AI-assisted knowledge assessment techniques for adaptive learning environments

Sein Minn

Computers and Education: Artificial Intelligence 3 (2022) 100050

Contents lists available at ScienceDirect

Computers and Education: Artificial Intelligence

journal homepage: www.sciencedirect.com/journal/computers-and-education-artificial-intelligence

AI-assisted knowledge assessment techniques for adaptive learning environments

Sein Minn [a,b,*]

[a] Ecole Polytechnique, Institut Polytechnique de Paris, France
[b] Inria, France

ARTICLE INFO

Keywords:
Adaptive learning environments
Intelligent tutoring systems
Educational data mining
Formative assessment
Summative assessment
Knowledge tracing
Cognitive diagnosis modelling

ABSTRACT

The growth of online learning, enabled by the availability on the Internet of different forms of didactic materials such as MOOCs and Intelligent Tutoring Systems (ITS), in turn, increases the relevance of personalized instructions for students in an adaptive learning environment. There are increasing interests as well as many challenges in the application of Artificial Intelligence (AI) techniques in educational settings to provide adaptive learning content to learners. Knowledge assessment is necessary for providing an adaptive learning environment. A student model serves as a fundamental building block of knowledge assessment in an adaptive learning environment. This paper intends to review the development of dominant families of student models with psychometric theory in early educational research, recent adaptations, and advances with machine learning and deep learning techniques. Our review covers not only the important families of student models but also why they were invented from both theoretical and practical viewpoints with AI and educational perspectives. We believe that the discussion covered in this review will be a valuable reference of introductory insights to AI for educational researchers, as well as an endeavor of introducing basic psychometric perspectives to AI experts for knowledge assessment in the field of learning science. Finally, we provide recent challenges and some potential directions for developing efficient knowledge assessment techniques in future adaptive learning ecosystems.

1. Introduction

Artificial Intelligence (AI) assisted knowledge assessment methods have been emerged from research laboratories into practical usage in real-world classrooms for providing adaptive learning environments (Baker, 2016; Romero & Ventura, 2020). Besides, some of them are already deployed in online educational settings for providing time and cost-saving quality education for students worldwide (e.g. MOOCs and other online education platforms) (Fauvel et al., 2018; Yu et al., 2017, 2020). Learning with a computer plays an essential role today and will become inevitable in the future of education. The year 2020 has demonstrated the essentiality of online learning systems which have achieved widespread usage due to the threat of COVID-19 (Adedoyin & Soykan, 2020; Khlaif, Salha, & Kouraichi, 2021). However, it became a challenge to provide each student with an adaptive learning experience in cost effective way as well as an opportunity to provide educational equity for every student around the world.

Adaptive learning instructions are required to improve individual learning gains and enhance the student experience in a personalized learning environment. It has been proved that adaptive learning is much more efficient than a traditional learning environment like classroom learning (Desmarais & d Baker, 2012; Romero & Ventura, 2020). An adaptive learning environment needs to address the huge challenge of large-scale personalization for the real world human learning process. Tutoring systems in various forms are equipped with adaptive learning instructions and, some successful systems are utilized by tens or hundreds of thousands of students a year with growing numbers (Baker, 2016; Hwang & Chang, 2011; Hwang, Chu, Yin, & Lin, 2008).

A successful adaptive learning environment requires two types of adaptations:

- The ability to provide highly specific, immediate, and effective feedback during problem-solving.
- The ability to structure adaptive learning content according to the individual skill proficiency of each student.

In the quest to better support the development of adaptive learning environments, learning science has captured the attention of computer

* Ecole Polytechnique, Institut Polytechnique de Paris, France.
E-mail address: seinminn@lix.polytechnique.fr.

https://doi.org/10.1016/j.caeai.2022.100050
Received 27 July 2021; Received in revised form 18 January 2022; Accepted 21 January 2022
Available online 7 February 2022

scientists for more than a decade. Researchers from fields of education and computer science have been trying to optimize the complex and time-consuming design of adaptive learning environments by using advanced data mining, machine learning, and deep learning techniques. They aim to deliver means for Intelligent tutoring systems (ITS) to assist the students more effectively. Several efforts have been witnessed to develop ITS for learning mathematics (Koedinger & Anderson, 1998; Razzaq et al., 2005), programming (Anderson & Reiser, 1985; Figueiredo & García-Peñalvo, 2020; Mitrovic, 2003; Sykes & Franek, 2003; Weragama & Reye, 2013), languages (Evens et al., 1997; Ferreira & Atkinson, 2008; Slavuj, Kovačić, & Jugo, 2015; Swartz & Yazdani, 2012) and so on. Guiding students efficiently and effectively in their learning process is a recurring topic in 21 century educational research.

To guide the students efficiently and effectively in ITS, it is essential to have an efficient method to assess students' skills (knowledge states) empirically. Whether students are trained by a teacher or a tutoring system, knowledge assessment is necessary to measure their learning gains for how well they have learned and evaluate the efficiency of teaching content and policies of the teacher or system for the future improvement (Wang, 2007). The assessed information will be used to provide immediate feedback (Shute, 2008) and recommend personalized learning materials (Hsu, Hwang, & Chang, 2010; Fu & Zhang, 2013). Various studies have shown that providing knowledge assessments during student learning improves their motivation and learning effectiveness (Bennett, 2011; Gardner, Sheridan, & White, 2002; Henly, 2003; Moss & Brookhart, 2019; Shute, 2008).

Nowadays, Tutoring systems continuously record a massive amount of data about student interactions, which can be used to assess their knowledge gains and learning preferences for enhancing learning experiences. However, human learning is grounded in complexity. It is inherently difficult to measure how much a student knows about a particular knowledge or skill and to assist them as needed on the spot for solving problems (Tsai, Tsai, & Lin, 2015). To challenge this issue, competitions on the application of AI in knowledge assessment have been performed. For example, completion can be found at Riiid AIEd Challenge 2020 [1] in Kaggle with $100,000 for prizes.

This paper aims to address one of the most fundamental questions in the area of cognitive diagnostic assessments: how consistent and accurate are the cognitive diagnosis and performance prediction of each dominant student model? We review historical and recent developments on the topic of knowledge assessment, which bring together the fields of cognitive modeling, psychometrics, and learning science. We pay attention to psychometrics and AI techniques that have given rise to the powerful knowledge assessment in intelligent learning environments. This paper is organized as follows: we discuss domain modeling with psychometric perspectives in section 2. Then in section 3, we present the two types of data (static and dynamic data) commonly used in the context of knowledge assessment. Later, we cover dominant families of psychometric and AI-assisted student models for knowledge assessment and student performance prediction and discussion in section 4. Then, we conclude with issues and challenges of recent student modeling and potential directions for the future in section 5.

2. Domain modeling

A domain model serves as a fundamental building block of ITS. In the psychometric modeling framework, it is necessary to perform domain modeling (extracting the Knowledge Components (KCs) or skills or latent attributes behind tasks) and use extracted KCs of tasks/items/problems/questions to perform knowledge assessment in the student model.

[1] https://www.kaggle.com/c/riiid-test-answer-prediction.

2.1. Knowledge Components representation and granularity

KCs are latent factors that determine the outcome of a student being tested on an item. KCs are latent to the extent that they are never observed. Only the outcome of the individual performing tasks is observed. Domain model with fine-grain Knowledge Components (KCs) is essential in student modeling. For example, when a problem such as

$$1 + 2 \times 3.5 = ?$$

is given to a student, we can consider answering this problem correctly requires mastery of the KCs or skills:

- *integer addition*
- *integer multiplication*
- *decimal notation*
- *decimal multiplication*
- *decimal addition*

This would be considered a fine-grain representation. In a coarser definition, we could consider a single skill involved in this task:

- *arithmetics*

and state that to successfully solve this problem, the students need to have a "good" level of mastery of arithmetics.

2.2. Knowledge Components, skills, items

For our purpose, KCs, or skills, can represent factual knowledge, problem-solving abilities, recognition of patterns and situations, etc. In general, they are the factors associated with an individual that determines the outcome of that individual over a task or item. One, or many, KCs can be associated with a single task.

The probability of getting a correct answer depends mainly on the mastery level of the skills behind a problem. Additionally, she may also require a special skill for the proper integration of all the skills together. In the Knowledge-Learning-Instruction (KLI) framework (Koedinger & Anderson, 1998), that kind of special skill is defined as an "integrative knowledge component" that integrates with all other KCs to produce a correct response.

2.2.1. Q-matrix

In basic psychometrics, it generally requires R: observed response matrix, A: student profile matrix, and Q: Q-matrix for building a student model. For example, a conjunctive model (discussed more details in the section 4.1) represents with following boolean matrix product:

$$R = A \odot Q$$

The response matrix R represents the students' outcome data, where success is assigned 1 and failure 0 in this matrix. For example, the following matrix is the outcome data of 4 students (respondents) and 5 questions (items).

$$R = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

The Q-matrix (Tatsuoka, 1983) Q, represents the item to skills mapping as shown below for 5 items and 3 latent skills:

$$Q = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

This is a conjunctive matrix and therefore the first item in this

example requires skills 1 and 2.

Finally, the student profile matrix, A, which is unknown in a real scenario, represents what is often termed the cognitive diagnostic. For the 4 students, it might look like:

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

For example, the first student only possesses skill 2 (mastery) in this student profile matrix A.

