LEARNING TO SUGGEST BREAKS: SUSTAINABLE OPTIMIZATION OF LONG-TERM USER ENGAGEMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Optimizing user engagement is a key goal for modern recommendation systems, but blindly pushing users towards consumption entails risks. To promote digital well-being, most platforms now offer a service that periodically prompts users to take breaks. These, however, must be set up manually, and so may be suboptimal for both users and the system. In this paper, we study the role of breaks in recommendation, and propose a framework for learning optimal breaking policies that promote and sustain long-term engagement. Based on the notion that user-system dynamics incorporate both positive *and* negative feedback, we cast recommendation as Lotka-Volterra dynamics.We give an efficient learning algorithm, provide theoretical guarantees, and evaluate our approach on semi-synthetic data.

1 INTRODUCTION

Recommendation systems are built with the primary goal of maximizing user engagement; this is typically achieved by recommending on the basis of learned predictive models, trained to predict for each user the potential relevance of new items. Ideally, improved predictive models should lead to increased engagement due to better and more useful recommendations. Nevertheless, the growing engagement of social media platforms has raised concerns about their tendency to drive people towards excessive consumption (Elhai et al., 2017; Lee et al., 2014), resulting in negative mental effects and ultimately leading to a decline in engagement.

To preserve user well-being, it has become common for major platforms to periodically suggest taking breaks (Constine, 2018; Perez, 2018). The idea behind breaks is that occasional disruptions curb the inertial urge for perpetual consumption, and can therefore aid in reducing 'mindless scrolling' (Rauch, 2018), or even addiction (Ding et al., 2016). As a general means for promoting well-being, breaking is psychologically well-grounded (e.g., Danziger et al., 2011; Sievertsen et al., 2016).

But from a system designer's perspective, this observation is puzzling: given the extensive efforts invested in maximizing engagement, why should the system intentionally suggest the opposite? In this work, we provide grounding for the role of suggested breaks in recommendation by modeling user-system interactions as a *complex system*. While recent theoretical models of interaction have been successful in explaining the emergence of societal phenomena such as filter bubbles and polarization of opinion (Jiang et al., 2019; Kalimeris et al., 2021), these models fall short of explaining the effects of suggested breaks due to their unboundedness.

We model user-system interaction as a discrete-time process, in which the user selects the interaction times, and the system decides whether to show content or suggest a break. To address the limitations of existing models, we propose a simple and minimal extension to modeling approach mentioned above. In the continuous limit, the user engagement model approaches a continuous-time Lotka-Volterra (LV) predator-prey system with control, allowing us to analyze the system using the rich tool-set developed in the field of population dynamics (Hofbauer et al., 1998).

In the continuous-time model, long-term engagement is associated with global equilibria of the Lotka-Volterra system, and optimization of break suggestion schedules turns into a problem of optimal control. We show how to efficiently solve for the optimal equilibrium; this provides us with a useful breaking schedule that can be utilized in the original problem space. Interestingly, optimal policies predicted by the LV model exhibit a second-order phase change with respect to system parameters, providing additional interpretation for the observed heterogeneity across users.

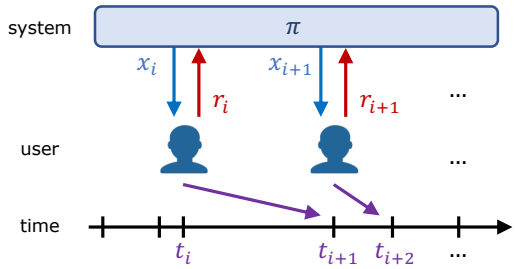

Figure 1: Interaction between the system and a user u over time. Once u decides to query the system at time $t_i \in \mathbb{R}$, she is recommended an item x_i by the policy π. In response, she provides rating feedback r_i, and decides when to interact next (t_{i+1}).

