Continuous Personalized Knowledge Tracing: Modeling Long-Term Learning in Online Environment

Chunpai Wang, Sherry Sahebi

University at Albany - SUNY

PersAI onalized

UNIVERSITY AT ALBANY
State University of New York

NSF

Problem

- Continuous Personalized Sequence Modeling

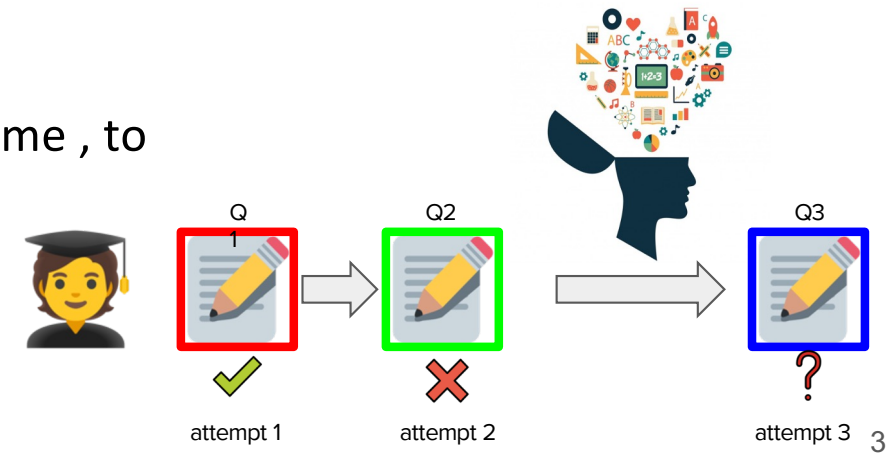
Problem

UNIVERSITY AT ALBANY
State University of New York

- Continuous Personalized Sequence Modeling

- Online Education Systems
  - abundant of practice problems with large topic varieties
  - students can have a long continuous learning experiences

- Knowledge Tracing (KT)
  - model students' knowledge level over time , to
    - predict student performance
    - create a study plan
    - recommend learning materials, etc.



Knowledge Tracing Approaches

UNIVERSITY AT ALBANY
State University of New York

- Traditional KT approaches
  - probabilistic and hidden Markov models (e.g. BKT [Corbett et al.])
  - logistic regression models (e.g., IRT [Frederic et al.], PFA [Philip et al.])
  - work well on small datasets

- Deep KT approaches
  - work well on large datasets (e.g., DKT [Piech et al.], DKVMN [Zhang et al], SAKT [Pandey et al.])

Knowledge Tracing Approaches

- Traditional KT approaches
  - probabilistic and hidden Markov models (e.g. BKT [Corbett et al.])
  - logistic regression models (e.g., IRT [Frederic et al.], PFA [Philip et al.])
  - work well on small datasets

- Deep KT approaches
  - work well on large datasets (e.g., DKT [Piech et al.], DKVMN [Zhang et al], SAKT [Pandey et al.])
  - but, aim to learn **global** patterns as opposed to **personalized** ones
    - learn a shared set of parameters for all students
  - fail to represent student knowledge in the **long run**
    - truncate long learning sequences, and train on shorter batches
    - lose dependence between student sequence batches
    - difficult on personalization

