popular education datasets. The results show KAN2CD outperforms both traditional and neural CDMs in performance. Furthermore, the learned structures of KANs in KAN2CD demonstrate higher interpretability than neural CDMs, comparable to traditional CDMs. Besides, the modified implementation ensures that KAN2CD's training cost remains competitive with existing models.

2 Preliminaries and Related Work

2.1 Preliminaries of Cognitive Diagnosis Task

In the cognitive diagnosis task, we consider N students, M exercises, and K knowledge concepts, represented by the sets $S = \{s_1, s_2, \ldots, s_N\}$, $E = \{e_1, e_2, \ldots, e_M\}$, and $C = \{c_1, c_2, \ldots, c_K\}$, respectively. The platform uses an exercise-concept relation matrix, the Q-matrix, provided by domain experts and denoted as $Q = (Q_{jk} \in \{0,1\})^{M \times K}$. Here, $Q_{jk} = 1$ indicates that exercise e_j involves knowledge concept c_k, and $Q_{jk} = 0$ otherwise.

The platform also maintains a log of students' responses to exercises, recorded as R_{log}. This log contains triplets (s_i, e_j, r_{ij}), where $s_i \in S$, $e_j \in E$, and $r_{ij} \in \{0,1\}$. Here, $r_{ij} = 1$ means student s_i correctly answered exercise e_j, while $r_{ij} = 0$ indicates an incorrect response.

Using R_{log} and the Q-matrix, the cognitive diagnosis task aims to assess students' proficiency in knowledge concepts by developing a model $\mathcal{F}$ to predict students' exercise scores. The model $\mathcal{F}$ uses three input features to predict the score of student s_i on exercise e_j: the student feature vector $\mathbf{h}_S \in \mathbb{R}^{1 \times D}$, the exercise feature vector $\mathbf{h}_E \in \mathbb{R}^{1 \times D}$, and the knowledge concept feature vector $\mathbf{h}_C \in \mathbb{R}^{1 \times K}$. The embeddings for students, exercises, and knowledge concepts are obtained as follows:

$$\begin{aligned} \mathbf{h}_S &= \mathbf{x}_i^S \times W_S, W_S \in \mathbb{R}^{N \times D}, \\ \mathbf{h}_E &= \mathbf{x}_j^E \times W_E, W_E \in \mathbb{R}^{M \times D} \\ \mathbf{x}_j^C &= \mathbf{x}_j^E \times Q = (Q_{j1}, \cdots, Q_{jK}), \\ \mathbf{h}_C &= \mathbf{x}_j^C \times W_Q, W_Q \in \mathbb{R}^{K \times D} \end{aligned}, \tag{1}$$

where D is the embedding dimension (usually equal to K for consistency), $\mathbf{x}_i^S \in \{0,1\}^{1 \times N}$ is the one-hot vector for student s_i, $\mathbf{x}_j^E \in \{0,1\}^{1 \times M}$ is the one-hot vector for exercise e_j, and W_S and W_E are trainable matrices in the embedding layers. Then, the model $\mathcal{F}$ outputs the predicted response $\hat{r}_{ij}$ as

$$\hat{r}_{ij} = \mathcal{F}(\mathbf{h}_S, \mathbf{h}_E, \mathbf{h}_C), \tag{2}$$

where $\mathcal{F}(\cdot)$ denotes the diagnostic function that integrates three types of inputs in various ways. Generally, after training the model $\mathcal{F}$ using students' response logs, each bit value of $\mathbf{h}_S$ reflects the student's proficiency in the respective knowledge concept.

2.2 Related Work on Cognitive Diagnosis

In the following, some representatives of traditional CDMs and neural CDMs will be reviewed.

Traditional CDMs

As a typical cognitive diagnosis model (CDM), DINA [Torre and J., 2009] predicts $\hat{r}_{ij}$ as

$$\hat{r}_{ij} = g^{1-nt}(1-sl)^{nt}, \text{where } nt = \textstyle\prod_k \boldsymbol{\theta}_k^{\boldsymbol{\beta}_k},\ \boldsymbol{\beta} = \mathbf{h}_C. \tag{3}$$

Here, $\boldsymbol{\theta} \in \{0,1\}^{1 \times K}$ and $\boldsymbol{\beta} \in \{0,1\}^{1 \times K}$ are binary latent features indicating the concepts mastered by the student and those contained in the exercise. $\boldsymbol{\theta}$ is derived from $\mathbf{h}_S$ using a fully connected (FC) layer and Gumbel-Softmax [Jang *et al.*, 2016].

The guessing factor $g \in \mathbb{R}^1$ and slipping factor $sl \in \mathbb{R}^1$ represent the probabilities of correctly answering by guessing and mistakenly answering despite mastering the concept. Both are transformed from $\mathbf{h}_E$ via FC layers. While the prediction process of DINA is interpretable, it suffers from poor performance on large-scale data.

As another typical CDM, the prediction process of IRT [Embretson and Reise, 2013] can be denoted as:

$$\begin{aligned} \hat{r}_{ij} = Sigmoid(a(\theta - \beta)),\ \theta \in \mathbb{R}^1 = FC(\mathbf{h}_S) \\ \beta \in \mathbb{R}^1 = FC(\mathbf{h}_E),\ a \in \mathbb{R}^1 = FC(\mathbf{h}_E) \end{aligned}, \tag{4}$$

where θ is obtained from $\mathbf{h}_S$ by an FC layer, denoting the student ability feature. β and a are transformed from $\mathbf{h}_S$ by two different FC layers, denoting the exercise difficulty and distinction features. As can be seen, the prediction of IRT can be easily understood and interpreted. However, IRT also does not perform well on some complex datasets.

As a multidimensional variant of IRT, MIRT [Reckase, 2009] applies the same logistic function to the linear transformation of the student ability vector $\boldsymbol{\theta} \in \mathbb{R}^{1\times K}$, the difficulty feature $\beta \in \mathbb{R}^1$, and the knowledge concept latent vector $\boldsymbol{\alpha} \in \mathbb{R}^{1\times K}$. That can be denoted as

$$\begin{aligned} \hat{r}_{ij} = Sigmoid(\sum \boldsymbol{\alpha} \odot \boldsymbol{\theta} - \beta) \\ \boldsymbol{\theta} = \mathbf{h}_S, \beta = FC(\mathbf{h}_E),\ \boldsymbol{\alpha} = FC(\mathbf{h}_C) \end{aligned}, \tag{5}$$

where student ability feature $\boldsymbol{\theta}$ and knowledge concept latent feature $\boldsymbol{\alpha}$ are multidimensional and can handle multidimensional data. Therefore, MIRT exhibits better performance than IRT without losing interpretability.

