# Reconciling Cognitive Modeling with Knowledge Forgetting: A Continuous Time-aware Neural Network Approach

**Haiping Ma**[1,2], **Jingyuan Wang**[1,2], **Hengshu Zhu**[3*], **Xin Xia**[4], **Haifeng Zhang**[4], **Xingyi Zhang**[1,5*] and **Lei Zhang**[1,6]
[1]Information Materials and Intelligent Sensing Laboratory of Anhui Province, Anhui University, China
[2]Institutes of Physical Science and Information Technology, Anhui University, China
[3]Baidu Talent Intelligence Center, Baidu Inc, China
[4]School of Mathematical Science, Anhui University, China
[5] School of Artificial Intelligence, Anhui University, China
[6]School of Computer Science and Technology, Anhui University, China
hpma@ahu.edu.cn, mbzyk1001@163.com, zhuhengshu@gmail.com, xxia98@stu.ahu.edu.cn, haifengzhang1978@gmail.com, xyzhanghust@gmail.com, zl@ahu.edu.cn

## Abstract

As an emerging technology of computer-aided education, cognitive modeling aims at discovering the knowledge proficiency or learning ability of students, which can enable a wide range of intelligent educational applications. While considerable efforts have been made in this direction, a long-standing research challenge is how to naturally integrate the forgetting mechanism into the learning process of knowledge concepts. To this end, in this paper, we propose a novel Continuous Time based Neural Cognitive Modeling (CT-NCM) approach to integrate the dynamism and continuity of knowledge forgetting into students' learning process modeling in a realistic manner. To be specific, we first adapt the neural Hawkes process with a specially-designed learning event encoding method to model the relationship between knowledge learning and forgetting with continuous time. Then, we propose a learning function with extendable settings to jointly model the change of different knowledge states and their interactions with the exercises at each moment. In this way, CT-NCM can simultaneously predict the future knowledge state and exercise performance of students. Finally, we conduct extensive experiments on five real-world datasets with various benchmark methods. The experimental results clearly validate the effectiveness of CT-NCM and show its interpretability in terms of knowledge learning visualization.

## 1 Introduction

Recent years have witnessed the rapid development of intelligent tutoring systems, which has accumulated a large amount of exercising records of students, and thus, enables a new data-driven paradigm of computer-aided education [Abidoğlu *et al.*, 2017]. Along this line, as an emerging technology, cognitive modeling has attracted increasing attention, which aims to discover students' knowledge proficiency or learning ability. Indeed, the results of cognitive modeling can benefit a wide range of intelligent educational applications, such as performance prediction [Chen *et al.*, 2005] and personalized course recommendation [Lan and Baraniuk, 2016].

In the past decade, while considerable efforts have been made in the field of cognitive modeling, a long-standing research challenge is how to naturally integrate the forgetting mechanism into the learning process of knowledge concepts [Nagatani *et al.*, 2019; Ghosh *et al.*, 2020]. To be specific, according to the theories of educational psychology [Atkinson and Shiffrin, 1968; Murre and Dros, 2015], human memories (e.g., instant, long-term, or short-term memories) usually have different forgetting patterns, which are influenced by various confounding factors, such as the current knowledge state and corresponding learning process of students. In recent literatures, some research efforts have been introduced for exploring the forgetting mechanism in cognitive modeling [Ghosh *et al.*, 2020; Nagatani *et al.*, 2019]. However, the proposed methods either depend on the manually-designed forgetting features or oversimplified process assumptions (e.g., fixed and discrete learning intervals), which significantly limit the flexibility and performance of downstream applications. Therefore, there still lacks a realistic cognitive modeling approach to strike a balance between the learning and forgetting processes of knowledge concepts.

To this end, in this paper, we innovatively propose a novel Continuous Time based Neural Cognitive Modeling (CT-NCM) approach to naturally integrate the dynamic and continuous forgetting behaviors into the modeling of students' learning process. To be specific, we first adapt neural Hawkes process with a specially-designed learning event encoding method to model the relationship between knowledge learning and forgetting, which can distinguish the impact of both positive and negative responses of students on their knowledge state. Then, we propose a learning function with extendable settings to jointly model the change of different knowl-

*Corresponding author

edge states and their interactions with the exercises at each moment. Furthermore, CT-NCM can simultaneously predict the future knowledge state and exercise performance of students. In particular, all the processes in CT-NCM are modeled with continuous time awareness, which can better fit the educational psychology theories and real-world applications scenarios. Finally, we conduct extensive experiments on five real-world datasets with four state-of-art approaches and two more variants of CT-NCM. The experimental results clearly validate the effectiveness of CT-NCM and show its interpretability in terms of knowledge learning visualization.

## 2 Related Work

Considering the dynamics of the knowledge concept learning process, many efforts have been undertaken to track the change of knowledge proficiency or learning ability of each student in cognitive modeling. These existing methods can be classified into the following two categories: (1) Traditional models which represented by Bayesian Knowledge Tracing (BKT) [Corbett and Anderson, 1994] and Factorization models [Vie and Kashima, 2019]; (2) Neural network based sequential models such as Deep Knowledge Tracing (DKT) [Piech *et al.*, 2015]. DKT is the first method of using Recurrent Neural Networks (RNNs) to fit student's knowledge state and infer current exercise performance based on historical learning records.

In recent years, some methods considering the forgetting mechanism have been developed in both categories. These proposed methods mainly depend on the manually-designed forgetting features, which largely limits the performance of learning and prediction. For example, Knowledge Proficiency Tracing [Huang *et al.*, 2020] utilizes a global exponential decay to represent the forgetting of knowledge state over time; DKT+Forgetting [Nagatani *et al.*, 2019] designs forgetting factors via counting past records, which is the same as some factorization models [Vie and Kashima, 2019; Lindsey *et al.*, 2014] do. Even though few works were proposed to design forgetting and learning via purely neural networks learning, the forgetting mechanism modeling in these works is still based on oversimplified process assumptions (e.g., fixed and discrete learning intervals). For example, Learning Process-consistent Knowledge Tracing (LPKT) [Shen *et al.*, 2021] mainly focuses on learning process modeling and just discontinuously learn the decline of student's knowledge state by designing a forgetting gate, the state is updated discontinuously by a simple weighted combination of learning and forgetting factors; context-aware attentive knowledge tracing [Ghosh *et al.*, 2020] uses a monotonic attention mechanism with a global exponential decay and discrete learning interval assumption to simply simulate forgetting process of different knowledge concepts knowledge.

