Leveraging Transferable Knowledge Concept Graph Embedding for Cold-Start Cognitive Diagnosis

Weibo Gao
Anhui Province Key Laboratory of Big Data Analysis and Application, School of Computer Science and Technology, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
weibogao@mail.ustc.edu.cn

Hao Wang
Anhui Province Key Laboratory of Big Data Analysis and Application, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
wanghao3@ustc.edu.cn

Qi Liu*
Anhui Province Key Laboratory of Big Data Analysis and Application, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
qiliuql@ustc.edu.cn

Fei Wang
Anhui Province Key Laboratory of Big Data Analysis and Application, School of Computer Science and Technology, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
wf314159@mail.ustc.edu.cn

Xin Lin
Anhui Province Key Laboratory of Big Data Analysis and Application, School of Computer Science and Technology, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
linx@mail.ustc.edu.cn

Linan Yue
Anhui Province Key Laboratory of Big Data Analysis and Application, School of Data Science, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
lnyue@mail.ustc.edu.cn

Zheng Zhang
Anhui Province Key Laboratory of Big Data Analysis and Application, School of Computer Science and Technology, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
zhangzheng@mail.ustc.edu.cn

Rui Lv
Anhui Province Key Laboratory of Big Data Analysis and Application, School of Computer Science and Technology, University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence
Hefei, China
lvrui2018@mail.ustc.edu.cn

Shijin Wang
State Key Laboratory of Cognitive Intelligence & iFLYTEK AI Research (Central China), iFLYTEK Co., Ltd
Hefei, China
sjwang3@iflytek.com

ABSTRACT

Cognitive diagnosis (CD) aims to reveal the proficiency of students on specific knowledge concepts and traits of test exercises (e.g., difficulty). It plays a critical role in intelligent education systems by supporting personalized learning guidance. However, recent developments in CD mostly concentrate on improving the accuracy of diagnostic results and often overlook the important and practical task: domain-level zero-shot cognitive diagnosis (DZCD). The primary challenge of DZCD is the deficiency of student behavior data in the target domain due to the absence of student-exercise interactions or unavailability of exercising records for training purposes. To tackle the cold-start issue, we propose a two-stage solution named TechCD (**T**ransferable knowledg**E** **C**oncept grap**H** embedding framework for **C**ognitive **D**iagnosis). The fundamental notion involves utilizing a pedagogical knowledge concept graph (KCG) as a mediator to connect disparate domains, allowing the transmission of student cognitive signals from established domains to the zero-shot cold-start domain. Specifically, a naive yet effective graph convolutional network (GCN) with the bottom-layer discarding operation is initially employed over the KCG to learn transferable student cognitive states and domain-specific exercise traits. Moreover, we give three implementations of the general TechCD framework following the typical cognitive diagnosis solutions. Finally, extensive experiments on real-world datasets not only prove that Tech can effectively perform zero-shot diagnosis, but also give some popular applications such as exercise recommendation.

*Corresponding author.

SIGIR '23, July 23–27, 2023, Taipei, Taiwan

ACM ISBN 978-1-4503-9408-6/23/07…$15.00
https://doi.org/10.1145/3539618.3591774

CCS CONCEPTS

• **Applied computing** → **E-learning**.

KEYWORDS

cognitive diagnosis; student performance prediction; cold-start; knowledge concept graph

ACM Reference Format:
Weibo Gao, Hao Wang, Qi Liu, Fei Wang, Xin Lin, Linan Yue, Zheng Zhang, Rui Lv, and Shijin Wang. 2023. Leveraging Transferable Knowledge Concept Graph Embedding for Cold-Start Cognitive Diagnosis. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23), July 23–27, 2023, Taipei, Taiwan.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3539618.3591774

1 INTRODUCTION

Intelligent education systems facilitate the personalized learning of students with computer-assisted technology by providing open access to abundant learning materials (e.g., exercises). Their prevalence and convenience have received great attention from both educators and the general public [27]. In these platforms, cognitive diagnosis (CD) plays a crucial role in providing customized applications tailored to individual needs [37]. Specifically, the goal of CD is to profile students' latent cognitive *proficiency* on specific knowledge concepts, as well as to reveal characteristics of the test exercises such as *difficulty* and *discrimination* [9, 37]. As the diagnostic results can support further educational applications, such as exercise recommendation [19, 46] and learning path suggestions [18, 39], a number of existing methods have tried to improve the accuracy of diagnostic results by fully exploiting the students' explicit response records (e.g., answering correctly or not).

However, many previous models face challenges with the "diagnostic system cold-start" problem. For instance, in online platforms, it is common to launch new businesses, e.g., *coursera.org* plans to release a series of new test exercises. For the new domain, there are no student-exercise interaction records available. Hereby, the diagnostic performance of previous approaches is often impaired as they only address the CD task in mature source domains where student-exercise interaction data are available. In this paper, we call the diagnostic system cold-start task as domain-level zero-shot cognitive diagnosis (DZCD). Different from previous studies on students or exercises cold-start for CD within a well-established source domain where interaction records are available [22, 33], in DZCD, students partially overlap across domains, and the mature source domains have rich student response records but the zero-shot target domain is brand new without student-exercise interactions. The DZCD is an important and practical task, typical applications include: (1) it is necessary to diagnose in advance when an online learning system intends to launch a new business; (2) students' behavior data in the target domain are unavailable due to collection limitations like privacy protection policy. Nevertheless, to the best of our knowledge, there is a severe lack of research on DZCD.

To provide reliable cognitive diagnosis for a zero-shot domain, inspired by the success of cross-domain modeling in various fields (e.g., recommender systems [25, 54]), one possible way is to define common student state characteristics by analyzing their past behaviors from a few accessible source domains, and represent the test exercises in the target domain using available features. The primary obstacle is to locate an appropriate mediator that can transmit student states between the established and target domains, enabling the execution of DZCD [54]. Some related studies have attempted

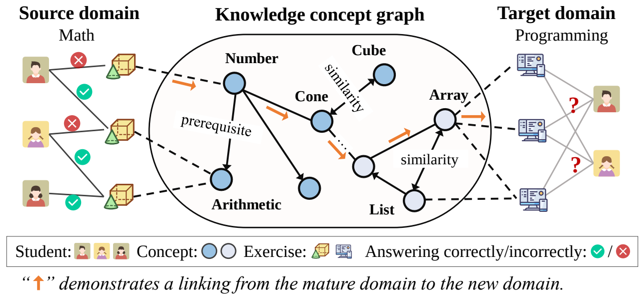

Figure 1: The example of a knowledge concept graph (KCG) connecting isolated exercises in each domain.

to utilize exercises' textual contents as the intermediary by learning universal and cross-domain exercise embeddings [22, 35]. However, there are two main drawbacks to these approaches. First, exercises' textual features may not accurately reflect the true meaning of the exercise due to linguistic bias [24]. For example, two exercises from course *Math* and course *Programming* may have the same description *"Calculate the circle's area"*, but they are testing different concepts, i.e., *Geometry* and *Programming Language*. Second, to proficiently adjust to diverse domains, the exercise text encoder necessitates domain-specific guidance, which has the potential to overfit and obstruct the transmission of cognitive signals between distinct domains [48, 54]. Therefore, it is desirable to find a more suitable intermediary to connect different domains.

In this paper, we employ a pedagogical knowledge concept graph (KCG) as the intermediary to facilitate the sharing of student cognitive states across different domains. The underlying rationale is that the KCG has the potential to connect different domains which can be a bridge to propagate student cognitive states. To elaborate, a KCG comprises numerous educational dependencies (as relations) to link knowledge concepts (as entities), which has been widely used in AI for Education [3, 34, 45]. Figure 1 illustrates an example of KCG with some educational dependency relations, e.g., the similarity relation links concept *Cone* and concept *Cube* since they belong to the same topic *Geometry*, while *Number* is the prerequisite concept of *Arithmetic* as the former is the learning basis of the latter logically. Obviously, the KCG has the capability to bridge different domains if it covers the knowledge concepts and associated exercises in each domain. For example, for two course-level domains, the *Math* (source domain) and *Programming* (target domain), each of their exercises associates at least one knowledge concept. This allows the two domains to be connected through the KCG as long as their associated concepts occur in the KCG, even though their exercises have no direct overlap. Thus, introducing a KCG as the intermediary across domains to propagate student cognitive states is promising, but also significantly challenging. Ideally, a proper KCG model for CD should have four essential properties: (1) *diagnosis-oriented*: the model can perform the CD task in the DZCD setting; (2) *student state propagation*: the KCG model should extract universal and transferable information for student embeddings so that student cognitive signals can be shared across domains; (3) *domain adaption*: for any cold-start domain which needs diagnosis, the model is expected to be domain-adaptive. (4) *application*: the diagnostic results can effectively support further intelligent services.

