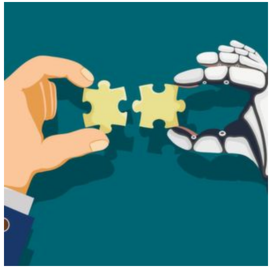


Enhancing human learning via spaced repetition optimization

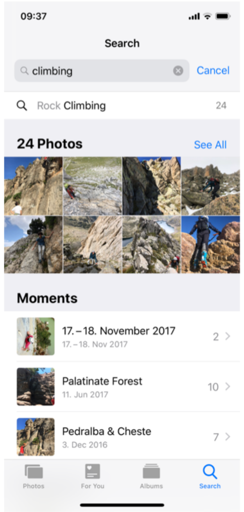
Manuel Gomez Rodriguez
Max Planck Institute for Software Systems

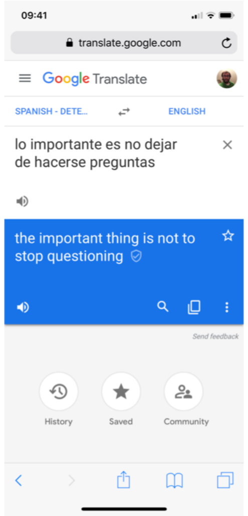
Includes joint work with Behzad Tabibian, Utkarsh Upadhyay, Abir De, Ali Zarezade and Bernhard Schölkopf

MAX PLANCK INSTITUTE
FOR SOFTWARE SYSTEMS

Machine learning for automation

Machine learning has **learned to perform a variety of tasks as well as humans, if not better!**



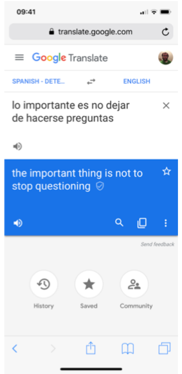


play complex games

recognize images **translate between languages**

Human-centered machine learning

Rather than learning to perform tasks, what about using **machine learning** to **help humans learn *better***?

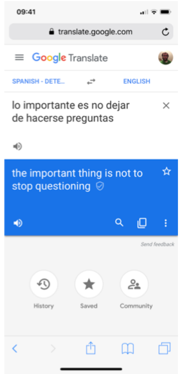


Machine learning translate between Spanish and English

Machine learning helps humans learn Spanish

Human-centered machine learning

Rather than learning to perform tasks, what about using **machine learning** to **help humans learn *better***?



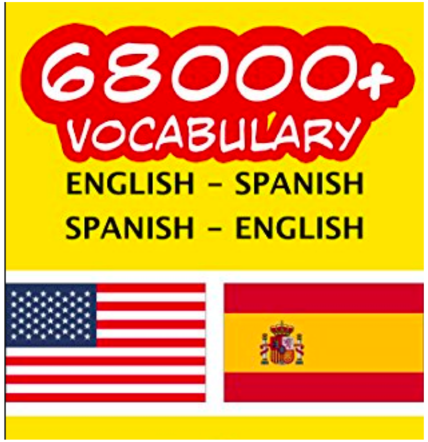
Machine learning translate between Spanish and English

Machine learning helps humans learn Spanish

What is learning? Declarative vs procedural learning

Declarative learning

Acquiring information that one can speak about



Learning a new language's vocabulary

Procedural learning

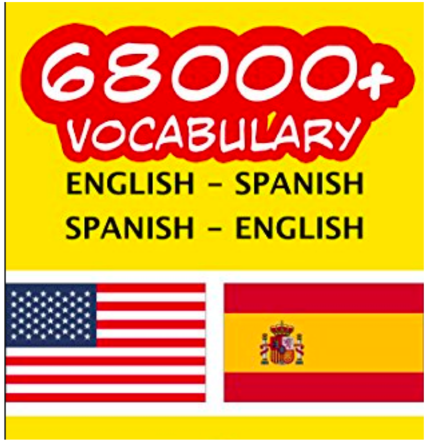
Acquiring mainly motor skills and habits

