Multi-Task Modeling of Student Knowledge and Behavior

Siqian Zhao
Department of Computer Science
University at Albany - SUNY
Albany, New York, USA
szhao2@albany.edu

Sherry Sahebi
Department of Computer Science
University at Albany - SUNY
Albany, New York, USA
ssahebi@albany.edu

ABSTRACT

Knowledge Tracing (KT) and Behavior Modeling (BM) are essential mining and discovery problems in education. KT models student knowledge based on prior performance with learning materials, while BM focuses on patterns such as student preferences, engagement, and procrastination. Traditional research in these areas focuses on each task individually, thereby overlooking their interconnections. However, recent research on multi-activity knowledge tracing suggests that student preferences for learning materials are key to understanding student learning. In this paper, we propose a novel multi-task model, the Multi-Task Student Knowledge and Behavior Model (KTBM), which combines KT and BM to improve both performance and interoperability. KTBM includes a multi-activity KT component and a preference behavior component while enabling robust information transfer between them. We conceptualize this approach as a multi-task learning problem with two objectives: predicting students' performance and their choices concerning learning material types. To address this dual-objective challenge, we employ a Pareto multi-task learning optimization algorithm. Our extensive experiments on three real-world datasets show that KTBM significantly enhances both KT and BM performance, demonstrating improvement across various settings and providing interpretable results.

CCS CONCEPTS

• Human-centered computing → User models; **• Computing methodologies → Multi-task learning**; **• Computer systems organization → Neural networks**.

KEYWORDS

Multi-task learning, Multi-Objective, Pareto learning, Knowledge tracing, Student behavior, Multi-activity

ACM Reference Format:
Siqian Zhao and Sherry Sahebi. 2024. Multi-Task Modeling of Student Knowledge and Behavior. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24), October 21–25, 2024, Boise, ID, USA*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3627673.3679823

This work is licensed under a Creative Commons Attribution International 4.0 License.

CIKM '24, October 21–25, 2024, Boise, ID, USA
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0436-9/24/10
https://doi.org/10.1145/3627673.3679823

1 INTRODUCTION

The expansion of online educational systems and the growing demand for online education have resulted in large-scale student log datasets, providing an opportunity for improved learner modeling in the education domain. Examples of such learner modeling include the tasks of Student Knowledge Tracing (KT) and Behavior Modeling (BM). Student knowledge tracing [2, 17] aims to model student knowledge based on their past performance in learning materials, with objectives like predicting future performance. Typically modeled as supervised sequence learning problems, state-of-the-art KT models address challenges such as guessing or slipping noise in student data [8, 22], modeling both knowledge acquisition and forgetting of learned concepts [1, 11, 44], continuous modeling over long sequences [12, 58], and learning from multiple types of learning materials [4, 68, 71]. In contrast to KT, behavior modeling focuses on tasks such as representing student engagement during learning [30], detecting procrastination [35, 55, 61], and modeling student choice and preference for future learning materials [32, 46, 60] based on their history.

While prior studies have suggested a relationship between student behavior and knowledge gain [23, 41, 69, 70], KT and BM tasks have traditionally been modeled separately. Consequently, the literature is limited in simultaneously modeling both tasks to leverage this relationship. Specifically, the associations between students' choice of learning materials and their knowledge have been under-investigated. Student knowledge may be influenced by their preference for learning materials. For example, a student's knowledge gain in a topic may differ if they read a book chapter instead of watching a video lecture. Conversely, a student's preference for what learning material to study may be affected by their knowledge. A student, who feels knowledgeable about a topic, may choose to solve an assignment rather than read the book. This bidirectional association is underrepresented in current KT and BM literature. Challenges in simultaneously modeling these two problems include efficiently representing knowledge and preference behavior states, robustly depicting information transfer between the two tasks, and balancing their respective objectives for mutual benefit.

We propose a multi-task learning model that combines knowledge tracing and preference behavior modeling while addressing the above challenges. Our proposed Multi-Task Student Knowledge and Behavior Model (KTBM) explicitly represents separate dynamic student knowledge and behavior states by providing a flexible adaptation of deep multi-type KT and Long Short-Term Memory (LSTM) [33] architectures. By incorporating the previous behavior state as an input to the student knowledge component and the current knowledge state as an input to the preference behavior component, KTBM models bidirectional relations between the two components and ensures robust information transfer. Finally, KTBM employs multi-objective optimization to learn the optimal solution for both KT and BM tasks without compromising one for the other.

In our experiments, we show that KTBM significantly improves both KT and BM task performance in three real-world datasets. This improvement is observed across different student groups, and the associations between these tasks can be interpreted by visualizing KTBM's knowledge and behavior estimations. The main contributions of our paper are listed below:

• We introduce KTBM for joint modeling of student knowledge and preference behavior as a multi-task learning problem.
• We propose a robust architecture for information transfer between the KT and BM tasks.
• We employ a multi-objective optimization that ensures a balanced, no-compromise approach between the KT and BM objectives.
• We demonstrate in our experiments that our model significantly improves both KT and BM performances across different groups in three datasets, and showcase the interpretability of our model through a case study.

2 RELATED WORK

Knowledge Tracing The KT task aims to estimate a student's knowledge state at each learning step as they interact with learning materials. Traditional KT approaches rely on predefined associations between learning materials and knowledge concepts [9, 18, 24, 48]. Traditional methods like Bayesian Knowledge Tracing (BKT) model concept mastery using binary variables, while regression-based approaches consider student ability and question difficulty [9, 18, 24, 48]. Newer models such as Deep Knowledge Tracing (DKT) use recurrent neural networks (RNNs), and Dynamic Key-Value Memory Network (DKVMN) utilizes memory-augmented neural networks (MANNs) to better model knowledge states [49, 66]. Attention mechanisms in models like Self-Attentive Knowledge Tracing (SAKT) and Attentive Knowledge Tracing (AKT) address interdependencies and long-term dependencies, enhancing understanding of student learning patterns [29, 45].

These models primarily focus on a single type of learning activity and neglect multiple types. While some incorporate non-assessed activities as additional features [13, 67], they don't explicitly track knowledge states during these interactions. A few approaches like MA-Elo, MA-FM, MVKM, DMKT, TAMKOT, and GMKT model knowledge from various activities. For instance, MA-Elo and MA-FM use predefined mappings to capture knowledge states [3, 4], MVKM employs multi-view tensor factorization [71], and DMKT extends DKVMN to include non-assessed activities [59]. TAMKOT utilizes an LSTM-based layer for knowledge transfer dynamics [72], and GMKT combines MANNs with Graph Neural Networks (GNNs) to enhance knowledge modeling through non-assessed activities, uniquely addressing both student knowledge and material type preferences [68]. None of them model student behavior and knowledge as a dual-objective multi-task learning problem, with separate states for each objective allowing for information exchange.

Behavior Modeling Student behavior modeling focuses on understanding students' behavior patterns during the learning process, including their habits, preferences, and behaviors [15, 28, 40, 51, 57]. Researchers investigate gamified learning, dropout rates, retention, and participation frequency [19, 36, 40, 56, 64, 65]. Motivation significantly influences course completion [42, 43], with gamified learning boosting engagement based on motivation type [7]. Studies show spaced practice enhances retention [47], and early success increases retention rates [20]. Sequential patterns in behavior, such as e-textbook navigation and question repetition, have been explored [34, 40, 63]. There's a strong correlation between learning behaviors and test scores, with frequent participation leading to higher grades [56]. However, no methods explicitly model both student behavior and knowledge simultaneously.

Moreover, DP-MTL [5] combines KT with modeling student behavior in multiple-choice questions, using option and user embeddings in a sequential multilayer perceptron to predict performance and option choice. This is a multi-task learning approach and the only method modeling both student knowledge and behavior this way. However, DP-MTL is limited to assessed methods and cannot leverage different learning material types to model student preferences. It also doesn't separate behavior and knowledge modeling into two components or allow information transfer between them.

3 PROBLEM FORMULATION

We aim to jointly model student preference behavior and knowledge by predicting the type of learning material students choose for their next activity and their performance on it. Without loss of generality, assume there are two types of materials: N_q questions (assessed) and N_l video lectures (non-assessed). We represent a student's entire trajectory of multi-type learning activities as a sequence of tuples $\{\langle i_1, z_1\rangle, \ldots, \langle i_t, z_t\rangle\}$, where each tuple $\langle i_t, z_t\rangle$ represents a specific student activity at time step t. In this context, $z_t \in 0, 1$ serves as a binary indicator that identifies the type of material interacted with at time t, with 0 denoting assessed material (questions) and 1 indicating non-assessed material (video lectures). Furthermore, i_t is defined as $\begin{cases}(q_t, r_t) & \text{if } z_t = 0\\ l_t & \text{if } z_t = 1\end{cases}$, where (q_t, r_t) represents an assessed activity, with q_t being the question interacted with and r_t being the student's response to that question at time step t. Conversely, l_t denotes a non-assessed activity at time step t involving the video lecture l_t. Given a student's historical trajectory of activities $\{\langle i_1, z_1\rangle, \ldots, \langle i_t, z_t\rangle\}$, we aim to predict the type of material z_{t+1} the student is likely to choose at the next time step $t + 1$, as well as the student's upcoming performance r_{t+1} on the question q_{t+1} if $z_{t+1} = 0$.

