Neural Cognitive Diagnosis for Intelligent Education Systems

Fei Wang,[1] **Qi Liu,**[1*] **Enhong Chen,**[1*] **Zhenya Huang,**[1]
Yuying Chen,[1] **Yu Yin,**[1] **Zai Huang,**[1] **Shijin Wang**[2]
[1]Anhui Province Key Lab. of Big Data Analysis and Application,
School of Computer Science and Technology, University of Science and Technology of China
[2]iFLYTEK Research
{wf314159, huangzhy, cyy33222, yxonic, huangzai}@mail.ustc.edu.cn, {qiliuql, cheneh}@ustc.edu.cn,
sjwang3@iflytek.com

Abstract

Cognitive diagnosis is a fundamental issue in intelligent education, which aims to discover the proficiency level of students on specific knowledge concepts. Existing approaches usually mine linear interactions of student exercising process by manual-designed function (e.g., logistic function), which is not sufficient for capturing complex relations between students and exercises. In this paper, we propose a general Neural Cognitive Diagnosis (NeuralCD) framework, which incorporates neural networks to learn the complex exercising interactions, for getting both accurate and interpretable diagnosis results. Specifically, we project students and exercises to factor vectors and leverage multi neural layers for modeling their interactions, where the monotonicity assumption is applied to ensure the interpretability of both factors. Furthermore, we propose two implementations of NeuralCD by specializing the required concepts of each exercise, i.e., the NeuralCDM with traditional Q-matrix and the improved NeuralCDM+ exploring the rich text content. Extensive experimental results on real-world datasets show the effectiveness of NeuralCD framework with both accuracy and interpretability.

1 Introduction

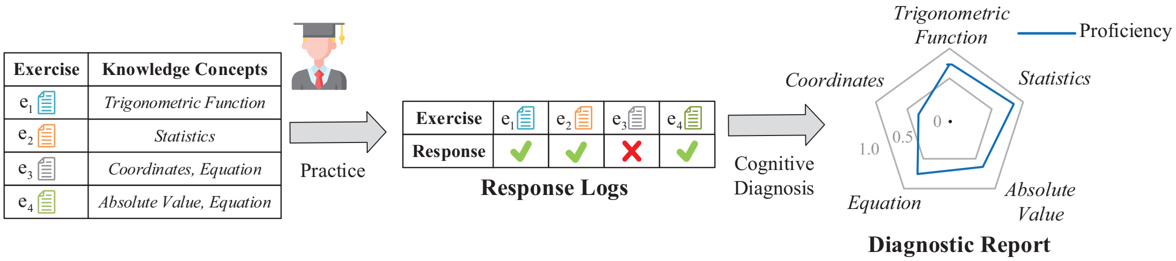
Cognitive diagnosis is a necessary and fundamental task in many real-world scenarios such as games (Chen and Joachims 2016), medical diagnosis (Guo et al. 2017), and education. Specifically, in intelligent education systems (Anderson et al. 2014; Burns et al. 2014), cognitive diagnosis aims to discover the states of students in the learning process, such as their proficiencies on specific knowledge concepts (Liu et al. 2018). Figure 1 shows a toy example of cognitive diagnosis. Generally, students usually first choose to practice a set of exercises (e.g., $e_1, \cdots, e_4$) and leave their responses (e.g., right or wrong). Then, our goal is to infer their actual knowledge states on the corresponding concepts (e.g., *Trigonometric Function*). In practice, these diagnostic reports are necessary as they are the basis of further services, such as exercise recommendation and targeted training (Kuh et al. 2011).

In the literature, massive efforts have been devoted for cognitive diagnosis, such as Deterministic Inputs, Noisy-And gate model (DINA) (De La Torre 2009), Item Response Theory (IRT) (Embretson and Reise 2013), Multidimensional IRT (MIRT) (Reckase 2009) and Matrix Factorization (MF) (Koren, Bell, and Volinsky 2009). Despite achieving some effectiveness, these works rely on handcrafted interaction functions that just combine the multiplication of student's and exercise's trait features linearly, such as logistic function (Embretson and Reise 2013) or inner product (Koren, Bell, and Volinsky 2009), which may not be sufficient for capturing the complex relationship between students and exercises (DiBello, Roussos, and Stout 2006). Besides, the design of specific interaction functions is also labor-intensive since it usually requires professional expertise. Therefore, it is urgent to find an automatic way to learn the complex interactions for cognitive diagnosis instead of manually designing them.

In this paper, we address this issue in a principled way of proposing a Neural Cognitive Diagnosis (NeuralCD) framework by incorporating neural networks to model complex non-linear interactions. Although the capability of neural networks to approximate continuous functions has been proved in many domains, such as natural language processing (Zhang et al. 2018) and recommender systems (Song et al. 2019), it is still highly nontrivial to adapt to cognitive diagnosis due to the following domain challenges. First, the black-box nature of neural networks makes them difficult to get explainable diagnosis results. That is to say, it is difficult to explicitly realize how much a student has mastered a certain knowledge concept (e.g., *Equation*). Second, traditional models are designed manually with non-neural functions, which makes it hard for them to leverage exercise text content. However, with neural network, it is worthy of finding ways to explore the rich information contained in exercise text content for cognitive diagnosis.

To address these challenges, we propose a NeuralCD framework to approximate interactions between students and exercises, yet preserving the explainability. We first project students and exercises to factor vectors and leverage multi-layers for modeling the complex interactions of student answering exercises. To ensure the interpretability

[*] Corresponding Authors.



Exercise	Knowledge Concepts
e_1	*Trigonometric Function*
e_2	*Statistics*
e_3	*Coordinates, Equation*
e_4	*Absolute Value, Equation*

Exercise	e_1	e_2	e_3	e_4
Response	✓	✓	✗	✓

Figure 1: A toy example of cognitive diagnosis. The student choose some exercises for practice and leave the response logs. With cognitive diagnosis methods, we get the diagnostic report containing the student's proficiency on each knowledge concept.

of both factors, we apply the monotonicity assumption taking from educational property (Reckase 2009) on the multi-layers. Then, we propose two implementations on the basis of the general framework, i.e., NeuralCDM and NeuralCDM+. In NeuralCDM, we simply extract exercise factor vectors from traditional Q-matrix (an example is shown in Figure 6) and achieve the monotonicity property with positive full connection layers, which shows feasibility of the framework. While in NeuralCDM+, we demonstrate how information from exercise text can be explored with neural network to extend the framework. Particularly, our NeuralCD is a general framework since it can cover many traditional models such as MF, IRT and MIRT. Finally, we conduct extensive experiments on real-world datasets, and the results show the effectiveness of NeuralCD framework with both accuracy and interpretability guarantee.

