Estimating Memory Retention Traces of Foreign Language Vocabulary from Reading Interaction Data

Carlos Pérez-Guerra, Jian Cao
School of Electrical and Computer Engineering
Shanghai Jiaotong University
800 Dongchuan Road, Shanghai, China
charliewar89@sjtu.edu.cn
cao-jian@sjtu.edu.cn

***Abstract*—While language-learning apps have empowered many users to learn languages inexpensively, the underlying technology limits the tasks they can offer to atomic, flashcard-like exercises. In this paper we present an algorithm that can effectively estimate memory traces from user interaction data that is generated in complex language learning tasks such as extensive reading. This technology can be applied to develop complex, intrinsically-motivating, high-value learning tasks based around long texts, television series or podcasts.**

***Index Terms*—spaced repetition; weakly supervised learning; memory retention; language learning**

Introduction

As technology permeates more and more aspects of our lives, developments in the fields of CALL (Computer-Assisted Language Learning) and its mobile counterpart MALL have empowered many foreign language learners to study languages more cheaply, conveniently and effectively than before. Companies such as Duolingo or Busuu are able to teach languages by splitting the required knowledge into atomic units (such as grammar rules or vocabulary items) and using Spaced-Repetition Scheduling (SRS) algorithms to schedule reviews. The latter are simple heuristics which increase or decrease the interval until the next review based on whether the user correctly recalled the item or not.

In spite of their relative success, there remain challenges in creating platforms that can match the efficacy of a traditional classroom setting. The explicit, atomic teaching nature of these platforms, consisting of single-word or single-sentence direct translation exercises simply cannot replace complex, self-directed, immersive activities such as long reading or listening tasks. This latter type of tasks is of great value to learners because they enable unconscious language acquisition to take place [1] and because they are intrinsically motivating (unlike revising flashcards).

How to support learners when engaging in complex language tasks, however, remains an open problem. If the data from user interactions when engaging in complex tasks could be mined and used to predict when a user is likely to benefit from revising a word, i.e. inferring the so-called *probability of recall* or *memory trace* distribution, then much higher-value software could be built. It would allow for truly adaptive, personalized learning experiences based around self-directed, complex language tasks: assisting learners in acquiring the specific vocabulary used in their favourite television series, podcasts or books; helping professionals in remembering foreign language vocabulary specific to their fields, or indeed faithfully replicating the mix of complex, self-directed language tasks and simple, individualized exercises that make one-to-one tutoring so effective.

The SRS heuristics which dominate the field are not the right paradigm for supporting learners in such tasks. Particularly, as language learners wish to bridge the gap between the intermediate and advanced levels, thousands of new words will have to be learnt, and it is at this stage when the problems of SRS become apparent: the wealth of data generated by users while they are interacting with extensive reading or videos goes unused, and additionally the implied exponential decay model of SRS heuristics underestimates users' memory traces, leading to many reviews being prematurely scheduled. As we discuss below, this lack of adaptiveness and inefficiency lead to user dissatisfaction and the high attrition rates which have been reported for these platforms.

In this paper, we present an alternative to the SRS paradigm. Our novel approach can approximate memory traces from *unscheduled* exposures, enabling revision or other exercises to be prioritized in a way that efficiently adapts to the immersive, high-value tasks we have been discussing. This, however, is a hard problem because the ground truth is not known. Our pipeline overcomes this by taking several general-purpose supervised learning procedures and modifying them to embed domain knowledge, which acts as a strong prior, thereby creating a specialized pipeline, an approach often referred to as weakly supervised learning.

Our contributions in this paper can be summarized as follows:

- A critical appraisal of the claims made by Duolingo regarding their popular Half-Life Regression algorithm [2]. Specifically, we show that the HLR approach fails to address the unlabeled nature of the data and does not learn a memory trace but simply learns a trivial *trend-following* model (i.e. that a user will continue to recall or forget indefinitely). We also contest Duolingo's claimed

TABLE I
SUMMARY OF POPULAR FUNCTIONAL FORMS IN MEMORY RETENTION MODELING LITERATURE.