4 MULTI-TASK STUDENT KNOWLEDGE AND BEHAVIOR MODEL (KTBM)

Modeling student knowledge and behavior simultaneously requires efficiently capturing knowledge and behavioral preference states, along with effective information transfer between them, to refine and strengthen the model. Therefore, we formulate this as a multi-task learning problem. We propose KTBM, a multi-task learning model that introduces two interconnected components: one for KT and another for BM, allowing information transfer between them. The KT component is a multi-activity transition-aware memory-augmented neural network (MANN [52]) that captures student

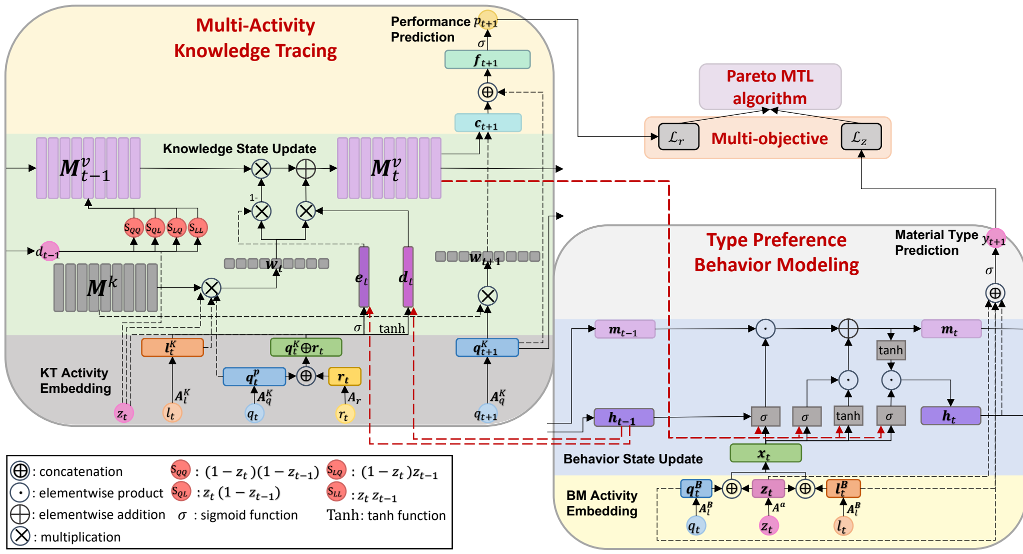

Figure 1: The architecture of the KTBM. Solid and dashed lines indicate the same connections but clarify overlapping lines.

knowledge acquisition from both assessed and non-assessed activities. The BM component learns student preferences for different material types by refining the LSTM [33] architecture. Knowledge transfer between the two components is facilitated through their hidden states. To address this multi-task learning problem, we formulate two objective functions and utilize Pareto MTL optimization to provide a balanced solution between the two objectives. An overview of KTBM's architecture is presented in Figure 1.

4.1 Multi-Activity Knowledge Tracing

The multi-activity KT component includes an embedding layer, a knowledge modeling layer, and a performance prediction layer. It connects with the BM component in the knowledge modeling layer.

4.1.1 ***KT Activity Embedding.*** First, KTBM constructs embedding vectors for each learning activity $\langle i_t, z_t \rangle$, which serve as inputs to effectively capture student knowledge, by leveraging the latent representations of the learning material (q_t and l_t) and student performance (r_t). We employ two underlying latent embedding matrices: $\boldsymbol{A}_q^K \in \mathbb{R}^{N_Q \times d_q^K}$ and $\boldsymbol{A}_l^K \in \mathbb{R}^{N_L \times d_l^K}$, which map all questions and video lectures into their respective latent spaces. Here, d_q^K and d_l^K specify the respective embeddings sizes. To represent student performance r_t within assessed activities, we map it into a higher-dimensional latent space. We use an embedding matrix $\boldsymbol{A}_r \in \mathbb{R}^{2 \times d_r}$ for mapping the binary student performance (e.g., success or failure), where d_r indicates the performance embeddings size. For numerical performance (e.g., exam scores), we apply a linear transformation $f(r_t) = r_t \boldsymbol{A}_r$ to project r_t into a higher-dimensional space, where $\boldsymbol{A}_r \in \mathbb{R}^{d_r}$.

4.1.2 ***Knowledge State Update.*** For tracing student knowledge through various types of learning activities while also modeling the impact of preference behavior on student knowledge acquisition, KTBM takes the embeddings of learning activities and the hidden preference behavior state $\boldsymbol{h}_{t-1}$ (Eq. 13) as inputs. We propose a transition-aware MANN to accurately capture the dynamic student knowledge state. We employ a static key matrix $\boldsymbol{M}^c \in \mathbb{R}^{N \times d_c}$ to represent N latent concepts that are characterized by d_c latent features. Additionally, we use a dynamic value matrix $\boldsymbol{M}_t^v \in \mathbb{R}^{N \times d_v}$ to track the student's knowledge mastery of these concepts over time t in d_v-size memory slot.

To update student knowledge at each time step t, KTBM first computes the correlation between the interacted learning material (either q_t or l_t) and each of the N latent concepts and obtains the attention weight vector $\boldsymbol{w}_t$. This specifies how the knowledge of the involved concept in $\boldsymbol{M}_t^v$ should be updated from the activity. $\boldsymbol{w}_t$ is calculated using material embeddings ($\boldsymbol{q}_t^K$ or $\boldsymbol{l}_t^K$ from $\boldsymbol{A}_q^K$ or $\boldsymbol{A}_l^K$) and the static key matrix $\boldsymbol{M}^c$ as follows:

$$w_t(i) = softmax(\big[(1 - z_t) \cdot \boldsymbol{R}_q^\top \boldsymbol{q}_t^K + z_t \cdot \boldsymbol{R}_l^\top \boldsymbol{l}_t^K\big]^\top \boldsymbol{M}^c(i)) \quad (1)$$

where $w_t(i)$ is the i-th element in the attention weight vector $\boldsymbol{w}_t \in \mathbb{R}^N$, and the softmax function is defined as $softmax(m_i) = e^{m_i} / \sum_j e^{m_j}$. $\boldsymbol{R}_q \in \mathbb{R}^{d_q^K \times d_c}$ and $\boldsymbol{R}_l \in \mathbb{R}^{d_l^K \times d_c}$ are mapping matrices that project the question and lecture activity embeddings to the concept feature space of $\boldsymbol{M}^k$. The terms $(1 - z_t)$ and z_t indicate which matrix should be used to map activity embeddings.

Similar to the transition-aware multi-activity KT methods [68, 72], KTBM uses a set of binary indicators to activate the corresponding knowledge transfer weights when students transition from one activity type (e.g., questions) to another (e.g., video lectures). Given two types of materials (represented by z_t), the following binary indicators are defined to indicate the four possible transitions at each time t:

$$\begin{aligned} s_{QQ} &= (1 - z_t)(1 - z_{t-1}) \qquad & s_{QL} &= z_t(1 - z_{t-1}) \\ s_{LQ} &= (1 - z_t)z_{t-1} \qquad & s_{LL} &= z_t z_{t-1} \end{aligned} \quad (2)$$

At each time step, only one transition occurs, meaning that only one of the transition indicators is active (equals 1). These transition indicators are used to update the student knowledge state $\boldsymbol{M}_t^v$ using the corresponding transition-specific weight matrices $\boldsymbol{T}_{**}$. For that, an *erase-followed-by-add* mechanism is employed. Which involves erasing previous redundant information before adding new information to $\boldsymbol{M}_t^v$. The updates are based on the student's activities at time t, their previous knowledge state $\boldsymbol{M}_{t-1}^v$, and their previous preference behavior state $\boldsymbol{h}_{t-1}$ from the BM component:

Erase:

$$\boldsymbol{e}_t = \sigma\big((1-z_t)\cdot \boldsymbol{E}_q^\top[\boldsymbol{q}_t^K \oplus \boldsymbol{r}_t] + z_t \cdot \boldsymbol{E}_l^\top \boldsymbol{l}_t^K + \boldsymbol{E}_b^\top \boldsymbol{h}_{t-1} + \boldsymbol{b}_e\big) \quad (3)$$

$$\begin{aligned}\tilde{\boldsymbol{M}}_t^v(i) =& \big[s_{QQ}\cdot \boldsymbol{T}_{QQ}\boldsymbol{M}_{t-1}^v + s_{LL}\cdot \boldsymbol{T}_{LL}\boldsymbol{M}_{t-1}^v + \\ & s_{QL}\cdot \boldsymbol{T}_{QL}\boldsymbol{M}_{t-1}^v + s_{LQ}\cdot \boldsymbol{T}_{LQ}\boldsymbol{M}_{t-1}^v\big](i)\cdot\big[\boldsymbol{1} - w_t(i)\boldsymbol{e}_t\big]\end{aligned} \quad (4)$$

