mathematics

MDPI

Article

Parallel Prediction Method of Knowledge Proficiency Based on Bloom's Cognitive Theory

Tiancheng Zhang [1,*], Hanyu Mao [1], Hengyu Liu [1], Yingjie Liu [1], Minghe Yu [1], Wenhui Wu [1], Ge Yu [1], Baoze Wei [2] and Yajuan Guan [2]

[1] School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China; 2010676@stu.neu.edu.cn (H.M.); 1710589@stu.neu.edu.cn (H.L.); 1801798@stu.neu.edu.cn (Y.L.); yuminghe@mail.neu.edu.cn (M.Y.); 2272047@stu.neu.edu.cn (W.W.); yuge@mail.neu.edu.cn (G.Y.)

[2] Department of Energy Technology, Aalborg University, 9220 Aalborg, Denmark; bao@energy.aau.dk (B.W.); ygu@energy.aau.dk (Y.G.)

* Correspondence: tczhang@mail.neu.edu.cn

Abstract: Knowledge proficiency refers to the extent to which students master knowledge and reflects their cognitive status. To accurately assess knowledge proficiency, various pedagogical theories have emerged. Bloom's cognitive theory, proposed in 1956 as one of the classic theories, follows the cognitive progression from foundational to advanced levels, categorizing cognition into multiple tiers including "knowing", "understanding", and "application", thereby constructing a hierarchical cognitive structure. This theory is predominantly employed to frame the design of teaching objectives and guide the implementation of teaching activities. Additionally, due to the large number of students in real-world online education systems, the time required to calculate knowledge proficiency is significantly high and unacceptable. To ensure the applicability of this method in large-scale systems, there is a substantial demand for the design of a parallel prediction model to assess knowledge proficiency. The research in this paper is grounded in Bloom's Cognitive theory, and a Bloom Cognitive Diagnosis Parallel Model (BloomCDM) for calculating knowledge proficiency is designed based on this theory. The model is founded on the concept of matrix decomposition. In the theoretical modeling phase, hierarchical and inter-hierarchical assumptions are initially established, leading to the abstraction of the mathematical model. Subsequently, subject features are mapped onto the three-tier cognitive space of "knowing", "understanding", and "applying" to derive the posterior distribution of the target parameters. Upon determining the objective function of the model, both student and topic characteristic parameters are computed to ascertain students' knowledge proficiency. During the modeling process, in order to formalize the mathematical expressions of "understanding" and "application", the notions of "knowledge group" and "higher-order knowledge group" are introduced, along with a parallel method for identifying the structure of higher-order knowledge groups. Finally, the experiments in this paper validate that the model can accurately diagnose students' knowledge proficiency, affirming the scientific and meaningful integration of Bloom's cognitive hierarchy in knowledge proficiency assessment.

Keywords: knowledge proficiency; cognitive diagnosis; Bloom's cognitive taxonomy; matrix factorization; parallel and distributed processing

MSC: 62H22

check for updates

Citation: Zhang, T.; Mao, H.; Liu, H.; Liu, Y.; Yu, M.; Wu, W.; Yu, G.; Wei, B.; Guan, Y. Parallel Prediction Method of Knowledge Proficiency Based on Bloom's Cognitive Theory. *Mathematics* **2023**, *11*, 5002. https://doi.org/10.3390/math11245002

Academic Editor: Simeon Reich

Received: 3 November 2023
Revised: 14 December 2023
Accepted: 17 December 2023
Published: 18 December 2023

1. Introduction

With the continuous advancement of internet and artificial intelligence technologies, AI algorithms have significantly impacted various aspects of our lives, such as smart city development [1–3], energy management [4,5], and financial services [6,7]. Similarly, the field of education has been continually evolving due to the influence of artificial intelligence [8–13] and internet technologies [14,15]. This is particularly evident in the context of 'Internet + Education', a new educational model that has emerged in recent years.

"Internet + Education" is a new educational model that has emerged in recent years. In this innovative educational paradigm, "tailored teaching" remains a crucial element in cultivating high-quality and highly capable talents [16]. As students' proficiency in knowledge reflects their mastery of it and embodies their cognitive differences, the precise calculation of students' knowledge proficiency is particularly vital for personalized online education [17,18]. It can guide the construction of personalized learning spaces and the creation of intelligent teaching environments.

The prediction of knowledge proficiency has consistently been a focal point in the field of education [19], forming the early research framework for computing knowledge proficiency in educational statistics. One of the early cognitive diagnostic models was the DINA model proposed by Torre et al. in 2009 [20]. The DINA model utilizes a discrete binary vector, denoted as $\alpha_i = \{\alpha_{i1}, \alpha_{i2}, \cdots, \alpha_{ik}\}$, to describe a student's cognitive characteristics. This binary vector indicates whether a student has mastered the knowledge concept associated with a given question. Specifically, $\alpha_{i1} = 1$ signifies that student i has mastered knowledge concept 1, and, vice versa, $\alpha_{i1} = 0$. The model sets the dimension of the vector to be equal to the number of knowledge concepts, while also considering a given question–knowledge concept matrix, denoted as Q, to interpret diagnostic results. With the application of data mining methods in knowledge proficiency prediction tasks, several models for computing knowledge proficiency based on matrix decomposition [21–23] and deep learning [24–35] have been introduced. The Knowledge Proficiency Tracking (KPT) [36] model stands as the exemplar among matrix factorization-based models, which refines traditional matrix factorization by incorporating prior information from the question feature matrix V and integrating a time window for log slicing to track interpretable student knowledge proficiency, represented by the matrix U. The Deep Knowledge Tracing (DKT) model [37], based on recurrent neural networks, employs the hidden vectors of recurrent neural networks to represent a student's knowledge state and subsequently uses this representation to predict the student's performance. Additionally, the surge in the number of students participating in real-world online education systems, such as Massive Open Online Courses, has markedly increased the time required to calculate knowledge proficiency, rendering it significantly high and unacceptable. This spike in time consumption deems the existing methods impractical for efficient evaluation and hinders the seamless progression of the educational process. The challenge now lies in overcoming this bottleneck to facilitate timely assessments and feedback, which are crucial for enhancing the learning experience and outcomes.

In summary, previous approaches to computing knowledge proficiency have often focused on dynamic changes in students' cognition, incorporating a temporal dimension to form three-dimensional tracking models, thereby establishing a comprehensive horizontal framework. However, given that cognition is a complex entity, Bloom's Cognitive Theory [38,39] in educational psychology has explicitly emphasized its hierarchical nature. It suggests that authentic cognitive structures should progress step by step. Therefore, it is imperative, within the framework of educational theory, to construct a longitudinal model for computing knowledge proficiency, enabling a multi-dimensional quantification of cognitive assessment. In the field of education, Bloom's Cognitive Theory primarily serves for setting educational objectives, which is reflected in the examination of different cognitive levels in questions. For example, when a student correctly answers, "the Pacific Ocean" to the question, "What is the largest ocean in the world?" it indicates a fundamental level of "knowing." To some extent, questions also assess higher levels of cognitive understanding, making them useful for item test and question reforms. In essence, current research has largely focused on constructing dynamic models for tracking the learning process to a certain extent, yet it has overlooked the hierarchical nature of cognition emphasized in educational psychology. This text proposes the construction of a longitudinal model for computing knowledge proficiency, modeling how students' proficiency changes under examinations of different cognitive levels in questions. As of now, in knowledge proficiency prediction models, no evidence has been found of the incorporation of Bloom's Cognitive Theory to address the prediction of knowledge proficiency. In addition, to overcome the bottleneck of time consumption, there is a pronounced demand for designing a parallel prediction model that can expediently assess knowledge proficiency across a large student population. By employing a parallel processing approach, it is envisaged that the evaluation time will be drastically reduced, making the method viable for large-scale systems and ensuring that educators and learners alike benefit from timely insights into the learning progress.

To address the aforementioned issues, this section proposes a Knowledge Proficiency Parallel Prediction Model based on Bloom's Cognitive Theory (Bloom Cognitive Diagnosis Model, BloomCDM). Drawing inspiration from probabilistic matrix factorization, BloomCDM endeavors to incorporate the Bloom hierarchical priors into the features of questions, thereby influencing the prediction of the student cognitive vector. The primary innovations and contributions of this paper include:

(1) Considering educational psychological theories, BloomCDM considers the three hierarchical levels of "knowing", "understanding", and "application", as well as students' proficiency features. BloomCDM employs parallel processing to model the changes in students' knowledge proficiency under different cognitive levels examined by questions. It projects the question feature matrix into the cognitive space of each level, denoted as parameters $\mathrm{V}^{\mathrm{R}}, \mathrm{V}^{\mathrm{C}}$, and V^{A}, and associates student proficiency parameters with question feature parameters based on the assumed relationships.

(2) To address the challenge of extracting hierarchical structure information from sparse data, a method for discovering higher-order knowledge groups is designed based on the specific application scenario described in this paper. This method can uncover the structure of higher-order knowledge groups from the original matrix of answered questions and extract valuable structural information, providing strong support for knowledge proficiency prediction based on Bloom's Cognitive Theory.

(3) Experimental results on two real-world online education datasets demonstrate that BloomCDM can effectively model students' learning and forgetting behaviors, continuously track their knowledge levels in real-time, and outperform existing models in predictive performance.

2. Related Work Problem

In this section, we categorize the related research on knowledge proficiency prediction into three main groups: Cognitive Diagnosis, Matrix Factorization, and Neural Networks.

Cognitive diagnosis, as one of the most challenging issues in educational science, has received widespread attention in the field of educational statistics. It involves the assessment of a subject's cognitive abilities, holding significant weight in psychometrics. Cognitive diagnostic models serve as crucial methods for conducting cognitive diagnostic measurements. These models are primarily categorized into continuous and non-continuous models. Continuous models use continuous variables to represent students' latent abilities. The Item Response Theory (IRT) proposed by Harvey et al. in 1999 [40] is a standard example of a continuous model. In this model, the prediction of students' performance on items is achieved by considering the one-dimensional latent variable θ representing student ability, along with latent variables representing item features, and utilizing a Logistic model. The Multidimensional Item Response Theory (MIRT), proposed by Reckase et al. in 2009 [41], is a multidimensional version of the IRT. It considers the multidimensional nature of abilities $\theta = (\theta_1, \ldots, \theta_m)^T$ and builds a model for predicting student performance based on assumptions of the monotonicity and independence of abilities. Following the classical continuous models, scholars have proposed improved IRT models in different application scenarios. Recognizing that latent traits (referred to as "abilities") in humanities possess a hierarchical structure, the Hierarchical Ordered Cognitive Diagnosis Model (HO-CDM) [42] was designed to calculate high- and low-level latent variables based on linear hierarchical assumptions. Considering the hierarchical structure of ability levels, subjects with similar ability levels are categorized to obtain the latent structure of the subject group. The common approach is to use the Mixed Rasch Model (MixIRT) [43]. The MixIRT model obtains latent classifications of latent traits that are not easily observed, allowing for the assessment of multidimensionality in the measurement structure and the detection of differences in subjects' response styles. Among non-continuous models, the most classical is the DINA model, which models students' levels of cognition using multidimensional discrete cognitive vectors. Improved versions of the DINA model include Seq-G-Dina [44] and pG-DINA [45]. In cases where the scoring rules for items are discretely multi-valued, Seq-G-Dina sets restricted mastery levels in conjunction with the generalized DINA model to obtain students' discrete cognitive vectors. pG-DINA, in comparison to the G-DINA model, handles the multi-level knowledge concept. It uses multiple values to represent different levels and calculates students' discrete cognitive vectors based on the multi-level mastery of knowledge concept.