Building upon the insights gained in the analysis, we design an efficient learning algorithm which optimizes breaking policies using past experimental data. Our method uses by reducing policy optimization into a small number of supervised prediction tasks, and carefully aggregates them to fit the behavioral model parameters. We prove agnostic generalization bounds, showing that the performance of our method approaches the best policy in class, and remains stable under bounded model mis-specification, prediction errors, and residual noise. We empirically evaluate the model using semi-synthetic simulation studies, and show improvement over policy optimization baselines.

2 Problem Setting

We consider a sequential recommendation setting in which users interact with a stream of recommended items over time. New users $u \in \mathcal{U}$ are sampled iid from some unknown distribution D, and begin interacting with the system. In each interaction, the system recommends an item x from the set of available items $\mathcal{X}$, and users respond by reporting as feedback their rating $r \in \mathbb{R}$ for x. We assume recommendations are governed by an existing and fixed *recommendation policy* π_0, which we refer to as the 'base' policy.

Engagement. The overall goal of the system is to maximize engagement, which we define as the number of interactions until some chosen time horizon T. Setting $t = 0$ as the (relative) time of joining for each user, the *interaction sequence* of user u under a recommendation policy π is:

$$S_u = \{(t_i, x_i, r_i) \mid t_i \leq T\} \sim \mathcal{S}(\pi; u) \tag{1}$$

where $t_i \in \mathbb{R}_+$ is the time of the i-th event, x_i is the recommended item, and r_i is the reported rating. $\mathcal{S}(\pi; u)$ is some unknown distribution over sequences, which permits dependence between tuples (t_i, x_i, r_i) over time.[1] Defining $\frac{1}{T}|S_u|$ as the *engagement rate* of u, the system seeks to maximize:

$$\text{LTER}(\pi) = \mathbb{E}_{u \sim D}\, \mathbb{E}_{S_u \sim \mathcal{S}(\pi;u)} \left[\tfrac{1}{T}|S_u|\right] \tag{2}$$

which we refer to as *expected long-term engagement rate*.

Breaking policies. We are interested in understanding how breaks affect engagement when applied on top of an existing recommendation policy. Formally, we consider compound policies $\pi = \psi \circ \pi_0$, where π_0 is the (fixed) base policy, and $\psi \mapsto \{0, 1\}$ is a learned *breaking policy* that can either override the base policy by prompting the user to break ($\psi = 1$), or pass on the recommended item $x \sim \pi_0$ ($\psi = 0$). Thus, our learning objective is:

$$\text{argmax}_{\psi \in \Psi}\ \text{LTER}(\psi \circ \pi_0) \tag{3}$$

where Ψ is a class of breaking policies. For simplicity, we assume that π_0 does not include breaks.

[1] Formally, $\mathcal{S}$ is a temporal point process (TPP) with markings.

3 ENGAGEMENT DYNAMICS

To discuss engagement optimization, S_u must be precisely defined. Broadly, we think of S_u as constructed sequentially by the user, where at each time t_i, the user decides on her next time of interaction t_{i+1} based on her experience with the recommended $x_i \sim \pi$. In what follows, we discuss dynamics focusing on a single user u, and hence for clarity drop notational dependence. We return to discussing multiple users in appendix C. Our first step to defining S_u considers rates: Maximizing the number of events $|S_u|$ is analogous to minimizing gaps between consecutive user queries, $\Delta t_i = t_{i+1} - t_i$; this, in turn, is akin to maximizing *instantaneous rates*:[2]

$$\lambda_i = \Delta t_i^{-1} \tag{4}$$

Our next step is to associate λ_i with recommendations. As a basis, the qualitative characteristics of existing models (Jiang et al., 2019; Kalimeris et al., 2021) are captured by a *momentum model*, which considers the λ_i as latent user states, and allows λ_i to depend on the previous λ_{i-1} as:

$$\lambda_i = \lambda_{i-1}\left(1 - \alpha + \beta_i\right) \tag{5}$$

where $\alpha \in [0, 1 + \beta_i]$ is a constant, and $\beta(\cdot)$ is some latent mapping from ratings r_i, interpreted here as the utility for u from consuming x_i, to temporal behavior. $\beta(\cdot)$ is monotonically increasing in r_i, so that more relevant content triggers more frequent visits to the platform. Eq. (5) asserts that engagement rate λ_i increases if the utility β_i is larger than some natural decay parameter α, and decreases otherwise; When recommendation quality is low and $\beta_i < \alpha$, engagement rate drops to zero, and users leave the system. But the converse setting—in which recommendations are effective and β_i is always larger than α—implies that engagement *increases indefinitely*, which is unrealistic.