Add:

$$\boldsymbol{d}_t = Tanh\big((1-z_t)\cdot \boldsymbol{D}_q^\top[\boldsymbol{q}_t^K \oplus \boldsymbol{r}_t] + z_t \cdot \boldsymbol{D}_l^\top \boldsymbol{l}_t^K + \boldsymbol{D}_b^\top \boldsymbol{h}_{t-1} + \boldsymbol{b}_d\big) \quad (5)$$

$$\boldsymbol{M}_t^v(i) = \tilde{\boldsymbol{M}}_t^v(i) + w_t(i)\boldsymbol{d}_t \quad (6)$$

where $\oplus$ denotes the concatenation operator, σ and $Tanh$ refer to the Sigmoid and Tanh activation functions, respectively. The erase vector $\boldsymbol{e}_t \in [0,1]^{d_v}$ (Eq. 3) is designed to remove redundant knowledge information from $\boldsymbol{M}_{t-1}^v$. The add vector $\boldsymbol{d}_t \in \boldsymbol{R}^{d_v}$(Eq. 5) captures the new knowledge that the student acquires at time t. Matrices $\boldsymbol{E}_q, \boldsymbol{D}_q \in \mathbb{R}^{(d_q^K+d_r)\times d_v}$, and $\boldsymbol{E}_l, \boldsymbol{D}_l \in \mathbb{R}^{d_l^K \times d_v}$, are for mapping the activity embedding to the concept feature space. $\boldsymbol{E}_b$ and $\boldsymbol{D}_b \in \mathbb{R}^{d_h+d_v}$ are for mapping the preference behavior state to the concept feature space. $\boldsymbol{b}_e$ and $\boldsymbol{b}_d \in \mathbb{R}^{d_v}$ are bias terms.

The student knowledge is captured via two mechanisms in this component. Once, implicitly by using the transition indicators and their associated transfer matrices $\boldsymbol{T}{*}{*}$, which influence how student knowledge is transferred from previous time steps in different ways. Another time, by explicitly incorporating the student preference behavior state from the BM component: for both the erase and add vectors, we use the mapping matrices $\boldsymbol{E}_b, \boldsymbol{D}_b$ to incorporate information from behavior state $\boldsymbol{h}_{t-1}$, to influence the student's knowledge. $\tilde{\boldsymbol{M}}_t^v(i)$ and $\boldsymbol{M}_t^v(i)$ (Eq. 4 and 6) indicate the i-th knowledge slot of $\boldsymbol{M}_t^v$ after the erasing and adding process. To this end, our KT component can accurately capture student knowledge from multiple types of activities, model the impact of student preference behavior on knowledge, and learn the different knowledge transfers among various activity types.

4.1.3 ***Student Performance Prediction.*** We predict a student's performance at the next time $t+1$ for a given question q_{t+1} based on their mastery knowledge of q_{t+1}'s concepts.

$$w_{t+1}(i) = softmax([\boldsymbol{R}_q^\top \boldsymbol{q}_{t+1}^p]^\top \boldsymbol{M}^k(i)) \quad (7)$$

$$\boldsymbol{c}_{t+1} = \sum_{i=1}^N w_{t+1}(i)\big[(1-z_t)\cdot \boldsymbol{M}_t^v \boldsymbol{T}_{QQ} + z_t \cdot \boldsymbol{M}_t^v \boldsymbol{T}_{LQ}\big](i) \quad (8)$$

$$\boldsymbol{f}_{t+1} = Tanh(\boldsymbol{W}_f^\top[\boldsymbol{c}_{t+1} \oplus \boldsymbol{q}_{t+1}] + \boldsymbol{b}_f) \quad (9)$$

First, we compute the attention weight vector $\boldsymbol{w}_{t+1}$ (Eq. 7) to determine the correlation between question q_{t+1} and each of the N latent concepts. Then, KTBM summarizes the student's knowledge state regarding question q_{t+1} in the read content $\boldsymbol{c}_{t+1}$ (Eq. 8) by taking the weighted sum of all memory slots in $\boldsymbol{M}_t^v$ using $\boldsymbol{w}_{t+1}$. Next, $\boldsymbol{c}_{t+1}$ is concatenated with the embedding vector of the next question $\boldsymbol{q}_{t+1}$ and passed it through a fully connected lay with a $Tanh$ activation function to obtain the summary vector $\boldsymbol{f}_{t+1}$ (Eq. 9), which represent the summarized student knowledge of q_{t+1}. Here, $\boldsymbol{W}_f \in \mathbb{R}^{(d_v+d_q^K)\times d_f}$ and $\boldsymbol{b}_f \in \mathbb{R}^{d_f}$ are the weight matrix and bias term, respectively, with d_f as the summary vector size. Finally, a fully connected layer with a *Sigmoid* activation function is applied to $\boldsymbol{f}_{t+1}$ to predict the student's performance p_{t+1}:

$$p_{t+1} = \sigma(\boldsymbol{W}_p^\top \boldsymbol{f}_{t+1} + b_p) \quad (10)$$

where p_{t+1} is the probability of the student correctly answering the next question q_{t+1}. The terms $\boldsymbol{W}_p \in \mathbb{R}^{d_s \times 1}$ and $b_p \in \mathbb{R}$ are the weight matrix and bias.

4.2 Type Preference Behavior Modeling

The BM component aims to model student behavior, primarily by examining their preferences for different types of materials, while also considering how their knowledge influences these preferences.

4.2.1 ***BM Activity Embedding.*** KTBM designs different behavior embedding matrices than those employed in the KT component. Specifically, $\boldsymbol{A}_q^B \in \mathbb{R}^{N_Q \times d_q^B}$ and $\boldsymbol{A}_l^B \in \mathbb{R}^{N_L \times d_l^B}$ are used as the two embedding matrices to map questions and video lectures into a latent behavior feature space for the BM component. These matrices vary in size from the KT component's embedding matrices and primarily capture various knowledge concepts to understand student behavior preferences. Additionally, KTBM employs $\boldsymbol{A}_z \in \mathbb{R}^{2\times d_z}$ to map the two learning material types into a latent space for BM.

4.2.2 ***Behavior State Update.*** This layer is a refined LSTM variant that can process various types of learning activities and leverage information from the dynamic value matrix $\boldsymbol{M}_t^v$ (Eq. 6) to effectively incorporate the influence of the student's knowledge on their behavior. At each time step t, KTBM uses the hidden vector $\boldsymbol{h}_t \in \mathbb{R}^{d_h}$ to track the state of student preference behavior, where d_h represents the hidden dimension size, as follows:

$$\boldsymbol{x}_t = (1-z_t)\cdot \boldsymbol{X}_q^\top[\boldsymbol{q}_t^B \oplus z_t] + z_t \cdot \boldsymbol{X}_l^\top[\boldsymbol{l}_t^B \oplus z_t] \quad (11)$$

$$\boldsymbol{K}_t = \boldsymbol{W}_k^\top \boldsymbol{M}_t^v + \boldsymbol{b}_k \quad (12)$$

$$\boldsymbol{h}_t = LSTM(\boldsymbol{h}_{t-1}^b, \boldsymbol{K}^t, \boldsymbol{x}_t) \quad (13)$$

First, $\boldsymbol{x}_t$ (Eq. 11) is computed to represent the combined representation of question and lecture activities into the same dimensional space d_x. Here, $\boldsymbol{q}_t^B$, $\boldsymbol{l}_t^B$, and $\boldsymbol{z}_t$ are the embeddings for question, video lecture, and material type that are obtained from $\boldsymbol{A}_q^B$, $\boldsymbol{A}_l^B$ and $\boldsymbol{A}_z$. $\boldsymbol{X}_q \in \mathbb{R}^{(d_q^B+d_z)\times d_x}$ and $\boldsymbol{X}_l \in \mathbb{R}^{(d_l^B+d_z)\times d_x}$ are used to map question and lecture activities. Moreover, we adapt the knowledge state $\boldsymbol{M}_t^v$ at time t to update the student preference behavior $\boldsymbol{h}_t$ for the same time step. We calculate $\boldsymbol{K}_t$ (Eq. 12) to summarize the student's knowledge for each concept, converting the knowledge value matrix $\boldsymbol{M}_t^v$ into a vector that can be used as input for LSTM. $\boldsymbol{W}_k \in \mathbb{R}^{d^v \times d_c}$ and $\boldsymbol{b}_k \in \mathbb{R}^{d_c}$ are weight matrix and bias. Finally, KTBM uses the behavior state $\boldsymbol{h}_{t-1}$ from the previous time step $t-1$, the representation of activity $\boldsymbol{x}_t$, and the adapted knowledge state $\boldsymbol{K}_t$ to compute the input gate, forget gate, candidate memory cell, and output gate, and accordingly updates $\boldsymbol{h}_t$ (Eq. 13).

