Teaching Multiple Concepts to Forgetful Learners

Anette Hunziker[†] **Yuxin Chen**[‡] **Oisin Mac Aodha**[‡] **Manuel Gomez Rodriguez**[*]
Andreas Krause[⋆] **Pietro Perona**[‡] **Yisong Yue**[‡] **Adish Singla**[*]

[†]University of Zurich & ETH Zurich, ahunziker@ini.uzh.ch
[‡]Caltech, {chenyux, macaodha, perona, yyue}@caltech.edu,
[⋆]ETH Zurich, krausea@ethz.ch
[*]MPI-SWS, {manuelgr, adishs}@mpi-sws.org

Abstract

How can we help a forgetful learner learn multiple concepts within a limited time frame? For long-term learning, it is crucial to devise teaching strategies that leverage the underlying forgetting mechanisms of the learners. In this paper, we cast the problem of adaptively teaching a forgetful learner as a novel discrete optimization problem, where we seek to optimize a natural objective function that characterizes the learner's expected performance throughout the teaching session. We then propose a simple greedy teaching strategy and derive strong performance guarantees based on two intuitive data-dependent parameters, which characterize the degree of diminishing returns of teaching each concept. We show that, given some assumptions of the learner's memory model, one can efficiently compute the performance bounds. Furthermore, we identify parameter settings of our memory models where greedy is guaranteed to achieve high performance. We have deployed our approach in two concrete applications, namely (1) an educational app for online vocabulary teaching and (2) an app for teaching novices how to recognize bird species. We demonstrate the effectiveness of our algorithm using simulations along with user studies.

1 Introduction

In many real-world educational applications, human learners often intend to learn more than one concept. For example, in a language learning scenario, a learner aims to memorize a number of words from a foreign language. In citizen science projects such as eBird [19], the goal of a learner is to recognize multiple bird species from a given geographic region. As the number of concepts increases, the learning problem may become overwhelmingly challenging due to the learner's limited memory and propensity to forget. It has been well established in the psychology literature that in the context of *human learning*, the knowledge of a learner decays rapidly without reconsolidation [7]. Somewhat analogously, in the sequential *machine learning* setting, modern machine learning methods, such as artificial neural networks, can be drastically disrupted when presented with new information from different domains, which leads to catastrophic interference and forgetting [10]. Therefore, to retain long-term memory (for both human and machine learners), it is crucial to devise teaching strategies that leverage the underlying forgetting mechanisms of the learners.

A prominent approach towards teaching forgetful learners is through repetition. Properly-scheduled repetitions and reconsolidations of previous knowledge have proven effective for a wide variety of real-world learning tasks, including piano practice [13, 16], surgery skills [22, 17, 4], video games [15, 18], and vocabulary learning [6], among others. For many of the above application domains, it has been shown that by carefully designing the scheduling policy, one can achieve substantial gains over simple heuristics (such as spaced repetition at fixed time intervals, or a simple round robin schedule) [5]. Unfortunately, despite the extensive empirical results in these fields, most of these scheduling techniques are based on heuristics, and little is known about their theoretical performance.

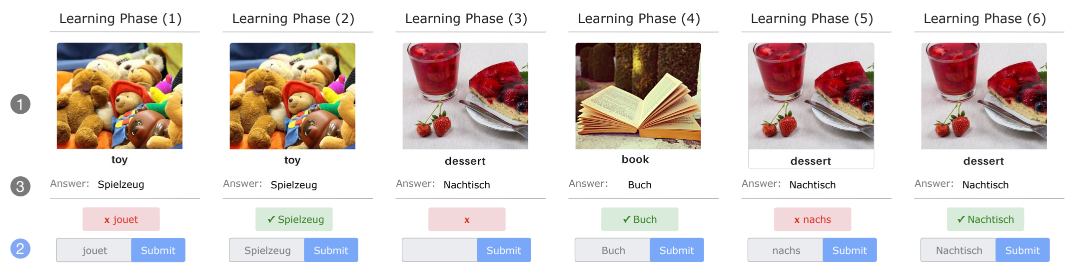


Figure 1: Illustration of our adaptive teaching framework applied to German vocabulary learning, shown here for six time steps. Each learning phase proceeds in three stages: (1) the system displays a flashcard with an image and its English description, (2) the learner inputs the German translation, and (3) the system provides feedback in the form of the correct answer if the input is incorrect.

In this paper, we explore the following research question: *Given limited time, can we help a forgetful learner efficiently learn multiple concepts in a principled manner?* More concretely, we consider an adaptive setting where at each time step, the teacher needs to pick a concept from a finite set based on the learner's previous responses, and the process iterates until the learner's time budget is exhausted. Given the memory model of the learner, what is an optimal teaching curriculum? How should this sequence be adapted based on the learner's performance history?

For a high-level overview of our approach, let us consider the example in Fig. 1, which illustrates one of our applications (cf. [1, 2]) on German vocabulary learning. Here, our goal is to teach the learner three German words within six iterations. One trivial approach could be to show the flashcards in a round robin fashion. However, the round robin sequence is deterministic and thus not capable of adapting to the learner's input. In contrast, our algorithm outputs a personalized teaching sequence based on the learner's performance history. Our algorithm is based on a novel formulation of the adaptive teaching problem. In §3, we propose a novel discrete optimization problem, where we seek to maximize a natural surrogate objective function that characterizes the learner's expected performance throughout the teaching session. Note that constructing the optimal teaching policy could be prohibitively expensive for long teaching sessions, as it boils down to solving a stochastic sequence optimization problem, which is NP-hard in general. In §4, we introduce our greedy algorithm, and derive strong performance guarantees based on two intuitive data-dependent parameters. We then show that for certain memory models of the learner, one can efficiently compute the performance bounds. Furthermore, we identify parameter settings of the memory models where the greedy algorithm is guaranteed to achieve high performance. We describe results for simulated learners in §5, and show significant improvements over baselines for the challenging task of teaching real humans in §6.

2 Related Work

Optimal scheduling with spaced repetition models Numerous studies in neurobiology and psychology have emphasized the importance of the *spacing* and *lag* effects in human learning. The spacing effect is the observation that spaced repetition produces greater improvements in learning compared to massed repetition (i.e., "cramming"). The lag effect refers to the benefit of introducing appropriate time lags between study sessions [21]. These findings lay the foundations of modern spaced repetition research, including widely-used heuristic-based approaches, such as Leitner [9], Pimsleur [11], and SuperMemo [3]. Settles and Meeder (2016) [14] introduced Half-life Regression (HLR) as a generalization of these heuristics, and showed that HLR in general outperforms the existing approaches. In this paper, we adopt a variant of the HLR to model the learner.

Recently, Reddy et al. (2016) [12] presented a queueing network for flashcard learning and provided a tractable algorithm to approximate a solution. However, their approach is specifically designed for Leitner systems, where the meters of learners' skills often do not adequately reflect what they have learned [14]. Tabibian et al. (2017) [20] considered optimizing learning schedules in continuous time for independent items, and used optimal control theory to derive optimal scheduling when optimizing for a penalized recall probability area-under-the-curve loss function. In contrast to [20], we consider the discrete time setting. We are interested in the scenario where a learner studies their flashcards at constant time intervals (e.g. on the way to work or before going to bed), rather than at arbitrary times.

Sequence optimization / sequential decision making Our theoretical framework is inspired by recent results on string submodular function maximization [23] and adaptive submodular optimization [8]. In particular, Zhang et al. (2016) [23] introduced the notion of string submodular functions, which, analogous to the classical notion of submodular set functions, enjoy similar performance guarantees for maximization deterministic sequence functions. However, we note that our setting is drastically different from [23]. The authors focus on deterministic string submodular functions, whereas our teaching algorithm operates in the stochastic setting, and our objective function is highly non-submodular. As a second note, our framework (in particular Corollary 2) can be viewed as a strict generalization of string submodular function maximization to the adaptive setting.

3 The Teaching Model

In this section, we first introduce the notation for our teaching model. Then, we describe the interactive teaching protocol and formally state the problem studied in this paper.

3.1 Target concepts and memory of the learner

Suppose that the teacher aims to teach the learner n *independent* concepts in a finite time horizon T. W.l.o.g., we assume that each concept $i \in \{1, \dots, n\}$ consists of one instance and a corresponding label[1]. For instance, in language learning, a concept corresponds to a word in the vocabulary of a second language. Let us use y_t to denote the event that the learner recalls a concept at time step t, where $y_t = 1$ means that the learner successfully recalls the label, (i.e., the learner correctly translates the word), and $y_t = 0$ otherwise. We assume that the learner's memory of concept i at time t is captured by a memory model $g_i\left(t, \psi\right) := \mathbb{P}\left[y_i = 1 \mid \psi\right]$. Here, $\psi \in \Psi$ denotes the historical events in which concept i was revealed and Ψ denotes the set of feasible histories. As an example, the probability of recalling concept i at time t for the exponential forgetting curve model is given by $g_i\left(t, \psi\right) = \exp(-\alpha n_i(\psi)(t-\ell))$, and the recall probability for the power-law forgetting curve model is given by $g_i\left(t, \psi\right) = (1 + \beta(t - \ell))^{-n_i(\psi)}$. Here, the variable $n_i(\psi)$ depends on the historical frequency of showing concept i, and α, β are scaling parameters that characterize the forgetting rate.