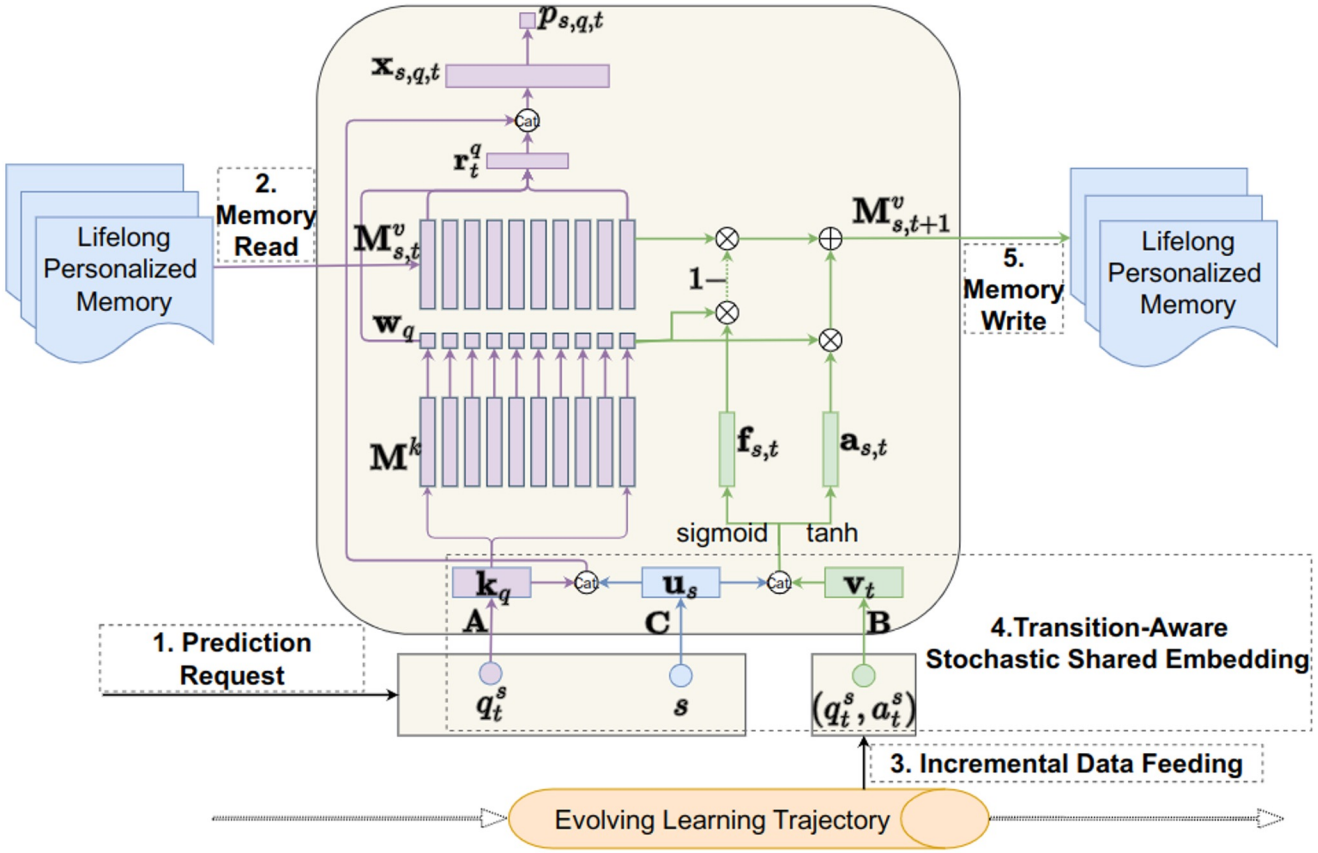
Our Proposed Solution: CPKT

UNIVERSITY AT ALBANY
State University of New York

- **Continuous Personalized Knowledge Tracing (CPKT)**:
  - personalized
    - personalized memory slots
    - personalized knowledge acquisition and forgetting patterns
  - continuous
    - online learning and inference paradigm
    - without truncating learning trajectories
  - generalizable
    - Transition-Aware Stochastic Shared Embedding (TA-SSE)
    - avoid overfitting