To remedy this gap, we introduce an additional latent variable, $z_i \in [0, 1]$, which we think of as 'interest', and whose role is to stabilize consumption via:[3]

$$\lambda_i = \lambda_{i-1}\left(1 - \alpha + \beta_i z_i\right) \tag{6}$$

We model z_i as also varying with time, and w.r.t. λ_{i-1}, as:

$$z_i = z_{i-1}\left(1 + \gamma(1 - z_{i-1}) - \delta\lambda_{i-1}\right) \tag{7}$$

for some constants $\gamma, \delta > 0$. Note that Eqs. (6, 7) are functionally similar, differing only in the sign of the constants (as per their opposing roles), and in the term $(1-z_{i-1})$ which ensures that z_i remains in $[0, 1]$. We refer to Eq. (6) as *positive feedback* and to Eq. (7) as *negative feedback*. Breaks are expressed in the dynamic model by setting $\beta = 0$ and $\delta = 0$: this causes λ_i to temporarily decrease (due to $-\alpha$), but allows z_i to replenish (thanks to $+\gamma$). To learn effective breaking schedules, our general strategy will be to optimize for 'sustainable habits', which we think of as the limiting behavior of $\frac{1}{T}|S_u|$ as $T \to \infty$.

3.1 THE CONTINUOUS LIMIT

In the continuous limit, the discrete model approaches a continuous-time Lotka-Volterra predator-prey system with control, allowing us to analyze the system using the rich tool-set developed in the field of population dynamics. To map Eqs. (6, 7) to continuous time, we define $\lambda(t)$ and $z(t)$ as the continuous analogs of λ_i and z_i, and define $\beta = \mathbb{E}_\pi[\beta_i]$ as the 'average effect' of recommendations on behavior. Lastly, we account for breaks: Consider some breaking policy ψ, and denote by p the expected breaking rate, namely $p = \mathbb{P}[\psi = 1]$. Under ψ, the expected values of β and δ become $(1 - p)\beta$ and $(1 - p)\delta$, respectively. This gives our final continuous model:[4]

$$\begin{aligned} \frac{\mathrm{d}\lambda}{\mathrm{d}t} &= -\alpha\lambda + \beta z\lambda(1 - p) \\ \frac{\mathrm{d}z}{\mathrm{d}t} &= \gamma z(1 - z) - \delta z\lambda(1 - p) \end{aligned} \tag{8}$$

characterized by the set of parameters $\theta = (\alpha, \beta, \gamma, \delta)$, and p.

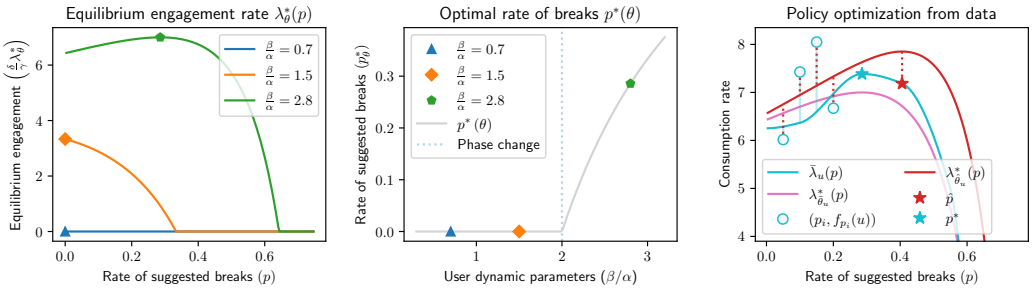

Figure 2: Schematic illustration of results: **(Left)** Equilibrium curves $\lambda^*(p)$ and optimal policies p^* (*markers*) for user types (θ_u) that: benefit from breaks (*green*), do not require breaks (*orange*), and will inevitably churn (*blue*). Curves are given by lemma B.1. **(Center)** The optimal policy p^* for all β/α, as given by lemma B.2. The optimal policy exhibits a second-order phase change at $\beta/\alpha = 2$ (see corollary B.3). **(Right)** Schematic diagram of the learning method described in appendix C. An illustration of the true counterfactual engagement curve (*cyan*), an ideal LV fit (*pink*), and an empirical LV fit (*red*) from observations (*circles*), showing similar optima (*stars*).