Learning how to ride a bike

What is learning? Declarative vs procedural learning

Declarative learning

Acquiring information that one can speak about



Learning a new language's vocabulary

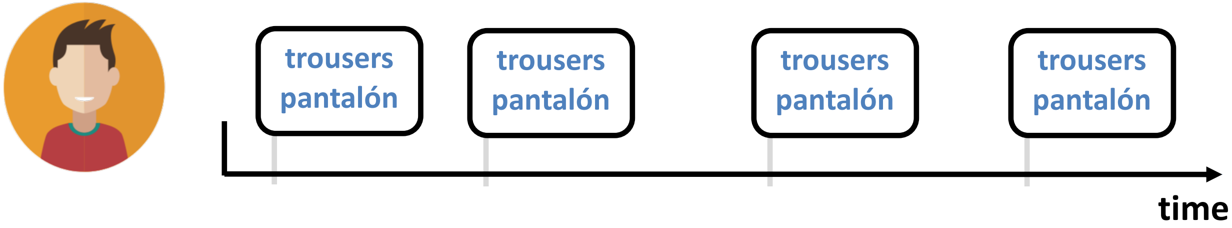
Procedural learning

Acquiring mainly motor skills and habits

Learning how to ride a bike

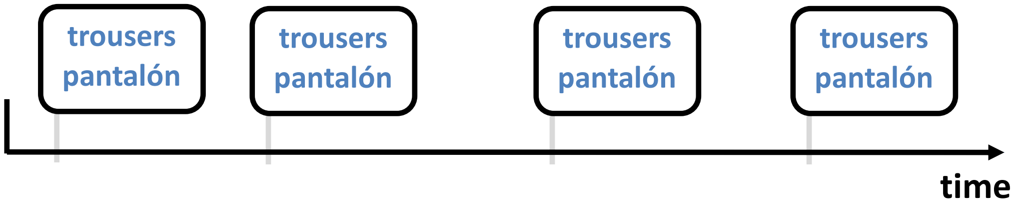
Learning by (spaced) repetition

Humans learn by repetition

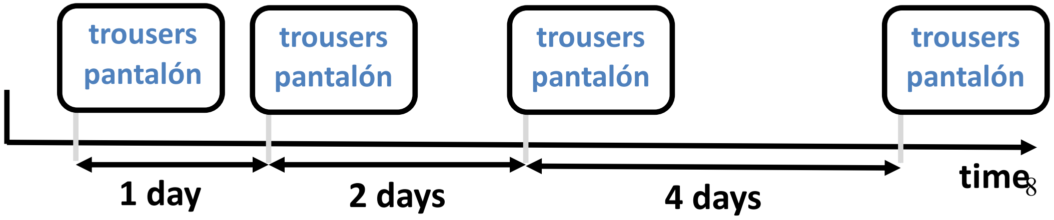


Learning by (spaced) repetition

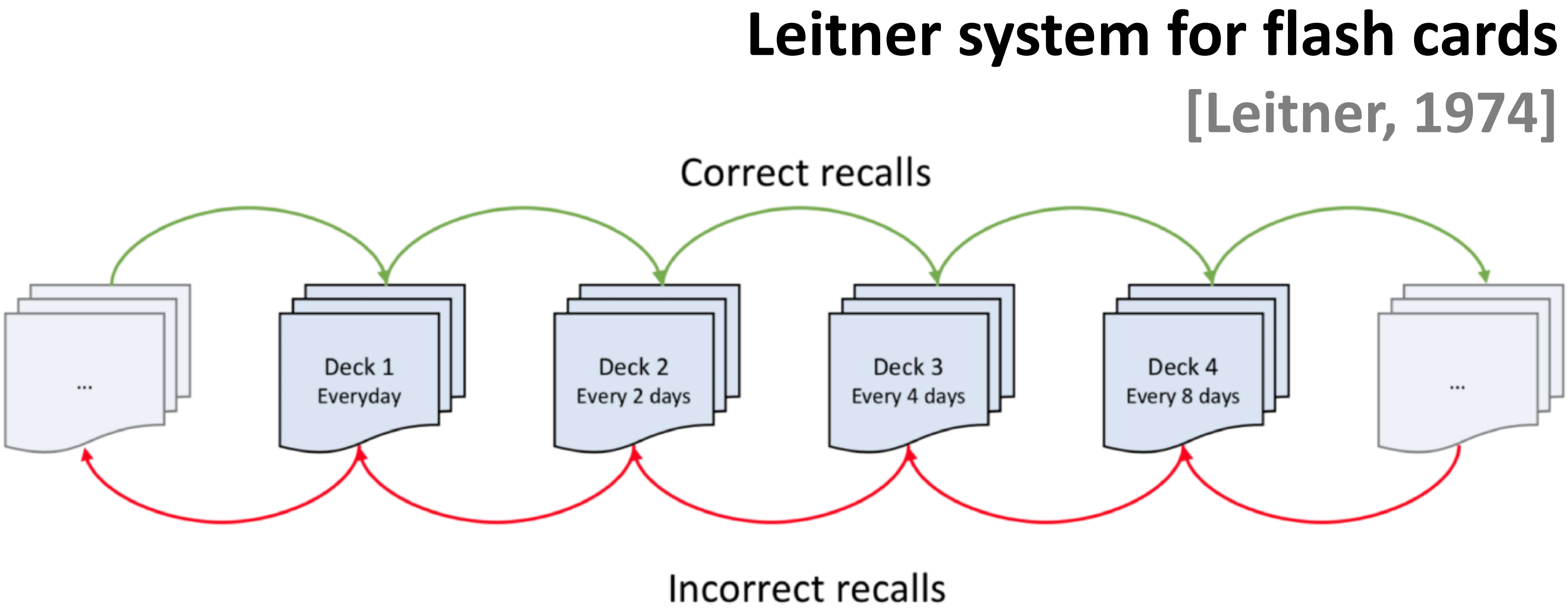
Humans learn by repetition



Since more than a century [Ebbinghaus, 1885], it is known that "spaced" repetition is important