4.2.3 ***Material Type Prediction.*** We use the student's hidden preference behavior state $\boldsymbol{h}_t$ to predict the material type at time step $t+1$, as follows:

$$\boldsymbol{s}_t = (1-z_t)\cdot \boldsymbol{S}_q^\top[\boldsymbol{q}_t^B \oplus z_t \oplus \boldsymbol{h}_t] + z_t \cdot \boldsymbol{S}_l^\top[\boldsymbol{l}_t^B \oplus z_t \oplus \boldsymbol{h}_t] + \boldsymbol{b}_s \quad (14)$$

$$y_{t+1} = \sigma(\boldsymbol{W_y}^\top \boldsymbol{s}_t + b_y) \quad (15)$$

Here, $\boldsymbol{s}_t \in \mathbb{R}^{d_s}$ (Eq. 14) is calculated to summarize the student preference behavior state according to the learning material in activity at time t, where d_s is $\boldsymbol{s}_t$'s dimension size. Then, y_{t+1} is calculated using $\boldsymbol{s}_t$, representing the probability that the next learning material type to be interacted with is a video lecture. $\boldsymbol{S}_q \in \mathbb{R}^{(d_q^B+d_z+d_h)\times d_s}$, $\boldsymbol{S}_l \in \mathbb{R}^{(d_l^B+d_z+d_h)\times d_s}$, $\boldsymbol{W}_y \in \mathbb{R}^{d_s\times 1}$, $\boldsymbol{b}_s \in \mathbb{R}^{d_s}$, and $\boldsymbol{b}_y \in \mathbb{R}$ are the corresponding weight matrices and bias terms.

4.3 Multi-Objective and Pareto Optimization

In the previous sections, we conceptualized student knowledge tracing and preference behavior modeling as two tasks. Since student knowledge and behavior are not directly observed and are difficult to quantify, we formulate two objectives for evaluating this multi-task learning challenge: (1) $\mathcal{L}_r$ for predicting student performance, and (2) $\mathcal{L}_z$ for predicting the learning material type that students will choose to learn from. These objectives are computed using binary cross-entropy losses, which compare the actual and predicted student performance (r_t and p_t), as well as the actual and predicted types of material (z_t and y_t), at every time step. The loss functions are defined as follows:

$$\mathcal{L}_r = -\sum_t \left(r_t \log p_t + (1-r_t)\log(1-p_t)\right) \tag{16}$$

$$\mathcal{L}_z = -\sum_t \left(z_t \log y_t + (1-z_t)\log(1-y_t)\right) \tag{17}$$

This dual-objective problem can be addressed by minimizing a combination of $\mathcal{L}_r$ and $\mathcal{L}_d$, balancing the student performance and material type objectives. However, effectively combining these objectives and determining the right trade-off is challenging and time-consuming [38]. Research in multi-objective optimization has developed strategies for identifying Pareto optimal solutions to address multi-task learning. These solutions represent trade-offs where no single objective can be improved without compromising another [21, 26, 27, 53, 54, 73]. However, due to the infinite number of Pareto optimal solutions, a single solution may not meet practitioners' needs [37, 38]. The Pareto MTL algorithm [38] addresses this problem by identifying a set of representative solutions using dividing vectors $\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_m$, dividing the problem into sub-problems and providing a well-rounded set of Pareto solutions. By employing the Pareto MTL algorithm to our problem, we obtain such a no-compromise set of Pareto solutions, allowing for optimal selection for predicting student performance and material type.

4.3.1 ***Model Complexity***. The time complexity of KTBM for a student activity sequence of length L_s is $O\Big(L_s \cdot \Big(N \cdot \max(d_q^K, d_l^K) \cdot d_c + N\cdot d_c + N\cdot d_v \cdot \max(d_q^K + d_r^K, d_l^K) + d_h \cdot \big(\max(d_q^B, d_l^B) + d_h\big) + d_h \cdot d_v\Big)\Big)$. Moreover, the time cost also depends on the number of dividing vectors set in the experiments.

5 EXPERIMENTS

We conduct three sets of experiments on three real-world datasets to evaluate our proposed method, KTBM. First, we compare KTBM's predictive ability with baseline methods in student performance and learning material type preference prediction tasks, including ablation studies to each model component. Second, we conduct student group analysis. Lastly, we visualize the states of learned students and behavior knowledge. Our code and sample data are available at GitHub [1].

5.1 Datasets

We use three real-world datasets for our experiments. The general statistics of each dataset can be found in Table 1. **EdNet**[2] [14]: This publicly available and anonymized dataset comes from Santa[3], a multi-platform AI tutoring service for Korean students preparing for the TOEIC[4] English test. EdNet collects a range of student learning activities across different material types and provides data in four distinct levels with varying extents in a consistent and organized manner. In our research, we use a preprocessed dataset from [68, 72], selecting data from the third level, which focuses on students' activities involving questions (assessed) and their associated question explanations (non-assessed). Each question is a multiple-choice item accompanied by an explanation. **Junyi**[5] [16, 50]: This dataset is another publicly available and anonymized dataset sourced from the Chinese e-learning platform Junyi Academy[6], designed to teach children math. The dataset covers eight math areas with varying levels of difficulty. Students begin at the easiest level and progress to more challenging levels as they master each area. We utilize the preprocessed data introduced in [10]. In this dataset, there are two types of student activities: solving problems (assessed) and reading problem hints (non-assessed). Junyi provides various formats of math problems, including fill-in-the-blank, judgmental, and multiple-choice questions. Each problem may be associated with one or multiple hints. **MORF**[7] [6]: This anonymized dataset is from an online course available via the MOOC Replication Framework (MORF) on Coursera[8]. The course, titled "Educational Data Mining", is divided into several modules, each focusing on a specific topic, such as "classification". Each module is designed to be completed in one week. During this week, students are required to watch five to seven video lectures and complete one assignment. Each assignment typically includes multiple questions in various formats. The dataset includes various student activities. For our study, we focus on two specific activities: watching video lectures (non-assessed) and completing assignments (assessed). The data is coarse-grained, recording entire assignment submissions rather than individual questions, with the overall score of each submission serving as the response in our experiments.

5.2 Baseline Methods

5.2.1 ***Student Performance Prediction Baselines***. We assess KTBM's capability in modeling student knowledge for predicting future student performance by comparing it with a total of 16 baselines. This comparison includes six assessed-only supervised KT models and four state-of-the-art multi-activity KT models (one of which is semi-supervised). To ensure a fair comparison, we also extend a multi-layer perceptron (MLP) and the six assessed-only

[1]https://github.com/persai-lab/2024-CIKM-KTBM
[2]https://github.com/riiid/ednet
[3]https://www.aitutorsanta.com/
[4]https://www.ets.org/toeic
[5]https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=1275
[6]https://official.junyiacademy.org/
[7]https://educational-technology-collective.github.io/morf/
[8]https://www.coursera.org/

Table 1: Descriptive statistics of 3 datasets.

Dataset	#Users	#Questions	Question Activities	Question Responses Mean	Question Responses STD	#Correct Question Responses	#Incorrect Question Responses	#Non-assessed materials	#Non-assessed Activities
MORF	686	10	12031	0.7763	0.2507	N/A	N/A	52	41980
EdNet	1000	11249	200931	0.5910	0.2417	118747	82184	8324	150821
Junyi	2063	3760	290754	0.6660	0.2224	193664	97090	1432	69050

supervised KT models to handle both assessed and non-assessed activities. We label these extended models by appending "+M" to the original model names.

The assessed-only supervised KT baselines are: **DKT** [49] is the first deep learning-based KT model that employs RNNs to model student knowledge gain. **DKVMN** [66] utilizes MANNs for KT, featuring a static key matrix for knowledge concepts and a dynamic value matrix for updating student knowledge. **DeepIRT** [62] is an extension of DKVMN that incorporates the one-parameter logistic item response theory (1PL-IRT) to mitigate overfitting. **SAKT** [45] uses a self-attentive model for KT to capture the relationships between activities at different time steps. **SAINT** [13] is a transformer-based method that applies deep self-attentive layers to separately model questions and responses. **AKT** [29] is a context-aware model that utilizes a monotonic attention mechanism to aggregate past student performances relevant to the current question.

The extended assessed-only baseline methods are: **MLP+M** [31] is a simple MLP that takes a student's three most recent assessed activities along with three non-assessed activities as input to predict student performance. **DKT+M** [67] and DKVMN+M: are extensions of DKT and DKVMN. They concatenate embedding vectors of all non-assessed learning materials the student interacted with between each pair of assessed activities, adding this as an additional feature of input question embedding. **SAINT+M** [13], SAKT+M, and AKT+M are variants of SAINT, SAKT, and AKT. Embeddings of all non-assessed learning materials along with their position encodings between two assessed activities are summarized as an additional feature for these models.