Motivated by the above considerations, we propose a general **T**ransferable knowledg**E** **C**oncept grap**H** framework to perform the domain-level zero-shot **C**ognitive **D**iagnosis (abbreviated as **TechCD**). The TechCD framework consists of two stages: the knowledge concept graph embedding (KCGE) stage and the domain adaptive diagnosis (DAD) stage. In the KCGE stage, a naive yet effective graph convolutional network (GCN) [49] is first employed over the KCG for representation learning by iteratively fusing neighboring aggregations in the KCG. To take full advantage of the connections between exercises and the KCG, we treat each exercise as part of the KCG for joint modeling with concept entities, so that the exercises can absorb structural information from the graph as their semantic descriptions. The most difficult aspect in this stage is to guarantee the *student state propagation* property. To this end, inspired by [54], we construct transferable student states by discarding the bottom layers of GCN (specific patterns) and only aggregating high-level ones (universal patterns) so that cognitive signals can be successfully propagated to other domains. We build domain-specific exercises and concepts by incorporating comprehensive semantic information from the KCG so that their embeddings comprise both universal and specific patterns, which ensures *domain adaptation.* In the DAD stage, the above embeddings are further fused to construct the traits of students (i.e., proficiency) and exercises (e.g., difficulty) by predicting student performance. In this way, the traits of students and exercises that need to be diagnosed can be refined satisfying the *diagnosis-oriented* property. It is worth mentioning that our general TechCD framework is well defined to be implemented by combining with existing CD solutions. For instance, we can have Tech-IRT by combining with IRT [9], Tech-MIRT with MIRT [30] and Tech-NeuralCD with NeuralCD [41], respectively. Finally, we conduct extensive experiments on four real-world datasets. The experimental results not only prove that TechCD is more effective in zero-shot student performance prediction since it can well capture the universal students' cognitive signals for propagation but also show the superior *application* property of the TechCD. For instance, TechCD can facilitate some personalized learning guidance such as exercise recommendation in cold-start scenarios.

2 RELATED WORK

2.1 Cognitive Diagnosis

Cognitive diagnosis (CD) is a fundamental task in many real-world scenarios such as games [4], medical diagnosis [47], and especially, education [10, 23]. The key spirit of CD is that it can be used to profile students' latent cognitive proficiency on specific knowledge concepts, as well as be applied to reveal characteristics of the test exercises such as difficulty and discrimination [9, 37] via exploiting student testing logs. These refined trait features could be applied to many intelligent applications, such as exercise recommendation [19, 46] and learning path suggestions [18, 39]. In the early years, cognitive diagnosis was mostly developed from the psychometric assumption that student cognitive states are stable in a short period of time (e.g., an exam) and thus can be diagnosed [10]. In general, these methods devote much effort to the design of student-exercise interaction functions, which are expected to automatically infer students' knowledge states. For instance, Item Response Theory (IRT) [9], Multidimensional IRT (MIRT) [30] and Deterministic Inputs, Noisy-And gate (DINA) [5] model the interaction of students and exercises linearly (e.g., leveraging the logistic-like function). Based on these traditional methods, some researchers introduce deep learning into cognitive diagnosis. For instance, Neural Cognitive Diagnosis (NeuralCD) [41] and Deep-IRT [38] exploit neural networks to learn the interaction function and trait embeddings automatically. Recently, to alleviate issues of student or exercise cold-start, and data sparsity in real-world scenarios, some studies have also considered incorporating exercise texts [22], the conceptual relations [10, 21] and more exceptions (e.g., slip and guess) [23] in students' learning process to enhance the interactive relations between students and exercises. However, to the best of our knowledge, research on how to cold-start a CD system remains unsolved.

2.2 Cold-Start Intelligent Systems

Cold-starting an intelligent system without historical interactions available for new users or items is a prevalent and practical concern in many domains [15, 32, 33, 51, 52]. This paper focuses on the task of cold-starting a cognitive diagnosis system in a zero-shot domain, which is of paramount importance to understanding the first batch of students' learning process, analyzing their knowledge proficiency and further helping improve equity in education [12]. To tackle this issue, many strategies have been utilized such as meta-learning [40], cross-domain modeling [25, 54] and reinforcement learning [7]. We pay attention to the idea of cross-domain modeling, which aims to characterize student state features based on their historical behaviors from some available source domains and represent the test exercises in the target domain with available features. The key challenge is to find a suitable intermediary to connect the mature and target domains. Some related studies on student performance prediction tasks [22, 35] utilize exercises' textual contents as the intermediary by learning universal and cross-domain exercise embeddings. However, these methods may be limited due to linguistic bias and cannot adapt to different domains effectively.

2.3 Pedagogical Knowledge Concept Graph

A pedagogical KCG contains numerous educational dependencies (as relations) to connect knowledge concepts (as entities). In general, the dependencies are constructed manually by domain experts or automatically through data-driven algorithms based on pedagogical prior knowledge [28], Among them, the most significant and common dependencies include similarity [26], collaboration [17], prerequisite [3], remedial [34] and hierarchy [21]. For example, a pair of concepts involved in the same topic or area or overlapping in some knowledge can be assigned with similarity dependency relations. Recently, some KCGs have been established in both academia and industry such as OpenEduKG[1] and SongshuAI KCG[2]. On the basis of KCGs, researchers attempt to incorporate them into many educational application tasks and obtain significant improvements [10, 26, 36]. Our TechCD properly incorporates a tailored pedagogical KCG into CD linking each domain so as to mitigate the domain-level zero-shot issue.

3 PRELIMINARIES

3.1 Cognitive Diagnosis Model

We first briefly introduce cognitive diagnosis models (CDMs). CDMs are developed to discover student proficiency levels on specific

[1]https://open.edukg.cn
[2]https://www.songshuai.com/education

knowledge concepts as well as exercise traits (e.g., difficulty) through fully exploiting their responses to several exercises [37]. Due to the real proficiency of students cannot be quantified explicitly, almost all of the previous CDMs are trained through the student performance prediction task, i.e., $\mathcal{F}_{\mathcal{CDM}}(\boldsymbol{u},\boldsymbol{v}) \rightarrow \hat{y}_{uv}$, where $\boldsymbol{u}$ and $\boldsymbol{v}$ are the latent traits of students and exercises, $\mathcal{F}_{\mathcal{CDM}}(\cdot)$ is the diagnostic interaction function, and $\hat{y}_{uv}$ is the predicted performance score. These traits of students and exercises can be refined with the optimization target of minimizing the difference between the predicted probability $\hat{y}_{uv}$ and the true response y_{uv} [10].

Generally, the differences between CDMs consist of the design of $\mathcal{F}_{\mathcal{CDM}}(\cdot)$ and the representations of trait $\boldsymbol{u}$ and $\boldsymbol{v}$. For example, IRT [9] uses single-dimension variables to represent the trait features and logistic-like function as the interaction function: $P(y_{ij}|\theta_i, a_j, b_j) = \frac{1}{1+e^{-1.7a_j(\theta_i - b_j)}}$, where θ_i characterizes student i's knowledge proficiency, and a_j and b_j represent exercise j's discrimination and difficulty. NeuralCD [41] exploits neural networks to fit the interaction function automatically: $\hat{y}_{ij} = F(\boldsymbol{u}_i, \boldsymbol{v}_j, \Theta_{CD})$, where $\boldsymbol{u}$ and $\boldsymbol{v}$ are the latent traits of students and exercises respectively, and $F(\cdot)$ is multi-layer neural networks. To summarize, we have the following general form of CDMs:

$$\hat{y}_{uv} = \mathcal{F}_{\mathcal{CDM}}(\boldsymbol{u}, \boldsymbol{v}, \Theta^*), \tag{1}$$

where Θ^* is the model parameter. To be noticed that, to ensure psychometric interpretability of prediction, CDMs should strictly follow the *Monotonicity* assumption [37]: the probability of correctly answering the exercise monotonically increases with student knowledge proficiency, i.e., $\frac{\partial \mathcal{F}}{\partial \boldsymbol{u}} > 0$.

