Fully Adaptive Framework: Neural Computerized Adaptive Testing for Online Education

Yan Zhuang[1], Qi Liu[1*], Zhenya Huang[1], Zhi Li[1], Shuanghong Shen[1], Haiping Ma[2]

[1]Anhui Province Key Laboratory of Big Data Analysis and Application, School of Data Science & School of Computer Science and Technology, University of Science and Technology of China
[2]Anhui University
{zykb, zhili03, closer}@mail.ustc.edu.cn, {qiliuql, huangzhy}@ustc.edu.cn, hpma@ahu.edu.cn

Abstract

Computerized Adaptive Testing (CAT) refers to an efficient and personalized test mode in online education, aiming to accurately measure student proficiency level on the required subject/domain. The key component of CAT is the "adaptive" question selection algorithm, which automatically selects the best suited question for student based on his/her current estimated proficiency, reducing test length. Existing algorithms rely on some manually designed and pre-fixed informativeness/uncertainty metrics of question for selections, which is labor-intensive and not sufficient for capturing complex relations between students and questions. In this paper, we propose a fully adaptive framework named Neural Computerized Adaptive Testing (NCAT), which formally redefines CAT as a reinforcement learning problem and directly learns selection algorithm from real-world data. Specifically, a bilevel optimization is defined and simplified under CAT's application scenarios to make the algorithm learnable. Furthermore, to address the CAT task effectively, we tackle it as an equivalent reinforcement learning problem and propose an attentive neural policy to model complex non-linear interactions. Extensive experiments on real-world datasets demonstrate the effectiveness and robustness of NCAT compared with several state-of-the-art methods.

1 Introduction

Computerized Adaptive Testing (CAT) is a novel and promising testing mode, which provides an efficient way to accurately measure student ability/proficiency level of a particular domain (e.g., Mathematics) by providing few questions. In contrast to traditional paper-and-pencil tests, CAT has much higher efficiency through tailoring a personalized test procedure for each examinee (Cheng 2009; Wang et al. 2016). Therefore, it has been widely used in various standardized tests, e.g., GRE. The "adaptive" in CAT refers to selecting the best suited question for each student, based on her current estimated proficiency.

To realizing such adaptability, CAT commonly requires two core components that work alternately: **(1) Cognitive Diagnosis Model (CDM)** and **(2) Selection Algorithm**. Figure 1 illustrates a toy example of a typical CAT. At each step of the test (index by t), **(1) CDM** first estimates a student's current proficiency based on previous responses. The most famous is item response theory (IRT):

$$P(a_{i,j}=1)=\sigma\left[\alpha_j(\theta_i-\beta_j)\right], \tag{1}$$

where $\sigma[\cdot]$ is the logistic function and $a_{i,j}$ is student i's response to question j (1 indicates a correct response). Each question is represented as two pre-calibrated parameters $\alpha, \beta \in \mathbb{R}$, called *discrimination* and *difficulty* respectively (Embretson and Reise 2013). IRT represents student's responses with a latent proficiency $\theta \in \mathbb{R}$, called *ability*. Recently, deep learning-based CDM (Wang et al. 2020a) uses more informative multidimensional vector to represent this proficiency θ. **(2)** The **selection algorithm** then selects the next question, guided by her current proficiency estimate $\hat{\theta}_i^t$ from CDM. Most of algorithms are model-specific, which are specially designed by experts according to the characteristics of different CDMs. E.g. (Chang 2015) proposed Maximum Fisher Information (MFI) specially designed for IRT, where the question with high-discrimination and whose difficulty is close to student's proficiency estimate tends to be selected (i.e., higher α and $\beta=\theta$ in Eq.(1)).

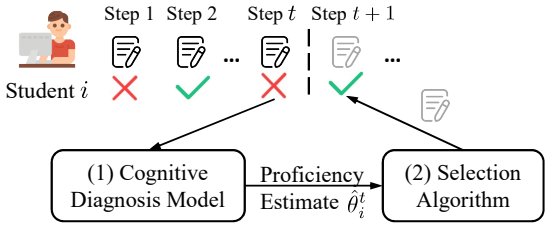


Figure 1: Illustration of a typical CAT procedure.

However, the adaptability in such selection algorithms is limited in three aspects:

- For students, the selection algorithm's efficiency heavily relies on the accuracy of current estimate $\hat{\theta}$. Thus, it lacks robustness and may cause severe loss of information using single θ to summarize complex interactions between student and question (DiBello, Roussos, and Stout 2007).
- For CDM, we have to understand how a specific CDM works in detail to design the matched selection algorithms (i.e., model-specific). Although many active learn-

*Corresponding Author.

ing methods (Bi et al. 2020) utilize CDM's output uncertainty to achieve the model-agnostic, different characteristics in individual CDMs are also ignored.

- For questions, such pre-defined algorithms usually have individual "preference" in selections (e.g., the questions with high discrimination are selected with relatively high frequency using MFI), which inevitably affect exposure control and decreases test security (Segall 2005).

Rather than hand-designing another sophisticated selection algorithms, we propose a fully adaptive framework named **N**eural **C**omputerized **A**daptive **T**esting (NCAT), which regards CAT as a reinforcement learning problem and attempts to learn a neural algorithm from large-scale students response data. In our framework, to make the selection algorithm learnable and directly capture both student interactions and characteristics of the given CDM, we redefine it as the objective of a bilevel optimization, similar to the meta-learning method (Ghosh and Lan 2021). Then, in order to simulate the dynamic interaction in CAT and solve the optimization effectively, we formally transform it into an equivalent reinforcement learning problem, further controlling question exposure rate. Subsequently, we propose a neural selection algorithm to model the complex non-linear interactions between students and questions, where two attentive modules are designed: 1) Double-channel Performance Learning (PL) module separately captures the information of versatile student performance; 2) Contradiction Learning (CL) module further identifies and extracts perturbations in student performance (e.g., guess and slip factors). Finally, we optimize the algorithm with efficient Q-learning for selecting the next question.

Extensive experiments on three real-world datasets demonstrate the effectiveness of our NCAT, i.e., measuring student proficiency accurately with the fewest questions. Furthermore, it is robust even at high noise rate and achieves exposure control under the CAT settings.