3.2 Model of interaction

We consider the following interactive teaching protocol. At iteration t, the teacher picks a concept from the set $\{1, \dots, n\}$ and presents an instance of it to the learner without revealing its label. The learner then tries to recall the concept. After the learner makes an attempt, the teacher collects the outcome y_t and reveals the true label. We use σ to denote the *sequence* of concepts picked by the teacher, and use σ_t to denote the t^{th} element of the sequence. At the end of iteration t, the teacher adds (σ_t, y_t) to the observation history $\psi := (\sigma_{1:t-1}, y_{1:t-1})$, and updates the memory model $g_i\left(t, \psi\right)$.

3.3 Objective function, policy and the optimal teaching problem

The goal of the teacher is to maximize the learner's performance in recalling all concepts after T iterations. A natural choice of the objective function is the average recall probability of all concepts at the end of the teaching session. This objective, however, does not explicitly capture the performance of the learner during the training phase, which may stretch over years for language learning. Therefore, to provide the learner with high proficiency as soon as possible, we optimize for concept retrievability during learning. We consider the following objective, which measures the learner's average cumulative recall probability for all the concepts across the teaching horizon

$$f(\sigma_{1:t}, y_{1:t}) = \frac{1}{nT} \sum_{i=1}^{n} \sum_{\tau=1}^{T} g_i\left(\tau + 1, \sigma_{1:\min(\tau,t)}, y_{1:\min(\tau,t)}\right). \tag{1}$$

Here, $g_i\left(\tau + 1, \sigma_{1:\min(\tau,t)}, y_{1:\min(\tau,t)}\right)$ denotes the probability of the learner recalling concept i correctly at time step $\tau + 1$, given the sequence of examples selected up to time step $\min(\tau, t)$. Intuitively, our objective function can be interpreted as the (discrete) area under the learner's forgetting curve over the entire teaching session (i.e., we are summing over the recall probabilities across all time steps up to τ (and hence to T), even when we have only observed the learner's history up to t).

[1] In the case where a concept consists of multiple instances, we consider the teacher, at time t, showing the full batch of instances to teach concept i.

Algorithm 1 The Greedy Algorithm

$\sigma \leftarrow \emptyset, y \leftarrow \emptyset$
for $t = \{1, \ldots, T\}$ **do**
  $i_t \leftarrow \arg\max_i \Delta\left(i \mid \sigma, y\right)$ ▷ Choose the item with the largest conditional marginal gain
  Observe y_t
  Set $\sigma \leftarrow \sigma \oplus i_t, y \leftarrow y \oplus y_t$
end for

The teacher's teaching strategy can be represented as a *policy* $\pi : \Psi \to \{1, \ldots, n\}$, which maps any observation history to the next concept to be revealed. For a given policy π, we use $(\sigma^\pi_{1:t}, y^\pi_{1:t})$ to denote a random trajectory from the policy until time t. The average utility of a policy π is defined as

$$F\left(\pi\right) = \mathbb{E}_{\sigma^\pi, y^\pi}[f(\sigma^\pi_{1:T}, y^\pi_{1:T})]\,. \tag{2}$$

Given the learner's memory model for each concept i and the time horizon T, we seek the optimal teaching policy that achieves the maximal average utility

$$\pi^* \in \max_\pi F\left(\pi\right). \tag{3}$$

Finding the optimal solution for Problem (3) is a formidable task. It requires searching through the space of all possible feasible policies. In fact, even for the simple setting where the objective function does not depend on the learner's responses, i.e., when $\forall y_{1:t}, f(\sigma_{1:t}, y_{1:t}) = f(\sigma_{1:t}, \cdot)$, Problem (3) reduces to a combinatorial optimization problem over sequences, which is NP-hard. In the following, we present a simple greedy algorithm, and provide a data-dependent lower bound on its average utility against the optimal policy. Moreover, we prove that under some additional conditions on the learner's memory model, one can efficiently compute such an empirical bound.

4 Algorithms and Theoretical Analysis

We consider a simple, greedy approach towards constructing teaching policies. Formally, given an observation history $(\sigma_{1:t-1}, y_{1:t-1})$, we define the conditional marginal gain of teaching a concept i at time t as

$$\Delta\left(i \mid \sigma_{1:t-1}, y_{1:t-1}\right) = \mathbb{E}_{y_t}[f(\sigma_{1:t-1} \oplus i, y_{1:t-1} \oplus y_t) - f(\sigma_{1:t}, y_{1:t})]\,, \tag{4}$$

where $\oplus$ denotes the concatenation operation, and the expectation is taken over the randomness of learner's recall y_t, conditioned on having observed $(\sigma_{1:t-1}, y_{1:t-1})$. The greedy algorithm, as described in Algorithm 1, iteratively picks the item that maximizes this conditional marginal gain.

4.1 Theoretical Guarantees

We now present a general theoretical framework for analyzing the performance of the adaptive greedy policy (Algorithm 1). Importantly, our bound depends on two natural properties of the objective function f, both related to the notion of *diminishing returns* of a sequence function. Intuitively, the following two properties reflect how much a bad choice by the greedy algorithm can affect the optimality of the solution.

Definition 1 (Online stepwise submodular coefficient)**.** *Fix policy π of length T. The online submodular coefficient of function f with respect to policy π at step t is defined as*

$$\gamma^\pi_t := \min_{\sigma^\pi_{1:t}, y^\pi_{1:t}} \gamma(\sigma^\pi_{1:t}, y^\pi_{1:t}) \tag{5}$$

where $\gamma(\sigma, y) = \min_{i, (\sigma', y'):|\sigma|+|\sigma'|<T} \frac{\Delta(i|\sigma,y)}{\Delta(i|\sigma\oplus\sigma', y\oplus y')}$ denotes the minimal ratio between the gain of any item i given current observation history (σ, y) and the gain of i in any future steps.

Definition 2 (Online stepwise backward curvature)**.** *Fix policy π of length T. The online backward curvature of function f with respect to policy π at step t is defined as*

$$\omega^\pi_t := \max_{\sigma^\pi_{1:t}, y^\pi_{1:t}} \omega(\sigma^\pi_{1:t}, y^\pi_{1:t}) \tag{6}$$

where $\omega(\sigma, y) = \max_\pi \mathbb{E}_{\sigma^\pi, y^\pi}\left[\frac{(f(\sigma,y)-f(\emptyset))-(f(\sigma\oplus\sigma^\pi, y\oplus y^\pi)-f(\sigma^\pi, y^\pi))}{f(\sigma,y)-f(\emptyset)}\right]$ denotes the normalized maximal expected second-order difference when considering the current observation history (σ, y).

Here, $\gamma(\sigma, y)$ and $\omega(\sigma, y)$ generalizes the notion of *string submodularity* and *total backward curvature* for sequence functions [23] to the stochastic setting. Intuitively, $\gamma(\sigma, y) \leq 1$ measures the degree of diminishing returns of a sequence function in terms of the *ratio* between the conditional marginal gains. If $\forall(\sigma, y), \gamma(\sigma, y) = 1$, then the conditional marginal gain of adding any item to any subsequent observation history is non-decreasing. In contrast, $\omega(\sigma, y)$ measures the degree of diminishing returns in terms of the *difference* between the marginal gains. As our first main theoretical result, we provide a data-dependent bound on the average utility of the greedy policy against the optimal policy.

Theorem 1. *Let π^g be the online greedy policy induced by Algorithm 1, and F be the objective function as defined in Eq. (2). Then for all policies π^*,*

$$F(\pi^g) \geq F(\pi^*) \sum_{t=1}^{T} \frac{\gamma^g_{T-t}}{T} \prod_{\tau=0}^{t-1} \left(1 - \frac{\omega^g_\tau \gamma^g_\tau}{T}\right), \tag{7}$$

where γ^g_t and ω^g_t denote the online stepwise submodular coefficient and online stepwise backward curvature of f with respect to the policy π^g at time step t.

The summand on the R.H.S. of Eq. (7) is in fact a lower bound on the expected one-step gain of the greedy policy. Therefore, if we run the greedy algorithm for only $s \leq T$ iterations, we can bound its expected utility by $F\left(\pi^{\text{g}}_{1:s}\right) \geq F(\pi^*) \sum_{t=1}^{s} \frac{\gamma^{\text{g}}_{T-t}}{T} \prod_{\tau=0}^{t-1}\left(1 - \frac{\omega^{\text{g}}_\tau \gamma^{\text{g}}_\tau}{T}\right)$, where π^* is the optimal policy (of length T). We can further relax the bound by considering the worst-case online stepwise submodularity ratio and curvature across all time steps.

Corollary 2. *Let $\gamma^g = \min_t \gamma^g_t$ and $\omega^g = \max_t \omega^g_t$. For all π^*, $F(\pi^g) \geq \frac{1}{\omega^g}\left(1 - e^{-\gamma^g \omega^g}\right) F(\pi^*)$.*

The proofs are deferred to the Appendix. Note that Corollary 2 generalizes the string submodular optimization framework of [23], which only holds under the *deterministic* setting, to the stochastic sequence optimization problem. In particular, for the special case where $\gamma^{\text{g}} = \omega^{\text{g}} = 1$ and $f(\sigma_{1:t}, y_{1:t})$ is independent of $y_{1:t}$, Corollary 2 reduces to $f(\sigma^{\text{g}}, \cdot) \geq \left(1 - e^{-1}\right) f(\sigma^*, \cdot)$ where $\sigma^{\text{g}}, \sigma^*$ denote the sequences selected by the greedy and the optimal algorithm. However, constructing the bounds in Theorem 1 and Corollary 2 requires us to compute $\gamma^{\text{g}}_t, \omega^{\text{g}}_t$ for $t \in \{1, \ldots T\}$, which is as expensive as computing $F(\pi^*)$. In the following subsection, we investigate a specific learner's model, and provide polynomial time approximation algorithms for computing theoretical lower bound in Theorem 1.