3.2 Knowledge Concept Graph

A pedagogical KCG contains knowledge concept entities and conceptual dependency relations, whereas, in the domain-level cold-starting settings, it additionally includes exercise entities and exercise-concept association relations.

Definition 1 (Knowledge concept graph). *Formally, the Knowledge concept graph (KCG) can be represented as $\mathcal{G} = \{\mathcal{E}, \mathcal{R}, \mathcal{P}\}$. $\mathcal{E}$ is the set of entities including knowledge concept sets C and their associated exercises. $\mathcal{R}$ is the set of relations including educational dependency relations between concepts (e.g., prerequisite and similarity) and association relations between exercises and concepts. $\mathcal{P}$ is offered in the form of entity-relation-entity triplet set $\mathcal{P} = \{(h, r, t)|h, t \in \mathcal{E}, r \in \mathcal{R}\}$, e.g., (concept cube, similarity, concept cone) and (exercise e_1, association, concept function).*

3.3 Problem Definition

In the domain-level zero-shot cognitive diagnosis (DZCD) scenario, we represent the mature/available source domain as $\mathcal{S}$ and the cold-start/zero-shot target domain as $\mathcal{T}$. The student sets and exercise sets in source domain $\mathcal{S}$ are denoted as $\mathcal{U}_\mathcal{S}$, $\mathcal{V}_\mathcal{S}$, and in target domain $\mathcal{T}$ are denoted as $\mathcal{U}_\mathcal{T}$, $\mathcal{V}_\mathcal{T}$, where $\mathcal{U}_\mathcal{T} \subset \mathcal{U}_\mathcal{S}$ and $\mathcal{V}_\mathcal{S} \cap \mathcal{V}_\mathcal{T} = \emptyset$. All student-exercise performance records for training (with label) are collected from the source domain, depicted as $L_\mathcal{S} = \{(u_i, v_j, y_{ij})|y_{ij} \in \{0,1\}, u_i \in \mathcal{U}_\mathcal{S}, v_j \in \mathcal{V}_\mathcal{S}\}$, where $y_{ij} = 1$ represents student u_i answers exercise v_j correctly, and $y_{ij} = 0$ otherwise. The student-exercise interactions from the target domain (without label), i.e., $L_\mathcal{T} = \{(u_i, v_j)|u_i \in \mathcal{U}_\mathcal{T}, v_j \in \mathcal{V}_\mathcal{T}\}$, are used to evaluate prediction performance in DZCD scenarios. Hereby, our TechCD model for the DZCD task is defined as:

Definition 2 (Domain-level zero-shot cognitive diagnosis). ***Given** student exercising records $L_\mathcal{S}$ in the source domain and the KCG, $\mathcal{G}$, the **goal** of TechCD for the DZCD task is to make the diagnosis on student and exercise traits in the target domain $\mathcal{T}$ through fully exploiting student-exercise interactive records $L_\mathcal{S}$ in the source domain $\mathcal{S}$ with student performance predictions.*

4 THE TECHCD FRAMEWORK

4.1 Framework Overview

Conducting cognitive diagnosis in domain-level zero-shot settings is non-trivial. It presents a critical challenge in learning portable and transferable student embeddings from their exercising performance records in the source domain. To overcome this problem, we propose a TechCD framework that incorporates a tailored pedagogical knowledge concept graph (KCG) as a bridge between the source and target domains. Our proposed TechCD framework consists of two stages: knowledge concept graph embedding (KCGE) and domain adaptive diagnosis (DAD). The KCGE stage (detailed in Section 4.2) learns entity embeddings from the semantic and structural information of the KCG. The DAD stage (detailed in Section 4.3) then conducts diagnosis by predicting student exercise performance. The entire structure of TechCD is depicted in Figure 2.

In the KCGE stage, the critical obstacle is to propagate student cognitive states from the source domain to the target domain. For this goal, we customize a KCG as the intermediary to link exercises in various domains. We use a straightforward but effective graph convolutional network (GCN) [49, 54] on the KCG to construct transferable student cognitive embeddings that transcend the exercise-related performance confined to the source domain. Besides learning transferable student embedding, this stage generates specific embeddings of exercise and concept entities by integrating the structure and semantic information from the KCG.

With the above embeddings, the DAD stage further constructs student proficiency traits and exercise difficulty and discrimination traits for domain-adaptive cognitive diagnosis with existing diagnostic models. The entire model is trained through predicting student performance on exercises, i.e., $\hat{y}_{uv} = \mathcal{F}_{\mathcal{CDM}}(L_\mathcal{S}, \mathcal{G}, \Theta^*)$, where parameter Θ^* is optimized from the source domain $\mathcal{S}$ as:

$$\Theta^* = \arg\min_{\Theta} \mathcal{L}(y(L_\mathcal{S}), \mathcal{G}). \tag{2}$$

It is worth mentioning that the KCGE stage and the DAD stage are trained in an end-to-end fashion with the above Eq. (2). Thus, the refined traits of students and exercises can be the diagnostic results. After training, TechCD can conduct zero-shot student performance predictions in the zero-shot target domain $\mathcal{T}$.

4.2 Knowledge Concept Graph Embedding

This stage aims to identify universal student states present in exercises of the source domain that can be transferred to the zero-shot domain via the customized KCG. For this goal, we apply a multi-layer GCN network[3] over the KCG to learn entity embeddings.

Generally, the KCG contains concept and exercise entities, as well as multiple conceptual dependency relations and exercise-concept association relations, as shown in Figure 2 (a). The educational

[3]Actually, various KCG embedding techniques have been proposed to extract meaningful embeddings [42, 49]. Since our focus is not to devise more sophisticated techniques for graph network embedding, we simply use a popular GCN to learn entity representations, to verify the effectiveness of incorporating the KCG into the DZCD.

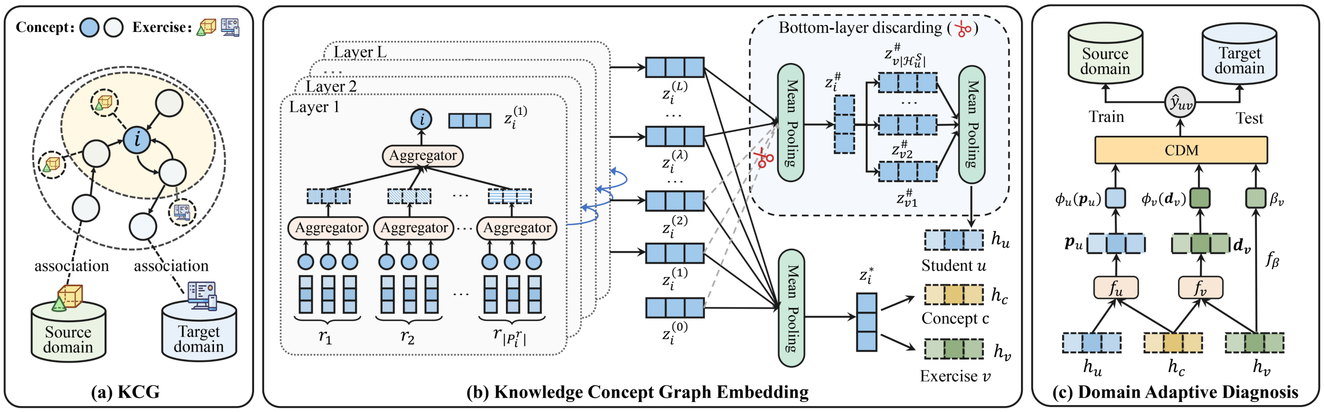

Figure 2: The overview architecture of TechCD: (a) a knowledge concept graph (KCG) links the source and the target domains; (b) the knowledge concept graph embedding for entity representation learning; (c) the domain adaptive diagnosis for DZCD.