After that, inspired by the classical Hawkes process [Hawkes, 1971], a recent approach HawkesKT [Wang *et al.*, 2021] has been proposed, which attempts to discover temporal cross-effects between different concepts with the help of collaborative filtering and matrix factorization. However, how the past learning behaviors affect the future learning gain and forgetting rate is modeled by linear accumulation in HawkesKT, which simplifies the cumulative effect of past learning history and contradicts the long-term memory theory in psychology. To this end, in order to design a realistic approach that can strike a balance between learning and forgetting processes in cognitive learning, here we propose a novel cognitive modeling approach CT-NCM, which can better fit the actual process of memory creation and forgetting in psychology.

## 3 Preliminaries

In this section, we first formally define the cognitive modeling problem. Then, we give our assumptions of the learning process based on related theory in psychology and a brief introduction of neural Hawkes process.

### 3.1 Problem Statement

Let $\mathcal{S} = \{s_1, s_2, \ldots, s_L\}$ be the set of $L$ students, $\mathcal{Q} = \{q_1, q_2, \ldots, q_M\}$ be the set of $M$ exercises, $\mathcal{K} = \{k_1, k_2, \ldots, k_N\}$ be the set of $N$ knowledge concepts. The relationship between exercises and knowledge concepts is denoted by $Q$-matrix $Q \in \mathbb{R}^{M \times N}$, where $Q_j$ is an $N$-dimensional binary vector that indicates which knowledge concepts are required for exercise $q_j$.

Generally, when an exercise is assigned to the student, she will give an answer and get a response, which can be seen as a learning behavior of the corresponding knowledge concepts. We denote the learning sequence of a student with $n$ learning behaviors as $\mathcal{R}_n = \{(q_1, k_1, t_1, r_1), (q_2, k_2, t_2, r_2), \ldots, (q_n, k_n, t_n, r_n)\}$, where the tuple $(q_i, k_i, t_i, r_i)$ is the $i$-th learning behavior; $q_i \in \mathcal{Q}$ is the question; $k_i \in \mathcal{K}$ is the knowledge concept associated with question $q_i$, which is obtained from $Q$; $t_i$ is the timestamp for the answer; and $r_i \in \{0, 1\}$ is the response (0 represents *fault* and 1 represents *true*). Our problem definition can be expressed as follows:

**Problem Definition.** *Given students's historical learning sequence* $\mathcal{R}_n = \{(q_1, k_1, t_1, r_1), (q_2, k_2, t_2, r_2), \ldots, (q_n, k_n, t_n, r_n)\}$*, our goal is twofold: (1) diagnosing the knowledge state of each student from time* $t_1$ *to* $t_n$ *through tracking the change of knowledge state over time; and (2) predicting the student's knowledge state and performance score on specific exercise at time* $t_{n+1}$.

### 3.2 Learning Process Assumption

The multi-store model of memory [Atkinson and Shiffrin, 1968; Murre and Dros, 2015] in the cognitive psychology assumes that different human memories (i.e., instant, short-term and long-term memories) have different forgetting patterns. For example, short-term memory decays relatively quickly after a learning behavior, while long-term memory is more stable but also changes over time. Meanwhile, the study of encoding and retrieval memory [Myers, 2004] proposes that the degree of forgetting is not only related to time, but also to what student already know. Based on these theories, we give the following assumptions about the updating mechanism of knowledge state during the learning process of students.

**Assumption.** *(1) The changing pattern of knowledge state is controlled by the changing mechanism of knowledge memory during the learning process. (2) The knowledge memory changes discontinuously with each successive learning behavior occurrence, and also forgets continuously toward a stable value as time elapses between learning behaviors. (3) Current learning gain and forgetting rate are determined by the complex cumulative effect of past learning history, e.g., the sequential ordering of past learning behaviors.*

This paper attempts to naturally model such updating mechanism of knowledge state in the learning process by adaptively applying the neural Hawkes process.

### 3.3 Neural Hawkes Process

Giving historical sequential events $\mathcal{G}_n = \{(e_1, t_1), (e_2, t_2), \ldots, (e_n, t_n)\}$, where each $e_i \in \{1, 2, \ldots, E\}$ is an event type and $0 < t_1 < t_2 < \ldots$ are timestamps of occurrence. Hawkes process [Hawkes, 1971] describes the intensity function of event $e$ over continuous time $t$ as shown in Eq. (1). From Eq. (1), we can see that the intensity of event $e$ in time $t$ is the addition of the following two parts: (1) the basic intensity $\mu_e \geq 0$ that the event births with; and (2) the sum of influence from events occurring before time $t$, where $\alpha_{j,e} \geq 0$ and $\delta_{j,e} \geq 0$ indicate the degree of event $j$ initially exciting event $e$ and the decay rate of this excitation respectively.

$$\lambda_e(t) = \mu_e + \sum_{g:t_g<t} \alpha_{e_g,e} exp(-\delta_{e_g,e}(t - t_g)). \quad (1)$$

Neural Hawkes process [Mei and Eisner, 2017] generalized Hawkes process by leveraging an improved LSTM [Hochreiter and Schmidhuber, 1997] approach to capture the real-world complex effect of the past on the future. Specifically, the event intensities are determined from the hidden state $\mathbf{g}(t)$ of a continuous-time LSTM according to Eq. (2). The hidden states $\mathbf{g}(t)$ can be continually obtained from the memory vector $\mathbf{c}(t)$ via Eq. (3), where $\mathbf{o}_i$ is the output gate after inputting the $i$-th event. The memory vector $\mathbf{c}(t)$ is updated discontinuously with each successive event occurrence and then decay continuously from current value $\mathbf{c}_i$ toward a stable value $\bar{\mathbf{c}}_i$ until the next event occurs in change rate $\boldsymbol{\delta}_i$ according to Eq. (4), where $\mathbf{c}_i$ and $\bar{\mathbf{c}}_i$ are produced by two interrelated LSTM units (named $LSTMcell_1$ and $LSTMcell_2$) and updated immediately once an event occurs.