Our code of NeuralCDM is available at https://github.com/bigdata-ustc/NeuralCD.

2 Related Work

In this section, we briefly review the related works from the following three aspects.

Cognitive Diagnosis. Existing works about student cognitive diagnosis mainly came from educational psychology area. DINA (De La Torre 2009; von Davier 2014) and IRT (Embretson and Reise 2013) were two of the most typical works, which model the result of a student answering an exercise as the interaction between the trait features of the student ($\boldsymbol{\theta}$) and the exercise ($\boldsymbol{\beta}$). Specifically, in DINA, $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ were binary, where $\boldsymbol{\beta}$ came directly from Q-matrix (a human labeled exercise-knowledge correlation matrix, an example is showed in Figure 6). Another two exercise factors, i.e. guessing and slipping (parameterized as g and s) are also taken into consideration. The probability of student i correctly answering exercise j was modeled as $P(r_{ij}=1|\boldsymbol{\theta}_i) = g_j^{1-\eta_{ij}}(1-s_j)^{\eta_{ij}}$, where $\eta_{ij} = \prod_k \theta_{ik}^{\beta_{jk}}$. On the other hand, in IRT, $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ were unidimensional and continuous latent traits, indicating student ability and exercise difficulty respectively. The interaction between the trait features was modeled in a logistic way, e.g., a simple version is $sigmoid(a(\boldsymbol{\theta}-\boldsymbol{\beta}))$, where a is the exercise discrimination parameter. Although extra parameters were added in IRT (Fischer 1995; Lord 2012) and latent trait was extended to multidimensional(MIRT) (Adams, Wilson, and Wang 1997; Reckase 2009), most of their item response functions were still logistic-like. These traditional models depended on manually designed functions, which was labor-intensive and restricted their scope of applications.

Matrix Factorization. Recently, some researches from data mining perspective have demonstrated the feasibility of MF for cognitive diagnosis. Student and exercise correspond to user and item in matrix factorization (MF). For instance, Toscher et al. (2010) improved SVD (Singular Value Decomposition) methods to factor the score matrix and get students and exercises' latent trait vectors. Thai-Nghe et al. (2010) applied some recommender system techniques including matrix factorization in the educational context, and compared it with traditional regression methods. Besides, Thai-Nghe et al. (2015) proposed a multi-relational factorization approach for student modeling in the intelligent tutoring systems. Despite their effectiveness in predicting students' scores on exercises, the latent trait vectors in MF is not interpretable for cognitive diagnosis, i.e. there is no clear correspondence between elements in trait vectors and specific knowledge concepts.

Artificial Neural Network. Techniques using artificial neural network have reached state-of-the-art in many areas, e.g., speech recognition (Chan et al. 2016), text classification (Zhang, Zhao, and LeCun 2015) and image captioning (Wang, Chen, and Hu 2019). There are also some educational applications such as question difficulty prediction (Huang et al. 2017), code education (Wu et al. 2019), formula image transcribing (Yin et al. 2018) and student performance prediction (Huang et al. 2019). However, using neural network for cognitive diagnosis is nontrivial as it performs poorly in parameter interpretation due to its inherent traits. To the best of our knowledge, deep knowledge tracing (DKT) (Piech et al. 2015) was the first attempt to model student learning process using recurrent neural network. However, DKT aims to predict students' scores, and does not make a distinction between an exercise and the knowledge concepts it contains, thus it's unsuitable for cognitive diagnosis. Few works with neural network have high interpretability for student cognitive diagnosis. Towards this end, in this paper we propose a neural cognitive diagnosis (NeuralCD) framework which borrows concepts from educational psychology and combine them with interaction functions learned from data. NeuralCD could achieve both high accuracy and interpretation with neural network. Besides, the framework is general that can cover many tradition models, and at the same time easy for extension.

3 Neural Cognitive Diagnosis

We first formally introduce cognitive diagnosis task. Then we describe the details of NeuralCD framework. After that, we design a specific diagnostic network NeuralCDM with traditional Q-matrix to show the feasibility of the framework, and an improved NeuralCDM+ by incorporating exercise text content for better performance. Finally, we demonstrate the generality of NeuralCD framework by showing its close relationship with some traditional models.

3.1 Task Overview

Suppose there are N Students, M Exercises and K Knowledge concepts at a learning system, which can be represented as $S = \{s_1, s_2, \ldots, s_N\}, E = \{e_1, e_2, \ldots, e_M\}$ and $K_n = \{k_1, k_2, \ldots, k_K\}$ respectively. Each student will choose some exercises for practice, and the response logs R are denoted as set of triplet (s, e, r) where $s \in S, e \in E$ and r is the score (transferred to percentage) that student s got on exercise e. In addition, we have Q-matrix (usually labeled by experts) $\mathbf{Q} = \{Q_{ij}\}_{M \times K}$, where $Q_{ij} = 1$ if exercise e_i relates to knowledge concept k_j and $Q_{ij} = 0$ otherwise.

Problem Definition *Given students' response logs R and the Q-matrix $\mathbf{Q}$, the goal of our cognitive diagnosis task is to mine students' proficiency on knowledge concepts through the student performance prediction process.*

3.2 Neural Cognitive Diagnosis Framework

Generally, for a cognitive diagnostic system, there are three elements need to be considered: student factors, exercise factors and the interaction function among them (DiBello, Roussos, and Stout 2006). In this paper, we propose a general NeuralCD framework to address them by using multi-layer neural network modeling, which is shown in Figure 2. Specifically, for each response log, we use one-hot vectors of the corresponding student and exercise as input and obtain the diagnostic factors of the student and exercise. Then the interactive layers learn the interaction function among the factors and output the probability of correctly answering the exercise. After training, we get students' proficiency vectors as diagnostic results. Details are introduced as bellow.

Student Factors. Student factors characterize the traits of students, which would affect the students' response to exercises. As our goal is to mine students' proficiency on knowledge concepts, we do not use the latent trait vectors as in IRT and MIRT, which is not explainable enough to guide students' self-assessment. Instead, we design the student factors as explainable vectors similar to DINA, but has a major difference that they are continuous. Specifically, We use a vector F^s to characterize a student, namely *proficiency vector*. Each entry of F^s is continuous ([0,1]), which indicates the student's proficiency on a knowledge concept. For example, $F^s = [0.9, 0.2]$ indicates a high mastery on the first knowledge concept but low mastery on the second. F^s is got through the parameter estimation process.