Different from the above traditional CDMs, MF [Koren *et al.*, 2009] is originally devised for recommender systems but can be used for CD. MF holds very high interpretability because its prediction process [Wang *et al.*, 2020] is very easy as $\hat{r}_{ij} = \sum \mathbf{h}_S \odot \mathbf{h}_E$. MF directly applies the inner product to student embedding $\mathbf{h}_S$ and exercise embedding $\mathbf{h}_E$ to compute the similarity. Larger similarity represents a higher probability of correctly the student answering the exercise. MF is quite simple yet effective compared to above CDMs.

Neural CDMs

To improve the diagnosis accuracy, Wang *et al.* incorporated NNs with high-interpretability traditional CDMs like IRT to propose a neural cognitive diagnosis framework (NCD) [Wang *et al.*, 2020; Yu *et al.*, 2024c], whose prediction is as

$$\begin{aligned} &\hat{r}_{ij} = FC_3(FC_2(FC_1(\mathbf{y}))) \\ &\mathbf{y} = \mathbf{h}_C \odot (\mathbf{f}_S - \mathbf{f}_{diff}) \times f_{disc}, \mathbf{f}_S \in \mathbb{R}^{1\times K} \\ &\mathbf{f}_S = Sigmoid(\mathbf{h}_S),\ \mathbf{f}_{diff} \in \mathbb{R}^{1\times K} = Sigmoid(\mathbf{h}_E) \\ &f_{disc} \in \mathbb{R}^1 = Sigmoid(FC(\mathbf{h}_E)), f_{disc} \in \mathbb{R}^1 \end{aligned}. \tag{6}$$

$\mathbf{f}_S$ denotes student ability vector, $\mathbf{f}_{diff}$ and f_{disc} denote exercise difficulty vector and distinction feature. The computation process of $\mathbf{y}$ is similar to IRT, and $\hat{r}_{ij}$ is obtained by applying three FC layers to $\mathbf{y}$.

The three fully connected (FC) layers in NCD present significant challenges for interpretability. This is because FC layers obscure the inference process, making it difficult to trace how input features are transformed into diagnostic results, thereby hindering users' intuitive understanding of the model's internal mechanisms and output reasoning. While NCD incorporates the monotonicity assumption as an effort to improve interpretability, this assumption alone does not provide sufficient clarity on how diagnostic outcomes are derived. In summary, although NCD achieves notable advancements in diagnostic accuracy, it does so at the cost of reduced model interpretability.More neural CDMs can be found in the **Appendix**

2.3 Kolmogorov-Arnold Networks (KANs)

To address the lack of interpretability in existing neural CDMs, which can be mainly attribute to the opaque nature of the MLP. This paper aims to incorporate KANs into neural CDMs or directly leverage KANs for cognitive diagnosis because high model interpretability of KANs as shown in [Liu *et al.*, 2024b]. In the following, KANs will be introduced briefly.

A L-layer MLPs can be written as interleaving of transformations W and activations σ:

$$\mathrm{MLP}(\mathbf{x}) = (W_{L-1} \circ \sigma \circ W_{L-2} \circ \sigma \circ \cdots \circ W_1 \circ \sigma \circ W_0)\,\mathbf{x}, \tag{7}$$

which approximates complex functional mappings through multiple layers of nonlinear transformations. However, its deeply opaque nature constrains the model's interpretability, posing challenges to intuitively understanding the internal decision-making process.

To address the issues of low parameter efficiency and poor interpretability in MLPs, Liu *et al.* [Liu *et al.*, 2024b] introduced the Kolmogorov-Arnold Network (KAN) that is inspired by Kolmogorov-Arnold representation theorem [Braun and Griebel, 2009]. Similar to MLP, a L-layer KAN can be described as a nesting of multiple KAN layers:

$$\mathrm{KAN}(\mathbf{x}) = (\mathbf{\Phi}_{L-1} \circ \mathbf{\Phi}_{L-2} \circ \cdots \circ \mathbf{\Phi}_1 \circ \mathbf{\Phi}_0)\,\mathbf{x}, \tag{8}$$

where $\mathbf{\Phi}_i$ represents the i-th layer of the whole KAN network. For each KAN layer with n_{in} -dimensional input and n_{out} -dimensional output, $\mathbf{\Phi}$ consist of $n_{in} * n_{out}$ 1-D learnable activation functions ϕ:

$$\mathbf{\Phi} = \{\phi_{q,p}\}, \quad p = 1, 2, \cdots, n_{\text{in}}, \quad q = 1, 2 \cdots, n_{\text{out}}. \tag{9}$$

When computing the result of the KAN network from layer l to layer $l+1$, it can be represented in matrix form:

$$\mathbf{x}_{l+1} = \underbrace{\begin{pmatrix} \phi_{l,1,1}(\cdot) & \phi_{l,1,2}(\cdot) & \cdots & \phi_{l,1,n_l}(\cdot) \\ \phi_{l,2,1}(\cdot) & \phi_{l,2,2}(\cdot) & \cdots & \phi_{l,2,n_l}(\cdot) \\ \vdots & \vdots & & \vdots \\ \phi_{l,n_{l+1},1}(\cdot) & \phi_{l,n_{l+1},2}(\cdot) & \cdots & \phi_{l,n_{l+1},n_l}(\cdot) \end{pmatrix}}_{\mathbf{\Phi}_l} \mathbf{x}_l. \tag{10}$$

In conclusion, KANs differentiate themselves from traditional MLPs by using learnable activation functions on the edges and parametrized activation functions as weights, eliminating the need for linear weight matrices. This design allows KANs to achieve comparable performance with smaller model sizes [Liu *et al.*, 2024b; Somvanshi *et al.*, 2024; Peng *et al.*, 2024]. Moreover, their structure enhances model interpretability without compromising performance, making them suitable for applications like scientific discovery [Liu *et al.*, 2024a]. In cognitive diagnostic tasks, KANs may offer precise diagnosis and analysis of learners' knowledge structures, aiding personalized teaching and precision education with intuitive data interpretation.