In some models, values of both A and R could be a real value between the interval [0, 1]. If a standard dot product was used, such real values could be found in A and the product $R = A \cdot Q$ might even yield values beyond the range [0, 1]. In such cases, rounding to binary values {0, 1} is often done. Note that domain modeling is usually done in static data (the type of data mentioned in the section 3.1).

Prediction of whether a student may or may not answer a problem correctly is estimated based on his/her mastery of skills in profile matrix A and required skills of problems in Q-matrix Q. The probability of getting a correct answer depends mainly on the mastery of the skills behind a problem. However, she may also require a special skill for the proper integration of all the skills together. In the Knowledge-Learning-Instruction (KLI) framework (Koedinger, Corbett, & Perfetti, 2012), that kind of special skill is defined as an "integrative knowledge component" that integrates with all other KCs to produce a correct response.

A mapping of KCs to problems (Q-matrix) is essential to the psychometric modeling framework and it has traditionally been done by human experts. Experts define the mapping of items to skills based on their own judgment. This is error-prone considering that skills may not be perfectly aligned with some items, may be ill-defined, or may simply involve counter-intuitive skills that are not defined in the skill set. Wrongly defined mapping will result in incorrect cognitive diagnostic. So, some researchers use latent factor analysis techniques and revert to using data-driven methods to automate this mapping and to help define, or refine the definition of KCs (Chiu, 2013; De, 2008; De, 2009; Desmarais & Naceur, 2013; Minn, Desmarais, & Fu, 2016; Templin & Henson, 2006; Templin Hensonet al., 2010).

3. Static and dynamic data

The general research framework around knowledge assessment is to analyze student performance data or item outcomes. The outcome is generally represented as a two-valued variable, success or failure, but it may also be continuous or nominal (fair/good/excellent).

Because we never observed Knowledge Components (KCs) or skills directly, a common way to evaluate how well a student model can assess the student's knowledge state and predict the outcomes (success or failure) of next problems within the learning system. Most of the data is collected in binary format. This data comes in two forms: static and dynamic.

3.1. Static data

The first form is *static* data as described in Fig. 1. Typical examples are standardized test results over a specific course content with the assumption that "No Learning" occurs during the testing, and therefore the student's knowledge state does not change. This type of data is also known as cross-sectional data. It comes in the form of a binary matrix, called *Response matrix*. Modeling with this type of data is often used in the context of *Summative Assessment*. The goal of summative assessment is to evaluate student learning at the end of an instructional unit by comparing it against some standard or benchmark. Information from summative assessments can be used formatively to guide their activities in subsequent courses. This type of data typically has only a few or no missing responses from respondents (students).

3.2. Dynamic data

The second kind of data is *dynamic* as described in Fig. 2, "No Learning" assumption would be unrealistic for tutoring systems, where student learning occurred while they use the system. In such cases, students sometimes try the same type of question several times, possibly for long time periods. The system records the full behavior of students throughout the whole learning phase. Dynamic data necessarily involves the notion of a sequence and is generally time-stamped in Fig. 2. This type of data is also known as longitudinal data. Modeling with this type of data is to perform *Knowledge Tracing* and is often used in the context of *Formative Assessment*. This type of data usually has a lot of missing values according to the nature of tutoring systems.

Intelligent tutoring systems are the environments that can benefit from a domain and student models. And as mentioned, they are typically used for periods of a few hours, up to many weeks or months in a row, and "Learning" is a strong factor in this situation.

3.3. Example of datasets

Here, we mention a few examples of some well-known publicly available datasets for static data and dynamic data in Table 1. Most of the datasets for static data can be found in the R package CDM (George, Robitzsch, Kiefer, Groß, & Ünlü, 2016). Public datasets for dynamic data are available from some tutoring platforms in which students interact with a computer-based learning system in educational settings. For example: ASSISTments [2]: an online tutoring system that was first created in 2004 which engages middle and high school students with scaffolded hints in their math problem. It consists of hundreds of items generated from a number of different templates, all pertaining to the same skill or skill grouping. Students are marked as having completed the problem set when they answer three items correctly in a row without asking for help. If students working on ASSISTments answer a problem correctly, they are given a new problem. If they answer it incorrectly, they are provided with a small tutoring session where they must answer a few questions that break the problem down into steps. In Cognitive Tutor (Koedinger & Anderson, 1998), a problem in the tutor can also consist of questions of differing skills. Once a student has mastered a skill, as determined by a Knowledge Tracing (KT) model, the student no longer needs to answer questions of that skill within a problem but must answer the other questions which are associated with the unmastered skill(s) (e.g. the probability of student knowledge mastery is less than the threshold of 95%). It divides lessons into multiple pieces called Units. A skill that appears in one unit is treated as another separate skill when appearing in a different Unit. It is built around scaffolded, formative assessment, where each step a student takes to answer a problem is counted as a different activity at each step, a different skill may be assessed.

3.3.1. Static datasets

3.3.1.1. Fraction. Middle-school test with 536 students over 20 fraction subtraction problems, 8 KCs, 10k responses (DeCarlo, 2011).

3.3.1.2. ECPE. Language test with 2922 students over 28 questions with 3 KCs. It has 81k responses (Templin & Hoffman, 2013).

3.3.1.3. TIMSS. 757 students over 23 math problems from the Trends in International Mathematics and Science Study test in 2003, 13 KCs, with 17k responses (Skaggs et al., 2016).

Several datasets of static data are available in the R package CDM (George et al., 2016).

[2] https://sites.google.com/site/assistmentsdata/.

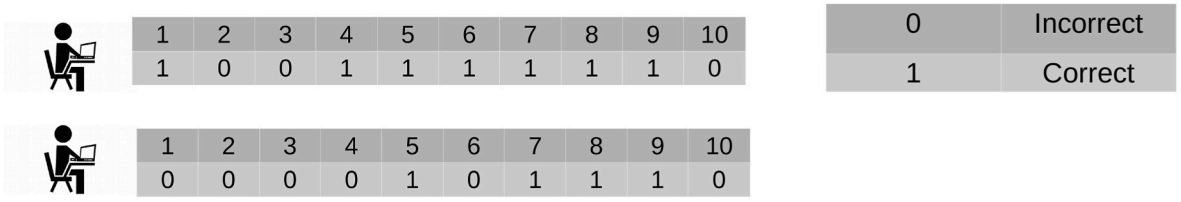

Fig. 1. Example of static data (Problem ID with student's responses): where every student has to answer the same number of problems in the same order during a standardized test.

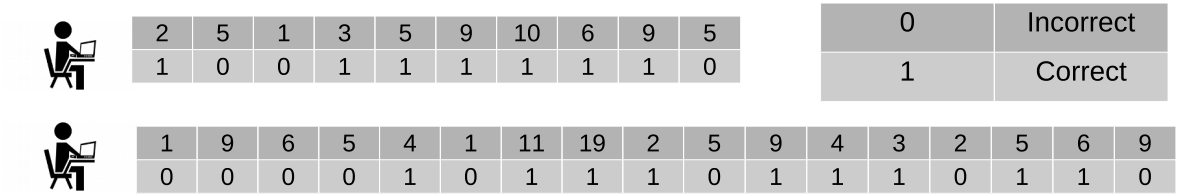

Fig. 2. Example of dynamic data (Problem ID with student's responses): where student can answer the problems and quit whenever she wants and problems are given in random order or based on her ability at each timestamp (e.g. first student quits the system after answering 10 questions while the second student keeps answering the questions).

Table 1
Overview of example datasets.

Dataset	Number of				Description
	Skills	Problems	Students	Records	
Static	8	20	536	10,720	Fraction (DeCarlo, 2011)
	3	28	2922	81,816	ECPE (Templin & Hoffman, 2013)
	13	23	757	17,411	TIMSS (Skaggs, Wilkins, & Hein, 2016)
Dynamic	123	13,002	4163	278,607	ASSISTments 09–10 (Razzaq et al., 2005)
	100	NA	19,840	683,801	ASSISTments 14–15 (Feng, Heffernan, & Koedinger, 2009)
	437	15,663	574	808,775	Algebra 2010 (Corbett, 2001)

3.3.2. *Dynamic datasets*

3.3.2.1. ASSISTments 2009–2010. This comes with two different datasets: *skill builder* dataset associated with a formative assessment where a student works until a the mastery threshold is reached (Razzaq et al., 2005) and the *non skill builder* dataset associated with summative assessment where a student has a fixed number of problems.

3.3.2.2. ASSISTments 2014–2015. This dataset contains records that represent a sequence of attempts to a related set of problems in a mastery learning problem set (Xiong, Zhao, Van Inwegen, & Beck, 2016). It contains no individual problem identifier and therefore problem difficulty cannot be computed from this data set.

3.3.2.3. Algebra 2005–2006. This is a development dataset released in KDD Cup 2010 competition from Carnegie Learning of PSLC DataShop. The PSLC DataShop released several datasets derived from Carnegie Learning's Cognitive Tutor (Corbett, 2001)..[3]

Although both ASSISTments and Cognitive Algebra Tutor are the systems used for practicing and learning mathematics skills for solving problems, the KDD Cup dataset is actually rather different from ASSISTments. A multitude of student models is compared based on these datasets. Various types of datasets are also available across different tutoring platforms. Example datasets in this literature review are publicly available publications from the internet. More datasets are found in (Romero & Ventura, 2020).

4. Student modeling

Once we have an efficient domain model (the mapping of skills to items), we can tackle the problem of Knowledge assessment with student modeling which constitutes another fundamental building block of ITS. To build an adaptive learning environment, it is essential to have an effective student model to assess student knowledge and trace their knowledge gains. An overview of knowledge assessment can be generally categorized into two main categories as in Fig. 3.