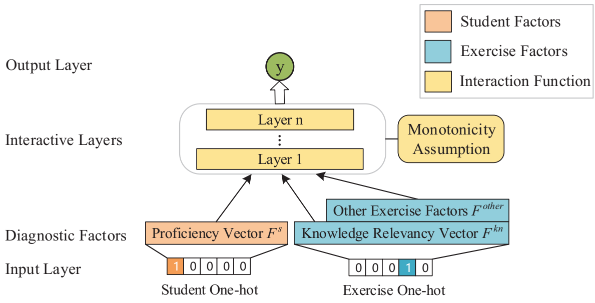


Figure 2: Structure of NeuralCD framework.

Exercise Factors. Exercise factors denote the factors that characterize the traits of exercises. We divide exercise factors into two categories. The first indicates the relationship between exercises and knowledge concepts, which is fundamental as we need it to make each entry of F^s correspond to a specific knowledge concept for our diagnosis goal. We call it *knowledge relevancy vector* and denote it as F^{kn}. F^{kn} has the same dimension as F^s, with the ith entry indicating the relevancy between the exercise and the knowledge concept k_i. Each entry of F^{kn} is non-negative. F^{kn} is previously given (e.g., obtained from Q-matrix). Other factors are of the second type and are optional. Factors from IRT and DINA such as knowledge difficulty, exercise difficulty and discrimination can be incorporated if reasonable.

Interaction Function. We use artificial neural network to obtain the interaction function for the following reasons. First, the neural network has been proven to be capable of approximating any continuous function (Hornik, Stinchcombe, and White 1989). The strong fitting ability of neural network makes it competent for capturing relationships among student and exercise factors. Second, with neural network, the interaction function can be learned from data with few assumptions (that behind traditional models). This makes NeuralCD more general and can be applied in broad areas. Third, the framework can be highly extendable with neural network. For instance, extra information such as exercise texts can be integrated in with neural network (We will discuss its extendability in the following subsections.). Mathematically, we formulate the output of NeuralCD framework as:

$$y = \varphi_n(\ldots \varphi_1(F^s, F^{kn}, F^{other}, \theta_f)), \qquad (1)$$

where φ_i denotes the mapping function of the ith MLP layer; F^{other} denotes factors other than F^s and F^{kn} (e.g., difficulty); and θ_f denotes model parameters of all the interactive layers.

However, due to some intrinsic characteristics, neural networks usually have poor performance on interpretation (Samek et al. 2016). Fortunately, we find that the monotonicity assumption, which is used in some IRT and MIRT models (Reckase 2009), can be utilized to ensure the interpretation of student and exercise factors. Monotonicity assumption is general and reasonable in almost all circumstance, thus it has little influence on the generality of NeuralCD framework. The assumption is defined as follows:

Monotonicity Assumption *The probability of correct response to the exercise is monotonically increasing at any dimension of the student's knowledge proficiency.*

This assumption should be converted as a property of the interaction function. Intuitively, we assume student s to answer exercise e correctly. During training, the optimization algorithm should increase the student's proficiency if the model output a wrong prediction (i.e., a value below 0.5). The increment of each knowledge proficiency is otherwise controlled by F^{kn}.

After introducing the structure of NeuralCD framework, we will next show some specific implementations. We first design a diagnostic model based on NeuralCD with extra exercise factors (i.e., knowledge difficulty and exercise discrimination)(§3.3), and further show its extendability by incorporating text information (§3.4) and generality by demonstrating how it covers traditional models (§3.5).

3.3 Neural Cognitive Diagnosis Model

Here we introduce a specific neural cognitive diagnosis model (NeuralCDM) under NeuralCD framework. Figure 3 illustrates the structure of NeuralCDM.

Student Factors. In NeuralCDM, each student is represented with a knowledge proficiency vector. The student factor F^s aforementioned is $\boldsymbol{h}^s$ here, and $\boldsymbol{h}^s$ is obtained by multiplying the student's one-hot representation vector $\boldsymbol{x}^s$ with a trainable matrix $\mathbf{A}$. That is,

$$\boldsymbol{h}^s = \mathrm{sigmoid}(\boldsymbol{x}^s \times \mathbf{A}), \tag{2}$$

in which $\boldsymbol{h}^s \in (0,1)^{1\times K}, \boldsymbol{x}^s \in \{0,1\}^{1\times N}, \mathbf{A} \in \mathbb{R}^{N\times K}$.

Exercise Factors. As for each exercise, the aforementioned exercise factor F^{kn} is $\boldsymbol{Q}_e$ here, which directly comes from the pre-given Q-matrix:

$$\boldsymbol{Q}_e = \boldsymbol{x}^e \times \mathbf{Q}, \tag{3}$$

where $\boldsymbol{Q}_e \in \{0,1\}^{1\times K}$, $\boldsymbol{x}^e \in \{0,1\}^{1\times M}$ is the one-hot representation of the exercise. In order to make a more precise diagnosis, we adopt other two exercise factors: knowledge difficulty $\boldsymbol{h}^{diff}$ and exercise discrimination h^{disc}. $\boldsymbol{h}^{diff} \in (0,1)^{1\times K}$, indicates the difficulty of each knowledge concept examined by the exercise, which is extended from exercise difficulty used in IRT. $h^{disc} \in (0,1)$, used in some IRT and MIRT models, indicates the capability of the exercise to differentiate between those students whose knowledge mastery is high from those with low knowledge mastery. They can be obtained by:

$$\boldsymbol{h}^{diff} = \mathrm{sigmoid}(\boldsymbol{x}^e \times \mathbf{B}), \mathbf{B} \in \mathbb{R}^{M\times K} \tag{4}$$

$$h^{disc} = \mathrm{sigmoid}(\boldsymbol{x}^e \times \mathbf{D}), \mathbf{D} \in \mathbb{R}^{M\times 1} \tag{5}$$

where $\mathbf{B}$ and $\mathbf{D}$ are trainable matrices.

