KAR^{3}L: Knowledge-Aware Retrieval and Representations aid Retention and Learning in Students

Matthew Shu[1*] **Nishant Balepur**[2*] **Shi Feng**[3*] **Jordan Boyd-Graber**[2]
[1]Yale University [2]University of Maryland [3]George Washington University
matthewmshu@gmail.com, nbalepur@umd.edu

Abstract

Flashcard schedulers rely on 1) *student models* to predict the flashcards a student knows; and 2) *teaching policies* to pick which cards to show next via these predictions. Prior student models, however, just use study data like the student's past responses, ignoring the text on cards. We propose **content-aware scheduling**, the first schedulers exploiting flashcard content. To give the first evidence that such schedulers enhance student learning, we build KAR^{3}L, a simple but effective content-aware student model employing deep knowledge tracing (DKT), retrieval, and BERT to predict student recall. We train KAR^{3}L by collecting a new dataset of 123,143 study logs on diverse trivia questions. KAR^{3}L bests existing student models in AUC and calibration error. To ensure our improved predictions lead to better student learning, we create a novel delta-based teaching policy to deploy KAR^{3}L online. Based on 32 study paths from 27 users, KAR^{3}L improves testing throughput over SOTA, showing KAR^{3}L's strength and encouraging researchers to look beyond historical study data to fully capture student abilities.[1]

1 Introduction

Flashcards help students learn answers to questions across subjects like trivia, vocabulary, and law (Wissman et al., 2012). In education, a flashcard **scheduler** dictates when students review old flashcards and when they learn new ones (Reddy et al., 2016). Many schedulers use **student models** to predict the probability a student can recall a flashcard (Mozer et al., 2019). A **teaching policy** then chooses the flashcards to show next based on recall predictions (Reddy et al., 2017). Teaching policies ensure personalized learning experiences by understanding a student's learning progress. Thus, every student model must use informative features to capture a student's knowledge state (Brusilovsky et al., 2015; Chrysafiadi et al., 2015).

To predict recall, existing student models (Settles and Meeder, 2016; Tabibian et al., 2019) just use data from the student's study history, like their past answers and time since last review. This data models student behavior, but it ignores a key aspect of the card: its textual content. Modeling the relations across flashcard content can enable student models to predict recall even on cards with no study data. For instance, if a student studies the question "Who was the *first* U.S. president" for the first time, existing student models cannot discriminatively predict if the student knows the answer, as the card lacks study data. However, if the student already studied "Who was the *second* U.S. president" and always answered it correctly, we can infer that the student knows U.S. presidents and likely can recall the first U.S. president. Existing schedulers cannot make these semantic inferences, limiting their ability to predict and schedule flashcards with no study data.

We propose **content-aware flashcard schedulers**—the first schedulers that exploit flashcard content. We connect this paradigm to Deep Knowledge Tracing (Piech et al., 2015, DKT), which embeds student study histories (Shin et al., 2021) to predict study correctness. Some DKT models, like LM-KT (Srivastava and Goodman, 2021), enhance these embeddings with language models, providing a strong basis for semantic inferences. While DKT models show promise on offline benchmarks, their adoption in real-world flashcard scheduling systems is absent, as there lacks concrete evidence that such models can improve learning (§3). Thus, our goal is to design a baseline content-aware scheduler using DKT and be the first to show that this model can successfully enhance student learning, motivating the benefits and potential of the paradigm.

Towards content-aware scheduling, we develop **KAR^{3}L**, a DKT student model using **K**nowledge-**A**ware **R**etrieval and **R**epresentations for **R**etention

[*] Equal contribution.

[1] Our data and code are available at https://github.com/Pinafore/fact-repetition.

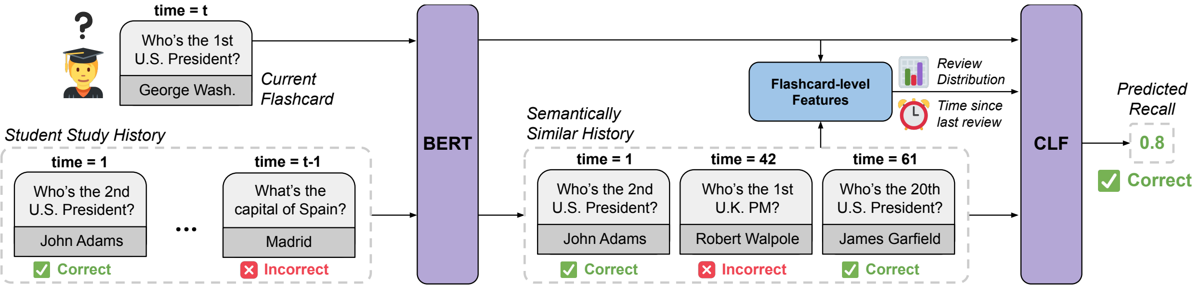


Figure 1: Overview of KAR^{3}L. Given a current flashcard and the student's study history as inputs, KAR^{3}L first uses a BERT retriever to obtain the most semantically similar cards from the study history. Next, the BERT embeddings of these retrieved flashcards, the embedding of the current flashcard, and flashcard-level features (e.g. time since last review), are fed through a classifier (CLF) to predict if the student knows the answer to the current flashcard.

and **L**earning (Figure 1). While DKT models usually embed the student's full study history, KAR^{3}L uses a BERT retriever (Lee et al., 2019) to find a subset of semantically similar cards in the study history, making KAR^{3}L the first retrieval-based student model (§4.1). Retrieval yields just study history with similar content to the current card, allowing KAR^{3}L to jointly improve efficiency and omit noisy parts of the history that distract the representation of the student's knowledge (§6.4). KAR^{3}L then encodes the retrieved history and current flashcard with the student's study data and BERT. This lets KAR^{3}L accurately predict recall even on unseen cards, as KAR^{3}L can infer student knowledge via study data from semantically related flashcards.

A general student model like KAR^{3}L cannot be trained on existing datasets, as they do not release flashcard text or have narrow domains (Settles and Meeder, 2016; Selent et al., 2016). Thus, we curate flashcards using content-rich trivia questions on diverse topics like fine arts, history, and pop culture. We develop a flashcard app and deploy our cards to 543 learners (§5)—forming a new dataset of 123,143 study logs to train content-aware models. Compared to baselines (Settles and Meeder, 2016; Srivastava and Goodman, 2021; Ye et al., 2022), KAR^{3}L gives the most accurate and well-calibrated recall predictions, demonstrating the offline strength of content-aware student models (§6).

Researchers typically stop at offline evaluation to claim scheduler superiority, but this does not assess a scheduler's main goal: enhancing student learning. To ensure the improved predictions in KAR^{3}L translate to better learning outcomes for students, we design the first online user study comparison with FSRS, the SOTA scheduler (§7). To do so, we create a novel delta-based teaching policy that picks flashcards predicted to enhance learning after a specified time delta (§7.1). We equip the KAR^{3}L student model with this policy to form KAR^{3}L$+\Delta$, the first content-aware scheduler to aid learning.

We have 27 new users study in our app and test their medium-term learning : **response accuracy**, the number of new flashcards learned, and **response time**, the time taken to recall answers. In 32 six-day studies, KAR^{3}L$+\Delta$ maintains learning accuracy while reducing recall time, showing testing throughput gains over FSRS (§7.3). KAR^{3}L $+\Delta$, a baseline content-aware scheduler, bests SOTA, revealing the strength of KAR^{3}L, motivating future works to build better content-aware schedulers, and encouraging researchers to look beyond study data to capture student abilities. Our contributions are:
1) We introduce content-aware scheduling, the first flashcard schedulers that exploit flashcard content.
2) We implement KAR^{3}L, a simple but effective content-aware student model that uses DKT, BERT, and retrieval. KAR^{3}L employs a novel delta-based teaching policy to facilitate online scheduling.
3) We collect and release a dataset of 123,143 study logs from 543 users on diverse trivia flashcards.
4) We design an online evaluation on two facets of medium-term learning to prove that content-aware models like KAR^{3}L can effectively aid learning.

2 Related Work

Flashcard Scheduling: Flashcards help students recall answers to questions, ranging from vocabulary (Komachali and Khodareza, 2012; Sitompul, 2013) to medicine (Schmidmaier et al., 2011; Lu et al., 2021). The order and spacing of flashcards when studying strongly affect the student's ability to recall information in the future (e.g. exam time) (Kornell, 2009), leading to research in systems that optimally schedule flashcards. Early models like Leitner (Leitner, 1974) and SuperMemo-2 (Wozniak, 1990) use rule-based heuristics to pick review dates. Subsequent schedulers draw from cognitive theory, deploying teaching policies with student model predictions. Settles and Meeder (2016) design half-life regression (HLR) student models as trained power-law and exponential forgetting curves (Ebbinghaus, 1913), theoretical memory decay models derived from empirical research.

Later works extend HLR and build teaching policies to schedule flashcards based on learned parameters and predictions from forgetting curve student models. Reddy et al. (2017) optimize their teaching policy with reinforcement learning, while MEMORIZE (Tabibian et al., 2019) and SELECT (Upadhyay et al., 2020) minimize cost values to optimize review times. Recent models like FSRS and DHP enhance these methods by optimizing spaced repetition via stochastic shortest path algorithms and time series data (Ye et al., 2022; Su et al., 2023).

Notably, the student models in these schedulers cannot discriminate between unseen flashcards and must arbitrarily show new cards, such as ordering by cards' creation dates (Elmes, 2021). Content-aware scheduling—predicting student recall using the content on cards—addresses this limitation.

Deep Knowledge Tracing: KAR^{3}L uses deep knowledge tracing (DKT) (Shin et al., 2021; Abdelrahman et al., 2023): neural models that predict if a student has knowledge of a specified concept, subject, or study item (e.g. flashcard). The first DKT model used an RNN to capture the temporal dynamics of a student's study history (Piech et al., 2015). Later works refine this model by incorporating graph representations (Yang et al., 2021; Song et al., 2022), forgetting features (Chen et al., 2017; Nagatani et al., 2019), and memory structures (Abdelrahman and Wang, 2019; Gu et al., 2022).