2 Related Work

Computerized Adaptive Testing. CAT consists of two components: cognitive diagnosis model (CDM) and selection algorithm. These two work alternately until the end of test (according to certain stopping rules) and then output the student's proficiency level estimated in the last step, feeding back to herself or instructors in a visual way to improve future learning. The goal of CAT is to accurately measure the proficiency of students by providing as few questions as possible (Chang 2015). Therefore, CAT is also a process of parameter estimation. The selection algorithm is the core component of CAT. For a long time, most algorithms were specially designed for IRT models, such as Maximum Fisher Information (MFI) (Lord 2012), Kullback-Leibler Information Index (KLI) (Chang and Ying 1996), and their multivariate extensions (Hooker, Finkelman, and Schwartzman 2009; Rudner 2002). Recently, MAAT (Bi et al. 2020) and BOBCAT (Ghosh and Lan 2021) leverage active learning and bilevel optimization in meta learning respectively to design algorithms, which show good performance and adaptability in deep neural network-based CDM (Wang et al. 2020a). However, these methods fail to adapt to real CAT scenarios, e.g., BOBCAT doesn't consider the complexity in student-question interactions and is unable to keep exposure balance in the application.

Reinforcement Learning. Reinforcement learning (RL) is the training of models to make a sequence of decisions. To get the model to do what we want, the model gets task-specific rewards for the actions it performs. The goal of the RL agent is to learn a policy π: which maximizes the expected cumulative reward of trajectory. Deep reinforcement learning, as one of state-of-the-art techniques (Arulkumaran et al. 2017), has shown superior abilities in many fields, such as games (Hessel et al. 2018; Rajeswaran, Mordatch, and Kumar 2020), robotics (Hu et al. 2020; Haarnoja et al. 2018) and recommender systems (Chen et al. 2019; Zhao et al. 2021). The biggest difficulty in applying RL to CAT is the definition of reward, facilitating the selection algorithm to learn from data and adapt to the given CDM. For instance, although (Nurakhmetov 2019; Li et al. 2020) proposed reinforcement learning methods to learn selection algorithms/policies, they can not be verified on real-world dataset.

3 Neural Computerized Adaptive Testing Framework

In this section, we first formalize the learnable selection algorithm in NCAT on the perspective of bilevel optimization and then transform it into an equivalent reinforcement learning problem to solve it effectively.

Problem Statement For each student, the proficiency level denoted as $\theta \in \mathbb{R}^d$ and θ_0 is her true value (unknown), where d refers to the dimension of the proficiency level (e.g., the number of knowledge concepts to be tested). During the test, the question q she has already answered is denoted as a tuple (q, a), where a equals 1 if she answers q correctly and 0 otherwise. Next, we provide the following definitions:

Typical CAT Process. Given the question bank $\mathcal{J} = \{q_1, q_2, ..., q_{|\mathcal{J}|}\}$, a complete CAT system includes two components: **(1)** A CDM $\mathcal{M}$, modeling her proficiency by predicting the probability she answers the question q correctly (binary classification), denoted as $\mathcal{M}(q|\theta) \in [0, 1]$; **(2)** A question selection algorithm π selects from $\mathcal{J}$ based on current estimate $\hat{\theta}$ in $\mathcal{M}$. More specifically, at step $t \in [1, T]$, CAT selects one question $q_t \sim \pi(\hat{\theta}^{t-1})$ for the student. After receiving response a_t, $\mathcal{M}$ updates and estimates new proficiency $\hat{\theta}^t$. The above process is repeated T steps. T is the same for all students in fixed-length designs, but not in the variable-length. The goal of CAT is to let the estimated proficiency close to θ_0 when the test is over, i.e., $\hat{\theta}^T \to \theta_0$.

3.1 The Learnable Selection Algorithm

Instead of hand-designing the selection algorithm, we define it as the objective of optimization, which directly learns from large-scale response data and is applied to novel students. Specifically, the learnable selection algorithm is defined under the meta learning settings (Finn, Abbeel, and Levine 2017; Lee et al. 2019): Let n denote the number of

students in the response dataset we use to train this algorithm π. The responses of student i is further divided into support set $\mathcal{D}_s^i$ and query set $\mathcal{D}_u^i$ randomly, where π sequentially select a total of t questions $\{q_1, ..., q_t\}$ with corresponding responses (i.e., $\mathcal{D}_s^i$) to estimate proficiency, and utilize it to optimize π on the query set $\mathcal{D}_u^i$. Following the bi-level paradigm in meta-learning (Franceschi et al. 2018; Ghosh and Lan 2021), the selection algorithm π in NCAT is redefined as the objective of bilevel optimization:

$$\pi^* = \arg\min_{\pi} \frac{1}{n} \sum_{i=1}^{n} \sum_{t=1}^{T} \frac{1}{|\mathcal{D}_u^i|} \sum_{(q,a)\in\mathcal{D}_u^i} l(a, \mathcal{M}(q|\hat{\theta}_i^t)), \quad (2)$$

$$\text{s.t. } \hat{\theta}_i^t = \arg\min_{\theta_i} \sum_{(q,a)\in\mathcal{D}_s^{i(t)}} l\left(a, \mathcal{M}\left(q|\theta_i\right)\right), \quad (3)$$

$$\text{where } \quad \mathcal{D}_s^{i(t)} = \{q_1, a_{i(1)}, ..., q_t, a_{i(t)}\} \quad \text{and}$$
$$q_t \sim \pi\left(q_1, a_{i(1)}, ..., q_{t-1}, a_{i(t-1)}\right).$$

In the *inner-level* optimization (Eq.(3)), the support set $\mathcal{D}_s^{i(t)}$ for student i is sequentially selected by algorithm π, according to her previous responses; we then minimize the binary cross-entropy loss $l(\cdot)$ on $\mathcal{D}_s^{i(t)}$ to estimate the proficiency $\hat{\theta}_i^t$ for the outer-level. In the *outer-level* optimization (Eq.(2)), we minimize the binary cross-entropy loss on the query set $\mathcal{D}_u^i$ of students to learn target selection algorithm π given current $\hat{\theta}$ (estimated in the inner-level).

This bilevel optimization exhibits the following desirable features: **(1)** The error of proficiency estimation is mainly caused by the difference in the selected questions, which further guides the optimization of π. Since the true θ_0 is unknown, we leverage the fit of estimate $\hat{\theta}_i^t$ on query set to measure such error (Chen, de la Torre, and Zhang 2013) in outer-level. **(2)** Because the test may stop at any time/step according to different stopping rules, we simplify the objective and sum all the test steps to minimize the loss, and these solutions are different from the previous bilevel-based method BOBCAT (Ghosh and Lan 2021). **(3)** The algorithm π is also model-agnostic. More importantly, it could be adapted to the given CDM $\mathcal{M}$ automatically by optimizing this problem for efficient selection. Once the question selection algorithm is learned, it does not need to update during CAT process and adaptively selects the next one based on previous responses.