Matrix factorization is a fundamental technique used in recommender systems to uncover latent semantic relationships. It is also a commonly used algorithm in educational data mining. There are two main application scenarios: student performance prediction and knowledge proficiency prediction. In student performance prediction, the common approach involves analyzing academic performance data to map each student into a latent feature vector. This is followed by predicting student performance on specific items. In 2010, Thai-Nghe et al. and Xiong et al. [46] proposed a matrix factorization method that considers time effects to predict student performance, aiming to capture the dynamic changes in learners' knowledge levels over time. This approach incorporates a dynamic time dimension into the task, resulting in a three-dimensional (student, item, time) matrix factorization model. In 2016, P. Nedungadi et al. [47] introduced a personalized weighted multi-relational matrix factorization model to enhance prediction accuracy in student performance tasks. The model proposed two methods for designing multiple relations. The first method considers the global cognitive average level and item deviation term. The second method only considers the item deviation term. Overall, matrix factorization has been proven to be a highly effective model for predicting student performance. These methods primarily obtain students' latent knowledge vectors through factorization. However, a common limitation of previous work is that these models operate like black boxes, and the resulting student representations are difficult to interpret. In other words, the latent vectors obtained from factorization lack interpretability and can only be considered as feature vectors. Defining them as students' cognitive vectors with the dimension of the knowledge concept is somewhat strained. Therefore, in the field of statistics, adding interpretable knowledge concept features to factorization models presents a new challenge in mining students' knowledge proficiency. The first knowledge proficiency prediction model based on matrix factorization was proposed by Chen et al. in 2017 [36], known as the KPT model, which improved upon traditional matrix factorization. The specific method involved adding prior information in the form of a Q matrix to the item feature matrix V matrix. The Q matrix is a binary item–knowledge concept matrix and is a necessary condition for cognitive diagnostic models. The paper further refined and used Q-matrix-based partial orders to reduce the subjective influence of expert annotations. It constructed a partial order set D_u to more accurately capture the pairwise relationships between two knowledge concepts (p, q). The experimental results demonstrated that KPT exhibited an excellent performance in a multi-model, multi-data comparison. Subsequently, in 2020, Z. Huang et al. [48] developed an improved version called the Enhanced Knowledge Proficiency Tracking (EKPT) model based on KPT to enhance prediction performance. The paper argued that KPT's partial order based on the Q matrix only models the feature representation for each individual item, ignoring the relationships between items. Therefore, in the EKPT model, a neighbor set N_{V_j} is defined for each item j, containing items similar to it with the same knowledge concept. Each item's V_j feature consists of two parts: the grouped feature vector of neighboring items and the feature vector of each item, constructing a dynamic cognitive diagnostic model influenced by multiple item features.

With the rapid advancement of neural networks, student data and types are becoming more refined, making education data mining based on deep learning a recent research focus. However, because knowledge proficiency calculation belongs to an interpretable task, feature vectors need to possess cognitive interpretability. The current situation, where deep learning has limited interpretative power, runs counter to this requirement. Therefore, knowledge proficiency calculation based on deep learning is in its infancy and represents a completely new research area. Recently, there have been two models for estimating student knowledge proficiency based on deep learning: DIRT [24] and NeuralCDM [49]. The Deep Item Response Theory (DIRT) model, proposed by Song Cheng et al. in 2019, challenges the conventional Item Response Theory (IRT) by contending that it merely utilizes students' response outcomes for cognitive diagnosis, without fully leveraging the semantics of the questions. The paper proposes an enhanced semantic deep learning framework for cognitive diagnosis. This involves reinforcing the influence of question text on question features through a neural network model. The IRT model is employed as the loss function, and the effectiveness and interpretability of the model are validated on a large-scale practical dataset. In 2020, Fei Wang et al. introduced the Neural Cognitive Diagnosis Model (NeuralCDM). The paper asserts that existing methods often employ the logistic function to mine linear interactions during student practice, which is insufficient for capturing the complex relationships between students and exercises. Therefore, the paper presents a general framework for neural cognitive diagnosis. It projects students and questions onto factor vectors, utilizes multiple neural layers for modeling, and applies the monotonicity assumption to ensure the interpretability of these two factors. This approach comprehensively considers both linear and nonlinear interactions.

3. Description and Mathematical Modeling about BloomCDM

In the field of educational knowledge proficiency prediction, the task typically occurs in large-scale examination scenarios. Therefore, the problem definition in this paper is as follows: Suppose there are N students, M questions, and K knowledge concepts. The set of students is denoted as S, and the set of questions is denoted as P. Given the student response logs R and the "question–knowledge concept" matrix Q labeled by educational experts, the objective of this paper is to model the knowledge proficiency calculation under Bloom's cognitive theory in the process of predicting student performance. This can be expressed using Equation (1):

$$F(\text{Bloom}, R_u, Q) \rightarrow U_u \tag{1}$$

In this context, Bloom serves as the theoretical prior. R_u represents the record of student u responses. Q denotes the "question–knowledge concept" matrix, which is annotated by experienced educational experts. When $Q_{jk} = 1$, it indicates that question j assesses knowledge concept k. When $Q_{jk} = 0$, it means question j does not assess knowledge concept k. $U_u = \{U_{u1}, U_{u2}, \cdots, U_{uK}\}$ represents the knowledge proficiency vector of student u, where each value in the vector signifies the proficiency of student u in knowledge concept k. From Equation (1), it can be observed that the focus of the study lies in designing the model F to incorporate the original scoring matrix R_u and the Q matrix data as the input, while embedding the interpretable Bloom prior. Not only in hierarchical modeling, where mathematical models corresponding to different levels are abstracted, but also in a vertical manner, inter-level modeling needs to be constructed to obtain the knowledge proficiency vector U_u. Table 1 lists important mathematical symbols required for the problem.

Table 1. Some important notations of BloomCDM.

Symbol	Meaning
N	Number of students
M	Number of questions
K	Number of knowledge concepts
H	Number of high-level knowledge concepts
R_{uj}	Result of student u on question j
U_u	Knowledge proficiency vector of student u
V_j^R	Vector indicating if question j assesses "knowing"
V_j^C	Vector indicating if question j assesses "understanding"
V_j^A	Vector indicating if question j assesses "application"
D_u	Partial order set for each student
R^C	Matrix of comprehension results
R^A	Matrix of application results
CV	Subgroup
PV	Parent group
P	Set of questions

Probability Matrix Factorization

The Probabilistic Matrix Factorization model (PMF) [50] assumes that the difference between the actual observed rating R_{uj} and the predicted rating follows a Gaussian distribution with a mean of 0 and a variance of σ^2, that is, σ^2. If $g(x) = 1/(1+\exp(-x))$ represents the logistic function, and it is assumed that all observed ratings are mutually independent, then we can obtain:

$$P\left(R \mid U,V,\sigma^2\right) = \prod\prod\left[N\left(R_{uj} \mid g\left(U_i^T V_j\right),\sigma^2\right)\right]^{I_{ij}} \tag{2}$$

Assuming that the feature matrices U and V for users and items respectively follow Gaussian distributions with means of 0 and variances of σ_u^2 and σ_v^2, and further assuming that the feature vectors of users in matrix U are independently and identically distributed, as well as the feature vectors of items in matrix V, the posterior distributions of the feature matrices U and V can be calculated using Bayes' theorem as follows:

$$\begin{gathered} p\left(U,V \mid R,\sigma^2,\sigma_u^2,\sigma_v^2\right) \propto p\left(R_{uj} \mid U_i^T V_j,\sigma^2\right)p\left(U \mid \sigma_u^2\right)p\left(V \mid \sigma_v^2\right) \\ = \prod_{u=1}^{N}\prod_{j=1}^{M} N\left(R_{uj} \mid U_i^T V_j,\sigma^2\right)]^{R_{ij}} \prod_{j=1}^{M} N\left(V_j \mid 0,\sigma_v^2 I\right) \prod_{u=1}^{N} N\left(U_u \mid 0,\sigma_u^2 I\right) \end{gathered} \tag{3}$$

Maximizing the posterior probability of the feature matrices U and V is equivalent to minimizing the negative logarithm of the expression above. Therefore, when the hyperparameters $\left(\sigma^2,\sigma_u^2,\sigma_v^2\right)$ are fixed, the objective function can be obtained as follows:

$$J_{PMF} = \sum_{R_u \neq 0}\left(R_{uj} - g\left(U_i^T V_j\right)\right)^2 + \frac{\sigma^2}{\sigma_u^2}\|U\|_F^2 + \frac{\sigma^2}{\sigma_v^2}\|V\|_F^2 + C \tag{4}$$

where C is a constant that does not depend on the parameters. Many well-established methods can be applied to parameter learning, such as gradient-based optimization methods commonly used in traditional matrix factorization models. Alternatively, probabilistic modeling approaches can be used, including the Expectation–Maximization algorithm (EM) [51], Markov Chain Monte Carlo (MCMC) [52], and other methods for estimating the maximum likelihood (MLE) [53] of parameters U and V.