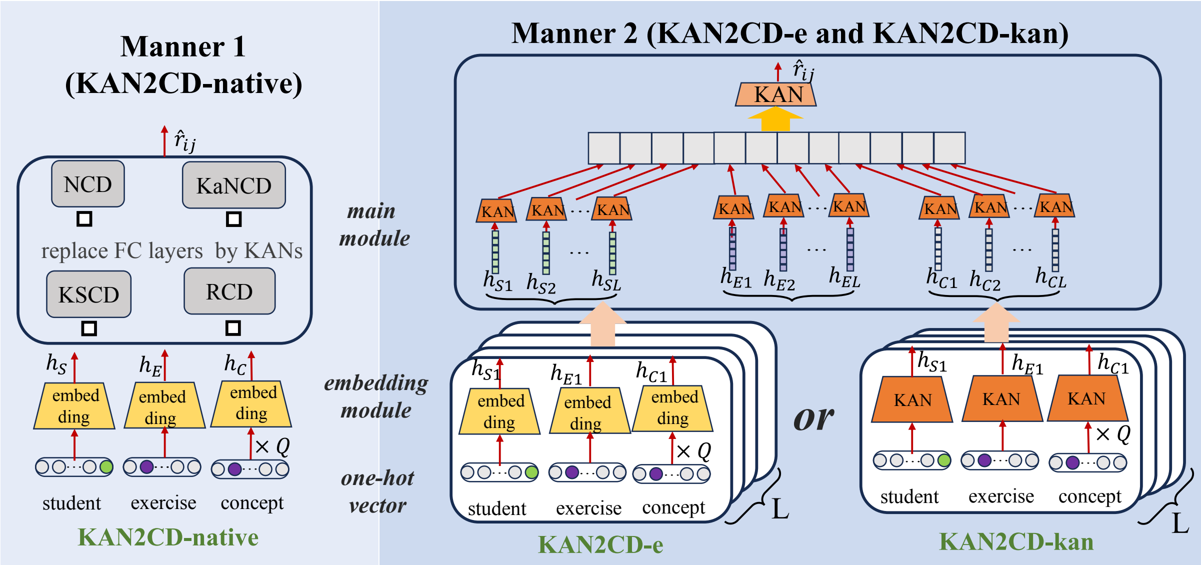

Figure 2: The Overview of the proposed KAN2CD, containing two implementation manners.

3 The Proposed KAN2CD

Overview. Figure 2 presents the overview of the proposed KAN2CD containing two implementation manners. In the first manner (termed **KAN2CD-native**), all FC layers in the utilized CDM (NCD or KaNCD or KSCD or RCD) are replaced by KANs with the same settings. In the second manner, there are two levels of KANs in the main module and two alternative embedding modules. In the main module, KANs at the lower level process the received student, exercise, and concept embeddings, while a KAN at the upper level combines and learns the lower KANs' outputs to get the prediction. For the second manner, **KAN2CD-e** adopts the common embedding layers as its embedding module, while **KAN2CD-kan** adopts KANs as its embedding module.

3.1 Manner 1: KAN2CD-native

KAN2CD-native directly replaces all FC layers in the utilized neural CDM. For example, when adopting the NCD as the main module, KAN2CD-native (can be denoted as NCD+) outputs prediction $\hat{r}_{ij}$ as

$$\begin{aligned} \hat{r}_{ij} &= KAN_2(\mathbf{y}),\ \mathbf{y} = \mathbf{h}_C \odot (\mathbf{f}_S - \mathbf{f}_{diff}) \times f_{disc} \\ \mathbf{f}_S &= Sigmoid(\mathbf{h}_S), \mathbf{f}_{diff} = Sigmoid(\mathbf{h}_E) \qquad . \\ f_{disc} &= Sigmoid(KAN_1(\mathbf{h}_E)) \end{aligned} \tag{11}$$

$KAN_1(\cdot)$ is used to get the exercise difficulty scalar, and $KAN_2(\cdot)$ is used to get $\hat{r}_{ij}$ from $\mathbf{y}$. Due to the page limit, more KAN2CD-native variants (taking KaNCD, KSCD, and RCD as the main modules) can be found in **Appendix**, denoted as KaNCD+, KSCD+, and RCD+.

3.2 Manner 2: KAN2CD-e and KAN2CD-kan

Different from manner 1 under existing CDMs, in manner 2, a novel aggregation framework is designed for cognitive diagnosis based on KANs, which consists of two modules, i.e., the embedding module to get the input embedding and the main module to process input embedding and get the prediction.

The embedding module

This paper designs two alternative embedding modules: the vanilla embedding module using the embedding layers and the KAN embedding module using KANs to get the embedding. For whichever type of embedding module,there are L sub-embedding modules to get L different initial embeddings $\{\mathbf{h}_{Sl}, \mathbf{h}_{El}, \mathbf{h}_{Cl} | 1 \le l \le L\}$. Multiple embeddings provides diverse representations and may cause better results, which is similar to multiple heads in Transformer [Vaswani *et al.*, 2017]

For the vanilla embedding module, each sub-embedding module's forward process is the same as Eq.(1), and the process of the vanilla embedding module can be denoted as

$$\begin{aligned} \mathbf{h}_{Sl} &= \mathbf{x}_i^S \times W_{Sl}, W_{Sl} \in \mathbb{R}^{N \times D} \\ \mathbf{h}_{El} &= \mathbf{x}_j^E \times W_{El}, W_{El} \in \mathbb{R}^{M \times D}, 1 \le l \le L. \\ \mathbf{h}_{Cl} &= \mathbf{x}_j^C \times W_{Ql}, W_{Ql} \in \mathbb{R}^{K \times D} \end{aligned} \tag{12}$$

For the KAN embedding module, each sub-embedding module's forward process is as

$$\begin{aligned} \mathbf{h}_{Sl} &= KAN_{Sl}(\mathbf{x}_i^S | \Phi_{Sl}) \\ \mathbf{h}_{El} &= KAN_{El}(\mathbf{x}_j^E | \Phi_{El}), 1 \le l \le L. \\ \mathbf{h}_{Cl} &= KAN_{Cl}(\mathbf{x}_j^C | \Phi_{Cl}) \end{aligned} \tag{13}$$

$\mathbf{h}_{Sl}$ has a length of D, and thus Φ_{Sl} in $KAN_{Sl}(\cdot)$ contains $D * N$ learnable functions to learn the embedding $\mathbf{h}_{Sl}$. Φ_{El} and Φ_{Cl} hold $D * M$ and $D * K$ learnable functions.

The main module

After getting the input embedding set $H = \{H_1, \ldots, H_{3*L}\} = \{\mathbf{h}_{Sl}, \mathbf{h}_{El}, \mathbf{h}_{Cl} | 1 \le l \le L\}$, the main module is used to porocess these embeddings to get the final prediction $\hat{r}_{ij}$ by two levels of KANs.