dependency relations reflect student learning rules and knowledge transferring logically, which can implicitly propagate student states [8], while the exercise-concept associations can enhance exercise representations by absorbing structural and semantic information from the KCG. Thereby, for each entity embedding, it needs to discriminate different relations by separately fusing neighboring information of each type of relation. We directly use the learnable embedding $\boldsymbol{e}_i \in \mathbb{R}^d$ of each entity e_i as the input of GCN, i.e., $\boldsymbol{z}_i^{(0)} = c_i$, where d is the embedding dimensional size. We conduct convolution operation [44] of GCN over the KCG L times with each iteration considering each type of relation separately, to aggregate L hop neighborhood information and generate L entity embeddings, $[\boldsymbol{z}_i^{(1)}, \boldsymbol{z}_i^{(2)}, \ldots, \boldsymbol{z}_i^{(L)}]$, where $\boldsymbol{z}_i^{(l)} \in \mathbb{R}^d$ denotes the l-th layer output of entity e_i. The GCN iteratively aggregates neighboring information of each entity e_i for each type of relation separately to enhance its representation through the message-passing-receiving mechanism [13, 43] as follows:

$$\boldsymbol{z}_i^{(l)} = \sum_{r \in \mathcal{R}_i} \frac{1}{|\mathcal{P}_i^r|} \sum_{(e_j, r, e_i) \in \mathcal{P}_i^r} \mathcal{W}_r \boldsymbol{z}_j^{(l-1)}, \tag{3}$$

where $\mathcal{R}_i$ is the subset of $\mathcal{R}$ consisting of the relation types of entity e_i. $\mathcal{P}_i^r$ is the subset of $\mathcal{P}$ contains all the triplets (e_j, r, e_i) of entity e_i with relation r. For each relation r, we use a learnable matrix $\mathcal{W}_r \in \mathbb{R}^{d \times d}$ to transform each concept/exercise entity feature vector to the same free embedding space.

After obtaining the above refined entity embeddings, the focus is on representing transferable student embeddings to address the challenge of student state propagation. We resort to the bottom-discarding operation [54] which is naturally compatible. It argues that the bottom layers of GCN preserve more domain-specific information, while the upper layers better represent universal and transferable information. This is intuitively reasonable because increasing the number of GCN layers can lead to over-smoothing, resulting in the loss of discriminative information [16], which makes it promising to effectively propagate student cognitive signals to zero-shot domains. Thus, we discard the lower-level entity embeddings by settings a hyper-parameter λ to aggregate transferable embeddings $\boldsymbol{z}_i^{\#}$. Additionally, we fuse all layer output embeddings of GCN as well as the original entity embedding, resulting in a complete semantic representation $\boldsymbol{z}_i^*$, as Eq. (4).

$$\boldsymbol{z}_i^{\#} = \frac{1}{L - \lambda + 1} \sum_{l=\lambda}^{L} \boldsymbol{z}_i^{(l)}, \boldsymbol{z}_i^{*} = \frac{1}{L + 1} \sum_{l=0}^{L} \boldsymbol{z}_i^{(l)}. \tag{4}$$

Hereby, we construct the transferable cognitive state of student u by absorbing general knowledge from the KCG as:

$$\boldsymbol{h}_u = \frac{1}{|\mathcal{H}_u^{\mathcal{S}}|} \sum_{v \in \mathcal{H}_u^{\mathcal{S}}} \boldsymbol{z}_v^{\#}, \tag{5}$$

where $\mathcal{H}_u^{\mathcal{S}}$ is the exercise set that student u has interacted with in the source domain $\mathcal{S}$. We use $\boldsymbol{z}_v^{\#}$ to represent student states, as it captures entity-specific information in the bottom-layer output, while $\boldsymbol{z}_v^*$ contains more general high-order information.

Besides generating student embeddings, this stage also outputs exercise and knowledge concept representations. For the exercise entity v, we directly assign its corresponding embedding $\boldsymbol{z}_v^*$ from the entity e_v in the KCG to it similar to [54], i.e., $\boldsymbol{h}_v = \boldsymbol{z}_v^*$. Note that exercise embeddings incorporate both general and domain-specific information by absorbing full semantic representations of exercise entities in the KCG, which can fill in the domain-adaption requirements intuitively. Similarly, we represent each concept c's embedding $\boldsymbol{h}_c$ with its full semantic representation $\boldsymbol{z}_c^*$, i.e., $\boldsymbol{h}_c = \boldsymbol{z}_c^*$.

4.3 Domain Adaptive Diagnosis

In this stage, we conduct domain-adaptive cognitive diagnosis with existing cognitive diagnosis models.

4.3.1 ***Diagnosed Trait Representation Modeling.*** In general, a cognitive diagnosis model (CDM) takes the traits of students (i.e., proficiency) and exercises (e.g., difficulties and discrimination) as the basic input [41]. Thus, it is crucial to represent the above traits that need to be diagnosed via the generated embeddings from the KCGE stage. Inspired by [10, 21], to generate the proficiency factor on each concept of each student, we incorporate the embeddings of knowledge concept entities into the transferable student's embedding. Thus, the student proficiency trait can be modeled as:

$$\boldsymbol{p}_u = (p_{u1}; p_{u2}; \cdots; p_{u|C|}), \text{ where } p_{uc} = f_u(\boldsymbol{h}_u \oplus \boldsymbol{h}_c) \in (0, 1). \tag{6}$$

In the above Eq. (6), vector $\boldsymbol{p}_u$ is student u's proficiency on $|C|$ knowledge concepts in the KCG. Each element of $\boldsymbol{p}_u$, i.e., p_{uc}, denotes student u's mastery level on concept c. A full connection layer [50] $f_u(\cdot)$ is used to fuse knowledge concept semantics into the student embedding with concatenation $\oplus$. Similarly, we also fuse knowledge concept information into exercise embeddings to calculate each exercise v's difficulty trait $\boldsymbol{d}_v$ on all concepts with the full connection layer $f_v(\cdot)$ as Eq. (7).

$$\boldsymbol{d}_v = (d_{v1}; d_{v2}; \cdots; d_{v|C|}), \text{ where } d_{vc} = f_v(\boldsymbol{h}_v \oplus \boldsymbol{h}_c) \in (0,1). \quad (7)$$

Besides, the discrimination β_v of each exercise v is directly obtained by transforming the exercise embedding to a latent factor with a neural network $f_\beta(\cdot)$, i.e., scalar $\beta_v = f_\beta(\boldsymbol{h}_v) \in (0,1)$, similar to [21].

4.3.2 ***Diagnostic Adaptor.*** Different diagnostic models characterize student and exercise features in different forms. Our aim is to establish a connection between students' cognitive proficiency, exercise traits, and the input forms of existing diagnostic models, through introducing the diagnostic adaptor. In general, for the given student u and exercise v, the CDM adaptor predicts student performance score $\hat{y}_{uv}$ as Eq. (8):

$$\hat{y}_{uv} = \mathcal{F}_{\mathcal{CDM}}(\phi_u(\boldsymbol{p}_u), \phi_v(\boldsymbol{d}_v), \beta_v), \quad (8)$$

where $\mathcal{F}_{\mathcal{CDM}}(\cdot)$ represents the existing diagnostic model and can be specified with many models like IRT [9], MIRT [30], etc. To cover different diagnostic models, we employ two transform functions, $\phi_u(\cdot)$ and $\phi_v(\cdot)$, to standardize the form of student proficiency $\boldsymbol{p}_u$ and exercise difficulty $\boldsymbol{d}_v$ so as to satisfy the input form of the adopted model. Besides, to ensure the monotonicity assumption of cognitive diagnosis, we restrict each parameter of $\mathcal{F}_{\mathcal{CDM}}$ to be positive, so that $\frac{\partial \mathcal{F}_{\mathcal{CDM}}}{\partial \phi_u(\boldsymbol{p}_u)} > 0$. Eq. (8) can be used to infer students' performance on exercises in both the source and target domains.