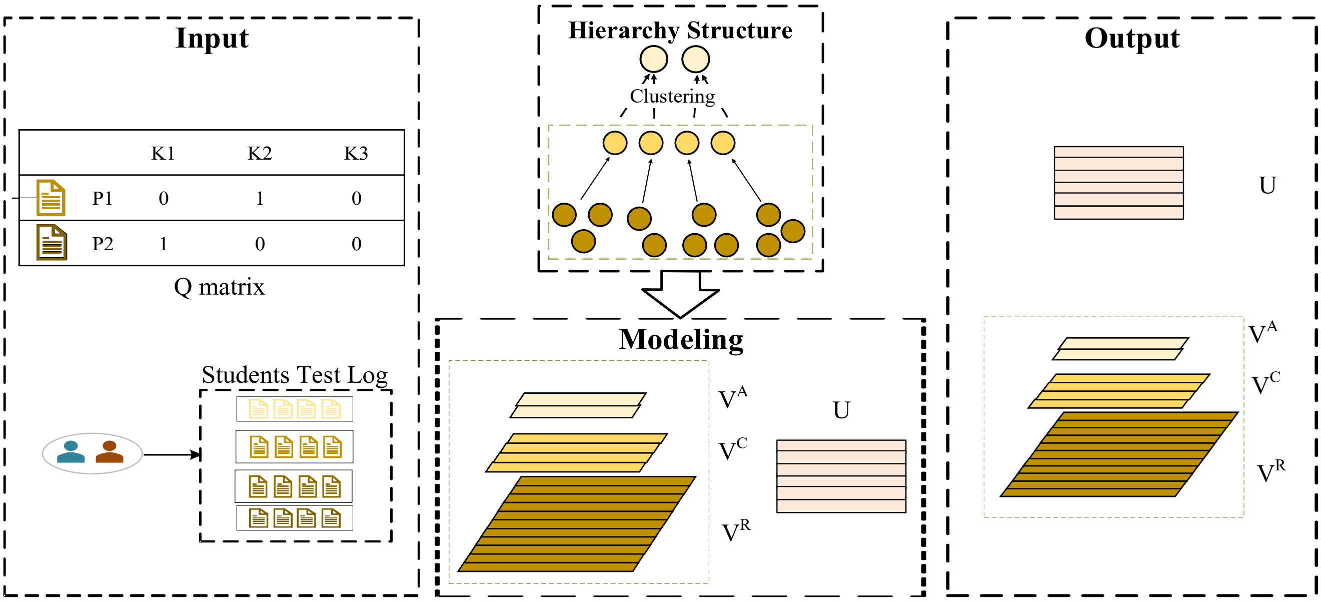
4. Knowledge Proficiency Prediction Model Based on Bloom's Cognitive Theory

In response to the collaborative filtering-based personalized question recommendation method, this section proposes the Knowledge Proficiency Calculation Model based on

Bloom's Cognitive Theory (BloomCDM). This model draws inspiration from PMF and incorporates hierarchical priors into question features to obtain knowledge proficiency vectors with good cognitive interpretability.

4.1. Framework of BloomCDM

This section introduces the proposed model framework of BloomCDM. The overall flowchart of BloomCDM is shown in Figure 1. From the figure, it can be observed that this recommendation method is mainly divided into three stages: the data input stage, the modeling stage, and the output stage. The core of the modeling stage is the embedding of the hierarchical structure, which includes modeling for "knowing", "understanding", and "application".



	K1	K2	K3
P1	0	1	0
P2	1	0	0

Figure 1. The Framework of the BloomCD model.

Specifically, the data input stage provides student log records and the expert-annotated matrix of "question–knowledge concept", defined as the R matrix and the Q matrix, respectively. In the modeling stage, hierarchical theory assumptions are applied. By combining the Q matrix, the R matrix at different levels is projected onto cognitive feature vectors in different cognitive spaces. Then, inter-level theory assumptions are applied, combining models to conduct inter-level feature modeling in a bottom-up manner. In the data output stage, after estimating the target parameters, the student's proficiency vector U and the vectors of "knowing", "understanding", and "application" features for the questions (denoted as $\mathrm{V}^{\mathrm{R}}, \mathrm{V}^{\mathrm{C}}$, and V^{A}) are output, realizing the knowledge proficiency calculation model based on Bloom's cognitive theory.

4.2. "Knowing" Model

"Knowing" implies understanding and memorization, representing the lowest level of knowledge. Students at this level do not grasp the meaning and internal relationships of the knowledge, relying on simple memorization to form a basic impression.

Based on this characteristic of "knowing", we focus on questions answered correctly. We assume that correctly answered questions have a higher probability of assessing the "knowing" level. In other words, if a question is answered correctly, it must have been tested and assessed for basic "knowing" compared to questions that were either not attempted or answered incorrectly. Therefore, at the "knowing" level, correctly answered questions take precedence over other questions, including those answered incorrectly or not attempted at all.

To align with this assumption, this paper employs a partial order [54] based on the R matrix to highlight the contribution of correctly answered questions, emphasizing the influence of "knowing". For a student u, the partial order $>_u^+$ can be defined as in Equation (5):

$$i >_u^+ j, \text{ if } R_{ui} = 1 \text{ and } R_{uj} = 0 \tag{5}$$

Equation (5) indicates that for a student u, if the response to a question i is 1, then i is considered to assess "knowing" more than other questions. Under the conditions set by the partial order $>_u^+$, the R matrix can be transformed into a set of comparative questions D_u using Equation (6).

$$D_u = \left\{(i,j) \mid i >_u^+ j\right\} \tag{6}$$

D_u can more accurately capture pairwise relationships between two questions (i,j) based on the student u, aiding in obtaining explanations for "knowing". With the partial order prior, this paper sets the knowledge matrix based on the R matrix as $V^R \in R^{M \times K}$, where M represents the number of questions and K represents the dimension of the knowledge concept. Since the partial order is based on the R matrix, the model aims to obtain the decomposed matrix of student proficiency $U \in R^{N \times K}$ and the question knowledge matrix V^R. For all question pairs (i,j) belonging to D_u, Θ represents the model's parameter set, which includes the parameters U and V^R. So, the objective function is divided into two parts. The first part is the model's likelihood function, and the second part contains the prior probability of Θ. For the first part $p(>_u^+ \mid \Theta)$, within the "knowing" level, it is assumed that question pairs (i,j) are independent of other question pairs. Therefore, the likelihood function $p(>_u^+ \mid \Theta)$ can be obtained, as shown in Equation (7):

$$p\left(>_u^+ \mid \Theta\right) = \prod_{(i,j) \in D_u} p\left(i >_u^+ j \mid \mathit{\Theta}\right) \tag{7}$$

In order to obtain the correct ordinal relationship under the condition of Θ, the probability that student u is more relevant to question i than to question j is defined as shown in Equation (8):

$$p\left(i >_u^+ j \mid \Theta\right) = \sigma\left(\overline{x}_{ui}(\theta) - \overline{x}_{uj}(\theta)\right) \tag{8}$$

where σ is the logistic function, designed to simultaneously satisfy the completeness, antisymmetry, transitivity, and optimization calculations for the ordinal $i >_u^+ j$. Because the logistic function itself is a continuous and centrally symmetric function, it satisfies the setting of the partial order. In addition, for the second part $p(\mathit{\Theta})$, following the traditional approach, assuming both U and V^R follow Gaussian distributions, therefore, $\mathit{\Theta}$ follows a Gaussian prior with a zero mean. Combining the first part of the objective function in Equations (7) and (8) with the Gaussian assumption of the second part, the posterior distribution of $\mathit{\Theta}$ under the condition of D_u can be obtained, as shown in Equation (9):

$$p\left(\mathit{\Theta} \mid >_u^+\right) \propto p\left(>_u^+ \mid \mathit{\Theta}\right) p(\mathit{\Theta}) = \prod_{(i,j) \in D_u} p\left(>_u^+ \mid V^P\right) p\left(V^P\right) \tag{9}$$

4.3. "Understanding" Model

"Understanding" means comprehending things without the need for deep insight. In contrast to "knowing", comprehension involves grasping shallow concepts and meanings, rather than just memorizing the knowledge concept. Based on the characteristics of comprehension, this study focuses on questions that assess the same knowledge concept. It is assumed that when the same knowledge concept is assessed, it provides a further measure of the level of understanding of that knowledge concept. In other words, when only questions related to a particular knowledge concept are considered, there will be a certain level of impression on the intrinsic logic and form of that knowledge concept, deepening the level of understanding.

Therefore, we propose the concept of "knowledge groups": an organizational structure formed by questions related to the same knowledge concept. Based on the question–knowledge concept category labels provided by the Q matrix, questions assessing the same knowledge concept are divided into non-overlapping, non-intersecting, and disjoint groups. A hierarchical understanding model is then constructed from bottom to top. Different from the Q matrix, knowledge groups, although originating from the Q matrix, define the basic cognitive computing units, making the flat and sparse Q matrix hierarchical for convenient scientific prediction. The introduction of knowledge groups is not only to capture the common knowledge-related features among questions but also to build a hierarchical structure more effectively.

Next, introduce some related concepts: knowledge groups, sub-groups CV_l^C, parent groups CV_l^C, the comprehension matrix V^C, and the rating matrix R^C.

Knowledge Groups: It is defined that questions belonging to the same knowledge concept form a knowledge group. For example, in Figure 1, 11 questions and 4 knowledge groups $\{1^{(V^C)}2^{(V^C)},\ 3^{(V^C)},\ 4^{(V^C)}\}$. In the hierarchical structure module, the first row of solid circles represents questions, and the second row of solid circles represents knowledge groups. Questions 1, 2, and 3 belong to the knowledge group $1^{(V^C)}$, questions 4 and 5 belong to the knowledge group $2^{(V^C)}$, questions 6, 7, and 8 belong to the knowledge group $3^{(V^C)}$, and questions 9, 10, and 11 belong to the knowledge group $4^{(V^C)}$. According to the task of this paper, the knowledge group has the following characteristics:

- Each element within a knowledge group belongs to one and only one knowledge concept.
- There is at least one element within a group.
- There is no overlap between groups.

Subgroup CV_l^C: Defined as the set of questions within the knowledge group l. For example, questions 1, 2, and 3 belong to $1^{(V^C)}$ and are a subgroup of knowledge concept 1. $CV_1^C = \{1, 2, 3\}$, and so on for other subgroups.

Parent Group PV_l^R: Question l can be mapped to the corresponding knowledge group through PV_l^R. For example, Questions 1, 2, and 3 belong to $1^{(V^C)}$, so the parent group $PV_1^R = 1$, and so on for other parent groups.

The comprehension matrix $V^C \in R^{K \times K}$ represents the matrix for comprehension, indicating the degree of examination of comprehension for each question. K represents the number of knowledge groups, and it is also the dimension of the feature vector. Since the R matrix does not have features related to knowledge groups, it would be challenging for the parameters to learn about comprehension. Therefore, this paper introduces the comprehension hierarchical result matrix R^C, which systematically associates knowledge groups with the result matrix R, further supervising the training of V^C.

In essence, R^C is a matrix that represents students' scores for knowledge groups. It has the dimensions $R^C \in R^{N \times K}$, where N represents the number of students, and K represents the number of knowledge groups. R_{uj}^C represents student u performance in knowledge group j. There are various ways to construct R^C. For example, when the data within each group are uniformly distributed, they can use the median or select the maximum and minimum values as representative values for the knowledge group. Since the data for knowledge groups are typically binary and may contain some missing values, the model uses the group's average as the representative value for the knowledge group. The calculation formula for the knowledge group result matrix is as follows (Equation (10)):

$$R_{uj}^C = \frac{1}{\left|CV_j^C\right|} \sum_{l \in CV_j^C} R_{ul} = \overline{R}_{u,CV_j^C} \tag{10}$$

When establishing the relationship between students, questions, and knowledge groups, the hierarchical nature of the question feature matrix is taken into account. These layers form an accumulating hierarchical framework, where reaching the next level depends on the previous level. Therefore, it is assumed that each layer's question feature matrix is sampled from its sub-feature matrix. For "knowing" and "understanding", $V^C \sim \mathcal{N}\left(V^R, \sigma_V^2 I\right)$. The generation process is as follows:

(1) Considering the hierarchical nature of "knowing" and "understanding", the prior of the comprehension matrix $V^C \sim \mathcal{N}\left(V^R, \sigma_V^2 I\right)$.
(2) The prior of knowledge proficiency U is $\mathcal{N}\left(0, \sigma_U^2 I\right)$.
(3) For each non-missing entry (u, j) in the "understanding" level, scores matrix $R_{uj}^C \sim \mathcal{N}(\langle U_u, V_j^C \rangle, \sigma_R^2 I)$.
(4) δ is an indicator function. If $R_{uj}^C \neq null$, then $\delta_{uj}^{R^C} = 1$; otherwise, $\delta_{uj}^{R^C} = 0$.