Abbr.	Name	Original Form	Simplified
(Exp)	Exponential	$R = \beta \cdot \exp(-\frac{\Delta}{\mu})$	$\exp(-\frac{\Delta}{\mu})$
(ESq)	Exponential Sqrt.	$R = \beta \cdot \exp(-\frac{\sqrt{\Delta}}{\mu})$	$\exp(-\frac{\sqrt{\Delta}}{\mu})$
(Hyp)	Hyperbolic	$R = \frac{1}{\beta+\frac{\Delta}{\mu}}$	$\frac{1}{1+\frac{\Delta}{\mu}}$
(HSq)	Hyperbolic Sqrt.	$R = \frac{1}{\beta+\frac{\sqrt{\Delta}}{\mu}}$	$\frac{1}{1+\frac{\sqrt{\Delta}}{\mu}}$
(Log)	Logarithmic	$R = \beta - \frac{\ln \Delta}{\mu}$	
(Pwr)	Power	$R = \beta \cdot \Delta^{-\frac{1}{\mu}}$	
(SWP)	Sim. Wickelgren Pwr	$R = \lambda(1 + \beta \cdot \Delta)^{-\frac{1}{\mu}}$	$(1 + \Delta)^{-\frac{1}{\mu}}$

results and show that trivial models can perform much better than their algorithm when evaluated using their chosen metrics.

- We present **Memory-Trace Regression (MTR)**, a weakly supervised regression pipeline which can reliably estimate vocabulary memory traces from unscheduled exposures by leveraging domain knowledge.
- We evaluate the MTR pipeline by means of real click-stream and scrollstream data collected from a custom-built assisted reading app. This dataset, which we have released to the public domain, is the only publicly available dataset which captures vocabulary acquisition from freely-directed reading and entirely unscheduled exposures.[1]

BACKGROUND

Memory retention models

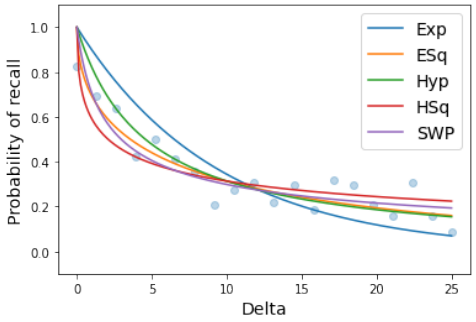
The field of memory retention modeling seeks to explain the effect of elapsed time on memory decay. It is of great interest to cognitive scientists and experimental psychologists, who have developed precise mathematical models to explain the scores of experiments that have been carried out since the late 19th century. The methodology used in many of these experiments usually involves exposing test subjects to a series of images or nonsense syllables and recording how many of these are remembered after Δ time. The resulting *retention rate*, or *probability of recall* distribution, which we will denote as R throughout the paper, can often be approximated quite precisely (up to $\sim 95\%$ explained variance [3]) with the two-parameter models which are summarised in Table I. In the notation used in the table, Δ represents the time elapsed since the last exposure to a specific item, μ represents the memory strength of the trace, and β is a parameter. In some cases β is simply a scaling parameter of Δ, so the models only have a single true degree of freedom if we assume $\Delta \in [0, 1]$; the table presents these simplified forms as well.

An extensive, authoritative meta-study [3] of 210 datasets, found Log, ESq, HSq and Pwr to have the best fitting ability out of the 105 2-parameter surveyed functional forms. These models are characterized by a decreasing rate of decay, i.e. the retention rate decays rapidly for a short period of time and then stabilizes above an asymptote or decays very slowly depending on the magnitude of μ. While no model was consistently the best fit across all datasets, the mean explained variance for the best fit for each dataset was 95%. Additionally, the paper found exponential decay models (Exp) to have the second-worst fitting ability, second only to a purely linear form.