Recent DKT models embed question content to improve study history encodings (Su et al., 2018; Yin et al., 2019; Lee et al., 2024). However, these models cannot be directly adapted for scheduling. LM-KT (Srivastava and Goodman, 2021) is based on GPT-2 so it can embed diverse questions, but its causal language modeling objective impedes the use of flashcard features, as next-token prediction models struggle to reason over numerical inputs (McLeish et al., 2024). Other text-aware models can use flashcard features but are designed for the math domain (Liu et al., 2019) and need content annotations of study items to discern relevant items in the study history. Conversely, KAR^{3}L is a classifier that can better encode numerical inputs via a feature embedding layer, and uses retrieval which can find relevant content without content annotations, combining the strengths of existing DKT models.

DKT models show promise when trained on offline benchmarks that assess if models can predict student study correctness, often the top-performing models (Abdelrahman et al., 2023). However, there is no concrete evidence that DKT models can or should be adopted to facilitate student learning in online applications like flashcard learning software. We bridge this gap by solving practical issues of DKT models to motivate their adoption, via: 1) the first retrieval-augmented DKT model to mitigate inefficiency and study history noise; 2) a new dataset of study logs on diverse questions to train content-aware models; and 3) a delta-based teaching policy and user study to prove DKT can enhance learning.

NLP in Education: Flashcard scheduling is just one educational task that benefits from NLP (Litman, 2016). Recent research in this area includes writing education content (Cui and Sachan, 2023), designing educational chatbots (Tyen et al., 2022; Liang et al., 2023; Siyan et al., 2024), exploring test-taking strategies like process of elimination (Ma and Du, 2023; Balepur et al., 2024), understanding student misunderstandings (Wang et al., 2024), and creating mnemonic devices (Lee and Lan, 2023). Our contributions, such as the introduction of content-aware scheduling and release of a new diverse study history dataset, will facilitate further research in NLP-powered educational tools.

3 Where Are Content-Aware Schedulers?

Our core method for content-aware scheduling (§4) uses BERT, DKT, and retrieval—techniques with proven benefits—so why have they not yet been adopted for scheduling? Our work identifies and addresses three key criteria needed to inspire broader adoption: 1) an effective DKT student model that does not need content annotations on study items and can efficiently be deployed (§4); 2) a large, diverse dataset to train content-aware models (§5); and 3) a user study to confirm content-aware schedulers benefit learning (§7). As a whole, these challenges indicate that the absence of content-aware schedulers stems from a lack of evidence showing that they meaningfully improve student learning.

We reveal that with the right modeling choices (§4), content-rich datasets (§5), and thorough user studies (§7), we can adeptly combine the strengths of language models, DKT, and retrieval to show that content-aware schedulers aid student learning.

4 KAR^{3}L Student Model Design

Our student model KAR^{3}L builds on Deep Knowledge Tracing (Naeini et al., 2015, DKT) and uses as inputs: 1) the flashcard f_t shown to the student at time t; and 2) a history of all past flashcards studied by the student $\mathcal{F} = \{f_1, f_2, ..., f_{t-1}\}$. We also assume a flashcard f can be mapped to study data $\mathcal{X}(f)$, like the student's total correct responses to f and its time since last review. Using these inputs, KAR^{3}L predicts correctness $a_t \in \{0, 1\}$ denoting whether the student will answer card f_t correctly.

KAR^{3}L predicts a_t in two steps (Figure 1). First, KAR^{3}L uses a BERT retriever $p(f_t \,|\, f_i)$ to find the top-k flashcards $\mathcal{F}' \subseteq \mathcal{F}$ most semantically relevant to f_t. Next, KAR^{3}L feeds f_t and $\mathcal{F}'$ as inputs to a classifier $p(a_t \,|\, f, \mathcal{F}')$, which represents the current flashcard f and each retrieved flashcard in $\mathcal{F}'$ with BERT embeddings and the features from $\mathcal{X}(z)$. We describe both of these steps next.

4.1 Flashcard Retrieval

A student's study history $\mathcal{F}$ is often long and diverse. DKT models encode all of $\mathcal{F}$, degrading efficiency, and their f_t predictions may also be worse if they use study data in $\mathcal{F}$ that is unlike f_t (§6.4). For example, if a student studies both math and history, embedding all of $\mathcal{F}$ may worsen predictions on history cards, as the study data on math is irrelevant. DKT datasets have predefined subject or knowledge component labels on study items to help models discern relevance (Koedinger et al., 2012). However, flashcard apps support user-created cards that may not fall into these categories, so we assume no access to study item labels.

To solve these issues, we design the first retrieval-augmented student model. In generation, retrievers limit noise in large corpora and improve efficiency by picking a subset of relevant items (Lewis et al., 2020; Balepur et al., 2023). We retrieve the flashcards $\mathcal{F}'$ from the student's study history $\mathcal{F}$ with the most similar semantic representations to the current flashcard f_t. This ensures KAR^{3}L makes predictions using the study history most similar to the card f_t, reducing the total cards to embed and study history noise (§6.4) without study item annotations. Further, §6.6 shows that retrieval has the additional benefit of uncovering concepts in flashcards.

We obtain the semantic similarity between each study history card $f_i \in \mathcal{F}$ and the current flashcard f_t via the dot product of f_i and f_t, represented by pretrained BERT (Devlin et al., 2019) embeddings:

$$\mathbf{d}(f_i) = \text{BERT}(f_i), \tag{1}$$

$$\mathbf{q}(f_t) = \text{BERT}(f_t), \tag{2}$$

$$p(f_t \,|\, f_i) \sim \mathbf{d}(f_i)^T \mathbf{q}(f_t). \tag{3}$$

Maximum Inner-Product Search (Shrivastava and Li, 2014) finds the k-highest values for $p(f_t \,|\, f_i)$, forming the top-k relevant cards $\mathcal{F}' \subseteq \mathcal{F}$ to f_t.

4.2 Feature Representation

After finding the semantically relevant flashcards $\mathcal{F}'$, we represent the flashcards in the history $f_i \in \mathcal{F}'$ and the current flashcard f_t with features for predicting student recall. Along with BERT embeddings from §4.1, we also use the flashcard-level heuristics from prior student models (Settles and Meeder, 2016; Tabibian et al., 2019). We define $\mathcal{X}(f)$ as a set of features that represent the study data on flashcard f, such as the distribution of the student's past responses to f and the time since its last review. We detail all features in Appendix A.2.

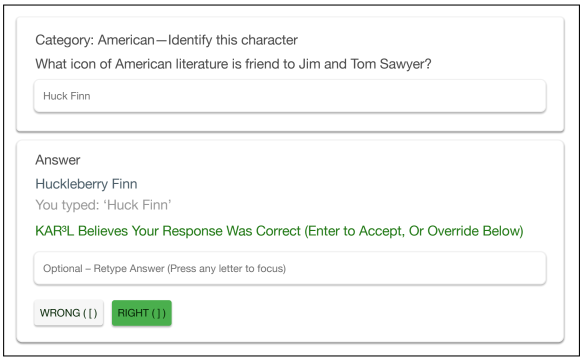
Using BERT embeddings and study data as features, we train a classifier $p(a_t \,|\, f_t, \mathcal{F}')$ that uses the current card f_t and retrieved history $\mathcal{F}'$ to predict correctness a_t, which is 0/1 if the student recalls f_t incorrectly/correctly. We feed study data $\mathcal{X}(f)$ and BERT embeddings for study history flashcards $f_i \in \mathcal{F}'$ and the current flashcard f_t to a linear layer to compute a hidden state $h \in \mathbb{R}^{768}$. We then feed h to a final linear layer to predict a_t. We train $p(a_t \,|\, f_t, \mathcal{F}')$ to minimize the cross-entropy loss λ of predicting a_t, using input representations of cards f and $\mathcal{F}'$, via mini-batch gradient descent.

5 Training KAR^{3}L

We now train the KAR^{3}L student model. We describe our data collection platform (§5.1), create flashcards (§5.2), collect student study data (§5.3), and outline our training procedure (§5.4).

5.1 Data Collection Platform

If we want KAR^{3}L to cater to users studying varied topics, we must train on flashcards with diverse content. However, existing KT and flashcard datasets are domain-specific and include no flashcard text or just a single vocab word, making them unfit for our study. ASSISTments (Selent et al., 2016), the most widely-used KT dataset (Abdelrahman et al., 2023),



Category: American—Identify this character

What icon of American literature is friend to Jim and Tom Sawyer?

Huck Finn

Answer

Huckleberry Finn

You typed: ‘Huck Finn’

KAR^{3}L Believes Your Response Was Correct (Enter to Accept, Or Override Below)

Optional – Retype Answer (Press any letter to focus)

Figure 2: Screenshot from our web-based flashcard app after a user submits their answer to a literature flashcard.

is restricted to arithmetic; Duolingo's spaced repetition dataset (Settles and Meeder, 2016) and EdNet (Choi et al., 2020) focus on English language learning. Hence, to assess how KAR^{3}L captures semantic ties across diverse topics, we build our own platform to collect data from real learners, based on a web and mobile flashcard app (Figure 2).

5.2 Flashcard Creation

With diversity in mind, we turn to trivia questions from the QANTA dataset (Rodriguez et al., 2019) to make flashcards. QANTA questions are multi-sentence, where each subsequent sentence in the question points to the same answer with decreasing difficulty. The questions span eleven topics, including literature, history, fine arts, pop culture, and mythology. To create cards from these questions, we sample a subset of the dataset and use a sentence from the question as the front of the flashcard and its answer as the back of the card. In total, we curate 23,918 unique flashcards spanning 11 diverse topics (details and examples in Appendix A.1).

5.3 Study Data Collection

We recruit users to study cards (§5.2) in our app from English trivia forums. Users study with three schedulers; two are Leitner (Leitner, 1974) and SM-2 (Wozniak, 1990), popular heuristic schedulers. The third is a DKT model (Appendix A.3) trained on the Protobowl dataset (Boyd-Graber et al., 2012). Over four months, 543 users gave 123,143 study logs. Each log has the card f_t studied by the user at time t, past study data $\mathcal{X}(f_t)$ on f_t, and label a_t denoting if the student answered f_t correctly. Tables 6 and 7 show all dataset columns. Users are referenced by ID and we award $200 to the fifteen users who study the most flashcards.