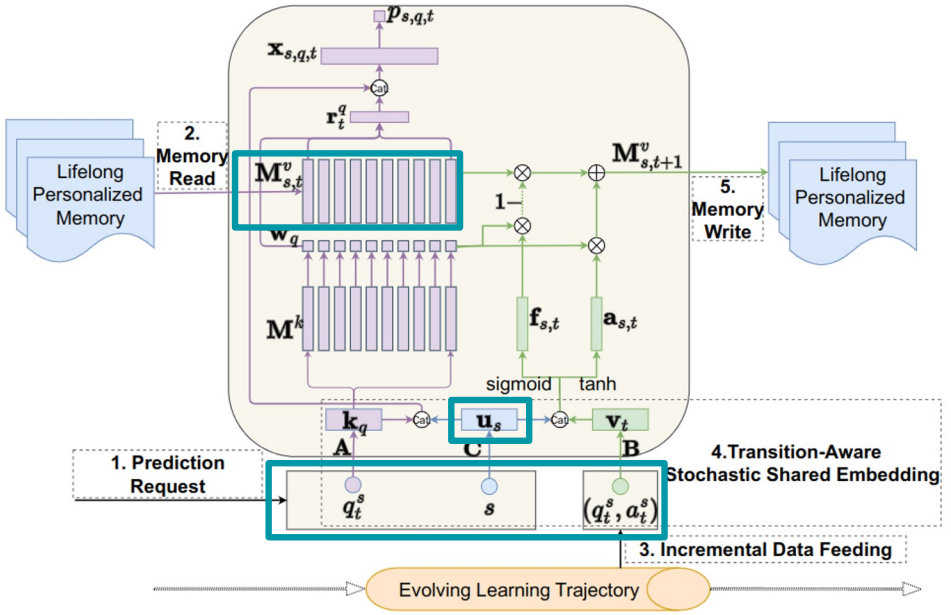
CPKT: Overall Architecture

UNIVERSITY AT ALBANY
State University of New York



CPKT: Personalized Memory Augmented Network

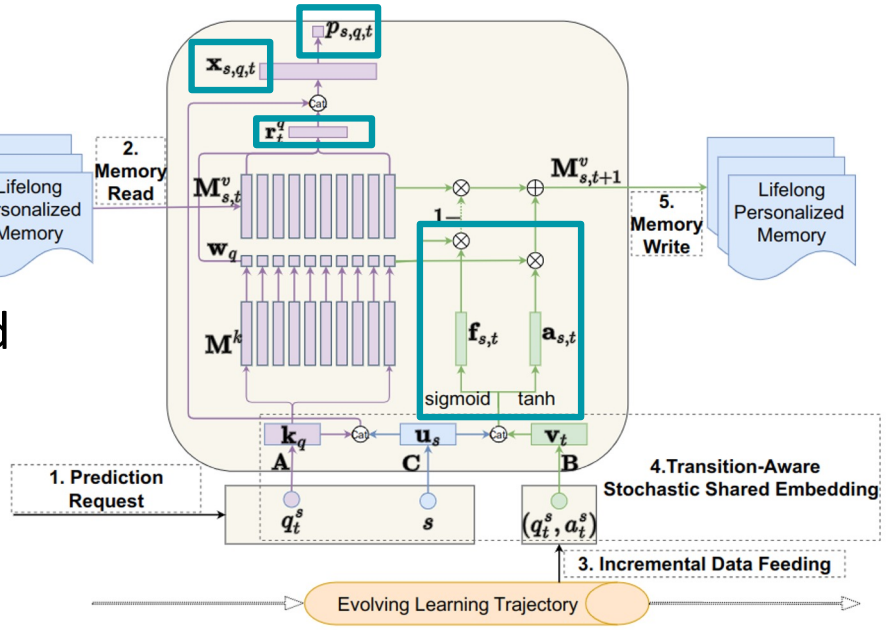
- Dynamic key-value memory networks with personalized parameters
  - global static latent concept features M^k
  - personalized dynamic value matrix $M^v_{s,t}$(student mastery state)
  - student embeddings $\boldsymbol{u}_s$



CPKT: Personalized Memory Augmented Network

- Personalized knowledge acquisition ($\boldsymbol{a}_s$) and forgetting ($\boldsymbol{f}_s$) rates to update student mastery over time
  - $\boldsymbol{f}_s = \sigma(w_3^T[v_t, u_s] + b_3)$
  - $M_{s,t}^v(i) = M_{s,t-1}^v(i) \otimes [\mathbf{1}^{d_h} - w_q(i)\boldsymbol{f}_s]$

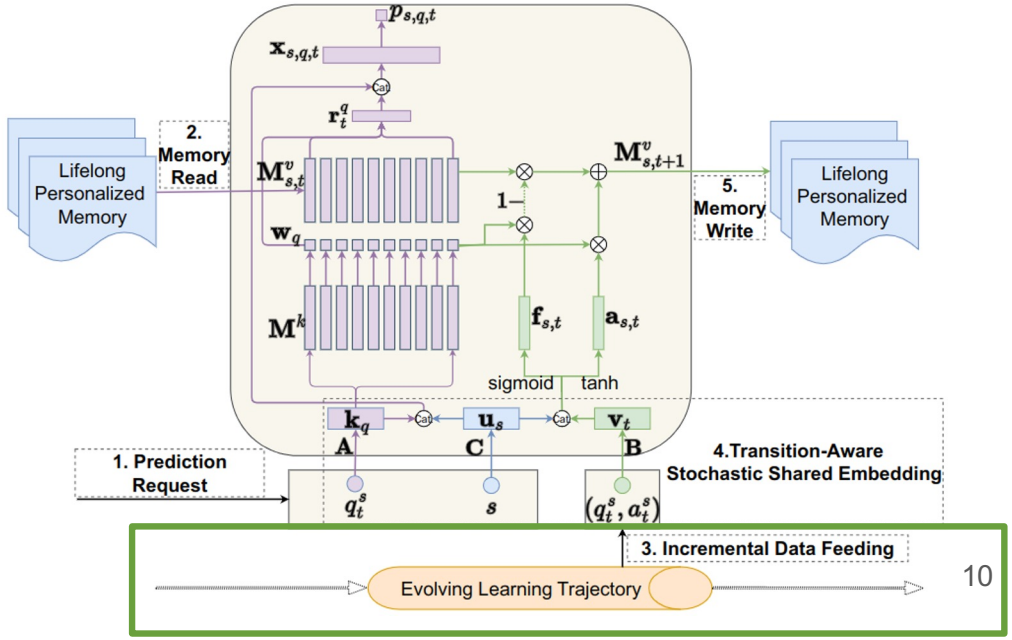
- Personalized knowledge level ($\boldsymbol{r}_{s,q,t}$) and ability-knowledge summary ($\boldsymbol{x}_{s,q,t}$)



CPKT: Continuous Knowledge Tracing

UNIVERSITY AT ALBANY
State University of New York

- Update model continuously over time with incremental data feeding
  - moving window to extract and train on the most recent historical records
  - avoid catastrophic interference problem