$$\lambda_e(t) = f_e(\mathbf{w}_e^T \mathbf{g}(t)), \quad (2)$$

$$\mathbf{g}(t) = \mathbf{o}_i \cdot (2\sigma(2\mathbf{c}(t)) - 1), t \in (t_i, t_{i+1}], \quad (3)$$

$$\mathbf{c}(t) = \bar{\mathbf{c}}_i + (\mathbf{c}_i - \bar{\mathbf{c}}_i) \exp(-\boldsymbol{\delta}_i(t - t_i)), t \in (t_i, t_{i+1}]. \quad (4)$$

## 4 The CT-NCM Method

In this section, we present how the CT-NCM model diagnoses student's knowledge state dynamically and predicts her future performance. As shown in Fig. 1, the CT-NCM consists of two main parts: (1) the learning and forgetting module, and (2) the performance prediction module. These two modules work periodically over time. Specifically, after a student has finished an exercise $q_i$ associated with knowledge concept $k_i$ in time $t_i$, the learning and forgetting module will make her knowledge state a change instantly and a continuous decay on the interval $(t_i, t_{i+1}]$ until question $q_{i+1}$ is assigned to her at $t_{i+1}$. After the knowledge learning gains in time $t_i$ and forgetting on the interval $(t_i, t_{i+1}]$, the knowledge state is updated to achieve the latest value. Then, in the performance prediction module, student performance on exercise $q_{i+1}$ is predicted according to her latest knowledge state.

| Nota. | Description |
|---|---|
| $M$ | the number of exercises |
| $N$ | the number of knowledge concepts |
| $t_i$ | the timestamp of student's $i$-th answer |
| $q_i$ | the question that student answers at time $t_i$ |
| $k_i$ | the concept that $q_i$ corresponds to |
| $\mathbf{c}(t)$ | the memory vector of continuous-time LSTM at time $t$ |
| $\mathbf{h}_s(t)$ | the student's knowledge state at time $t$ |
| $\boldsymbol{\delta}_i$ | the descent rate vector of $\mathbf{c}(t)$ on interval $(t_i, t_{i+1}]$ |
| $d$ | the dimensions of $\mathbf{c}(t)$ and $\mathbf{h}_s(t)$ |
| $\hat{y}_t^s$ | the predicted response of student $s$ at time $t$ |

Table 1: A list of main notations used in this work.

For facilitating description, the discussion below only takes place in a time interval $(t_i, t_{i+1}]$, where the left open means that the learning behavior (the student answers exercise $q_i$) has occurred, and right closed means that the next learning behavior has not occurred (i.e., the exercise $q_{i+1}$ is assigned to the student in time $t_{i+1}$, but the student has not answered it). Some main notations used in this work have been shown in Table 1.

### 4.1 Learning and Forgetting Module

We model the student's knowledge learning and forgetting over time by adaptively utilizing neural Hawkes process on the student learning sequence $\mathcal{R}_n = \{(q_1, k_1, t_1, r_1), (q_2, k_2, t_2, r_2), \ldots, (q_n, k_n, t_n, r_n)\}$. More specifically, we use the update mechanism of the hidden state $\mathbf{g}(t)$, which is the output of continuous-time LSTM in neural Hawkes process adaptively, to track the change of student knowledge state $\mathbf{h}_s(t)$ because of discontinuous knowledge learning and continuous forgetting. In the following, how the knowledge state updates instantly because of learning behavior $(q_i, k_i, t_i, r_i)$ and decays continuously in the interval $(t_i, t_{i+1}]$ due to knowledge forgetting will be introduced in detail respectively.

**Learning.** We firstly formally define learning event embedding $\mathbf{x}^i$ based on the learning behavior tuple $(q_i, k_i, t_i, r_i)$, which is the $i$-th input of continuous-time LSTM. We consider that two facts existing: (1) the essence of learning behavior $(q_i, k_i, t_i, r_i)$ is the learning of the knowledge concept $k_i$; and (2) the different correctness of student's answers affects her knowledge state in different degrees.