Interaction Function. The first layer of the interaction layers is inspired by MIRT models. We formulate it as:

$$\boldsymbol{x} = \boldsymbol{Q}_e \circ (\boldsymbol{h}^s - \boldsymbol{h}^{diff}) \times h^{disc}, \tag{6}$$

where $\circ$ is element-wise product. Following are two full connection layers and an output layer:

$$\boldsymbol{f}_1 = \phi(\mathbf{W}_1 \times \boldsymbol{x}^T + \boldsymbol{b}_1), \tag{7}$$

$$\boldsymbol{f}_2 = \phi(\mathbf{W}_2 \times \boldsymbol{f}_1 + \boldsymbol{b}_2), \tag{8}$$

$$y = \phi(\mathbf{W}_3 \times \boldsymbol{f}_2 + b_3), \tag{9}$$

where ϕ is the activation function. Here we use Sigmoid.

Different methods can be used to satisfy the monotonicity assumption. We adopt a simple strategy: restrict each element of $\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3$ to be positive. It can be easily proved that $\frac{\partial y}{\partial h_i^s}$ is positive for each entry h_i^s in $\boldsymbol{h}^s$. Thus monotonicity assumption is always satisfied during training.

The loss function of NeuralCDM is cross entropy between output y and true label r:

$$loss_{CDM} = -\sum_i (r_i \log y_i + (1 - r_i)\log(1 - y_i)). \tag{10}$$

After training, the value of $\boldsymbol{h}^s$ is what we get as diagnosis result, which denotes the student's knowledge proficiency.

3.4 NeuralCD Extension with Text Information

In NeuralCDM and some traditional methods (e.g., DINA), Q-matrix is the source of information about the exercise knowledge concept. However, manually-labeled Q-matrix may be deficient because of inevitable errors and subjective bias (Liu, Xu, and Ying 2012; DiBello, Roussos, and Stout 2006). On the other hand, exercise texts have been proved to be highly related to some exercise features (e.g., difficulty, relevant knowledge concepts) (Su et al. 2018; Huang et al. 2017), thus it can be leveraged to refine the Q-matrix. For example, in Q-matrix, maybe only '*Equation*' is labeled for an equation solving exercise. However, we may discover that '*Division*' is also required due to the existence of '$\div$' in the text. Traditional cognitive models didn't leverage text content due to the limitation of their handcraft non-neural interaction functions. However, with neural network, we are able to incorporate text information into our framework. We denote the extended model as NeuralCDM+, and present its structure in Figure 4.

Specifically, we first pre-train a CNN (convolutional neural network) to predict knowledge concepts related to the input exercise. CNN has advantage of extracting local information in text processing, thus it's able to capture important words from texts (e.g., words that are highly relative to certain knowledge concepts). The network takes concatenated word2vec embedding of words in texts as input, and output the relevancy of each predefined knowledge concept (that has occurred in data) to the exercise. Human-labeled Q-matrix is used as label for training. We define $V_i^k = \{V_{ij_1}, V_{ij_2}, \ldots, V_{ij_k}\}$ as the set of top-k knowledge concepts of exercise e_i outputted by the CNN.

Then we combine V_i^k with Q-matrix. Although there are defects in human-labeled Q-matrix, it still has high confidence. Thus we consider knowledge concepts labeled by Q-matrix are more relative than concepts in $\{k_j | k_j \in V_i^k \ and \ Q_{ij} = 0\}$. To achieve this, we adopt a pairwise

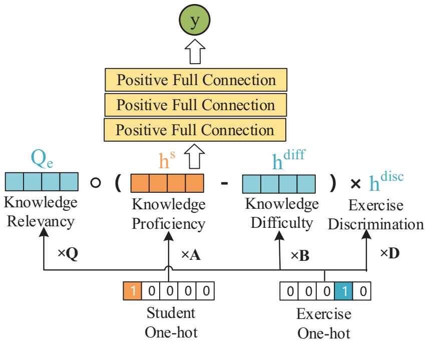


Figure 3: Neural cognitive diagnosis model. The color of orange, blue and yellow indicate student factors, exercise factors and interacion function respectively.

Bayesian method as follows. For convenience, we define partial order $>_i^+$ as:

$$a >_i^+ b, \text{ if } Q_{ia} = 1 \text{ and } Q_{ib} = 0 \text{ and } b \in V_i^k, \tag{11}$$

and define the partial order relationship set as $D_V = \{(i,a,b)|a >_i^+ \ b, i = 1, 2, \dots, M\}$. Following traditional Bayesian treatment, we assume $\tilde{\mathbf{Q}}$ follows a zero mean Gaussian prior with standard deviation σ of each dimension. To give Q-matrix labels higher confidence, we define $p(a >_i^+ b|\tilde{\boldsymbol{Q}}_i)$ with a pairwise logistic-like function:

$$p(a >_i^+ b|\tilde{\boldsymbol{Q}}_i) = \frac{1}{1 + e^{-\lambda(\tilde{Q}_{ia} - \tilde{Q}_{ib})}}. \tag{12}$$

The parameter λ controls the discrimination of relevance values between labeled and unlabeled knowledge concepts. The log posterior distribution over D_V on $\tilde{\mathbf{Q}}$ is finally formulated as:

$$\begin{aligned}\ln p(\tilde{\mathbf{Q}}|D_V) &= \ln \prod_{(i,a,b)\in D_V} p(a >_i^+ b|\tilde{\boldsymbol{Q}}_i)p(\tilde{\boldsymbol{Q}}_i) \\ &= \sum_{i=1}^{M}\sum_{a=1}^{K}\sum_{b=1}^{K} I(a >_i^+ b) \ln \frac{1}{1 + e^{-\lambda(\tilde{Q}_{ia} - \tilde{Q}_{ib})}} \\ &+ C - \sum_{i=1}^{M}\sum_{j=1}^{K} \frac{\tilde{Q}_{ij}^2}{2\sigma^2},\end{aligned} \tag{13}$$

where C is a constant that can be ignored during optimization. Before using $\tilde{\mathbf{Q}}$ in NeuralCDM, we need to restrict its elements to the range $(0, 1)$, and set elements of concepts unlabeled or not predicted to 0. Thus, $Sigmoid(\tilde{\mathbf{Q}}) \circ \mathbf{M}$ is used to replace $\mathbf{Q}$ in NeuralCDM, where $\mathbf{M} \in \{0, 1\}^{M \times K}$ is a mask matrix, and $M_{ij} = 1$ if $j \in V_i^k$ or $Q_{ij} = 1$; $M_{ij} = 0$ otherwise. $\tilde{\mathbf{Q}}$ is trained together with the cognitive diagnostic model, thus the loss function is:

$$loss = -\ln p(\tilde{\mathbf{Q}}|D_V) + loss_{CDM}. \tag{14}$$

3.5 Generality of NeuralCD

In this subsection we show that NeuralCD is a general framework which can cover many traditional cognitive diagnostic models. Using Eq. (6) as the first layer, we now show the close relationship between NeuralCD and traditional models, including MF, IRT and MIRT.