4 Summary of Results

Optimal equilibria. Associating each user with a personal set of LV parameters $\theta_u = (\alpha_u, \beta_u, \gamma_u, \delta_u)$, we begin by analyzing the equilibrium of eq. (8), deriving a closed-form formula for the equilibrium as a function of θ_u and p. Optimizing for p yields an optimal breaking policy in the continuous space, and which is then translated to the original problem space (see Lemma B.2, Appendix B.3). The second-order phase change in p^* can be interpreted as dividing the latent user space to users who benefit from breaks, and users that don't (see Corollary B.3 and Figure 2).

Learning to suggest breaks. Given these insights, our second goal is to introduce and study the novel task of *learning to suggest a break*. Current breaking solutions are entirely heuristic, and effectively provide no guarantees. As an alternative, we propose an algorithm for learning optimal breaking policies from data: for any recommendation policy, our algorithm finds a breaking schedule that optimizes long-term engagement by proactively overriding the base policy when this is deemed necessary. Our approach works by embedding users in 'LV-space'—the set of all possible LV trajectories, which effectively serves us as a parameterized hypothesis class (See Appendix C). Our learning algorithm enjoys the following useful property: given *predictions* of user engagement, the policy optimization problem decomposes over users, and can be solved independently for each one.

Theoretical guarantees. Our main theoretical result (Theorem D.1) is an agnostic bound on the expected long-term engagement of our learned breaking policy, relative to the optimal policy in the class. We show that for any recommendation policy, the gap decomposes into three distinct additive terms: (i) predictive error, (ii) modeling error (i.e., embedding distortion), and (iii) variance around the (theoretical) steady state. These provide an intuitive interpretation of the bound, as well as means to understand the effects of different modeling choices on outcomes. The proof relies on carefully weaving LV equilibrium analysis within conventional concentration bounds for learning.

Simulation study. Finally, we provide an empirical evaluation of our approach on semi-synthetic data (Appendix E). Using two real datasets, we generate simulated user interaction sequences in a way that captures the essence of our model, but is different from the actual continuous-time dynamics we optimize over. Results show that despite this gap, our approach improves significantly over myopic baselines, and often closely matches an optimal oracle. Taken together, these results demonstrate the potential utility of our approach.

[2] As $\frac{1}{T}|S_u|$ is a rate of events with instantaneous frequency λ_i, maximizing the empirical rate is asymptotically equivalent to maximizing the *harmonic mean* of λ_i.

[3] Similar notions have been considered in Leqi et al. (2021) who model satiation, and Kleinberg & Immorlica (2018) who model fatigue; these, too, model variation in affinity, rather than time.

[4] Eqs. (6, 7) can be obtained from Eq. (10) via the Euler method.

REFERENCES

Roy F Baumeister, Ellen Bratslavsky, Mark Muraven, and Dianne M Tice. Ego depletion: is the active self a limited resource. *Journal of personality and social psychology.*, 74(5), 1998.

Bruce O Bergum and Donald J Lehr. Vigilance performance as a function of interpolated rest. *Journal of Applied Psychology*, 46(6):425, 1962.

Junyu Cao, Wei Sun, Zuo-Jun (Max) Shen, and Markus Ettl. Fatigue-aware bandits for dependent click models. In *AAAI*, pp. 3341–3348, 2020.

Allison JB Chaney. Recommendation system simulations: A discussion of two key challenges. *arXiv preprint arXiv:2109.02475*, 2021.