Here, σ represents the standard deviation. The posterior distributions for the parameters question knowledge matrix V^R, comprehension matrix V^C, and knowledge proficiency matrix U are given by Equation (11):

$$\begin{aligned}
&P\left(V^R, V^C, U \,\middle|\, >_u^+, R^C, \sigma_{R^C}^2, \sigma_U^2, \sigma_V^2, \right) \\
&\propto P\left(>_u^+ \mid U, V^R\right) P\left(R^C \middle| U, V^C, \sigma_R^2\right) P\left(V^C \middle| \sigma_V^2\right) P\left(U \middle| \sigma_U^2\right) P\left(V^R \middle| \sigma_V^2\right) \\
&= \textstyle\prod_{(i,j)\in D_u} \sigma\left(\overline{x}_{ui} - \overline{x}_{uj}\right) \\
&\quad \prod_{u,j} \mathcal{N}\left(R_{uj}^C \mid \langle U_u, V_j^C \rangle, \sigma_R^2\right)^{\delta_{uj}^{R^C}} \\
&\mathcal{N}\left(V^C \mid V^R, \sigma_V^2 I\right) \mathcal{N}\left(V^R \mid 0, \sigma_V^2 I\right)\ \mathcal{N}\left(U \mid 0, \sigma_U^2 I\right)
\end{aligned} \tag{11}$$

4.4. *"Appilication" Model*

Application refers to the ability to apply learned concepts, rules, and principles. In comparison to understanding, application not only assesses the degree of comprehension and grasp of a particular knowledge concept but also emphasizes the ability to apply that understanding to learn other knowledge concepts. It is a cross-knowledge-point ability.

Based on the characteristics of application, the research focuses on questions that examine similar knowledge concepts. Assuming that similar knowledge concepts are being examined, it is easier to identify the intrinsic connections between knowledge concepts. In other words, when questions address similar knowledge concepts, it is easier to measure the level of conceptual or knowledge transfer and further deepen the understanding of knowledge concepts.

This paper innovatively introduces the concept of higher-order knowledge groups, establishing an organizational structure for higher-order knowledge groups and constructing a cross-knowledge-concept model. Like the understanding level, the knowledge groups examining similar knowledge concepts are divided into non-overlapping, non-intersecting clusters. From these similar knowledge concepts, higher-order knowledge groups are abstracted. This not only succinctly describes the aggregation level of knowledge groups but also captures features related to similar knowledge groups, facilitating the construction of a hierarchical structure. Higher-order knowledge groups, based on knowledge groups, represent a higher level of abstraction. Similar to knowledge groups, there are several related concepts: higher-order knowledge groups, subgroups CV_l^A, parent groups PV_l^C, the application matrix V^A, and the application rating matrix R^A.

With the introduction of higher-order knowledge groups, there arises a correspondence between knowledge groups and higher-order knowledge groups. As shown in the diagram, knowledge groups $\{1^{(V^C)}, 2^{(V^C)}\}$ belong to the higher-order knowledge group $1^{(V^A)}$, and knowledge groups $\{3^{(V^C)}, 4^{(V^C)}\}$ belong to the higher-order knowledge group $2^{(V^A)}$.

Subgroup CV_l^A: CV_l^A is defined as the set of knowledge groups for higher-order knowledge group l. For example, knowledge groups 1 and 2 belong to $1^{(V^A)}$, so $CV_1^A = \{1, 2\}$, and so on for other subgroups.

Parent group PV_l^C: Knowledge group l can be mapped to the corresponding higher-order knowledge group through PV_l^C. For example, knowledge groups 1 and 2 belong to $1^{(V^A)}$, so $PV_1^C = 1$, and so on for other parent groups.

$V^A \in \mathrm{R}^{H \times K}$ represents the application matrix, where H is the number of higher-order knowledge groups. Similar to understanding, V^A is still obtained through supervised training. Therefore, this paper also introduces the application level result matrix R^A to further supervise the training of V^A. The size of R^A is $\mathrm{R}^{N \times H}$, where R_{uj}^A represents the student u performance on higher-order knowledge group j. R^A can be constructed in various ways, and the model still uses the average within the group as the representative value. The calculation formula for the higher-order knowledge group result matrix is given by Equation (12).

$$R_{uj}^A = \frac{1}{\left|CV_j^A\right|} \sum_{l \in CV_j^A} R_{ul}^C = \overline{R}_{u,CV_j^A} \tag{12}$$

One important point to note is that in most application scenarios, information about higher-order knowledge group structures is implicit and not readily accessible. The methods for discovering such structures will be discussed in Chapter Four of this paper. For the purposes of this section, it can be assumed that information about the higher-order knowledge group structure is obtained through the higher-order knowledge group structure discovery algorithm in the next section, and it is not acquired during the model training process.

Similar to Equation (11), the posterior distribution of parameters V^R, V^C, V^A, and U under the conditions of the partial order and the result matrices for each level can be expressed as Equation (13):

$$\begin{aligned}
&P\left(V^R, V^C, V^A, U \mid >_u^+, R^C, R^A, \sigma_R^2, \sigma_R^2, \sigma_U^2, \sigma_V^2\right)\\
&\propto P\left(U, V^R \middle| >_u^+\right) P\left(R^C \middle| U, V^C, \sigma_R^2\right) P\left(R^A \middle| U, V^A, \sigma_R^2\right)\ P\left(U \middle| \sigma_U^2\right) P\left(V^R \middle| \sigma_V^2\right) P\left(V^C \middle| \sigma_V^2\right) P\left(V^A \middle| \sigma_V^2\right)\\
&= \prod\nolimits_{(i,j) \in D} \sigma\left(\overline{x}_{ui} - \overline{x}_{uj}\right)\\
&\prod_{u,j} \mathcal{N}\left(R_{uj}^C \mid \langle U_u, V_j^C \rangle, \sigma_R^2\right)^{\delta_{uj}^{R^C}}\\
&\prod_{u,j} \mathcal{N}\left(R_{uj}^A \mid \langle U_u, V_j^A \rangle, \sigma_R^2\right)^{\delta_{uj}^{R^A}}\\
&\mathcal{N}\left(V^C \mid V^R, \sigma_V^2 I\right) \mathcal{N}\left(V^A \mid V^C, \sigma_V^2 I\right) \mathcal{N}\left(V^R \mid 0, \sigma_V^2 I\right) \mathcal{N}\left(U \mid 0, \sigma_U^2 I\right)
\end{aligned} \tag{13}$$

5. Model Learning and Prediction

In order to better understand the proposed BloomCDM, this section summarizes the probabilistic graphical representation of the model in Figure 2, where shaded and unshaded variables represent observed and latent variables.

Based on the detailed description in the previous section, given the student response matrix R, the comprehension-level knowledge group matrix R^C, the application-level higher-order knowledge group matrix R^A, and the partial order $>_u^+$, the model's goal is to learn the parameters $\phi = \left[U, V^P, V^C, V^A\right]$. Through the BloomCDM model, the posterior distribution of the target parameters ϕ is obtained under the condition of partial order and the result matrix at each level.

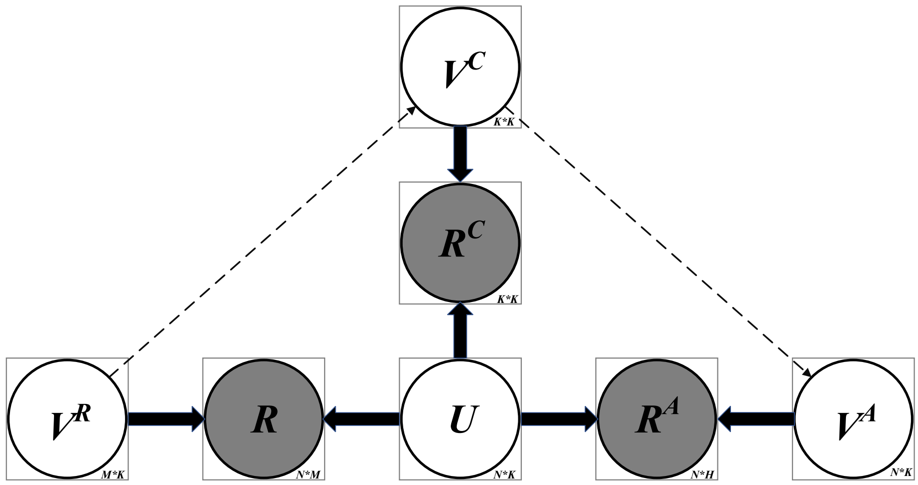


Figure 2. Graphical model of BloomCDM.

Therefore, the logarithmic posterior maximization in Formula (13) is equivalent to minimizing the objective function in Formula (14):

$$\begin{aligned}
&\mathcal{L} = \ln p\left(V^R, V^C, V^A, U \mid >_u^+, R^C, R^A, \sigma_{\mathrm{R}}^2, \sigma_U^2, \sigma_V^2\right) \\
&\propto - \sum_{(i,j)\in D_u} \left(ln\sigma\left(U_u \cdot V_i^R - U_u \cdot V_j^R \right) \right) \\
&- \tfrac{1}{2\sigma_R^2} \sum_{u=1}^{N} \left(R_{uj}^C - \left\langle U_u, V_j^C \right\rangle \right)^2 \\
&- \tfrac{1}{2\sigma_R^2} \sum_{u=1}^{N} \sum_{j=1}^{H} \delta_{uj}^{R^A} \left(R_{uj}^A - \left\langle U_u, V_j^A \right\rangle \right)^2 \\
&- \tfrac{1}{2\sigma_V^2} \sum_{j=1}^{M} \left(V_j^R - V_{PV_j^R}^C \right)^2 \\
&\qquad - \tfrac{1}{2\sigma_V^2} \sum_{j=1}^{K} \left(V_j^C - V_{PV_j^C}^A \right)^2 \\
&- \frac{1}{2\sigma_V^2} \sum_{j=1}^{M} \left(V_j^R \right)^2 - \frac{1}{2\sigma_U^2} \sum_{u=1}^{N} U_u^2
\end{aligned} \tag{14}$$

The three most commonly used methods for solving machine learning parameters are: ordinary least squares, gradient descent, and the Fastest Newton method. Since the objective function is convex, the model adopts the stochastic gradient descent (SGD) method [55]. To achieve more accurate convergence, the model does not directly apply SGD but instead follows the random block coordinate descent to sequentially update U and all V at each level.