3.2 Reinforcement Learning Formulation

In the above formulation, we notice that the selection algorithm π can be learned by solving an optimization. Formally, the target Eq.(2) can be transferred as:

$$\begin{aligned} & \min_{\pi} \frac{1}{n} \sum_{i=1}^{n} \sum_{t=1}^{T} \frac{1}{|\mathcal{D}_u^i|} \sum_{(q,a)\in\mathcal{D}_u^i} l(a, \mathcal{M}(q|\hat{\theta}_i^t)) \\ \triangleq \quad & \max_{\pi} \mathbb{E}_{i\sim\pi} \left[\sum_{t=1}^{T} -\frac{1}{|\mathcal{D}_u^i|} \sum_{(q,a)\in\mathcal{D}_u^i} l(a, \mathcal{M}(q|\hat{\theta}_i^t)) \right] \\ = \quad & \max_{\pi} \mathbb{E}_{i\sim\pi} \left[\sum_{t=1}^{T} -\mathcal{L}_{\mathcal{M}}(\mathcal{D}_u^i, \hat{\theta}_i^t) \right], \end{aligned} \quad (4)$$

where $\mathcal{L}_{\mathcal{M}}(\cdot)$ is the average binary cross-entropy loss on query set given predictor $\mathcal{M}$. As a result, the bilevel optimization is transformed into maximizing the expected cumulative reward (i.e., $-\mathcal{L}_{\mathcal{M}}(\mathcal{D}_u^i, \hat{\theta}_i^t)$) in reinforcement learning settings.

In details, as a RL problem, $\langle S, A, P, R, \gamma\rangle$ in the MDP are defined as: **(1)** State S is a set of states, and $s_t \in S$ is the available information/responses the CAT has for student i at step t, i.e., $s_t = \{q_1, a_{i(1)}, ..., q_{t-1}, a_{i(t-1)}\}$. **(2)** Action set A is the question bank $\mathcal{J}$. Each question/action is only selected at most once per student/trajectory. **(3)** Transition P is the transition function with $P(s_{t+1}|s_t, q_t)$ being the probability of seeing state s_{t+1} after taking action q_t at s_t. In CAT, the uncertainty comes from correction $a_{i(t)}$ of the student's responses at step t. **(4)** Reward R is the negative loss of the estimated proficiency of student i on query set at step t, i.e., $r_i^t = -\mathcal{L}_{\mathcal{M}}(\mathcal{D}_u^i, \hat{\theta}_i^t)$ derived from Eq.(4). Figure 2(a) illustrates the overview of our NCAT framework.

In this way, CAT is treated as a decision-making process: *Given student's previous responses and a specific CDM, which question is best suited to accurately measure her proficiency.* Actually, the transformation of student's responses, proficiency estimate, and decision process of selection algorithm affect and depend on each other, which evolves into a complicated system. Hence, compared with viewing dynamics of CAT as a whole optimization process, RL framework could explore more possible "best-fitting" questions for different students in a *long-term* view. For instance, compared with the greedy selection based on student's previous responses, the combination of sufficient and diverse questions usually provides a more comprehensive and accurate measurement of student proficiency (Bi et al. 2020; Liu et al. 2019), which will be verified in Experiments.

4 Attentive Neural Selection Algorithm

Based on the above NCAT framework in reinforcement learning, we implement the selection algorithm in it with a hierarchical attentive neural network for modeling complex interactions between students and questions. In Figure 2(b), we presented its architecture, which mainly consists of a double-channel Performance Learning (PL) component, a Contradiction Learning (CL) component, and a policy layer. First, PL separately captures the information of versatile student performance since correct and incorrect responses for students are usually imbalanced (i.e., incorrect responses are usually much fewer than the correct) (Zhou et al. 2021). Second, CL identifies and extracts contradictions in student's performance, attempting to alleviate the impact of perturbations (i.e., guess and slip factors). Finally, the policy layer makes the next selection and the well-known Q-learning algorithm is utilized to optimize the policy.

Question Embedding: Given the current state of student i, $s_t = \{q_1, a_{i(1)}, ..., q_{t-1}, a_{i(t-1)}\}$, the entire question set of $\{q_j\}$ are converted into embedding vectors $\{\mathbf{q}_j\}$ of dimension d by embedding each $q_j \in \mathcal{J}$ in a continuous space, which is an embedding matrix $\mathbf{E} \in \mathbb{R}^{|\mathcal{J}|\times d}$. Because student's different responses (correct and incorrect) on the same question provide different information, each question

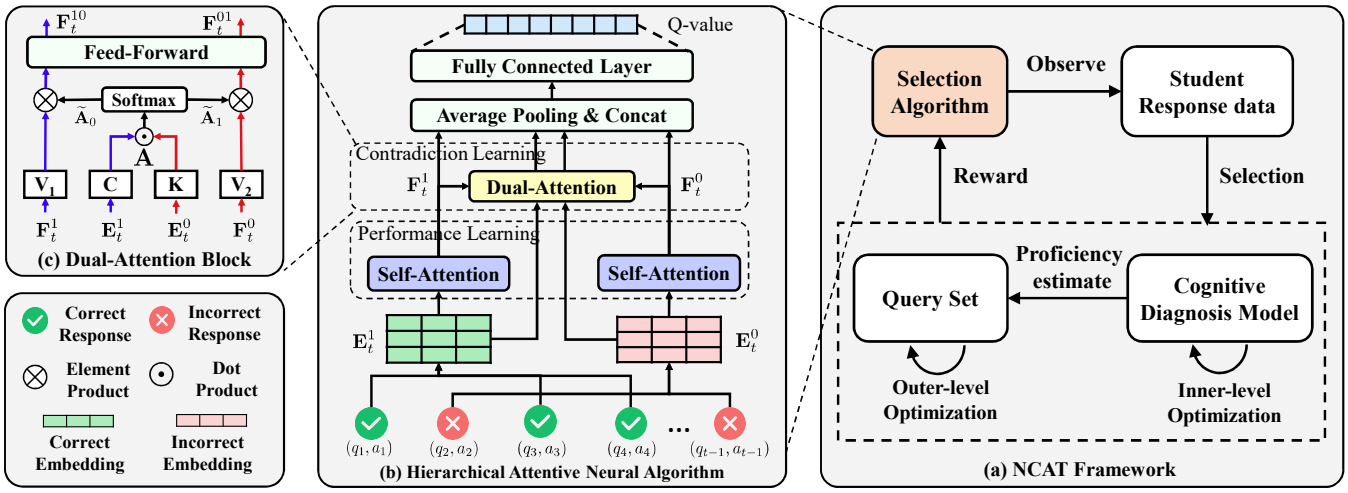


Figure 2: Illustration of our (a) NCAT framework (right), and its (b) hierarchical attentive neural network (middle) which integrates two main components, i.e., double-channel Performance Learning (PL) and (c) Contradiction Learning (CL).

has separate representations: $\mathbf{E}^1$ and $\mathbf{E}^0$ are correct and incorrect embedding matrix respectively.