Another popular model is Wickelgren's Power Function [4], as well as its simplified, single-degree of freedom version (SWP) [5], which addresses the discontinuity of Pwr when $\Delta = 0$.

Fig. 1 illustrates the fitting abilities of these models to synthetically generated data.

Fig. 1. Fits of popular single degree of freedom models to synthetically generated data.



Our work is indebted to the almost century and a half of research in this direction, and our contribution is in showing how to leverage these elegant models to estimate memory traces when the ground truth is unknown.

Spaced-Repetition Scheduling

The SRS family of algorithms is used to schedule vocabulary reviews in many language-learning applications. They offer a solution to the spaced-repetition scheduling problem: Given a set of items $\mathcal{I}$, and a fixed total amount of time T which can be devoted to studying these items, how should the review sessions for each item be allocated to maximize the average retention rate at the end of some predetermined time interval?

SRS algorithms apply a heuristic to systematically lengthen the interval between reviews on successful recall and shorten it otherwise. Several SRS algorithms have been generalized [2] to the form $R = 2^{\frac{\Delta}{\mu}}$, where $\mu = 2^{\theta^T x}$. In this formulation, the optimal time to review, according to the algorithm, is given when $R = 0.5$.

The main limitation of SRS algorithms is that they are only capable of mapping successes or failures that occur at scheduled times to future review times, forcing users to adapt to the schedules and limiting the sort of tasks that software platforms can offer to simple, atomic, flashcard-like exercises. As a result, high user attrition rates have been reported: for example, a study of 62 college students of Spanish found

[1]The repository located at www.github.com/csalg/mtr contains the dataset used for our evaluation as well as a Jupyter notebook to reproduce all the tables and figures used throughout the paper.

that 'students were reluctant to use the app and reported low enjoyment', which resulted in a user attrition rate of more 80% by the end of the 11-week study [6].

Additionally, there is inconclusive evidence for the improved effectiveness assumed by the problem statement in spacing reviews: [7] found massed readings (five times in one day), to be as effective as spaced readings (five times daily), and the ten-year overview in [8] concluded that the advantages of spacing reviews are only appreciated 'under a highly specific set of conditions'. The results of studies comparing fixed and spaced study regimes are mixed, and at best concede just a mild advantage for spaced strategies [9], [10]. There is, however, a well-known correlation between testing and increased retention rate, and the purported benefits of SRS could simply be attributed to additional testing, regardless of scheduling [11].

Weakly Supervised Learning

Weakly Supervised Learning attempts to find ways to work with data that is labeled incorrectly or inexactly, and has been successful in tasks as diverse as pixel-level annotation of images from coarse labels [12], biomedical text mining [13] and even factor analysis of particle phenomena [14]. In the case of inferring more detailed information from coarse labels, the methodology generally involves treating the problem as underconstrained and using prior domain knowledge to prune the hypothesis search space. This can be done through rule-based / generative data annotation [15], appending a prior to the assumed population distribution and regressing on the conditional distribution [16], and the use of custom evaluation metrics and loss functions [14].

A Critical Appraisal of HLR

Before introducing our procedure, we shall conduct a critical appraisal of the recent, highly influential HLR algorithm [2]. It shall serve as a cautionary tale against naively posing memory tracing as a supervised learning problem and motivate our design in the next section.

HLR was developed by researchers at Duolingo, and generalizes several SRS algorithms to the form $R = 2^{\frac{\Delta}{\mu}}$, where $\mu = 2^{\theta^T x}$. The main claim of the paper is that by posing the problem as a supervised regression problem and fitting the weights θ, a memory trace can be approximated. We will not only prove that this is far from the case, and HLR simply learns to predict that users will continue succeeding or failing indefinitely, but *a fortiori* that the problem is ill-posed because trivial trend-following models can minimize the metrics, whereas actual memory traces would score poorly.

The publicly available dataset has dimensions $12.9\text{M} \times 12$. We will refer to different features using the following notation: Δ denotes the time elapsed since the last known exposure; 'history_seen' and 'history_correct' are denoted by H; and 'session_seen' and 'session_correct' by S.