5.4 Training Setup

We sort study records chronologically and use a 75/25 train/evaluation split for KAR^{3}L. We retrieve $k = 5$ items from the student's study history for all experiments using FAISS (Johnson et al., 2019). Tables 6, 7, and 8 list all study data features collected. We train KAR^{3}L using all features for ablations (§6.4) and the online evaluation (§7), and use just a subset of all features for the other offline experiments (§6), detailed in Appendix A.8.

6 Offline Evaluation

We evaluate KAR^{3}L on our dataset to highlight the offline strength of content-aware *student models*. This result motivates our design of a content-aware *scheduler* (§7), eventually allowing us to prove that such schedulers improve student learning (§7.3).

6.1 Baselines

We compare KAR^{3}L with popular student models:
1) Half Life Regression (HLR) models a_t with an exponential forgetting curve (Settles and Meeder, 2016). This curve is fit using the student's past study responses and time since the last study of f_t.
2) Leitner moves a flashcard f_t up or down numbered slots based on student responses to f (Leitner, 1974). We use five slots and calculate a_t as the slot position of f_t divided by the total number of slots.
3) Super Memo (SM)-2 predicts a_t like Leitner does, adjusting its value based on the student's proportion of successful recalls of f_t (Wozniak, 1990).
4) FSRS computes intermediate difficulty, stability, and retrievability scores to schedule flashcards (Ye et al., 2022). We take the retrievability score, defined as recall probability, as a prediction for a_t.
5) LM-KT is a representative DKT model using language models (GPT-2 Med), like KAR^{3}L (Srivastava and Goodman, 2021). Via causal language modeling, it predicts a_t with the input sequence $\mathcal{F}$.
5) GPT-3.5 (Ouyang et al., 2022) is five-shot prompted to predict a_t based on the five most recent study items in the student's study history $\mathcal{F}$. We use `gpt-3.5-turbo` and 0 temperature.

6.2 Evaluation Metrics

A student model's training objective is to give binary predictions for whether a student can recall a flashcard, but a student's familiarity with flashcards is not binary but on a continuous spectrum. Thus, a strong student model must be able to discern cards that are familiar or unfamiliar to students across a

	Seen Cards				*Unseen Cards*			
Model	AUC (↑)	ECE (↓)	Acc Correct (↑)	Acc Incorrect (↑)	AUC (↑)	ECE (↓)	Acc Correct (↑)	Acc Incorrect (↑)
HLR	0.370	0.387	0.738	0.144	-	-	-	-
Leitner	0.752	0.234	0.833	0.380	-	-	-	-
SM-2	0.660	0.200	0.826	0.419	-	-	-	-
FSRS	0.752	0.111	0.921	**0.524**	-	-	-	-
LM-KT	0.658	0.285	0.805	0.327	0.684	0.234	0.667	0.590
GPT-3.5	0.427	0.544	0.613	0.241	0.519	0.481	0.467	0.571
KAR^{3}L	**0.864**	**0.091**	**0.980**	0.250	**0.786**	**0.085**	**0.680**	**0.740**

Table 1: Student model ability to predict which flashcards students know (Accuracy when Correct/Incorrect, AUC) and how well they know them (ECE). ↑ (and ↓) denote if higher (or lower) scores are better. Best scores are in **bold**. KAR^{3}L outperforms baselines in 7/8 metrics, showcasing the offline strength of content-aware scheduling.

range of thresholds, rather than one binary prediction. We employ two metrics to capture this nuance: area under the ROC curve (**AUC**) and expected calibration error (Naeini et al., 2015, **ECE**); the former measures how well the model predicts across all thresholds, and the latter measures if predictions are well-calibrated. We also show accuracy when the student answered correctly ($a_t = 1$) and incorrectly ($a_t = 0$)—the models' training objective.

One benefit of content-aware models is that they can make semantic inferences and predict recall on cards without study data. To test this for KAR^{3}L, LM-KT, and GPT-3.5, we group our metrics by if the student has seen the current flashcard f_t or not.

6.3 Quantitative Comparison

We study KAR^{3}L's ability to model student recall (Table 1). On seen cards, KAR^{3}L surpasses models in all metrics except for accuracy on incorrect flashcards. In fact, all models struggle to predict when a student will *incorrectly* answer a flashcard they have already studied, with the best model (FSRS) barely exceeding random guessing (0.524). Thus, there is a clear opportunity to close this accuracy gap, which could be achieved by focusing on temporal dynamics of student memory (Ye et al., 2022). On unseen cards, KAR^{3}L bests LM-KT and GPT-3.5, the only other models able to predict recall on unseen cards, in all metrics. Further, for KAR^{3}L, there is an AUC gap between unseen and seen cards. Thus, student modeling on cards with no study data is still a challenge and may benefit from even larger LLMs to capture semantics (Appendix A.6).

KAR^{3}L underperforms in Acc Incorrect on seen cards, but we argue AUC and ECE are better indicators of a student model's abilities. Accuracy uses one cutoff (0.5) to decide if a student can recall a flashcard, but other cutoffs like 0.71 (Pavlik Jr et al., 2020) and 0.94 (Eglington and Pavlik Jr, 2020) are also valid. Thus, a robust student model must cater to varied cutoffs, an ability better assessed by AUC and ECE. Overall, KAR^{3}L is the strongest student model, with the best ability to discern which flashcards students know and how well they know them.

	Seen Cards		*Unseen Cards*	
Model	AUC (↑)	ECE (↓)	AUC (↑)	ECE (↓)
KAR^{3}L BERT	**0.780**	**0.108**	**0.740**	**0.124**
No BERT	0.692	0.127	0.612	0.205
No $\mathcal{X}(z)$	0.680	0.135	0.620	0.191

Table 2: KAR^{3}L versus ablations that discard BERT embeddings and discard flashcard-level features $\mathcal{X}(z)$. Both are useful for accurate and calibrated predictions.

Metric	$k = 0$	$k = 5$	$k = 10$	$k = 15$	$k = 20$
AUC (Seen)	**0.864**	**0.864**	0.861	0.851	0.851
ECE (Seen)	0.098	**0.091**	**0.091**	0.106	0.105
AUC (Unseen)	0.776	**0.786**	0.777	0.757	0.768
ECE (Unseen)	0.111	**0.085**	0.086	0.171	0.155

Table 3: Top-k AUC/ECE for new KAR^{3}L training runs (AUC same trend). $k = 20$ is max without OOM. Retrieval improves AUC and ECE on unseen cards, but retrieving too many cards can distract recall predictions.

6.4 Ablation Study

To attribute the performance gains in KAR^{3}L, we ablate its components. Adding BERT embeddings and flashcard-level study data both improve ECE and AUC (Table 2), showing that content and historical study data both enhance student modeling. Further, on unseen cards, where capturing semantic relations should be most useful, No BERT has the worst AUC and ECE. Thus, making semantic inferences across flashcards is a valuable strategy for student modeling on unseen flashcards.

We also assess how retrieval affects KAR^{3}L (Table 3). On seen cards, KAR^{3}L has similar ECE and AUC with ($k > 0$) and without ($k = 0$) retrieval, meaning that when study data exists, using just the current flashcard is sufficient to predict recall. On unseen cards, where study data is absent, retrieval

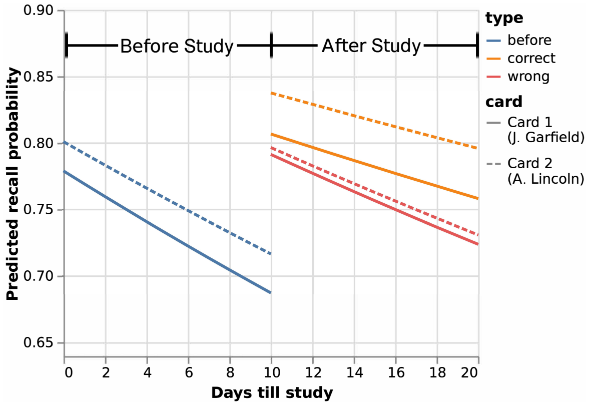


Figure 3: Forgetting curve for US history cards. When Card 1 is studied, KAR^{3}L's prediction of the semantically related Card 2 increases, despite not being studied.

boosts metrics from $k = 0$ to $k = 5$ and $k = 10$. However, retrieving too many flashcards ($k \geq 15$) on seen and unseen cards harms all metrics versus $k = 0$, likely because the excess flashcards are irrelevant, distracting KAR^{3}L. Thus, retrieval helps content-aware schedulers focus on short, relevant contexts for improved recall predictions without needing human annotations to represent relevance.

6.5 Forgetting Curve Analysis

Forgetting curves describe how a student's familiarity with a flashcard changes over time, measured through predicted recall probability (Ebbinghaus, 1913; Murre and Dros, 2015). While many student models explicitly fit to exponential or power-law forgetting curves (Upadhyay et al., 2020; Settles and Meeder, 2016), KAR^{3}L does not in exchange for more flexible memory representations. To simulate a forgetting curve in KAR^{3}L, we first define a set of times $\mathcal{T}$ (zero to twenty days, in one-day increments). Then, for each time $t \in \mathcal{T}$, KAR^{3}L predicts the recall of a flashcard f as if we were at time t, updating the features in $\mathcal{X}(f)$ accordingly.

To see how BERT enables KAR^{3}L to make semantic inferences across flashcards, we show simulated forgetting curves for two related flashcards. In Figure 3, Card 1 (James Garfield, 20th U.S. president) is studied once on day 0 and again on day 10. When Card 1 is recalled correctly on day 10, the recall prediction of Card 2 (Abraham Lincoln, 16th U.S. president) increases despite not being studied. This highlights the benefit of making semantic inferences across flashcards: KAR^{3}L intelligently adjusts recall predictions across semantically-related flashcards based on the learner's study of just one.