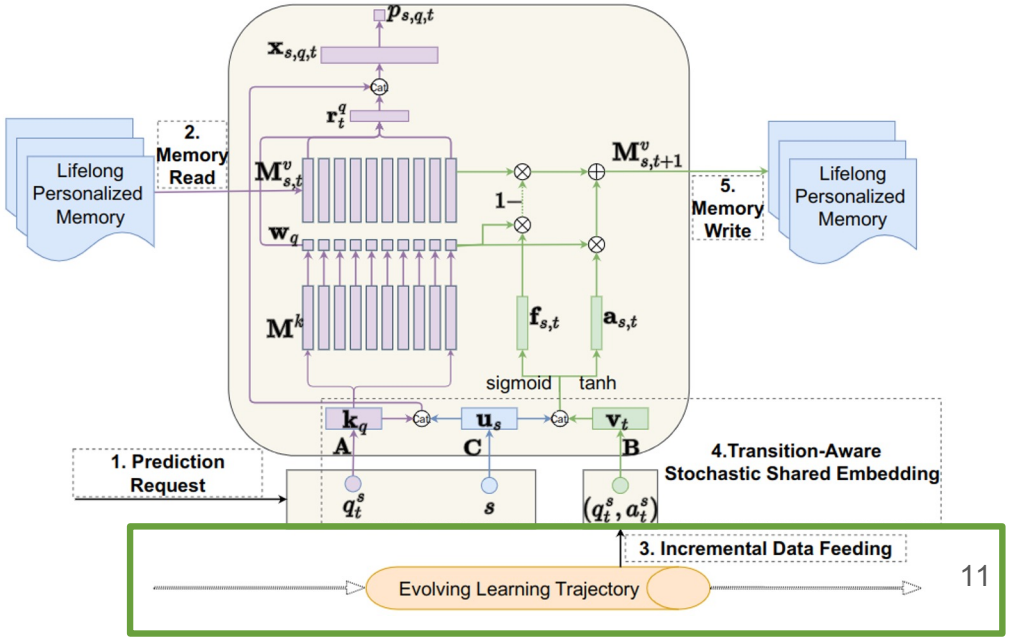


CPKT: Continuous Knowledge Tracing

UNIVERSITY AT ALBANY
State University of New York

- Update model continuously over time with incremental data feeding
  - moving window to extract and train on the most recent historical records
  - avoid catastrophic interference problem

- Caveat:
  - many parameters
  - easily overfitted



CPKT: Stochastic Shared Embedding to Rescue

- Data-driven regularization of embedding layers
- Stochastically swapping embeddings during the training
- Which embeddings to swap?
  - Transition-Aware Stochastic Shared Embedding (TA-SSE) for problems
  - random for students

CPKT: Transition-Aware Stochastic Shared Embedding

- Adds generalizability and avoids overfitting
  - generate item transition matrix T according to learning trajectories :

$$T_{i,j} = prob(j|i) = \frac{|i \to j|}{i}$$

  - stochastically swap problems in the learning trajectory based on T

Continuous Learning Algorithm Summary

Generate item transition matrix $\mathbf{T}$

For each test item index t do

For each problem response pair (q_i^s, a_i^s) in each student s's last t-H to t-1 responses

With probability ρ

replace q_i^s 's embedding and response with a random problem according to $\mathbf{T}$

For each student s with embedding u_s

With probability ρ

replace u_s with random student embedding u_z

Update the personalized parameters by forward and backward passes

Predict the target student response at t

Collect the target student response as new data

Update the transition matrix T

Increase t by 1

Three Research Questions

UNIVERSITY AT ALBANY
State University of New York

- RQ1. How is the model performance compared with state of-the-art baselines?

- RQ2. How do different proposed components affect its prediction performance?

- RQ3. How does the model perform on the users with different lengths of learning trajectories?

RQ1: CPKT Outperforms Baselines

UNIVERSITY AT ALBANY
State University of New York

Methods	MORF	ASSIST2015	EdNet	Junyi
	RMSE	AUC	AUC	AUC
DKT	0.1990±0.0087	0.7142±0.0029	0.6349±0.0048	0.8709±0.0072
DKVMN	0.1995±0.0067	0.7047±0.0034	0.6291±0.0070	0.8685±0.0084
SAKT	0.1975±0.0075	0.6997±0.0153	0.6296±0.0059	0.8208±0.0091
SAINT	0.2190±0.0186	0.6533±0.0135	0.5058±0.0070	0.8406±0.0078
AKT	0.2417±0.0112	0.6870±0.0159	0.6303±0.0070	0.8133±0.0152
IEKT	0.2481±0.0054	0.7204±0.0027	0.5980±0.0133	0.8721±0.0026
CPKT	**0.1752±0.0081**	**0.7274±0.0032**	**0.6558±0.0072**	**0.8802±0.0072**

RQ2: Ablation Study

- Both personalization and TA-SSE Help

Methods	MORF	ASSIST2015	EdNet	Junyi
	RMSE	AUC	AUC	AUC
CPKT-W/O-Pers	0.1919±0.0092	0.7202±0.0040	0.5745±0.0036	0.8546±0.0078
CPKT-W/O-SSE	0.1895±0.0067	0.7092±0.0049	0.5753±0.0058	0.8333±0.0051
CPKT	**0.1752±0.0081**	**0.7274±0.0032**	**0.6558±0.0072**	**0.8802±0.0072**

RQ3: Better Performance for Longer Trajectories

- Arrange all students based on their test sequence lengths (corresponding to trajectory lengths) into three groups with roughly equal sizes: short, medium, long.