An important characteristic of this dataset is the extreme skewness of the label distribution. The label 'p_recall' is valued $[0, 1]$ and most of the values are clustered around 0 and 1, with $>85\%$ of the values equal to 0.99, and $>94\%$ greater than 0.5 (Fig. 2). The original paper does not address how these labels were cast.

TABLE II
Predictive performance of several estimators on the Duolingo dataset

Regressor	X	MAE	MAE($y < 0.5$)	W. Acc.
Logistic Regression	H	0.466	0.486	0.538
Logistic Regression	S	0.0335	0.0111	1.00
Logistic Regression	Δ	0.467	0.488	0.517
Logistic Regression	$H^\frown S$	0.0335	0.0111	1.00
Logistic Regression	$H^\frown S^\frown \Delta$	0.0335	0.0111	1.00
HLR	H	0.150	0.901	0.506
HLR	S	0.0943	0.378	0.789
HLR	Δ	0.124	0.953	0.500
HLR	$H^\frown S$	0.0937	0.377	0.790
HLR	$H^\frown S^\frown \Delta$	0.0785	0.369	0.791

HLR and Logistic Regression were trained using 5-fold, 2-pass CV on subsets of the dataset, denoted using the $^\frown$ notation to refer to the concatenation operator (e.g. the subset containing the history and session features is denoted as $H^\frown S$). Table II evaluates the performance of the estimators using the MAE (which is the metric chosen by the authors of the original paper), the MAE of values of $y < 0.5$, denoted MAE($y < 0.5$), as well as the weighted binary accuracy[2].

It is clear from the table that something is not right. The best performing regressor according to these metrics is the baseline, Logistic Regression, and furthermore it only requires two variables, 'session_seen' and 'session_correct', to obtain 100% accurate predictions. Additionally, when S is removed from the dataset, both regressors lose their ability to accurately predict values where $y < 0.5$. The S dataset does not include Δ, so it is clear that no memory trace is being inferred; instead, both estimators are assuming that a user will continue to fail or succeed indefinitely based on how many successes or failures happened in the session.

We will use the term *trend-following* to refer to this phenomenon of estimators learning a trivial model which performs very well in general-purpose metrics by simply assuming users will continue to fail or succeed indefinitely. This phenomenon has also been reported for Deep Knowledge Networks in the somewhat related task of knowledge component mastery estimation [17]. The main problem with trend-following is that these models lack the ability to predict trend-reversals, which is exactly what is necessary to assist users in structuring their revision.

In closing, the two main flaws with the HLR approach can be summarised as follows:

- A *naive data annotation procedure*. A continuous, ground truth value of R cannot be observed, however the proce-

[2]Our results contradict the results in the original paper, where the MAE for Logistic Regression is claimed to be 0.211. These results were fit using the implementation in the popular 'sklearn' library for Python 3.8.5, whereas the original implementation used custom code, which appears to be incorrect. The authors have been contacted with our results.

dure assumes that the quasi-binary-valued 'p_recall' label corresponds to the ground truth.
- The use of *general-purpose evaluation metrics* to assess the generalization ability of the fit. This is problematic because the ground truth is unknown, and because such metrics can be minimized by trivial trend-following models more easily than by approximations of the memory trace distribution (which, we recall, stabilizes far from the bounds, and hence would produce a high error rate when taking the norm of the residuals except for very large or very small Δ).

MEMORY-TRACE REGRESSION

Our proposed MTR pipeline addresses the limitations inherent in naively posing the problem as a supervised regression problem by embedding domain knowledge in each of the data annotation, fitting and evaluation phases.