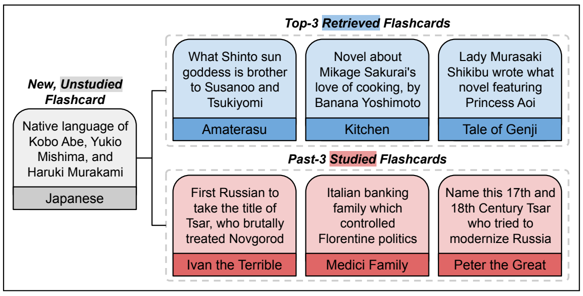

Top-3 Retrieved Flashcards
New, Unstudied Flashcard
Native language of Kobo Abe, Yukio Mishima, and Haruki Murakami
Japanese
What Shinto sun goddess is brother to Susanoo and Tsukiyomi
Amaterasu
Novel about Mikage Sakurai's love of cooking, by Banana Yoshimoto
Kitchen
Lady Murasaki Shikibu wrote what novel featuring Princess Aoi
Tale of Genji
Past-3 Studied Flashcards
First Russian to take the title of Tsar, who brutally treated Novgorod
Ivan the Terrible
Italian banking family which controlled Florentine politics
Medici Family
Name this 17th and 18th Century Tsar who tried to modernize Russia
Peter the Great

Figure 4: Top-3 retrieved vs past-3 studied flashcards when the user studies a new card on Japanese literature.

6.6 KAR^{3}L Retrieval Case Study

When using a subset of study history, some work uses the past-k cards instead of retrieving k cards (Pavlik et al., 2021). Using the past-k cards lowers AUC (Appendix A.5), but the following case study where the user sees a new card on Japanese authors also reveals the strength of retrieval (Figure 4). The top-3 cards retrieved by KAR^{3}L are topically related to the current card (Japanese novels and Shinto), but the past-3 cards seen by the user are on European history. Despite the topic shift, KAR^{3}L can still predict student recall, as the retrieved cards' study data reveal the student's knowledge of Japanese literature. As KAR^{3}L groups similar cards with retrieval, KAR^{3}L can also propose new concepts in the current flashcard that the student may know, like a concept of Japanese culture. Thus, KAR^{3}L recovers concepts without the content annotations used in DKT datasets (Shen et al., 2024), further showing the benefits of retrieval.

7 Online Evaluation

Our offline evaluation (§6) shows KAR^{3}L predicts recall accurately, but this does not capture the main goal of any educational tool: **enhancing learning**. Thus, while researchers often stop at offline evaluation to claim scheduler superiority, we now provide evidence that content-aware schedulers can improve student learning over FSRS, the SOTA scheduler. We propose a teaching policy that equips DKT student models like KAR^{3}L for scheduling (§7.1), design a test mode user study in our app to measure two facets of medium-term learning (§7.2), and assess student learning outcomes to compare the learning benefits of KAR^{3}L versus FSRS (§7.3).

7.1 A Delta-Based Teaching Policy for KAR^{3}L

After KAR^{3}L predicts if the student can recall a given flashcard, a teaching policy decides *when* to show this card next (Hunziker et al., 2019). DKT models directly predict recall, so the only existing compatible policy for DKT models like KAR^{3}L is **threshold-based**, which picks flashcards near specified retention levels like 0.9 (Ye et al., 2022) or 0.94 (Eglington and Pavlik Jr, 2020). A threshold-based policy for KAR^{3}L could run the classifier $p(a_t \mid f_t, \mathcal{F}')$ on every flashcard and pick those with predicted recall near a specified retention level. But without extra heuristics, this method will not schedule flashcards that KAR^{3}L predicts a student cannot recall or forgot (i.e. predicted recall of 0), even if studying such cards would aid subject mastery.

To solve this, we design a **delta-based teaching policy** that calculates how a student's recall improves over a given time interval Δ after studying a flashcard. Concretely, given a candidate flashcard f at time t, the policy uses KAR^{3}L to simulate what the student's recall will be after Δ time, $t' = t + \Delta$, based on if the student does or does not study f at t. By selecting cards with the highest differences in the recall predictions at t', which can include cards with initially low predicted recall, the policy aligns with our core goal of maximizing student recall.

When KAR^{3}L predicts a user can recall f correctly at time t with probability $p_t(f)$, we aim to compute by how much their future recall $p_{t'}(f)$ will increase if f is studied at t, called the **delta score:**

$$p_{t'}(f \mid \text{study } f \text{ at } t) - p_{t'}(f \mid \text{no study at } t). \quad (4)$$

To get $p_{t'}(f \mid \text{no study } f \text{ at } t)$, future recall probability when the user does not study, we obtain a prediction from KAR^{3}L as if we were at time t'.

For $p_{t'}(f \mid \text{study } f \text{ at } t)$, future recall when the user studies, we consider two possible outcomes of studying the card f at time t: the student's answer could be correct or incorrect. We weigh these two outcomes by the model's initial prediction $p_t(f)$:

$$\begin{aligned} &p_{t'}(f \mid \text{study } f \text{ at } t) := \\ &\qquad p_{t'}(f \mid \text{correct at } t) \cdot p_t(f) \qquad (5) \\ &\quad + p_{t'}(f \mid \text{incorrect at } t) \cdot (1 - p_t(f)). \end{aligned}$$

To find the two $p_{t'}$ values, we update the study data $\mathcal{X}(f)$ to query KAR^{3}L as if we are at time t' and if the student answered correctly/incorrectly at time t.

The cards with the n-highest delta scores (Eq. 4) are the next n cards scheduled. With a delta-based teaching policy, we ensure flashcards are picked for maximizing learning, preventing cards from being neglected solely due to having low predicted recall.

7.2 Test Mode Setup

We deploy FSRS and KAR^{3}L with our delta teaching policy (i.e. KAR^{3}L+Δ), with Δ equal to 1 day. We recruit users with the same procedure as §5.3 and award all users who complete our study $50. Studying how schedulers impact learning is difficult, as: (1) users want to study diverse subjects; (2) users have varying background knowledge; and (3) learning, a multifaceted process, is hard to quantify.

For (1), we create two test sets fixed across users, both with 20 manually-written cards from seven diverse QANTA bonus questions (Elgohary et al., 2018); bonus questions in QANTA are grouped as three distinct subquestions that test the same concepts, forming a testbed for evaluating the ability of content-aware schedulers to make semantic inferences. Users are tested on the same cards, which likely have some facts of interest to the user. For (2), we use a within-subject design (Lindsey et al., 2014), where users study with KAR^{3}L for one test set, and FSRS for the other. Hence, users try both schedulers, preserving background knowledge.

For (3), we focus on **medium-term learning**, which we define as the ability to learn flashcards over several days. This goal reflects the time span students report preparing for exams; students often study flashcards for five days or less and study no more than half their flashcards at a time (Wissman et al., 2012). Mirroring this, our users review ten cards from the test set of 20, scheduled by FSRS or KAR^{3}L + Δ, daily for five days. On day six, users complete a post-test and study all 20 test set cards.

Medium-term learning has not been deeply studied, so we use two measures of memory strength that have been explored in prior research: response accuracy and response time (MacLeod and Nelson, 1984; Maris and Van der Maas, 2012). We define **response accuracy** (Alshammari, 2019) as the gain in accuracy on the post-test from when users first see cards (pre-test). Higher response accuracy means that the user has learned more flashcards.

Response accuracy captures *how many* facts a user has learned, but not *how well* or *quickly* they can recall these facts. Thus, we define **response time** as the mean time needed for users to recall flashcard answers (Ericsson, 1985)—the time from when the card is viewed until when the answer is submitted. We calculate response time on two splits: flashcards answered correctly; and all flashcards studied. Lower response time on cards answered *correctly* means the user is more familiar

Model	Pre/Post-Test Acc	Response Corr/All	TTP
FSRS	0.41 / **0.88**	6.58 sec / 6.82 sec	2.58
KAR^{3}L +Δ	**0.42** / 0.86	**6.15 sec / 6.27* sec**	**2.74**

Table 4: Post-test metrics for KAR^{3}L+Δ vs FSRS users. Best in **bold**. * means t-test significance ($p < 0.05$). KAR^{3}L + Δ users have similar pre-test and post-test accuracy versus FSRS users but much lower post-test response time, enhancing testing throughput (TTP).

with cards they know, since they recall correct answers faster, while lower response time on *all* cards means that the user completes the post-test faster.

We combine response accuracy and time into a single post-test performance metric (Chignell et al., 2014) and define **testing throughput (TTP)** as the average total correct answers (20 $*$ post-test accuracy) divided by the average response time on all cards—cards answered correctly per second spent on the post-test. Higher TTP indicates that the student has a deeper understanding of the test mode flashcards, as they can correctly answer the same number of flashcards while taking less time.

7.3 User Study Results

We collect 32 six-day study sessions from 27 students with FSRS and KAR^{3}L + Δ (Table 4). Both schedulers help users achieve similar mastery in test mode, more than doubling pre-test to post-test accuracy (0.42 to 0.86). While response accuracy is similar, KAR^{3}L+Δ has lower post-test response time on cards answered correctly, suggesting that KAR^{3}L users are more familiar with the cards answered correctly. Finally, KAR^{3}L + Δ users maintain post-test accuracy while reducing recall time on all cards studied. As a result, KAR^{3}L + Δ users obtain higher testing throughput (TTP) than FSRS users, as users studying with KAR^{3}L answer a similar number of facts correctly in less time (2.74 vs 2.58 flashcards answered correctly per second).

Combining offline (Table 1) and online (Table 4) results, KAR^{3}L + Δ: 1) produces more accurate and calibrated recall predictions than FSRS on seen cards; 2) can predict recall on unseen cards, unlike FSRS; 3) matches FSRS in enhancing response accuracy; and 4) enables users to recall answers faster than FSRS, increasing testing throughput. Holistically, KAR^{3}L + Δ bests FSRS. Since KAR^{3}L + Δ is a baseline content-aware scheduler and it already rivals SOTA, content-aware scheduling is a promising paradigm, which we hope will motivate works to build better content-aware schedulers and look beyond study data to fully capture student abilities.