Under such consideration, we generate $\mathbf{x}^i$ by combining

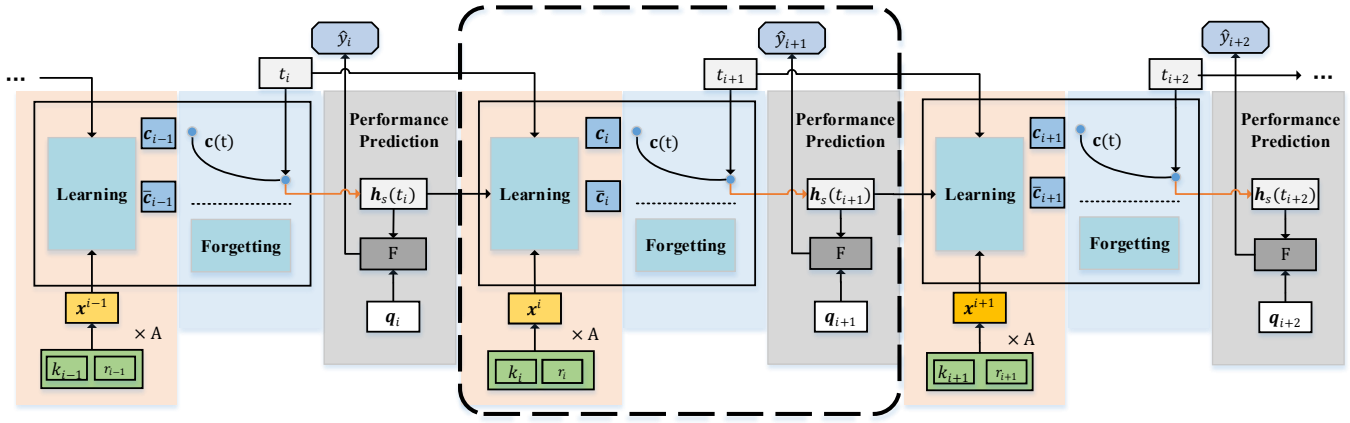


Figure 1: The overview of CT-NCM method, which consists of two modules: learning and forgetting module (the orange and light blue boxes) and performance prediction module (the light grey box). In the period $(t_i, t_{i+1}]$ (the dotted frame), the learning part receives an learning event $(k_i, r_i)$ as input, then the memory vector $\mathbf{c}(t)$ is updated and achieved the latest value $\mathbf{c}_i$, and the knowledge state $\mathbf{h}_s(t_i)$ is updated correspondingly. After that, the memory vector continuously decays until the next learning event occurs, which makes the knowledge state continuously updated and achieved the latest value $\mathbf{h}_s(t_{i+1})$. In the performance prediction module, an interaction function $F$ is set to receive student knowledge state $\mathbf{h}_s(t_{i+1})$ and exercise one hot representation $\mathbf{q}_{i+1}$ to predict the answering performance $\hat{y}_{i+1}$ according to Eq. (10).

the knowledge concept $k_i$ and the response $r_i$ as:

$$a_j^i = \begin{cases} 1 & j = k_i + N \cdot r_i, \\ 0 & otherwise. \end{cases} \tag{5}$$

$$\mathbf{x}^i = \mathbf{a}^i \times \mathbf{A}, \tag{6}$$

where $\mathbf{x}^i \in \mathbb{R}^m$, $\mathbf{a}^i = \{a_j^i \mid j = 1, ..., 2N\}$, $\mathbf{A} \in \mathbb{R}^{2N \times m}$, $m$ is the embedding size. $\mathbf{A}$ is an embedding matrix that needs training.

Entering the event embedding $\mathbf{x}^i$ into continuous-time LSTM, the student knowledge state is updated instantly from the value $\mathbf{h}_s(t_i)$ to the latest value $\mathbf{h}_s^i = \mathbf{o}_i \cdot (2\sigma(2\mathbf{c}_i) - 1)$, where $\mathbf{c}_i$ is the latest memory vector by updating the previous memory vector $\mathbf{c}(t_i)$ according to Eq. (7). Here, the $\mathbf{c}(t_i)$ and $\mathbf{h}_s(t_i)$ are the latest memory vector and student knowledge state vector before the learning event $\mathbf{x}^i$ occurring.

$$\mathbf{c}_i = LSTMcell_1((\mathbf{c}(t_i), \mathbf{h}_s(t_i)), \mathbf{x}^i). \tag{7}$$

**Forgetting.** After the update owing to the learning event $\mathbf{x}^i$, the knowledge state will face a continuous decay until the next learning event $\mathbf{x}^{i+1}$ occurs, because of the forgetting over time. Specifically, the memory vector $\mathbf{c}(t)$ decays from $\mathbf{c}_i$ towards a steady-state value $\bar{\mathbf{c}}_i$ with rate $\boldsymbol{\delta}_i$, where both $\bar{\mathbf{c}}_i$ and $\boldsymbol{\delta}_i$ are determined when the learning event $\mathbf{x}^i$ occurs. Here, $\bar{\mathbf{c}}_i$ is calculated according to Eq. (8); and $\boldsymbol{\delta}_i$ is calculated as a positive vector by Eq. (9), which can guarantee the knowledge state decay in interval $(t_i, t_{i+1}]$ because of forgetting. Correspondingly, the knowledge state $\mathbf{h}_s(t)$ changes over time from $\mathbf{h}_s^i$ to $\mathbf{h}_s(t_{i+1})$ by getting its value from $\mathbf{c}(t)$ according to Eq. (3).

$$\bar{\mathbf{c}}_i = LSTMcell_2((\bar{\mathbf{c}}_{i-1}, \mathbf{h}_s(t_i)), \mathbf{x}^i), \tag{8}$$

$$\boldsymbol{\delta}_i = f_s(\mathbf{W} \cdot [\mathbf{x}^i, \mathbf{h}_s(t_i)] + \mathbf{b}), \tag{9}$$

where $\mathbf{W} \in \mathbb{R}^{d \times (m+d)}$, $\mathbf{b} \in \mathbb{R}^{d \times 1}$, $f_s(x) = \log(1 + e^x)$ is a softplus function that can guarantee the positive characterization of the decay vector.

## 4.2 Performance Prediction Module

With the latest knowledge state $\mathbf{h}_s(t_{i+1})$, we will predict student's performance on the exercise $q_{i+1}$ which is assigned to the student at $t_{i+1}$. Let $\mathbf{q}_{i+1}$ represent the one hot vector of $q_{i+1}$, the probability $\hat{y}_{i+1}$ that indicates the likelihood her answer $\mathbf{q}_{i+1}$ correctly can be written formally as:

$$\hat{y}_{i+1} = F(\mathbf{h}_s(t_{i+1}), \mathbf{q}_{i+1}), \tag{10}$$

where $F(\cdot)$ is the interaction function between student factors and exercise factors. Inspired by a classical educational psychology theory, the Item Response Theory (IRT) [Lord, 1980], we design $F(\mathbf{h}_s, \mathbf{q})$ with extendable settings carefully, as shown in Eq. (11). The interpretable parameters $\mathbf{h}_s^{pro}$ represents student factor (e.g., ability level) that can be obtained from $\mathbf{h}_s$; $\mathbf{h}_q^{diff}$ and $h_q^{disc}$ represent the difficulty and discrimination levels of exercise $q$ respectively.

$$\begin{aligned} F(\mathbf{h}_s, \mathbf{q}) &= F_1(\mathbf{h}_s^{pro}, \mathbf{h}_q^{diff}, h_q^{disc}) \\ &= F_2(F_3((\mathbf{h}_s^{pro} - \mathbf{h}_q^{diff}) \times h_q^{disc})), \end{aligned} \tag{11}$$

Here, we give a special student-exercise interaction function denoted $f(\mathbf{h}_s, \mathbf{q})$ under such extended formulation in Eq. (11). To be specific, the student factor $\mathbf{h}_s^{pro}$ is set to $\mathbf{h}_s$ and the two exercise factors are set as Eq. (12) and (13). $F_3$ is set as a three-layer fully connected neural network to fit the complex interactions, and $F_2$ is a sigmoid function for normalizing the prediction value into range of $(0, 1)$.

$$\mathbf{h}_q^{diff} = \sigma(\mathbf{q} \times \mathbf{B}), \mathbf{B} \in \mathbb{R}^{M \times d}, \tag{12}$$

$$h_q^{disc} = \sigma(\mathbf{q} \times \mathbf{C}), \mathbf{C} \in \mathbb{R}^{M \times 1}, \tag{13}$$

where $M$ is the number of exercises, $d$ is the hidden size, $\sigma$ is the sigmoid function, $\mathbf{B}$ and $\mathbf{C}$ are two embedding matrixes that need training.