$$\frac{\partial \mathcal{L}_R}{U_u} = \frac{1}{1 + \mathrm{e}^{\mathrm{U_u} \cdot V_i^R - \mathrm{U_u} \cdot V_j^R}} \left(V_i^R - V_j^R \right) - \frac{1}{\sigma_U^2} U_u \tag{15}$$

$$\frac{\partial \mathcal{L}_{R^C}}{U_u} = \frac{1}{\sigma_R^2} \sum_{j=1}^{K} \delta_{uj}^{R^C} \left(R_{uj}^C - \left\langle U_u, V_j^C \right\rangle \right) V_j^C - \frac{1}{\sigma_U^2} U_u \tag{16}$$

$$\frac{\partial \mathcal{L}_{R^A}}{U_u} = \frac{1}{\sigma_R^2} \sum_{j=1}^{H} \delta_{uj}^{R^A} \left(R_{uj}^A - \left\langle U_u, V_j^A \right\rangle \right) V_j^A - \frac{1}{\sigma_U^2} U_u \tag{17}$$

$$\frac{\partial \mathcal{L}_R}{V_j^R} = \frac{\mathrm{U_u}}{1 + \mathrm{e}^{\mathrm{U_u} \cdot V_i^R - \mathrm{U_u} \cdot V_j^R}} - \frac{1}{\sigma_V^2} V_j \tag{18}$$

$$\frac{\partial \mathcal{L}_R}{V_j^C} = \frac{1}{\sigma_R^2} \sum_{j=1}^{K} \delta_{ij}^{R^C} \left(R_{ij}^C - \left\langle U_i, V_j^C \right\rangle \right) U_i - \frac{1}{\sigma_V^2} \sum_{j=1}^{\mathrm{K}} \left[\left| CV_j^C \right| \left(V_j^C - \overline{V}_{CV_j^C}^R \right) + \left(V_j^C - V_{PV_j^C}^A \right) \right] \tag{19}$$

$$\frac{\partial \mathcal{L}_R}{V_j^A} = \frac{1}{\sigma_R^2} \sum_{j=1}^{H} \delta_{ij}^{R^A} \left(R_{ij}^A - \left\langle U_i, V_j^A \right\rangle \right) U_i - \frac{1}{\sigma_V^2} \sum_{j=1}^{\mathrm{H}} \left[\left| CV_j^A \right| \left(V_j^A - \overline{V}_{CV_j^A}^C \right) \right] \tag{20}$$

Algorithm 1 outlines the learning process for rating prediction in BloomCDM. It can be observed that the time complexity is linearly related to the order. This study sets the order to be 3, assuming there are non-empty entries in all result matrices. The average time per student is $tr = r/N$. In each iteration, the time complexity for U is $O(N \times tr) = O(K \times r)$, and for V, it is $O(3 \times M \times tr) = O(K \times r)$. Therefore, the total complexity of parameter learning in each iteration is $O(K \times r)$, which is linearly related to the records.

Algorithm 1: The Learning Algorithm of BloomCDM

Input: Matrices R, Subgroup CV^C, CV^A, standard deviations σ_U^2,σ_V^2,σ_R^2, learning rate η, number of iterations T, number of hierarchical levels L = 3.
Output: Student feature matrix U, hierarchical feature matrices V^P, V^C, V^A, predicted orthogonal matrix $\widetilde{R}$.

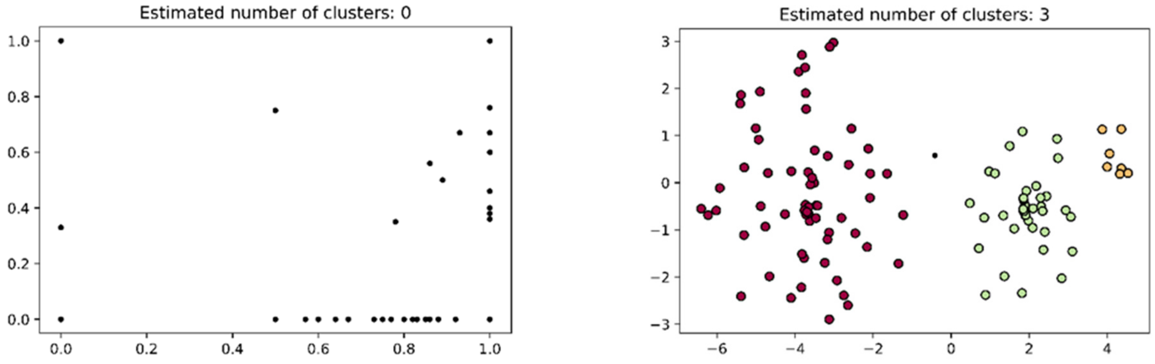
```
for $t = 1, 2, \cdots, T$ do
    for $l = 1, 2, \cdots L$ do
        if $l = 1$ do
            $U \leftarrow U + \eta [\partial \mathcal{L}_R / U_u]_{u=1}^{N}$ via Equation (15)
            $V^R \leftarrow V^R + \eta \left[ \partial \mathcal{L}_R / V_j^R \right]_{j=1}^{M}$ via Equation (18)
        end if
        if $\mathrm{l} = 2$ do
            $\mathrm{U} \leftarrow \mathrm{U} + \eta [\partial \mathcal{L}_{\mathrm{R^C}} / \mathrm{U_u}]_{\mathrm{u=1}}^{\mathrm{N}}$ via Equation (16)
            $\mathrm{V^C} \leftarrow \mathrm{V^C} + \eta \left[ \partial \mathcal{L}_{\mathrm{R^C}} / \mathrm{V_j^C} \right]_{\mathrm{j=1}}^{\mathrm{K}}$ via Equation (19)
        end if
        if $\mathrm{l} = 3$ do
            $\mathrm{U} \leftarrow \mathrm{U} + \eta [\partial \mathcal{L}_{\mathrm{R^A}} / \mathrm{U_u}]_{\mathrm{u=1}}^{\mathrm{N}}$ via Equation (17)
            $\mathrm{V^A} \leftarrow \mathrm{V^A} + \eta \left[ \partial \mathcal{L}_{\mathrm{R^A}} / \mathrm{V_j^A} \right]_{\mathrm{j=1}}^{\mathrm{H}}$ via Equation (20)
        end if
    end for
end for
return $\widetilde{\mathrm{R}} = \left\langle \mathrm{U_u}, \mathrm{V_j^R} \right\rangle, \widetilde{\mathrm{R^c}} = \left\langle \mathrm{U_u}, \mathrm{V_j^C} \right\rangle, \widetilde{\mathrm{R^A}} = \langle \mathrm{U_u}, \mathrm{V_j^A} \rangle$//Output Feature Matrix and Prediction Matrix
```

6. High-Order Knowledge Group Structure Detection

As mentioned earlier, the application level involves the ability to understand a single piece of knowledge deeply and apply it to learn other related knowledge, demonstrating a cross-topic proficiency. Hence, we introduce the concept of "higher-order knowledge groups" to construct a model that spans different knowledge areas. However, in some application scenarios, this hierarchical structure may be implicit, making it challenging to intuitively obtain structural information. To address this, we propose a method for discovering higher-order knowledge group structures, abstracting them from existing knowledge groups, and thereby enhancing the construction of these higher-order structures. This approach provides strong support for calculating proficiency based on Bloom's cognitive theory.

Due to the sparsity of student feedback, it is often challenging to design an appropriate clustering algorithm that effectively groups related knowledge areas together. We consider two intuitive hypotheses. The first hypothesis involves directly applying clustering algorithms within knowledge groups. The second hypothesis, considering the high-dimensional and sparse nature of the data, suggests a two-stage research approach: first reducing the dimensionality of knowledge group features, followed by applying a clustering algorithm.

According to Hypothesis 1, we initially employ the DBSCAN algorithm to discover the high-order knowledge group structures in a clustering scenario. DBSCAN is a clustering algorithm that does not require specifying the number of clusters, making it particularly suitable for discovering high-order knowledge groups. The DBSCAN algorithm's underlying idea is to create a circle with a radius of ε around each data point, classifying all data points into core points, boundary points, and noise points. A data point is considered a core point if it has more than minPts neighbors within this radius. Data points in the vicinity of a core point and with a minimum number of neighbors, minPts, are referred to as directly density-reachable points. If a directly density-reachable point is identified as a core point, then its neighboring points are defined as density-reachable. All points in the same cluster are density-connected. Points with a number of neighbors less than minPts are considered boundary points, as there are no other data points in their vicinity, and they are treated as noise [56]. The clustering effect of directly applying DBSCAN is illustrated in Figure 3a. From the figure, it can be observed that in the direct application case, the DBSCAN algorithm fails to determine the number of clusters and does not uncover the similar characteristics of knowledge groups.



(**a**) Hypothesis 1 (**b**) Hypothesis 2

Figure 3. The comparison of the clustering method.

According to hypothesis two, we first perform dimensionality reduction on the features of knowledge groups, followed by applying DBSCAN for clustering. In the dimensionality reduction phase, the t-distributed stochastic neighbor embedding algorithm (t-SNE) [57] is an important method proposed in 2008 for reducing dimensionality. The primary purpose of this model is to visualize high-dimensional data, and it has found widespread application in various fields. The t-SNE algorithm proceeds in three steps: first, it calculates the pairwise probabilities of being neighbors in the high-dimensional space. Second, it calculates the pairwise probabilities of being neighbors in the low-dimensional space. In the third step, the algorithm attempts to minimize the differences between these conditional probabilities (or similarities) in both high-dimensional and low-dimensional spaces to achieve an optimal representation of data points in the low-dimensional space. The Kullback–Leibler (KL) divergence measures the difference between the distributions of high-dimensional and low-dimensional data. To minimize the KL divergence, t-SNE employs gradient descent to reduce the distance between the two distributions. In summary, in the scenario of knowledge group feature reduction, t-SNE minimizes the differences between two distributions: the high-dimensional feature distribution of knowledge groups and their distribution in the low-dimensional space.

In the clustering phase, DBSCAN is again applied to the low-dimensional data points for cluster discovery. The clustering effect under hypothesis two is illustrated in Figure 3b.

Different colors represent different clusters, it can be clearly observed that different clusters are discernible, resulting in a cluster count of three. The clustering module demonstrates higher accuracy in capturing the relationships between knowledge groups. Algorithm 2 demonstrates the process of discovering high-order knowledge group structures.

Algorithm 2: Higher-Order Knowledge Group Structure Detection

Input: Understanding result matrix $R^C = \{R_1^C, R_2^C, \ldots R_n^C\}$, perplexity $Perp(i)$, learning rate η, number of iterations T, radius ε, minPts, initialize label set as "undefined".
Output: Number of high-order knowledge groups, label set.

```
$DB = t\_SNE(R^C, Perp, T, \eta)$//Using t-SNE for dimensionality reduction, obtain the low-dimensional data points after reduction.
    for every data point $p$ in $DB$ do
        if $label(p) \neq undefined$ then continue//Select an untreated point.
            Neighbour $N \leftarrow$ RangeQuery($T, p, \varepsilon$)//Find points that are density-reachable from point p, and add them to the neighborhood N.
        if $|N| < minPts$ then//If the number of neighbors is less than minPts, then point p is temporarily marked as noise.
            $label(p) \leftarrow$ noise
      c $\leftarrow$ next cluster label
      label(p) $\leftarrow$ c
      neighbour set S $\leftarrow$ N{p}
      for every data point q in S do
        if label(q) = noise then label(q) $\leftarrow$ c
        if label(q) $\neq$ undefined then continue
        N $\leftarrow$ RangeQuery(T, q, $\varepsilon$)
        label(q) $\leftarrow$ c
        if |N| < minPts then continue
        S $\leftarrow$ S $\cup$ N
      end for
    end for
return label, unique(label)
```

7. Experiment

In this section, we first introduce our experimental dataset and settings. Then, we report the experimental results in terms of three aspects: (1) student performance prediction task; (2) knowledge proficiency diagnostic task; (3) visualized knowledge proficiency task.