4.1 Double-Channel Performance Learning

To better model student performance (i.e., observation s_t), we first use two-channel self-attention blocks to process correct and incorrect responses independently, as illustrated in Figure 2(b). Assume that k_1 and k_0 represent the number of correct and incorrect responses at step t, respectively. Denote the questions answered as two embedding matrices, $\mathbf{E}_t^z = [\mathbf{q}_1^z, \mathbf{q}_2^z, ..., \mathbf{q}_{k_z}^z]^\top$ with $z \in \{0, 1\}$ (1 denotes the correct and 0 otherwise). $\mathbf{E}_t^z$ are input into the self-attention block, which generally consists of two sub-layers, i.e., a self-attention layer and a point-wise feed-forward network. Specifically, the self-attention is defined as:

$$\mathbf{S}_t^z = \text{Attention}(\mathbf{E}_t^z \mathbf{W}_1^{z,c}, \mathbf{E}_t^z \mathbf{W}_1^{z,k}, \mathbf{E}_t^z \mathbf{W}_1^{z,v}),$$

where the projection matrices $\mathbf{W}_1^{z,c}, \mathbf{W}_1^{z,k}, \mathbf{W}_1^{z,v} \in \mathbb{R}^{d \times d}$ are the corresponding learnable parameters. Attention function is implemented by scaled dot-product operation:

$$\text{Attention}(\mathbf{C}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{C}\mathbf{K}^\top}{\sqrt{d}}\right)\mathbf{V},$$

where $\mathbf{C}$, $\mathbf{K}$, $\mathbf{V}$ represent the queries, keys, and values respectively (Vaswani et al. 2017). $\frac{1}{\sqrt{d}}$ is the scaling factor to avoid large values of the inner product (Zou et al. 2020).

To endow this component with nonlinearity and consider interactions within different latent dimensions, we apply a point-wise feed-forward network to $\mathbf{S}_t^z$. The computation is defined as:

$$\mathbf{F}_t^z = \text{FFN}(\mathbf{S}_t^z) = \sigma(\mathbf{S}_t^z \mathbf{W}^{(1)} + \mathbf{b}^{(1)})\mathbf{W}^{(2)} + \mathbf{b}^{(2)},$$

where σ is the activation function, here, we use ReLU function; the weight matrices $\mathbf{W}^{(1)}$, $\mathbf{W}^{(2)}$ are of shape $\mathbb{R}^{d \times d}$ and $\mathbf{b}^{(1)}$, $\mathbf{b}^{(2)}$ are the bias terms. So $\{\mathbf{F}_t^0, \mathbf{F}_t^1\}$ are the output of this double-channel Performance Learning component.

4.2 Contradiction Learning

The complexity of student's behavior in the CAT is mainly reflected in guess and slip factors (Vie et al. 2017; Liu et al. 2018; Gao et al. 2021): For example, when faced with a multiple-choice question with 4 options, even if the student doesn't master it, there is a 25% chance of answering it correctly (i.e., guess factor); When faced with a simple one, there may be a small chance (e.g., 5%) to answer it wrong (i.e., slip factor). In order to achieve the ultimate goal of CAT, i.e., measuring the true proficiency level of student, the selection algorithm should identify and eliminate these perturbations in her performance for better selection.

When the guess or slip factors occur, there may be some *contradictions* between the correct and incorrect responses. For example, the student answer a *Multiplication* question (harder) correctly, but answer an *Addition* question (simpler) wrong. So the possible contradiction is: *Multiplication* may be guessed or slip factor in *Addition*, or both. We design a novel dual-attention operation to capture the contradiction between correct and incorrect responses:

$$\alpha_{ij} = \frac{\mathbf{W}_2^{0,c}\mathbf{q}_i^0 \cdot (\mathbf{W}_2^{1,k}\mathbf{q}_j^1)^\top}{\sqrt{d}}, \tag{5}$$

where α_{ij} is the contradiction score of question $\mathbf{q}_i^0$ and $\mathbf{q}_j^1$, weight matrices $\mathbf{W}_2^{0,c}$, $\mathbf{W}_2^{1,k}$ are of shape $\mathbb{R}^{d \times d}$. All α_{ij} form the score matrix $\mathbf{A} \in \mathbb{R}^{k_0 \times k_1}$. Next, the contradiction scores $\mathbf{A}$ are normalized by softmax function from row and column dimension respectively: $\widetilde{\alpha}_{ij}^0 = \frac{\exp(\alpha_{ij})}{\sum_{j=1}^{k_1}\exp(\alpha_{ij})}$ and $\widetilde{\alpha}_{ij}^1 = \frac{\exp(\alpha_{ij})}{\sum_{i=1}^{k_0}\exp(\alpha_{ij})}$ form $\widetilde{\mathbf{A}}_0, \widetilde{\mathbf{A}}_1 \in \mathbb{R}^{k_0 \times k_1}$ respectively.

To further extract the information of question pairs with contradiction, we use the generated matrices $\mathbf{F}_t^1$ and $\mathbf{F}_t^0$ in Performance Learning (i.e., $\mathbf{V}_1$ and $\mathbf{V}_2$ in Figure 2(c)) to do dual-attention and feed-forward operations with scores matrices $\widetilde{\mathbf{A}}_0, \widetilde{\mathbf{A}}_1$, respectively.

$$\mathbf{F}_t^{10} = \text{FFN}(\widetilde{\mathbf{A}}_0\mathbf{F}_t^1),\ \mathbf{F}_t^{01} = \text{FFN}(\widetilde{\mathbf{A}}_1^\top\mathbf{F}_t^0),$$

where $\mathbf{F}_t^{10} \in \mathbb{R}^{k_0 \times d}$ and $\mathbf{F}_t^{01} \in \mathbb{R}^{k_1 \times d}$ are contradictory feature matrices of questions in the two channels respectively.

4.3 Policy Layer

Finally, the selection algorithm predicts the next selection's score based on the four matrices $\{\mathbf{F}_t^0, \mathbf{F}_t^1, \mathbf{F}_t^{01}, \mathbf{F}_t^{10}\}$, after multiple attention blocks that adaptively extract information of previously responses. In view of the hypothesis: the student's proficiency is unchanged during the CAT process (Wainer et al. 2000). Thus the order of each response is not important and we use average-pooling for each matrix; concatenate them into a vector $\mathbf{u}_t \in \mathbb{R}^{4d}$. The approximate action-value function, denoted by $Q_\pi(s_t, \cdot) = \left[Q_\pi(s_t, q_1), ..., Q_\pi(s_t, q_{|\mathcal{J}|})\right]$, can be represented using a feed-forward layer

$$Q_\pi(s_t, \cdot) = \sigma(\mathbf{u}_t \mathbf{W}^{(1)} + \mathbf{b}^{(1)})\mathbf{W}^{(2)} + \mathbf{b}^{(2)},$$

where $\mathbf{W}^{(1)} \in \mathbb{R}^{4d \times d}$, $\mathbf{W}^{(2)} \in \mathbb{R}^{d \times |\mathcal{J}|}$ are weight matrices and $\mathbf{b}^{(1)} \in \mathbb{R}^d$ and $\mathbf{b}^{(2)} \in \mathbb{R}^{|\mathcal{J}|}$ are the bias terms. Once we have $Q_\pi(s_t, \cdot)$, the optimal learning policy becomes readily available, which is $\pi^*(s_t) = \arg\max_{q \in \mathcal{J}} Q_\pi(s_t, q)$.