8 Conclusion

We introduce and successfully implement the first content-aware scheduler with KAR^{3}L + Δ, a simple but effective model using DKT, BERT, retrieval, and a novel delta-based teaching policy. Our offline evaluation on a newly-collected dataset shows KAR^{3}L provides accurate and well-calibrated recall predictions, while our online evaluation reveals KAR^{3}L improves testing throughput over SOTA. Thus, we give the first evidence that content-aware schedulers can be used to improve student learning.

Content-aware scheduling enhances personalization in learning tools, as models can infer student knowledge gaps through semantic inferences. Given this strength, we hope future works extend content to modalities beyond text, such as images or audio. In online evaluation, KAR^{3}L surpasses FSRS in testing throughput, but there are still many facets of learning that can be measured with online studies; these learning metrics could not only be used for evaluation but also training signals. In all, we show the strength of content-aware scheduling, which we hope will motivate works to look beyond *single* study items and also model relations *between* study items to fully capture student abilities.

9 Limitations

One limitation of KAR^{3}L, which is shared with all DKT models, is that our model uses BERT representations and a neural classifier, resulting in a slower inference time compared to student models which only use flashcard-level features. To minimize this limitation, we consider two design choices. First, rather than embedding the entire student history like existing DKT models, we perform top-k retrieval, enabling KAR^{3}L to have a consistent inference time that does not scale with the size of the student's study history. Second, we implement our retriever operations with FAISS (Johnson et al., 2019), an efficient vector database. This allows us to quickly look up the representation of each flashcard derived from QANTA, eliminating the time needed to tokenize and feed each flashcard through BERT, and efficiently perform Maximum Inner-Product Search. During our test mode user study, we did not receive complaints about the inference time or efficiency of KAR^{3}L.

Further, while KAR^{3}L is an effective framework for student modeling on our diverse dataset on trivia questions, we have not analyzed the performance of KAR^{3}L on domain-specific DKT benchmarks that do not have easily accessible textual content, such as EdNet (Choi et al., 2020) or ASSISTments[2] (Selent et al., 2016). Since the primary goal of this work was to design a general-purpose content-aware scheduler, we require a diverse and content-focused dataset to evaluate our model. In future works, when more datasets provide textual content, it would be interesting to study if the accuracy and calibration of KAR^{3}L is still strong, or how transfer learning techniques (Weiss et al., 2016) can help KAR^{3}L adapt to specific domains.

Finally, exploring more advanced retrievers and language models could improve the predictions of KAR^{3}L even further. As this was the first demonstration of content-aware scheduling and using retrieval for flashcard learning, we focused on the most basic language model and retriever: an off-the-shelf BERT model. We consider it a positive sign that this simple design leads to large offline AUC and ECE gains, which will motivate future works, including new iterations of KAR^{3}L, that improve content-aware scheduling with more advanced retrieval techniques and language models.

10 Ethical Considerations

The goal of adaptive student models like KAR^{3}L is to make personalized predictions about a student's knowledge level. When making such tailored predictions, it is crucial to ensure that these student models are not exploiting private information about the user. To ensure that models trained on our dataset do not succumb to this risk, our dataset only identifies users by a numerical ID. Further, our data collection and test mode user study were both approved by an Institutional Review Board (IRB), allowing us to fully address any potential risks of our study. All users were compensated based on a raffle system, and could win either $10, $15, or $50 based on the study. In our advertisements, it was made clear to users before signing up that they would be part of a research study.

Acknowledgements

We would like to thank members of the CLIP lab at the University of Maryland and external collaborators for their discussions of this work, including Jarrett Ye, Jack Wang, David Martinez, Dayeon Ki, Yu Hou, Yoonjoo Lee, Hyojung Han, Haozhe An, Erik Harpstead, Paulo Carvalho, Rachel Rudinger, Pedro Rodriguez, Denis Peskoff, Roger Craig, Sydney Montoya, and Wenyan Li. We additionally thank Nathaniel Klein for helping name KAR^{3}L. This material is based upon work supported by the National Science Foundation under Grant No. IIS-2403436 (Boyd-Graber) and DGE-2236417 (Balepur). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Cloud computing resources were made possible by a gift from Adobe Research.

References

Ghodai Abdelrahman and Qing Wang. 2019. Knowledge tracing with sequential key-value memory networks. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pages 175–184.

Ghodai Abdelrahman, Qing Wang, and Bernardo Nunes. 2023. Knowledge tracing: A survey. *ACM Computing Surveys*, 55(11).

Mohammad T Alshammari. 2019. Design and learning effectiveness evaluation of gamification in e-learning systems. *International Journal of Advanced Computer Science and Applications*, 10(9).

Nishant Balepur, Jie Huang, and Kevin Chang. 2023. Expository text generation: Imitate, retrieve, paraphrase. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11896–11919, Singapore. Association for Computational Linguistics.

Nishant Balepur, Shramay Palta, and Rachel Rudinger. 2024. It's not easy being wrong: Large language models struggle with process of elimination reasoning. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 10143–10166, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Jordan Boyd-Graber, Brianna Satinoff, He He, and Hal Daumé III. 2012. Besting the quiz master: Crowdsourcing incremental classification games. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 1290–1301.

Peter Brusilovsky, Sibel Somyürek, Julio Guerra, Roya Hosseini, Vladimir Zadorozhny, and Paula J Durlach. 2015. Open social student modeling for personalized learning. *IEEE Transactions on Emerging Topics in Computing*, 4(3):450–461.

[2] To obtain the textual content of ASSIStments, you must send an additional email for verification and agree to a Terms of Use before your request can even be processed. In contrast, our dataset will release the flashcard content for public use.

Yuying Chen, Qi Liu, Zhenya Huang, Le Wu, Enhong Chen, Run ze Wu, Yu Su, and Guoping Hu. 2017. Tracking knowledge proficiency of students with educational priors. *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*.

Mark Chignell, Tiffany Tong, Sachi Mizobuchi, and William Walmsley. 2014. Combining speed and accuracy into a global measure of performance. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 58, pages 1442–1446. SAGE Publications Sage CA: Los Angeles, CA.

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 *Artificial Intelligence in Education: 21st International Conference, AIED 2020, Ifrane, Morocco, July 6–10, 2020, Proceedings, Part II 21*, pages 69–73. Springer.

Konstantina Chrysafiadi, Maria Virvou, Konstantina Chrysafiadi, and Maria Virvou. 2015. Student modeling for personalized education: A review of the literature. *Advances in personalized web-based education*, pages 1–24.

Peng Cui and Mrinmaya Sachan. 2023. Adaptive and personalized exercise generation for online language learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10184–10198, Toronto, Canada. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Hermann Ebbinghaus. 1913. *Memory: A contribution to experimental psychology.* Teachers College Press, New York.

Luke G. Eglington and Philip I. Pavlik Jr. 2020. Optimizing practice scheduling requires quantitative tracking of individual item performance. *npj Science of Learning*, 5(1):1–10.

Ahmed Elgohary, Chen Zhao, and Jordan Boyd-Graber. 2018. Dataset and baselines for sequential open-domain question answering. In *Empirical Methods in Natural Language Processing*.

Damien Elmes. 2021. Anki. powerful, intelligent flashcards.

K Anders Ericsson. 1985. Memory skill. *Canadian Journal of Psychology/Revue canadienne de psychologie*, 39(2):188.

Hengnian Gu, Xiaoxiao Dong, and Dongdai Zhou. 2022. Dynamic key-value memory networks based on concept structure for knowledge tracing. In *2022 4th International Conference on Computer Science and Technologies in Education (CSTE)*, pages 290–294. IEEE.

Anette Hunziker, Yuxin Chen, Oisin Mac Aodha, Manuel Gomez Rodriguez, Andreas Krause, Pietro Perona, Yisong Yue, and Adish Singla. 2019. Teaching multiple concepts to a forgetful learner. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547.

Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, San Diega, CA, USA.

Kenneth R Koedinger, Albert T Corbett, and Charles Perfetti. 2012. The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive science*, 36(5):757–798.

Maryam Eslahcar Komachali and Mohammadreza Khodareza. 2012. The effect of using vocabulary flash card on iranian pre-university students' vocabulary knowledge. *International Education Studies*, 5(3):134–147.

Nate Kornell. 2009. Optimising learning using flashcards: Spacing is more effective than cramming. *Applied Cognitive Psychology: The Official Journal of the Society for Applied Research in Memory and Cognition*, 23(9):1297–1317.

Jaewook Lee and Andrew Lan. 2023. Smartphone: Exploring keyword mnemonic with auto-generated verbal and visual cues. In *International Conference on Artificial Intelligence in Education*, pages 16–27. Springer.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

Unggi Lee, Sungjun Yoon, Joon Seo Yun, Kyoungsoo Park, YoungHoon Jung, Damji Stratton, and Hyeoncheol Kim. 2024. Difficulty-focused contrastive learning for knowledge tracing with a large language model-based difficulty prediction. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4891–4900, Torino, Italia. ELRA and ICCL.

S Leitner. 1974. So lernt man lernen (herder, freiburg, germany).

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Kai-Hui Liang, Sam Davidson, Xun Yuan, Shehan Panditharatne, Chun-Yen Chen, Ryan Shea, Derek Pham, Yinghua Tan, Erik Voss, and Luke Fryer. 2023. ChatBack: Investigating methods of providing grammatical error feedback in a GUI-based language learning chatbot. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pages 83–99, Toronto, Canada. Association for Computational Linguistics.

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

Diane Litman. 2016. Natural language processing for enhancing teaching and learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30.

Qi Liu, Zhenya Huang, Yu Yin, Enhong Chen, Hui Xiong, Yu Su, and Guoping Hu. 2019. Ekt: Exercise-aware knowledge tracing for student performance prediction. *IEEE Transactions on Knowledge and Data Engineering*, 33(1):100–115.

Matthew Lu, John H Farhat, and Gary L Beck Dallaghan. 2021. Enhanced learning and retention of medical knowledge using the mobile flash card application anki. *Medical Science Educator*, 31(6):1975–1981.

Chenkai Ma and Xinya Du. 2023. Poe: Process of elimination for multiple choice reasoning. *arXiv preprint arXiv:2310.15575*.

Colin M MacLeod and Thomas O Nelson. 1984. Response latency and response accuracy as measures of memory. *Acta Psychologica*, 57(3):215–235.