We also compare KTBM with the following multi-activity KT models: **MVKM** [71] focuses on knowledge modeling only and is a multi-view tensor factorization method that models student knowledge acquisition from various types of learning activities. It constructs separate tensors for different types of activities. **DMKT** [59] is based on DKVMN, this model distinguishes between different read and write operations for different types of learning activities. However, it focuses solely on student knowledge modeling. **TAMKOT** [72] is a transition-aware KT model built on LSTM. It learns multiple knowledge transfer matrices to explicitly model knowledge transfer between different activity types. However, it only models student knowledge without considering behavior, its objective function focuses solely on student performance prediction. **GMKT** [68] is another transition-aware method that leverages MANN and incorporates GNNs to enhance the modeling of student knowledge through non-assessed learning activities. It is the only existing method with an objective for predicting the type of materials. However, it does not explicitly model student behavior in a distinct component and does not perform Pareto MTL optimization.

5.2.2 ***Material Type Prediction Baselines.*** To assess the efficacy of KTBM in predicting the types of learning materials, we compare it with four deep sequential baseline models. These models include two standard RNN methods and two variants of multi-activity KT methods. The two RNN baseline methods employed are: **LSTM**[33]: This RNN architecture is renowned for its ability to capture long-term dependencies, making it particularly suitable for tasks that require a comprehensive understanding of entire data sequences. **MANN**[52]: This model enhances RNN with an external memory, which supports the storage and retrieval of information over long sequences. This capability is highly advantageous for tasks requiring prolonged information retention and manipulation.

To facilitate a fair comparison, we utilized learning material embeddings alongside material type embeddings as inputs for the aforementioned models, focusing exclusively on predicting the upcoming type of material. Furthermore, we incorporated two variants of multi-activity knowledge modeling methods: **TAMKOT** We preserved the knowledge modeling architecture and applied an MLP to the learned hidden behavior and knowledge states specifically for predicting the type of learning material. **GMKT** We performed a grid search to find the best trade-off for predicting material type instead of student performance.

5.3 Experiment Setup

5.3.1 ***Evaluation Protocol.*** We employ a 5-fold student-stratified cross-validation to partition the data. In each fold, sequences from 80% of the students form the training set, while the remaining 20% make up the testing set. Additionally, 20% of the training set is used for hyperparameter tuning. For the student performance prediction task, we utilize the Area Under the Curve (AUC) metric to evaluate model performance for both the EdNet and Junyi datasets, as student responses are binary (success or failure). In the MORF dataset, assignments are graded numerically. We normalize the students' assignment scores to a range of [0,1] based on the maximum possible score for each assignment. The Root Mean Squared Error (RMSE) is employed to evaluate prediction performance in the MORF dataset. Since all datasets have two types of materials, we use the AUC metric for the learning material type preference prediction task.

5.3.2 ***Implementation Details.*** We develop KTBM using PyTorch[9]. Following standard practices in sequential data experiments [25, 39, 49, 68], we ensure uniform sequence lengths by truncating or padding them as necessary. The sequence length, denoted as L_s, is treated as a hyperparameter and is tuned using the validation set. All model parameters are initialized with random values drawn from a Gaussian distribution with a mean of 0 and a standard deviation of 0.2. To mitigate the issue of exploding gradients, we employ norm clipping. The Adam optimizer is used for parameter learning. For Pareto MTL optimization, we utilize five evenly distributed dividing vectors $\{(cos(\frac{k\pi}{10}, sin(\frac{k\pi}{10}))|k = 0, 1, ..., 5\}$. A

[9]https://pytorch.org/

coarse-grained grid search is conducted to identify the optimal hyperparameters. The best hyperparameters are reported in Table 2.

Table 2: Learned Best Hyperparameters of KTBM

Dataset	d_q^K	d_l^K	d_r	d_c	d_v	d_s	N	d_q^B	d_l^B	d_z	d_h
EdNet	64	32	32	32	32	32	8	16	16	16	96
Junyi	32	32	32	64	64	32	32	32	32	16	64
MORF	32	8	16	32	32	32	8	8	8	8	32

Table 3: Student Performance Prediction Results. The best and second-best results are in boldface and underlined, respectively. $**$ and $*$ indicate paired t-test $p-value < 0.05$ and $p-value < 0.1$, respectively, compared to KTBM.

Methods	EdNet AUC	Junyi AUC	MORF RMSE
DKT	0.6393^{**}	0.8623^{**}	0.1990^{**}
DKVMN	0.6296^{**}	0.8558^{**}	0.1995^{**}
SAKT	0.6334^{**}	0.8053^{**}	0.1975^{**}
SAINT	0.5205^{**}	0.7951^{**}	0.2190^{**}
AKT	0.6393^{**}	0.8093^{**}	0.2417^{**}
DeepIRT	0.6290^{**}	0.8498^{**}	0.1946^{**}
DKT+M	0.6372^{**}	0.8652^{**}	0.1942^{**}
DKVMN+M	0.6343^{**}	0.8513^{**}	0.2071^{**}
SAKT+M	0.6323^{**}	0.7911^{**}	0.1981^{**}
SAINT+M	0.5491^{**}	0.7741^{**}	0.2007^{**}
AKT+M	0.6404^{**}	0.8099^{**}	0.2226^{**}
MLP+M	0.6102^{**}	0.7290^{**}	0.2428^{**}
MVKM	–	–	0.1936^{**}
DMKT	0.6394^{**}	0.8561^{**}	0.1856^{**}
TAMKOT	0.6786^{*}	0.8745^{**}	0.1857^{**}
GMKT	<u>0.6819</u>	<u>0.8960</u>	<u>0.1802^{*}</u>
KTBM	**0.6838**	**0.8989**	**0.1778**
KTBM-BM	0.6802	0.8928	0.1825

Table 4: Material Type Prediction Results. The best and second-best results are in boldface and underlined, respectively. $**$ and $*$ indicate paired t-test $p-value < 0.05$ and $p-value < 0.1$, respectively, compared to KTBM.

Methods	EdNet AUC	Junyi AUC	MORF AUC
LSTM	0.8768^{**}	0.9069^{**}	0.9221^{*}
MANN	<u>0.8933^{*}</u>	0.9299^{**}	0.9223^{*}
TAMKOT	0.8929^{**}	0.9355^{*}	0.9256^{*}
GMKT	0.8932^{*}	<u>0.9360^{*}</u>	<u>0.9257^{*}</u>
KTBM	**0.8992**	**0.9390**	**0.9272**
KTBM-KT	0.8898	0.9243	0.9223

5.4 Prediction Performance Comparison

For both student performance and material type preference prediction experiments, we report the mean results across five folds for each method and perform a paired t-test comparing each baseline to KTBM. MVKM is only run on the MORF dataset due to its limitations with high-dimensional data and computation time. The experiment results are presented in Tables 3 and 4 for student performance and material type preference prediction, respectively.

Dividing Vectors Observation. Through experiments exploring different dividing vectors of Pareto MTL, we observed that using extreme dividing vectors, such as $(0, 1)$ or $(1, 0)$, consistently achieved optimal prediction performance for each specific task (student performance/material type) across all datasets for KTBM, while the other task saw limited improvement. However, improvements for both tasks were achieved when the dividing vector was set to $(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$, corresponding to the direction of $\frac{\pi}{4}$. While the best trade-off value optimized by the Pareto MTL algorithm varied across datasets, our objective was to obtain meaningful results to improve predictions for both student performance and material type. Therefore, we only report the experiment results with the dividing vector set to $(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$.

Student Performance Prediction. Our experimental results show that KTBM outperforms all baseline methods in predicting student performance for all datasets. These findings highlight KTBM's capability to effectively track knowledge and accurately predict student performance. KTBM's superior performance compared to other multi-activity baselines, including the assessed-only method variants ("+M"), underscores the advantage of simultaneously modeling student behavior and knowledge. This integration allows information transfer between the two, enhancing our understanding of student knowledge acquisition. This also indicates that student knowledge is influenced by preference behavior. Note that, among the baselines, GMKT includes type preference prediction as an objective. The better results of KTBM and GMKT compared to MVKM, DMKT, and TAMKOT demonstrate that incorporating behavior as an objective and formulating a multi-objective problem for student performance and material type preference prediction improves our understanding of students' knowledge. However, GMKT does not explicitly model student behavioral preferences in material selection in a separate BM component and does not employ any multi-objective optimization, including Pareto MTL. KTBM shows superior prediction performance than GMKT. This result indicates the importance of explicit behavior modeling along with knowledge tracing in students, in addition to the effectiveness of Pareto MTL. To summarize, KTBM, a multi-task learning model that jointly models student behavior and knowledge, represents them with explicitly distinct states, and transfers information between these states, can enhance our understanding of student knowledge and improve predictions of student performance.