Question Selection. Since the test length of CAT is generally short, using the deterministic strategy (i.e., $\arg\max_q Q_\pi(s_t, q)$ will limit the variety of the selected questions. It will inevitably affect exposure control and test security will be compromised (Segall 2005). For example, when the test length is 10 and both the candidate questions and the first selected one are the same for all students, there will only be $2^{10} - 1 = 1023$ different questions at most totally in all tests due to the binary response. Therefore, given the current state (i.e., previous responses), the probability of selecting question q is proportional to its Q-value:

$$P(q|s_t) = \frac{e^{Q_\pi(s_t, q)/\nu}}{\sum_{q \in \mathcal{J}} e^{Q_\pi(s_t, q)/\nu}}, \tag{6}$$

where ν is a temperature parameter that is slowly reduced during the test process for greedier selection towards the end. By introducing randomness at the beginning of the test, the selected questions and test paths could be enriched due to different start points. As test progresses, the certainty of the selection continues to increase. When $\nu \to 0$, Eq.(6) is equivalent to $\arg\max Q_\pi$. Obviously, full randomization is the simplest method of exposure control which results in the same exposure rate. However, it conflicts with the idea of selecting "best fitting" questions to accelerate the measurement process. The balance between measurement accuracy and exposure rate will be further studied in Section 5.5.

4.4 Policy Learning

We use Q-Learning (Mnih et al. 2013) to learn the policy π (i.e., all parameters in the above network). To obtain a good approximate action-value function, the state-action space needs to be sufficiently explored (Li et al. 2020). The so-called ϵ-greedy exploration is adopted in the deep Q-learning algorithm. Specifically, in the t-th step, a random action is selected with probability ϵ, and a greedy action q_t is chosen with probability $1 - \epsilon$. The selection algorithm π (parameterized by $\boldsymbol{\phi}$) can be trained by minimizing the mean-squared loss function, defined as follows:

$$\mathcal{L}(\boldsymbol{\phi}) = \mathbb{E}_{(s_t, q_t, r_{i(t)}, s_{t+1}) \sim \mathbf{B}} \left[(y_t - Q_\pi(s_t, q_t))^2\right],$$
$$y_t = r_i(s_t, q_t) + \gamma \max_{q_{t+1} \in \mathcal{J}} Q_\pi(s_{t+1}, q_{t+1}),$$

where γ is the discount factor and y_t is the target value based on optimal *Bellman Equation* (Sutton and Barto 2018). $\mathbf{B} = \{(s_t, q_t, r_{i(t)}, s_{t+1})\}$ is a large replay buffer storing the past experience, where samples are taken in mini-batch training. By differentiating the loss function w.r.t. π, the gradient is:

$$\nabla_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\phi}) = \mathbb{E}\left[(y_t - Q_\pi(s_t, q_t)) \nabla_{\boldsymbol{\phi}} Q_\pi(s_t, q_t)\right].$$

5 EXPERIMENTS

In this section, we conduct extensive experiments on three real-world datasets to evaluate the effectiveness of NCAT. We mainly focus on answering the following research questions: **(RQ1)** How can NCAT outperform existing question selection algorithms under the CAT setting? **(RQ2)** What kind of question will be selected by NCAT? **(RQ3)** If NCAT well captures the complex relationship between students and questions? **(RQ4)** What's the influence of various components in NCAT?

5.1 Experimental Settings

Datasets. We use three real-world educational datasets, namely ASSIST, EXAM, and NIPS-EDU. ASSIST (Pardos et al. 2013) is collected from an online tutoring system ASSISTments and records students' practice logs on mathematics and knowledge concepts related to the questions. EXAM was collected from an online educational system that provides homework, examinations, and evaluation for students. It collected records of junior high school students on mathematical exercises. NIPS-EDU (Wang et al. 2020b) refers to the dataset in NeurIPS 2020 Education Challenge. It is collected from students' answers to questions from Eedi (an educational platform), and the datasets can be found in https://github.com/bigdata-ustc/EduData.

Data Partition and Evaluation Methods. We perform 5-fold cross validation for all datasets; for each fold, we use 60%-20%-20% students for training[1], validation, and testing respectively. Furthermore, we partition the questions responded to by each student into the support set ($\mathcal{D}_s^i$ 70%) and query set ($\mathcal{D}_u^i$, 30%). These partitions are also generated randomly in each training epoch to prevent overfitting. In the testing, **1)** we utilize different methods to select questions in $\mathcal{J}_i$; **2)** CDM updates estimation with the corresponding responses; **3)** evaluate CDM's performance on predicting binary-valued student responses on the query set $\mathcal{D}_u^i$. Therefore, we use both accuracy (ACC) and Area Under ROC (AUC) as metrics to evaluate the performance of different selection algorithms. All methods are developed and trained on a Tesla K20m GPU.

[1]The traditional pre-defined selection algorithms (e.g., MFI) have no training process.

Table 1: The performance of different methods on Student Performance Prediction task with ACC and AUC metrics. The boldfaced indicates the statistically significant improvements (i.e., two-sided t-test with $p < 0.01$) over the best baseline.