Leitner system for flash cards

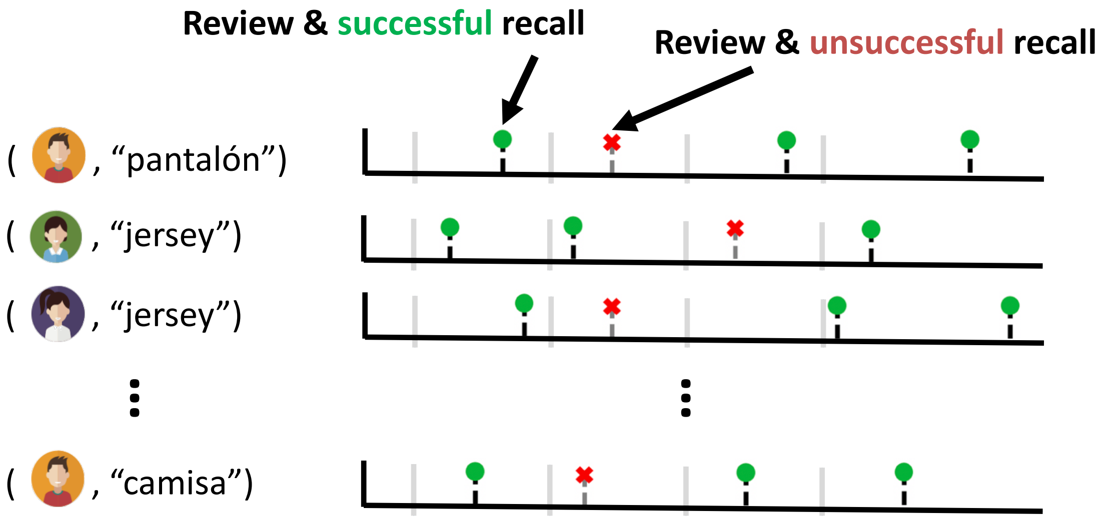


The promise of modern online learning platforms

Use fine-grained monitoring & greater degree of control to optimize when to review

duolingo

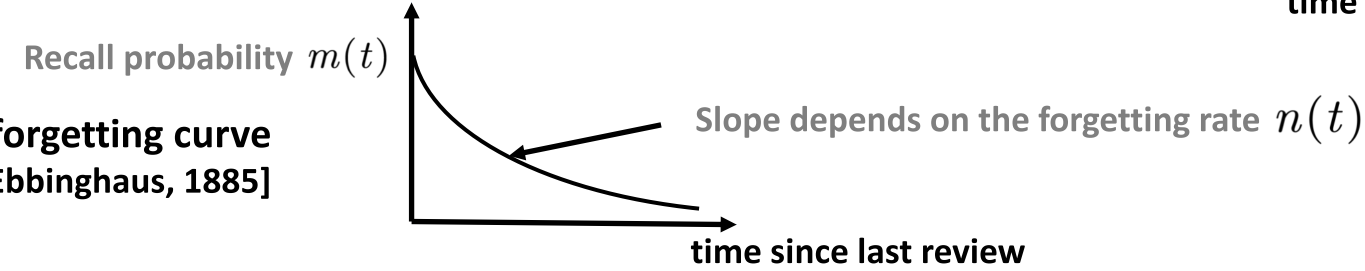
Online learning platform



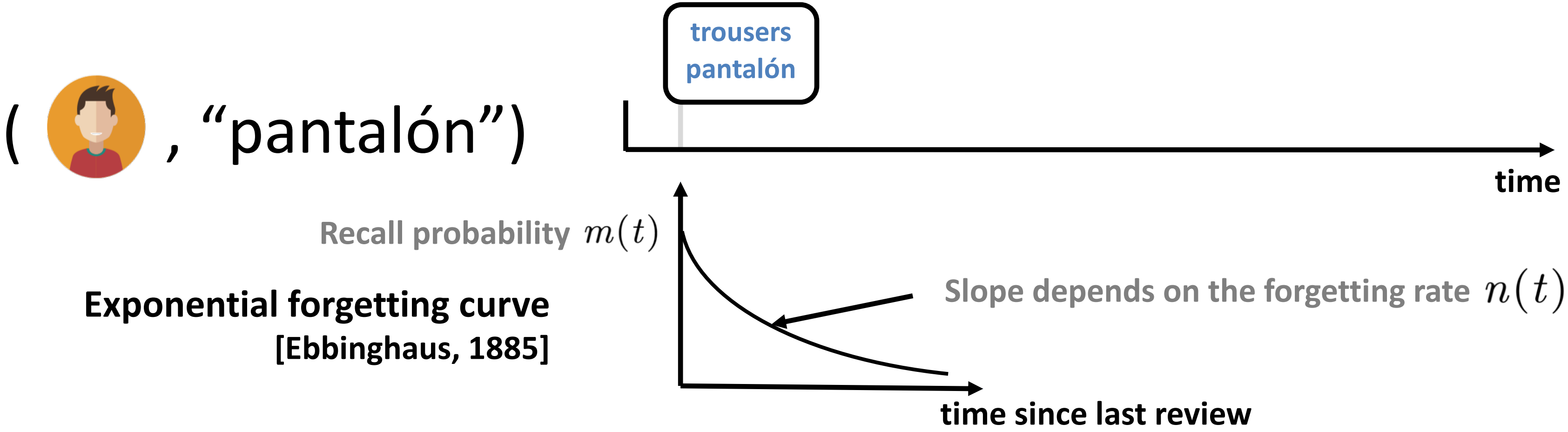
Can we predict when a learner will recall (or forget)?