Finally, in order to illustrate the extendibility of Eq. (11) for different variants, we use two representative IRT-based interaction functions: the item response term (IRT) [Lord, 1980] and neural cognitive model (NCD) [Wang *et al.*, 2020] to set $F(\cdot)$, named CT-NCM_IRT and CT-NCM_NCD, as follows:

| Datasets | ASSISTment12 | | ASSISTment17 | | Slepemapy.cz | | Junyi | | EdNet-KT1 | |
|---|---|---|---|---|---|---|---|---|---|---|
| Metrics | ACC | AUC | ACC | AUC | ACC | AUC | ACC | AUC | ACC | AUC |
| DKT | 0.7239 | 0.6939 | 0.6768 | 0.7029 | 0.8060 | 0.7183 | 0.8382 | 0.7026 | 0.6798 | 0.6583 |
| DKT-V | 0.7533 | 0.7701 | 0.7044 | *0.7536* | 0.8097 | 0.7687 | 0.8450 | 0.7721 | 0.7214 | 0.7582 |
| DKT+Forgetting | 0.7327 | 0.7266 | 0.6793 | 0.7075 | 0.8100 | 0.7692 | 0.8413 | 0.7323 | 0.6894 | 0.6855 |
| AKT | *0.7553* | *0.7748* | 0.7061 | 0.7515 | *0.8118* | *0.7761* | *0.8478* | *0.7836* | **0.7283** | *0.7688* |
| HawkesKT | 0.7481 | 0.7617 | 0.6845 | 0.7033 | 0.8082 | 0.7588 | 0.8410 | 0.7609 | 0.7165 | 0.7495 |
| CT-NCM_IRT | 0.7541 | 0.7841 | 0.7251 | 0.7926 | 0.8118 | 0.7811 | 0.8458 | 0.7777 | 0.7230 | 0.7627 |
| CT-NCM_NCD | 0.7535 | 0.7844 | 0.7319 | 0.7995 | 0.8130 | 0.7857 | 0.8467 | 0.7791 | 0.7246 | 0.7648 |
| CT-NCM | **0.7609** | **0.7945** | **0.7430** | **0.8166** | **0.8136** | **0.7866** | **0.8480** | **0.7842** | *0.7271* | **0.7691** |

Table 2: Performance of CT-NCM, its two variants and all baseline methods on all datasets on predicting future student responses, where the best methods are bold, the second-best models are marked in italics and underlined.

| Dataset | Student | Concept | Exercise | Interaction |
|---|---|---|---|---|
| ASSISTment12 | $25.3k$ | 245 | $50.9k$ | $2,621.3k$ |
| ASSISTment17 | $1.7k$ | 102 | $3.2k$ | $942.8k$ |
| Slepemapy.cz | $81.7k$ | $1,458$ | $2.9k$ | $9,786.5k$ |
| Junyi | $175.4k$ | 40 | $0.7k$ | $25,670.2k$ |
| EdNet-KT1 | $685.4k$ | 141 | $12.3k$ | $95,023.7k$ |

Table 3: Statistics of all datasets.

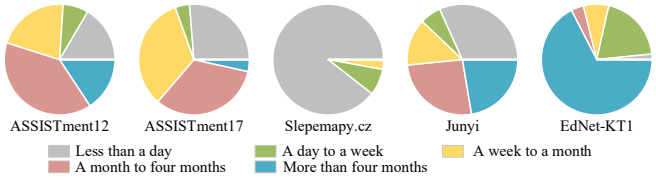


Figure 2: The pie charts of sequence time span for five datasets.

**IRT.** Taking its classical formulation $y = \sigma((h_s^{pro} - h_q^{diff}) \times h_q^{disc})$ as an example, where $h_s^{pro}$, $h_q^{diff}$ and $h_q^{disc}$ are unidimensional student proficiency, exercise difficulty and discrimination respectively. Given $\mathbf{h}_s \in \mathbb{R}^{d\times 1}$ and $\mathbf{q} \in \mathbb{R}^{1\times M}$, $h_s^{pro}$ is obtained through $h_s^{pro} = \sigma(\mathbf{W} \times \mathbf{h}_s)$, $h_q^{disc}$ is obtained by Eq. (13), $h_q^{diff}$ is obtained via $h_q^{diff} = \sigma(\mathbf{q}\times\mathbf{D})$, where $\mathbf{D} \in \mathbb{R}^{M\times 1}$.

**NCD.** It models student-exercise interactions more generally as: $y = \sigma(f'(Q_q \circ (\mathbf{h}_s^{pro} - \mathbf{h}_q^{diff}) \times h_q^{disc}))$, where $f'$ denotes multi-layer fully connected layer, $\mathbf{h}_s^{pro}$ and $\mathbf{h}_q^{diff}$ are mapped into the knowledge space with the help of exercise-knowledge relationship vector $Q_q$ through the element-wise product $\circ$. Actually, NCD is an implementation of the extendable formulation in Eq. (11) via the settings: $F_3$ in Eq. (11) is set by $f'$; $F_2$ is set as a sigmoid function; $h_s^{pro}$ is obtained from $\mathbf{h}_s^{pro} = \sigma(\mathbf{W} \times \mathbf{h}_s)$; $h_q^{disc}$ is set as same as Eq. (13); and $\mathbf{h}_q^{diff}$ is set according to $\mathbf{h}_q^{diff} = \sigma(\mathbf{q} \times \mathbf{E})$. Where $\mathbf{W} \in \mathbb{R}^{N\times d}, \mathbf{E} \in \mathbb{R}^{M\times N}$ are two matrices need training.