Dataset	**ASSIST**						**NIPS-EDU**						**EXAM**					
CDM	IRT			NCDM			IRT			NCDM			IRT			NCDM		
Metric	ACC						ACC						ACC					
Step	5	10	20	5	10	20	5	10	20	5	10	20	5	10	20	5	10	20
RAND	0.7099	0.7168	0.7241	0.7036	0.7096	0.7205	0.6290	0.6583	0.6868	0.6212	0.6632	0.6921	0.7212	0.7540	0.8100	0.7230	0.7618	0.8144
MFI	0.7224	0.7296	0.7412	–	–	–	0.6464	0.6765	0.7056	–	–	–	0.7495	0.7753	0.8382	–	–	–
KLI	0.7230	0.7298	0.7418	–	–	–	0.6456	0.6719	0.7016	–	–	–	0.7556	0.7841	0.8431	–	–	–
MAAT	0.7233	0.7291	0.7422	0.7268	0.7314	0.7487	0.6481	0.6742	0.7134	0.6432	0.6811	0.7180	0.7590	0.7972	0.8450	0.7623	0.8003	0.8455
BOBCAT	0.7262	0.7328	0.7488	0.7298	0.7409	0.7492	0.6559	0.6812	0.7225	0.6630	0.6952	0.7236	0.7663	0.7991	0.8444	0.7712	0.8100	0.8442
NCAT	**0.7330**	**0.7478**	**0.7562**	**0.7354**	**0.7539**	**0.7564**	**0.6606**	**0.7032**	**0.7321**	**0.6742**	**0.7159**	**0.7334**	**0.7813**	**0.8165**	**0.8516**	**0.7837**	**0.8235**	**0.8546**
Metric	AUC						AUC						AUC					
Step	5	10	20	5	10	20	5	10	20	5	10	20	5	10	20	5	10	20
RAND	0.6884	0.6978	0.7102	0.6891	0.6971	0.7191	0.6550	0.6872	0.7229	0.6591	0.6988	0.7260	0.6714	0.6923	0.7683	0.6791	0.7021	0.7795
MFI	0.6992	0.7100	0.7297	–	–	–	0.6728	0.7066	0.7458	–	–	–	0.6950	0.7188	0.7834	–	–	–
KLI	0.7004	0.7104	0.7316	–	–	–	0.6706	0.7026	0.7390	–	–	–	0.7056	0.7293	0.7869	–	–	–
MAAT	0.7112	0.7122	0.7350	0.7135	0.7211	0.7444	0.6729	0.7034	0.7481	0.6713	0.7139	0.7473	0.7013	0.7345	0.7910	0.7023	0.7376	0.7984
BOBCAT	0.7155	0.7200	0.7423	0.7181	0.7356	0.7463	0.6841	0.7100	0.7571	0.6910	0.7204	0.7622	0.7067	0.7374	0.7908	0.7107	0.7421	0.7921
NCAT	**0.7187**	**0.7318**	**0.7546**	**0.7208**	**0.7391**	**0.7525**	**0.6889**	**0.7307**	**0.7600**	**0.7043**	**0.7394**	**0.7661**	**0.7124**	**0.7489**	**0.8022**	**0.7131**	**0.7589**	**0.8152**

Compared Methods. The selection algorithm in CAT needs to rely on Cognitive Diagnosis Model (CDM) as mentioned above. Our experiment mainly involves two classic CDM: Traditional item response theory **IRT** (Embretson and Reise 2013) and recently proposed deep learning models (e.g., **NCDM** (Wang et al. 2020a)). The codes of different CDM are available at https://github.com/bigdata-ustc/EduCDM. We compare NCAT with the following selection algorithms:

- **MFI** (Lord 2012): It is one of the most widely used selection strategies which selects the one with the maximum Fisher information. This method only depends on IRT.
- **KLI** (Chang and Ying 1996): It uses Kullback-Leibler information to measure the divergence between two consecutive posteriors of proficiency. It also depends on IRT.
- **BOBCAT** (Ghosh and Lan 2021): It's the first bilevel optimization framework, which adopts an approximate gradient estimate method, for CAT to learn a data-driven selection algorithm. It is agnostic to the underlying CDM.
- **MAAT** (Bi et al. 2020): It proposes an active learning-based method, which measures questions' informativeness by calculating Expected Model Change (EMC) caused by each question. It is also agnostic to the underlying CDM.
- **RAND**: The random selection strategy is a benchmark to quantify the improvement of other methods.

Parameter Setting.[2] As 20 is sufficient for a typical test, we set the max length $T = 20$. All the parameters about questions in CDM (such as the difficulty and discrimination in IRT) will be estimated in advance using the training set as the ground-truth like (Chen, de la Torre, and Zhang 2013). We report the result of each method with its optimal hyper-parameter settings on validation set. We set the embedding size $d = 128$ and the learning rate in RL algorithm to 0.001. The temperature parameter ν in Eq.(6) is set to $2^{-0.1t}$ which is slowly reduced during test. The capacity of the replay buffer for Q-learning is set to 10000 in experiments. The exploration factor ϵ decays from 1 to 0 during training.

[2]The code is available at https://github.com/bigdata-ustc/NCAT

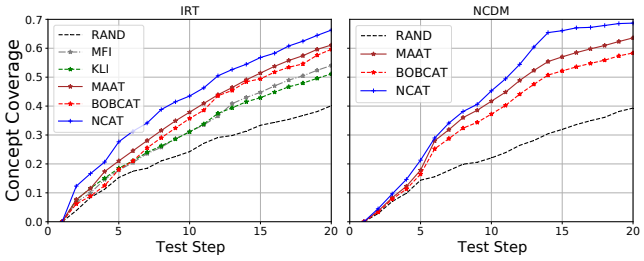


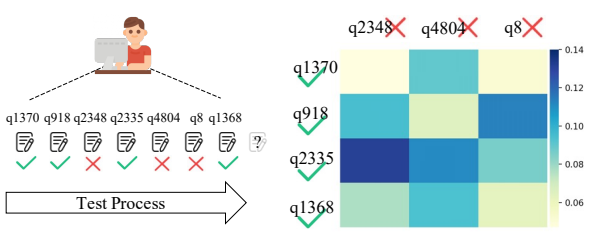
Figure 3: Diversity Comparison with Coverage Metric. The left is the result on IRT, and the right is on NCDM.

5.2 Performance Comparison (RQ1)

In order to verify the efficiency of the selection algorithm, Student Performance Prediction task is generally used in CAT. Table 1 reports the mean accuracy and AUC metric on query sets of different methods after t-th step of selection, and we show the results at step 5, 10, and 20 of tests. The results are quite consistent with our intuition. We have the following observations:

(1) We can see that NCAT achieves the best performance on the accuracy and AUC over three benchmark datasets and all steps, significantly outperforming the state-of-the-art methods. It means that our proposed method can capture students' response patterns quickly to select best-fitting questions for them, and adapt this strategy to new students.

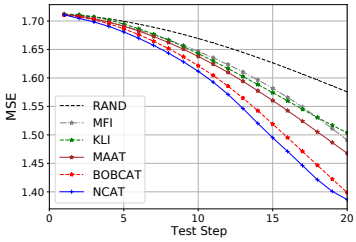
(2) Both NCAT and BOBCAT significantly outperform other pre-fixed selection algorithms (e.g., MAAT). This shows that learning the selection algorithm from data explicitly could improve the efficiency of CAT. Compared with BOBCAT, NCAT framework in reinforcement learning could improve the efficiency in selection utilizing our proposed attention mechanism.