4.2 Performance Analysis: Half-life Regression (HLR) Learners

We consider the case of HLR learners with the following exponential forgetting curve model

$$g_i(\tau, (\sigma, y)) = 2^{-\frac{\tau - \ell_i}{h}} \tag{8}$$

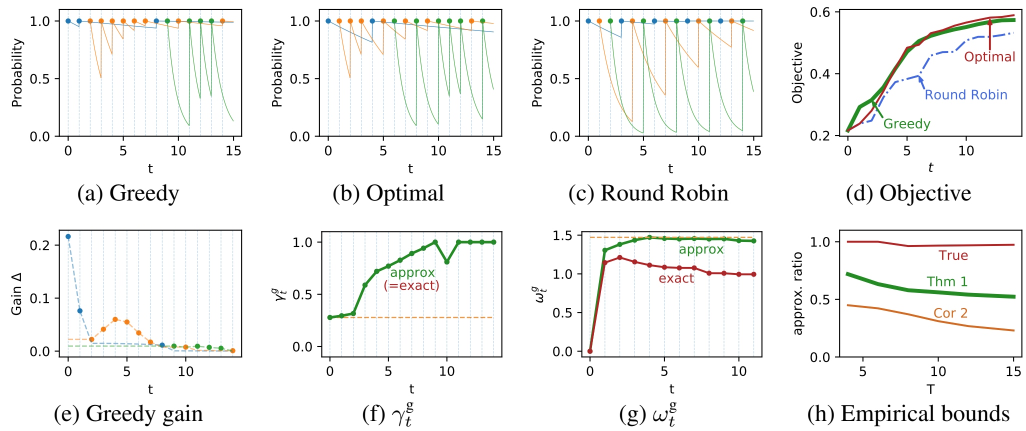
where ℓ_i is the last time concept i was taught, and $h_i = 2^{\theta_i n_i}$ denotes the half life of the learner's recall probability of concept i. Here, $\theta_i = (a_i, b_i, c_i)$ parametrizes the retention rate of the learner's memory, and $n_i = (n^i_+, n^i_-, 1)^\top$, where $n^i_+ := |\{t : \sigma_t = i \wedge y_t = 1\}|$ and $n^i_- := |\{t : \sigma_t = i \wedge y_t = 0\}|$ denote the number of correct recalls and and incorrect recalls of concept i in (σ, y).

We would like to bound the performance of Algorithm 1. While computing $\gamma^{\text{g}}_t, \omega^{\text{g}}_t$ is NP-hard in general, we show that one can efficiently approximate $\gamma^{\text{g}}_t, \omega^{\text{g}}_t$ in the deterministic setting.

Theorem 3. *Assume that the learner is characterized by the HLR model (Eq. (8)) where $\forall i,\ a_i = b_i$. We can compute empirical bounds on γ_t, ω_t in polynomial time.*

We defer the proof of Theorem 3, as well as the approximation algorithms for γ_t, ω_t to the Appendix. In Fig. 2, we demonstrate the behavior of three teaching algorithms on a toy problem with $T = 15, n = 3$. Fig. 2a-2c shows the learner's forgetting curve (i.e., recall probabilities) and the sequences selected by three algorithms: Greedy (Algorithm 1), Optimal (the optimal solution for Problem (3)), and Round Robin (a fixed round robin teaching schedule for all concepts). Observe that Greedy starts with easy concepts (i.e., concepts with higher memory retention rates), moves on to teaching new concepts when the learner has "enough" retention for the current concept, and repeats previous examples towards the end of the teaching session. This behavior is similar to the optimal teaching sequence, and achieves higher utility in comparison to the fixed round robin scheduling (Fig. 2d).

In Fig. 2e-2g, we demonstrate the behavior of the conditional marginal gain, the empirical bounds on $\gamma^{\text{g}}_t, \omega^{\text{g}}_t$, as well as the exact values of $\gamma^{\text{g}}_t, \omega^{\text{g}}_t$ when running the greedy algorithm. In particular, in



(a) Greedy (b) Optimal (c) Round Robin (d) Objective

(e) Greedy gain (f) γ_t^{g} (g) ω_t^{g} (h) Empirical bounds

Figure 2: Performance analysis for the greedy algorithm when teaching a HLR learner three concepts. Each colored marker from Fig. 2a–2c represents a different concept, with $\theta_1 = (2.50, 2.50, 1.26)$ for blue, $(\theta_2 = 1.00, 1.00, -1.00)$ for orange, and $\theta_3 = (0.08, 0.08, -0.88)$ for the green concept. Intuitively, concepts with higher θ_i values are easier to teach.

Fig. 2e, we see that the marginal gain of the orange item is increasing in the early stages (as opposed to many classical discrete optimization problems that exhibit the diminishing returns property), which makes the analysis of the greedy algorithm non-trivial. Note that our algorithm for computing γ_t^{g} actually outputs the *exact* value of γ_t^{g} (a näive approach to computing γ_t^{g} is via extensive enumeration of all possible teaching sequences). In Fig. 2h, we plug in the empirical bounds on γ_t^{g} and ω_t^{g} to Theorem 1 and Corollary 2, and plot the empirical approximation bounds on $F(\pi^{\mathrm{g}})/F(\pi^*)$ as a function of the teaching horizon T. For problem instances with a large teaching horizon T, it is infeasible to compute the true approximation ratio. However, one can still efficiently compute the empirical approximation bound as a useful indicator of the greedy performance.

Theorem 3 shows that it is feasible to compute explicit lower bounds on the utility of Algorithm 1 against the maximal achievable utility. The following proposition, proven in the Appendix, shows that for certain types of learners, the greedy algorithm is guaranteed to achieve a high utility.

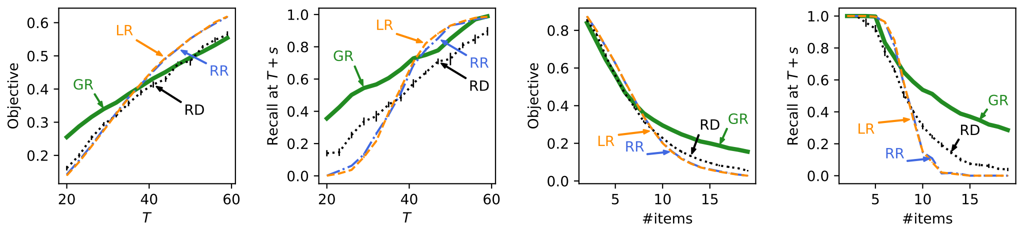
Proposition 4. *Consider the task of teaching a HLR learner n independent concepts in time horizon T, where all concepts share the same parameter configurations, i.e., $\forall\, i, \theta_i = (a, a, 0)$. A sufficient condition for the greedy algorithm to achieve $1 - \epsilon$ utility is $a \geq \max\left\{\log T, \log(3n), \log\left(\frac{2n^2}{\epsilon T}\right)\right\}$.*

5 Simulations

In this section, we experimentally evaluate our algorithm by simulating learners' responses based on a known memory model. This allows us to inspect the behavior of our algorithm and several baseline algorithms in a controlled setting, which we cannot explicit access in a real-world user study.

Dataset We simulated concepts of three different types: "easy", "medium", and "hard". The learner's memory for each concept is captured by an independent HLR model. Concepts of the same type share the same parameter configurations. Specifically, for "easy" concepts, the parameters are $\theta_1 = (a_1 = 6.37, b_1 = 1.00, c_1 = -0.26)$, for "medium", $\theta_2 = (a_2 = 1.52, b_2 = 1.00, c_2 = -2.73)$, and for "hard", $\theta_3 = (a_3 = 1.38, b_3 = 1.00, c_3 = -3.11)$. Our parameters θ_i are chosen by first fixing $b_i = 1$, and then calculating the corresponding values of a_i and c_i by which the learner's recall probability of item i drops to a preset recall probability in the immediate next step after showing concept i. For an "easy" concept, one can compute the corresponding recall probability in the next step according to Eq. (8): $g_1(t = 2, \sigma_1 = 1, y_1 = 1) = 2^{-1/(2^{a_1+c_1})} = 0.99$ and $g_1(t = 2, \sigma_1 = 1, y_1 = 1) = 2^{-1/(2^{b_1+c_1})} = 0.66$. Similarly, these recall probabilities for "medium" concepts is $(0.20, 0.10)$, and for "hard" concepts they are $(0.10, 0.05)$.

Evaluation metric We consider two different criteria when assessing the performance of the candidate algorithms. Our first evaluation metric is the objective value as defined Eq. (1), which measures the learner's average *cumulative* recall probability across the entire teaching session. The



(a) F vs. T ($m = 5$) (b) Recall vs. T ($m = 5$) (c) F vs. m ($T = 60$) (d) Recall vs. m ($T = 60$)

Figure 3: Simulation results

second evaluation metric is the learner's average recall probability of all concepts at the *end* of the teaching session. We call this objective "Recall at $T + s$", where $s \geq 0$ is a integer measuring how far in the future we choose to evaluate the learner's recall.