In the lower level, there are $3 \times L$ KANs used for handling the input embedding set and obtaining the latent vector $\mathbf{v} = \{v_1, v_2, \dots, v_{3*L}\}$, where the forward pass process is as follows:

$$v_i = KAN_i^{low}(H_i|\Phi_i^{low}),\ 1 \le i \le 3 * L. \tag{14}$$

Afthewards, in the upper level, a unified 2-layer KAN KAN^{up} is used to process to the latent vector $\mathbf{v}$ as

$$\hat{r}_{ij} = KAN^{up}(\mathbf{v}|\Phi_1^{up}, \Phi_2^{up}) = \Phi_2^{up} \circ \mathbf{ls} = \Phi_2^{up} \circ \Phi_1^{up} \circ \mathbf{v}, \tag{15}$$

where Φ_1^{up} contains $D \times 3 * L$ learnable functions and Φ_2^{up} contains $1 \times K$ learnable functions. Note that there is a latent vector $\mathbf{ls} \in \mathbb{R}^{1\times K}$ with length of K, which can be used to represent the student's knowledge mastery across different knowledge concepts.

While in manner 1, the student's knowledge mastery vector (i.e., the student knowledge proficiency vector) is still represented by the latent student ability vector. For example, $\mathbf{f}_S$ represents the student knowledge proficiency in NCD+ and KaNCD+, while $\hat{\mathbf{h}_S}$ represents the student knowledge proficiency in KSCD+ and RCD+.

3.3 Model Training and Implementation

Model Training. To train the proposed KAN2CD model, the Adam optimizer [Kingma and Ba, 2014] is used to mine the following cross entropy loss [De Boer *et al.*, 2005] between model's output $\hat{r}_{ij}$ and true response r_{ij}:

$$\mathcal{L} = -\sum\nolimits_{(s_i,e_j,r_{ij})\in\mathcal{R}_{log}} (r_{ij}\log y_{ij}) + (1 - r_{ij})\log(1 - y_{ij})\,. \tag{16}$$

Implementation Modification. The original implementation of KANs is inefficient because intermediate variables X need to be expanded to perform different pre-given activation functions, which will demand more memory and incur higher training costs. Considering activation functions can be linear combinations of a fixed set of B-splines basis functions $B = \{B_1(\cdot), \dots, B_L(\cdot)\}$, the process of one activation function can be rewritten as $[B_1(X), \dots, B_L(X)] \times W_{linear}, W_{linear} \in \mathbb{R}^{L\times 1}$, i.e., activation with multiple basis functions and combine them linearly.

To further enhance the computational efficiency of KANs, we have integrated the FastKAN implementation as detailed by Li et al. [Li, 2024]. Standard KANs traditionally rely on B-spline activation functions, which can be computationally intensive. The FastKAN variant circumvents this bottleneck by approximating these B-spline functions with Radial Basis Functions (RBFs). This substitution offers a more streamlined and efficient architectural approach for deploying KANs, as RBFs typically involve simpler and faster computations. Consequently, the adoption of FastKAN not only accelerates the individual KAN modules but also improves the overall performance and scalability of our entire KAN2CD framework.

4 Experiments

The following experiments aim to answer the following researcher questions:

RQ1: How about the effectiveness of manner 1 of KAN2CD?

RQ2: How about the effectiveness of manner 2 of KAN2CD, i.e., KAN2CD-e(-kan)?

RQ3: How about the efficiency and complexity of the proposed KAN2CD?

RQ4: How about the interpretability (visualization) of KAN2CD?

RQ5: How can KAN2CD be interpreted?

RQ6: What is the sensitivity of KAN2CD to hyperparameters?

4.1 Experimental Settings

Datasets. To verify the proposed KAN2CD, we conducted experiments on four education datasets, including ASSISTments [Feng *et al.*, 2009], SLP [Lu *et al.*, 2021] JunYi [Chang *et al.*, 2015] and FrcSub [AICFE, 2018]. Their detailed descriptions are in **Appendix**.

Compared Models. Four traditional CDMs (including IRT, MIRT, DINA, and MF) were taken as baselines. Besides, four neural CDMs (including NCD, KaNCD, KSCD, and RCD) are taken as baselines to validate the effectiveness of KAN2CD-native.

Model Settings. For all models, their embedding dimension D is set to the number of concepts K. The hyperparameters of comparison models follow the original papers. For KAN2CD, the batch size and learning rate are set to 128 and 0.002, the training epoch number is set to 20. All models were implemented in PyTorch 2.3.0 and executed under an Intel 13700k CPU. Accuracy (ACC) and *Area Under the ROC Curve* (AUC) were used as evaluation metrics.

4.2 Effectiveness of The KAN2CD (RQ1 & RQ2)

To answer RQ1 and RQ2, Table 1 summarizes the performance of all CDMs on four datasets in terms of AUC and ACC. To assess statistical significance, Wilcoxon rank-sum tests (5% significance level) were conducted. Table 2 shows the results, where '+', '-', and '≈' indicate whether KAN2CD-e or KAN2CD-kan is better than, worse than, or similar to baselines, respectively. Key observations are as follows.

Firstly, KAN2CD-native variants (NCD+, KaNCD+, KSCD+, RCD+) achieve significantly better AUC and ACC values than traditional CDMs (IRT, MIRT, DINA, MF) across ASSISTments, SLP, and JunYi datasets. For example, they consistently outperform traditional CDMs in AUC and achieve competitive ACC. On FrcSub, their ACC values are slightly worse than MF, reflecting the general trend of neural CDMs underperforming MF on this dataset. Despite this, KAN2CD-native models show clear advantages over traditional approaches across most datasets.