(a)

Correct	Related Concepts	Difficulty
q1370	Unary Linear Inequality	0.069
q918	Real Number	0.150
q2335	Linear Equations in Three Variables	0.171
q1368	Cube Root	0.140

Incorrect	Related Concepts	Difficulty
q2348	Linear Equation in One Variable	0.151
q4804	Function	0.164
q8	Real Number	0.154



(b)

Figure 4: (a) shows the visualization of contradiction score matrix in student's performance and the questions' information. (b) shows MSE comparison with guess and slip factors as test step increases on EXAM dataset.

5.3 Coverage of Knowledge Concept (RQ2)

To gain a better insight into NCAT, we take a close look at the characteristics of questions selected by the algorithm. We measure the *concept coverage* of different algorithms, i.e., the proportion of knowledge concepts (e.g., Algebra and Geometry in mathematics) covered in their selected questions at each test step (Bi et al. 2020). Specifically, let $K = \{k_1, k_2, ..., k_{|K|}\}$ be the set of knowledge concepts related to all questions. At step t, if the question related to knowledge concept k is among all the selected questions $\mathcal{J}_t$, it can be denoted as $k \in \mathcal{J}_t$. The concept converge metric is

$$Cov(\pi) = \frac{1}{|K|} \sum_{k \in K} \mathbb{1}[k \in \mathcal{J}_t].$$

In Figure 3, in NCAT, the coverage grows relatively fast during the test and approaches the limit of 1. Intuitively, maximizing the concept coverage in selected questions make the measurement and diagnosis of students more *comprehensive*. The results verify that exploring in RL framework and increasing the concept diversity in selection enhance the accuracy of CAT, which is exactly what NCAT attempts to do.

5.4 Complex Exercising Interactions (RQ3)

Contradiction in Responses. To further illustrate the learning of contradiction in student's performance, we visualize the intermediate results of the contradiction score **A** in Eq.(5). Figure 4(a) illustrates a student's 7 responses before the next selection (4 correct and 3 incorrect) on EXAM dataset. We also display corresponding information about these questions, including their index, difficulty (estimated by IRT), and related knowledge concepts. The contradiction score between q2335 (correct) and q2348 (incorrect) is higher than other pairs and the correct is more difficult than the incorrect. This reveals that there may be a guess factor in q2335 or slip factor in q2348 or both. Meanwhile, although q2335 and q4804 have a similar difficulty comparison to the above, the score is much lower. That is intuitive that the relevance of knowledge concepts between them is lower than that between q2335 and q2348, thus ignoring their contradiction. These observations imply that our proposed NCAT provides a good way to capture the complex relationship between questions and students for better selections.

Table 2: Ablation analysis (Accuracy T@20) on three datasets in IRT. '↑' and '↓' indicate performance increase and drop respectively, compared with default version.

Architecture	ASSIST	NIPS-EDU	EXAM
Default	0.7562	0.7321	0.8516
NCAT-C	0.7415↓	0.7059↓	0.8351↓
NCAT-P	0.7508↓	0.7235↓	0.8417↓
$\nu \to 0$	0.7586↑	0.7329↑	0.8530↑

Table 3: Question exposure rate (mean) for different methods using simulation study (using ASSIST dataset). The default NCAT is boldfaced and $\nu = 2^{-0.1t}$.

Method	MFI	KLI	MAAT	BOBCAT	**NCAT**	NCAT ($\nu \to 0$)
Exposure	14.4%	13.7%	13.9%	16.3%	**5.2%**	15.1%

Proficiency Estimation with Guess and Slip. The ultimate goal of CAT is to estimate the student's proficiency θ. Since the true θ_0 is unknown, in addition to the Prediction task in RQ1, we also adopt simulation study: Constructing students' proficiency θ_0 artificially and generate corresponding responses, which are used for proficiency estimation (Linden, van der Linden, and Glas 2000). So we evaluate it using mean squared error (MSE) over all constructed students, i.e., $\mathbb{E}\|\hat{\theta} - \theta_0\|$. In order to simulate real testing scenarios and verify the robustness of NCAT under guess and slip factors: when the generated response is 0, there is a 25% probability of being changed to 1 (i.e., guess factor); when the response is 1, there is a 5% probability of being changed to 0 (i.e., slip factor). We show the MSE of EXAM in Figure 4(b). We conduct this with simple IRT and utilize the proficiency parameters learned on the whole datasets as the ground truth, instead of generating them. We can see that even in the presence of multiple perturbations, NCAT framework also performs well on MSE metric, further confirming its high accuracy and robustness in proficiency estimation.

5.5 Ablation Study (RQ4)

Since there are many components in our framework, we analyze their impacts via an ablation study. Table 2 shows the performance of our default method and its 3 variants on three

datasets. We introduce the variants and analyze their effect:

(1) NCAT-C and NCAT-P: They are the variants of NCAT that only use Contradiction Learning and Performance Learning module respectively. NCAT-C only captures the contradiction in student's observed performance but prohibits the matrix generated by Performance Learning, which significantly reduces its accuracy. NCAT-P performs worse than the default because it ignores the perturbations in student's performance (i.e. guess and slip factors). This enables us to safely draw the conclusion that it is advisable to model the complex interactions between students and questions considering the contradiction aspects.

(2) $\nu \to 0$: $\nu \to 0$ means that the policy is deterministic which selects the question with the largest Q-value. In other words, measurement accuracy is taken into account, but exposure rate is not. Not surprisingly, although its results are slightly better than the default setting, NCAT achieves a balance between accuracy and exposure control combined with the results in Table 3. The overall low exposure rate can effectively ensure the safety and fairness of test.

6 Conclusions

In this paper, we proposed a fully adaptive CAT framework called NCAT which provides a general way to learn the selection algorithm from real-world data for online education. Specifically, in order to conquer the limitations in manual-designed selection algorithms, NCAT is recast and solved in effective reinforcement learning settings. Furthermore, an attentive neural selection algorithm was proposed as the implementation to model complex non-linear interactions in tests. Extensive experiments demonstrated that NCAT can successfully capture complex relationships between students and questions (e.g., guess and slip factors), and measure students' proficiency accurately, reducing test length.

Acknowledgments

This research was partially supported by grants from the National Natural Science Foundation of China (Grants No. 61922073, U20A20229), the Fundamental Research Funds for the Central Universities (Grants No. WK2150110021), the Key Program of Natural Science Project of Educational Commission of Anhui Province (No. KJ2020A0036), and the Anhui Provincial Natural Science Foundation(No. 2108085QF272).

References

Arulkumaran, K.; Deisenroth, M. P.; Brundage, M.; and Bharath, A. A. 2017. Deep Reinforcement Learning: A Brief Survey. *IEEE Signal Processing Magazine*, 34(6): 26–38.