7.1. Dataset

This experiment focuses on studying students' cognitive performance in the context of solving problems. Therefore, it requires data on students' performance on questions, with the most crucial being the scores they achieve in their attempts. Currently, many studies in the field of cognitive diagnostics rely on private datasets provided by educational institutions and organizations, and openly available datasets are relatively scarce. Hence, in this paper's experiment, two real-world datasets were utilized: the ASSIST dataset and the Hangzhou Dianzi University Online Judge Platform dataset (HDU).

The ASSIST dataset is derived from the publicly available dataset "Assistments 2009–2010 Skill Builder". This public dataset was curated by Professor Heffernan and his team from the online education platform Assistments. It provides detailed records of students' log data for the years 2009–2010, including student IDs, question IDs, attempt results, educational modes, and the number of attempts by students, and it also annotates the corresponding skills for each question, such as "Percent Of", "Absolute Value", and so on.

The HDU dataset originates from the Hangzhou Dianzi University Online Judge Platform (OJ). The OJ platform is an online judging system where users can submit their code for evaluation. It collects user-submitted code, compiles and runs it independently, and verifies the correctness of the students' solutions. The OJ platform is open to all users, allowing them to submit code for assessment once they are logged into the system. In the "Realtime Judge Status" module of HDU's OJ platform, there is a large amount of historical user log data for problem solving. The target fields for data retrieval in this experiment include "Author", "ProID", "Judge Status", and "Submit Time." The data were collected from student logs for the years 2020–2021. The categories of knowledge concepts corresponding to the questions mainly include topics such as Graph Theory, Tree Theory, Depth-First Search, Sorting, and more.

To address the tasks proposed in this paper, several preprocessing steps were implemented in this experiment:

(1) Deduplication: Both datasets have a temporal sequence, meaning that for the same question, a student may have multiple response records. Therefore, the log data represents a sequence. This experiment retained only the first response record, considering the first response as the true reflection of the student's cognition to ensure the uniqueness of the student's response

(2) Dealing with Long-Tail Distribution: Both datasets exhibit a long-tail distribution, with some students having very few log entries, indicating low activity in answering questions. This could potentially affect the diagnostic results. In this experiment, students with fewer than 15 log entries and questions with fewer than 15 log entries were filtered out, ensuring that each student and question have sufficient data for diagnosing the student's cognition.

(3) Standardizing Response States: Since the HDU dataset has four types of response states ("Compilation Error", "Timeout", "Wrong Answer", "Accepted"), this experiment excluded the first two states ("Compilation Error" and "Timeout") and considered only "Wrong Answer" and "Accepted."

The preprocessed data form the basis for the subsequent analysis and experimentation. The steps mentioned above were crucial in ensuring the quality and reliability of the results obtained from the experiment.

7.2. Experiment Setting

7.2.1. BloomCDM Configuration

To evaluate the performance of the BloomCDM model, this experiment divided the dataset into training and testing sets in an 80–20 ratio. It investigated the performance prediction task under different levels of data sparsity. For the hyperparameters in the BloomCDM model, the default settings are as follows:

Learning rate η: 0.001, Standard deviation $\sigma_U^2, \sigma_V^2, \sigma_R^2$: 0.001.

The hyperparameter settings for the high-order knowledge group discovery method are as follows:

Perp(i): 30, Learning rate η: 50, Radius ε: 2, Minpt: 21.

7.2.2. Baseline Methods

To assess the performance of the BloomCDM model and validate its robustness, we compared it against six baseline methods.

(1) IRT: Item Response Theory (IRT) is a classical cognitive diagnostic model in educational statistics. It is represented by Equation (21). It constructs a model for calculating the probability of a student's response considering a one-dimensional latent ability variable θ and item feature latent variable d.

$$p_{ij} = \sigma\big(\theta_i - d_j\big) \tag{21}$$

(2) MIRT: Multidimensional Item Response Theory (MIRT) is the multidimensional version of the IRT model, as shown in Equation (22). It considers the multidimensionality of ability $\theta = (\theta_1, \ldots, \theta_m)^T$ and builds a model for calculating the probability of a

student's response based on a monotonicity assumption and ability independence assumption.

$$p_{ij} = \sigma\left(\left\langle \theta_i, d_j \right\rangle + \delta_j\right) \tag{22}$$

(3) PMF: Probabilistic Matrix Factorization (PMF) is a widely used algorithm in recommendation systems. It employs factorization methods to decompose the response logs into latent feature matrices for students U and items V.

(4) QPMF: QPMF [36] is a variant of PMF that introduces the Q-matrix to enhance the interpretability of PMF. The embedding method utilizes a V-matrix-based partial order to emphasize the contribution of the knowledge concept assessed by items.

(5) BPR: Bayesian Personalized Ranking (BPR) is a classic algorithm in recommendation systems. BPR combines a likelihood function $P(i > uj \mid \Theta)$ constructed based on partial order relationships with a prior probability $P(\Theta)$ to perform Bayesian analysis on student response logs.

(6) BloomCDM-RC: BloomCDM-RC is a simplified version of BloomCDM. It only considers the "Remember" and "Comprehension" levels and does not take into account the Application level.

Specifically, the selected baselines are widely used in traditional cognitive diagnostics (IRT, MIRT), recommendation system models (PMF, BPR, QPMF), and a variant of the proposed model (BloomCDM-RC). In the following experiments, both BloomCDM and the baselines are implemented in Python. All experiments were conducted on a Linux server with 4 2.0 GHz Intel Xeon E5-2620 CPUs and 100 GB of memory. To ensure fairness, all parameters of these baselines are tuned for optimal performance.

7.3. Results

7.3.1. Analysis of Student Performance Prediction Results

Evaluating the performance of cognitive diagnostic models can be challenging, as we do not have access to the true knowledge levels of students. In most studies, diagnostic results are typically obtained by predicting student performance, and the performance of these prediction tasks indirectly serves as an evaluation metric for the model. In addition, in a real educational environment, students' practice feedback is often not simply wrong or right but may also be the student's specific score. A typical example is the scoring method for objective and subjective questions. To this end, we utilized both classification and regression evaluation metrics based on the set hyperparameters. Specifically, we employed two common metrics for regression models, Root Mean Square Error (RMSE) and Mean Absolute Error (MAE), and two common metrics for classification models, Accuracy (ACC) and Area Under the ROC Curve (AUC). Table 2 presents the experimental results for the models in the task of predicting student scores.

Table 2. Comparative experimental results of all models in student performance prediction tasks.

	ASSIST				**HDU**			
Model	**RMSE**	**MAE**	**ACC**	**AUC**	**RMSE**	**MAE**	**ACC**	**AUC**
IRT	0.463	0.398	0.648	0.648	0.533	0.398	0.676	0.625
MIRT	0.450	0.386	0.750	0.678	0.471	0.386	0.736	0.750
PMF	0.460	0.394	0.657	0.657	0.479	0.394	0.724	0.657
QPMF	0.451	0.388	0.674	0.683	0.460	0.397	0.744	0.687
BPR	0.449	0.386	0.678	0.750	0.449	0.366	0.722	0.678
BloomCDM-RC	0.422	0.370	0.785	0.785	0.412	0.370	0.754	0.785
BloomCDM	**0.421**	**0.364**	**0.836**	**0.886**	**0.407**	**0.364**	**0.766**	**0.836**

Taking into consideration the classification and regression experimental results across all datasets, we can conclude the following:

Regardless of whether it is classification or regression, BloomCDM outperforms traditional cognitive diagnostic models like IRT and MIRT and recommendation models like

BPR and PMF. This indicates that the design proposed in this paper is scientifically effective and can enhance the model's predictive performance on student scores.

BloomCDM performs better than BloomCDM_RC, especially in terms of the ACC metric. Compared to BloomCDM_RC, BloomCDM shows an improvement of 0.049 and 0.012 points in ACC at the ASSIT and HDU datasets, suggesting that modeling the application level is indispensable for capturing features of similar knowledge concepts and improving predictive ability.

The traditional cognitive diagnostic model MIRT performs well on the AUC metric and in some cases even outperforms recommendation models like BPR and PMF. This is an intriguing phenomenon. The subsequent focus of research in this area could be on augmenting the MIRT model with the knowledge concept to calculate more accurate knowledge proficiency.

In terms of AUC and ACC metrics, the performance of the ASSIST dataset is superior to that of the HDU dataset. This discrepancy may arise from the fact that the OJ platform in the HDU dataset allows for free practice, potentially introducing more latent factors that influence the student performance data.

7.3.2. Knowledge Proficiency Diagnosis Task

As mentioned earlier in the model framework in Figure 1, U_u represents the proficiency of student u, where each value in the vector represents the level of mastery of different knowledge concepts for the student u. To assess the accuracy of students' diagnostic results, a new metric called Degree of Agreement (DOA) is introduced, as shown in Formula (23). Intuitively, if student a has a better mastery of knowledge concept k compared to student b, then student a is more likely to answer questions related to k correctly. Therefore, in this experiment, the Consistency (DOA) [58] metric is used to evaluate the ordering of students' proficiency in knowledge.

$$OA(k) = \frac{1}{Z}\sum_{a=1}^{N}\sum_{b=1}^{N}\delta(F_{ak}^{s}, F_{bk}^{s})\sum_{j=1}^{M} I_{jk}\frac{J(j,a,b) \wedge \delta\left(r_{aj}, r_{bj}\right)}{J(j,a,b)} \tag{23}$$

$Z = \sum_{a=1}^{N}\sum_{b=1}^{N}\delta\left(F_{ak}^{s}, F_{bk}^{s}\right)$, and F_{ak}^{s} represents the proficiency of student a in knowledge concept k. δ, I, and J are indicator functions. If $x > y$, $\delta(x,y) = 1$; otherwise, $\delta(x,y) = 0$. If question j assesses knowledge concept k, $I_{jk} = 1$; otherwise, $I_{jk} = 0$. If both student a and student b attempted question j, then $J(j,\ a,b) = 1$; otherwise $J(j,\ a,b) = 0$. In other words, after considering the pairwise relationships between students' questions and knowledge concepts, $DOA(k)$ evaluates the ordering of proficiency in knowledge k. for all students. By summing and averaging $DOA(k)$. for all knowledge concepts, we obtain DOA, which assesses the quality of the diagnosis results (i.e., the proficiency diagnosis degree of the model).