Group	#Users	Range of Test Length	Mean AUC			P-Value of CPKT vs.	
			DKT	DKVMN	CPKT	DKT	DKVMN
Short	199	[0,110]	$0.64	0.6427	0.6321	p=0.8421	p=0.4767
Medium	210	[110,380]	0.6422	0.6363	0.644	p=0.7375	p=0.1398
Long	234	[380,900]	0.6413	0.6315	**0.6475**	**p=0.0745**	**p=1.51e-05**

Group	#Users	Range of Test Length	Mean BCE			P-Value of CPKT vs.	
			DKT	DKVMN	CPKT	DKT	DKVMN
Short	199	[0,110]	0.655	0.6884	**0.653**	p=0.8583	**p=0.0154**
Medium	210	[110,380]	0.6474	0.675	**0.6375**	**p=0.0475**	**p=2.87e-09**
Long	234	[380,900]	0.6562	0.6942	**0.6389**	**p=4.71e-05**	**p=1.70e-19**

Conclusions

Continuous Personalized Knowledge Tracing (CPKT):

- ★ could track personalized student knowledge
- ★ over long learning trajectories using an online learning and prediction paradigm
- ★ using a transition-aware stochastic shared embedding regularization method that could resolve the overfitting issues

Source code of CPKT: https://github.com/persai-lab/CIKM2023-CPKT
Contact: Chunpai Wang chunpaiwang@gmail.com

This paper is based upon work supported by the National Science Foundation under Grant No. 2047500.

PersAI onalized

UNIVERSITY AT ALBANY
State University of New York

NSF

Related Work

- Traditional Knowledge Tracing
  - probabilistic models (e.g. BKT) and logistic models (e.g. IRT, AFM)
  - works well on small dataset
  - allow personalization (learn student specific parameters)
- Deep Knowledge Tracing (KT)
  - works well on large dataset (e.g. DKT, DKVMN, SAKT)
  - aim to learn global patterns (learn a shared set of parameters for all students)
    - truncate long learning sequences, and train on shorter batches
    - lose dependence between student sequence batches
    - difficult on personalization
    - fail to represent student knowledge in the long term

Motivation

UNIVERSITY AT ALBANY
State University of New York

Online Education System

- abundant of practice problems with large topic varieties
- student can have a “lifelong” or continuous learning experience
  - can practice same topics many times
  - can learn a variety of topics over long periods of time
  - demand of personalized long-term learning experience

MIT OCW, edX, Future Learn, ALISON, UDACITY, coursera, KHAN ACADEMY, UoPeople, udemy, iversity

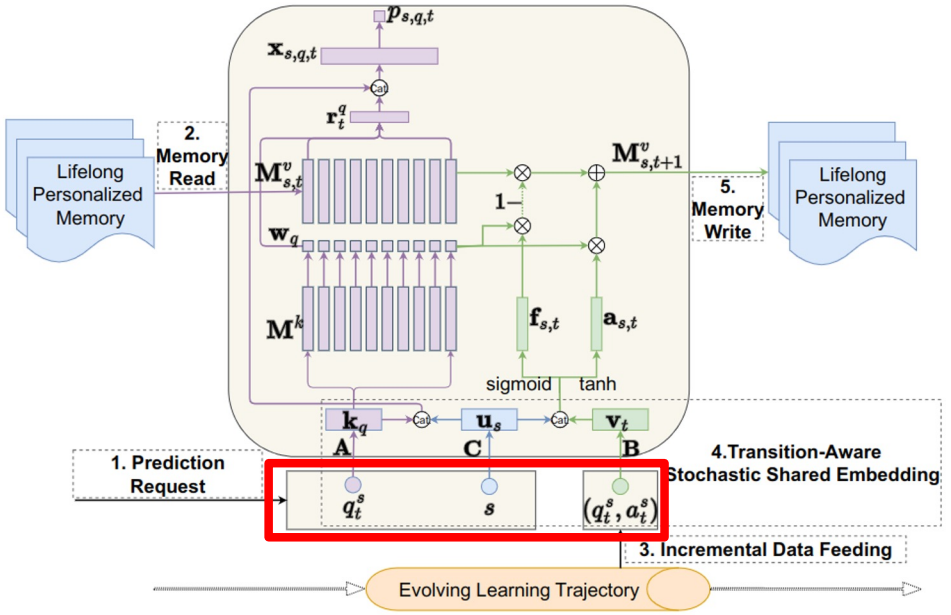
CPKT

UNIVERSITY AT ALBANY
State University of New York

- We propose **Continuous Personalized Knowledge Tracing (CPKT)**:
  - personalized KT
    - track individualized student knowledge and predict student performance
    - personalized memory slots to maintain learner's knowledge in a lifelong manner
    - personalized knowledge acquisition and forgetting patterns
  - continuous KT by online learning and inference paradigm
    - mimic the real world long-term continuous learning scenario
    - without truncating learning trajectories to train the model
  - incorporation of personalization and continuous model learning
    - transition-aware stochastic shared embedding
    - avoid overfitting