Gunter Maris and Han Van der Maas. 2012. Speed-accuracy response models: Scoring rules based on response time and accuracy. *Psychometrika*, 77(4):615–633.

Sean McLeish, Arpit Bansal, Alex Stein, Neel Jain, John Kirchenbauer, Brian R Bartoldson, Bhavya Kailkhura, Abhinav Bhatele, Jonas Geiping, Avi Schwarzschild, et al. 2024. Transformers can do arithmetic with the right embeddings. *arXiv preprint arXiv:2405.17399*.

Michael C Mozer, Melody Wiseheart, and Timothy P Novikoff. 2019. Artificial intelligence to support human instruction. *Proceedings of the National Academy of Sciences*, 116(10):3953–3955.

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

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

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*, pages 3101–3107.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Philip I Pavlik, Luke G Eglington, and Leigh M Harrell-Williams. 2021. Logistic knowledge tracing: A constrained framework for learner modeling. *IEEE Transactions on Learning Technologies*, 14(5):624–639.

Philip I Pavlik Jr, Andrew M Olney, Amanda Banker, Luke Eglington, and Jeffrey Yarbro. 2020. The mobile fact and concept textbook system (mofacts). *Grantee Submission*.

Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J Guibas, and Jascha Sohl-Dickstein. 2015. Deep knowledge tracing. *Advances in neural information processing systems*, 28.

Siddharth Reddy, Igor Labutov, Siddhartha Banerjee, and Thorsten Joachims. 2016. Unbounded human learning: Optimal scheduling for spaced repetition. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1815–1824.

Siddharth Reddy, Sergey Levine, and Anca Dragan. 2017. Accelerating human learning with deep reinforcement learning. In *NIPS workshop: teaching machines, robots, and humans*.

Pedro Rodriguez, Shi Feng, Mohit Iyyer, He He, and Jordan L. Boyd-Graber. 2019. Quizbowl: The case for incremental question answering. *ArXiv*, abs/1904.04792.

Ralf Schmidmaier, Rene Ebersbach, Miriam Schiller, Inga Hege, Matthias Holzer, and Martin R Fischer. 2011. Using electronic flashcards to promote learning in medical students: retesting versus restudying. *Medical education*, 45(11):1101–1110.

Douglas Selent, Thanaporn Patikorn, and Neil Heffernan. 2016. Assistments dataset from multiple randomized controlled experiments. In *Proceedings of the Third (2016) ACM Conference on Learning@ Scale*, pages 181–184.

Burr Settles and Brendan Meeder. 2016. A trainable spaced repetition model for language learning. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 1848–1858.

Shuanghong Shen, Qi Liu, Zhenya Huang, Yonghe Zheng, Minghao Yin, Minjuan Wang, and Enhong Chen. 2024. A survey of knowledge tracing: Models, variants, and applications. *IEEE Transactions on Learning Technologies*.

Dongmin Shin, Yugeun Shim, Hangyeol Yu, Seewoo Lee, Byungsoo Kim, and Youngduck Choi. 2021. Saint+: Integrating temporal features for ednet correctness prediction. In *LAK21: 11th International Learning Analytics and Knowledge Conference*, pages 490–496.

Anshumali Shrivastava and Ping Li. 2014. Asymmetric lsh (alsh) for sublinear time maximum inner product search (mips). *Advances in neural information processing systems*, 27.

Elsa Yusrika Sitompul. 2013. Teaching vocabulary using flashcards and word list. *Journal of English and Education*, 1(1):52–58.

Li Siyan, Teresa Shao, Zhou Yu, and Julia Hirschberg. 2024. Eden: Empathetic dialogues for english learning. *arXiv preprint arXiv:2406.17982*.

Xiangyu Song, Jianxin Li, Qi Lei, Wei Zhao, Yunliang Chen, and Ajmal Mian. 2022. Bi-clkt: Bi-graph contrastive learning based knowledge tracing. *Knowledge-Based Systems*, 241:108274.

Megha Srivastava and Noah Goodman. 2021. Question generation for adaptive education. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 692–701, Online. Association for Computational Linguistics.

Jingyong Su, Junyao Ye, Liqiang Nie, Yilong Cao, and Yongyong Chen. 2023. Optimizing spaced repetition schedule by capturing the dynamics of memory. *IEEE Transactions on Knowledge and Data Engineering*.

Yu Su, Qingwen Liu, Qi Liu, Zhenya Huang, Yu Yin, Enhong Chen, Chris Ding, Si Wei, and Guoping Hu. 2018. Exercise-enhanced sequential modeling for student performance prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Behzad Tabibian, Utkarsh Upadhyay, Abir De, Ali Zarezade, Bernhard Schölkopf, and Manuel Gomez-Rodriguez. 2019. Enhancing human learning via spaced repetition optimization. *Proceedings of the National Academy of Sciences*, 116(10):3988–3993.

Gladys Tyen, Mark Brenchley, Andrew Caines, and Paula Buttery. 2022. Towards an open-domain chatbot for language practice. In *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)*, pages 234–249, Seattle, Washington. Association for Computational Linguistics.

Utkarsh Upadhyay, Graham Lancashire, Christoph Moser, and Manuel Gomez-Rodriguez. 2020. Large-scale randomized experiment reveals machine learning helps people learn and remember more effectively. *arXiv:2010.04430 [cs, stat]*. ArXiv: 2010.04430.

Rose Wang, Pawan Wirawarn, Omar Khattab, Noah Goodman, and Dorottya Demszky. 2024. Backtracing: Retrieving the cause of the query. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 722–735, St. Julian's, Malta. Association for Computational Linguistics.

Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. 2016. A survey of transfer learning. *Journal of Big data*, 3(1):1–40.

Kathryn T Wissman, Katherine A Rawson, and Mary A Pyc. 2012. How and when do students use flashcards? *Memory*, 20(6):568–579.

Piotr A Wozniak. 1990. Optimization of learning. *Unpublished master's thesis, Poznan University of Technology. Poznan, Poland*.

Yang Yang, Jian Shen, Yanru Qu, Yunfei Liu, Kerong Wang, Yaoming Zhu, Weinan Zhang, and Yong Yu. 2021. Gikt: a graph-based interaction model for knowledge tracing. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2020, Ghent, Belgium, September 14–18, 2020, Proceedings, Part I*, pages 299–315. Springer.

Junyao Ye, Jingyong Su, and Yilong Cao. 2022. A stochastic shortest path algorithm for optimizing spaced repetition scheduling. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22, page 4381–4390, New York, NY, USA. Association for Computing Machinery.

Yu Yin, Qi Liu, Zhenya Huang, Enhong Chen, Wei Tong, Shijin Wang, and Yu Su. 2019. Quesnet: A unified representation for heterogeneous test questions. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1328–1336.

A Appendix

A.1 Flashcard Creation and Examples

Each question q in QANTA consists of a list of clues $q = \{c_1, ..., c_n\}$ describing a single answer $\mathcal{A}$. To convert q into a flashcard f, we first map each unique $\mathcal{A}$ to its corresponding questions $\mathcal{Q} = \{q_1, ..., q_m\}$. Next, we take a random sample of clues $\mathcal{C} = \{c_1, ..., c_p\}$ from $\mathcal{Q}$. Using this data, we create f, where the clue $c \in \mathcal{C}$ is the front of f (i.e. the question shown to the user) and the answer $\mathcal{A}$ is the back of f. Repeating this process for the entire dataset, we obtain 23,918 unique flashcards spanning 11 diverse subjects.

In Table 5, we provide examples of the flashcards created from the QANTA dataset spanning 11 unique topics. The QANTA dataset is publicly available and we used the dataset with its intended research use. These flashcards are deployed into KAR^{3}L, where 543 participants produced 123,143 study records. In these records, 44.02 percent contain studies on new flashcards and students correctly answer the flashcard shown 71.80 percent of the time. On average, each user studied 226.78 flashcards during the data collection period.

The data used to create our flashcards is well-established and was not significantly altered in this work, so we did not check if the dataset uniquely identifies individuals or contains offensive content. All of our flashcards are written in English, containing facts primarily targeted toward students in the American high school and college education systems. The only additional data we collected beyond this that is released is the user's ID, the date and time of study, and which flashcards this user got correct and incorrect, which pose no harms.

In Tables 6, 7, and 8 we show the descriptions and summary statistics of the qualitative, quantitative, and time-based feature columns in our released dataset, respectively.

A.2 KAR^{3}L Classifier Features

Our released dataset also includes all the hand-picked features that KAR^{3}L uses along with BERT representations (Table 7). We normalize each feature using mean value and standard deviation computed on the training set. Features with IDs 7.17 through 7.22 are computed using the Leitner (Leitner, 1974) and SM-2 (Wozniak, 1990) algorithms.

In each study session, KAR^{3}L $+ \Delta$ additionally tracks the session-level variants of features 7.11-13. In test mode, features such as the user's accuracy are frozen to ensure that KAR^{3}L $+ \Delta$ does not have an unfair advantage compared to FSRS.

For all offline experiments except for the ablation, KAR^{3}L is trained on features 7.1, 7.4, 7.7, 7.10-14, and 7.16. For the ablation and online evaluation, KAR^{3}L is trained on all the features.

A.3 DKT Model for User Study

One of the models we in our user study to collect our flashcard dataset was a DKT model trained on the Protobowl dataset (Rodriguez et al., 2019). The Protobowl dataset contains QANTA questions with human responses to these questions, but this can be adapted to a flashcard dataset for content-aware models by selecting instances where users answered and practiced with the same question, treating these questions as study items. After converting the dataset to this format, we trained a DKT model similar to KAR^{3}L, but it does not use retrieval, and thus only uses BERT and flashcard features on the current flashcard to predict recall.

We designed this model to study whether it was possible to design a content-aware scheduler by only adapting existing datasets, rather than collecting a new one. While we found this to be feasible, training directly on a specialized flashcard dataset instead of using some dataset adaptation techniques would expectedly improve the performance of a content-aware flashcard scheduler. Thus, we opted to collect a new dataset and train a content-aware model (KAR^{3}L) on this dataset for the best possible results in the online user study.