Material Type Prediction. Similarly, KTBM surpasses all baseline methods in predicting the type of learning material for all datasets. This result highlights the model's adeptness at capturing students' preferences for selecting learning materials and accurately predicting their future choices. Specifically, when comparing KTBM with LSTM and MANN, which do not consider or model student knowledge at all, the superior results of KTBM demonstrate that preference behavior is influenced by student knowledge. Students choose learning materials based on their knowledge and whether they have successfully solved a question or understood a video lecture content. Moreover, KTBM outperforms variants of TAMKOT and GMKT, which include both student performance and material type preference prediction objectives but do not explicitly model student behavior in a specific BM component, consider the relationship between behavior and knowledge, or use multi-objective

Table 5: Results for Groups with Different Average Performance Ranges on EdNet Data, $*$ indicate paired t-test $p-value < 0.05$ compared to KTBM.

Range of Avg Performance	Student Performance AUC				Material Type AUC			
	DKT	TAMKOT	GMKT	KTBM	LSTM	TAMKOT	GMKT	KTBM
$[0, 0.57]$	0.6315*	0.6508*	0.6527	**0.6527**	0.8675*	0.8810	0.8819	**0.8825**
$[0.57, 0.67]$	0.6367*	0.6599*	0.6685	**0.6696**	0.8791*	0.8860	0.8869*	**0.8997**
$[0.67, 1]$	0.6304*	0.6604*	0.6718*	**0.6761**	0.8780*	0.8964*	0.8973*	**0.9094**

Table 6: Results for Groups with Different Ratio of Non-assessed Activity on EdNet Data, $*$ indicate paired t-test $p-value < 0.05$ compared to KTBM.

Range of Non-Assessed Activity Ratio	Student Performance AUC				Material Type AUC			
	DKT	TAMKOT	GMKT	KTBM	LSTM	TAMKOT	GMKT	KTBM
$[0, 0.4]$	0.6761*	0.6823*	0.6844	**0.6845**	0.8177*	0.8269	0.8271	**0.8275**
$[0.4, 0.48]$	0.6359*	0.6837*	0.6849*	**0.6887**	0.8879*	0.8969*	0.8980*	**0.9073**
$[0.48, 1]$	0.6194*	0.6702*	0.6775*	**0.6821**	0.9038*	0.9120*	0.9131*	**0.9214**

optimization. This underscores the importance of formulating a multi-task learning model that combines the modeling of student behavior and knowledge, incorporating the impact of knowledge on preference behavior to improve the insights of student behavior.

Overall, our results across all datasets for both tasks demonstrate that multi-objective multi-task modeling of student knowledge and tracking their preference behaviors in explicit KT and BM states, while allowing information transfer between them, leads to a deeper mutual understanding of these aspects, ultimately benefiting each task.

5.4.1 ***Ablation Studies.*** To evaluate the effect of each component, first, we remove the BM component and the material type preference objective $\mathcal{L}_z$, creating KTBM-BM, to see if BM improves knowledge understanding. Then, we remove the KT component and the student performance objective $\mathcal{L}_r$, creating KTBM-KT, to evaluate the role of knowledge in modeling student behavior. The results for the two ablations are in Table 3 and 4. Both KTBM-BM and KTBM-KT exhibited poorer performance compared to the complete KTBM model. This indicates that student knowledge and behavior mutually influence each other. It again highlights the importance of simultaneously tracing student knowledge and modeling their behavioral preference.

5.5 Student Group Analysis

The presented results so far demonstrate the better average performance of KTBM for all students with different knowledge levels and preference behaviors. To understand where KTBM provides the most improvement in BM and KT tasks, we analyzed the results in different student groups. First, we examine KTBM's ability to predict student performance and material type for students with different average grades. Second, we investigate how the proportion of non-assessed vs. assessed activities in a student sequence relates to KTBM's predictions of student performance and material type choice. Due to space limitations, we only present results for these two analyses based on EdNet data.

For each of the two studies, we used a specific measurement for each sequence: the sequence's average score and the sequence's ratio of non-assessed activities, respectively. We then categorized all student sequences into three groups using the 33% and 66% percentiles of these measurements for each analysis, ensuring each group had a roughly equal number of sequences. We computed the AUC for each group using KTBM and compared it with the baselines DKT, TAMKOT, and GMKT for student performance prediction. Additionally, we compared KTBM's AUC with the baselines LSTM, TAMKOT, and GMKT for material type preference prediction. The results of these two studies are shown in Tables 5 and 6.

Sequence's Average Score. The results for student performance prediction show that prediction performance in all models is better for students with higher scores in assessed materials. In other words, the better the student does, the easier to predict their performance. Additionally, while the performance improvement for GMKT in the lowest score group ($[0, 0.57]$) is modest, the improvement between KTBM and other baselines, including GMKT, increases as the student's average grade increases, highlighting the effectiveness of explicitly modeling behavior and knowledge, and adding material-type objectives in enhancing performance predictions for the higher-scoring student group ($[0.67, 1]$). Furthermore, the prediction of learning material type is also more accurate for sequences with higher average scores across all models, suggesting that better-performing student scores make it easier to predict their material type selections. While KTBM's improvement in the lowest score group ($[0, 0.57]$) is again limited compared to TAMKOT and GMKT, it is more pronounced in the two higher score groups, the improvement is similar between these two groups. This indicates that combining models of knowledge and behavioral types and facilitating information transfer between them, improves the prediction accuracy of material types, especially in sequences with relatively high scores.

Non-assessed Activities Ratio. For student performance prediction, we can see that all models, except DKT, perform best in the middle group ($[0.4, 0.48]$) who only worked with non-assessed activities between 57% and 67% of the time. DKT is the only model in the table that does not have any non-assessed activity information. These results show that having a more imbalanced ratio of non-assessed to assessed activities complicates student performance predictions. However, KTBM shows the largest increase in performance prediction at the highest non-assessed activity ratio (group $[0.48, 1]$). This indicates that our model enhances student performance prediction in sequences with a higher proportion of non-assessed activities, compared to assessed ones, which is the most unstable group of all. Since non-assessed learning materials are not graded, they do not provide reliable feedback to accurately estimate how the student has learned from the learning non-assessed activity. As a result, when the number of non-assessed activities increases, KT and performance prediction will be more unstable. Improvement of KTBM compared to the baselines, especially in this group, shows the benefits of explicitly modeling behavior and integrating material-type objectives to better capture the impact of non-assessed activities on student knowledge. Furthermore, for material type preference prediction, as the ratio of non-assessed activities increases, all models achieve more accurate prediction performance. Notably, KTBM shows the largest improvement in the

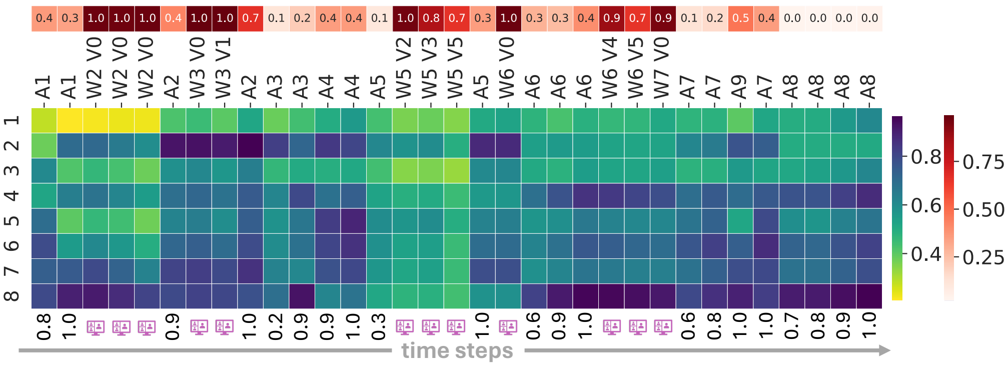

Figure 2: Visualization of the knowledge state and preference behavior for a sample student in the MORF dataset. The bottom heatmap shows the student knowledge state. The second top x-axis shows the titles of the learning materials the student interacted with at each time step. The bottom x-axis indicates the student's actual performance in assessed activities or a 'screen' icon for non-assessed activities. The y-axis represents latent concepts. The top heatmap (red) shows the predicted probabilities that the next material to be interacted with is a video lecture for the corresponding time step.

group with ratios between $[0.4, 0.48]$. This suggests that KTBM's ability to predict material type improves more for sequences with a relatively balanced mix of assessed and non-assessed activities.

5.6 Knowledge and Behavior State Visualization

To determine whether KTBM can uncover interpretable insights into student knowledge states and behavioral preferences, we visualize a representation of the learned states. Specifically, at each time step, we represent behavioral preference by calculating the probability that the student activity is non-assessed, using Eq. 15. For the student knowledge state, we predict their performance for each concept at each time. We use a masked attention weight $\tilde{\boldsymbol{w}}_t = [0, ..., w_i, ..., 0]$ to compute the masked read content $\tilde{\boldsymbol{c}}_t$ and the masked summary vector $\tilde{\boldsymbol{f}}_t$ and calculate the knowledge state of each concept using Eq. 10. We illustrate these states with heatmaps in Figure 2, showcasing the knowledge state (bottom, in blue/green) and preference behavior (top, in red) for a sample student from the MORF dataset. The x-axis between the two heatmaps indicates the titles of attempted learning activities, using abbreviations like 'W* V**' for video lectures of week * and 'A*' for assignments of week *. The bottom x-axis displays the student's actual performance for an assignment attempt or a 'screen' for a video lecture attempt, with the y-axis representing the latent concept.