4.1. *Diagnostic Classification Models*

In the early stage of educational research, many student models were proposed by psychometricians to assess students' knowledge state in exam data (static data) with a long history within the psychometrics field. We refer to models that combine a student profile matrix and a Q-matrix as Diagnostic Classification Models (DCM) (Templin Hensonet al., 2010). Many models come from psychometrics. Compared to the classic Item Response Theory with continuous latent attributes, which is covered later, DCM represents discrete latent attributes. These attribute patterns are binary vectors with 1 indicating mastery of that latent attribute and 0 otherwise. These patterns provide feedback to teachers to help with designing remedial instructions. They are alternatively called as: Restricted Latent Class Models (Haertel, 1989), Latent Response Models (Maris, 1995), Multiple Classification Latent Class Models (Maris, 1999), Cognitive Diagnosis Models (Templin & Henson, 2006), Cognitive Psychometric Models (Rupp, 2007), Structured Item Response Models (Rupp & Mislevy, 2007), Structured Located Latent Class Models (Xu & von Davier, 2008).

A variety of DCMs has been proposed over the past decade. Depending on the nature of the models, DCMs can be classified into two categories,

- *compensatory*
- *non-compensatory*

[3] https://pslcdatashop.web.cmu.edu/KDDCup/downloads.jsp.

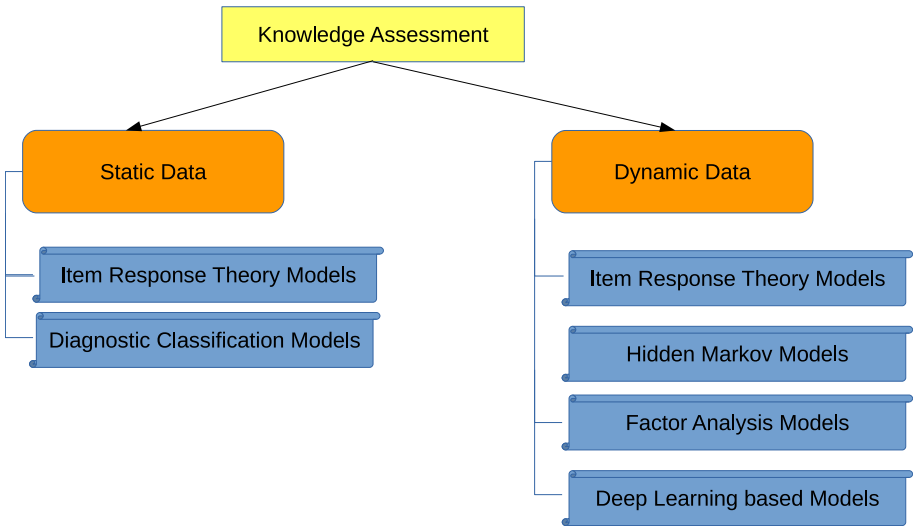

Fig. 3. An overview of general knowledge assessment.

Non-compensatory DCM models presume that each item requires a particular set of skills and lacking any one of them would lead the student to failure. The model is called deterministic-input noisy-and (DINA) model (De, 2009). It is a well-known non-compensatory model. By contrast, the noisy-or-gate (DINO) (Templin & Henson, 2006) is at the other extreme of this requirement: any skill is sufficient for success. The DINA model is also considered a *conjunctive* model, whereas DINO is considered a *disjunctive* one. Regardless of their formulation, all of the above models share the general idea of relating each item to single or multiple latent attributes.

4.1.1. Compensatory models

An example of a compensatory model is found in a real study to assess the prevalence of pathological gambling traits (Templin Hensonet al., 2010). Respondents who agreed with the following assertion:

> "Gambling got me into trouble over my financial situation."
>
> May involve the presence of two attributes:
>
> Attribute 1: breaks the law such as forgery, fraud, theft, or embezzlement to finance gambling.
> Attribute 2: depends on money provided by others to relieve a desiderata situation caused by gambling.

Any attribute may trigger a positive answer to the assertion, but not necessarily. However, the presence of both attributes is more likely to trigger a positive answer than any single attribute alone. A similar argument can be made for two skills linked to an item. Mastery of any of the two skills (traits) may lead to success at the item, but the mastery of both will make the success more likely.

The deterministic input, noisy-or-gate (DINO) model is a well-known *disjunctive* model in skills modeling (Templin & Henson, 2006). The DINO model is a full compensatory DCM. Respondents have a high probability of providing a correct answer with any one of the required skills instead of all of the required skills. Upon the given response matrix (which represents whether respondents provide a correct response to items or not and is represented with binary value), we are going to assess skills or attributes behind those items. An assessment consisting of I items considers measuring a domain of K attributes or skills.

Let X_{ij}, $i = 1, 2, \ldots, I, j = 1, 2, \ldots, J$, be a binary 0/1 response for item j by respondent i with 1 representing the respondent providing a correct response to the item and 0 otherwise. The attribute pattern for respondent i, α_i is a vector of length K with binary 0/1 elements with 1 meaning the respondent has mastered the attribute and 0 otherwise. For a test requiring K attributes, respondents can be classified into one of the 2^K possible attribute patterns.

P_{ij} is the probability of respondent i answers item j correctly, the probability is given by DINO model as follows:

$$P_{ij} = P(X_{ij} = 1|\xi_{ij}) = (1 - s_i)^{\xi_{ij}} g_i^{1-\xi_{ij}} \tag{1}$$

where

$$\xi_{ij} = 1 - \prod_{j=1}^{J} (1 - \alpha_{ik})^{q_{jk}}$$

$$s_j = P(X_{ij} = 0|\xi_{ij} = 1)$$

$$g_j = P(X_{ij} = 1|\xi_{ij} = 0)$$

where k is the skill, X_{ij} is the response (outcome) of the respondent i to the item j, q_{jk} is the $(j,k)^{th}$ element of Q-matrix Q, α_{ik} is the attribute pattern for that student i. The model is parameterized by s_j slip (the chances that a student that masters all required skills has a wrong answer to the item j), and g_j guess (the chances that a student that masters none of the required skills has a correct answer to the item j).

Based on the formula above, item j is correct with two possible probabilities. If a respondent has mastered none of the required attributes, they are still likely to provide a correct answer via guessing, so the correct response probability, in this case, is g_j. When a respondent has mastered at least one of the required attributes, the correct response probability is $1 - s_j$. The DINO model can be estimated using Markov chain Monte Carlo (MCMC) (Templin & Henson, 2006) or as a constrained log-linear model with latent classes.

4.1.2. Non-compensatory models

Non-compensatory DCMs require all skills for success to an item. For instance, to solve the math problem of

$$5 \times 3 - 9 = ?$$

the respondent requires all of these two elementary math skills:

- integer multiplication
- integer subtraction

The most popular DCM non-compensatory model is DINA, the noisy-and-gate. Similar to DINO model, DINA model also takes into account the possibility that a respondent with all required skills misses an item

and possibility through careless errors (slip parameter) and the possibility that respondent who lack at least one of the required skills gives a correct response by guessing (guess parameter).

For the DINA model, P_{ij}, the probability of respondent i answers item j correctly, is calculated as follows:

$$P_{ij} = P(X_{ij} = 1|\xi_{ij}) = (1 - s_j)^{\xi_{ij}} g_j^{1-\xi_{ij}} \tag{2}$$

where:

$$\xi_{ij} = \prod_{j=1}^{J} \alpha_{ik}^{q_{jk}}$$

$$s_j = P(X_{ij} = 0|\xi_{ij} = 1)$$

$$g_j = P(X_{ij} = 1|\xi_{ij} = 0)$$

where k is the skill, X_{ij} is the response (outcome) of the respondent i to the item j, q_{jk} is the $(j,k)^{th}$ element of Q-matrix Q, α_{ik} is the attribute pattern for that student i. The model is parameterized by s_j slip (the chances that a student that masters all required skills has a wrong answer to the item j), and g_j guess (the chances that a student that masters none of the required skills has a correct answer to the item j).

Response outcome is binary: {0, 1}. When $\xi_{ij} = 1$, the student i has mastered all the required skills and 0 otherwise.

4.2. Item Response Theory models

For over a few decades, Item Response Theory (IRT) (Rasch, 1961, 1993) served as the basis of assessment mechanism for computerized adaptive testing environments (Desmarais & d Baker, 2012). In IRT, the probability of getting success on a task (item) increases as a function of the level of proficiency (ability) behind all tasks with the assumption that the student knowledge state is static in an exam. Student knowledge state θ is assessed by her proficiency when a test is performed and each item tested helps bring information to refine the estimate of the knowledge state. In addition to the static knowledge assumption, original IRT models a single skill and assumes the test items are unidimensional (González-Brenes, Huang, & Brusilovsky, 2014; van der Linden & Hambleton, 2013; Wilson, Karklin, Han, & Ekanadham, 2016).

In standardized tests, students' proficiency is assessed by one static latent variable. The Rasch Model (Hambleton, Swaminathan, & Rogers, 1991) is an example of IRT and it shows a strong theoretical background both in terms of being grounded in psychometric measurement and a sound mathematical framework. IRT takes dichotomous item response outcomes and assigns student i with a proficiency θ_i, it can be measured after each question. Each item j has its own difficulty β_j.