Secondly, KAN2CD-e achieves top-tier performance across all datasets, consistently surpassing traditional and neural CDMs. The Wilcoxon test validates its reliability, showing 8/0/0 outcomes against traditional CDMs and 7/0/1 against neural CDMs like KaNCD and KSCD. KAN2CD-kan, though slightly less effective than KAN2CD-e on certain datasets, remains competitive. On datasets SLP, JunYi, and FrcSub, it achieves comparable outcomes, outperforming most baselines except KSCD. Wilcoxon test results show 8/0/0 against traditional models and balanced outcomes against neural CDMs, e.g., 4/2/2 against NCD and 4/3/1 against KSCD. Despite

Dataset	Assistments		SLP		JunYi		FrcSub	
Method	AUC↑	ACC↑	AUC↑	ACC↑	AUC↑	ACC↑	AUC↑	ACC↑
IRT	72.02%	70.25%	80.91%	74.29%	74.80%	72.74%	80.63%	57.14%
MIRT	65.84%	63.90%	72.78%	71.90%	69.59%	69.50%	81.93%	69.12%
DINA	72.15%	68.06%	77.24%	71.43%	75.81%	68.18%	80.66%	78.16%
MF	70.55%	68.26%	79.22%	72.80%	79.48%	74.15%	84.10%	81.36%
NCD	74.84%	72.15%	84.76%	80.72%	80.70%	76.73%	90.12%	70.15%
NCD+	75.71%	71.91%	84.72%	80.38%	80.38%	76.12%	90.66%	74.30%
KaNCD	76.44%	**73.33**%	85.21%	81.61%	80.80%	76.15%	90.11%	76.68%
KaNCD+	76.99%	73.54%	85.25%	81.91%	82.06%	77.23%	91.44%	78.36%
RCD	75.91%	72.99%	85.57%	79.37%	**83.25**%	**78.67**%	89.39%	73.83%
RCD+	77.10%	73.78%	86.38%	80.12%	83.33%	78.76%	89.46%	74.53%
KSCD	76.55%	73.04%	85.90%	81.02%	82.17%	77.83%	90.49%	80.27%
KSCD+	76.72%	73.01%	86.06%	80.87%	83.43%	78.67%	90.66%	82.93%
KAN2CD-e	**76.64**%	72.96%	**86.08**%	**82.66**%	83.18%	78.39%	**91.27**%	**84.58**%
KAN2CD-kan	72.89%	71.55%	85.91%	81.91%	83.14%	78.41%	90.38%	83.30%

Table 1: Overall performance comparison: the best/second-best is bolded/ underlined. KAN2CD-native is not involved.

+/-/≈	IRT	MIRT	DINA	MF	NCD	KaNCD	RCD	KSCD
KAN2CD-e	8/0/0	8/0/0	8/0/0	8/0/0	8/0/0	7/0/1	5/1/2	7/0/1
KAN2CD-kan	8/0/0	8/0/0	8/0/0	8/0/0	4/2/2	3/2/3	4/2/2	4/3/1

Table 2: Wilcoxon rank sum test results with a 5% significance level. +, -, and ≈ indicate KAN2CD-e/KAN2CD-kan is better than, worse than, and similar to baselines.

being marginally outperformed by KAN2CD-e and KSCD, KAN2CD-kan remains an effective method in cognitive diagnosis, surpassing traditional CDMs and several neural counterparts.

4.3 Complexity&Efficiency of KAN2CD (RQ3)

To demonstrate the performance of the proposed KAN2CD model in terms of complexity and efficiency, Table 3 provides a comparison of neural cognitive diagnostic models and KAN2CD models in terms of parameter count and training runtime across different datasets. Here, 'K' represents kilo, and 'h' stands for hours.

As can be seen, the parameters of KAN2CD-native are less than half of the corresponding neural CDM, while parameters of KAN2CD-e and KAN2CD-kan are much more than neural CDMs, especially for KAN2CD-kan. The higher parameter count in KAN2CD-kan contributes to its increased runtime. Despite this, the runtimes of KAN2CD-native, which utilizes our custom KAN implementation (models without a specific suffix like '+$\mathcal{O}$' or '+$\mathcal{A}$'), are close to those of neural CDMs. This indicates that the efficiency of our KAN2CD framework with our optimized KAN components is competitive. This enhanced efficiency is primarily attributed to our modified implementation of KANs. The original PyKAN implementation (denoted with a '+$\mathcal{O}$' suffix) is notably inefficient. For instance, as shown in Table 4, NCD+$\mathcal{O}$ required approximately 0.9 on ASSISTments and 1.1 on JunYi, while KaNCD+$\mathcal{O}$ took about 0.9 on ASSISTments and 0.35 on JunYi. These times are substantially longer than those of our optimized versions or standard neural CDMs. More experimental can be found **in appendix**.

To further enhance efficiency, we incorporated FastKAN, an faster KAN variant that leverages RBF (Radial Basis Function) approximations. Models employing this FastKAN implementation are distinguished by a '+$\mathcal{A}$' suffix (e.g., KAN2CD-e+$\mathcal{A}$, NCD+$\mathcal{A}$, KaNCD+$\mathcal{A}$). The integration of FastKAN leads to significant reductions in training time, positioning these '+$\mathcal{A}$' suffixed models as the most computationally efficient among the KAN-based approaches we evaluated. Table 4 details these efficiency gains, demonstrating that models with the '+$\mathcal{A}$' suffix consistently train faster on both the ASSISTments and JunYi datasets compared to their counterparts using our standard custom KAN implementation (e.g., KAN2CD-e+$\mathcal{A}$ vs. KAN2CD-e) and are vastly superior to the original PyKAN ('+$\mathcal{O}$') versions.

4.4 Visualization of KAN2CD (RQ4)

To address RQ3, Figure 3 presents the final KAN layer structures for NCD+ and KaNCD+ on the JunYi dataset (selected for its manageable number of knowledge concepts, K). This layer is crucial as it replaces traditional Multi-Layer Perceptrons (MLPs) in the prediction process.

As shown in Figure 3, the learned KAN structures for both models are strikingly similar, more importantly for interpretability, the KANs are markedly sparse. For instance, only a few connections (e.g., 8 out of 39) are retained in the final structures. These preserved connections correspond to key knowledge concepts. We observe that these selected knowledge nodes typically exhibit higher in-degree and out-degree within the knowledge graph. This structural prominence sug-

Param.	NCD	NCD+	KaNCD	KaNCD+	RCD	RCD+	KSCD	KSCD+	KAN2CD-kan	KAN2CD-e
SLP	232K	83K	91K	50K	279K	115K	94K	81K	4143K	471K
JunYi	219K	68K	79K	36K	266K	97K	74K	67K	3420K	416K

Runtime	NCD	NCD+	KaNCD	KaNCD+	RCD	RCD+	KSCD	KSCD+	KAN2CD-kan	KAN2CD-e
ASSIST	0.12h	0.17h	0.15h	0.2h	0.16h	0.28h	0.46h	1.12h	15h	1.2h
JunYi	0.02h	0.02h	0.06h	0.08h	0.03h	0.03h	0.08h	0.13h	1.3h	0.33h

Table 3: Model parameter and training runtime comparisons between neural CDMs and KAN2CD on SLP & JunYi and ASSISTments & JunYi. ('K' means kilo, 'h' represents hours).