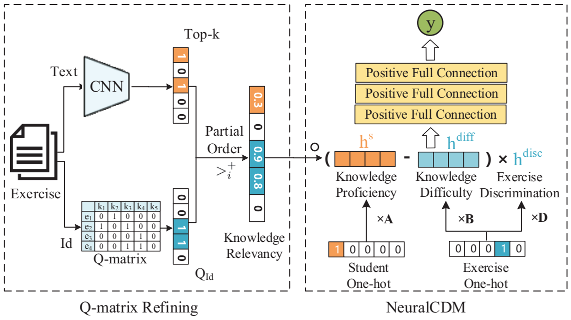
MF. $\boldsymbol{Q}_e$ and $\boldsymbol{h}^s$ can be seen as exercise and student latent trait vectors respectively in MF. By setting $\boldsymbol{h}^{diff} \equiv \mathbf{0}$ and $h^{disc} \equiv 1$, the output of the first layer is $\boldsymbol{x} = \boldsymbol{Q}_e \circ \boldsymbol{h}^s$. Then in order to work like MF (i.e., $y = \boldsymbol{Q}_e \cdot \boldsymbol{h}^s$), all the rest of layers need to do is to sum up the values of each entry in $\boldsymbol{x}$, which is easy to achieve. Monotonicity assumption is not applied in MF approaches.

IRT. Take the typical formation of IRT $y = \text{Sigmoid}((h^s - h^{diff}) \times h^{disc})$ as example. Set $Q_e \equiv 1$, and let $\boldsymbol{h}^s$ and $\boldsymbol{h}^{diff}$ be unidimensional, the output of the first layer is $x = (h^s - h^{diff}) \times h^{disc}$, followed by a Sigmoid activation function. Monotonicity assumption is achieved by limiting h^{disc} to be positive. Other variations of IRT (e.g., $y' = C + (1 - C)y$ where C is guessing parameter) can be realized with a few changes.

MIRT. One direct extension from IRT to MIRT is to use multidimensional latent trait vectors of exercises and student. Here we take the typical formation proposed in (Adams, Wilson, and Wang 1997) as example:

$$y = \frac{e^{\mathbf{Q}_e \cdot \boldsymbol{h}^s - d_e}}{1 + e^{\mathbf{Q}_e \cdot \boldsymbol{h}^s - d_e}}. \tag{15}$$

Let $h^{disc} \equiv 1$, the output of the first layer given by Eq. (6) is $\boldsymbol{x} = \mathbf{Q}_e \circ (\boldsymbol{h}^s - \boldsymbol{h}^{diff})$. By Setting $\mathbf{W}_1 = [1 \ \ 1 \ \ \cdots \ \ 1]$, $\boldsymbol{b}_1 = \mathbf{0}$ and $\phi(x) = x$ in Eq. (7), we have $f_1 = \mathbf{Q}_e \cdot \boldsymbol{h}^s - d_e$ (where $d_e = \boldsymbol{Q}_e \cdot \boldsymbol{h}^{diff}$). All the rest of the layers need to do is to approximate the function $g(f_1) = 1 - \text{Sigmoid}(f_1)$, which can be easily achieved with two more layers. Monotonicity assumption can be realized if each entry of $\boldsymbol{Q}_e$ is restricted to be positive.



	k_1	k_2	k_3	k_4	k_5
e_1	0	1	0	0	0
e_2	1	0	1	0	0
e_3	0	0	0	1	0
e_4	0	0	1	1	0

Figure 4: Extended neural cognitive diagnosis model. The knowledge relevancy vectors are replaced with vectors in Q-matrix that is refined by leveraging exercise texts.

3.6 Discussion

We have introduced the details of NeuralCD framework and showed special cases of it. It's necessary to point out that the student's proficiency vector F^s and exercise's knowledge relevancy vector F^{kn} are basic factors needed in NeuralCD framework. Additional factors such as exercise discrimination can be integrated into if reasonable. The formation of the first interactive layer is not limited, but it's better to contain the term $F^s \circ F^{kn}$ to ensure that each dimension of

F^s corresponds to a specific knowledge concept. The positive full connection is only one of the strategies that implement monotonicity assumption. More sophisticated network structures can be designed as the interaction layers. For example, recurrent neural network may be used to capture the time characteristics of the student's learning process.

4 Experiments

We first compare our NeuralCD models with some baselines on the student performance prediction task. Then we make some interpretation assessments of the models.

4.1 Dataset Description

We use two real-world datasets in the experiments, i.e., Math and ASSIST. Math dataset supplied by iFLYTEK Co., Ltd. is collected from the widely-used online learning system Zhixue[1], which contains mathematical exercises and logs of high school examinations. ASSIST (ASSISTments 2009-2010 "skill builder") is an open dataset collected by the ASSISTments online tutoring systems (Feng, Heffernan, and Koedinger 2009), which only provides student response logs and knowledge concepts[2]. We choose the public corrected version that eliminates the duplicated data issue proposed by previous work (Xiong et al. 2016). Table 1 summarizes basic statistics of the datasets.

We filter out students with less than 30 and 15 response logs for Math and ASSIST respectively to guarantee that each student has enough data for diagnosis. Therefore for dataset Math, we got 2,507 exercises with 497 knowledge concepts for diagnostic network, and the remaining exercises with knowledge concepts not appearing in logs are used for the Q-matrix refining part of NeuralCDM+. We perform a 80%/20% train/test split of each student's response log. As for ASSIST, we divide the response logs in the same way with Math, but NeuralCDM+ is not evaluated on this dataset as exercise text is not provided. All models are evaluated with 5-fold cross validation.

Students' knowledge proficiencies are stable in Math as the dataset is composed of logs from examinations. However, a student's proficiency on a knowledge concept may change as he will be continually given exercises of that concept until meeting certain criterion (e.g., answering 3 relevant exercises correctly in a row). To analyze whether static models (e.g., NeuralCD models and static traditional models) are suitable to apply on ASSIST, we compare two metrics between Math and ASSIST. The first metric is the average amount of logs that each student toke for each knowledge concept:

$$\text{AVG}_{\#log} = \frac{\sum_i^N \sum_j^K Log(i,j)}{\sum_i^N \sum_j^K I(Log(i,j) > 0)}, \tag{16}$$

where $Log(i,j)$ is the amount of exercises student s_i answered that related to knowledge concept k_j. Further, another metric is the mean standard deviation of scores r_{ij} that $Log(i,j) > 1$ as:

Table 1: Dataset summary.