4.2.1. IRT

The main idea of IRT is estimating a probability that student i answers item j correctly by using student ability and item difficulty,

$$P_{ij} = \sigma(\theta_i - \beta_j) = \frac{1}{1 + e^{\theta_i - \beta_j}} \tag{3}$$

where σ is the logistic function, θ_i is the student i proficiency on the topic tested, and β_j is the difficulty of item j.

The multidimensional-IRT model (a variant of IRT) can handle two and more dimensions (Reckase & McKinley, 1991; Briggs, 2003) but its complexity is much greater, and has not yet been used widely in a personalized learning environment (Desmarais & d Baker, 2012).

An important consideration is that IRT is not considered a *Knowledge Tracing* approach to the extent that it makes the assumption that the student does not learn during the process of testing. It is considered a *Knowledge Assessment* approach and each new item tested helps bring information to refine the estimate of the knowledge state, θ_i of student i. We will see later that other approaches explicitly model the learning process and are therefore named *Knowledge Tracing* approaches.

Note that when IRT is used in the context of a learning environment, where the knowledge state is *expected* to change, the assessment is often done on the first attempt at items, in the context of an exercise where multiple attempts are allowed. Alternatively, multiple trials can also be counted as different items and an index t will be used on β_j that corresponds to the trial.

4.2.2. IRT*

Wilson et al. (Wilson et al., 2016) proposed a Bayesian extension of IRT, that is used in the context of dynamic data (We will refer to this model as IRT*). which uses the Bayesian approach and regularizes on $logP_{ij}$ by imposing independent standard normal prior distributions over each θ_i and β_j. It maximizes on log posterior probability of $\{\theta_i, \beta_j\}$ given the response data $\{r : (i, j, r, t) \in D\}$, where response $r \in \{0, 1\}$, t is the time of each attempt and the student response data D as a set of tuples (i, j, r, t) indicating the student, item, response, and time of each response.

$$\begin{aligned} \log P(\{\theta_i\}, \{\beta_j\}|D) &= \sum_{(i,j,r,t)\in D} r \log \sigma(\theta_i - \beta_j) + (1 - r) \\ &\log(1 - \sigma(\theta_i - \beta_j)) - \frac{1}{2}\sum_i \theta_i^2 - \frac{1}{2}\sum_j \beta_j^2 + C \end{aligned} \tag{4}$$

IRT* leverages the information of both items and students that directly interact in the system. Maximum a posteriori (MAP) estimates of θ_i and β_j are computed by the Newton-Raphson method. We will refer this model as IRT*.

IRT* shows competitive performance to Deep Knowledge Tracing (DKT, a neural networks based knowledge tracing model mentioned in section 4.5) in terms of student performance prediction.

4.3. Hidden Markov models

With the development of adaptive and interactive learning environments, the assumption of a static student knowledge state is no longer tenable and models that drop this assumption have emerged in the late 1990s along with the term *Knowledge Tracing*, as discussed earlier.

Bayesian Knowledge Tracing (BKT) is the earliest approach to model a learner's changing knowledge state and is arguably the first model to relax the assumption on static knowledge states. In the original BKT, a single skill is tested per item and the learner's state of skill mastery is inferred at each time step based on a sequence of her previous outcomes (Corbett & Anderson, 1994). This approach is particularly relevant for tutors that use exercises and scaffolding as the main vehicle for learning and that monitor fine-grained skill mastery to decide on the next step. It relies on the Markov model to infer mastery states, from "not learned" to "learned" and to the extent the probabilities above depend either on fixed parameters and on the state in the time step t in Fig. 4.

The standard BKT model is comprised of 4 parameters which are typically learned from the data while building a model for each skill. The model's inferred probability mainly depends on the parameters which are used to estimate how a student masters a skill given that student's chronological sequence of binary outcomes (correct/incorrect) to questions of that skill thus far.

BKT has four parameters:

- $P(L_0^k)$: the prior probability of a student mastering the skill k;
- $P(T^k)$: the probability a student, who does not currently master the skill k, will master it after the next practice opportunity;
- $P(G^k)$: the probability a student guesses a question and gets a correct answer despite not mastering the skill k (*guess*); and
- $P(S^k)$: the probability a student answers a question incorrectly despite mastering the skill k (*slip*).

In a typical learning environment with BKT, the estimate of student mastery of a skill is continually updated every time student gives a

response to an item as in Figs. 4 and 5. $P(L^k_{t+1})$ is the probability that a student masters the skill k at time $t+1$. $P(L^k_{t+1})$, skill mastery is computed as:

$$P(L^k_t|Correct) = \frac{P(L^k_{t-1})(1-P(S^k))}{P(L^k_t)(1-P(S^k)) + (1-P(L^k_t))P(G^k)} \quad (5)$$

$$P(L^k_t|Incorrect) = \frac{P(L^k_{t-1})P(S^k)}{P(L^k_t)P(S^k) + (1-P(L^k_t))(1-P(G^k))} \quad (6)$$

$$P(L^k_{t+1}) = P(L^k_t|Obs) + (1-P(L^k_t|Obs))P(T^k) \quad (7)$$

And P_{ij} is the probability of student i applying the skill k correctly of problem j at time $t+1$. The prediction is performed according to the mastery of skill $P(L^k_{t+1})$ at time $t+1$ as follows:

$$P_{ij} = P_{ij}(k_{t+1}) = P(L^k_{t+1}).(1-P(S^k)) + (1-P(L^k_{t+1})).P(G^k) \quad (8)$$

First, $P(L^k_{t+1})$, the probability that a student masters the skill k at timestamp $t+1$ is updated by using the equation (7) with equation (5) or (6) according to the observation on student's outcome *Obs* where *Obs* $\in$ {*Correct*, *Incorrect*} the response evidence at timestamp t. Then, the system estimates the possibility that the student will apply the skill k correctly of problem j at time $t+1$ by using equation (8). With Bayesian knowledge tracing algorithm used in cognitive tutoring system, the student will be instructed to practice similar problems with a particular skill, when the system does not recognize that the student has sufficient knowledge about that skill (e.g. the probability that the student knows the skill is less than 95%) (d Baker, Corbett, & Aleven, 2008).

Several extensions of BKT have been introduced as: BKT with contextualized guessing and slipping parameters (d Baker et al., 2008), BKT with an estimated probability of transition from the use of help features (Baker & Yacef, 2009), BKT with the use of student prior knowledge to the skills (Pardos & Heffernan, 2010), BKT with the use of item difficulty (Pardos & Heffernan, 2011), BKT with the use of clusters of different student groups (Pardos, Trivedi, Heffernan, & Sárközy, 2012), BKT with student-specific parameters (Yudelson, Koedinger, & Gordon, 2013), BKT with the use of general features (González-Brenes et al., 2014), BKT with the use of item difficulty and student ability profile (Minn, Vie, Koh, Kashima, & Zhu, 2022).

4.4. Factor analysis models

Performance Factor Analysis (PFA) was adapted from Learning Factor Analysis (LFA (Cen, Koedinger, & Junker, 2006)) to allow the creation of a "model overlay" that traces predictions for individual students with individual skills so as to provide the adaptive instruction to automatically remediate current performance (Pavlik et al., Koedinger).

LFA accumulates learning for student i using one or more skills k.

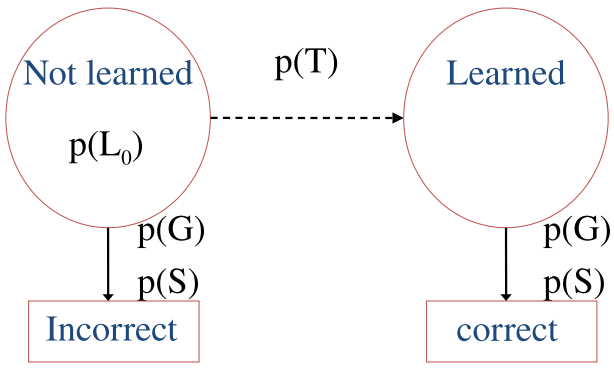

Fig. 4. Example of standard BKT which estimate student's knowledge state transition from "not learned" to "learned" of a skill.

$$\log P_{ij} = \theta_i + \sum_{k \in KCs(j)} (\beta_k + \gamma_k n_{ik}) \quad (9)$$

where n_{ik} is the number of attempts of student i on skill k, and $KCs(j)$ is the number of skills associated to item j. LFA is parameterized by θ_i the ability for student i, β_k the bias (easiness) for the skill k, and γ_k the bias for a prior attempt to skill k.

This model is an extension of the Rasch model (unidimensional IRT) which has an equivalent form to Equation (9) with γ_k set to 0 and only a single β_k value.

Pavlik et al. (Pavlik et al., Koedinger) adapted the LFA model with sensitivity to the indicator of student learning performance. The PFA model allows conjunction by summing the contributions from all skills needed in a performance. This kind of "compensatory" model of multi-skill learning allows the lack of one KC to compensate for the presence of another in addition to showing conjunctive effects.

PFA also relaxes the static knowledge assumption and models multiple skills simultaneously (Pavlik et al., Koedinger). Its basic structure is:

$$\log P_{ij} = \sum_{k \in KCs(j)} (\beta_k + \gamma_k s_{ik} + \rho_k f_{ik}) \quad (10)$$

where s_{ik} the number of successful attempts of student i on skill k, f_{ik} the number of failure attempts of student i on skill k, and $KCs(j)$ the number of skills associated to item j. PFA is parameterized by β_k the bias for the skill k, γ_k: the bias for success attempt to skill k, ρ_k the bias for failure attempt to skill k.