( , "pantalón")

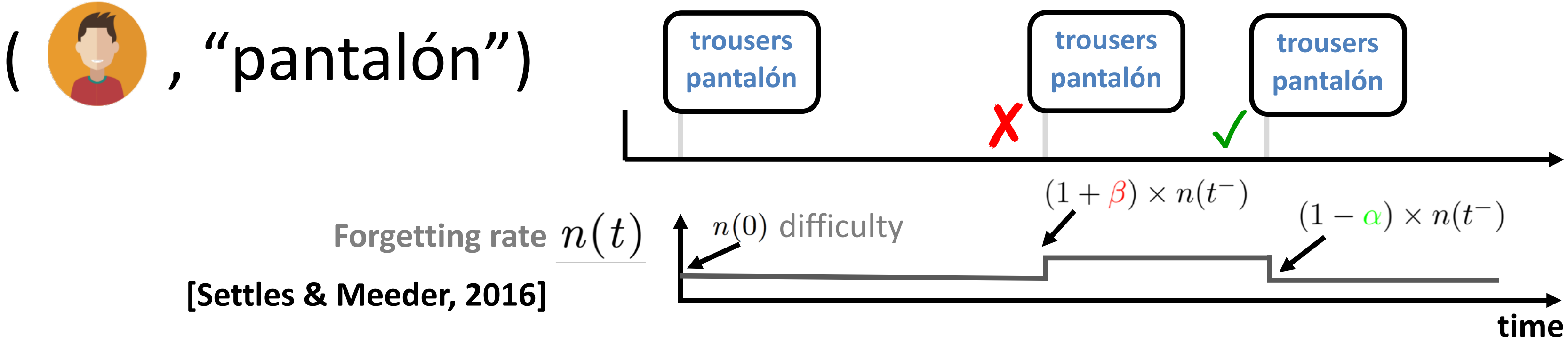
Exponential forgetting curve
[Ebbinghaus, 1885]



Can we predict when a learner will recall (or forget)?



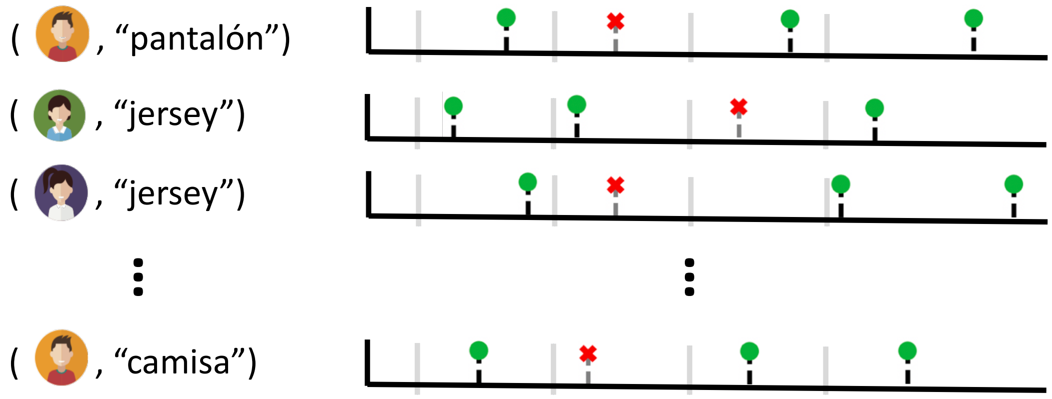
Exponential forgetting curve
[Ebbinghaus, 1885]



[Settles & Meeder, 2016]

A machine learning memory model

Given enough **historical learning data:**



We can use a variant of **half-life regression [Settles & Meeder, 2016]** to fit the model parameters, i.e.,

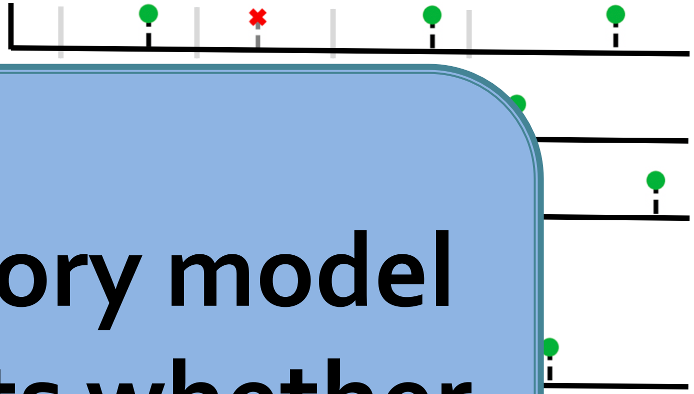
One difficulty parameter $n_i(0)$ per item i

Single set of parameters for α and β for all items

A machine learning memory model

Given enough **historical** (, “pantalón”)

learnin



The (fitted) memory model accurately predicts whether a learner will recall a word!