Finally, we use the popular cross-entropy loss function to optimize the whole model by minimizing the difference between the predicted probability $\hat{y}_{uv}$ and the true response y_{uv}.

$$\mathcal{L} = -\sum_{(u,v,y_{uv}) \in L_{\mathcal{S}}} \left(y_{uv} \log \hat{y}_{uv} + (1 - y_{uv}) \log (1 - \hat{y}_{uv})\right). \quad (9)$$

By optimizing with the above loss, these input traits of the student and the exercise in Eq. (8), i.e., $\boldsymbol{p}_u$, $\boldsymbol{d}_v$ and β_v, can be jointly refined serving as the diagnostic results of students and exercises.

4.3.3 ***Instantiating the TechCD.*** Taking the student trait $\phi_u(\boldsymbol{p}_u)$ and exercise traits $\phi_v(\boldsymbol{d}_v)$ and β_v as input factors, we specify the diagnostic adaptor $\mathcal{F}_{\mathcal{CDM}}(\cdot)$ in Eq. (8) of TechCD with IRT, MIRT and NeuralCD as follows:

IRT [9] takes the unidimensional student proficiency, exercise difficulty and discrimination as input. To specify with IRT, we project $\boldsymbol{p}_u$ and $\boldsymbol{d}_v$ to scalars p_u and d_v respectively by setting ϕ_u and ϕ_v as mean pooling. The $\mathcal{F}_{\mathcal{CDM}}(\cdot)$ is a logistic-like function: $\hat{y}_{uv} = \text{sigmoid}(\beta_v \cdot (p_u - d_v))$.

MIRT [30] models the interaction between multidimensional student proficiency $\boldsymbol{p}_u$ and exercise difficulty $\boldsymbol{d}_v$ using a logistic-like function. We set the output dimensions of ϕ_u and ϕ_v as $D > 1$. The $\mathcal{F}_{\mathcal{CDM}}(\cdot)$ is shown as: $\hat{y}_{uv} = \text{sigmoid}(\boldsymbol{p}_u^T \boldsymbol{d}_v + \beta_v)$.

NeuralCD [41] directly takes student proficiency $\boldsymbol{p}_u$ and exercise difficulty as input. Additionally, it requires masking the irrelevant knowledge proficiency by a vector $\boldsymbol{Q}_v = \{0,1\}^{|C| \times 1}$ where $q_{v,c} = 1$ if exercise v associates concept c and $q_{v,c} = 0$ otherwise. C is the knowledge concept set. The $\mathcal{F}_{\mathcal{CDM}}(\cdot)$ is a multi-layer neural networks ϕ with non-negative weights to keep explainability: $\hat{y}_{uv} = \phi(\boldsymbol{Q_v} \circ (\boldsymbol{p}_u^T - \boldsymbol{d}_v) \cdot \beta_v)$, where $\circ$ is element-wise product.

Table 1: Some basic statistics of the datasets.

Datasets	CM	AM	Junyi	ASSIST
#Student	21,068	21,059	10,000	5,730
#Exercise	6,257	3,263	706	4,973
#Knowledge concept	1,251	990	706	122
#Record	351,146	171,380	353,835	225,314
#Record per student	16.7	8.1	35.4	39.3

5 EXPERIMENTS

We conduct comprehensive experiments to address the following research questions:

- **RQ1** Can the TechCD framework effectively handle the domain-level zero-shot cognitive diagnosis task?
- **RQ2** How about the effectiveness of modeling the KCG by the TechCD framework?
- **RQ3** Can the TechCD utilize the out-of-domain datasets for the performance improvement?
- **RQ4** How to apply TechCD to provide personalized guidance?

5.1 Datasets

5.1.1 Basic Description. We conduct experiments on the following four real-world representative datasets:

- Core Math (CM) and Advanced Math (AM) are two subsets of the *MATH-2021* dataset, collected supplied by iFLYTEK Co., Ltd., which is collected from the iFLYTEK Learning Machine[4]. They have overlapping students while their exercises have no overlap.
- Junyi[5] [2] contains student online learning logs on mathematical exercises which is crawled from a Chinese online learning platform. Nowadays Junyi is widely used in the evaluation of online education tasks [10, 21]. We randomly select 10,000 students' exercising records from Junyi for experiments.
- ASSISTments-2012-2013 (ASSIST)[6] is an open dataset collected by the ASSISTments online tutoring systems, which has become popular benchmark datasets for cognitive diagnosis. We randomly select about 5,000 exercises and their related records.

All the datasets provide student exercising records and exercise-concept correlations, where each exercise associates one knowledge concept. Besides, AM and CM provide the exercises' contents, and Junyi provides the conceptual prerequisite and similarity relations labeled by experts. Each dataset is treated as a domain, i.e., the source or target domain. Among them, there is no overlap between the students in the Junyi and ASSIST datasets and those in the MATH dataset. For each dataset, we reserve only the first attempt of each exercise for each dataset to ensure that the attribute state of students is static following the [10, 41]. We evaluate the performance of DZCD on the target domain using the refined model trained in the source domain. We split each source domain's dataset by randomly selecting two historical interactions from each student's logs for validation, with the remaining data serving as the training

[4] https://xxj.xunfei.cn/
[5] https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=1198
[6] https://drive.google.com/file/d/1cU6Ft4R3hLqA7G1rIGArVfelSZvc6RxY/view

Table 2: Some detailed statistics of the KCG.

Entity	#Concept	2,594
	#Exercise	15,199
	#Total	17,793
Relation	#Total	5
Triple	#Conceptual dependency	7,926
	#Exercise-concept association	15,469
	#Total	23,395

set, similar to the widely used *leave-one-out* evaluation [14, 31]. Besides, to train the Oracle models (Section 5.2), we also split the target domain's dataset into training (70%), validation (10%), and test sets (20%), similar to [54]. The basic statistics of the datasets are presented in Table 1.

5.1.2 Knowledge Concept Graph Construction. To bridge exercises across different domains, it needs to tailor a unified knowledge concept graph (KCG) linking each domain. For this purpose, we adopt a hierarchical mathematical KCG (abbreviated as MathKCG) to connect all domains (i.e., all the datasets). Specifically, MathKCG is published by the online education platform, i.e., Luna[7]. It covers 39.5% knowledge concepts in our datasets and provides two significant types of conceptual relations, i.e., hierarchy [21] and similarity [26] relations. We first align each exercise-related concept in datasets and concepts in MathKCG based on conceptual names. Then, for the isolated concepts in each dataset that cannot be linked to MathKCG, we build conceptual similar and prerequisite relations [3] between them via exploiting student performance logs using the statistical method [10]. Hereby, based on the generated relations, the MathKCG, and those relations provided by Junyi, each concept can be linked to a KCG. Additionally, the exercises are linked to their associated knowledge concept in the KCG. The final KCG includes concept and exercise entities, and four types of conceptual relations (i.e., hierarchy-in-MathKCG, similarity-in-MathKCG, the constructed similarity and prerequisite relations via our datasets) as well as the exercise-concept association relations.

We conduct all experiments on the same KCG. The detailed statistics of the KCG are presented in Table 2.

5.2 Baselines

To verify the effectiveness of our model, we present three implementations based on TechCD framework that combine typical diagnosis methods. In particular, we implement Tech-IRT, Tech-MIRT and Tech-NeuralCD following IRT, MIRT and NeuralCD, respectively.

- IRT [9]: IRT models unidimensional students and exercises' features with a logistic-like function.
- MIRT [30]: As the multidimensional extension of IRT, MIRT models multiple knowledge proficiency of students and exercises.
- NeuralCD [41]: NeuralCD is one of the most popular deep learning-based CD methods, which models high-order and complex student-exercise interaction functions with a multilayer perceptron (MLP).

We select a series of baselines for comparison. Among them, the random and oracle methods indicate the lower and upper bounds of performance, following the previous setups [54]. For each baseline (excluding Random), we also select IRT, MIRT and NeuralCD as their diagnostic functions. The details are listed as follows:

[7]https://luna.bdaa.pro

- Random: The random method predicts the students' scores randomly from $Uniform(0,1)$.
- Oracle: The oracle baseline is trained with the student-exercise interactive records of both source and target domains. Hence, it should perform better than other compared methods.
- NLP-based: Some related researches [22, 35] utilize exercises' textual contents as an intermediary of the source and target domain for student performance predictions. Thus, we adopt Bert [6] as the encoder to encode exercises' textual contents to generate their embeddings. To implement the NLP-based diagnosis method, we use learnable embeddings as student proficiency and introduce two functions to transform textual content features into exercises' difficulties and discrimination.
- GCN-based: We add a baseline that utilizes only the last-layer output as entities' embeddings and does not differentiate between the different relations in the KCG for comparison.