Dataset	Math	ASSIST
#Students	10,268	4,163
#Exercises	917,495	17,746
#Knowledge concepts	1,488	123
#Response logs	864,722	324,572
#Knowledge concepts per exercise	1.53	1.19
$\text{AVG}_{\#log}$	2.28	8.05
$\text{STD}_{\#log>1}$	0.305	0.316

$$\text{STD}_{\#log>1} = \underset{s_i \in S}{\text{mean}}(\underset{\substack{k_j \in K_n,\\ Log(i,j)>1}}{\text{mean}} (std_{ij})), \tag{17}$$

where std_{ij} is the standard deviation of scores that student s_i got for exercises related to knowledge concept k_j. As the results showed in Table 1, although ASSIST has a much larger $\text{AVG}_{\#log}$ than Math, their $\text{STD}_{\#log>1}$ are close. Therefore, it is reasonable to assume that the knowledge states of students in ASSIST are also stable, and our static NeuralCD models and baselines are applicable for both dataset. There will be more discussions in Model Interpretation.

4.2 Experimental Setup

The dimensions of the full connection layers (Eq. (7) $\sim$ (9)) are 512, 256, 1 respectively, and Sigmoid is used as activation function for all of the layers. We set hyperparameters $\lambda = 0.1$ (Eq. (12)) and $\sigma = 1$ (Eq. (13)). For k in top-k knowledge concepts selecting, we use the value that make the predicting network reach 0.85 recall. That is, in our experiment, $k = 20$. We initialize the parameters with *Xavier* initialization (Glorot and Bengio 2010), which fill the weights with random values sampled from $\mathcal{N}(0, std^2)$, where $std = \sqrt{\frac{2}{n_{in}+n_{out}}}$. n_{in} is the number of neurons feeding into the weights, and n_{out} is the number of neurons the results is fed to.

The CNN architecture we use in NeuralCDM+ contains 3 convolutional layers followed by a full connection output layer. MaxPooling are used after 1st and 3rd convolutional layers. The channels of convolutional layers are 400, 200, 100, and kernel sizes are set to 3, 4, 5 respectively. We adopt ReLu activation function for convolution layers and Sigmoid for the output layer. Multi-label binary cross entropy is used as loss function for training the CNN.

To evaluate the performance of our NeuralCD models, we compare them with previous approaches, i.e., DINA, IRT, MIRT and PMF. All models are implemented by PyTorch using Python, and all experiments are run on a Linux server with four 2.0GHz Intel Xeon E5-2620 CPUs and a Tesla K20m GPU.

4.3 Experimental Results

Student Performance Prediction The performance of a cognitive diagnosis model is difficult to evaluate as we can't

[1] https://www.zhixue.com

[2] https://sites.google.com/site/assistmentsdata/home/assistment-2009-2010-data/skill-builder-data-2009-2010

Table 2: Experimental results on student performance prediction.

	Math			ASSIST		
Model	Accuracy	RMSE	AUC	Accuracy	RMSE	AUC
DINA	0.593±.001	0.487±.001	0.686±.001	0.650±.001	0.467±.001	0.676±.002
IRT	0.782±.002	0.387±.001	0.795±.001	0.674±.002	0.464±.002	0.685±.001
MIRT	0.793±.001	0.378±.002	0.813±.002	0.701±.002	0.461±.001	0.719±.001
PMF	0.763±.001	0.407±.001	0.792±.002	0.661±.002	0.476±.001	0.732±.001
NeuralCDM	0.792±.002	0.378±.001	0.820±.001	**0.719±.008**	**0.439±.002**	**0.749±.001**
NeuralCDM+	**0.804±.001**	**0.371±.002**	**0.835±.002**	-	-	-

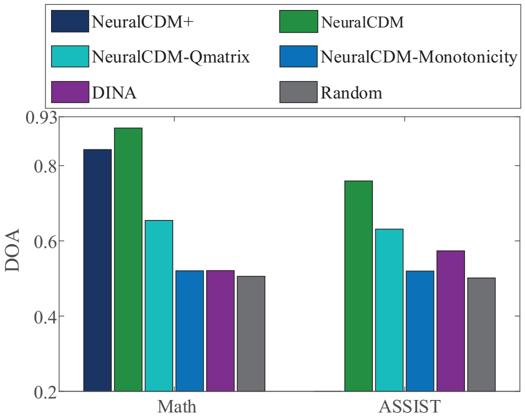
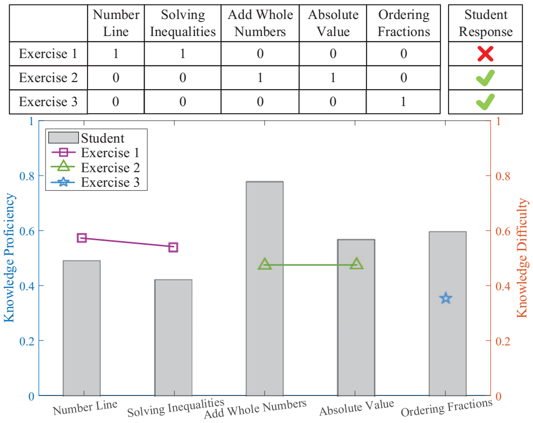


Figure 5: DOA results of models. In NeuralCD models, there is clear correspondence between entries in $\boldsymbol{h}^s$ and knowledge concepts, thus their diagnosis results have high DOA. Removing Q-matrix or monotonicity assumption would reduce the performance.



	Number Line	Solving Inequalities	Add Whole Numbers	Absolute Value	Ordering Fractions	Student Response
Exercise 1	1	1	0	0	0	✗
Exercise 2	0	0	1	1	0	✓
Exercise 3	0	0	0	0	1	✓

Figure 6: Diagnosis example of a student in ASSIST. The upper part is the Q-matrix of 3 exercises and corresponding response logs. The lower part shows the diagnosed student's knowledge proficiencies (bars) and knowledge difficulties of each exercise (points).

obtain the true knowledge proficiency of students. As diagnostic result is usually acquired through predicting students' performance in most works, performance on these prediction tasks can indirectly evaluate the model from one aspect (Liu et al. 2018). Considering that all the exercises we used in our data are objective exercises, we use evaluation metrics from both classification aspect and regression aspect, including accuracy, RMSE (root mean square error) (Pei et al. 2018) and AUC (area under the curve) (Bradley 1997).