We ca 16]

to fit

One

Single set of parameters for α and β for all items

Reviewing rates to decide when to review

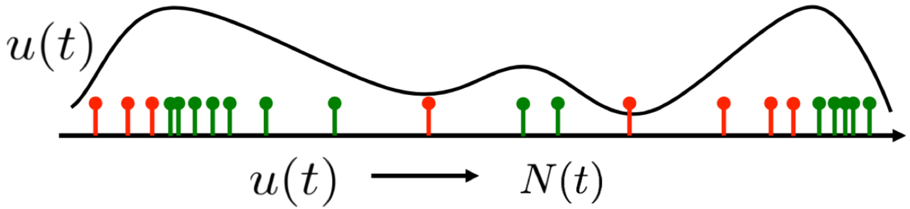
Instead of **optimizing** for the **exact reviewing time,** we will **optimize** for the **optimal** ***rate of reviewing***

Agent

Online learning platform

Environment

Learner



Rate of reviewing

Counting process

Reviewing rates to decide when to review

Instead of **optimizing** for the **exact reviewing time,** we will **optimize** for the **optimal** ***rate of reviewing***

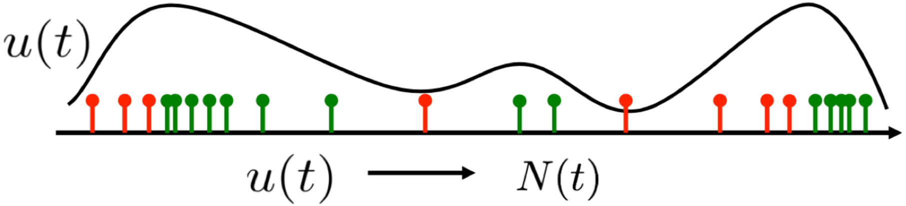
Agent

duolingo

Online learning platform

Environment

Learner



Rate of reviewing **Counting process**

Under this view, we can **express the memory model and rate of reviewing** as a **dynamical system**

Finding the optimal reviewing rates

Goal: optimally **trade off recall and reviewing rate for all items over time, i.e., minimizing the loss**

Parameter that controls **trade off**

$$\ell(m(t), u(t)) = \frac{1}{2}\sum_{i\in\mathcal{I}}(1 - \underbrace{m_i(t)}_{\text{Recall probability}})^2 + \frac{1}{2}\sum_{i\in\mathcal{I}} q_i \underbrace{u_i^2(t)}_{\text{Reviewing rate}}$$

Recall probability

Reviewing rate

Finding the optimal reviewing rates

Goal: optimally **trade off recall and reviewing rate for all items over time, i.e., minimizing the loss**

Parameter that controls **trade off**

$$\ell(m(t), u(t)) = \frac{1}{2} \sum_{i \in \mathcal{I}} (1 - \underbrace{m_i(t)}_{\text{Recall probability}})^2 + \frac{1}{2} \sum_{i \in \mathcal{I}} q_i \underbrace{u_i^2(t)}_{\text{Reviewing rate}}$$

➡ We can use **stochastic optimal control** to **show that** the **optimal reviewing rate per item is:**

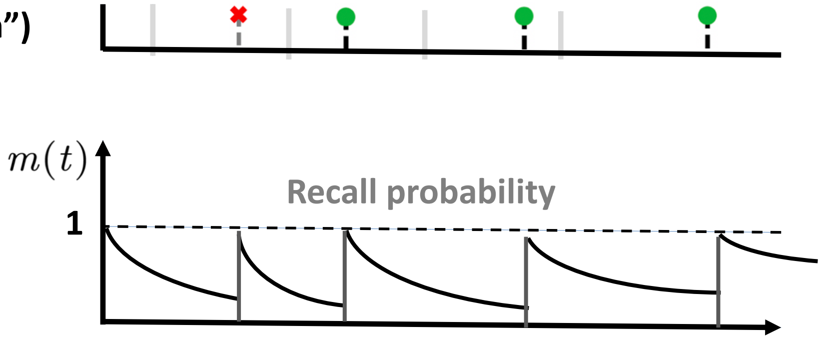
$$u_i^*(t) = q_i^{-1/2}(1 - m_i(t))$$

The Memorize algorithm

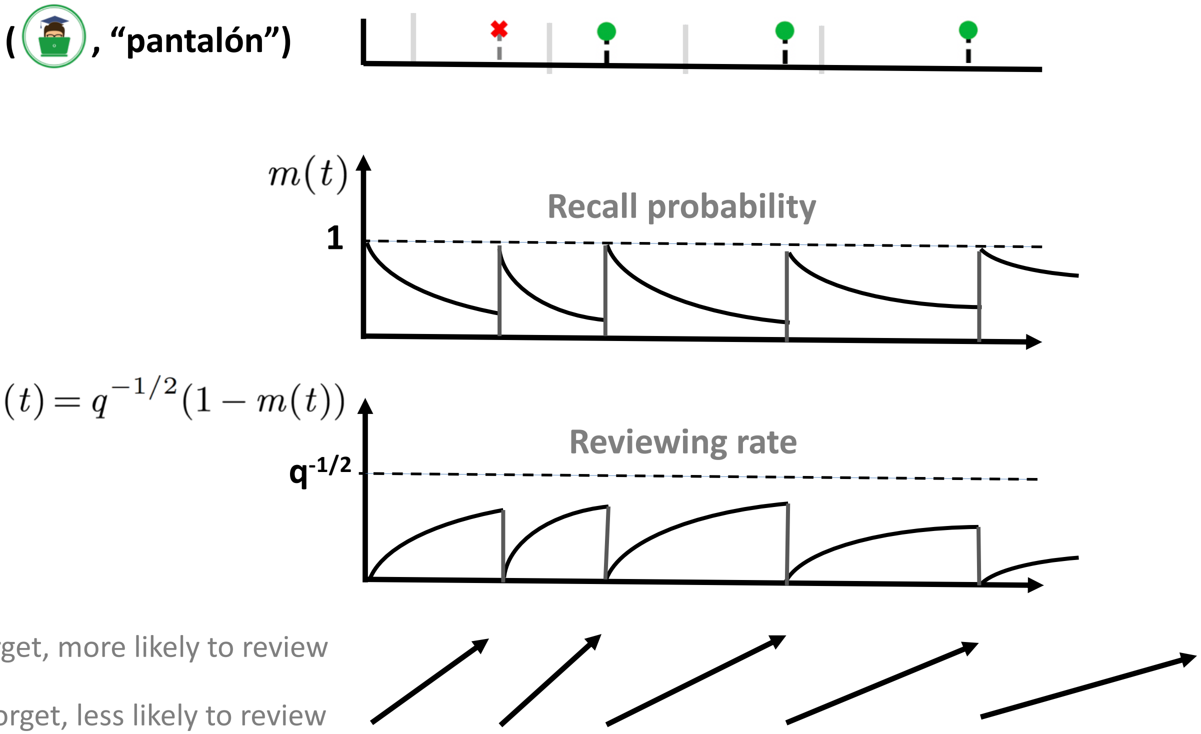
( , “pantalón”)