We first observe that students' knowledge generally increases from the beginning to the end of the semester across almost all concepts despite fluctuations throughout the learning process, with some concepts experiencing decreases. This suggests that while students gain knowledge from learning activities, they may also forget some of the gained knowledge at times. For example, examining the last four attempts of 'A8', we see an increase in knowledge for concept eight, but a decrease for concept two, indicating potential forgetting of this concept from these activities. Moreover, our analysis shows that the learned behavior state can represent meaningful student preference behaviors. Initially, KTBM randomly guesses (≈ 0.4) about the type of material the student will choose to interact with. However, as it processes more student activities, KTBM learns more about student preference behaviors and makes more accurate predictions of material type. Furthermore, it reveals that students typically continue attempting an assignment until they achieve a perfect score before moving on to the next module. Based on their score, they decide whether to switch to watching a video lecture to improve their knowledge after receiving a very low score or to immediately retry the assignment without watching any video lectures. Our KTBM successfully captures these signals in modeling student knowledge and behavior. For instance, after initially scoring 0.3 on 'A5', the student switches to watching video lectures from week 5 before retrying 'A5'. The learned preference behavior from KTBM shows a high probability (1.0) that the student will switch to a lecture after scoring 0.3. Additionally, the knowledge state increases after the first two lecture activities of 'W5'. Conversely, after attempting 'A4' and scoring 0.9, the student tries 'A4' again instead of switching to a lecture, and KTBM learns a low preference for switching to video lectures in this scenario. Overall, this visualization demonstrates that student knowledge and behavior are interrelated and showcases an example of how KTBM results can be interpreted.

6 CONCLUSIONS

In this paper, we proposed a multi-task student knowledge and behavior model (KTBM) that effectively combines knowledge tracing and behavior modeling to enhance both tasks. By modeling the interrelationships between student knowledge and material type preference behavior, KTBM demonstrated significant improvements in performance prediction and showcased its interpretability. Our experiments showed that KTBM improves student performance and preference predictions across all student groups, and is particularly effective for predicting performance in the most challenging group: students engaged primarily in non-assessed activities. Further, our adaptation of a Pareto MTL optimization algorithm successfully addressed the dual-objective challenge, as evidenced by the enhanced results across three real-world datasets.

ACKNOWLEDGMENTS

This paper is based upon work supported by the National Science Foundation under Grant Number 2047500.

REFERENCES

[1] Ghodai Abdelrahman and Qing Wang. 2022. Deep graph memory networks for forgetting-robust knowledge tracing. *IEEE Transactions on Knowledge and Data Engineering* (2022).

[2] Ghodai Abdelrahman, Qing Wang, and Bernardo Nunes. 2023. Knowledge tracing: A survey. *Comput. Surveys* 55, 11 (2023), 1–37.

[3] Solmaz Abdi. 2022. Learner models for learnersourced adaptive educational systems. (2022).

[4] Solmaz Abdi, Hassan Khosravi, Shazia Sadiq, and Ali Darvishi. 2021. Open Learner Models for Multi-Activity Educational Systems. In *International Conference on Artificial Intelligence in Education*. Springer, 11–17.

[5] Suyeong An, Junghoon Kim, Minsam Kim, and Juneyoung Park. 2022. No task left behind: Multi-task learning of knowledge tracing and option tracing for better student assessment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 4424–4431.

[6] Juan Miguel L Andres, Ryan S Baker, George Siemens, Dragan Gašević, and Catherine A Spann. 2016. Replicating 21 findings on student success in online learning. *Technology, Instruction, Cognition, and Learning* 10, 4 (2016), 313–333.

[7] Patrick Buckley and Elaine Doyle. 2016. Gamification and student motivation. *Interactive learning environments* 24, 6 (2016), 1162–1175.

[8] May Kristine Jonson Carlon and Jeffrey S Cross. 2022. Knowledge tracing for adaptive learning in a metacognitive tutor. *Open Education Studies* 4, 1 (2022), 206–224.

[9] Hao Cen, Kenneth Koedinger, and Brian Junker. 2006. Learning factors analysis–a general method for cognitive model evaluation and improvement. In *International conference on intelligent tutoring systems*. Springer, 164–175.

[10] Haw-Shiuan Chang, Hwai-Jung Hsu, and Kuan-Ta Chen. 2015. Modeling Exercise Relationships in E-Learning: A Unified Approach.. In *EDM*. 532–535.

[11] Mingzhi Chen, Quanlong Guan, Yizhou He, Zhenyu He, Liangda Fang, and Weiqi Luo. 2022. Knowledge tracing model with learning and forgetting behavior. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 3863–3867.

[12] Song Cheng, Qi Liu, Enhong Chen, Kai Zhang, Zhenya Huang, Yu Yin, Xiaoqing Huang, and Yu Su. 2022. AdaptKT: A domain adaptable method for knowledge tracing. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*. 123–131.

[13] Youngduck Choi, Youngnam Lee, Junghyun Cho, Jineon Baek, Byungsoo Kim, Yeongmin Cha, Dongmin Shin, Chan Bae, and Jaewe Heo. 2020. Towards an appropriate query, key, and value computation for knowledge tracing. In *Proceedings of the 7th ACM Conference on Learning at Scale*. ACM, New York, NY, USA, 341–344.

[14] Youngduck Choi, Youngnam Lee, Dongmin Shin, Junghyun Cho, Seoyon Park, Seewoo Lee, Jineon Baek, Chan Bae, Byungsoo Kim, and Jaewe Heo. 2020. Ednet: A large-scale hierarchical dataset in education. In *International Conference on Artificial Intelligence in Education*. Springer, 69–73.

[15] Konstantina Chrysafiadi and Maria Virvou. 2013. Student modeling approaches: A literature review for the last decade. *Expert Systems with Applications* 40, 11 (2013), 4715–4729.

[16] CMU DataShop. 2015. Junyi dataset. https://pslcdatashop.web.cmu.edu/Project?id=244.

[17] Albert T Corbett and John R Anderson. 1994. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction* 4, 4 (1994), 253–278.

[18] Albert T Corbett and John R Anderson. 1994. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction* 4, 4 (1994), 253–278.

[19] Fisnik Dalipi, Ali Shariq Imran, and Zenun Kastrati. 2018. MOOC dropout prediction using machine learning techniques: Review and research challenges. In *2018 IEEE global engineering education conference (EDUCON)*. IEEE, 1007–1014.

[20] Liesbeth De Paepe, Chang Zhu, and Koen DePryck. 2018. Drop-out, retention, satisfaction and attainment of online learners of Dutch in adult education. *International Journal on E-Learning* 17, 3 (2018), 303–323.

[21] Jean-Antoine Désidéri. 2012. Mutiple-gradient descent algorithm for multiobjective optimization. In *European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS 2012)*.

[22] Thanh-Nam Doan and Shaghayegh Sahebi. 2019. Rank-based tensor factorization for student performance prediction. In *12th International Conference on Educational Data Mining (EDM)*.

[23] Anmei Dong, Morris Siu-Yung Jong, and Ronnel B King. 2020. How does prior knowledge influence learning engagement? The mediating roles of cognitive load and help-seeking. *Frontiers in psychology* 11 (2020), 591203.

[24] Fritz Drasgow and Charles L Hulin. 1990. Item Response Theory. *Handbook of Industrial and Organizational Psychology* (1990), 577–636.

[25] Mahidhar Dwarampudi and NV Reddy. 2019. Effects of padding on LSTMs and CNNs. *arXiv preprint arXiv:1903.07288* (2019).

[26] Jörg Fliege and Benar Fux Svaiter. 2000. Steepest descent methods for multicriteria optimization. *Mathematical methods of operations research* 51 (2000), 479–494.

[27] Jorg Fliege and A Ismael F Vaz. 2016. A method for constrained multiobjective optimization based on SQP techniques. *SIAM Journal on Optimization* 26, 4 (2016), 2091–2119.

[28] Chase Geigle and ChengXiang Zhai. 2017. Modeling MOOC student behavior with two-layer hidden Markov models. In *Proceedings of the fourth (2017) ACM conference on learning@ scale*. 205–208.

[29] Aritra Ghosh, Neil Heffernan, and Andrew S Lan. 2020. Context-Aware Attentive Knowledge Tracing. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, New York, NY, USA, 2330–2339.

[30] Patricia Goldberg, Ömer Sümer, Kathleen Stürmer, Wolfgang Wagner, Richard Göllner, Peter Gerjets, Enkelejda Kasneci, and Ulrich Trautwein. 2021. Attentive or not? Toward a machine learning approach to assessing students' visible engagement in classroom instruction. *Educational Psychology Review* 33 (2021), 27–49.