Data annotation

At the very beginning of our pipeline, the only data available to us are the clickstream and scrollstream logs $\mathcal{L}$ of the form (t_i, l_i, u_i, m_i), which respectively stand for timestamp, lemma, user id and message, such as (1602784275, 'run', 'user843', 'WORD_WAS_CLICKED'). Our immediate goal is to map $\mathcal{L} \to (X, \Delta, \tilde{y})$, where X is a matrix of predictors, Δ is a vector of time elapsed since the last known exposure to the word, and $\tilde{y}$ is a label that coarsely approximates the retention rate for a time period.

The messages in the logs can be interpreted as meaning that the user correctly recalled a word, failed to recall a word or neither. When interpreted like this, only a small minority of the logs will be recall failures, and the vast majority will be successful recalls or unknown, a skewed distribution which mirrors the Duolingo dataset. Some of this skewness is due to users choosing texts which are understandable, and therefore comprising mostly of vocabulary which is recalled easily. However much of it is explained by *localized high-frequency*: as a direct consequence of Zipf's law [18], words tend to appear nearby, and users can remember the meaning of a word after clicking once or twice, being further aided by context. This acts as a confounding variable and invalidates the i.i.d. assumption for recalls.

The localized high-frequency effect is addressed using a rule-based labeling heuristic:

The time-domain is split into intervals of equal length and $s([t_{i-1}, t_i])$, the *recall score* for the time interval, is calculated using the number of recalls and failures (denoted x^+, x^- respectively) for the interval, weighing the failures linearly and the recalls sublinearly. MTR uses

$$s([t_{i-1}, t_i]) \triangleq \frac{\sqrt{x^+}}{x^- + \sqrt{x^+}}$$

Partitioning the time domain into intervals also allows the use of nearby recall scores to smooth the data by applying interpolation and moving averages; in the case of MTR, linear interpolation is applied to fill the time steps with missing data[3], followed by a bi-directional exponential moving average. This has the effect of increasing a low score if it happens in the neighbourhood of high scores and vice versa.

At the end of the procedure we arrive at the *smoothed recall score*, which shall serve as our target $\tilde{y}$, since it is a coarse approximation of continuous values arising from the population retention rate R.

The rest of the data is annotated as follows: Δ is simply the time elapsed between the end of the previous timestep and X is a collection of summary statistics describing the log stream: counters, intervals and streaks, as well as any desired custom-engineered features.

Fitting

The memory trace curve can be described by the functional forms in Table I. Since these are single degree of freedom models, they act as a strong prior and drastically prune our hypothesis search space. These functional forms are hard to train without some additional manipulation: their form should be changed to make μ the dependent variable. The latent variable μ, unlike R, enjoys many desireable properties for painless regressability: it is continuous, positive-valued, often monotonically increasing, etc. Any regression technique that is robust to noisy labels can then be used to regress on μ, from which $y_{\text{predicted}}$ can be trivially calculated.

Since X is used to predict μ, the memory strength, it should not include Δ or any other values that change in between the last exposure and the current time.

Model Evaluation

Our assumption that $\tilde{y}$ is coarse and noisy (non-Gaussian noise) invalidates any theoretical argument for using general-purpose metrics such as the MSE or the MAE. Additionally, due to the nature of the available data, label values are biased towards the bounds of the $[0, 1]$ range, and hence residuals can still be minimized with trivial, trend-following models (although not to the extent seen in the previous section). However one silver lining of this bias towards the bounds is that information on *trend-reversals* is preserved. Since predicting trend-reversals can only be done reliably by a close estimate of R, we can build metrics around this characteristic of our data and desired estimator. The effect is amplified as Δ increases, because the rate of decay must be very close to R in order to predict *long-term trend-reversals* with minimum error. Evaluation according to these effects can be accomplished by means of the weighted MSE:

$$\text{WMSE}(w)(\tilde{y}, \tilde{y}_{\text{predicted}}) \triangleq |w \odot (\tilde{y} - \tilde{y}_{\text{predicted}})|_2^2$$

where $\odot$ is the element-wise (Hadamard) product.

The weight w allows weighing some residuals more heavily than others, and can be used to encode the characteristics discussed above. The trend-reversal weight for each residual

[3]These interpolated values are only used for calculating moving averages and not as labels.