The Memorize algorithm

(, "pantalón")



The Memorize algorithm



$$u^*(t) = q^{-1/2}(1 - m(t))$$

More likely to forget, more likely to review

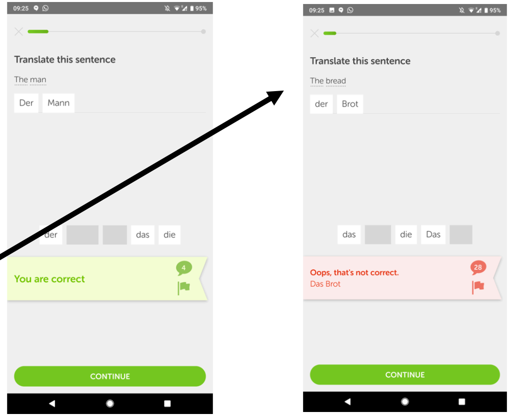
Less likely to forget, less likely to review

Experiments on Duolingo

Natural experiment **using data from Duolingo, a popular language-learning online platform:**

➡ **12 million study sessions during 2 weeks**

In each session, a learner answers multiple questions with multiple words

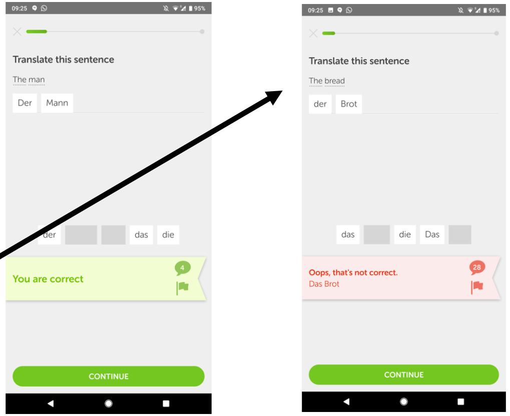


Experiments on Duolingo

Natural experiment **using data from Duolingo, a popular language-learning online platform:**

➡ **12 million study sessions during 2 weeks**

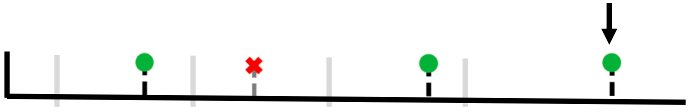
In each session, a learner answers multiple questions with multiple words



➡ **5.3 million unique (user, word) pairs**

$r_i(t) = 1$ ➡ **Successful recall:** The **learner answered** all **the questions** containing **“pantalón” correctly**

(, “pantalón”)



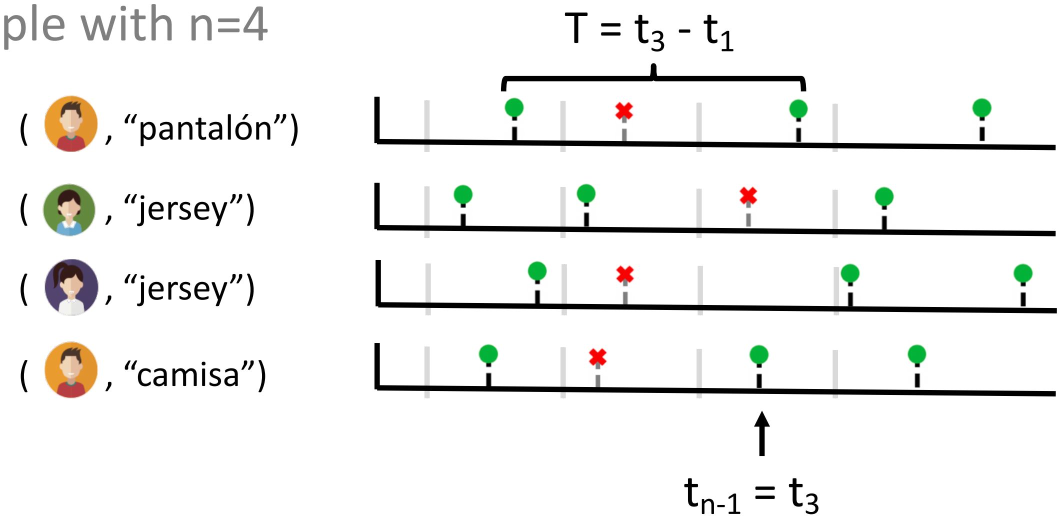
Each event ➡ one session

$r_i(t) = 0$ ➡ **Unsuccessful recall: >= 1 question wrong** with **“pantalón”**

How do we compare different reviewing algorithms?