Baselines To demonstrate the performance of our adaptive greedy policy (referred to as GR), we consider three baseline algorithms. The first baseline, denoted by RD, is the random teacher that presents a random concept at each time step. The second baseline is round robin, denoted by RR, which picks concepts according to a fixed round robin schedule. Our third baseline is a variant of the greedy approach employed in the original HLR paper [14] (where we consider a sightly different formulation of the half life), which can be considered as a generalization of the popular Leitner / Pimsleur approaches. At each iteration, the teacher chooses to display the concept with lowest recall probability according to the HLR memory model of the learner. We refer to this algorithm as LR.

Simulation results We first evaluate the performance of our algorithm against the baselines as a function of the teaching horizon T. In Fig. 3a and Fig. 3b, we plot the objective value and average recall at $T + s$ for all algorithms over 10 random trials, where we set $s = 10$, $m = 5$ with half medium and half difficult concepts, and vary $T \in [20, 60]$. As we can see from both plots, GR consistently outperforms the random baseline in all scenarios. For reasonably small m, when we are teaching multiple concepts with very limited resources (i.e., small budget on T), our greedy approach (GR) outperforms the other baselines. The performance of the lowest recall (LR) and round robin (RR) improves and eventually beats GR as we increase the budget — this behavior is expected, as it corresponds to the scenario where all items get a fair chance of repetition with abundant time budget. Furthermore, our analysis from §4.2 suggests that hard concepts (i.e., items with low retention rate) suffer more from the non-diminishing returns effect (see Fig. 2e), and thus can keep the myopic policy from approaching the optimal utility. In Fig. 3c and Fig. 3d, we show the performance plot for a fixed teaching horizon of $T = 60$ when we vary the number of concepts $m \in [2, 20]$. Here we observe a similar behavior as before. Our results suggest that GR is optimized for the more challenging problem of teaching multiple concepts given a tight time budget.

6 User Study

We have developed online apps for two concrete real-world applications: (1) German vocabulary teaching [2], and (2) teaching novices to recognize bird species as part of a citizen science project [1]. We now briefly introduce the two systems, and present the results of deploying our vocabulary learning app to real human learners.

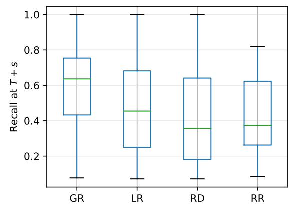
Datasets As part of our beta testing for the German vocabulary teaching app, we collected 100 English-German word pairs in the form of flashcards, each associated with a descriptive image. To extract a fine-grained signal for our user study, we further categorize the words into three difficulty levels based on a thorough evaluation of each word from a domain expert. For the bird teaching app, we collected an initial set of 18 of the most common bird species in North America. Examples from both datasets can be seen in Fig. 4a-4b.

Online teaching interface We set up a simple and intuitive adaptive teaching interface to keep the learners engaged in our user study (see Fig. 4d). In the following discussion, we use German vocabulary learning as an example. Importantly, to establish an experimental setup that accurately reflects our modeling assumptions, we integrate the following design ideas.

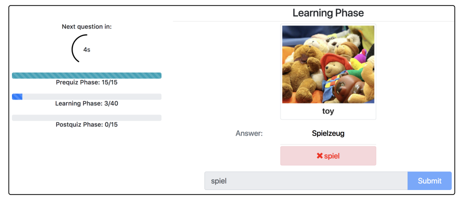
An important component of the user evaluation is to understand the learner's bias (or prior knowledge), which we cannot easily assess purely based on the learner's feedback while learning. To resolve this

(a) Samples from the German dataset

(b) Samples from the Birds dataset



(c) Results for teaching German



(d) Teaching interface

Figure 4: Results from user study and experimental setup

issue, we introduce a prequiz phase for the user study, where we test the learner's knowledge of all the words in the task by asking them to type in translations before the learning phase starts. After the learning phase, the learner will enter a postquiz testing phase. By recording this change in the learner's performance, we can estimate the gain of the teaching session.

To leverage the lag effect of human learning, we impose a minimum time window for each flashcard presentation. In the learning phase, after a user enters her input for a question, she will have 10 seconds to review the correct answers provided by the system, before proceeding to the next question. Furthermore, we also set a maximal answering time for a question to prevent unnecessary delays of the teaching process. Therefore, the user will learn in constant time intervals, which is well-aligned with our discrete-time problem formulation.

Another important aspect is the short-term memory effect. In general, it is highly non-trivial to carry out large scale user studies that span over weeks/months (even though it better fits our HLR model of the learner). Given the physical constraints of real-world experiments, we consider shorter teaching sessions around 25-30 mins, involving teaching 15 words across a total number of 40 iterations. To mitigate the short-term memory effect raised by our experimental setting, we impose an additional constraint on our algorithm (henceforth GR) for the user study, such that it does not pick the same concept twice in a row (otherwise, a learner will simply "copy" the answer she sees on the previous screen). Furthermore, when computing the postquiz score, we exclude the first five entries at the postquiz phase (from a randomly shuffled test sequence) to further reduce the short-memory bias.

Experimental Results For the user study, we focus on the German vocabulary learning problem, and run each candidate algorithm with $m = 15, T = 40$ on 30 workers each on Amazon Mechanical Turk. Note that for these real-world experiments, we do not have explicit access to the learner's memory model. While it is possible to fit a HLR model through an extensive pre-study survey as in [14], we observe from our simulated experiments that our adaptive algorithm is robust to a wide range of parameter configurations. After a thorough validation on the simulated learners, we choose $\theta = (6.0, 2.0, 0.0)$ for both the GR and LR as the "robust" version of the two teaching algorithms. Results for real human workers are shown in Fig. 4c. Overall, GR achieved higher gain than the baselines. Although a fair number of learners fail to achieve good performance, GR managed to teach a larger fraction of the "fast" learners achieving better performance compared to the baselines, which suggests that our framework is a promising strategy for vocabulary teaching.

7 Conclusions

We presented an algorithmic framework for teaching multiple concepts to forgetful learners. We proposed a novel discrete formulation of teaching based on stochastic sequence function optimization, and provided a general theoretical tool for deriving performance bounds. We showed that although the theoretical performance bound is NP-hard to compute in general, we can efficiently compute such bounds for certain memory models of the learner. We have implemented a publicly available learning platform for two concrete applications. We believe our results have made an important step towards bringing the theoretical understanding of machine teaching closer to real-world applications where the forgetting phenomenon is an intrinsic factor.

Acknowledgments This work was supported in part by NSF Award #1645832, Northrop Grumman, Bloomberg, AWS Research Credits, Google as part of the Visipedia project, and a Swiss NSF Early Mobility Postdoctoral Fellowship.

References

[1] Website for teaching birds species. https://www.teaching-birds.cc.

[2] Website for teaching German vocabulary. https://www.teaching-german.cc.

[3] *SuperMemo: your breakthrough speed-learning software : version 6 : Benutzerhandbuch und Referenz*. 1993.

[4] Steven Arild Wuyts Andersen, Peter Trier Mikkelsen, Lars Konge, Per Cayé-Thomasen, and Mads Sølvsten Sørensen. Cognitive load in distributed and massed practice in virtual reality mastoidectomy simulation. *The Laryngoscope*, 126(2), 2016.

[5] David A Balota, Janet M Duchek, and Jessica M Logan. *Is expanded retrieval practice a superior form of spaced retrieval? A critical review of the extant literature*. Psychology Press New York, NY, 2007.

[6] Kristine C Bloom and Thomas J Shuell. Effects of massed and distributed practice on the learning and retention of second-language vocabulary. *The Journal of Educational Research*, 74(4):245–248, 1981.

[7] Hermann Ebbinghaus. *Über das gedächtnis: untersuchungen zur experimentellen psychologie*. Duncker & Humblot, 1885.

[8] Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, 42:427–486, 2011.

[9] S. Leitner and R. Totter. *So lernt man lernen*. Angewandte Lernpsychologie ein Weg zum Erfolg. Herder, 1972.

[10] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989.

[11] Paul Pimsleur. A memory schedule. *The Modern Language Journal*, 51(2):73–75, 1967.

[12] Siddharth Reddy, Igor Labutov, Siddhartha Banerjee, and Thorsten Joachims. 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. ACM, 2016.

[13] Grace Rubin-Rabson. Studies in the psychology of memorizing piano music: II. A comparison of massed and distributed practice. *Journal of Educational Psychology*, 31(4):270, 1940.

[14] Burr Settles and Brendan Meeder. A trainable spaced repetition model for language learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1848–1858, 2016.

[15] Wayne L Shebilske, Barry P Goettl, Kip Corrington, and Eric Anthony Day. Interlesson spacing and task-related processing during complex skill acquisition. *Journal of Experimental Psychology: Applied*, 5(4):413, 1999.

[16] Amy L Simmons. Distributed practice and procedural memory consolidation in musicians' skill learning. *Journal of Research in Music Education*, 59(4):357–368, 2012.

[17] Edward N Spruit, Guido PH Band, and Jaap F Hamming. Increasing efficiency of surgical training: effects of spacing practice on skill acquisition and retention in laparoscopy training. *Surgical endoscopy*, 29(8):2235–2243, 2015.

[18] Tom Stafford and Michael Dewar. Tracing the trajectory of skill learning with a very large sample of online game players. *Psychological science*, 25(2):511–518, 2014.

[19] Brian L Sullivan, Christopher L Wood, Marshall J Iliff, Rick E Bonney, Daniel Fink, and Steve Kelling. ebird: A citizen-based bird observation network in the biological sciences. *Biological Conservation*, 142(10):2282–2292, 2009.