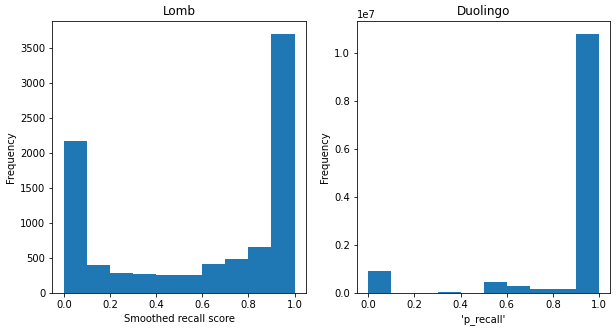
can be quantified using the distance between $\tilde{y}$ and its predecessor $\tilde{y}_{\text{previous}}$, denoted $\tau \triangleq |\tilde{y} - \tilde{y}_{\text{previous}}|$, whereas the product $\tau \odot \Delta$ is used as a weight to assess long-term trend-reversal predictive ability.

Evaluation

In order to assess the performance of our procedure, an assisted reading app named *Lomb* was built, in the style of *ReadLang* (www.readlang.com) and *LingQ* (www.lingq.com). The app assists users in the task of reading texts written in a foreign language by presenting translations for highighted words, as well as sentence translations for clicked sentences. The highlighted words were assembled in a revision panel, and users could click on the words to see the definition or scroll down. All of these events were tracked and a log with the lemmatized form of the word was persisted into a database. The messages 'TEXT__SENTENCE_READ' and 'REVISION__NOT_CLICKED' were interpreted as successful recalls, 'TEXT__WORD_WAS_HIGHLIGHTED' and 'REVISION__CLICKED' were interpreted as failures.

In total, $\sim$ 89K logs were collected over a period of 4 months. A significant limitation of our dataset is that it was mostly generated by two users out of 13 users who participated in the platform, hence this data is not suitable for training a model that generalizes well to unseen data from many users. However, to the best of our knowledge, this is the only publicly available dataset which captures users engaging in a complex, freely-directed task in a foreign language with no scheduled exposures. The logs span a period of 161 days.

Fig. 2. Comparison of the label value distribution in the Lomb and Duolingo datasets.



The MTR pipeline was implemented using Python 3.8.5 using the popular scientific computing libraries `numpy`, `pandas` and `sklearn`. Application of the data annotation procedure produced a dataset with $\sim$ 8.9K rows with a balanced label distribution, albeit heavily biased towards the edges (Fig. 2). Fitting[4] was performed using 5-fold 2-pass CV, and the confidence intervals for different evaluation metrics are presented in Table III. The following estimators were fit:

- Linear and logistic regression as baselines (from the `sklearn` library).

[4]The values of $\tilde{y}$ were clipped to $[0.1, 0.9]$ for the MTR regressors and $[0.001, 0.999]$ for HLR as this improved their performance.

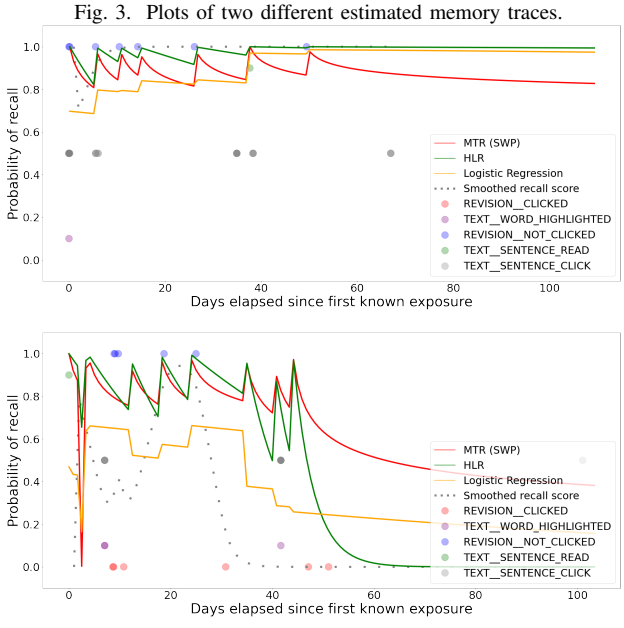
TABLE III
Fitting performance of different memory trace regressors on the Lomb Dataset