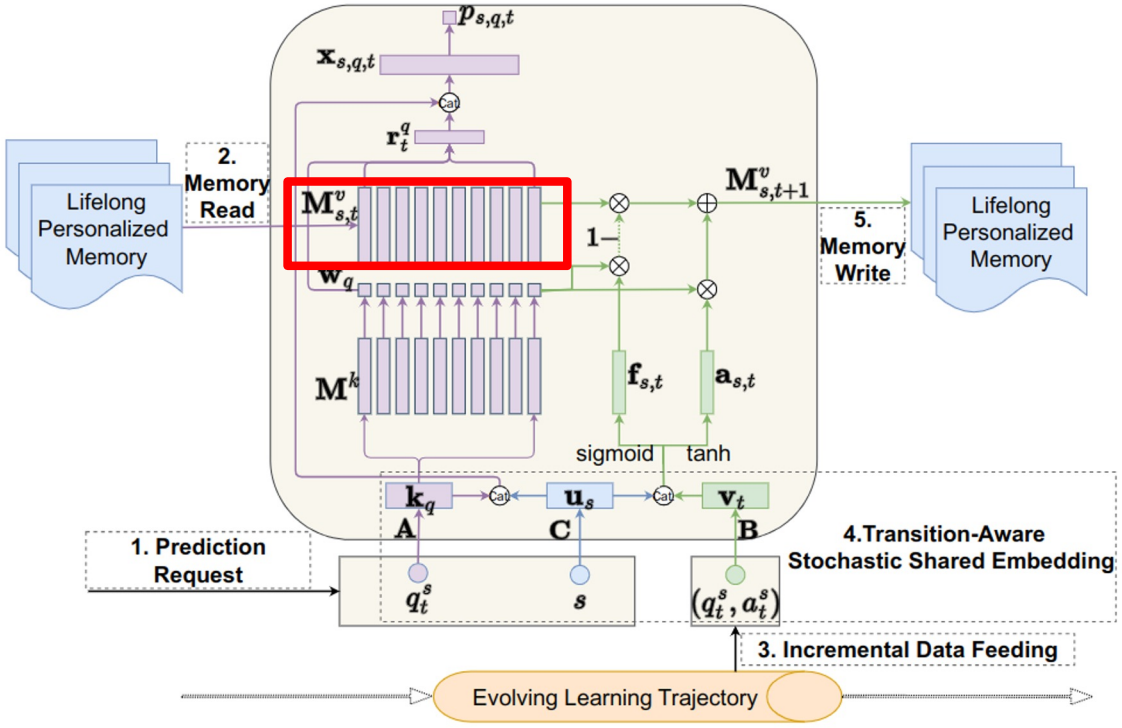
CPKT: Personalized Memory Augmented Network

- Personalized knowledge tracing:
  - given $\{(q_1^s, a_1^s), (q_2^s, a_2^s), \ldots, (q_{t-1}^s, a_{t-1}^s), q_t^s\}$, predict a_t^s.
  - the general deep learning-based models omit the superscript s in the context and do not differentiate distinct students' historical records.



CPKT: Personalized Memory Augmented Network

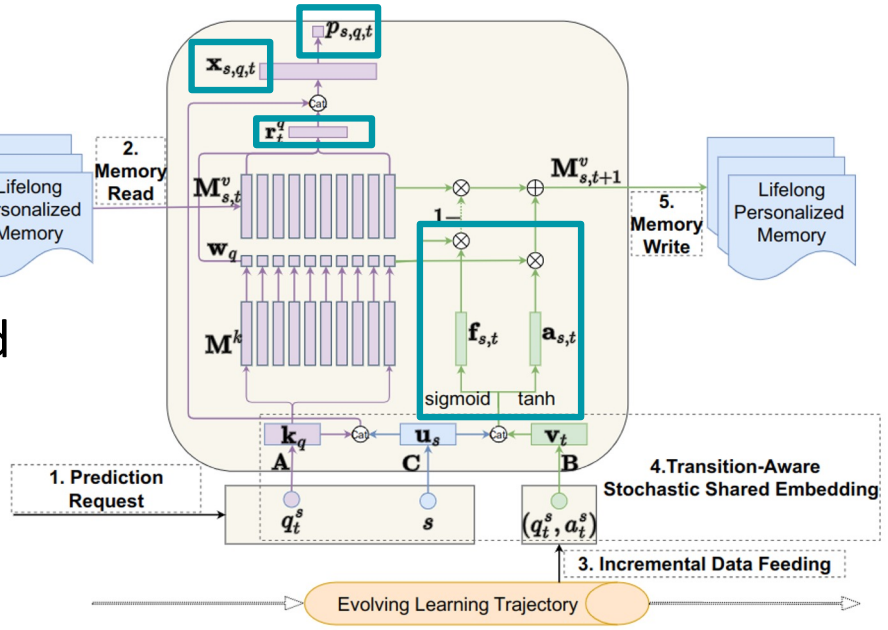
- Personalized Memory Slots
  - maintain a dynamic value matrix $\mathbf{M}_{s,t}^{v}$ for each student