(1) We **group (user, word) pairs by** their **number of reviews n** and their **training period $T = t_{n-1} - t_1$**

Example with n=4

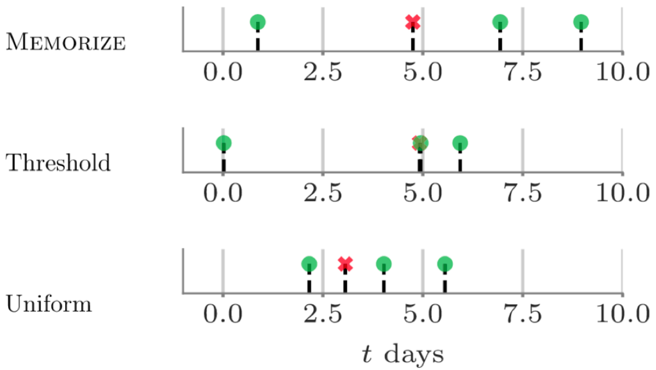


Key idea

First n-1 reviews as ***study*** and **last review n** as ***test***

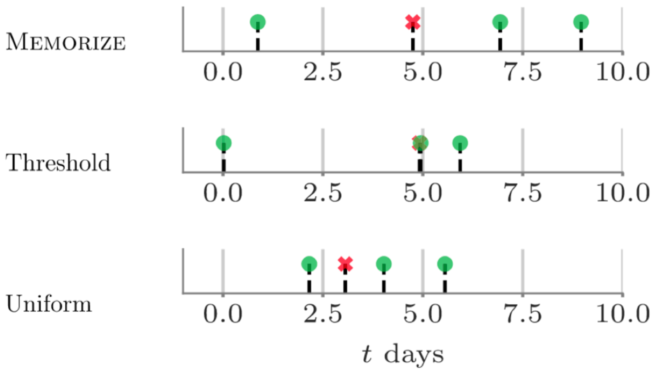
How do we compare different reviewing algorithms?

(2) For each recall pattern, we pick **top 25% pairs in terms of log-likelihood** for all **reviewing methods**

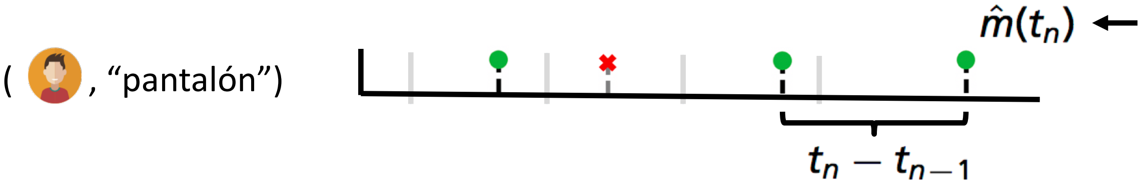


How do we compare different reviewing algorithms?

(2) For each recall pattern, we pick **top 25% pairs in terms of log-likelihood** for all **reviewing methods**



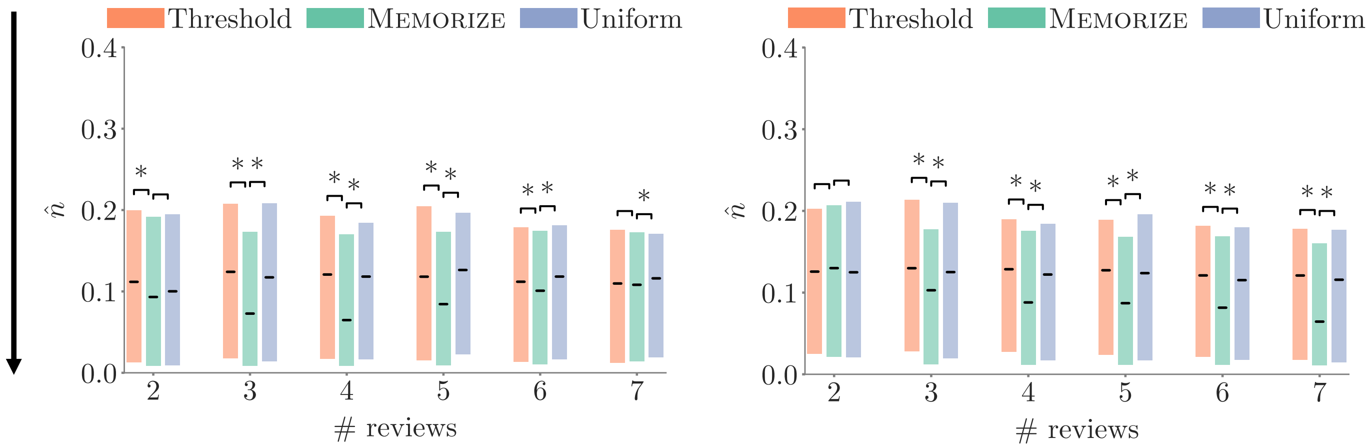
(3) For each pair, we compute an **empirical estimate of the forgetting rate using** only the **last study session** and the **retention interval $t_n – t_{n-1}$**