[20] Behzad Tabibian, Utkarsh Upadhyay, Abir De, Ali Zarezade, Bernhard Schoelkopf, and Manuel Gomez-Rodriguez. Optimizing human learning. *arXiv preprint arXiv:1712.01856*, 2017.

[21] Ovid J Tzeng. Stimulus meaningfulness, encoding variability, and the spacing effect. *Journal of Experimental Psychology*, 99(2):162–166, 1973.

[22] EGG Verdaasdonk, LPS Stassen, RPJ Van Wijk, and J Dankelman. The influence of different training schedules on the learning of psychomotor skills for endoscopic surgery. *Surgical endoscopy*, 21(2):214–219, 2007.

[23] Zhenliang Zhang, Edwin KP Chong, Ali Pezeshki, and William Moran. String submodular functions with curvature constraints. *IEEE Transactions on Automatic Control*, 61(3):601–616, 2016.

A Proofs

A.1 Proof of the main results (Theorem 1 and Corollary 2)

A.1.1 Notations and Definitions

For simplicity, we first introduce the notation which will be used in the proof.

Let us use function $\phi(i,t)$ to represent a learner's recall of item i at t, where $\phi(i,t)=1$ indicates that the learner recalls item i correctly at time t, and $\phi(i,t)=0$ otherwise. We call the function ϕ a *realization*, and use Φ to denote a random realization. A realization ϕ is consistent with the observation history $(\sigma_{1:t}, y_{1:t})$, if $\phi(\sigma_\tau,\tau)=y_\tau$ for all $\tau\in\{1,\dots,t\}$. We denote such case by $\phi\sim(\sigma_{1:t},y_{1:t})$.

We further use $(\sigma^\pi(\phi), y^\pi(\phi))$ to denote the sequence of items and observations obtained by running policy π under realization ϕ. Here, $\sigma^\pi(\phi)$ denotes the sequence of items selected by π if the learner is responding according to ϕ.

Similarly with the conditional marginal gain of an item (Eq. (4)), we define the conditional marginal gain of a policy as follows.

Definition 3 (Conditional marginal gain of a policy)**.** *Given observation history $(\sigma_{1:t}, y_{1:t})$ and an item i, the conditional marginal gain of a policy π is defined as*

$$\Delta\left(\pi \mid \sigma_{1:t}, y_{1:t}\right) = \mathbb{E}[f(\sigma_{1:t}\oplus\sigma^\pi(\Phi), y_{1:t}\oplus y^\pi(\Phi)) - f(\sigma_{1:t}, y_{1:t}) \mid \Phi\sim(\sigma_{1:t},y_{1:t})]\,. \tag{9}$$

A.1.2 Proof of Theorem 1

To prove Theorem 1, we first establish a lower bound on the one-step gain of the greedy algorithm. The following lemma provides a lower bound of the one-step conditional marginal gain of the greedy policy π^{g} against the conditional marginal gain of any policy (of length T).

Lemma 5. *Suppose we have selected sequence $\sigma_{1:t}$ and observed $y_{1:t}$. Then, for any policy π of length T,*

$$\max_i \Delta\left(i \mid \sigma_{1:t}, y_{1:t}\right) \geq \frac{\gamma_t^\pi}{T}\Delta\left(\pi \mid \sigma_{1:t}, y_{1:t}\right) \tag{10}$$

Proof. By Definition 3 we know that for all π it holds that

$$\begin{aligned}
\Delta\left(\pi \mid \sigma_{1:t}, y_{1:t}\right) &= \mathbb{E}[f(\sigma_{1:t}\oplus\sigma^\pi_{1:T}(\Phi), y_{1:t}\oplus y^\pi_{1:T}(\Phi)) - f(\sigma_{1:t}, y_{1:t}) \mid \Phi\sim(\sigma_{1:t},y_{1:t})]\\
&\overset{(a)}{=} \mathbb{E}\Bigg[\sum_{\tau=1}^{T}\big(f(\sigma_{1:t}\oplus\sigma^\pi_{1:\tau}(\Phi), y_{1:t}\oplus y^\pi_{1:\tau}(\Phi)) -\\
&\qquad\qquad f(\sigma_{1:t}\oplus\sigma^\pi_{1:\tau-1}(\Phi), y_{1:t}\oplus y^\pi_{1:\tau-1}(\Phi))\big) \mid \Phi\sim(\sigma_{1:t},y_{1:t})\Bigg]\\
&= \sum_{\tau=1}^{T}\mathbb{E}\left[f(\sigma_{1:t}\oplus\sigma^\pi_{1:\tau}(\Phi), y_{1:t}\oplus y^\pi_{1:\tau}(\Phi)) -\right.\\
&\qquad\qquad \left. f(\sigma_{1:t}\oplus\sigma^\pi_{1:\tau-1}(\Phi), y_{1:t}\oplus y^\pi_{1:\tau-1}(\Phi)) \mid \Phi\sim(\sigma_{1:t},y_{1:t})\right]
\end{aligned}$$

$$\overset{(b)}{=} \sum_{\tau=1}^{T} \mathbb{E}\left[\mathbb{E}\left[f(\sigma_{1:t} \oplus \sigma^{\pi}_{1:\tau}(\Phi'), y_{1:t} \oplus y^{\pi}_{1:\tau}(\Phi')) - f(\sigma_{1:t} \oplus \sigma^{\pi}_{1:\tau-1}(\Phi'), y_{1:t} \oplus y^{\pi}_{1:\tau-1}(\Phi')) \;\middle|\; \Phi' \sim (\sigma_{1:t} \oplus \sigma^{\pi}_{1:\tau-1}(\Phi), y_{1:t} \oplus y^{\pi}_{1:\tau-1}(\Phi))\right] \;\middle|\; \Phi \sim (\sigma_{1:t}, y_{1:t})\right]$$

$$\overset{\text{Eq. (4)}}{=} \sum_{\tau=1}^{T} \mathbb{E}\left[\Delta\left(\sigma^{\pi}_{\tau}(\Phi') \mid \Phi' \sim \sigma_{1:t} \oplus \sigma^{\pi}_{1:\tau-1}(\Phi), y_{1:t} \oplus y^{\pi}_{1:\tau-1}(\Phi)\right) \;\middle|\; \Phi \sim (\sigma_{1:t}, y_{1:t})\right] \quad (11)$$

Here, step (a) is a telescoping sum, and step (b) is by the law of total expectation.

Further, by the definition of γ_t (Definition 1) we know that for all π and ϕ it holds that

$$\max_i \Delta\left(i \mid \sigma_{1:t}, y_{1:t}\right) \geq \gamma^{\pi}_t \Delta\left(\sigma^{\pi}_{\tau}(\Phi') \mid \Phi' \sim \sigma_{1:t} \oplus \sigma^{\pi}_{1:\tau-1}(\phi), y_{1:t} \oplus y^{\pi}_{1:\tau-1}(\phi)\right) \quad (12)$$

Combining Eq. (11) with Eq. (12) to get

$$\begin{aligned}
\Delta\left(\pi \mid \sigma_{1:t}, y_{1:t}\right) &\overset{\text{Eq. (11)}}{=} \sum_{\tau=1}^{T} \mathbb{E}\left[\Delta\left(\sigma^{\pi}_{\tau}(\Phi') \mid \Phi' \sim \sigma_{1:t} \oplus \sigma^{\pi}_{1:\tau-1}(\Phi), y_{1:t} \oplus y^{\pi}_{1:\tau-1}(\Phi)\right) \;\middle|\; \Phi \sim (\sigma_{1:t}, y_{1:t})\right] \\
&\overset{\text{Eq. (12)}}{\leq} \sum_{\tau=1}^{T} \mathbb{E}\left[\frac{1}{\gamma^{\pi}_t} \max_i \Delta\left(i \mid \sigma_{1:t}, y_{1:t}\right) \;\middle|\; \Phi \sim (\sigma_{1:t}, y_{1:t})\right] \\
&= \frac{T}{\gamma^{\pi}_t} \max_i \Delta\left(i \mid \sigma_{1:t}, y_{1:t}\right)
\end{aligned}$$

which completes the proof. □

In the following we provide the proof of Theorem 1.