### 4.3 Optimization

All parameters in CT-NCM are learned by joint training the learning sequences of students. A binary cross entropy loss between the performance prediction $\hat{y}_t^s$ for student $s$ at time $t$ and its corresponding ground truth $y_t^s$ is used to optimize these parameters, as follows:

$$\mathcal{L} = -\sum_s \sum_t \left(y_t^s \log \hat{y}_t^s + (1 - y_t^s) \log \left(1 - \hat{y}_t^s\right)\right). \quad (14)$$

## 5 Experiment

### 5.1 Experimental Settings

**Dataset Description**

Five public available datasets are used in our experiments, i.e., ASSISTment12[1], ASSISTment17[2], Slepemapy.cz[3], Junyi[4], and EdNet-KT1[5]. ASSISTment12 [Feng *et al.*, 2009] is collected from ASSISTments online tutoring service system and contains student exercising data for the school year 2012-2013. ASSISTment17 [Patikorn *et al.*, 2018] is published in ASSISTments Longitudinal Data Mining Competition in 2017. Slepemapy.cz [Papoušek *et al.*, 2016] comes from the online adaptive system slepemapy.cz, which is for practicing geography facts. Junyi [Chang *et al.*, 2015] is gathered from Junyi Academy, which is an E-learning platform built in 2012. And EdNet-KT1 [Choi *et al.*, 2020] is collected by Santa, an artificial intelligence tutoring system aiding students in preparing for the TOEIC.

For calculate efficiency, we set the max sequence length to 100 and truncate student learning sequences longer than 100 to several sub-sequences following to [Shen *et al.*, 2021]. Afterward, sequences with lengths less than 5 are removed to ensure that each sequence has enough data for cognitive modeling. The statistics and time span distribution of sequences for five datasets are shown in Table 3 and Fig. 2.

**Comparison Approaches and Metrics**

As for comparison approaches, in addition to four state-of-art approaches including DKT [Piech *et al.*, 2015], DKT+Forgetting [Nagatani *et al.*, 2019], AKT [Ghosh *et al.*, 2020] and HawkesKT [Wang *et al.*, 2021], we also design

[1]https://sites.google.com/site/assistmentsdata/datasets/2012-13-school-data-with-affect

[2]https://sites.google.com/view/assistmentsdatamining/dataset

[3]https://www.fi.muni.cz/adaptivelearning/?a=data

[4]https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=1198

[5]https://github.com/riiid/ednet

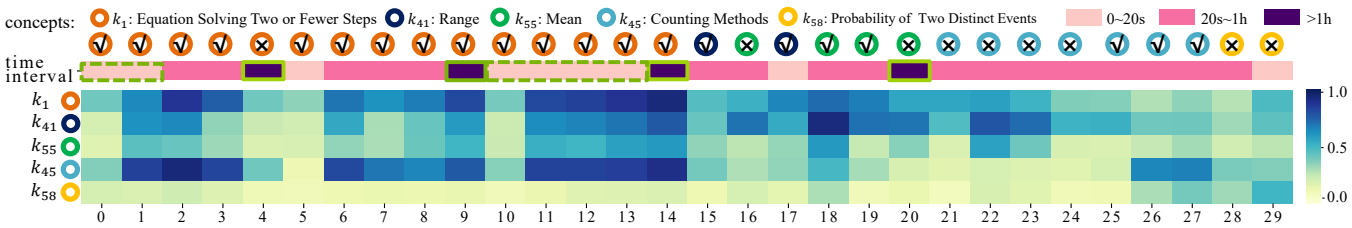


Figure 3: Visualization of student's knowledge mastery degree over 30 consecutive learning steps. Five different colors on the left and upper represent five different knowledge concepts, and the boxes with different colors in the time interval line represent different time interval ranges between current and next steps; symbols in the colored circle represent the correctness of student's answer on the corresponding concept at that step; and the main blue heatmap represents student's knowledge mastery degree over steps.

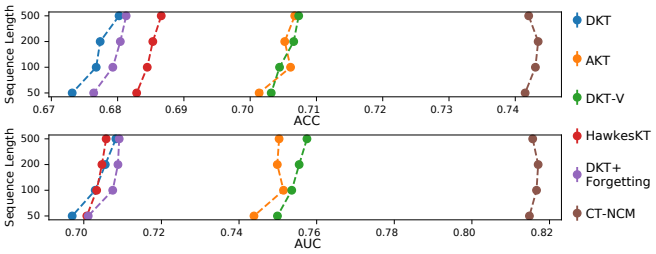


Figure 4: Robustness analysis based on ASSISTment17.

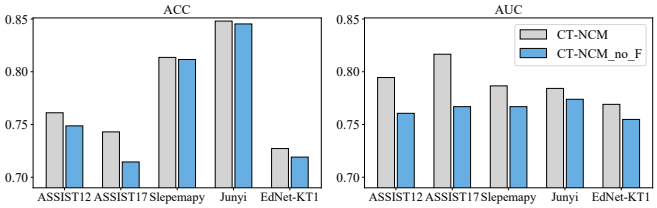


Figure 5: The results of the ablation study.

a variant of DKT named DKT-V which has the same performance prediction module as the CT-NCM so as to more fairly validate the superiority of the learning and forgetting mechanism in CT-NCM. Moreover, we also provide the results of another two versions of CT-NCM: CT-NCM_IRT and CT-NCM_NCD, whose performance prediction modules are implemented with IRT [Lord, 1980] and NCD [Wang *et al.*, 2020] respectively. As for metrics, we use accuracy (ACC) and the area under the receiver operating characteristics curve (AUC) to evaluate the performance of all methods on predicting binary-valued future student responses to exercises.