Table 2 shows the experimental results of all models on student performance prediction task. The error bars after '±' is the standard deviations of 5 evaluation runs for each model. From the table, we can observe that NeuralCD models outperform almost all the other baselines on both datasets, indicating the effectiveness of our framework. In addition, the better performance of NeuralCDM+ over NeuralCDM proves that the Q-matrix refining method is effective, and also demonstrates the importance of fine estimated knowledge relevancy vectors for cognitive diagnosis.

Model Interpretation To assess the interpretability of NeuralCD framework (i.e., whether the diagnostic result is reasonable), we further conduct several experiments.

Intuitively, if student a has a better mastery on knowledge concept k than student b, then a is more likely to answer exercises related to k correctly than b (Chen et al. 2017). We adopt Degree of Agreement (DOA) (Pirotte et al. 2007) as the evaluation metric of this kind of ranking performance. For knowledge concept k, $DOA(k)$ is formulated as:

$$DOA(k) = \frac{1}{Z}\sum_{a=1}^{N}\sum_{b=1}^{N}\delta(F^s_{ak}, F^s_{bk})\sum_{j=1}^{M} I_{jk}\frac{J(j,a,b)\wedge\delta(r_{aj}, r_{bj})}{J(j,a,b)}, \quad (18)$$

where $Z = \sum_{a=1}^{N}\sum_{b=1}^{N}\delta(F^s_{ak}, F^s_{bk})$. F^s_{ak} is the proficiency of student a on knowledge concept k. $\delta(x,y) = 1$ if $x > y$ and $\delta(x,y) = 0$ otherwise. $I_{jk} = 1$ if exercise j contains knowledge concept k and $I_{jk} = 0$ otherwise. $J(j,a,b) = 1$ if both student a and b did exercise j and $J(j,a,b) = 0$ otherwise. We average $DOA(k)$ on all knowledge concepts to evaluate the quality of diagnostic result (i.e., knowledge proficiency acquired by models).

Among traditional models, we only compare with DINA, since for IRT, MIRT and PMF, there are no clear correspondence between their latent features and knowledge concepts. Besides, we conduct experiments on two reduced NeuralCDM models. In the first reduced model (denoted as NeuralCDM-Qmatrix), knowledge relevancy vectors are estimated during unsupervised training instead of getting from Q-matrix. While in another reduced model (denoted as NeuralCDM-Monotonocity), monotonicity assumption is removed by eliminating the positive restriction on the full connection layers. These two reduced models are used to demonstrate the importance of fine-estimated knowledge relevancy vector and monotonicity assumption respectively. Furthermore, we conduct an extra experiment in which students' knowledge proficiencies are randomly estimated, and compute the DOA for comparison.

Figure 5 presents the experimental results. From the figure we can observe that DOAs of NeuralCDM and NeuralCDM+ are significantly higher than baselines, which proves that knowledge proficiencies diagnosed by them are reasonable. The DOAs of NeuralCDM-Qmatrix and NeuralCDM-Monotonicity are much lower than NeuralCDM, which indicates that both information from Q-matrix and monotonicity assumption are important for getting interpretable diagnosis results (knowledge proficiency vectors). DOA of DINA is slightly higher than Random due to the use of Q-matrix. Besides, NeuralCDM performs much better on Math than on ASSIST. This is mainly due to the contradictions in logs, i.e., a student may answer some exercises containing knowledge concept k_j correctly while others containing k_j wrong (reasons may be the change of knowledge proficiency, or other knowledge concepts contained by the exercises). As showed in Table 1, ASSIST has much larger $\mathrm{AVG}_{\#log}$ and slightly higher $\mathrm{STD}_{\#log>1}$ than Math dataset, which makes more contradictions in logs. Longer logs with more contradictions would decrease DOA.

Case Study. Here we present an example of a student's diagnostic result of NeuralCDM on dataset ASSIST in Figure 6. The upper part of Figure 6 shows the Q-matrix of three exercises on five knowledge concepts and the response of a student to the exercises. The bars in the underneath subfigure represent the student's proficiency on each knowledge concept. The lines with different colors and markers represent the knowledge difficulties of the three exercises (for clarity, we only present difficulties of relevant knowledge concepts for each exercise). We can observe from the figure that the student is more likely to response correctly when his proficiency satisfies the requirement of the exercise. For example, exercise 3 requires the mastery of '*Ordering Fraction*' and corresponding difficulty is 0.35. The student's proficiency on '*Ordering Fraction*' is 0.60, which is higher than required, thus he answered it correctly. Both knowledge difficulty ($\boldsymbol{h}^{diff}$) and knowledge proficiency ($\boldsymbol{h}^{s}$) in NeuralCDM are explainable as expected.

4.4 Discussion.

From the above experiments, we can observe that NeuralCD models provide both accurate and interpretable results for cognitive diagnosis.

There still some directions for future studies. First, we may make our effort to design a more efficient model for knowledge concept prediction, which would promote the performance of NeuralCDM+. Second, the positive restriction on neural network weights may limit the approximate ability, thus we would like to explore more flexible methods to satisfy the monotonicity assumption. Third, since students' knowledge statuses change in many online self-learning circumstances, we would like to extend NeuralCD for dynamic cognitive diagnosis.

5 Conclusion

In this paper, we proposed a neural cognitive diagnostic framework, NeuralCD framework, for students' cognitive diagnosis. Specifically, we first discussed fundamental student and exercise factors in the framework, and placed a monotonicity assumption on the framework to ensure its interpretability. Then, we implemented a specific model NeuralCDM under the framework to show its feasibility, and further extended NeuralCDM by incorporating exercise text to refine Q-matrix. Extended experimental results on real-world datasets showed the effectiveness of our models with both accuracy and interpretability. We also showed that NeuralCD could be seen as the generalization of some traditional cognitive diagnostic models (e.g., MIRT). The structure of the diagnostic network in our work is designed intuitively. However, with the high flexibility and potential of neural network, we hope this work could lead to further studies.

6 Acknowledgments

This research was partially supported by grants from the National Natural Science Foundation of China (Grants No. 61922073, 61672483, U1605251, 61727809), the Science Foundation of Ministry of Education of China & China Mobile (No. MCM20170507), and the Iflytek joint research program.

References

Adams, R. J.; Wilson, M.; and Wang, W.-c. 1997. The multidimensional random coefficients multinomial logit model. *Applied psychological measurement* 21(1):1–23.