5.3 Evaluation Metrics and Other Settings

5.3.1 Metrics. To evaluate model performance, we adopt different metrics from the perspectives of classification and regression following the [10]. From the classification perspective, a student answering incorrectly or correctly can be represented as a negative (0) or positive (1) instance respectively. Thus, we use *Accuracy* (ACC) and *Area Under the ROC Curve* (AUC) for measuring. From the regression perspective, we select *Root Mean Square Error* (RMSE) to quantify the distance between the predicted score (i.e., the probability that a student answers correctly) and the actual one.

5.3.2 Implementation Details. For those models that employ NeuralCD and MIRT as diagnostic functions, we set the dimensions of student and exercise vectors as the number of diagnosed knowledge concepts $|C|$, similar to [41]. The dimensions of neural network layers are 1024 and 512 for all models with NeuralCD diagnostic function. Regarding the GCN layers, under the "AM as source" setting, we use 5 layers for L and a discarding parameter λ of 3. Under the "CM as source" setting, we use 5 layers for L and a discarding parameter λ of 2. For training, all network parameters are initialized with Xavier initialization [11]. Furthermore, we set the mini-batch size as 256 and the learning rate as 0.0005 for each model. Each model is implemented by PyTorch [29] and optimized by Adam optimizer [20]. All experiments are run on a Linux server with two 3.00GHz Intel Xeon Gold 5317 CPUs and one Tesla A100 GPU. The code is available at https://github.com/bigdata-ustc/TechCD.

5.4 Student Performance Prediction (RQ1)

To answer RQ1, we compare the performance of our model with several baselines on the domain-level zero-shot student performance prediction task. We switch CM and AM datasets as the target domain since their students overlap. It is worth mentioning that Junyi and ASSIST are used in Section 5.6 to demonstrate how TechCD utilizes out-domain datasets from other platforms for the DZCD task, as they are collected from different platforms. The overall prediction performance is reported in Table 3. The combination of S-CM (AM) and T-AM (CM) denotes CM (AM) as the source domain for training and AM (CM) as the target domain for testing. We have the following observations: (1) For different diagnostic implementations (i.e., IRT, MIRT and NeuralCD as Diagnostic function), our proposed TechCD framework almost outperforms all baseline models (including Random, NLP-based and GCN-based

Table 3: Performance comparison. The best zero-shot student performance prediction is highlighted in bold, the runner-up is underlined, and ↑ (↓) means the higher (lower) score the better performance, the same as below. * indicates the oracle result.

Dataset	Metric	IRT				MIRT				NeuralCD				Random
		Oracle	NLP	GCN	TechCD	Oracle	NLP	GCN	TechCD	Oracle	NLP	GCN	TechCD	
S-CM T-AM	ACC (%) ↑	77.89*	59.84	56.72	**63.45**	73.83*	56.44	56.74	**64.73**	74.65*	56.44	57.05	**57.06**	50.13
	AUC (%) ↑	84.98*	65.32	56.62	**67.42**	79.26*	65.52	56.60	**68.90**	81.07*	57.09	**57.44**	53.68	50.14
	RMSE (%) ↓	38.91*	47.98	50.75	**47.59**	48.40*	48.30	50.79	**47.06**	41.17*	49.69	50.72	**49.49**	57.70
S-AM T-CM	ACC (%) ↑	77.67*	55.88	56.92	**57.72**	74.07*	55.88	56.92	**57.78**	74.34*	55.88	56.80	**56.99**	49.91
	AUC (%) ↑	85.50*	50.68	56.62	**58.99**	81.16*	**60.56**	56.62	59.02	81.61*	53.67	**57.55**	52.40	49.89
	RMSE (%) ↓	39.08*	53.21	54.46	**52.85**	47.93*	**48.52**	50.50	52.85	41.52*	49.87	50.72	**49.57**	57.78

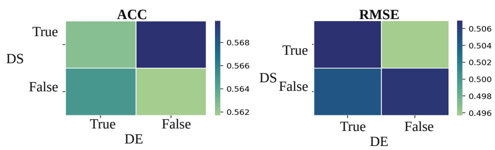

Figure 3: The ACC and RMSE comparisons on S-AM and T-CM. The darker (lighter) means the better for ACC (RMSE).

models) on both CM and AM target domains, which indicates the effectiveness of TechCD on predicting student performance under the cold-start setting. (2) Both GCN-based and TechCD employ the knowledge concept graph linking both source and target domains. However, GCN-based methods are unable to discard bottom-layer information and discriminate different relations in the KCG. In contrast, TechCD outperforms GCN-based methods, which positively supports its effectiveness.

In the following parts, we primarily present the experimental results of Tech-NeuralCD as the representative ones, since other diagnosis functions can be abstracted as the special cases of NeuralCD [41].

5.5 Bottom-Layer Discarding Analysis (RQ2)

The TechCD framework relies on the bottom-layer discarding operation [54] to generate transferable embeddings. We refer to the operation of discarding bottom-layer embedding of student and exercise embeddings as DS and DE, respectively. To evaluate the impact of this operation, we perform various experiments with different combinations of DS and DE. The comparisons of ACC and RMSE scores under the setting of S-AM and T-CM are visualized in Figure 3. The experimental results indicate that the best performance is achieved by only discarding the bottom-layer output from the KCG for students (DS), highlighting the effectiveness of extracting transferable information. However, when both DS and DE are used simultaneously, the performance is weakened, emphasizing the importance of maintaining specific patterns for exercises.

5.6 Improving with Out-Domain Datasets (RQ3)

The tailored KCG can link different domains including those within the same platform and those across platforms. The previous experiments focus on evaluating performance within source and target domains that share overlapping students. This part shows how powerful is TechCD for utilizing out-domain datasets from other platforms under two typical cold-start scenarios [54].

5.6.1 Accessible Student Records (ASD). In the scenario, student performance records $L_{\mathcal{S}}$ in the source domain $\mathcal{S}$ and out-domain records $L_{\mathcal{O}}$ in the target domain $\mathcal{O}$ are both available. Thus, $L_{\mathcal{S}}$ and

Table 4: Performance of TechCD trained on different settings.

Training	Target	ACC (%) ↑	AUC (%) ↑	RMSE (%) ↓
Random	AM	50.13	50.14	57.70
CM	AM	57.06	**53.68**	**49.49**
(LA) Junyi	AM	53.71	50.49	49.80
(LA) Assist	AM	54.83	49.77	49.85
(ASD) CM+Junyi	AM	56.60	52.10	49.84
(ASD) CM+Assist	AM	**57.08**	51.95	49.69
(ASD) CM+Junyi+Assist	AM	56.73	52.11	49.57

$L_{\mathcal{O}}$ can be used to jointly train the model with Eq. (2) as:

$$\Theta^* = \arg\min_{\Theta} \mathcal{L}(y(L_{\mathcal{S}} + L_{\mathcal{O}}), \mathcal{G}). \tag{10}$$

5.6.2 Limited Access (LA). In the setting, student performance records are unavailable due to privacy protection policies. To address this scenario, the out-domain $\mathcal{O}$ are introduced to refine the KCG by replacing source domain's datasets $L_{\mathcal{S}}$ with out-domain datasets $L_{\mathcal{O}}$ in Eq. (2) as:

$$\Theta^* = \arg\min_{\Theta} \mathcal{L}(y(L_{\mathcal{O}}), \mathcal{G}). \tag{11}$$

Table 4 lists the performance of Tech-NeuralCD, indicating the following observations. In the ASD setting, the out-domain datasets can partly improve the prediction performance of TechCD. In the LA setting, with the out-domain datasets, TechCD can get a promising performance compared with random predictions. These findings confirm the KCG can absorb out-domain datasets effectively.

5.7 Popular Applications of TechCD (RQ4)

The above experiments have proved that TechCD can complete the DZCD task effectively. In this part, we demonstrate two special applications of our TechCD that are in need of industrial practice.