A.4 Forgetting Curves

In this section, we provide more examples of forgetting curves. Unlike previous work which forces the student model to follow a specific forgetting function, we consider student modeling as predicting the binary outcome of each study. In Figure 5, we showcase the flexibility of our forgetting curves, and display visualizations of how the forgetting curve of a flashcard changes depending on its predicted recall probability.

A.5 Past-k Ablations

In Table 9, we present an additional ablation where we use top-k retrieval versus past-k retrieval in KAR^{3}L. We find that using top-k retrieval outperforms past-k retrieval on 3/4 metrics, suggesting that retrieval returns more relevant study items than just using the past-k items that the user has studied. Since the gap between these two methods is relatively small, we intuit that retrieval only shows clearer advantages over past-k retrieval when a user shifts topics while studying, like in the case study in §6.6. In fact, we find that only 5.7% of our dataset has topic shifts; this suggests that topic shifts are relatively rare, explaining why the performance gains are small. However, retrieval still boosts performance, so we argue that empirically it is an effective design choice.

A.6 Embedding Model Comparison

In Table 10, we compare the original KAR^{3}L BERT model, with and without top-5 retrieval, with a KAR^{3}L model using LLaMA embeddings. LLaMA returns an embedding of size 5192, so we were unable to train a version with retrieval using these embeddings due to resource constraints. We find that on seen cards, the AUC and ECE of both BERT variants outperforms LLaMA; this suggests that the model may be overweighting LLaMA embedding importance compared to the student's study data. While the BERT model with retrieval performs better on unseen cards, its performance on seen cards is similar to the model without retrieval. This aligns with out intuition that retrieval benefits unseen card prediction while past study data for seen cards is sufficient. However, on unseen cards, KARL LLaMA without retrieval has stronger than AUC than KARL BERT with retrieval, showing that models that can make stronger semantic inferences will have more accurate predictions on unseen cards. Note though that LLaMA has worse ECE, suggesting that larger LLMs are more overconfident when predicting student recall. This confirm the strength of our retrieval-augmented method, as gains cannot be achieved just by scaling the embedding model without retrieval.

A.7 Retriever Method Comparison

In Table 11, we compare KAR^{3}L's BERT semantic similarity retrieval method with an alternative BM-25 retriever. We find that the original model using BERT similarity outperforms BM-25 across all metrics, confirming the notion that more advanced retrievers are better at identifying relevant items in a student's study history.

A.8 Training Details

We train KAR^{3}L for 12 hours using a single NVIDIA RTX:A4000 GPU. Some model variants were trained using a single NVIDIA A100 GPU with 80 GB GPU memory. Parameters were manually selected without search. We use the default BERT configuration, and our classifier is implemented in PyTorch[3] with the following layers: 1) BERT model; 2) Dropout; 3) Linear Layer; 4) GELU Activation Layer; 5) Layer Normalization; 6) Dropout; 7) Linear Layer. We minimize the binary cross-entropy loss of this model. We use the Adam optimizer (Kingma and Ba, 2015), a learning rate of 0.00005, a batch size of 64,and 10 epochs.

All baselines are implemented using the official code provided by the authors of the respective papers, and all hyperparameters in the model implementations were chosen according to the reported hyperparameters in the paper. LM-KT was trained on a single NVIDIA RTX:A6000 GPU and training parameters were the default values in the provided code. FSRS was trained using the Collaboratory notebook provided by the authors.

All metrics were reported from a single run. Accuracy, AUROC, and Expected Calibration Error were all computed using scikit-learn.[4]

A.9 User Study Instructions

During our user studies, we ensure to provide clear instructions to our users. First, on the home page of our app, the users can read the procedures of our IRB (Figure 6). This serves as detailed instructions for our user studies. Second, when it is time for the user to study test mode, a popup appears giving brief instructions about the next test set they are about to study (Figure 7). Users were also made aware of the confidentiality of their data (Figure 8), which can be viewed by clicking on the IRB link on our home page. The rest of the instructions are outlined in our advertisements.

[3] https://pytorch.org/

[4] https://scikit-learn.org/

Category	Example Front (Question)	Example Back (Answer)	Category Frequency
Philosophy	William James and, later, Richard Rorty continued a strain of philosophy largely inaugurated by this philosopher	C.S. Peirce	425
Trash (Pop Culture)	Recently-cancelled FOX show which starred Eliza Dushku as Echo and which was directed by Joss Whedon	Dollhouse	496
Mythology	To pay Thrym back for stealing Mjolnir, Thor wore bridal clothes to disguise himself as this goddess	Freya	1194
Science	Metal which is used in aerospace applications due to its durability, with atomic number 22 and symbol Ti	Titanium	2423
Current Events	A citizen of this nation crashed a plane into the French Alps in March 2015	Federal Republic of Germany	99
Fine Arts	Revived by Wanda Landowski, it helped bring success to its 1955 performer, Glenn Gould	the Goldberg Variartions	3158
Religion	This text was originally written in unknown characters referred to as "Reformed Egyptian"	The Book of Mormon	807
Literature	Chief harpooner of the Pequod, a cannibal companion of Ishmael in Melville's Moby Dick	Queequeg	4822
Social Science	Luigi Pasinetti created a fifteen-equation mathematical model of this economist's views	David Ricardo	966
Georgraphy	The Franz Joseph and Fox Glaciers can be found in this nation, notable for existing at low altitudes	New Zealand	1053
History	This figure ruled during the Interregnum and led the New Model Army after the assassination of Charles I	Oliver Cromwell	8476

Table 5: Examples of flashcards spanning 11 topics derived from the QANTA dataset.

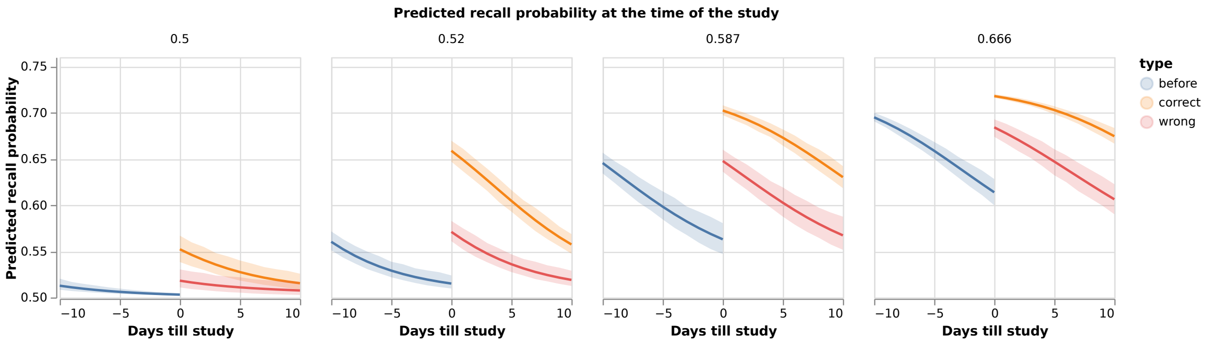


Figure 5: The average forgetting curve ten days before and after a study (day zero) for both when the user succeeds and fails at recalling the flashcard. Unlike exponential forgetting models, the convexity of our forgetting curve depends on both the current predicted recall and the outcome of the most recent study, adding more flexibility.

ID	Column	Description	Num Unique
6.1	user_id	Identifier for the user	543
6.2	card_id	Identifier for the flashcard	18663
6.3	card_text	Text on the flashcard	17554
6.4	deck_id	Identifier for the deck (subject) being studied	71
6.5	deck_name	Name of the deck (subject) being studied	60

Table 6: Qualitative value columns in our released dataset, along with descriptions and number of unique values.

ID	Column	Description	Mean	Min	Max
7.1	is_new_fact	Whether the user has (not) seen this card before	0.44	0.00	1.00
7.2	user_n_study_positive	Number of times the user has studied and answered correctly	1.95e03	0.00	1.47e04
7.3	user_n_study_negative	Number of times the user has studied and answered incorrectly	471.14	0.00	3.52e03
7.4	user_n_study_total	Number of times the user has studied in total	2.42+e03	0.00	1.63e04
7.5	card_n_study_positive	Number of times the card has been answered correctly	6.78	0.00	82.00
7.6	card_n_study_negative	Number of times the card has been answered incorrectly	1.93	0.00	35.00
7.7	card_n_study_total	Number of times the card has been studied in total	8.72	0.00	92.00
7.8	usercard_n_study_positive	Number of times the user answered this card correctly	1.04	0.00	24.00
7.9	usercard_n_study_negative	Number of times the user answered this card incorrectly	0.60	0.00	19.00
7.10	usercard_n_study_total	Number of times the user answered this card in total	1.64	0.00	26.00
7.11	acc_user	Accuracy of this user in all studies	0.70	0.00	1.00
7.12	acc_card	Accuracy of all users on this card	0.60	0.00	1.00
7.13	acc_usercard	Accuracy of the user on this card	0.33	0.00	1.00
7.14	usercard_delta	Time (hours) since the previous study of this card by this user.	194.85	0.00	6.79e03
7.15	usercard_delta_previous	The previous delta. Zero if first or second study.	2.92e05	0.00	1.52e07
7.16	usercard_prev_response	The result (correct/incorrect) of the previous study	0.37	0.00	1.00
7.17	leitner_box	Partition the user is in according to the Leitner system	1.48	0.00	10.00
7.18	sm2_efactor	Easiness factor computed by SM-2	1.15	0.00	2.50
7.19	sm2_interval	Interval computed by SM-2	1.26e05	0.00	1.86e08
7.20	sm2_repetition	Repetition factor computed by SM-2	0.88	0.00	24.00
7.21	delta_to_leitner	Days until Leitner would schedule the card for review	64.21	-6.8e03	2.40e04
7.22	delta_to_sm2	Days until SM-2 would schedule the card for review	2.91e05	-97.00	4.58e07
7.23	elapsed_milliseconds	Time user spent thinking about the answer before submitting	1.04e04	0.00	6.00+e04
7.24	n_minutes_spent	Number of minutes the user spent on the app	353.28	0.00	2.37e03
7.25	correct_on_first_try	Did the user answer the card correctly on the very first study.	0.26	0.00	1.00
7.26	response	Correct / Incorrect	0.71	0.00	1.00

Table 7: Quantitative value columns in our released dataset, along with descriptions, min, max, and mean values.