Anderson, A.; Huttenlocher, D.; Kleinberg, J.; and Leskovec, J. 2014. Engaging with massive online courses. In *Proceedings of the 23rd international conference on World wide web*, 687–698. ACM.

Bradley, A. P. 1997. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition* 30(7):1145–1159.

Burns, H.; Luckhardt, C. A.; Parlett, J. W.; and Redfield, C. L. 2014. *Intelligent tutoring systems: Evolutions in design*. Psychology Press.

Chan, W.; Jaitly, N.; Le, Q.; and Vinyals, O. 2016. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4960–4964. IEEE.

Chen, S., and Joachims, T. 2016. Predicting matchups and preferences in context. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 775–784. ACM.

Chen, Y.; Liu, Q.; Huang, Z.; Wu, L.; Chen, E.; Wu, R.; Su, Y.; and Hu, G. 2017. Tracking knowledge proficiency of students with educational priors. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 989–998. ACM.

De La Torre, J. 2009. Dina model and parameter estimation: A didactic. *Journal of educational and behavioral statistics* 34(1):115–130.

DiBello, L. V.; Roussos, L. A.; and Stout, W. 2006. 31a review of cognitively diagnostic assessment and a summary of psychometric models. *Handbook of statistics* 26:979–1030.

Embretson, S. E., and Reise, S. P. 2013. *Item response theory*. Psychology Press.

Feng, M.; Heffernan, N.; and Koedinger, K. 2009. Addressing the assessment challenge with an online system that tutors as it assesses. *User Modeling and User-Adapted Interaction* 19(3):243–266.

Fischer, G. H. 1995. Derivations of the rasch model. In *Rasch models*. Springer. 15–38.

Glorot, X., and Bengio, Y. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 249–256.

Guo, X.; Li, R.; Yu, Q.; and Haake, A. R. 2017. Modeling physicians' utterances to explore diagnostic decision-making. In *IJCAI*, 3700–3706.

Hornik, K.; Stinchcombe, M.; and White, H. 1989. Multilayer feedforward networks are universal approximators. *Neural networks* 2(5):359–366.

Huang, Z.; Liu, Q.; Chen, E.; Zhao, H.; Gao, M.; Wei, S.; Su, Y.; and Hu, G. 2017. Question difficulty prediction for reading problems in standard tests. In *AAAI*, 1352–1359.

Huang, Z.; Yin, Y.; Chen, E.; Xiong, H.; Su, Y.; Hu, G.; et al. 2019. Ekt: Exercise-aware knowledge tracing for student performance prediction. *IEEE Transactions on Knowledge and Data Engineering*.

Koren, Y.; Bell, R.; and Volinsky, C. 2009. Matrix factorization techniques for recommender systems. *Computer* 42(8):30–37.

Kuh, G. D.; Kinzie, J.; Buckley, J. A.; Bridges, B. K.; and Hayek, J. C. 2011. *Piecing together the student success puzzle: research, propositions, and recommendations: ASHE Higher Education Report*, volume 116. John Wiley & Sons.

Liu, Q.; Wu, R.; Chen, E.; Xu, G.; Su, Y.; Chen, Z.; and Hu, G. 2018. Fuzzy cognitive diagnosis for modelling examinee performance. *ACM Transactions on Intelligent Systems and Technology (TIST)* 9(4):48.

Liu, J.; Xu, G.; and Ying, Z. 2012. Data-driven learning of q-matrix. *Applied psychological measurement* 36(7):548–564.

Lord, F. M. 2012. *Applications of item response theory to practical testing problems*. Routledge.

Pei, H.; Yang, B.; Liu, J.; and Dong, L. 2018. Group sparse bayesian learning for active surveillance on epidemic dynamics. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Piech, C.; Bassen, J.; Huang, J.; Ganguli, S.; Sahami, M.; Guibas, L. J.; and Sohl-Dickstein, J. 2015. Deep knowledge tracing. In *Advances in Neural Information Processing Systems*, 505–513.

Pirotte, A.; Renders, J.-M.; Saerens, M.; et al. 2007. Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. *IEEE Transactions on Knowledge & Data Engineering* 19(3):355–369.

Reckase, M. D. 2009. Multidimensional item response theory models. In *Multidimensional Item Response Theory*. Springer. 79–112.

Samek, W.; Binder, A.; Montavon, G.; Lapuschkin, S.; and Müller, K.-R. 2016. Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems* 28(11):2660–2673.

Song, K.; Ji, M.; Park, S.; and Moon, I.-C. 2019. Hierarchical context enabled recurrent neural network for recommendation. *arXiv preprint arXiv:1904.12674*.

Su, Y.; Liu, Q.; Liu, Q.; Huang, Z.; Yin, Y.; Chen, E.; Ding, C.; Wei, S.; and Hu, G. 2018. Exercise-enhanced sequential modeling for student performance prediction. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Thai-Nghe, N., and Schmidt-Thieme, L. 2015. Multi-relational factorization models for student modeling in intelligent tutoring systems. In *Knowledge and Systems Engineering (KSE), 2015 Seventh International Conference on*, 61–66. IEEE.

Thai-Nghe, N.; Drumond, L.; Krohn-Grimberghe, A.; and Schmidt-Thieme, L. 2010. Recommender system for predicting student performance. *Procedia Computer Science* 1(2):2811–2819.

Toscher, A., and Jahrer, M. 2010. Collaborative filtering applied to educational data mining. *KDD cup*.

von Davier, M. 2014. The dina model as a constrained general diagnostic model: Two variants of a model equivalency. *British Journal of Mathematical and Statistical Psychology* 67(1):49–71.

Wang, W.; Chen, Z.; and Hu, H. 2019. Hierarchical attention network for image captioning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 8957–8964.

Wu, M.; Mosse, M.; Goodman, N.; and Piech, C. 2019. Zero shot learning for code education: Rubric sampling with deep learning inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 782–790.

Xiong, X.; Zhao, S.; Van Inwegen, E. G.; and Beck, J. E. 2016. Going deeper with deep knowledge tracing. *International Educational Data Mining Society*.

Yin, Y.; Huang, Z.; Chen, E.; Liu, Q.; Zhang, F.; Xie, X.; and Hu, G. 2018. Transcribing content from structural images with spotlight mechanism. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2643–2652. ACM.

Zhang, M.; Wang, W.; Liu, X.; Gao, J.; and He, Y. 2018. Navigating with graph representations for fast and scalable decoding of neural language models. In *Advances in Neural Information Processing Systems*, 6308–6319.

Zhang, X.; Zhao, J.; and LeCun, Y. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, 649–657.