Dataset	NCD+	NCD $+\mathcal{O}$	NCD $+\mathcal{A}$	KaNCD+	KaNCD $+\mathcal{O}$	KaNCD $+\mathcal{A}$	KAN2CD-e	KAN2CD-e $+\mathcal{O}$	KAN2CD-e $+\mathcal{A}$
ASSISTments	0.17 h	0.9 h	**0.07 h**	0.2 h	0.9 h	**0.11 h**	1.20 h	11.3 h	**0.33 h**
JunYi	0.02 h	1.1 h	**0.01 h**	0.08 h	0.35 h	**0.03 h**	0.33 h	6.8 h	**0.19 h**

Table 4: Training runtime comparison on ASSISTments & JunYi (without RCD). Order: our implementation, original PyKAN ($+\mathcal{O}$), FastKAN ($+\mathcal{A}$). Runtimes are in hours (h).

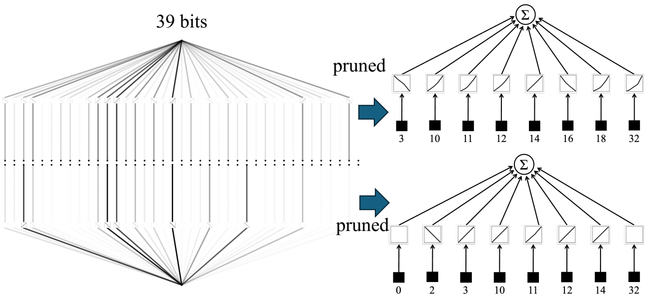

Figure 3: Visualization of NCD+ (upper) and KaNCD+ (lower) on JunYi datasets.

gests that, in the learning process, they maintain a highly interconnected and influential relationship with other concepts, often serving as crucial hubs for knowledge construction and flow. To visually contextualize these key concepts, Figure 4 depicts the prerequisite and successor relationships among all knowledge concepts. In this graph, the knowledge concepts identified as key by the KAN (and thus retained) are distinctly marked (e.g., in red), highlighting their central roles and interconnections within the broader knowledge structure. Consequently, by focusing on these essential connections, which are further contextualized by their depiction in Figure 4, the models effectively identify the most salient concepts.

4.5 How to interpert KAN2CD (RQ5)

Taking NCD+ in Figure 3 as an example, y is conceptualized as the difference between student ability and item difficulty. These differences quantitatively capture the contrast between a student's mastery level of various concepts and the challenges presented by the corresponding items. Specifically, a larger y_i denotes that the student's ability in the i-th concept significantly surpasses the difficulty of the associated item, indicating a higher probability of a correct response. Conversely, a smaller y_i reflects an insufficient ability relative to the item difficulty, leading to a lower likelihood of success.

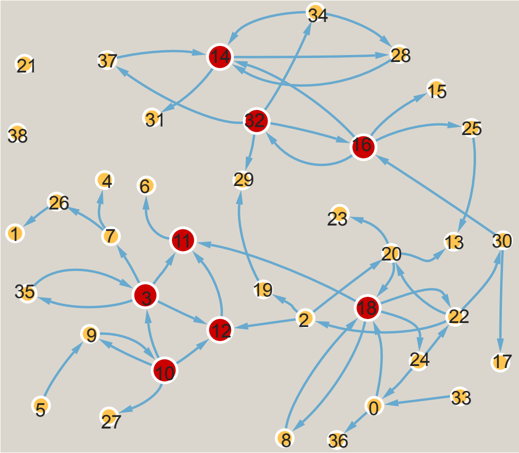

Figure 4: KAN-selected concepts (red) in the JunYi dataset.

The formula $\text{KAN}^2(y) = y_3^{-1} + y_{10} + y_{11}^2 + y_{12} + y_{14} + y_{16}^{-1} + y_{18}^2 + y_{32}^2$ elucidates the role of these differences in the prediction process. Each term involving y_i delineates the influence and importance of the difference for a specific concept. For instance, y_3^{-1} suggests that the contribution of the third concept difference is inversely proportional, with its influence diminishing as y_3 increases. This may highlight scenarios where the effect of foundational concepts diminishes after reaching a certain threshold. On the other hand, y_{11}^2 underscores a quadratic effect for the eleventh concept difference, indicating an amplified impact of variations in the ability-difficulty difference for this concept. Such behavior is often associated with items of higher difficulty that exhibit greater sensitivity to ability differences.

By introducing KAN, this framework offers a clear symbolic representation of how students' abilities impact predictions, enhancing interpretability compared to traditional MLP models. Additionally, it provides students with an intuitive understanding of ability-item difficulty relationships and gives

l	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
SLP	83.56%	84.68%	85.62%	86.06%	**86.08%**	85.68%	84.89%	84.62%
FrcSub	90.26%	90.89%	91.11%	91.16%	**91.27%**	91.13%	91.12%	90.82%

Table 5: AUC performance of KAN2CD-e under varying l on the SLP and FrcSub dataset.

Model	$(K \to 512 \to 256 \to 1)$	$(K \to 128 \to 1)$	$(K \to 16 \to 1)$	$(K \to 1)$
NCD+	75.43%	75.37%	75.43%	**75.71%**
KaNCD+	75.41%	75.86%	76.77%	**76.99%**

Table 6: AUC performance of KAN2CD-native across different hidden layer sizes on the ASSISTments dataset.

KAN2CD-e	$(K \to 1)$	$(K \to 8 \to 1)$	$(K \to 16 \to 1)$	$(K \to 32 \to 16 \to 1)$
SLP	**86.08%**	85.84%	86.04%	85.58%
FrcSub	91.27%	**91.29%**	91.27%	91.03%

Table 7: AUC performance of KAN2CD-e across different hidden layer sizes on SLP and FrcSub dataset.

teachers clearer insights for designing targeted strategies.

4.6 What is the sensitivity of KAN2CD to hyperparameters? (RQ6)

Impact of KAN hyperparameters:

To investigate the effect of B-splines in KAN on KAN2CD performance under different order of piecewise polynomial (P) and number of grid intervals (G), experiments were conducted on the FrcSub dataset, as show in Figure 5.