PFA does not consider student proficiency θ, because it assumes that θ cannot be estimated ahead of time in adaptive situations (Pavlik et al., Koedinger).

Varieties of models have been introduced to compete in terms of predictive power. Those include Difficulty, Ability, and Study History (DASH) (Lindsey, Shroyer, Pashler, & Mozer, 2014), Sparse factor analysis for learning and content analytics (SPARFA) (Lan, Waters, Studer, & Baraniuk, 2014), Knowledge Tracing Machines (Vie & Kashima, 2019), DAS3H: Skill, and Student Skill practice History (Choffin, Popineau, Bourda, & Vie, 2019).

4.5. Deep learning-based models

Deep learning has more recently been used for Knowledge Tracing, but as it did in many domains before, their performance challenges the dominant student models.

Piech et al. (Piech et al., 2015) introduced Deep Knowledge Tracing (DKT) in 2015. Akin to BKT, it uses data in which skills are tried and the performance outcome is used to predict future sequence attempts. DKT uses large numbers of artificial neurons for representing latent knowledge state along with a temporal dynamic structure and allows a model to learn the student's knowledge state from data. It encodes skill and student response attempts in a one-hot feature input vector as input for each time t. The output layer y_t provides the predicted probability that the student would answer that particular next problem correctly at time $t+1$ as in Fig. 6. Recurrent Neural Networks (RNNs) map an input sequence of vectors $x_1, \ldots, x_T$, to an output sequence of vectors $y_1, \ldots, y_T$ based on a sequence of hidden states $h_1, \ldots, h_T$ a successive encodings of relevant information from past observations. It is defined by the following equations:

$$h_t = \tanh(W_{hx}x_t + W_{hh}h_{t-1} + b_h) \quad (11)$$

$$y_t = \sigma(W_{yh}h_t + b_y) \quad (12)$$

In DKT, both tanh and the sigmoid function are applied element-wise and parameterized by an input weight matrix W_{hx}, recurrent weight matrix W_{hh}, initial state h_0, the hidden state h_t, and output weight matrix W_{yh}. Biases for latent and output units are represented by b_h and b_y. x_t is a

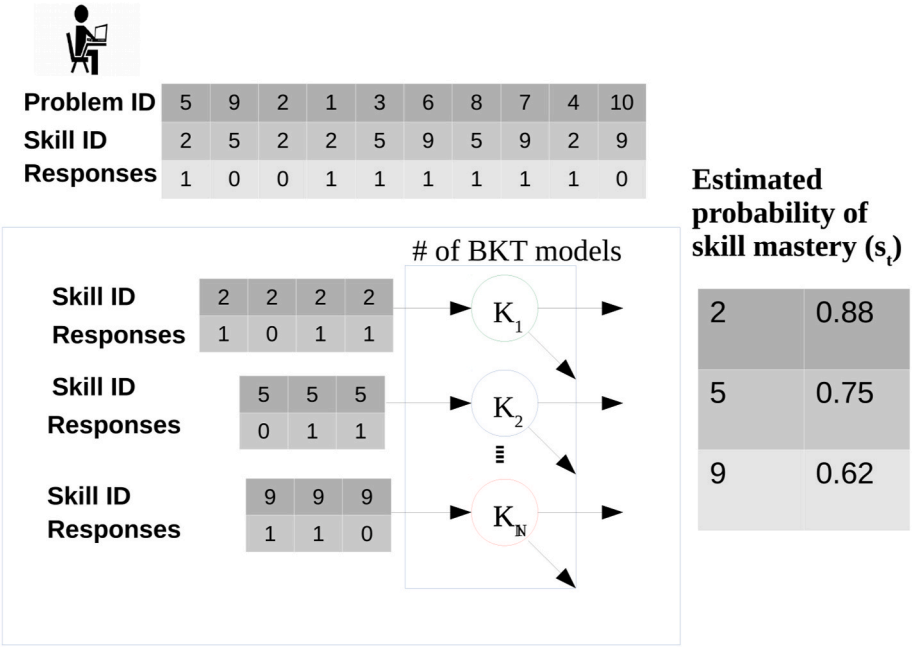

Fig. 5. BKT (skill-specific models): skill mastery assessment.

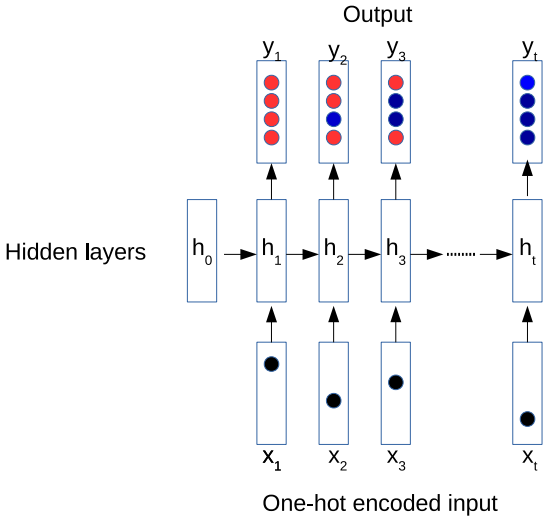

Fig. 6. DKT architecture.

one-hot encoded vector of the student interaction $x_t = \{k_t, r_t\}$ that represents the combination of which skill k_t was practiced with student response r_t, so $x_t \in \{0,1\}^{2M}$ according to number M of unique skills. The output y_t is a vector of number of skills K, where each value represents the probability that the student would answer the particular problem with associated skill k correctly at time $t + 1$.

Thus, the probability of student i will answer problem j associated with skill k correctly at time $t + 1$ can be retrieved from vector y_t.

$$P_{ij} = P_{ij}(k_{t+1}) \in y_t \tag{13}$$

DKT uses Recurrent Neural Networks (RNNs) to represent the latent knowledge space of students along with the number of practices dynamically. The increase in students' knowledge through an assignment can be inferred by utilizing the history of students' previous performance.

Deep learning has an impact on a wide range of fields like natural language processing (such as machine translation, language modeling, question answering (Chen et al., Chenet al.; Edunov et al., 2018; Dhingra et al., Salakhutdinov; Devlin, Chang, Lee, & Toutanova, 2018)), image classification, and computer vision (such as image captioning (Vinyals, Toshev, Bengio, & Erhan, 2015; Xu et al., 2015)). While it is beyond the scope of our work to include a comprehensive foundation in deep learning, we refer interested readers to the introductory textbooks (Goodfellow, Bengio, & Courville, 2016; LeCun, Bengio, & Hinton, 2015). Here, we briefly mention variants of recent deep learning-based knowledge tracing models. They are Dynamic Key-Value Memory Networks (Zhang, Shi, King, & Yeung, 2017), DKT with Dynamic Student Classification (Minn et al., 2018a), Prerequisite-Driven DKT (Chen, Lu, Zheng, & Pian, 2018), Exercise-Enhanced RNN with Attention (Su et al., 2018), Dynamic Student Classification on Memory Networks (Minn, Desmarais, Zhu, Xiao, & Wang, 2019), Deep hierarchical knowledge tracing (Wang, Ma, & Gao, 2019), Sequential Key-Value Memory Networks (Abdelrahman & Wang, 2019), DKT with convolutions (Yang et al., 2008), Graph-based Interaction Knowledge Tracing (Yang et al., Yu), BKT-LSTM with meaningful features (Minn, 2012).

4.6. Evaluation

Evaluation of student models is required to guarantee that the model accurately assesses students' knowledge states during their interactions with learning systems. In the case of summative assessment, a student's ability to solve a problem on a given skill is equally relevant to predicting any other problems associated with that skill. For example, IRT is measured by using statistics such as model fit (Khalid). In the case of formative assessment, student models are typically validated with reference to two criteria. The first way to evaluate the student model is measured by predicting future student performance within the learning system. The second way is validation with external measures (e.g. student's knowledge gain in the post-tests) (Desmarais & d Baker, 2012).

In both cases of assessment, the actual student knowledge state is latent and non-observable but can be estimated based on their past performance on other items. Estimated knowledge state on a skill is used to predict the student's performance (success or failure) on the next problems associated with that skill. So student models are usually

compared with their predictability in students' performance.

Thus, student models are commonly evaluated in terms of Area Under the ROC Curve (AUC) (Hanley & McNeil, 1982), root mean squared error (RMSE) (Chai & Draxler, 2014). AUC measures the entire two-dimensional area underneath the entire ROC curve from (0,0) to (1, 1) for predictions where the value being estimated is the probability of problem correctness. An AUC of 0.50 represents the score achievable by random guess. A higher AUC score represents higher accuracy. RMSE is used as a criterion to assess the model fit as well as a variant of Brier score. Lower values of RMSE indicate better performance of the model. AUC and RMSE provide robust metrics of evaluation and commonly measured in the task of knowledge tracing and student performance prediction (Gervet et al., 2020; Minn et al., 2022; Yudelson et al., 2013).

4.6.1. Methodology

For all datasets of static data, we apply datasets available in CDM package (Robitzsch, Kiefer, George, & Uenlue, 2015). Those are recorded from standardized test results over a specific course content (such as language and mathematics tests). So, it has no missing values or duplicate records in those datasets. For all datasets of dynamic data, only the first correct attempts to original problems are considered in our experiment. This is a standard practice in the field. We remove data with missing values for skills and problems with duplicate records. To the best of our knowledge, these are the well-known publicly available datasets for knowledge assessment and knowledge tracing problems.