Proof of Theorem 1. By the definition of ω_t (Definition 2,Eq. (6)) we know that for all π it holds that

$$\omega_t \geq 1 - \frac{\mathbb{E}[f(\sigma_{1:t} \oplus \sigma^{\pi}(\Phi), y_{1:t} \oplus y^{\pi}(\Phi)) - f(\sigma^{\pi}(\Phi), y^{\pi}(\Phi)) \mid \Phi \sim (\sigma_{1:t}, y_{1:t})]}{f(\sigma_{1:t}, y_{1:t})}$$

Therefore, we get

$$\begin{aligned}
\Delta\left(\pi \mid \sigma_{1:t}, y_{1:t}\right) &= \mathbb{E}[f(\sigma_{1:t} \oplus \sigma^{\pi}(\Phi), y_{1:t} \oplus y^{\pi}(\Phi)) - f(\sigma_{1:t}, y_{1:t}) \mid \Phi \sim (\sigma_{1:t}, y_{1:t})] \\
&\geq \mathbb{E}[f(\sigma^{\pi}(\Phi), y^{\pi}(\Phi)) - \omega_t f(\sigma_{1:t}, y_{1:t}) \mid \Phi \sim (\sigma_{1:t}, y_{1:t})] \quad (13)
\end{aligned}$$

Now suppose that we have run greedy policy π^{g} up to time step t and have observed sequence $(\sigma^{\mathrm{g}}_{1:t}, y^{\mathrm{g}}_{1:t})$. Combining Lemma 5 (Eq. (10)) with Eq. (13), we get

$$\begin{aligned}
\max_i \Delta\left(i \mid \sigma^{\mathrm{g}}_{1:t}, y^{\mathrm{g}}_{1:t}\right) &= \mathbb{E}\left[f(\sigma^{\mathrm{g}}_{1:t+1}(\Phi), y^{\mathrm{g}}_{1:t+1}(\Phi)) - f(\sigma^{\mathrm{g}}_{1:t}, y^{\mathrm{g}}_{1:t}) \mid \Phi \sim (\sigma^{\mathrm{g}}_{1:t}, y^{\mathrm{g}}_{1:t})\right] \\
&\geq \frac{\gamma_t}{T} \cdot \mathbb{E}\left[f(\sigma^{\pi}(\Phi), y^{\pi}(\Phi)) - \omega_t f(\sigma^{\mathrm{g}}_{1:t}, y^{\mathrm{g}}_{1:t}) \mid \Phi \sim (\sigma^{\mathrm{g}}_{1:t}, y^{\mathrm{g}}_{1:t})\right]
\end{aligned}$$

which implies

$$\begin{aligned}
&\mathbb{E}\left[f(\sigma^{\mathrm{g}}_{1:t+1}(\Phi), y^{\mathrm{g}}_{1:t+1}(\Phi)) \mid \Phi \sim (\sigma^{\mathrm{g}}_{1:t}, y^{\mathrm{g}}_{1:t})\right] \\
\geq &\frac{\gamma_t}{T} \cdot \mathbb{E}\left[f(\sigma^{\pi}(\Phi), y^{\pi}(\Phi)) \mid \Phi \sim (\sigma^{\mathrm{g}}_{1:t}, y^{\mathrm{g}}_{1:t})\right] + \left(1 - \frac{\gamma_t \omega_t}{T}\right) f(\sigma^{\mathrm{g}}_{1:t}, y^{\mathrm{g}}_{1:t}) \quad (14)
\end{aligned}$$

Therefore, we get

$$\begin{aligned}
F\left(\pi^{\mathrm{g}}\right) &= \mathbb{E}\left[f(\sigma^{\mathrm{g}}_{1:T}(\Phi), y^{\mathrm{g}}_{1:T}(\Phi))\right] \\
&\overset{(a)}{=} \mathbb{E}\left[\mathbb{E}\left[f(\sigma^{\mathrm{g}}_{1:T}(\Phi), y^{\mathrm{g}}_{1:T}(\Phi)) \mid \Phi \sim (\sigma^{\mathrm{g}}_{1:T-1}(\Phi'), y^{\mathrm{g}}_{1:T-1}(\Phi'))\right]\right] \\
&\overset{\text{Eq. (14)}}{\geq} \mathbb{E}\left[\frac{\gamma_{T-1}}{T} \cdot \mathbb{E}\left[f(\sigma^{\pi}(\Phi), y^{\pi}(\Phi)) \mid \Phi \sim (\sigma^{\mathrm{g}}_{1:T-1}(\Phi'), y^{\mathrm{g}}_{1:T-1}(\Phi'))\right]\right] +
\end{aligned}$$

$$\mathbb{E}\Big[\Big(1-\frac{\gamma_{T-1}\omega_{T-1}}{T}\Big)\,f(\sigma^{\mathrm{g}}_{1:T-1}(\Phi'),y^{\mathrm{g}}_{1:T-1}(\Phi'))\Big]$$

$$\overset{(b)}{=}\frac{\gamma_{T-1}}{T}\cdot\mathbb{E}[f(\sigma^{\pi}(\Phi),y^{\pi}(\Phi))]+\Big(1-\frac{\gamma_{T-1}\omega_{T-1}}{T}\Big)\cdot\mathbb{E}\big[f(\sigma^{\mathrm{g}}_{1:T-1}(\Phi'),y^{\mathrm{g}}_{1:T-1}(\Phi'))\big]$$

$$=\frac{\gamma_{T-1}}{T}\cdot F(\pi)+\Big(1-\frac{\gamma_{T-1}\omega_{T-1}}{T}\Big)\cdot\mathbb{E}\big[f(\sigma^{\mathrm{g}}_{1:T-1}(\Phi),y^{\mathrm{g}}_{1:T-1}(\Phi))\big] \tag{15}$$

where step (a) and step (b) are by the law of total expectation. Recursively applying Eq. (15) gives us

$$\begin{aligned}F(\pi^{\mathrm{g}})&\geq\frac{\gamma_{T-1}}{T}\cdot F(\pi)+\Big(1-\frac{\gamma_{T-1}\omega_{T-1}}{T}\Big)\cdot\mathbb{E}\big[f(\sigma^{\mathrm{g}}_{1:T-1}(\Phi),y^{\mathrm{g}}_{1:T-1}(\Phi))\big]\\&\geq\Big(\frac{\gamma_{T-1}}{T}+\Big(1-\frac{\gamma_{T-1}\omega_{T-1}}{T}\Big)\frac{\gamma_{T-2}}{T}\Big)F(\pi)+\\&\quad\Big(1-\frac{\gamma_{T-1}\omega_{T-1}}{T}\Big)\Big(1-\frac{\gamma_{T-2}\omega_{T-2}}{T}\Big)\mathbb{E}\big[f(\sigma^{\mathrm{g}}_{1:T-2}(\Phi),y^{\mathrm{g}}_{1:T-2}(\Phi))\big]\\&\geq\dots\\&\geq F(\pi)\sum_{t=1}^{T-1}\frac{\gamma_{T-t}}{T}\prod_{\tau=1}^{t-1}\Big(1-\frac{\gamma_\tau\omega_\tau}{T}\Big)\end{aligned}$$

which completes the proof. □

A.1.3 Proof of Corollary 2

Proof of Corollary 2. Since $\gamma^{\mathrm{g}}=\min_t\gamma_t$ and $\omega^{\mathrm{g}}=\max_t\omega_t$, by Theorem 1 we obtain

$$\begin{aligned}F(\pi^{\mathrm{g}})&\geq F(\pi)\sum_{t=1}^{T}\frac{\gamma_{T-t}}{T}\prod_{\tau=0}^{t-1}\Big(1-\frac{\gamma_\tau\omega_\tau}{T}\Big)\\&\geq F(\pi)\frac{\gamma^{\mathrm{g}}}{T}\sum_{t=1}^{T}\Big(1-\frac{\gamma^{\mathrm{g}}\omega^{\mathrm{g}}}{T}\Big)^t\\&=F(\pi)\frac{1}{\omega^{\mathrm{g}}}\left(1-\Big(1-\frac{\gamma^{\mathrm{g}}\omega^{\mathrm{g}}}{T}\Big)^T\right)\end{aligned}$$

which completes the proof. □

A.2 Proof of Theorem 3

In this section, we provide the proof for Theorem 3. In particular, we divide the proof into two parts. In §A.2.1, we propose a polynomial time algorithm which outputs a lower bound on γ^g_t; in §A.2.2, we provide an upper bound on ω^{g}_t which can be computed in linear time.

A.2.1 Empirical lower bound on γ_t for the case $a=b$

Let us use $\mathrm{count}(\sigma,i)$ to denote the function that returns the number of times item i appears in sequence σ. We first show the following lemma.

Lemma 6. *Fix $s\leq t$. For any $\sigma'\in\{\sigma:|\sigma|=t,count(\sigma,i)=s\}$, we have*

$$\Delta\left(i\mid\sigma^i_{t,1:s},\cdot\right)\geq\Delta\left(i\mid\sigma',\cdot\right)$$

where $\sigma^i_{t,1:s}:=\underbrace{i\oplus i\oplus\cdots\oplus i}_{s\text{ times}}\oplus\underbrace{\_\oplus\_\oplus\cdots\oplus\_}_{t-s\text{ times}}$ *denotes the sequence of items of length t, where the first s items are item i and the remaining $t-s$ items are empty.*

Proof. By definition of the marginal gain (Eq. (4))

$$\Delta\left(i\mid\sigma,y\right)=\mathbb{E}[f(\sigma_{1:t}\oplus i,y_{1:t}\oplus\Phi(i,t+1))-f(\sigma_{1:t},y_{1:t})\mid\Phi\sim(\sigma_{1:t},y_{1:t})]$$

For the case $a=b$, the objective function f is independent of the observed outcomes of the learner's recall. That is,

$$\Delta\left(i\mid\sigma_{1:t},\cdot\right)=f(\sigma_{1:t}\oplus i,\cdot)-f(\sigma_{1:t},\cdot)$$