Regressor	MSE	WMSE(τ)	WMSE($\tau \odot \Delta$)
Linear Regression	**0.114**	0.0182 $\pm$ 0.0030	0.000978 $\pm$ 0.00074
Logistic Regression	0.116	0.0249 $\pm$ 0.0043	0.00138 $\pm$ 0.0011
HLR	0.171	0.0139 $\pm$ 0.0031	0.000560 $\pm$ 0.00034
HLR ($X \frown \Delta$)	0.171	0.0133 $\pm$ 0.0031	0.000421 $\pm$ 0.00020
MTR-Exp	0.246	0.0134 $\pm$ 0.0030	0.000397 $\pm$ 0.00020
MTR-ESq	0.198	0.0114 $\pm$ 0.0018	0.000351 $\pm$ 0.00015
MTR-Hyp	0.247	0.0140 $\pm$ 0.0028	0.000365 $\pm$ 0.00017
MTR-HSq	0.197	0.0114 $\pm$ 0.0018	0.000333 $\pm$ 0.00014
MTR-SWP	0.176	**0.0109** $\pm$ **0.0021**	**0.000332** $\pm$ **0.00013**

- The simplified forms in Table I, paired with Linear Regression as base regressor. Other base regressors were fitted, but their performance was similar or worse than LR, so their results have been omitted.
- The HLR functional form $2^{\frac{\Delta}{2H}}$, which we will denote as HLR, albeit foregoing SGD in favour of Linear Regression as above.

Lastly, the plots in Fig. 3 allow visualization of the estimated traces produced by Linear Regression, HLR and MTR-SWP (the best performing estimator according to the WMSE metrics).

Fig. 3. Plots of two different estimated memory traces.



The results show that:

- While Linear Regression and Logistic Regression performed well when evaluated using the unweighted MSE, their error was 1.8 and 2.5 times higher than the best regressor in the WMSE(τ) metric, and 3 and 4 times higher when evaluated using the WMSE($\tau \odot \Delta$) metric. This confirms the superior trend reversal and long-term predictive properties of the MTR regressors.
- The performance of HLR was generally better than the linear baselines but significantly poorer than that

of the MTR regressors, particularly in predicting long-term trend-reversals. This suggests that it is not a close approximation to the population R.

- Among the MTR models, the best fit is achieved by MTR-HSq and MTR-SWP with relatively minor performance differences, whereas Exp performs second-worst (after linear models). This situation mirrors the conclusion reached in the large meta-study discussed earlier [3], and hence we can conclude our models significantly correlate with the population R, even though they are inferred from unlabeled data.

The plots in Fig. 3 allow us a glimpse into how the different memory traces explain the events from the logs. The poor trend-reversal predictive ability of Logistic Regression can be intuitively understood because it is a flat, non-decaying line which simply follows the trend. HLR is limited by its exponential decay form: it either decays violently fast or does not decay at all, which suggests that this form is unsuitable for estimating memory traces. Only the MTR-SWP regressor approximates a realistic memory trace like the ones observed in the scientific literature.

Related Work

Previous work has considered the problem as a standard supervised classification task; examples include a model-free reinforcement learning approach [19], and the online scheduling algorithm MEMORIZE [20].

Conclusion & Future Work

The current work highlights some of the challenges involved in estimating memory traces from user interaction data and presents a weakly supervised regression pipeline for their estimation. We argue that posing the problem as a naive classification task ignores the confounding variables affecting the labels, the fact that continuous values for the ground truth are not known, or the inadequacy of general-purpose evaluation metrics. Our method addresses these issues and can reliably estimate memory traces from user interaction data.