In comparison with classical models, we only consider MIRT, PMF, and QPMF, as these models take into account multi-valued student feature vectors, with QPMF additionally incorporating relationships between knowledge concepts and questions. Due to the characteristics of the knowledge proficiency diagnosis task, this experiment employs the DOA metric to compare the diagnostic accuracy across multiple datasets. Figure 4 presents the experimental results, from which several conclusions can be drawn based on the experimental results table: First, both BloomCDM and BloomCDM-RC exhibit significantly higher DOA values compared to the baseline models. This demonstrates that they yield reasonable knowledge proficiency diagnoses and, considering cognitive levels, aids in calibrating the prediction of knowledge proficiency. Second, BloomCDM outperforms BloomCDM-RC in terms of precision, indicating a certain improvement in the diagnostic effect. Additionally, QPMF performs better than PMF in terms of diagnosis, highlighting the importance of Q-matrix information in obtaining interpretable diagnostic results (knowledge proficiency vectors).

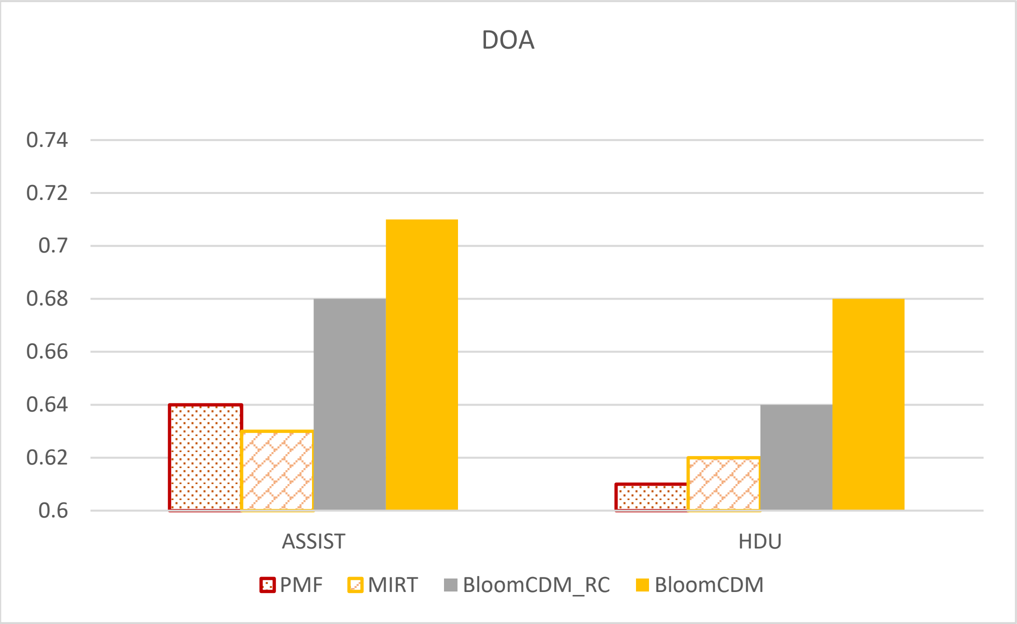
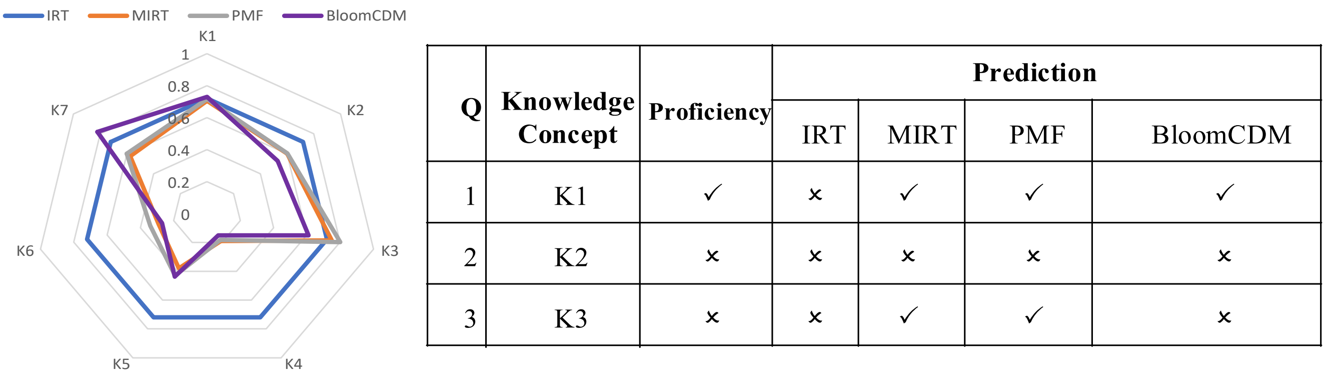


Figure 4. Knowledge proficiency diagnosis performance for all knowledge concepts.

7.3.3. Visualizing Knowledge Proficiency

An example of a cognitive diagnosis of student knowledge proficiency is provided. As shown in Figure 5, the radar chart displays the knowledge proficiency diagnosis results of a student in four models: IRT, MIRT, PMF, and BloomCDM. Since IRT only diagnoses students' unidimensional ability, the ability has the same value for all questions, resulting in a regular polygon in the diagnostic result of IRT in Figure 5. Therefore, compared to IRT, PMF and MIRT can provide more accurate knowledge concept diagnosis results because they diagnose students' multidimensional knowledge proficiency. However, IRT obtained incorrect results on the first question, and MIRT and PMF had similar diagnostic results but made a prediction error on the third question. Only BloomCDM made correct predictions for all three questions.



Q	Knowledge Concept	Proficiency	Prediction			
			IRT	MIRT	PMF	BloomCDM
1	K1	✓	×	✓	✓	✓
2	K2	×	×	×	×	×
3	K3	×	×	✓	✓	×

Figure 5. Visualization of a student's proficiency on knowledge concepts and the parameters of three questions.

8. Conclusions and Future Work

In response to the shortcomings of previous models for assessing proficiency, which often overlooked the hierarchical nature of cognition in educational psychology, this paper proposes a proficiency assessment model named BloomCDM based on Bloom's cognitive taxonomy. The model takes as input students' responses to test items and a matrix of item–knowledge associations (Q-matrix) annotated by experts in the educational field.

The first step of this model involves making theoretical assumptions regarding the hierarchy and inter-level relationships, abstracting a mathematical model, and projecting item feature vectors into the three cognitive spaces of "knowledge", "comprehension", and "application."

(1) "Knowing" Modeling: Based on the theoretical definition of knowledge, we make assumptions and use the $>_u^+$ partial order to learn the knowledge features of items from student response data.

(2) "Understanding" Modeling: Following the theoretical definition of comprehension, this level focuses on items assessing the same knowledge concept. When similar knowledge concepts are assessed, it is assumed that a deeper understanding of the knowledge concept can be measured. The model constructs a comprehension calculation model specific to the knowledge concept in the form of knowledge groups. This captures features related to the same knowledge among items and learns the comprehension features of items.

(3) "Application" Modeling: Based on the theoretical definition of application, this level focuses on items assessing similar knowledge concepts. It is assumed that when similar knowledge concepts are assessed, it is easier to discover the inherent connections between knowledge concepts. The model constructs a cross-knowledge-concept model in the form of high-order knowledge groups, learning the application features of items.

(4) To address the challenge of obtaining hierarchical structure information from sparse data, a high-order knowledge group discovery method is designed in this specific application scenario. It can discover high-order knowledge group structures and mine structural information, providing robust support for proficiency assessment based on Bloom's cognitive levels.

The extensive experiments demonstrate that the model accurately diagnoses students' proficiency, affirming the meaningfulness of considering Bloom's cognitive levels in proficiency assessment. The design presented in this paper is scientifically effective.

In future research, we will explore the following aspects:

(1) While the effectiveness of BloomCDM designed using probabilistic graphical models is evident, neural networks have demonstrated a strong performance in handling nonlinear problems and feature embeddings, achieving accuracies that probabilistic models may not easily attain. It would be interesting to investigate whether interpretable deep learning models can be constructed. Such models could potentially provide better predictions of student performance, building upon the knowledge proficiency obtained.

(2) It is intriguing that the traditional cognitive diagnostic model, MIRT, performs well in terms of the AUC metric, even outperforming recommendation models like BPR and PMF. This phenomenon warrants further investigation. We propose that a promising direction for future models could involve augmenting the MIRT model with knowledge concepts to compute more precise estimates of knowledge proficiency.