Algorithm 2 Computing the empirical lower bound on the greedy online stepwise submodular coefficient

```
Require: σ_{1:t}; y_{1:t}
  for i = {1, ..., n} do
      CurrentGain_i ← Δ(i | σ_{1:t}, y_{1:t})
      for τ = {1, ..., T − t} do
          for s ∈ {1, ..., τ} do
              σ' ← (i ⊕ i ⊕ ··· ⊕ i) [s times] ⊕ (_ ⊕ _ ⊕ ··· ⊕ _) [τ − s times]   ▷ Only consider insertions in the front
              v_{τ,s} ← Δ(i | σ_{1:t} ⊕ σ', ·)   ▷ Gain of item i at t + τ, with s insertions
          end for
      end for
      FutureGain_i ← max_{τ,s} v_{τ,s}   ▷ Maximal gain of item i at future time steps
  end for
  γ_t ← min_i CurrentGain_i / FutureGain_i   ▷ Choosing the minimal ratio among all items
  return γ_t
```

$$= \frac{1}{nT} \sum_{i=1}^{n} \sum_{\tau=1}^{T} \left\{ g_i\left(\tau+1, \sigma_{1:t} \oplus i, \cdot\right) - g_i\left(\tau+1, \sigma_{1:t}, \cdot\right) \right\}$$
$$= \frac{1}{nT} \sum_{i=1}^{n} \sum_{\tau=t+1}^{T} \left\{ g_i\left(\tau+1, \sigma_{1:t} \oplus i, \cdot\right) - g_i\left(\tau+1, \sigma_{1:t}, \cdot\right) \right\}$$

Denote $\Sigma_{t,s}^{i} = \{\sigma : |\sigma| = t, \text{count}(\sigma, i) = s\}$. For any $\sigma, \sigma' \in \Sigma_{t,s}^{i}$, we know that

$$\sum_{\tau=t+1}^{T} g_i\left(\tau+1, \sigma_{1:t} \oplus i, \cdot\right) = \sum_{\tau=t+1}^{T} g_i\left(\tau+1, \sigma'_{1:t} \oplus i, \cdot\right)$$

Therefore,

$$\max_{\sigma_{1:t} \in \Sigma_{t,s}^{i}} \Delta\left(i \mid \sigma_{1:t}, \cdot\right) = \frac{1}{nT} \sum_{i=1}^{n} \sum_{\tau=t+1}^{T} \left\{ g_i\left(\tau+1, \sigma_{1:t} \oplus i, \cdot\right) - \min_{\sigma_{1:t} \in \Sigma_{t,s}^{i}} g_i\left(\tau+1, \sigma_{1:t}, \cdot\right) \right\}$$
$$\overset{(a)}{=} \frac{1}{nT} \sum_{i=1}^{n} \sum_{\tau=t+1}^{T} \left\{ g_i\left(\tau+1, \sigma_{1:t} \oplus i, \cdot\right) - g_i\left(\tau+1, \sigma_{t,1:s}^{i}, \cdot\right) \right\}$$

Here, step (a) is due to the fact that the learner's recall of an item is monotonously decreasing (therefore showing item i earlier leads to lower recall in the future). Therefore, it completes the proof. □

An approximation algorithm for γ_t is provided in Algorithm 2.

A.2.2 Empirical upper bound on ω_t for the case $a = b$

In this section, we derive an upper bound on ω_t which can be computed in polynomial time.

Using the notation defined in §A.1, we can rewrite the definition of the online greedy stepwise backward curvature ω_t as

$$\omega_t = \max_{\pi} \left\{ 1 - \frac{\mathbb{E}\left[f(\sigma_{1:t}^{\mathrm{g}} \oplus \sigma^{\pi}(\Phi), y_{1:t}^{\mathrm{g}} \oplus y^{\pi}(\Phi)) - f(\sigma^{\pi}(\Phi), y^{\pi}(\Phi)) \mid \Phi \sim (\sigma_{1:t}^{\mathrm{g}}, y_{1:t}^{\mathrm{g}}) \right]}{f(\sigma_{1:t}^{\mathrm{g}}, y_{1:t}^{\mathrm{g}})} \right\}$$

For the case $a = b$, the objective function f is independent of the observed outcomes of the learner's recall (i.e., f is a deterministic function of the input teaching sequence). Therefore,

$$\omega_t = \max_{\pi} \left\{ 1 - \frac{f(\sigma_{1:t}^{\mathrm{g}} \oplus \sigma^{\pi}, \cdot) - f(\sigma^{\pi}, \cdot)}{f(\sigma_{1:t}^{\mathrm{g}}, \cdot)} \right\}$$
$$= 1 + \max_{\pi} \left\{ \frac{f(\sigma^{\pi}, \cdot) - f(\sigma_{1:t}^{\mathrm{g}} \oplus \sigma^{\pi}, \cdot)}{f(\sigma_{1:t}^{\mathrm{g}}, \cdot)} \right\}$$

For simplicity let us use $\sigma^{\mathrm{g}+\pi} := \sigma^{\mathrm{g}}_{1:t} \oplus \sigma^{\pi}$ to denote the concatenated sequence, and w.l.o.g, assume that π represent the one which maximizes the RHS of the above equation (i.e., π is the optimal policy). Substituting the objective function f in the above equation with its definition (Eq. (1)), we get

$$\begin{aligned}
\omega_t &= 1 + \frac{1}{nT}\frac{1}{f(\sigma^{\mathrm{g}}_{1:t}, \cdot)} \sum_{i=1}^{n} \sum_{\tau=1}^{T} \left\{ g_i\left(\tau+1, \sigma^{\pi}_{1:\tau}, \cdot\right) - g_i\left(\tau+1, \sigma^{\mathrm{g}+\pi}_{1:\tau}, \cdot\right) \right\} \\
&= 1 + \frac{1}{nT}\frac{1}{f(\sigma^{\mathrm{g}}_{1:t}, \cdot)} \sum_{i=1}^{n} \left\{ \sum_{\tau=1}^{T-t} g_i\left(\tau+1, \sigma^{\pi}_{1:\tau}, \cdot\right) + \sum_{\tau=T-t+1}^{T} g_i\left(\tau+1, \sigma^{\pi}_{1:\tau}, \cdot\right) \right. \\
&\qquad \left. - \sum_{\tau=t+1}^{T} g_i\left(\tau+1, \sigma^{\mathrm{g}+\pi}_{1:\tau}, \cdot\right) - \sum_{\tau=1}^{t} g_i\left(\tau+1, \sigma^{\mathrm{g}+\pi}_{1:\tau}, \cdot\right) \right\} \\
&= 1 + \frac{1}{nT}\frac{1}{f(\sigma^{\mathrm{g}}_{1:t}, \cdot)} \sum_{i=1}^{n} \left\{ \sum_{\tau=T-t+1}^{T} g_i\left(\tau+1, \sigma^{\pi}_{1:\tau}, \cdot\right) - \sum_{\tau=1}^{t} g_i\left(\tau+1, \sigma^{\mathrm{g}+\pi}_{1:\tau}, \cdot\right) \right. \\
&\qquad \underbrace{\left. + \sum_{\tau=1}^{T-t} g_i\left(\tau+1, \sigma^{\pi}_{1:\tau}, \cdot\right) - \sum_{\tau=t+1}^{T} g_i\left(\tau+1, \sigma^{\mathrm{g}+\pi}_{1:\tau}, \cdot\right) \right\}}_{\leq 0} \\
&\leq 1 + \frac{1}{nT}\frac{1}{f(\sigma^{\mathrm{g}}_{1:t}, \cdot)} \sum_{i=1}^{n} \left\{ \sum_{\tau=T-t+1}^{T} g_i\left(\tau+1, \sigma^{\pi}_{1:\tau}, \cdot\right) - \sum_{\tau=1}^{t} g_i\left(\tau+1, \sigma^{\mathrm{g}+\pi}_{1:\tau}, \cdot\right) \right\}
\end{aligned} \tag{16}$$

Let $\sigma^{i}_{1:t} := \underbrace{i \oplus i \oplus \cdots \oplus i}_{t \text{ times}}$ denote the sequence of items of length t that consists of all i's. Then, clearly

$$\sum_{\tau=T-t+1}^{T} g_i\left(\tau+1, \sigma^{\pi}_{1:\tau}, \cdot\right) \leq \sum_{\tau=T-t+1}^{T} g_i\left(\tau+1, \sigma^{i}_{1:\tau}, \cdot\right) \tag{17}$$

Combining Eq. (16) with Eq. (17) we get

$$\begin{aligned}
\omega_t &\leq 1 + \frac{1}{nT}\frac{1}{f(\sigma^{\mathrm{g}}_{1:t}, \cdot)} \sum_{i=1}^{n} \left\{ \sum_{\tau=T-t+1}^{T} g_i\left(\tau+1, \sigma^{\pi}_{1:\tau}, \cdot\right) - \sum_{\tau=1}^{t} g_i\left(\tau+1, \sigma^{\mathrm{g}+\pi}_{1:\tau}, \cdot\right) \right\} \\
&\leq 1 + \frac{1}{nT}\frac{1}{f(\sigma^{\mathrm{g}}_{1:t}, \cdot)} \sum_{i=1}^{n} \left\{ \sum_{\tau=T-t+1}^{T} g_i\left(\tau+1, \sigma^{i}_{1:\tau}, \cdot\right) - \sum_{\tau=1}^{t} g_i\left(\tau+1, \sigma^{\mathrm{g}+\pi}_{1:\tau}, \cdot\right) \right\} \\
&= 1 + \frac{1}{nT}\frac{1}{f(\sigma^{\mathrm{g}}_{1:t}, \cdot)} \sum_{i=1}^{n} \left\{ \sum_{\tau=T-t+1}^{T} g_i\left(\tau+1, \sigma^{i}_{1:\tau}, \cdot\right) - \sum_{\tau=1}^{t} g_i\left(\tau+1, \sigma^{\mathrm{g}}_{1:\tau}, \cdot\right) \right\}
\end{aligned} \tag{18}$$

Proof of Theorem 3. Clearly, both the empirical bounds on γ^{g}_t (Algorithm 2) and ω^{g}_t (RHS of Eq. (18)) can be computed in polynomial time. Plugging the values into Theorem 1 and Corollary 1 we get a polynomial time approximation of the empirical bound. □

A.3 Proof of Proposition 4

In this section, we provide the proof of Proposition 4.