5.7.1 Diagnostic Report Generation. Providing diagnostic reports to students via the CD method is one of the most typical intelligent applications in intelligent education, which can help students understand their learning process. Traditional diagnosis methods diagnose students' proficiency on knowledge concepts limited in the source domain, while our TechCD can further infer students' cognitive states in the target domain. We randomly select one student in the CM datasets to generate her diagnostic reports using Tech-NeuralCD and traditional NeuralCD trained on the CM dataset. We also sample a subgraph of KCG which covers some knowledge concepts of CM and Junyi with similarity and prerequisite relations. Figure 4 (a) and (b) present diagnostic reports of both models and

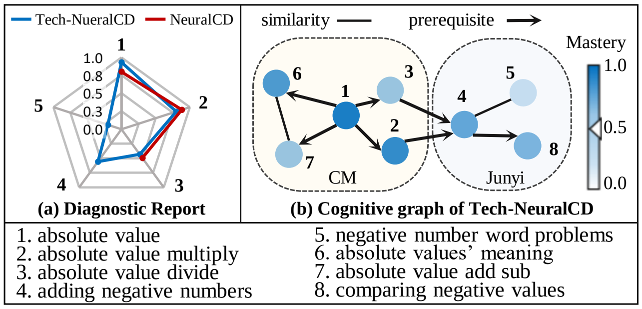

Figure 4: The example of diagnostic reports.

the cognitive graph of Tech-NeuralCD respectively. From the figure, we observe that: (1) For the mastery levels of knowledge concepts (1, 2 and 3) in CM, Tech-NeuralCD and NeurcalCD can output similar diagnosis results, indicating that both models can perform well on in-domain datasets. (2) For those knowledge concepts (4 and 5) sampled from Junyi (as a cold-start domain), NeuralCD is unable to provide a diagnosis, while Tech-NeuralCD is still able to perform effectively. (3) In the cognitive graph of Tech-NeuralCD, the deeper the color of the concept entity, the higher its cognitive level. We find the diagnosis results reasonable and interpretable. For example, the student's proficiency on concepts 3 and 4 is poor, which is reflected in her poor mastery of concept 5. This is expected as mastering concepts 3 and 4 are prerequisites for learning concept 5. Additionally, the mastery levels on concepts 6 and 7 are similar as they belong to the same topic (i.e., absolute value).

5.7.2 Exercise Recommendation. The diagnostic results can be utilized to suggest appropriate exercises to students, rather than relying on their own search efforts. A proper recommender system generally takes into account two key objectives: (*O1: smoothness*) the difficulty levels of a series of recommendations should avoid drastic variations as students learn knowledge gradually [53]; (*O2: engagement*) the recommendations should not be too challenging or easy to keep students' enthusiasm [18]. For these goals, we implement a simple yet effective strategy[8] to recommend x exercises for each student. Concretely, with a refined CDM, we first predict each student's performance on each exercise as Eq. (8). All exercises can be divided into two sets that answer correctly (positive samples) or not (negative samples) according to prediction results. Then, we sample $\frac{x}{2}$ exercises from each of the positive and negative samples. For each sampling, we require the selected exercise's difficulty to be close to a threshold (0.5 in this paper) to ensure the *smoothness* objective. Finally, we can get the recommendation lists for each student, which satisfy the above objectives.

We conduct recommendations on the challenging target domain that traditional CDMs are unable to handle. Table 5 lists ten exercise recommendations on T-AM for a randomly selected student using the refined Tech-NeuralCD model trained on S-CM dataset. The table also includes the diagnosed exercise difficulties and student mastery levels of the associated concepts, as well as the student's true performance on the exercises as recorded in the T-AM dataset. We can see that: (1) The recommended exercises are tailored to the student's proficiency, neither too easy nor too difficult. Some of them will challenge the student, while others will serve as "gifts"

[8]TechCD can support many complex and popular exercise recommendation approaches like [1, 18], this part uses the simple recommendation method as an example.

Table 5: Exercise recommendation of TechCD.

S-CMT-AM	①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩
Exercise id	232	1,632	2,432	30	123	2,003	3,020	175	220	250
Mastery (%)	67.23	38.24	40.07	23.00	48.63	57.30	84.33	54.27	48.24	57.78
Difficulty (%)	50.20	51.30	49.93	50.21	49.98	50.00	50.03	50.10	49.99	49.93
Performance	✓	×	×	×	✓	✓	✓	×	✓	✓

that can help increase her engagement with the material. (2) For exercises that the student answers correctly (incorrectly), the proficiency of the corresponding concept is almost higher (lower) than the exercise's difficulty, indicating that students answer correctly when their proficiency meets the difficulty. It confirms TechCD's diagnoses are effective in the cold-start domain.

6 CONCLUSION

This paper presents a study on the domain-level zero-shot cognitive diagnosis (DZCD) task. DZCD is an important task for the lack of student behavior data in the target domain due to the absence of student-exercise interactions or unavailability of exercising records for training. To tackle this, we propose a general and transferable framework TechCD that utilizes a pedagogical knowledge concept graph (KCG) to connect different domains and propagate students' universal cognitive states. The learned student embeddings by TechCD are transferable, while the exercise embeddings are domain-specific, enabling TechCD to perform domain-adaptive zero-shot cognitive diagnosis in the target domain. Finally, extensive experiments on real-world datasets not only prove that TechCD can effectively make the cognitive diagnosis task for a zero-shot domain and outperform several alternative baselines, but also show the superior application potential such as personalized exercise recommendation of TechCD. In our future research, we will focus on developing more advanced methods for constructing educational KCGs that can better connect different domains. Additionally, we plan to explore more sophisticated approaches for integrating conceptual relationships to further improve TechCD's performance in the DZCD scenario. Ultimately, we hope that our work will inspire and inform future studies and applications in this area.

Acknowledgements. This research was partially supported by grants from the National Key Research and Development Program of China (No. 2021YFF0901003), National Natural Science Foundation of China (No. 62202443), and Open Research Fund of the State Key Laboratory of Cognitive Intelligence (iED2022-002).

REFERENCES

[1] Haoyang Bi, Haiping Ma, Zhenya Huang, Yu Yin, Qi Liu, Enhong Chen, Yu Su, and Shijin Wang. 2020. Quality meets Diversity: A Model-Agnostic Framework for Computerized Adaptive Testing. In *2020 IEEE International Conference on Data Mining (ICDM)*. IEEE, 42–51.
[2] Haw-Shiuan Chang, Hwai-Jung Hsu, and Kuan-Ta Chen. 2015. Modeling Exercise Relationships in E-Learning: A Unified Approach.. In *EDM*. 532–535.
[3] Penghe Chen, Yu Lu, Vincent W Zheng, and Yang Pian. 2018. Prerequisite-driven deep knowledge tracing. In *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 39–48.
[4] Shuo Chen and Thorsten Joachims. 2016. Predicting matchups and preferences in context. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 775–784.
[5] Jimmy De La Torre. 2009. DINA model and parameter estimation: A didactic. *Journal of educational and behavioral statistics* 34, 1 (2009), 115–130.
[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[7] Nan Ding and Radu Soricut. 2017. Cold-start reinforcement learning with softmax policy gradient. *Advances in Neural Information Processing Systems* 30 (2017).

[8] Henry C Ellis. 1965. The transfer of learning. (1965).

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

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

[11] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth International conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 249–256.

[12] Margaret Grogan. 1999. Equity/equality issues of gender, race, and class. *Educational Administration Quarterly* 35, 4 (1999), 518–536.

[13] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 639–648.

[14] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th International conference on world wide web*. 173–182.

[15] Minlie Huang, Xiaoyan Zhu, and Jianfeng Gao. 2020. Challenges in building intelligent open-domain dialog systems. *ACM Transactions on Information Systems (TOIS)* 38, 3 (2020), 1–32.

[16] Wenbing Huang, Yu Rong, Tingyang Xu, Fuchun Sun, and Junzhou Huang. 2020. Tackling over-smoothing for general graph convolutional networks. *arXiv preprint arXiv:2008.09864* (2020).

[17] Xiaoqing Huang, Qi Liu, Chao Wang, Haoyu Han, Jianhui Ma, Enhong Chen, Yu Su, and Shijin Wang. 2019. Constructing Educational Concept Maps with Multiple Relationships from Multi-Source Data. In *2019 IEEE ICDM*. IEEE, 1108–1113.