### Experimental Settings

We performed 5-fold cross-validation in the experiment[6]. For each fold, 80% of the learning sequences are split as the training set (70%) and the validation set (10%), while the rest 20% are used as the test set. We implement DKT, DKT+Forgetting, AKT and HawkesKT according to their original papers. To be specific, if the parameters are insensitive to different datasets in the original paper, they are set as same as the original paper (e.g., the all parameters of HawkesKT, the parameters apart from learning rate in AKT); otherwise, the parameter tuning was conducted on the validation set according to the value range provided in the original papers (e.g., the parameter of learning rate in AKT, the parameters of DKT and DKT+Forgetting). For CT-NCM and its two versions, we conducted the hyperparameter tuning based on the validation set. To be specific, we used $\{16, 32, 64, 128\}$, $\{16, 32, 64, 128\}$ and $\{0, 0.1, 0.2, 0.25\}$ as values of the embedding size, hidden size, and dropout rate with Adam optimizer. For DKT-V, the same part with DKT is set according to DKT, whereas the same part with CT-NCM is set according to CT-NCM.

For each sequence, all interactions except for the first one were used for training, verifying and testing. All experiments were implemented with *PyTorch* by Python and conducted with Tesla P100 and GeForce RTX3090 GPU.

## 5.2 Experimental Results

### Overall Performance

Table 2 lists the performance of our CT-NCM and all comparison approaches on predicting student future performance across five datasets. Here we report the average values over 5-fold cross-validation. The optimal results are shown in bold, and the sub-optimal results are marked in italics and underlined. Additionally, Tables 4 and 5 in Appendix provide the win/tie/loss results for the proposed CT-NCM against the compared algorithms in terms of AUC and ACC with the two-sided T-test ($p < 0.01$), respectively. From the three tables, we can make the following two observations. (1) The CT-NCM performs significantly better than all baseline on four datasets and comparably with the best baseline (i.e., AKT) on one dataset (i.e., EdNet-KT1), which validates the effectiveness of the CT-NCM. (2) All the three versions of our CT-NCM achieve remarkable performance on all datasets, which can indicate that the relationship modeling between knowledge learning and forgetting has excellent performance on tracing the change of knowledge state.

### Robustness Analysis

The length of learning sequences is an important hyperparameter in the modeling of the dynamic learning process. We report the changes in the performance of our model and all baselines when the length of learning sequence changes based

[6]https://github.com/BIMK/Intelligent-Education/tree/main/CTNCM

on ASSISTment17 dataset, which is shown in Fig. 4. Here, we set the length of learning sequence to 50, 100, 200 and 500 respectively. Note that we chose ASSISTment17 to conduct this experiment since it has the largest average sequence length (about 500). From this figure, we can see that CT-NCM exhibits higher stability as the learning sequence length changes in terms of AUC and ACC.

**Ablation Study**

Furthermore, to validate the effectiveness of forgetting modeling in CT-NCM, we design one variant of CT-NCM, which does not consider the **F**orgetting process by setting every time interval of sequences with 0, named "CT-NCM_no_F". Fig. 5 shows the experimental results of the proposed CT-NCM and the variant "CT-NCM_no_F" on all datasets in terms of AUC and ACC. For ease of view, we use several easily identified abbreviations for original dataset names. The comparison results strongly confirm the effectiveness of forgetting modeling.

**Learning Process Visualization**

In order to validate the interpretability of our proposed method in modeling student's learning process, we visualized the knowledge mastery degree changes of a student from the test set in ASSISTment12 dataset, as shown in Fig. 3. Besides, the data for visualization was obtained by CT-NCM_NCD. Generally, student knowledge state's changing degree over each step is brought by both learning and forgetting, which makes it difficult to analyze the individual influence of these two on the state. Therefore, we roughly made the following assumption: The knowledge state changing is mainly owing to the knowledge learning when the time interval is particularly short (e.g., the light color boxes marked in dotted green boxes on the time row); otherwise, it is mainly due to the knowledge forgetting when the time interval is particularly long (e.g. the deep color boxes marked in the solid green boxes on the time row).

From the figure, we have the following observations. Firstly, the knowledge mastery degree of all the observed concepts improves significantly on steps with particularly short time intervals (from steps 0 to 2 and from steps 10 to 14) owing to the learning effect. Secondly, the knowledge mastery degree of all the observed concepts decay on steps with particularly long time intervals (e.g., from step 4 to 5, step 9 to 10, step 14 to 15 and step 20 to 21) due to forgetting effect. Intuitively, the above analysis clearly validates the effectiveness of our method on modeling the learning and forgetting process of students.

## 6 Conclusion

In this paper, we proposed a novel cognitive modeling approach with continuous time-aware neural networks, namely CT-NCM. A unique perspective of CT-NCM is to strike a balance between the learning and forgetting processes of knowledge concepts in a realistic manner. Specifically, we adapted the neural Hawkes process with a specially-designed learning event encoding method to model the knowledge learning and forgetting processes with continuous time awareness. Then, an extendable performance prediction function was designed to jointly model the change of different knowledge states and their interactions with the exercises at each moment. In this way, CT-NCM can simultaneously predict the future knowledge state and exercise performance of students. Experimental results on five real-world datasets clearly show the performance gain and good interpretability of our proposed model.

## A Appendices

The win/tie/loss results for the CT-NCM against the compared algorithms based on two-sided T-test ($p< 0.01$) are supplemented in Table 4 and Table 5. Note that the approach "DKT+F" is short for "DKT+Forgetting", "HKT" is short for "HawkesKT".

| Datasets | DKT | DKT-V | DKT+F | AKT | HKT |
|---|---|---|---|---|---|
| ASSIST12 | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 |
| ASSIST17 | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 |
| Slepemapy | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 |
| Junyi | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 |
| EdNet-KT1 | 1/0/0 | 1/0/0 | 1/0/0 | 0/1/0 | 1/0/0 |
| In-total | 5/0/0 | 5/0/0 | 5/0/0 | 4/1/0 | 5/0/0 |

Table 4: The win/tie/loss results for CT-NCM against the compared algorithms in terms of AUC for Table 2 with two-sided T-test ($p< 0.01$).

| Datasets | DKT | DKT-V | DKT+F | AKT | HKT |
|---|---|---|---|---|---|
| ASSIST12 | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 |
| ASSIST17 | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 |
| Slepemapy | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 | 1/0/0 |
| Junyi | 1/0/0 | 1/0/0 | 1/0/0 | 0/1/0 | 1/0/0 |
| EdNet-KT1 | 1/0/0 | 1/0/0 | 1/0/0 | 0/0/1 | 1/0/0 |
| In-total | 5/0/0 | 5/0/0 | 5/0/0 | 3/1/1 | 5/0/0 |

Table 5: The win/tie/loss results for CT-NCM against the compared algorithms in terms of ACC for Table 2 with two-sided T-test ($p< 0.01$).