← Fraction of correct recalls in last study session

Performance vs study duration and # of reviews

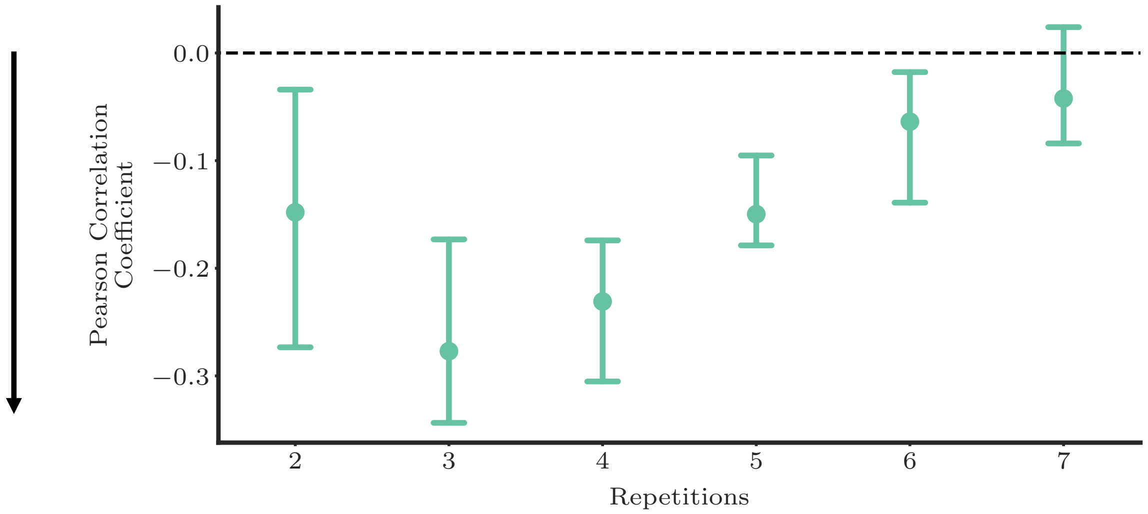
Lower empirical forgetting rate is better



MEMORIZE offers a **clear competitive advantage** with respect to the **uniform** and the **threshold-based baselines**

Can we tell something about a specific learner?

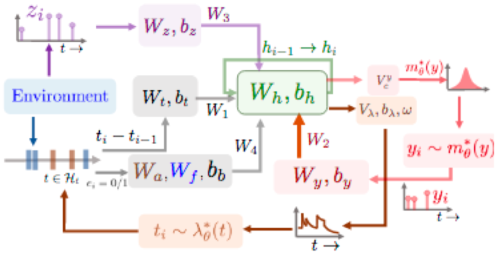
Lower correlation,
greater gains
due to MEMORIZE



Whenever **a specific learner follows Memorize** more closely, her **performance** is **superior**

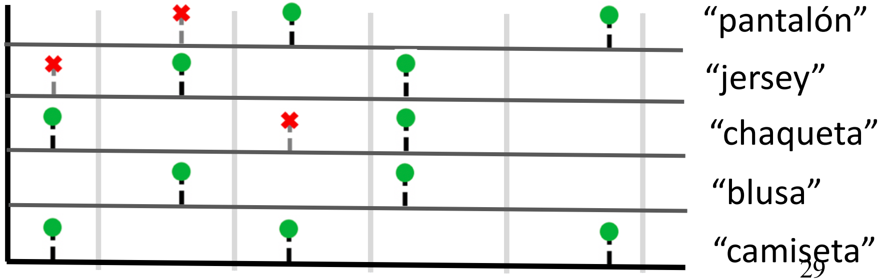
More on optimizing spaced repetition

Beyond quadratic losses and parametric memory models ➡ Approach based on deep RL
[Upadhyay et al., NeurIPS 2018]



From **spaced repetition** to **spaced selection**

Learner chooses time of review, we **optimize items per session** ➡
[Hunziker et al., NeurIPS 2019]



What's next?

Interventional experiments on **apps** for **personalized learning** developed by **swift.ch**

In progress:

	Variant	Improvement	Probability to beat baseline	Probability to be best variant
Random →	Control group 7,068 users	Baseline	Baseline	<0.1%
Previous (non-ML) algorithm →	Variant A 3,902 users	60.4% 50.8% to 70.6%	>99.9%	16%
Our algorithm →	Variant B 3,932 users	65.3% 55.5% to 75.7%	>99.9%	84%

Thanks!

For more details, have a look at our **paper** at **PNAS, 2019**

RESEARCH ARTICLE

Enhancing human learning via spaced repetition optimization

Behzad Tabibian, Utkarsh Upadhyay, Abir De, Ali Zarezade, Bernhard Schölkopf, and Manuel Gomez-Rodriguez

PNAS March 5, 2019 116 (10) 3988-3993; first published January 22, 2019 https://doi.org/10.1073/pnas.1815156116

Edited by Richard M. Shiffrin, Indiana University, Bloomington, IN, and approved December 14, 2018 (received for review September 3, 2018)

and **code**

github.com/Networks-Learning/memorize