Bi, H.; Ma, H.; Huang, Z.; Yin, Y.; Liu, Q.; Chen, E.; Su, Y.; and Wang, S. 2020. Quality meets Diversity: A Model-Agnostic Framework for Computerized Adaptive Testing. In *2020 IEEE International Conference on Data Mining (ICDM)*, 42–51. IEEE.

Chang, H.-H. 2015. Psychometrics behind computerized adaptive testing. *Psychometrika*, 80(1): 1–20.

Chang, H.-H.; and Ying, Z. 1996. A global information approach to computerized adaptive testing. *Applied Psychological Measurement*, 20(3): 213–229.

Chen, J.; de la Torre, J.; and Zhang, Z. 2013. Relative and absolute fit evaluation in cognitive diagnosis modeling. *Journal of Educational Measurement*, 50(2): 123–140.

Chen, M.; Beutel, A.; Covington, P.; Jain, S.; Belletti, F.; and Chi, E. H. 2019. Top-k off-policy correction for a REINFORCE recommender system. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 456–464.

Cheng, Y. 2009. When cognitive diagnosis meets computerized adaptive testing: CD-CAT. *Psychometrika*, 74(4): 619–632.

DiBello, L.; Roussos, L.; and Stout, W. 2007. Review of cognitively diagnostic assessment and a summary of psychometric models. CR Rao, & S. Sinharay (Eds.), Handbook of statistics, Vol. 26: Psychometrics (pp. 970–1030).

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

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, 1126–1135. PMLR.

Franceschi, L.; Frasconi, P.; Salzo, S.; Grazzi, R.; and Pontil, M. 2018. Bilevel programming for hyperparameter optimization and meta-learning. In *International Conference on Machine Learning*, 1568–1577. PMLR.

Gao, W.; Liu, Q.; Huang, Z.; Yin, Y.; Bi, H.; Wang, M.-C.; Ma, J.; Wang, S.; and Su, Y. 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.

Ghosh, A.; and Lan, A. 2021. BOBCAT: Bilevel Optimization-Based Computerized Adaptive Testing. arXiv:2108.07386.

Haarnoja, T.; Zhou, A.; Abbeel, P.; and Levine, S. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, 1861–1870. PMLR.

Hessel, M.; Modayil, J.; Van Hasselt, H.; Schaul, T.; Ostrovski, G.; Dabney, W.; Horgan, D.; Piot, B.; Azar, M.; and Silver, D. 2018. Rainbow: Combining improvements in deep reinforcement learning. In *Thirty-second AAAI conference on artificial intelligence*.

Hooker, G.; Finkelman, M.; and Schwartzman, A. 2009. Paradoxical results in multidimensional item response theory. *Psychometrika*, 74(3): 419–442.

Hu, J.; Niu, H.; Carrasco, J.; Lennox, B.; and Arvin, F. 2020. Voronoi-Based Multi-Robot Autonomous Exploration in Unknown Environments via Deep Reinforcement Learning. *IEEE Transactions on Vehicular Technology*, 69(12): 14413–14423.

Lee, H.; Im, J.; Jang, S.; Cho, H.; and Chung, S. 2019. Melu: Meta-learned user preference estimator for cold-start recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1073–1082.

Li, X.; Xu, H.; Zhang, J.; and hua Chang, H. 2020. Deep Reinforcement Learning for Adaptive Learning Systems. arXiv:2004.08410.

Linden, W. J.; van der Linden, W. J.; and Glas, C. A. 2000. *Computerized adaptive testing: Theory and practice*. Springer.

Liu, Q.; Huang, Z.; Yin, Y.; Chen, E.; Xiong, H.; Su, Y.; and Hu, G. 2019. Ekt: Exercise-aware knowledge tracing for student performance prediction. *IEEE Transactions on Knowledge and Data Engineering*, 33(1): 100–115.

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): 1–26.

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

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; and Riedmiller, M. 2013. Playing Atari with Deep Reinforcement Learning. arXiv:1312.5602.

Nurakhmetov, D. 2019. Reinforcement learning applied to adaptive classification testing. In *Theoretical and Practical Advances in Computer-based Educational Measurement*, 325–336. Springer, Cham.

Pardos, Z. A.; Baker, R. S.; San Pedro, M. O.; Gowda, S. M.; and Gowda, S. M. 2013. Affective states and state tests: Investigating how affect throughout the school year predicts end of year learning outcomes. In *Proceedings of the third international conference on learning analytics and knowledge*, 117–124.

Rajeswaran, A.; Mordatch, I.; and Kumar, V. 2020. A Game Theoretic Framework for Model Based Reinforcement Learning. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 7953–7963. PMLR.

Rudner, L. M. 2002. An examination of decision-theory adaptive testing procedures. In *annual meeting of the American Educational Research Association*.

Segall, D. O. 2005. Computerized adaptive testing. *Encyclopedia of social measurement*, 1: 429–438.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement learning: An introduction*. MIT press.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

Vie, J.-J.; Popineau, F.; Bruillard, É.; and Bourda, Y. 2017. A review of recent advances in adaptive assessment. *Learning analytics: fundaments, applications, and trends*, 113–142.

Wainer, H.; Dorans, N. J.; Flaugher, R.; Green, B. F.; and Mislevy, R. J. 2000. *Computerized adaptive testing: A primer*. Routledge.

Wang, F.; Liu, Q.; Chen, E.; Huang, Z.; Chen, Y.; Yin, Y.; Huang, Z.; and Wang, S. 2020a. Neural cognitive diagnosis for intelligent education systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 6153–6161.

Wang, S.; Lin, H.; Chang, H.-H.; and Douglas, J. 2016. Hybrid computerized adaptive testing: From group sequential design to fully sequential design. *Journal of Educational Measurement*, 53(1): 45–62.

Wang, Z.; Lamb, A.; Saveliev, E.; Cameron, P.; Zaykov, Y.; Hernández-Lobato, J. M.; Turner, R. E.; Baraniuk, R. G.; Barton, C.; Jones, S. P.; Woodhead, S.; and Zhang, C. 2020b. Diagnostic questions: The neurips 2020 education challenge. *arXiv preprint arXiv:2007.12061*.

Zhao, X.; Gu, C.; Zhang, H.; Yang, X.; Liu, X.; Liu, H.; and Tang, J. 2021. DEAR: Deep Reinforcement Learning for Online Advertising Impression in Recommender Systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 750–758.

Zhou, Y.; Liu, Q.; Wu, J.; Wang, F.; Huang, Z.; Tong, W.; Xiong, H.; Chen, E.; and Ma, J. 2021. Modeling Context-aware Features for Cognitive Diagnosis in Student Learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2420–2428.

Zou, L.; Xia, L.; Gu, Y.; Zhao, X.; Liu, W.; Huang, J. X.; and Yin, D. 2020. Neural interactive collaborative filtering. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 749–758.