[18] Zhenya Huang, Qi Liu, Chengxiang Zhai, Yu Yin, Enhong Chen, Weibo Gao, and Guoping Hu. 2019. Exploring multi-objective exercise recommendations in online education systems. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 1261–1270.

[19] Yujia Huo, Derek F Wong, Lionel M Ni, Lidia S Chao, and Jing Zhang. 2020. Knowledge modeling via contextualized representations for LSTM-based personalized exercise recommendation. *Information Sciences* 523 (2020), 266–278.

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

[21] Jiatong Li, Fei Wang, Qi Liu, Mengxiao Zhu, Wei Huang, Zhenya Huang, Enhong Chen, Yu Su, and Shijin Wang. 2022. HierCDF: A Bayesian Network-based Hierarchical Cognitive Diagnosis Framework. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 904–913.

[22] Qi Liu, Zhenya Huang, Yu Yin, Enhong Chen, Hui Xiong, Yu Su, and Guoping Hu. 2019. Ekt: Exercise-aware knowledge tracing for student performance prediction. *IEEE Transactions on Knowledge and Data Engineering* 33, 1 (2019), 100–115.

[23] Qi Liu, Runze Wu, Enhong Chen, Guandong Xu, Yu Su, Zhigang Chen, and Guoping Hu. 2018. Fuzzy cognitive diagnosis for modelling examinee performance. *ACM Transactions on Intelligent Systems and Technology (TIST)* 9, 4 (2018), 1–26.

[24] Ye Liu, Han Wu, Zhenya Huang, Hao Wang, Jianhui Ma, Qi Liu, Enhong Chen, Hanqing Tao, and Ke Rui. 2020. Technical phrase extraction for patent mining: A multi-level approach. In *2020 IEEE International Conference on Data Mining (ICDM)*. IEEE, 1142–1147.

[25] Nima Mirbakhsh and Charles X Ling. 2015. Improving top-n recommendation for cold-start users via cross-domain information. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 9, 4 (2015), 1–19.

[26] Hiromi Nakagawa, Yusuke Iwasawa, and Yutaka Matsuo. 2019. Graph-based Knowledge Tracing: Modeling Student Proficiency Using Graph Neural Network. In *2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. IEEE, 156–163.

[27] Tuan Nguyen. 2015. The effectiveness of online learning: Beyond no significant difference and future horizons. *MERLOT Journal of online learning and teaching* 11, 2 (2015), 309–319.

[28] Liangming Pan, Chengjiang Li, Juanzi Li, and Jie Tang. 2017. Prerequisite relation learning for concepts in moocs. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1447–1456.

[29] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* 32 (2019).

[30] Mark D Reckase. 2009. Multidimensional item response theory models. In *Multidimensional item response theory*. Springer, 79–112.

[31] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).

[32] Andrew I Schein, Alexandrin Popescul, Lyle H Ungar, and David M Pennock. 2002. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual International ACM SIGIR conference on Research and development in information retrieval*. 253–260.

[33] Robin Schmucker and Tom M Mitchell. 2022. Transferable Student Performance Modeling for Intelligent Tutoring Systems. *arXiv preprint arXiv:2202.03980* (2022).

[34] Yi Shang, Hongchi Shi, and Su-Shing Chen. 2001. An intelligent distributed environment for active learning. *Journal on Educational Resources in Computing (JERIC)* 1, 2es (2001), 4–es.

[35] Yu Su, Qingwen Liu, Qi Liu, Zhenya Huang, Yu Yin, Enhong Chen, Chris Ding, Si Wei, and Guoping Hu. 2018. Exercise-enhanced sequential modeling for student performance prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[36] Shan-Yun Teng, Jundong Li, Lo Pang-Yun Ting, Kun-Ta Chuang, and Huan Liu. 2018. Interactive unknowns recommendation in e-learning systems. In *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 497–506.

[37] Shiwei Tong, Jiayu Liu, Yuting Hong, Zhenya Huang, Le Wu, Qi Liu, Wei Huang, Enhong Chen, and Dan Zhang. 2022. Incremental Cognitive Diagnosis for Intelligent Education. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1760–1770.

[38] Emiko Tsutsumi, Ryo Kinoshita, and Maomi Ueno. 2021. Deep-IRT with Independent Student and Item Networks. *International Educational Data Mining Society* (2021).

[39] Kurt VanLehn. 2011. The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational psychologist* 46, 4 (2011), 197–221.

[40] Manasi Vartak, Arvind Thiagarajan, Conrado Miranda, Jeshua Bratman, and Hugo Larochelle. 2017. A meta-learning perspective on cold-start recommendations for items. *Advances in neural information processing systems* 30 (2017).

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

[42] Hao Wang, Enhong Chen, Qi Liu, Tong Xu, Dongfang Du, Wen Su, and Xiaopeng Zhang. 2018. A united approach to learning sparse attributed network embedding. In *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 557–566.

[43] Hao Wang, Defu Lian, Hanghang Tong, Qi Liu, Zhenya Huang, and Enhong Chen. 2021. Hypersorec: Exploiting hyperbolic user and item representations with multiple aspects for social-aware recommendation. *ACM Transactions on Information Systems (TOIS)* 40, 2 (2021), 1–28.

[44] Hao Wang, Tong Xu, Qi Liu, Defu Lian, Enhong Chen, Dongfang Du, Han Wu, and Wen Su. 2019. MCNE: An end-to-end framework for learning multiple conditional network representations of social network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1064–1072.

[45] Minjie Wang, Da Zheng, Zihao Ye, Quan Gan, Mufei Li, Xiang Song, Jinjing Zhou, Chao Ma, Lingfan Yu, Yu Gai, et al. 2019. Deep graph library: A graph-centric, highly-performant package for graph neural networks. *arXiv preprint arXiv:1909.01315* (2019).

[46] Zhengyang Wu, Ming Li, Yong Tang, and Qingyu Liang. 2020. Exercise recommendation based on knowledge concept prediction. *Knowledge-Based Systems* 210 (2020), 106481.

[47] Jie Xu, Cheng Deng, Xinbo Gao, Dinggang Shen, and Heng Huang. 2017. Predicting Alzheimer's disease cognitive assessment via robust low-rank structured sparse model. In *IJCAI: proceedings of the conference*, Vol. 2017. NIH Public Access, 3880.

[48] Linan Yue, Qi Liu, Yichao Du, Yanqing An, Li Wang, and Enhong Chen. 2022. DARE: Disentanglement-Augmented Rationale Extraction. *Advances in Neural Information Processing Systems* 35 (2022), 26603–26617.

[49] Si Zhang, Hanghang Tong, Jiejun Xu, and Ross Maciejewski. 2019. Graph convolutional networks: a comprehensive review. *Computational Social Networks* 6, 1 (2019), 1–23.

[50] Hao Zhao, Ming Lu, Anbang Yao, Yurong Chen, and Li Zhang. 2020. Learning to draw sight lines. *International Journal of Computer Vision* 128 (2020), 1076–1100.

[51] Hao Zhao, Ming Lu, Anbang Yao, Yiwen Guo, Yurong Chen, and Li Zhang. 2017. Physics inspired optimization on semantic transfer features: An alternative method for room layout estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 10–18.

[52] Hao Zhao, Ming Lu, Anbang Yao, Yiwen Guo, Yurong Chen, and Li Zhang. 2020. Pointly-supervised scene parsing with uncertainty mixture. *Computer Vision and Image Understanding* 200 (2020), 103040.

[53] Wayne Xin Zhao, Wenhui Zhang, Yulan He, Xing Xie, and Ji-Rong Wen. 2018. Automatically learning topics and difficulty levels of problems in online judge systems. *ACM Transactions on Information Systems (TOIS)* 36, 3 (2018), 1–33.

[54] Jianhuan Zhuo, Jianxun Lian, Lanling Xu, Ming Gong, Linjun Shou, Daxin Jiang, Xing Xie, and Yinliang Yue. 2022. Tiger: Transferable Interest Graph Embedding for Domain-Level Zero-Shot Recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 2806–2816.