Author Contributions: Methodology, Y.L.; Writing—original draft, B.W., Y.L. and H.M.; Writing—review and editing, Y.G., H.L. and W.W.; Supervision, M.Y., T.Z. and G.Y.; Funding acquisition, T.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (62137001, 62272093) and, in part, by the Fundamental Research Funds for the Central Universities under Grant N2216005.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, T.; Chen, L.; Jensen, C.S.; Pedersen, T.B.; Gao, Y.; Hu, J. Evolutionary Clustering of Moving Objects. In Proceedings of the 2022 IEEE 38th International Conference on Data Engineering (ICDE), Kuala Lumpur, Malaysia, 9–12 May 2022; pp. 2399–2411.
2. Li, T.; Huang, R.; Chen, L.; Jensen, C.S.; Pedersen, T.B. Compression of Uncertain Trajectories in Road Networks. *Proc. VLDB Endow.* **2020**, *13*, 1050–1063. [CrossRef]
3. Li, T.; Chen, L.; Jensen, C.S.; Pedersen, T.B. TRACE: Real-time Compression of Streaming Trajectories in Road Networks. *Proc. VLDB Endow.* **2021**, *14*, 1175–1187. [CrossRef]
4. Li, Y.; Zhang, H.; Liang, X.; Huang, B. Event-Triggered Based Distributed Cooperative Energy Management for Multienergy Systems. *IEEE Trans. Ind. Inform.* **2019**, *15*, 2008–2022. [CrossRef]
5. Li, Y.; Gao, D.W.; Gao, W.; Zhang, H.; Zhou, J. Double-Mode Energy Management for Multi-Energy System via Distributed Dynamic Event-Triggered Newton-Raphson Algorithm. *IEEE Trans. Smart Grid* **2020**, *11*, 5339–5356. [CrossRef]
6. Cui, Z.; Sun, X.; Pan, L.; Liu, S.; Xu, G. Event-based incremental recommendation via factors mixed Hawkes process. *Inf. Sci.* **2023**, *639*, 119007. [CrossRef]
7. Wei, W.; Fan, X.; Li, J.; Cao, L. Model the complex dependence structures of financial variables by using canonical vine. In Proceedings of the 21st ACM International Conference on Information and Knowledge Management, Maui, HI, USA, 29 October–2 November 2012; pp. 1382–1391.
8. Fan, X.; Xu, R.Y.D.; Cao, L.; Song, Y. Learning Nonparametric Relational Models by Conjugately Incorporating Node Information in a Network. *IEEE Trans. Cybern.* **2017**, *47*, 589–599. [CrossRef] [PubMed]
9. Fan, X.; Li, B.; Luo, L.; Sisson, S.A. Bayesian Nonparametric Space Partitions: A Survey. In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, Virtual Event, 19–26 August 2021; pp. 4408–4415.
10. Fan, X.; Li, B.; Sisson, S.A.; Li, C.; Chen, L. Scalable Deep Generative Relational Models with High-Order Node Dependence. *arXiv* **2019**, arXiv:1911.01535.
11. Guo, J.; Du, L.; Liu, H.; Zhou, M.; He, X.; Han, S. GPT4Graph: Can Large Language Models Understand Graph Structured Data? An Empirical Evaluation and Benchmarking. *arXiv* **2023**, arXiv:2305.15066.
12. Du, L.; Chen, X.; Gao, F.; Fu, Q.; Xie, K.; Han, S.; Zhang, D. Understanding and Improvement of Adversarial Training for Network Embedding from an Optimization Perspective. In Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, Virtual Event, 21–25 February 2022; pp. 230–240.
13. Bi, W.; Du, L.; Fu, Q.; Wang, Y.; Han, S.; Zhang, D. MM-GNN: Mix-Moment Graph Neural Network towards Modeling Neighborhood Feature Distribution. In Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, Singapore, 27 February–3 March 2023; pp. 132–140.
14. Cao, L.; Chen, H.; Fan, X.; Gama, J.; Ong, Y.-S.; Kumar, V. Bayesian Federated Learning: A Survey. *arXiv* **2023**, arXiv:2304.13267.
15. Chen, H.; Liu, H.; Cao, L.; Zhang, T. Bayesian Personalized Federated Learning with Shared and Personalized Uncertainty Representations. *arXiv* **2023**, arXiv:2309.15499.
16. Shengquan, Y.; Xiaofeng, W. The Transformation Research of Education Supply in the Era of "Internet +". *Open Educ. Res.* **2017**, *1*, 29–36.
17. Liu, H.; Zhang, T.; Li, F.; Gu, Y.; Yu, G. Tracking Knowledge Structures and Proficiencies of Students with Learning Transfer. *IEEE Access* **2021**, *9*, 55413–55421. [CrossRef]
18. Gao, W.; Wang, H.; Liu, Q.; Wang, F.; Lin, X.; Yue, L.; Zhang, Z.; Lv, R.; Wang, S. Leveraging Transferable Knowledge Concept Graph Embedding for Cold-Start Cognitive Diagnosis. In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, Taipei Taiwan, 23–27 July 2023; pp. 983–992.
19. Zhou, Q.; Mou, C.; Yang, D. Research Progress on Educational Data Mining: A Survey. *J. Softw.* **2015**, *11*, 3026–3042.
20. de La Torre, J. DINA model and parameter estimation: A didactic. *J. Educ. Behav. Stat.* **2009**, *1*, 115–130. [CrossRef]
21. Cen, H.; Koedinger, K.; Junker, B. Learning Factors Analysis—A General Method for Cognitive Model Evaluation and Improvement. *Lect. Notes Comput. Sci.* **2006**, *4053*, 164–175.
22. Jiang, P.; Wang, X.; Sun, B. Preference Cognitive Diagnosis for Predicting Examinee Performance. In Proceedings of the 2020 IEEE 2nd International Conference on Computer Science and Educational Informatization (CSEI), Xinxiang, China, 12–14 June 2020; pp. 63–69.
23. Liu, H.-y.; Zhang, T.-c.; Wu, P.-w.; Yu, G. A review of knowledge tracking. *J. East China Norm. Univ. (Nat. Sci.)* **2019**, *05*, 1–15.
24. Cheng, S.; Liu, Q.; Chen, E.; Huang, Z.; Huang, Z.; Chen, Y.; Ma, H.; Hu, G. DIRT: Deep Learning Enhanced Item Response Theory for Cognitive Diagnosis. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, China, 3–7 November 2019; pp. 2397–2400.
25. Guo, X.; Huang, Z.; Gao, J.; Shang, M.; Shu, M.; Sun, J. Enhancing Knowledge Tracing via Adversarial Training. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event, 20–24 October 2021; pp. 367–375.
26. Wei, L.; Li, B.; Li, Y.; Zhu, Y. Time Interval Aware Self-Attention approach for Knowledge Tracing. *Comput. Electr. Eng.* **2022**, *102*, 108179. [CrossRef]
27. Xu, L.; Wang, G.; Guo, L.; Wang, X. Long- and Short-term Attention Network for Knowledge Tracing. In Proceedings of the 2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy, 18–23 July 2022; pp. 1–9.

28. Ma, Y.; Han, P.; Qiao, H.; Cui, C.; Yin, Y.; Yu, Y. SPAKT: A Self-Supervised Pre-TrAining Method for Knowledge Tracing. *IEEE Access* **2022**, *10*, 72145–72154. [CrossRef]
29. Sun, X.; Zhao, X.; Li, B.; Ma, Y.; Sutcliffe, R.; Feng, J. Dynamic Key-Value Memory Networks with Rich Features for Knowledge Tracing. *IEEE Trans. Cybern.* **2022**, *8*, 8239–8245. [CrossRef]
30. Zhang, N.; Li, L. Knowledge Tracing with Exercise-Enhanced Key-Value Memory Networks. In Proceedings of the Knowledge Science, Engineering and Management: 14th International Conference, KSEM 2021, Tokyo, Japan, 14–16 August 2021; pp. 566–577.
31. Mao, S.; Zhan, J.; Li, J.; Jiang, Y. Knowledge Structure-Aware Graph-Attention Networks for Knowledge Tracing. In Proceedings of the International Conference on Knowledge Science, Engineering and Management, Singapore, 6–8 August 2022; Volume 1, pp. 309–321.
32. Gan, W.; Sun, Y.; Sun, Y. Knowledge structure enhanced graph representation learning model for attentive knowledge tracing. *Int. J. Intell. Syst.* **2022**, *3*, 2012–2045. [CrossRef]
33. Wu, Z.; Huang, L.; Huang, Q.; Huang, C.; Tang, Y. SGKT: Session graph-based knowledge tracing for student performance prediction. *Expert Syst. Appl.* **2022**, *206*, 117681. [CrossRef]
34. Liu, H.; Zhang, T.; Li, F.; Yu, M.; Yu, G. A Probabilistic Generative Model for Tracking Multi-Knowledge Concept Mastery Probability. CoRR abs. *arXiv* **2023**, arXiv:2302.08673.
35. Yu, M.; Li, F.; Liu, H.; Zhang, T.; Yu, G. ContextKT: A Context-Based Method for Knowledge Tracing. *Appl. Sci.* **2022**, *12*, 8822. [CrossRef]
36. Chen, Y.; Liu, Q.; Huang, Z.; Wu, L.; Chen, E.; Wu, R.; Su, Y.; Hu, G. Tracking knowledge proficiency of students with educational priors. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, Singapore, 6–10 November 2017; pp. 989–998.
37. Piech, C.; Spencer, J.; Huang, J.; Ganguli, S.; Sahami, M.; Guibas, L.; Sohl-Dickstein, J. Deep knowledge tracing. *arXiv* **2015**, arXiv:1506.05908.
38. Bloom, B.S.; Krathwohl, D.R. *Taxonomy of Educational Objectives*; Cognitive Domain; Longmans, Green: London, UK, 1956; Volume 1, p. 20.
39. Anderson, L.W.; Krathwohl, D.R. *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives: Complete Edition*; Addison Wesley Longman, Inc.: Reading, MA, USA, 2001.
40. Harvey, R.J.; Hammer, A.L. Item response theory. *Couns. Psychol.* **1999**, *3*, 353–383. [CrossRef]
41. Ackerman, T.A. *Multidimensional Item Response Theory Models*; Wiley StatsRef: Statistics Reference Online; Wiley: Hoboken, NJ, USA, 2014.
42. Sheng, Y.; Wikle, C.K. Bayesian multidimensional IRT models with a hierarchical structure. *Educ. Psychol. Meas.* **2008**, *3*, 413–430. [CrossRef]
43. Rost, J.R. An integration of two approaches to item analysis. *Appl. Psychol. Meas.* **1990**, *3*, 271–282. [CrossRef]
44. Ma, W.; de la Torre, J. A sequential cognitive diagnosis model for polytomous responses. *Br. J. Math. Stat. Psychol.* **2016**, *3*, 253–275. [CrossRef]
45. Chen, J.; de la Torre, J. A general cognitive diagnosis model for expert-defined polytomous attributes. *Appl. Psychol. Meas.* **2013**, *6*, 419–437. [CrossRef]
46. Xiong, L.; Chen, X.; Huang, T.K.; Schneider, J.; Carbonell, J.G. Temporal collaborative filtering with bayesian probabilistic tensor factorization. In Proceedings of the 2010 SIAM International Conference on Data Mining, Columbus, OH, USA, 29 April–1 May 2010; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2010; pp. 211–222.
47. Nedungadi, P.; Smruthy, T.K. Personalized multi-relational matrix factorization model for predicting student performance. In *Intelligent Systems Technologies and Applications*; Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 163–172.
48. Huang, Z.; Liu, Q.; Chen, Y. Learning or forgetting? A dynamic approach for tracking the knowledge proficiency of students. *ACM Trans. Inf. Syst.* **2020**, *2*, 1–33. [CrossRef]
49. Wang, F.; Liu, Q.; Chen, E.; Huang, Z.; Chen, Y.; Yin, Y.; Huang, Z.; Wang, S. Neural cognitive diagnosis for intelligent education systems. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 4, pp. 6153–6161.
50. Mnih, A.; Salakhutdinov, R.R. Probabilistic matrix factorization. *Adv. Neural Inf. Process. Syst.* **2007**, *20*, 1257–1264.
51. Moon, T.K. The expectation-maximization algorithm. *IEEE Signal Process. Mag.* **1996**, *6*, 47–60. [CrossRef]
52. Carlin, B.P.; Chib, S. Bayesian model choice via Markov chain Monte Carlo methods. *J. R. Stat. Soc.* **1995**, *3*, 473–484. [CrossRef]
53. White, H. Maximum likelihood estimation of misspecified models. *Econom. J. Econom. Soc.* **1982**, 1–25. [CrossRef]
54. Rendle, S.; Freudenthaler, C.; Gantner, Z.; Schmidt-Thieme, L. BPR: Bayesian personalized ranking from implicit feedback. *arXiv* **2012**, arXiv:1205.2618.
55. Bottou, L. Stochastic gradient descent tricks. In *Neural Networks: Tricks of the Trade*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 421–436.
56. Ester, M.; Kriegel, H.P.; Sander, J.; Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, Portland, OR, USA, 2–4 August 1996; Volume 34, pp. 226–231.

57. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *11*, 2579–2605.
58. Huang, Z.; Liu, Q.; Chen, E.; Zhao, H.; Gao, M.; Wei, S.; Su, Y.; Hu, G. Question Difficulty Prediction for READING Problems in Standard Tests. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 1.

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