## Acknowledgements

This research was partially supported by grants from the National Key Research and Development Project (NO. 2018AAA0100105), the National Natural Science Foundation of China (NO. 62107001, 61836013, 61976001), the Anhui Provincial Natural Science Foundation (NO. 2108085QF272), and the Key Program of Natural Science Project of Educational Commission of Anhui Province (NO. KJ2020A0036). Haiping Ma gratefully acknowledges the support of the CCF-Tencent Open Fund.

## References

[Abidoğlu *et al.*, 2017] Ülkü Pişkin Abidoğlu, Oya Ertuğruloğlu, and Niyal Büyükeğilmez. Importance of computer-aided education for children with autism spectrum disorder (asd). *Eurasia Journal of Mathematics, Science and Technology Education*, 13(8):4957–4964, 2017.

[Atkinson and Shiffrin, 1968] Richard C Atkinson and Richard M Shiffrin. Human memory: A proposed system and its control processes. In *Psychology of Learning and Motivation*, volume 2, pages 89–195. Elsevier, 1968.

[Chang *et al.*, 2015] Haw-Shiuan Chang, Hwai-Jung Hsu, and Kuan-Ta Chen. Modeling exercise relationships in e-learning: A unified approach. In *International Educational Data Mining Society*, pages 532–535, 2015.

[Chen *et al.*, 2005] Chih-Ming Chen, Hahn-Ming Lee, and Ya-Hui Chen. Personalized e-learning system using item response theory. *Computers & Education*, 44(3):237–255, 2005.

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

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

[Feng *et al.*, 2009] Mingyu Feng, Neil Heffernan, and Kenneth Koedinger. Addressing the assessment challenge with an online system that tutors as it assesses. *User Modeling and User-Adapted Interaction*, 19(3):243–266, 2009.

[Ghosh *et al.*, 2020] Aritra Ghosh, Neil Heffernan, and Andrew S Lan. Context-aware attentive knowledge tracing. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2330–2339, 2020.

[Hawkes, 1971] Alan G Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90, 1971.

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

[Huang *et al.*, 2020] Zhenya Huang, Qi Liu, Yuying Chen, Le Wu, Keli Xiao, Enhong Chen, Haiping Ma, and Guoping Hu. Learning or forgetting? a dynamic approach for tracking the knowledge proficiency of students. *ACM Transactions on Information Systems*, 38(2):1–33, 2020.

[Lan and Baraniuk, 2016] Andrew S Lan and Richard G Baraniuk. A contextual bandits framework for personalized learning action selection. In *International Educational Data Mining Society*, pages 424–429, 2016.

[Lindsey *et al.*, 2014] Robert V Lindsey, Jeffery D Shroyer, Harold Pashler, and Michael C Mozer. Improving students' long-term knowledge retention through personalized review. *Psychological Science*, 25(3):639–647, 2014.

[Lord, 1980] Frederic M Lord. Applications of item response theory to practical testing problems. lawrence eribaum associates. *Inc, Hillsdale, NJ*, 1980.

[Mei and Eisner, 2017] Hongyuan Mei and Jason Eisner. The neural hawkes process: A neurally self-modulating multivariate point process. *Advances in Neural Information Processing Systems*, pages 5754–6767, 2017.

[Murre and Dros, 2015] Jaap MJ Murre and Joeri Dros. Replication and analysis of ebbinghaus' forgetting curve. *PloS ONE*, 10(7):e0120644, 2015.

[Myers, 2004] David G Myers. *Psychology, in modules*. Macmillan, 2004.

[Nagatani *et al.*, 2019] Koki Nagatani, Qian Zhang, Masahiro Sato, Yan-Ying Chen, Francine Chen, and Tomoko Ohkuma. Augmenting knowledge tracing by considering forgetting behavior. In *The World Wide Web Conference*, pages 3101–3107, 2019.

[Papoušek *et al.*, 2016] Jan Papoušek, Radek Pelánek, and Vít Stanislav. Adaptive geography practice data set. *Journal of Learning Analytics*, 3(2):317–321, 2016.

[Patikorn *et al.*, 2018] Thanaporn Patikorn, Neil T Heffernan, and Ryan S Baker. Assistments longitudinal data mining competition 2017: A preface. In *Proceedings of the Workshop on Scientific Findings from the ASSISTments Longitudinal Data Competition, International Conference on Educational Data Mining*, 2018.

[Piech *et al.*, 2015] Chris Piech, Jonathan Spencer, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas Guibas, and Jascha Sohl-Dickstein. Deep knowledge tracing. *Advances in Neural Information Processing Systems*, pages 505–513, 2015.

[Shen *et al.*, 2021] Shuanghong Shen, Qi Liu, Enhong Chen, Zhenya Huang, Wei Huang, Yu Yin, Yu Su, and Shijin Wang. Learning process-consistent knowledge tracing. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1452–1460, 2021.

[Vie and Kashima, 2019] Jill-Jênn Vie and Hisashi Kashima. Knowledge tracing machines: Factorization machines for knowledge tracing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 750–757, 2019.

[Wang *et al.*, 2020] Fei Wang, Qi Liu, Enhong Chen, Zhenya Huang, Yuying Chen, Yu Yin, Zai Huang, and Shijin Wang. Neural cognitive diagnosis for intelligent education systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6153–6161, 2020.

[Wang *et al.*, 2021] Chenyang Wang, Weizhi Ma, Min Zhang, Chuancheng Lv, Fengyuan Wan, Huijie Lin, Taoran Tang, Yiqun Liu, and Shaoping Ma. Temporal cross-effects in knowledge tracing. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 517–525, 2021.