ID	Column	Description
8.1	utc_datetime	Datetime object for when the study of this card occurred
8.2	utc_date	Date version of utc_datetime

Table 8: Timestamp data columns in our released dataset.

	Seen Cards		*Unseen Cards*	
Model	AUC (↑)	ECE (↓)	AUC (↑)	ECE (↓)
KAR^{3}L Full	**0.780**	**0.108**	**0.740**	0.124
Past-k	0.758	0.119	0.729	**0.119**
No BERT	0.692	0.127	0.612	0.205
No $\mathcal{X}(z)$	0.680	0.135	0.620	0.191

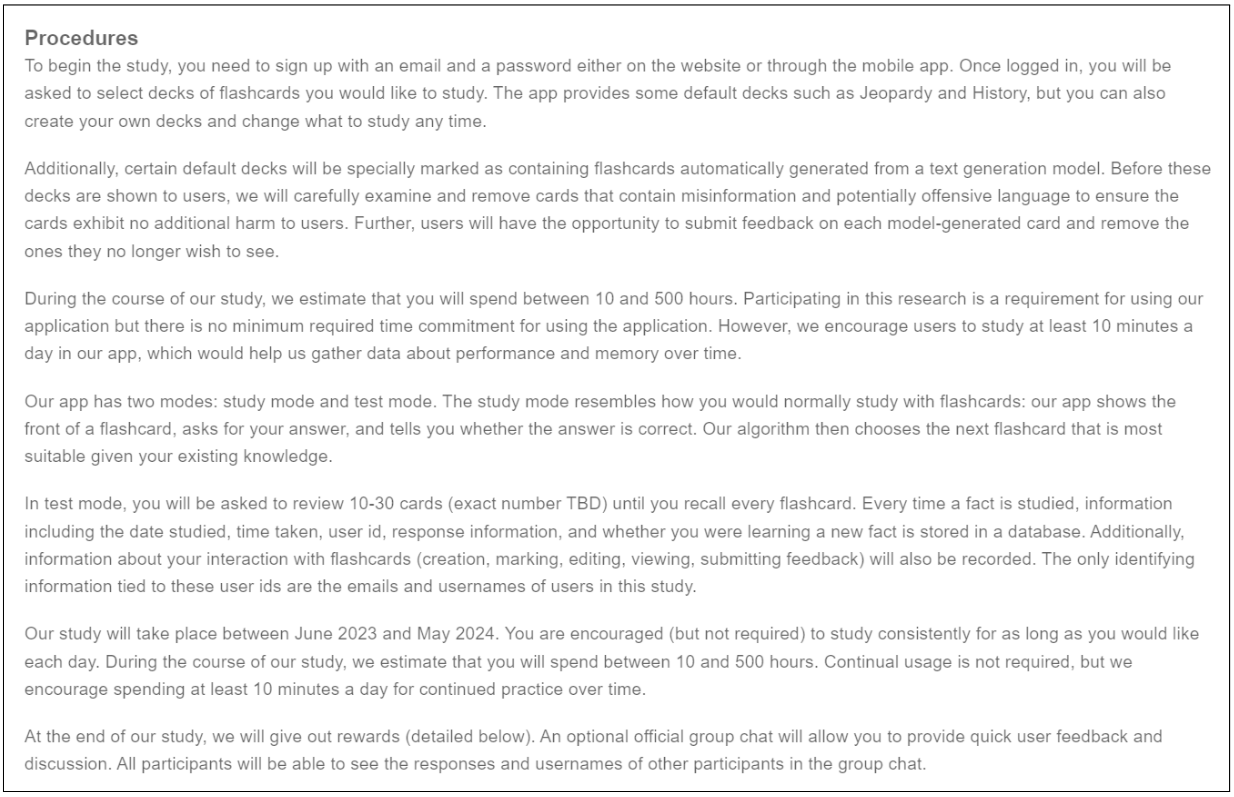
Table 9: KAR^{3}L ablations including past-k retrieval versus top-k (KAR^{3}L Full) retrieval. The No $\mathcal{X}(z)$ and No BERT ablations retain the top-k retrieval. $k = 5$ for all models.

	Seen Cards		*Unseen Cards*	
Embedding Model	AUC (↑)	ECE (↓)	AUC (↑)	ECE (↓)
BERT (w/ Retrieval)	**0.864**	**0.091**	0.786	**0.085**
BERT (no Retrieval)	**0.864**	0.098	0.776	0.11
LLaMA (no Retrieval)	0.841	0.107	**0.810**	0.169

Table 10: Comparison of KAR^{3}L models using BERT and LLaMA embeddings. A LLaMA model with retrieval could not be trained due to OOM error (limited GPU resources).

	Seen Cards		*Unseen Cards*	
Retrieval Method	AUC (↑)	ECE (↓)	AUC (↑)	ECE (↓)
BERT	**0.864**	**0.091**	**0.786**	**0.085**
BM-25	0.858	0.106	0.771	0.139

Table 11: Comparison of KAR^{3}L using different dense and sparse retrieval methods: BERT and BM-25.



Procedures

To begin the study, you need to sign up with an email and a password either on the website or through the mobile app. Once logged in, you will be asked to select decks of flashcards you would like to study. The app provides some default decks such as Jeopardy and History, but you can also create your own decks and change what to study any time.

Additionally, certain default decks will be specially marked as containing flashcards automatically generated from a text generation model. Before these decks are shown to users, we will carefully examine and remove cards that contain misinformation and potentially offensive language to ensure the cards exhibit no additional harm to users. Further, users will have the opportunity to submit feedback on each model-generated card and remove the ones they no longer wish to see.

During the course of our study, we estimate that you will spend between 10 and 500 hours. Participating in this research is a requirement for using our application but there is no minimum required time commitment for using the application. However, we encourage users to study at least 10 minutes a day in our app, which would help us gather data about performance and memory over time.

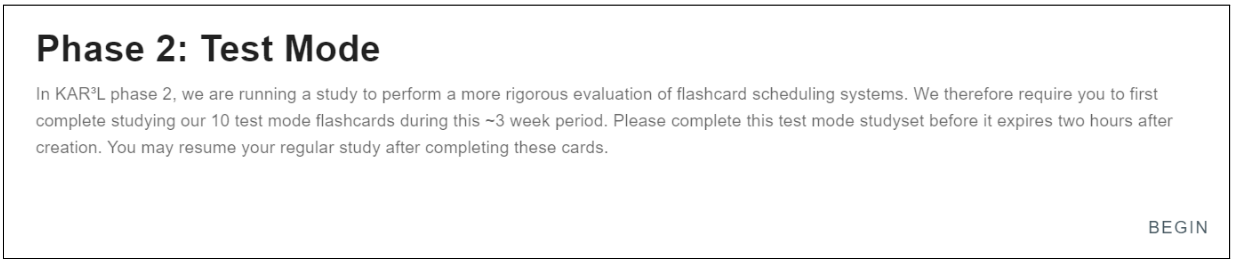
Our app has two modes: study mode and test mode. The study mode resembles how you would normally study with flashcards: our app shows the front of a flashcard, asks for your answer, and tells you whether the answer is correct. Our algorithm then chooses the next flashcard that is most suitable given your existing knowledge.

In test mode, you will be asked to review 10-30 cards (exact number TBD) until you recall every flashcard. Every time a fact is studied, information including the date studied, time taken, user id, response information, and whether you were learning a new fact is stored in a database. Additionally, information about your interaction with flashcards (creation, marking, editing, viewing, submitting feedback) will also be recorded. The only identifying information tied to these user ids are the emails and usernames of users in this study.

Our study will take place between June 2023 and May 2024. You are encouraged (but not required) to study consistently for as long as you would like each day. During the course of our study, we estimate that you will spend between 10 and 500 hours. Continual usage is not required, but we encourage spending at least 10 minutes a day for continued practice over time.

At the end of our study, we will give out rewards (detailed below). An optional official group chat will allow you to provide quick user feedback and discussion. All participants will be able to see the responses and usernames of other participants in the group chat.

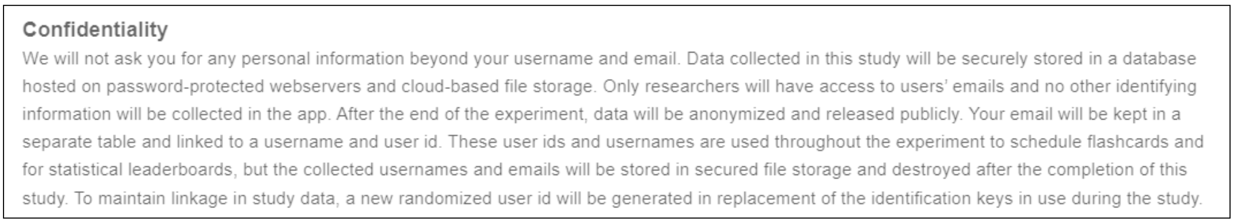
Figure 6: Procedures and instructions shown to users.



Phase 2: Test Mode

In KAR^{3}L phase 2, we are running a study to perform a more rigorous evaluation of flashcard scheduling systems. We therefore require you to first complete studying our 10 test mode flashcards during this ~3 week period. Please complete this test mode studyset before it expires two hours after creation. You may resume your regular study after completing these cards.

Figure 7: Test Mode popup instructions shown to users just before completing a test mode study.



Confidentiality

We will not ask you for any personal information beyond your username and email. Data collected in this study will be securely stored in a database hosted on password-protected webservers and cloud-based file storage. Only researchers will have access to users' emails and no other identifying information will be collected in the app. After the end of the experiment, data will be anonymized and released publicly. Your email will be kept in a separate table and linked to a username and user id. These user ids and usernames are used throughout the experiment to schedule flashcards and for statistical leaderboards, but the collected usernames and emails will be stored in secured file storage and destroyed after the completion of this study. To maintain linkage in study data, a new randomized user id will be generated in replacement of the identification keys in use during the study.

Figure 8: Confidentiality details shown to users.