CPKT: Personalized Memory Augmented Network

- Personalized knowledge acquisition ($\boldsymbol{a}_{s,t}$) and forgetting ($\boldsymbol{f}_{s,t}$) rates to update student mastery over time
  - $\boldsymbol{f}_{s,t} = \sigma(w_3^T[v_t, u_s] + b_3)$
  - $M_{s,t}^v(i) = M_{s,t-1}^v(i) \otimes [\mathbf{1}^{d_h} - w_q(i)\boldsymbol{f}_{s,t}]$

- Personalized knowledge level ($\boldsymbol{r}_{s,q,t}$) and ability-knowledge summary ($\boldsymbol{x}_{s,q,t}$)

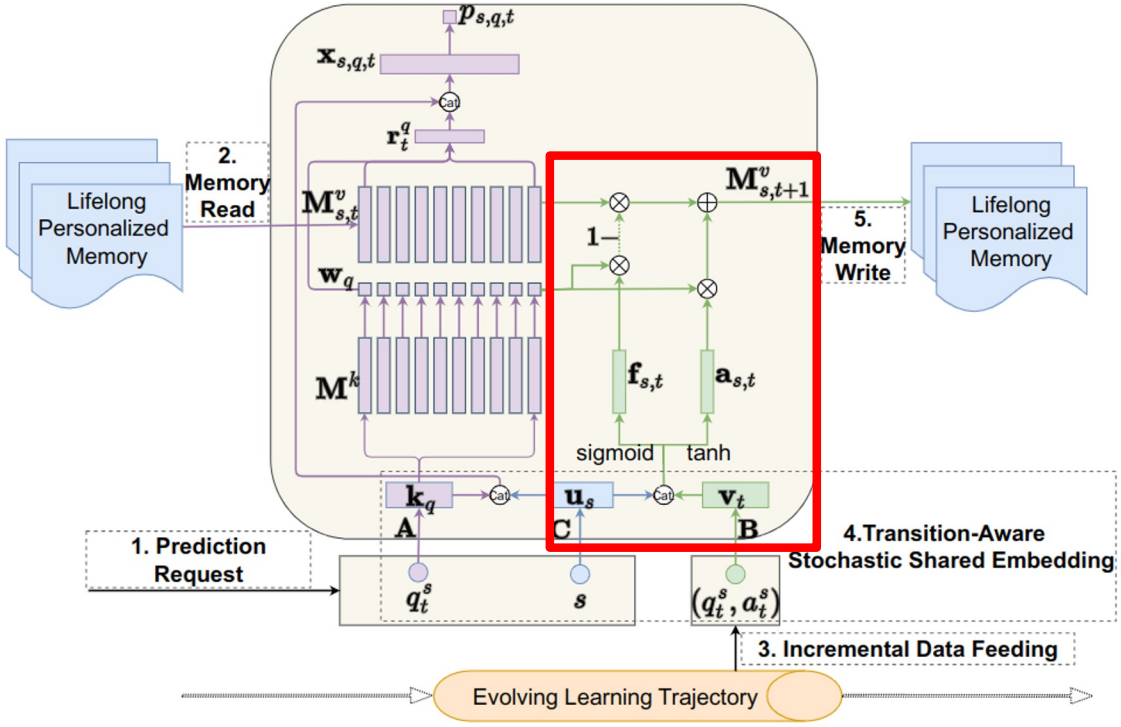


CPKT: Personalized Memory Augmented Network

- Personalized Knowledge Acquisition and Forgetting

$$\mathbf{a}_{s,t} = \mathrm{Tanh}\left(\mathbf{W}_4^{\top}[\mathbf{v}_t, \mathbf{u}_s] + \mathbf{b}_4\right)^T$$

$$\mathbf{M}_{s,t}^{v}(i) = \tilde{\mathbf{M}}_{s,t-1}^{v}(i) + w_q(i)\mathbf{a}_{s,t}$$



CPKT: Stochastic Shared Embedding to Rescue

- Stochastic Shared Embedding (SSE) [1]
  - data-driven regularization of embedding layers
  - stochastically swapping similar item embeddings during the training
  - implicitly adds exponentially many distinct reordering layers above the embedding layer and leads to exponentially many models trained at the same time
  - the loss landscape with SSE regularization becomes smoother and leads to better generalization.
  - requires an auxiliary knowledge graph to compute the switching probability distribution

[1] Liwei Wu, Shuqing Li, Cho-Jui Hsieh, and James L Sharpnack. 2019. Stochastic Shared Embeddings: Data-driven Regularization of Embedding Layers. Advances in Neural Information Processing Systems 32 (2019).