Since this is a first effort, there is room for improvement in the data annotation and evaluation phases, or by attempting to fit models with more degrees of freedom.

References

[1] S. Krashen, "Does duolingo 'trump' university-level language learning," *International Journal of Foreign Language Teaching*, vol. 9, no. 1, pp. 13–15, 2014.

[2] B. Settles and B. Meeder, "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)*, 2016, pp. 1848–1858.

[3] D. C. Rubin and A. E. Wenzel, "One hundred years of forgetting: A quantitative description of retention." *Psychological review*, vol. 103, no. 4, p. 734, 1996.

[4] W. A. Wickelgren, "Single-trace fragility theory of memory dynamics," *Memory & Cognition*, vol. 2, no. 4, pp. 775–780, 1974.

[5] J. T. Wixted, S. K. Carpenter, and others, "The wickelgren power law and the ebbinghaus savings function," *Psychological Science*, vol. 18, no. 2, p. 133, 2007.

[6] A. E. Seibert Hanson and C. M. Brown, "Enhancing l2 learning through a mobile assisted spaced-repetition tool: An effective but bitter pill?" *Computer Assisted Language Learning*, vol. 33, nos. 1-2, pp. 133–155, 2020.

[7] S. D. M. Austin, "A study in logical memory," *The American Journal of Psychology*, pp. 370–403, 1921.

[8] B. J. Underwood, "Ten years of massed practice on distributed practice." *Psychological Review*, vol. 68, no. 4, p. 229, 1961.

[9] T. Nakata, "Effects of expanding and equal spacing on second language vocabulary learning: Does gradually increasing spacing increase vocabulary learning?" *Studies in Second Language Acquisition*, vol. 37, no. 4, pp. 677–711, 2015.

[10] F. N. Dempster, "Spacing effects and their implications for theory and practice," *Educational Psychology Review*, vol. 1, no. 4, pp. 309–330, 1989.

[11] H. L. Roediger III and J. D. Karpicke, "The power of testing memory: Basic research and implications for educational practice," *Perspectives on psychological science*, vol. 1, no. 3, pp. 181–210, 2006.

[12] F. Saleh, M. S. Aliakbarian, M. Salzmann, L. Petersson, S. Gould, and J. M. Alvarez, "Built-in foreground/background prior for weakly-supervised semantic segmentation," in *European conference on computer vision*, 2016, pp. 413–432.

[13] G. Szarvas, "Hedge classification in biomedical texts with a weakly supervised selection of keywords," in *Proceedings of acl-08: HLT*, 2008, pp. 281–289.

[14] P. T. Komiske, E. M. Metodiev, B. Nachman, and M. D. Schwartz, "Learning to classify from impure samples with high-dimensional data," *Physical Review D*, vol. 98, no. 1, p. 011502, 2018.

[15] S. Mehrabi *et al.*, "DEEPEN: A negation detection system for clinical text incorporating dependency relation into negex," *Journal of biomedical informatics*, vol. 54, pp. 213–219, 2015.

[16] Y. He, "Incorporating sentiment prior knowledge for weakly supervised sentiment analysis," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 11, no. 2, pp. 1–19, 2012.

[17] X. Ding and E. C. Larson, "Why deep knowledge tracing has less depth than anticipated." *International Educational Data Mining Society*, 2019.

[18] W. Li, "Random texts exhibit zipf's-law-like word frequency distribution," *IEEE Transactions on information theory*, vol. 38, no. 6, pp. 1842–1845, 1992.

[19] S. Reddy, S. Levine, and A. Dragan, "Accelerating human learning with deep reinforcement learning," 2017.

[20] B. Tabibian, U. Upadhyay, A. De, A. Zarezade, B. Schölkopf, and M. Gomez-Rodriguez, "Enhancing human learning via spaced repetition optimization," *Proceedings of the National Academy of Sciences*, vol. 116, no. 10, pp. 3988–3993, 2019.