In our experiment, 5 fold cross-validations are used to make predictions on all datasets. Each fold involves splitting each dataset into 80% training data and 20% test data at problem level in static data and at the student level in dynamic data. All these models are trained and tested on the exact same sets of training and testing datasets. In static data, problem responses (20%) from each student are chosen for testing and 80% of students' responses in the data are used for training of the student models. Students' responses to problems in the testing dataset are predicted by using learned parameters in each student model. In the context of dynamic data, the next response of a student is predicted by using current and previous response sequences in chronological order. Training of the student model is done by using 80% of students' response sequences in the dataset and the rest 20% of students' response sequences are used for testing. We apply the same hyperparameters from their original papers and those were chosen in the context of best performance. We compare problem student correctness prediction dominant student models mentioned above: DINA (De, 2009), IRT (Reckase, 2009), IRT* (Wilson et al., 2016), BKT (Corbett & Anderson, 1994), PFA (Pavlik et al., Koedinger), DKT (Piech et al., 2015). But we do not compare with other variants, because they are more or less similar and do not show significant performance differences.

4.6.2. Results

In our experiment, DINA shows similar (a bit lower) performance as IRT in fraction dataset, but it shows better performance in ECPE (language test) and lower performance in TIMSS (Mathematical test) for static data.

In the case of experiments in dynamic data, there are differences in the results of original DKT and DKT observed in this analysis. In the original DKT (Piech et al., 2015), the author utilized the data with duplicate records. We removed duplicate records and only take first-correct attempts into consideration in our data. More detailed analysis on DKT performance with various types of data processing in different deep learning platforms can be referred into the paper (Xiong et al., 2016). However, consistent with previous studies, DKT outperforms BKT by a large margin, where BKT serves as a baseline model among other dominant student models. Although PFA performs better than BKT, it shows lower performance than DKT and IRT* in all datasets (see Tables 2 and 3).

A notable result is that, in all of our data sets, IRT comes out as the top predictor in both static and dynamic datasets (except for ECPE and ASS-14 datasets). Given that IRT with a single latent skill, this is an unexpected result that shows the importance of comparing with other dominant approaches. However, some hybrid models that are not included in this review may have higher predictive performance than dominant models. (e.g. IKT (Minn et al., 2022) that is based on human interpretable features: skill mastery, ability profile, and item difficulty).

The strong predictive performance of IRT can be explained by two factors.

(1) IRT uses item difficulty in its prediction. This is supported by the fact that ASS-14 only has information on skills tried and does not allow the computation of item difficulty. For this data set, IRT* is above BKT, but below all other models.
(2) IRT*, being a single skill model, effectively ends up allowing learning transfer (discussed in the section 4.7) across skills, albeit at the cost of fine-grain skill assessment.

Table 2
RMSE result for all tested datasets.

Models	Static			Dynamic		
	Fraction	ECPE	TIMSS	ASS-09	ASS-14	Algebra
DINA	0.387 1	0.445 5	0.478 1	–	–	–
IRT	0.379 3	0.458 7	0.459 8			
IRT*				0.441 3	0.446 9	0.370 3
PFA	–	–	–	0.455 8	0.420 2	0.392 3
BKT	–	–	–	0.471 2	0.510 2	0.439 8
DKT	–	–	–	0.449 7	0.420 2	0.380 1

Table 3
AUC result for all tested datasets.

Models	Static			Dynamic		
	Fraction	ECPE	TIMSS	ASS-09	ASS-14	Algebra
DINA	0.869 0	0.707 7	0.709 5	–	–	–
IRT	0.870 7	0.626 1	0.736 3			
IRT*				0.751 2	0.670 5	0.812 3
PFA	–	–	–	0.701 2	0.692 1	0.761 5
BKT	–	–	–	0.651 1	0.610 1	0.642 1
DKT	–	–	–	0.723 1	0.708 2	0.786 3

4.7. Discussion

This article has reviewed the dominant student models used in both static and dynamic data for knowledge assessment and student performance prediction. In modeling with static data, student models are constructed with the assumption of "No Learning" due to the knowledge of the student does not change during a test. However, student models are designed to formulate "Learning" occurred when a student gets several opportunities of practicing the same skills in modeling of dynamic data.

Learning transfer implies that students can transfer their acquired skills to new situations and across problems involving a different skill set. Student models such as IRT, PFA, and DKT aim to capture this learning transfer phenomenon. They introduced a factor that represents the learning accumulated on all skills through practice and then utilized this factor as a predictor of success in further practice. These models have outperformed the standard BKT model without the skill transfer mechanism, and have shed new light on the importance to consider skill transfer, and have given rise to further research on the subject.

All of the recent neural networks based student models have a huge amount of parameters and complex structures, so it is difficult to provide a psychologically meaningful explanation that reflects human cognitive evolution theory. That is the main issue of recent deep learning-based student models for knowledge tracing (Khajah, Lindsey, & Mozer,

2016). Some researchers have also tried to improve the interpretability in student modeling and gotten some promising results with human interpretable features with a simple causal graph (Minn et al., 2022).

When a student learns with an intelligent tutoring system, they practice a specific skill through answering several questions, and the ITS checks their mastery of skills according to whether they were able to provide correct answers in their previous problems. However, even with a high level of mastery of a particular skill, students may incorrectly answer some problems. We can potentially tend to regard such a situation as the result of problem misunderstanding, or the student cannot utilize the skill properly in that particular problem under a new circumstance. Thus, student models with a provision of problem characteristics perform better than student models without problem information in student performance prediction (Minn et al., 2018b; Minn et al., 2019; Su et al., 2018; Wang et al., 2019; Yang et al., Yu). In some cases, new skills are required to solve a problem, those were not associated (or identified) yet in the tutoring system, so we can also assume that students may need to learn these new skills rather than practicing on a skill. That kind of new skill can be defined as an "integrative knowledge component". It integrates with all other KCs to produce a correct response (Koedinger et al., 2012).

This study also confirms an unexpected result that was first observed by Wilson et al. (Wilson et al., 2016). The IRT* model is a better performer in predicting the next item outcome for dynamic data. This result can be explained by the item difficulty factor which is explicitly taken into the models (Minn et al., 2022; Wilson et al., 2016) and not by DKT nor other student models without consideration of item information. This conclusion is further supported by the observation that the advantage of IRT* vanishes for the ASS-14 data set, for which item difficulty is not accessible because only the skill involved in the exercises is given. Additionally, some student models with explicit usage of item information provide better predictive performance than models without item information (Minn et al., 2019, 2022; Yeung, 2019).

Finally, performance results reported in some research papers are not directly comparable (even for the experiments done on the same dataset). The performance of each student model may also differ according to the types of data processing (e.g. the performance of the student model that builds on the data with duplicate records versus the data with first-correct attempts only). Xiong et al. (Xiong et al., 2016) and Wilson et al. (Wilson et al., 2016) re-examined one of the key datasets called ASS-09 and used BKT and DKT for comparing the differences in model performance among different types of data processing. It shows significant differences in prediction performance according to their data processing methods. More details can be found in (Xiong et al., 2016).

Each model has its own significant characteristics. We summarize the characteristics of described dominant student models in Table 4.

In this comparison, we only compare the characteristics of original models in their families. A multitude of variant and hybrid models was proposed throughout the decades. More works can be referred to the variants of each family in section 4.

5. Future directions and conclusion

A goal of an adaptive learning environment is to increase students' learning gains whilst keeping students in their zone of proximal development (Wertsch, 1984). So we need to assess students' knowledge state changing at each opportunity of their practices. Knowledge assessment serves as a core engine of intelligent tutoring systems. A student may or may not answer a problem correctly is mainly depending on his/her mastery of the required skill(s) for that problem. However, having a high level of proficiency in all required skills of the problem may not provide the correctness on that problem in some cases. It may require a new additional skill (e.g. integrative skill) to answer the problem correctly. Without having that kind of skill, the students may carry misconceptions at the problem level (e.g. they do not understand the problem very well, or they cannot utilize the mastered skill properly in that particular problem under a new circumstance). Some student models remedy this problem by taking problem difficulty into account in student modeling. These models show improvement in predictive performance with interpretability to the extent (Minn et al., 2019, 2022; Wilson et al., 2016), but it is not enough for providing an explanation on what kind of misconception may occur in which stage of student's problem-solving. We further need to investigate an efficient student model that can provide the reasoning about misconception at a high degree of granularity for diagnostic propose.

Student models are often developed by using a combination of experts' domain knowledge in their fields (including feature engineering and knowledge engineering) with the help of domain experts and student modeling by psychometrics, machine learning, and data mining techniques for providing diagnostic and prognostic reasoning with psychologically meaningful parameters. Recent deep learning based student models show impressive prediction power in the task of student performance prediction. However, these models work all these tasks together in complex network structures with a huge amount of parameters. Hence, it is difficult to analyze their parameters in terms of a psychologically meaningful way (Khajah et al., 2016). We believe that efforts on Explainable AI and causal inference will provide a solution to this limitation. Some researchers are already working on interpretability for student modeling but the outcomes have not reached yet a satisfactory level of the psychologically meaningful stage (Lu, Wang, Meng, & Chen, 2020; Minn et al., 2022; Yeung, 2019).

To summarize, we cover the dominant families of student modeling techniques with psychometric theory, recent adaptations, and advances with machine learning and deep learning techniques. Moreover, we aim to provide a better understanding of the usages of AI in knowledge assessment for educational researchers and practitioners, and the development of student models on different data types with their challenges and potential solutions for the future ecosystem of adaptive learning environments.