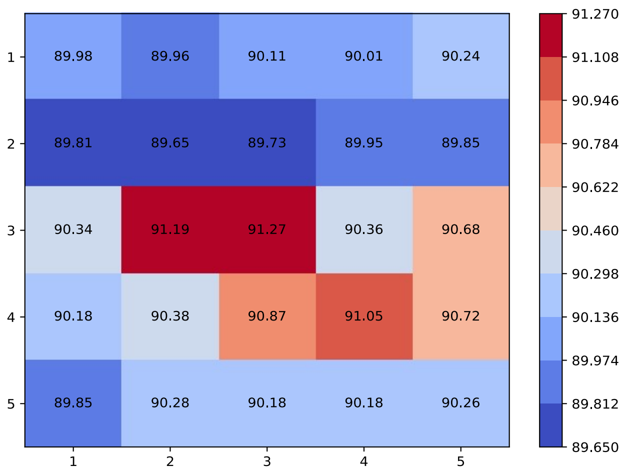

Figure 5: AUC performance of KAN2CD-e under varying B-splines settings

When G and P are both 1, the model's AUC is relatively low at 89.98%, indicating weak fitting ability. As P and G increase, AUC improves, reaching 91.19% for $G = 3$ and $P = 2$, and 91.27% for $G = 3$ and $P = 3$. This shows that model complexity enhances fitting ability. However, further increases in P and G, such as when both are 5, maintain a high AUC of about 90.26%, but no considerable performance improvement is observed. This suggests that while moderate increases in complexity improve performance, excessive complexity may lead to overfitting.

Impact of hyperparameter l on KAN2CD-e: To analyze the impact of hyperparameter l on KAN2CD-e, we evaluate AUC on the SLP and FrcSub datasets as l varies from 1 to 8 (Table 5). AUC initially increases, reaching a peak before slightly declining. For SLP, AUC rises from 83.56% at $l = 1$ to 86.08% at $l = 5$, then drops slightly.

Impact of hidden layer size on manner 1 Evaluating KAN2CD-native on ASSISTments with varied hidden layer configurations (from 3-layer ($K \to 512 \to 256 \to 1$) to 1-layer ($K \to 1$), where K is knowledge concepts revealed, per Table 6, that increasing hidden layers struggles to boost AUC. NCD+ performance was stable, peaking at 75.71% with the 1-layer ($K \to 1$) setup. KaNCD+ improved with simpler models, its AUC rising from 75.41% (3-layer) to 76.99% (1-layer ($K \to 1$)); a 2-layer model ($K \to 16 \to 1$) also achieved 76.77%. Consequently, simpler hidden layer designs often prove superior for these cognitive diagnosis tasks.

Effect of hidden layer size on manner 2 For KAN2CD-e, hidden layer configurations from one ($K \to 1$) to three ($K \to 32 \to 16 \to 1$) layers (K denotes knowledge concepts) were evaluated on SLP and FrcSub dataset (see Table 7). On SLP, a single layer performed best (86.08%); for FrcSub, a two-layer model ($K \to 8 \to 1$) was slightly better (91.29%), though differences across depths were minimal. These results indicate simpler architectures are optimal, as deeper layers provide negligible benefits and may overfit.

5 Conclusion

In this paper, we introduce KAN2CD to improve the interpretability of neural CDMs using KANs. KAN2CD can be implemented in two ways: the first replaces MLPs with KANs to enhance interpretability, while the second constructs a novel CDM entirely with KANs. Experimental results validate the effectiveness and high interpretability of KAN2CD.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (No.62302010, No.62107001), in part by China Postdoctoral Science Foundation (No.2023M740015), in part by the Postdoctoral Fellowship Program (Grade B) of China Postdoctoral Science Foundation (No.GZB20240002), and in part by the Anhui Province Key Laboratory of Intelligent Computing and Applications (No. AFZNJS2024KF01).

References

[AICFE, 2018] AICFE. Frcsub dataset. http://staff.ustc.edu.cn/%7Eqiliuql/data/math2015.rar, 2018.

[Anderson *et al.*, 2014] Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, and Jure Leskovec. Engaging with massive online courses. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 687–698, 2014.

[Beck, 2007] Joseph Beck. Difficulties in inferring student knowledge from observations (and why you should care). In *Proceedings of the 13rd International Conference of Artificial Intelligence in Education*, pages 21–30, 2007.

[Braun and Griebel, 2009] Jürgen Braun and Michael Griebel. On a constructive proof of kolmogorov's superposition theorem. *Constructive approximation*, 30:653–675, 2009.

[Chang *et al.*, 2015] Haw-Shiuan Chang, Hwai-Jung Hsu, Kuan-Ta Chen, et al. Modeling exercise relationships in e-learning: A unified approach. In *Proceedings of the 2015 International Educational Data Mining Society (EDM*, pages 532–535, 2015.

[De Boer *et al.*, 2005] Pieter-Tjerk De Boer, Dirk P Kroese, Shie Mannor, and Reuven Y Rubinstein. A tutorial on the cross-entropy method. *Annals of operations research*, 134:19–67, 2005.

[Embretson and Reise, 2013] Susan E Embretson and Steven P Reise. *Item Response Theory*. Psychology Press, 2013.

[Fan *et al.*, 2021] Feng-Lei Fan, Jinjun Xiong, Mengzhou Li, and Ge Wang. On interpretability of artificial neural networks: A survey. *IEEE Transactions on Radiation and Plasma Medical Sciences*, 5(6):741–760, 2021.

[Feng *et al.*, 2009] Mingyu Feng, Neil Heffernan, and Kenneth Koedinger. Addressing the assessment challenge with an online system that tutors as it assesses. *User Modeling and User-adapted Interaction*, 19(3):243–266, 2009.

[Gao *et al.*, 2021] Weibo Gao, Qi Liu, Zhenya Huang, Yu Yin, Haoyang Bi, Mu-Chun Wang, Jianhui Ma, Shijin Wang, and Yu Su. Rcd: Relation map driven cognitive diagnosis for intelligent education systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 501–510, 2021.

[Hu *et al.*, 2023] Liya Hu, Zhiang Dong, Jingyuan Chen, Guifeng Wang, Zhihua Wang, Zhou Zhao, and Fei Wu. Ptadisc: A cross-course dataset supporting personalized learning in cold-start scenarios. In *Proceedings of the 37th Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

[Jang *et al.*, 2016] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[Kingma and Ba, 2014] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[Koren *et al.*, 2009] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[Li, 2024] Ziyao Li. Kolmogorov-arnold networks are radial basis function networks. 2024.

[Liu *et al.*, 2024a] Ziming Liu, Pingchuan Ma, Yixuan Wang, Wojciech Matusik, and Max Tegmark. Kan 2.0: Kolmogorov-arnold networks meet science. *arXiv preprint arXiv:2408.10205*, 2024.

[Liu *et al.*, 2024b] Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y Hou, and Max Tegmark. Kan: Kolmogorov-arnold networks. *arXiv preprint arXiv:2404.19756*, 2024.