CPKT: Transition-Aware Stochastic Shared Embedding

- Adds generalizability and avoids overfitting
  - propose to generate and use a transition matrix based on the student learning trajectories.
  - learning trajectories typically contain some information on the learning material similarities.
  - better generalization.
  - Q by Q transition matrix T:

$$T_{ij} = \text{prob}(j \mid i) = \frac{|i \rightarrow j|}{|i|}$$

Algorithm

Algorithm 1: CPKT

Input: Observed student responses Ω_{obs}, including other students' and the target student's historical responses. Hyperparameter ρ denotes by SSE threshold and H denotes by time window size.

Sort each student's responses by the timestamp.

Generate the learning transition matrix $\mathbf{T}$ based on all observed learning trajectories.

Generate a dictionary $\mathcal{D}$ that stores each problem as key and a list of corresponding observed score or correctness from all students as value.

for *each testing time index* t **do**

  Extract each student's responses between time index $t-H$ and $t-1$, denotes by $\Omega_{t-H}^{t-1} = \{(q_i^s, a_i^s)\}_{i=t-H}^{i=t-1}$.

  Feed Ω_{t-H}^{t-1} along with student IDs into model.

  for *each problem* q_i^s *and corresponding interaction* $(q_i^s, a_i^s) \in \Omega_{t-H}^{t-1}$ **do**

    Identify the problem embedding $\mathbf{k}_i$ and interaction embedding $\mathbf{v}_i$.

    Generate a random number $\gamma \in [0, 1]$.

    if $\gamma < \rho$ **then**

      Replace $\mathbf{k}_i$ with $\mathbf{k}_j$, where $j \sim T_{ij} = prob(j \mid i)$

      Randomly sample a response a_j for problem q_j from $\mathcal{D}$.

      Identify the interaction embedding $\mathbf{v}_j$ for (q_j, a_j), and replace $\mathbf{v}_i$ with $\mathbf{v}_j$.

    end

  end

Algorithm

```
    for each student s do
        Identify the student embedding u_s.
        Generate a random number γ ∈ [0, 1].
        if γ < ρ then
            Randomly sample a student z from all students.
            Identify the student embedding u_z.
            Replace u_s with u_z.
        end
    end
    Forward and backward pass with the new embeddings
      to train the model by minimizing the training loss.
    Predict the target student's response at time t.
    Collect the target student's new response into Ω_obs.
    Update the transition matrix T as well as 𝒟.
    Increase the testing time index by 1.
end
```

- RMSE (H denotes the sliding window size)

$$\ell_{RMSE} = \sqrt{\frac{\sum_s \sum_t^{t+H} (a_t^s - p_t^s)^2}{n}}$$

- Binary Cross Entropy

$$\ell_{BCE} = -\sum_s \sum_t^{t+H} \left(a_t^s \log p_t^s + (1 - a_t^s) \log (1 - p_t^s)\right)$$

Experimental Setup

UNIVERSITY AT ALBANY
State University of New York

5-Fold User Stratified Cross Validation

- Observe all historical records for all training users.
- Observe all records before a preset threshold time index for all testing users.
- Predict test users' performance after the threshold time index.
- The testing records will be uncovered over time, along with the rolling prediction.
- Set the threshold index at roughly 10% of the maximum sequence lengths for each data.

Descriptive Statistics of 4 Real World Datasets.

Dataset	Users	Questions	Question Record	Mean Question Responses	STD Question Responses	Correct Question Responses	Incorrect Question Responses	Max Sequence Length
MORF	686	10	12031	0.7763	0.2507	N/A	N/A	46
ASSIST2015	19840	100	683801	N/A	N/A	500379	183,422	1000
EdNet	1000	11249	200931	N/A	N/A	118767	82,184	1,000
Junyi	1564	142	120984	N/A	N/A	86654	34328	1000

Experimental Setup

UNIVERSITY AT ALBANY
State University of New York

Table 4: Hyperparameters of CPKT.

Dataset	d_h	d_u	N	H
MORF	32	2	11	20
ASSIST2015	128	8	6	150
EdNet	128	8	41	150
Junyi	128	4	7	200

Descriptive Statistics of 4 Real World Datasets.

Dataset	Users	Questions	Question Record	Mean Question Responses	STD Question Responses	Correct Question Responses	Incorrect Question Responses	Max Sequence Length
MORF	686	10	12031	0.7763	0.2507	N/A	N/A	46
ASSIST2015	19840	100	683801	N/A	N/A	500379	183,422	1000
EdNet	1000	11249	200931	N/A	N/A	118767	82,184	1,000
Junyi	1564	142	120984	N/A	N/A	86654	34328	1000