Statements on open data and ethics

In this study, we used the datasets from standardized tests and distinct tutoring scenarios in which students interact with a computer-based testing/learning system in educational settings. All of them are publicly available online and links to those datasets are provided.

Table 4
Comparison of characteristics of dominant student models.

Models	DINA (De, 2009)	IRT (Rasch, 1961, 1993)	PFA (Pavlik et al., Koedinger)	BKT (Corbett & Anderson, 1994)	DKT (Piech et al., 2015)
Work on data	Static	Both	Dynamic	Dynamic	Dynamic
Learn on skill(s)	Multiple	Single	Multiple	Single	Single
Student ability	Binary	Continuous	Continuous	Continuous	Continuous
Interpretability	Yes	Yes	Yes	Yes	No
Sequential Model	No	No	No	Yes	Yes
Learning transfer	No	Yes	Yes	No	Yes
Use of item information	No	Yes	No	No	No
Predictive Performance	High	High	Medium	Low	High

Declaration of competing interest

The author declares that I have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The author is especially grateful to Professor Michel Desmarais, Polytechnique Montreal, and thanks to everyone who participated in the discussions, for their valuable comments and helpful suggestions.

References

Abdelrahman, G., & Wang, Q. (2019). Knowledge tracing with sequential key-value memory networks. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval* (pp. 175–184).

Adedoyin, O. B., & Soykan, E. (2020). Covid-19 pandemic and online learning: The challenges and opportunities. *Interactive Learning Environments*, 1–13.

Anderson, J. R., & Reiser, B. J. (1985). The LISP tutor. *Byte, 10*(4), 159–175.

Baker, R. S. (2016). Stupid tutoring systems, intelligent humans. *International Journal of Artificial Intelligence in Education, 26*(2), 600–614.

d Baker, R. S., Corbett, A. T., & Aleven, V. (2008). More accurate student modeling through contextual estimation of slip and guess probabilities in bayesian knowledge tracing. In *International conference on intelligent tutoring systems* (pp. 406–415). Springer.

Baker, R. S., & Yacef, K. (2009). The state of educational data mining in 2009: A review and future visions. *JEDM—Journal of Educational Data Mining, 1*(1), 3–17.

Bennett, R. E. (2011). Formative assessment: A critical review. *Assessment in Education: Principles, Policy & Practice, 18*(1), 5–25.

Cen, H., Koedinger, K., & Junker, B. (2006). Learning factors analysis – a general method for cognitive model evaluation and improvement. In *International conference on intelligent tutoring systems* (pp. 164–175). Springer.

Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)?–Arguments against avoiding RMSE in the literature. *Geoscientific Model Development, 7*(3), 1247–1250.

D. C. Briggs, M. Wilson (2003) An introduction to multidimensional measurement using Rasch models, Applied Measurement, 4(1), 87-100.

M. X. Chen, O. Firat, A. Bapna, M. Johnson, W. Macherey, G. Foster, L. Jones, N. Parmar, M. Schuster, Z. Chen, et al., The best of both worlds: Combining recent advances in neural machine translation, Proceedings of the 56th annual meeting of the association for computational linguistics.

Chen, P., Lu, Y., Zheng, V. W., & Pian, Y. (2018). Prerequisite-driven deep knowledge tracing. In *2018 IEEE international conference on data mining (ICDM)* (pp. 39–48). IEEE.

Chiu, C.-Y. (2013). Statistical refinement of the Q-matrix in cognitive diagnosis. *Applied Psychological Measurement, 37*(8), 598–618.

Choffin, B., Popineau, F., Bourda, Y., & Vie, J.-J. (2019). DAS3H: Modeling student learning and forgetting for optimally scheduling distributed practice of skills. In *The international conference on educational data mining*.

Corbett, A. (2001). Cognitive computer tutors: Solving the two-sigma problem. In *International conference on user modeling, adaptation, and personalization* (pp. 137–147). Springer.

Corbett, A. T., & Anderson, J. R. (1994). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction, 4*(4), 253–278.

De La Torre, J. (2008). An empirically based method of Q-matrix validation for the DINA model: Development and applications. *Journal of Educational Measurement, 45*(4), 343–362.

De La Torre, J. (2009). DINA model and parameter estimation: A didactic. *Journal of Educational and Behavioral Statistics, 34*(1), 115–130.

DeCarlo, L. T. (2011). On the analysis of fraction subtraction data: The DINA model, classification, latent class sizes, and the Q-matrix. *Applied Psychological Measurement, 35*(1), 8–26.

Desmarais, M. C., & d Baker, R. S. (2012). A review of recent advances in learner and skill modeling in intelligent learning environments. *User Modeling and User-Adapted Interaction, 22*(1–2), 9–38.

Desmarais, M. C., & Naceur, R. (2013). A matrix factorization method for mapping items to skills and for enhancing expert-based Q-matrices. In *International conference on artificial intelligence in education* (pp. 441–450). Springer.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT* (pp. 4171–4186).

B. Dhingra, H. Liu, Z. Yang, W. W. Cohen, R. Salakhutdinov, Gated-attention readers for text comprehension, Proceedings of the 55th annual meeting of the association for computational linguistics.

S. Edunov, M. Ott, M. Auli, D. Grangier, Understanding back-translation at scale, Proceedings of the 2018 conference on empirical methods in natural language processing.

Evens, M., Chang, R.-C., Lee, Y. H., Shim, L. S., Woo, C. W., & Zbang, Y. (1997). CIRCSIM-Tutor: An intelligent tutoring system using natural language dialogue. In *Fifth conference on applied natural language processing: Descriptions of system demonstrations and videos* (pp. 13–14).

Fauvel, S., Yu, H., Miao, C., Cui, L., Song, H., Zhang, L., et al. (2018). Artificial intelligence powered MOOCs: A brief survey. In *2018 IEEE international conference on agents (ICA)* (pp. 56–61). IEEE.

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

Ferreira, A., & Atkinson, J. (2008). Designing a feedback component of an intelligent tutoring system for foreign language. In *International conference on innovative techniques and applications of artificial intelligence* (pp. 277–290). Springer.

Figueiredo, J., & García-Peñalvo, F. J. (2020). Intelligent tutoring systems approach to introductory programming courses. In *Eighth international conference on technological ecosystems for enhancing multiculturality* (pp. 34–39).

Fu, S., Zhang, Y., et al. (2013). On the recommender system for university library. *IADIS International Conference, e-Learning*, 215–222.

Gardner, L., Sheridan, D., & White, D. (2002). A web-based learning and assessment system to support flexible education. *Journal of Computer Assisted Learning, 18*(2), 125–136.

George, A. C., Robitzsch, A., Kiefer, T., Groß, J., & Ünlü, A. (2016). The R package CDM for cognitive diagnosis models. *Journal of Statistical Software, 74*(2), 1–24.

Gervet, T., Koedinger, K., Schneider, J., Mitchell, T., et al. (2020). When is deep learning the best approach to knowledge tracing? *JEDM—Journal of Educational Data Mining, 12*(3), 31–54.

González-Brenes, J., Huang, Y., & Brusilovsky, P. (2014). General features in knowledge tracing to model multiple subskills, temporal item response theory, and expert knowledge. In *The 7th international conference on educational data mining* (pp. 84–91). University of Pittsburgh.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Haertel, E. H. (1989). Using restricted latent class models to map the skill structure of achievement items. *Journal of Educational Measurement, 26*(4), 301–321.

Hambleton, R. K., Swaminathan, H., & Rogers, H. J. (1991). *Fundamentals of item response theory*. Sage.

Hanley, J. A., & McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology, 143*(1), 29–36.

Henly, D. C. (2003). Use of Web-based formative assessment to support student learning in a metabolism/nutrition unit. *European Journal of Dental Education, 7*(3), 116–122.

Hsu, C.-K., Hwang, G.-J., & Chang, C.-K. (2010). Development of a reading material recommendation system based on a knowledge engineering approach. *Computers & Education, 55*(1), 76–83.

Hwang, G.-J., & Chang, H.-F. (2011). A formative assessment-based mobile learning approach to improving the learning attitudes and achievements of students. *Computers & Education, 56*(4), 1023–1031.

Hwang, G.-J., Chu, H.-C., Yin, P.-Y., & Lin, J.-Y. (2008). An innovative parallel test sheet composition approach to meet multiple assessment criteria for national tests. *Computers & Education, 51*(3), 1058–1072.

Khajah, M., Lindsey, R. V., & Mozer, M. C. (2016). How deep is knowledge tracing?. In *Proceedings of the international conference on artificial intelligence in education*.

M. N. Khalid, reportIRT model fit from different perspectives, (Ph.D. Thesis).

Khlaif, Z. N., Salha, S., & Kouraichi, B. (2021). Emergency remote learning during COVID-19 crisis: Students' engagement. *Education and Information Technologies*, 1–23.

Koedinger, K. R., & Anderson, J. R. (1998). Illustrating principled design: The early evolution of a cognitive tutor for Algebra symbolization. *Interactive Learning Environments, 5*(1), 161–179.

Koedinger, K. R., Corbett, A. T., & Perfetti, C. (2012). The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive Science, 36*(5), 757–798.

Lan, A. S., Waters, A. E., Studer, C., & Baraniuk, R. G. (2014). Sparse factor Analysis for learning and content analytics. *Journal of Machine Learning Research, 15*(1), 1959–2008.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). *Deep Learning, Nature, 521*(7553), 436–444.

van der Linden, W. J., & Hambleton, R. K. (2013). *Handbook of modern item response theory*. Springer Science & Business Media.