[Lu *et al.*, 2021] Yu Lu, Yang Pian, Ziding Shen, Penghe Chen, and Xiaoqing Li. SLP: A multi-dimensional and consecutive dataset from K-12 education. In *Proceedings of the 29th International Conference on Computers in Education Conference*, volume 1, pages 261–266, 2021.

[Ma *et al.*, 2022] Haiping Ma, Manwei Li, Le Wu, Haifeng Zhang, Yunbo Cao, Xingyi Zhang, and Xuemin Zhao. Knowledge-sensed cognitive diagnosis for intelligent education platforms. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 1451–1460, 2022.

[Nabizadeh *et al.*, 2020] Amir Hossein Nabizadeh, José Paulo Leal, Hamed N Rafsanjani, and Rajiv Ratn Shah. Learning path personalization and recommendation methods: A survey of the state-of-the-art. *Expert Systems with Applications*, 159:113596, 2020.

[Patikorn *et al.*, 2018] Thanaporn Patikorn, Neil T Heffernan, and Ryan S Baker. Assistments longitudinal data mining competition 2017: A preface. In *Proceedings of the 2018 International Conference on Educational Data Mining*, 2018.

[Peng *et al.*, 2024] Yanhong Peng, Miao He, Fangchao Hu, Zebing Mao, Xia Huang, and Jun Ding. Predictive modeling of flexible ehd pumps using kolmogorov-arnold networks. *arXiv preprint arXiv:2405.07488*, 2024.

[Qian *et al.*, 2024] Hong Qian, Shuo Liu, Mingjia Li, Bingdong Li, Zhi Liu, and Aimin Zhou. Orcdf: An oversmoothing-resistant cognitive diagnosis framework for student learning in online education systems. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2455–2466, 2024.

[Reckase, 2009] Mark D Reckase. *Multidimensional Item Response Theory Models*. Springer, 2009.

[Samek *et al.*, 2016] Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE Transactions on Neural Networks and Learning Systems*, 28(11):2660–2673, 2016.

[Shen *et al.*, 2024] Junhao Shen, Hong Qian, Shuo Liu, Wei Zhang, Bo Jiang, and Aimin Zhou. Capturing homogeneous influence among students: Hypergraph cognitive diagnosis for intelligent education systems. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2628–2639, 2024.

[Somvanshi *et al.*, 2024] Shriyank Somvanshi, Syed Aaqib Javed, Md Monzurul Islam, Diwas Pandit, and Subasish Das. A survey on kolmogorov-arnold network. *arXiv preprint arXiv:2411.06078*, 2024.

[Stojanoski *et al.*, 2018] Bobby Stojanoski, Kathleen M Lyons, Alexandra AA Pearce, and Adrian M Owen. Targeted training: Converging evidence against the transferable benefits of online brain training on cognitive function. *Neuropsychologia*, 117:541–550, 2018.

[Torre and J., 2009] D. L. Torre and J. Dina model and parameter estimation: A didactic. *Journal of Educational and Behavioral Statistics*, 34(1):115–130, 2009.

[Urdaneta-Ponte *et al.*, 2021] María Cora Urdaneta-Ponte, Amaia Mendez-Zorrilla, and Ibon Oleagordia-Ruiz. Recommendation systems for education: Systematic review. *Electronics*, 10(14):1611, 2021.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[Wang *et al.*, 2020] Fei Wang, Qi Liu, Enhong Chen, Zhenya Huang, Yuying Chen, Yu Yin, Zai Huang, and Shijin Wang. Neural cognitive diagnosis for intelligent education systems. In *Proceedings of the 2020 AAAI Conference on Artificial Intelligence*, volume 34, pages 6153–6161, 2020.

[Wang *et al.*, 2022] Fei Wang, Qi Liu, Enhong Chen, Zhenya Huang, Yu Yin, Shijin Wang, and Yu Su. Neuralcd: a general framework for cognitive diagnosis. *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[Wang *et al.*, 2024] Fei Wang, Weibo Gao, Qi Liu, Jiatong Li, Guanhao Zhao, Zheng Zhang, Zhenya Huang, Mengxiao Zhu, Shijin Wang, Wei Tong, et al. A survey of models for cognitive diagnosis: New developments and future directions. *arXiv preprint arXiv:2407.05458*, 2024.

[Wu *et al.*, 2024] Siyu Wu, Yang Cao, Jiajun Cui, Runze Li, Hong Qian, Bo Jiang, and Wei Zhang. A comprehensive exploration of personalized learning in smart education: From student modeling to personalized recommendations. *arXiv preprint arXiv:2402.01666*, 2024.

[Yang *et al.*, 2023] Shangshang Yang, Haoyu Wei, Haiping Ma, Ye Tian, Xingyi Zhang, Yunbo Cao, and Yaochu Jin. Cognitive diagnosis-based personalized exercise group assembly via a multi-objective evolutionary algorithm. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 7(3):829–844, 2023.

[Yu *et al.*, 2024a] Xiaoshan Yu, Chuan Qin, Dazhong Shen, Haiping Ma, Le Zhang, Xingyi Zhang, Hengshu Zhu, and Hui Xiong. Rdgt: enhancing group cognitive diagnosis with relation-guided dual-side graph transformer. *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[Yu *et al.*, 2024b] Xiaoshan Yu, Chuan Qin, Dazhong Shen, Shangshang Yang, Haiping Ma, Hengshu Zhu, and Xingyi Zhang. Rigl: A unified reciprocal approach for tracing the independent and group learning processes. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4047–4058, 2024.

[Yu *et al.*, 2024c] Xiaoshan Yu, Chuan Qin, Qi Zhang, Chen Zhu, Haiping Ma, Xingyi Zhang, and Hengshu Zhu. Disco: A hierarchical disentangled cognitive diagnosis framework for interpretable job recommendation. In *IEEE International Conference on Data Mining (ICDM) 2024*, 2024.

[Zhang *et al.*,] Zheng Zhang, Wei Song, Qi Liu, Qingyang Mao, Yiyan Wang, Weibo Gao, Zhenya Huang, Shijin Wang, and Enhong Chen. Towards accurate and fair cognitive diagnosis via monotonic data augmentation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

[Zhang *et al.*, 2021] Yu Zhang, Peter Tiňo, Aleš Leonardis, and Ke Tang. A survey on neural network interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 5(5):726–742, 2021.

[Zhang *et al.*, 2023] Zheng Zhang, Qi Liu, Hao Jiang, Fei Wang, Yan Zhuang, Le Wu, Weibo Gao, and Enhong Chen. Fairlisa: fair user modeling with limited sensitive attributes information. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 41432–41450, 2023.