Suppose there are n items, and T is a multiple of n. Fix a, and assume that $a_i = b_i = a$ and $c_i = 0$ for all $i \in \{1, \ldots, n\}$. We first show a sufficient condition on a under which the greedy policy reduces to the round robin policy.

Recall from Eq. (8) that the recall probability of an item is

$$g_i(\tau, \cdot) = 2^{\frac{\tau - \ell}{h_i}} \tag{19}$$

where $h_i = 2^{an_i}$ denotes the half life of item i, and n_i denotes the number of times item i is presented so far.

Now assume that the greedy algorithm picks item i at $t = 1$. Then, in order for the greedy algorithm not to pick the same item at $t = 2$, we need to make sure that at $t = 2$, the gain of item i is smaller than the gain of the best item. To achieve that, there must exist some other item j, such that

$$\Delta\left(j \mid \sigma_1 = i\right) > \Delta\left(i \mid \sigma_1 = i\right)$$

That is,

$$\sum_{t=2}^{T}\left(g_j\left(t, \sigma_1 = i, \sigma_2 = j\right) - g_j\left(t, \sigma_1 = i\right)\right) > \sum_{t=2}^{T}\left(g_i\left(t, \sigma_1 = i, \sigma_2 = i\right) - g_i\left(t, \sigma_1 = i\right)\right)$$

A sufficient condition for the above inequality to hold is

$$\begin{aligned} g_j\left(T, \sigma_1 = i, \sigma_2 = j\right) - g_j\left(T, \sigma_1 = i\right) &= g_j\left(T, \sigma_1 = i, \sigma_2 = j\right) \\ &> g_i\left(T, \sigma_1 = i, \sigma_2 = i\right) - g_i\left(T, \sigma_1 = i\right) \end{aligned}$$

Plugging in the definition of g_i, g_j, we get

$$2^{-\frac{T-1}{2^a}} > 2^{-\frac{T-1}{2^{2a}}} - 2^{-\frac{T}{2^a}} \tag{20}$$

It is easy to verify numerically that a sufficient condition for Eq. (20) to hold is

$$a \geq \log T \tag{21}$$

Next, we provide a lower bound on the cost of the round robin algorithm. Let $\sigma_{1:T}$ be the round robin teaching sequence. W.l.o.g., assume that the order of items shown in each round is $1, 2, \ldots, n$. Therefore,

$$\begin{aligned} f(\sigma_{1:T}) &= \frac{1}{nT}\sum_{i=1}^{n}\sum_{\tau=1}^{T} g_i\left(\tau + 1, \sigma_{1:\tau}\right) \\ &= \frac{1}{nT}\sum_{i=1}^{n}\sum_{r=1}^{T/n}\sum_{\tau=1}^{n} g_i\left((r-1)n + \tau + 1, \sigma_{1:(r-1)n+\tau}\right) \\ &\geq \frac{1}{nT}\sum_{i=1}^{n}\sum_{r=1}^{T/n} n g_i\left(rn + 1, \sigma_{1:(r-1)n+\tau}\right) \\ &= \frac{1}{T}\sum_{i=1}^{n}\sum_{r=1}^{T/n} g_i\left(rn + i, \sigma_{1:(r-1)n+\tau}\right) \end{aligned}$$

For simplicity, define $p_{i,r} = g_i\left(rn + i, \sigma_{1:(r-1)n+\tau}\right)$. We thus have

$$f(\sigma_{1:T}) = \frac{1}{T}\sum_{i=1}^{n}\sum_{r=1}^{T/n} p_{i,r} \tag{22}$$

Observe that for $r \in \{1, \ldots, T/n\}$, it holds that

$$\frac{1 - p_{i,r+1}}{1 - p_{i,r}} \geq \frac{1 - p_{i,r+2}}{1 - p_{i,r+1}}, \text{ and } 1 - p_{i,r} \geq 1 - p_{i,r+1} \tag{23}$$

From the above inequalities we get

$$\begin{aligned} 1 - p_{i,r+1} &= \left(1 - p_{i,r}\right)\frac{1 - p_{i,r+1}}{1 - p_{i,r}} \\ &\leq \left(1 - p_{i,r}\right)\frac{1 - p_{i,r}}{1 - p_{i,r-1}} \\ &\leq \left(1 - p_{i,r-1}\right)\frac{1 - p_{i,r-1}}{1 - p_{i,r-2}} \cdot \frac{1 - p_{i,r}}{1 - p_{i,r-1}} \end{aligned}$$

$$\leq (1-p_{i,r-1})\left(\frac{1-p_{i,r-1}}{1-p_{i,r-2}}\right)^2$$
$$\leq (1-p_{i,1})\left(\frac{1-p_{i,2}}{1-p_{i,1}}\right)^r$$

Therefore, we have

$$\begin{aligned}\sum_{r=1}^{T/n}(1-p_{i,r}) &\leq (1-p_{i,1})+(1-p_{i,1})\frac{1-p_{i,2}}{1-p_{i,1}}+\cdots+(1-p_{i,1})\left(\frac{1-p_{i,2}}{1-p_{i,1}}\right)^{T/n-1}\\ &=\sum_{r=1}^{T/n}(1-p_{i,1})\left(\frac{1-p_{i,2}}{1-p_{i,1}}\right)^{r-1}\\ &=\frac{(1-p_{i,1})\left(1-\left(\frac{1-p_{i,2}}{1-p_{i,1}}\right)^{T/n}\right)}{1-\left(\frac{1-p_{i,2}}{1-p_{i,1}}\right)}\\ &\leq \frac{(1-p_{i,1})^2}{p_{i,2}-p_{i,1}}\end{aligned} \tag{24}$$

Combining Eq. (22) with Eq. (24) we get

$$\begin{aligned}f(\sigma_{1:T}) &= \frac{1}{T}\sum_{i=1}^{n}\sum_{r=1}^{T/n}p_{i,r}\\ &=1-\frac{1}{T}\sum_{i=1}^{n}\sum_{r=1}^{T/n}(1-p_{i,r})\\ &\geq 1-\frac{1}{T}\sum_{i=1}^{n}\frac{(1-p_{i,1})^2}{p_{i,2}-p_{i,1}}\\ &\overset{(a)}{=} 1-\frac{n}{T}\frac{(1-p_{i,1})^2}{p_{i,2}-p_{i,1}}\end{aligned}$$

where step (a) is due to the fact that $p_{i,1}=2^{-n/2^a}$, and $p_{i,2}=2^{-n/2^{2a}}$ for all i.

Now suppose that we would like to lower bound the utility $f(\sigma_{1:T})$ by $1-\epsilon$. Therefore,

$$\frac{n}{T}\frac{(1-p_{i,1})^2}{p_{i,2}-p_{i,1}} \leq \epsilon \tag{25}$$

While it is challenging to solve Eq. (25) in an analytical form, we consider a stronger condition to simplify the calculation. Consider a configuration of a which also satisfies the following inequality

$$1-p_{i,2} \leq \frac{1-p_{i,1}}{2} \tag{26}$$

Therefore, a sufficient condition for Inequality (25) to hold is

$$\frac{(1-p_{i,1})^2}{p_{i,2}-p_{i,1}} = \frac{(1-p_{i,1})^2}{(1-p_{i,1})-(1-p_{i,2})} \overset{\text{Eq. (26)}}{\leq} \frac{(1-p_{i,1})^2}{(1-p_{i,1})-\frac{1-p_{i,1}}{2}} = 2(1-p_{i,1}) \leq \frac{\epsilon T}{n}$$

Plugging in $p_{i,1}=2^{-n/2^a}$ into the above inequality, we get

$$2^{-n/2^a} \geq 1-\frac{\epsilon T}{2n} \tag{27}$$

Now, let us consider the following two cases:

C1 $1-\frac{\epsilon T}{2n} > 0$ (that is, $\epsilon < 2n/T$). In this case, we get

$$a \geq \log\left(\frac{n}{\log\left(\frac{1}{1-\epsilon T/(2n)}\right)}\right)$$

$$
\begin{aligned}
&= \log n - \log\log\left(\frac{1}{1-\epsilon T/(2n)}\right) \\
&\overset{(a)}{\geq} \log n - \log\left(\left(\frac{1}{1-\epsilon T/(2n)}\right) - 1\right) \\
&= \log\left(\left(\frac{2n^2}{\epsilon T}\right) - n\right)
\end{aligned}
$$

where step (a) is by the inequality $\log(x) \leq x - 1$ for $x > 0$. A feasible configuration of a satisfying the above inequality is

$$
a \geq \log\left(\frac{2n^2}{\epsilon T}\right) \tag{28}
$$

It is easy to verify that Condition Eq. (28) also satisfies our additional constraint Eq. (26).

C2 A second case is $\epsilon \geq 2n/T$. In this case, Eq. (27) holds for all a, and we only need to find a feasible configuration of a that satisfies Eq. (26). A suitable choice of such a constraint is

$$
a \geq \log(3n) \tag{29}
$$

Combining Eq. (21) Eq. (28) and Eq. (29) we obtain

$$
a \geq \max\left\{\log T, \log(3n), \log\left(\frac{2n^2}{\epsilon T}\right)\right\}
$$

which finishes the proof.