Lindsey, R. V., Shroyer, J. D., Pashler, H., & Mozer, M. C. (2014). Improving students' long-term knowledge retention through personalized review. *Psychological Science, 25*(3), 639–647.

Lu, Y., Wang, D., Meng, Q., & Chen, P. (2020). Towards interpretable deep learning models for knowledge tracing. In *Proceedings of the international conference on artificial intelligence in education* (pp. 185–190).

Maris, E. (1995). Psychometric latent response models. *Psychometrika, 60*(4), 523–547.

Maris, E. (1999). Estimating multiple classification latent class models. *Psychometrika, 64*(2), 187–212.

S. Minn, BKT-LSTM: Efficient Student Modeling for knowledge tracing and student performance prediction, arXiv preprint arXiv:2012.12218.

Minn, S., Desmarais, M. C., & Fu, S. (2016). Refinement of a Q-matrix with an ensemble technique based on multi-label classification algorithms. In *European conference on technology enhanced learning* (pp. 165–178). Springer.

Minn, S., Desmarais, M. C., Zhu, F., Xiao, J., & Wang, J. (2019). Dynamic student classification on memory networks for knowledge tracing. In *Pacific-asia conference on knowledge discovery and data mining* (pp. 163–174). Springer.

Minn, S., Vie, J.-J., Koh, T., Kashima, H., & Zhu, F. (2022). Interpretable knowledge tracing: Simple and efficient student modeling with causal relations. In *Twelfth AAAI symposium on educational advances in artificial intelligence*.

Minn, S., Yu, Y., Desmarais, M. C., Zhu, F., & Vie, J.-J. (2018a). Deep knowledge tracing and dynamic student classification for knowledge tracing. In *2018 IEEE international conference on data mining (ICDM)* (pp. 1182–1187). IEEE.

Minn, S., Zhu, F., & Desmarais, M. C. (2018b). Improving knowledge tracing model by integrating problem difficulty. In *2018 IEEE international conference on data mining workshops (ICDMW)* (pp. 1505–1506). IEEE.
Mitrovic, A. (2003). An intelligent SQL tutor on the web. *International Journal of Artificial Intelligence in Education, 13*(2–4), 173–197.
Moss, C. M., & Brookhart, S. M. (2019). *Advancing formative assessment in every classroom: A guide for instructional leaders*. ASCD.
Pardos, Z. A., & Heffernan, N. T. (2010). Modeling individualization in a Bayesian networks implementation of knowledge tracing. In *International conference on user modeling, adaptation, and personalization* (pp. 255–266). Springer.
Pardos, Z. A., & Heffernan, N. T. (2011). KT-IDEM: Introducing item difficulty to the knowledge tracing model. In *International conference on user modeling, adaptation, and personalization* (pp. 243–254). Springer.
Pardos, Z. A., Trivedi, S., Heffernan, N. T., & Sárközy, G. N. (2012). Clustered knowledge tracing. In *International conference on intelligent tutoring systems* (pp. 405–410). Springer.
P. I. Pavlik Jr, H. Cen, K. R. Koedinger, Performance factors analysis–A new alternative to knowledge tracing., Online Submission.
Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L. J., et al. (2015). Deep knowledge tracing. In *Advances in neural information processing systems* (pp. 505–513).
Rasch, G. (1961). On general laws and the meaning of measurement in psychology. In *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability* (Vol. 4, pp. 321–333).
Rasch, G. (1993). *Probabilistic models for some intelligence and attainment tests*. ERIC.
Razzaq, L., Feng, M., Nuzzo-Jones, G., Heffernan, N., Koedinger, K., Junker, B., et al. (2005). The Assistment project: Blending assessment and assisting. In *International conference on artificial intelligence in education* (pp. 555–562).
Reckase, M. D. (2009). Multidimensional item response theory models. In *Multidimensional item response theory* (pp. 79–112). Springer.
Reckase, M. D., & McKinley, R. L. (1991). The discriminating power of items that measure more than one dimension. *Applied Psychological Measurement, 15*(4), 361–373.
Robitzsch, A., Kiefer, T., George, A. C., & Uenlue, A. (2015). *CDM: Cognitive diagnosis modeling*. R package version 4.5-0 http://CRAN.R-project.org/package=CDM.
Romero, C., & Ventura, S. (2020). Educational data mining and learning analytics: An updated survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 10*(3), e1355.
Rupp, A. (2007). The answer is in the question: A guide for investigating the theoretical potentials and practical limitations of cognitive psychometric models. *International Journal of Testing, 7*, 95–125.
Rupp, A. A., & Mislevy, R. J. (2007). Cognitive foundations of structured item response models. *Cognitive Diagnostic Assessment for Education: Theory and Applications*, 205–241.
Shute, V. J. (2008). Focus on formative feedback. *Review of Educational Research, 78*(1), 153–189.
Skaggs, G., Wilkins, J. L., & Hein, S. F. (2016). Grain size and parameter recovery with TIMSS and the general diagnostic model. *International Journal of Testing, 16*(4), 310–330.
Slavuj, V., Kovačić, B., & Jugo, I. (2015). Intelligent tutoring systems for language learning. In *2015 38th international convention on information and communication technology, electronics and microelectronics (MIPRO)* (pp. 814–819). IEEE.
Su, Y., Liu, Q., Liu, Q., Huang, Z., Yin, Y., Chen, E., et al. (2018). Exercise-enhanced sequential modeling for student performance prediction. In *Thirty-second AAAI conference on artificial intelligence*.
Swartz, M. L., & Yazdani, M. (2012). *Intelligent tutoring systems for foreign language learning: The bridge to international communication* (Vol. 80). Springer Science & Business Media.
Sykes, E. R., & Franek, F. (2003). A prototype for an intelligent tutoring system for students learning to program in Java(TM). In *Proceedings of the IASTED international conference on computers and advanced technology in education* (pp. 78–83). Citeseer.
Tatsuoka, K. K. (1983). Rule space: An approach for dealing with misconceptions based on item response theory. *Journal of Educational Measurement, 20*(4), 345–354.
Templin, J. L., & Henson, R. A. (2006). Measurement of psychological disorders using cognitive diagnosis models. *Psychological Methods, 11*(3), 287.
Templin, J., Henson, R. A., et al. (2010). *Diagnostic measurement: Theory, methods, and applications*. Guilford Press.
Templin, J., & Hoffman, L. (2013). Obtaining diagnostic classification model estimates using mplus. *Educational Measurement: Issues and Practice, 32*(2), 37–50.
Tsai, F.-H., Tsai, C.-C., & Lin, K.-Y. (2015). The evaluation of different gaming modes and feedback types on game-based formative assessment in an online learning environment. *Computers & Education, 81*, 259–269.
Vie, J.-J., & Kashima, H. (2019). Knowledge tracing machines: Factorization machines for knowledge tracing. In *Proceedings of the 33th AAAI conference on artificial intelligence* (pp. 750–757). https://arxiv.org/abs/1811.03388.
Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3156–3164).
Wang, T.-H. (2007). What strategies are effective for formative assessment in an e-learning environment? *Journal of Computer Assisted Learning, 23*(3), 171–186.
Wang, T., Ma, F., & Gao, J. (2019). Deep hierarchical knowledge tracing. In *Proceedings of the 12th international conference on educational data mining*.
Weragama, D., & Reye, J. (2013). The PHP intelligent tutoring system. In *International conference on artificial intelligence in education* (pp. 583–586). Springer.
Wertsch, J. V. (1984). The zone of proximal development: Some conceptual issues. *New Directions for Child and Adolescent Development, 1984*(23), 7–18.
Wilson, K. H., Karklin, Y., Han, B., & Ekanadham, C. (2016). Back to the Basics: Bayesian extensions of IRT outperform neural networks for proficiency estimation. In *Proceedings of the international conference on educational data mining*.
Xiong, X., Zhao, S., Van Inwegen, E. G., & Beck, J. E. (2016). Going deeper with deep knowledge tracing. In *International conference on educational data mining*.
Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., et al. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning* (pp. 2048–2057).
Xu, X., & von Davier, M. (2008). Fitting the structured general diagnostic model to NAEP data. *ETS Research Report Series, 2008*(1), i–18.
Y. Yang, J. Shen, Y. Qu, Y. Liu, K. Wang, Y. Zhu, W. Zhang, Y. Yu, GIKT: A graph-based interaction model for knowledge tracing, machine learning and knowledge discovery in databases.
S. Yang, M. Zhu, J. Hou, X. Lu, Deep knowledge tracing with convolutions, arXiv preprint arXiv:2008.01169.
Yeung, C. K. (2019). Deep-IRT: Make deep learning based knowledge tracing explainable using item response theory. In *Proceedings of the 12th international conference on educational data mining* (pp. 683–686).
Yudelson, M. V., Koedinger, K. R., & Gordon, G. J. (2013). Individualized bayesian knowledge tracing models. In *International conference on artificial intelligence in education* (pp. 171–180). Springer.
Yu, J., Luo, G., Xiao, T., Zhong, Q., Wang, Y., Feng, W., et al. (2020). MOOCCube: A large-scale data repository for NLP applications in MOOCs. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 3135–3142).
Yu, H., Miao, C., Leung, C., & White, T. J. (2017). Towards AI-powered personalization in MOOC learning. *Npj Science of Learning, 2*(1), 1–5.
Zhang, J., Shi, X., King, I., & Yeung, D.-Y. (2017). Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th international conference on world wide web* (pp. 765–774).