[31] Simon Haykin. 1994. *Neural networks: a comprehensive foundation*. Prentice Hall PTR.

[32] Qiwei He, Francesca Borgonovi, and Marco Paccagnella. 2021. Leveraging process data to assess adults' problem-solving skills: Using sequence mining to identify behavioral patterns across digital tasks. *Computers & Education* 166 (2021), 104170.

[33] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[34] Jinho Kim, Jung Yeon Park, and Wim Van Den Noortgate. 2020. On the use of Bayesian probabilistic matrix factorization for predicting student performance in online learning environments. *10.4995/HEAd20. 2020.11137* (2020), 751–759.

[35] Jeanine Kirchner-Krath, Manuel Schmidt-Kraepelin, Sofia Schöbel, Mathias Ullrich, Ali Sunyaev, and Harald FO Von Korflesch. 2024. Outplay Your Weaker Self: A Mixed-Methods Study on Gamification to Overcome Procrastination in Academia. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–19.

[36] Mukesh Kumar, AJ Singh, and Disha Handa. 2017. Literature survey on student's performance prediction in education using data mining techniques. *International Journal of Education and Management Engineering* 7, 6 (2017), 40–49.

[37] Jian-Yu Li, Zhi-Hui Zhan, Yun Li, and Jun Zhang. 2023. Multiple tasks for multiple objectives: A new multiobjective optimization method via multitask optimization. *IEEE Transactions on Evolutionary Computation* (2023).

[38] Xi Lin, Hui-Ling Zhen, Zhenhua Li, Qing-Fu Zhang, and Sam Kwong. 2019. Pareto multi-task learning. *Advances in neural information processing systems* 32 (2019).

[39] Angela Lopez-del Rio, Maria Martin, Alexandre Perera-Lluna, and Rabie Saidi. 2020. Effect of sequence padding on the performance of deep learning models in archaeal protein functional prediction. *Scientific reports* 10, 1 (2020), 14634.

[40] Mehrdad Mirzaei. 2020. *Discriminative Factorization Models for Student Behavioral Pattern Detection and Classification*. State University of New York at Albany.

[41] Mehrdad Mirzaei, Shaghayegh Sahebi, and Peter Brusilovsky. 2020. Structure-Based Discriminative Matrix Factorization for Detecting Inefficient Learning Behaviors. In *2020 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*. IEEE, 283–290.

[42] Libby V Morris, Catherine Finnegan, and Sz-Shyan Wu. 2005. Tracking student behavior, persistence, and achievement in online courses. *The Internet and Higher Education* 8, 3 (2005), 221–231.

[43] Libby V Morris, Sz-Shyan Wu, and Catherine L Finnegan. 2005. Predicting retention in online general education courses. *The American Journal of Distance Education* 19, 1 (2005), 23–36.

[44] Koki Nagatani, Qian Zhang, Masahiro Sato, Yan-Ying Chen, Francine Chen, and Tomoko Ohkuma. 2019. Augmenting knowledge tracing by considering forgetting behavior. In *The world wide web conference*. 3101–3107.

[45] Shalini Pandey and George Karypis. 2019. A Self-Attentive Model for Knowledge Tracing. In *Proceedings of the 12th International Conference on Educational Data Mining*. International Educational Data Mining Society, 384–389.

[46] Shalini Pandey, Andrew Lan, George Karypis, and Jaideep Srivastava. 2020. Learning student interest trajectory for mooc thread recommendation. In *2020 International Conference on Data Mining Workshops (ICDMW)*. IEEE, 400–407.

[47] Philip I Pavlik Jr and John R Anderson. 2005. Practice and forgetting effects on vocabulary memory: An activation-based model of the spacing effect. *Cognitive science* 29, 4 (2005), 559–586.

[48] Philip I Pavlik Jr, Hao Cen, and Kenneth R Koedinger. 2009. Performance Factors Analysis–A New Alternative to Knowledge Tracing. *Online Submission* (2009).

[49] Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas Guibas, and Jascha Sohl-Dickstein. 2015. Deep Knowledge Tracing. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1* (Montreal, Canada). MIT Press, Cambridge, MA, USA, 505–513.

[50] Chen Pojen, Hsieh Mingen, and Tsai Tzuyang. 2020. Junyi Academy Online Learning Activity Dataset: A large-scale public online learning activity dataset from elementary to senior high school students. *Dataset available from https://www.kaggle.com/junyiacademy/learning-activity-public-dataset-by-junyi-academy* (2020).

[51] Sherry Sahebi, Mengfan Yao, Siqian Zhao, and Reza Feyzi Behnagh. 2024. MoMENt: Marked Point Processes with Memory-Enhanced Neural Networks for User Activity Modeling. *ACM Transactions on Knowledge Discovery from Data* 18, 6 (2024), 1–32.

[52] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. 2016. Meta-learning with memory-augmented neural networks. In *International conference on machine learning*. PMLR, 1842–1850.

[53] Panos Seferlis and Michael C Georgiadis. 2004. *The integration of process design and control*. Elsevier.

[54] Ozan Sener and Vladlen Koltun. 2018. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems* 31 (2018).

[55] Teck Kiang Tan and Lakshminarayanan Samavedham. 2022. The learning process matter: A sequence analysis perspective of examining procrastination using learning management system. *Computers and Education Open* 3 (2022), 100112.

[56] Joni Tornwall, Lin Lu, and Kui Xie. 2020. Frequency of participation in student response system activities as a predictor of final grade: An observational study. *Nurse Education Today* 87 (2020), 104342.

[57] K VanLehn. 1988. Student modeling. Foundations of Intelligent Tutoring Systems. M. Polson and J. Richardson.

[58] Chunpai Wang and Shaghayegh Sahebi. 2023. Continuous Personalized Knowledge Tracing: Modeling Long-Term Learning in Online Environments. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 2616–2625.

[59] Chunpai Wang, Siqian Zhao, and Shaghayegh Sahebi. 2021. Learning from Non-Assessed Resources: Deep Multi-Type Knowledge Tracing. *International Educational Data Mining Society* (2021).

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

[61] Mengfan Yao, Siqian Zhao, Shaghayegh Sahebi, and Reza Feyzi Behnagh. 2021. Relaxed clustered hawkes process for student procrastination modeling in MOOCs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 4599–4607.

[62] Chun Kit Yeung. 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*. International Educational Data Mining Society, 683–686.

[63] Chengjiu Yin, Fumiya Okubo, Atsushi Shimada, Misato Oi, Sachio Hirokawa, and Hiroaki Ogata. 2015. Identifying and analyzing the learning behaviors of students using e-books. In *Proceedings of the 23rd international conference on computers in education*. Asia-Pacific Society for Computers in Education Hangzhou, China, 118–120.

[64] Leila Zahedi, Stephanie J Lunn, Samira Pouyanfar, Monique S Ross, and Matthew W Ohland. 2020. Leveraging machine learning techniques to analyze computing persistence in undergraduate programs. In *2020 ASEE Virtual Annual Conference Content Access*.

[65] L Ross Zahedi. 2019. Implications of gamification in learning environments on computer sci-ence students: A comprehensive study. In *126th Annual Conference and Exposition of American Society for Engineering Education*.

[66] Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung. 2017. Dynamic Key-Value Memory Networks for Knowledge Tracing. In *Proceedings of the 26th International Conference on World Wide Web* (Perth, Australia). ACM, New York, NY, USA, 765–774.

[67] Liang Zhang, Xiaolu Xiong, Siyuan Zhao, Anthony Botelho, and Neil T Heffernan. 2017. Incorporating Rich Features into Deep Knowledge Tracing. In *Proceedings of the 4th ACM Conference on Learning at Scale*. ACM, New York, NY, USA, 169–172.

[68] Siqian Zhao and Shaghayegh Sahebi. 2023. Graph-Enhanced Multi-Activity Knowledge Tracing. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 529–546.

[69] Siqian Zhao and Sherry Sahebi. 2024. Exploring Simultaneous Knowledge and Behavior Tracing. In *Proceedings of the 17th International Conference on Educational Data Mining*. 927–932.

[70] Siqian Zhao and Shaghayegh Sahebi. 2024. Towards Multi-Objective Behavior and Knowledge Modeling in Students. In *Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization*. 183–188.

[71] Siqian Zhao, Chunpai Wang, and Shaghayegh Sahebi. 2020. Modeling Knowledge Acquisition from Multiple Learning Resource Types. In *Proceedings of The 13th International Conference on Educational Data Mining*. International Educational Data Mining Society, 313–324.

[72] Siqian Zhao, Chunpai Wang, and Shaghayegh Sahebi. 2022. Transition-Aware Multi-Activity Knowledge Tracing. In *2022 IEEE International Conference on Big Data (Big Data)*. IEEE, 1760–1769.

[73] Eckart Zitzler and Lothar Thiele. 1999. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE transactions on Evolutionary Computation* 3, 4 (1999), 257–271.