Allison JB Chaney, Brandon M Stewart, and Barbara E Engelhardt. How algorithmic confounding in recommendation systems increases homogeneity and decreases utility. In *Proceedings of the 12th ACM conference on recommender systems*, pp. 224–232, 2018.

Donghui Chen and Robert J Plemmons. Nonnegativity constraints in numerical analysis. In *The birth of numerical analysis*, pp. 109–139. World Scientific, 2010.

Josh Constine. Instagram says 'you're all caught up' in first time-well-spent feature. techcrunch, 2018. URL https://techcrunch.com/2018/05/21/scroll-responsibly/. [Online; accessed 23-September-2022].

Shai Danziger, Jonathan Levav, and Liora Avnaim-Pesso. Extraneous factors in judicial decisions. *Proceedings of the National Academy of Sciences*, 108(17):6889–6892, 2011. doi: 10.1073/pnas.1018033108.

Sarah Dean and Jamie Morgenstern. Preference dynamics under personalized recommendations. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, EC '22, pp. 795–816, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450391504. doi: 10.1145/3490486.3538346.

Xiang Ding, Jing Xu, Guanling Chen, and Chenren Xu. Beyond smartphone overuse: identifying addictive mobile apps. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 2821–2828, 2016.

Jacob P Duncan, Teresa Aubele-Futch, and Monica McGrath. A fast-slow dynamical system model of addiction: Predicting relapse frequency. *SIAM Journal on Applied Dynamical Systems*, 18(2): 881–903, 2019.

Jon D Elhai, Robert D Dvorak, Jason C Levine, and Brian J Hall. Problematic smartphone use: A conceptual overview and systematic review of relations with anxiety and depression psychopathology. *Journal of affective disorders*, 207:251–259, 2017.

Nir Eyal. *Hooked : how to build habit-forming products / Nir Eyal with Ryan Hoover.* Portfolio/Penguin, New York, updated edition edition, 2019. ISBN 9781591847786.

J Doyne Farmer. Market force, ecology and evolution. *Industrial and Corporate Change*, 11(5): 895–953, 2002.

Simona Gilboa, Arie Shirom, Yitzhak Fried, and Cary Cooper. A meta-analysis of work demand stressors and job performance: examining main and moderating effects. *Personnel psychology*, 61(2):227–271, 2008.

Nico S Gorbach, Stefan Bauer, and Joachim M Buhmann. Scalable variational inference for dynamical systems. *Advances in neural information processing systems*, 30, 2017.

Wenshuo Guo, Karl Krauth, Michael Jordan, and Nikhil Garg. The stereotyping problem in collaboratively filtered recommender systems. In *Equity and Access in Algorithms, Mechanisms, and Optimization*, pp. 1–10. 2021.

F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015.

William S Helton and Paul N Russell. Rest is still best: The role of the qualitative and quantitative load of interruptions on vigilance. *Human factors*, 59(1):91–100, 2017.

Robert A Hennfng, Steven L Sauter, Gavriel Salvendy, and Edward F Krieg Jr. Microbreak length, performance, and stress in a data entry task. *Ergonomics*, 32(7):855–864, 1989.

Josef Hofbauer, Karl Sigmund, et al. *Evolutionary games and population dynamics*. Cambridge university press, 1998.

Jiri Hron, Karl Krauth, Michael I Jordan, Niki Kilbertus, and Sarah Dean. Modeling content creator incentives on algorithm-curated platforms. *arXiv preprint arXiv:2206.13102*, 2022.

Nicolas Hug. Surprise: A python library for recommender systems. *Journal of Open Source Software*, 5(52):2174, 2020.

Eugene Ie, Chih-wei Hsu, Martin Mladenov, Vihan Jain, Sanmit Narvekar, Jing Wang, Rui Wu, and Craig Boutilier. Recsim: A configurable simulation platform for recommender systems. *arXiv preprint arXiv:1909.04847*, 2019.

Ray Jiang, Silvia Chiappa, Tor Lattimore, András György, and Pushmeet Kohli. Degenerate feedback loops in recommender systems. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 383–390, 2019.

Daniel Kahneman. *Attention and effort*, volume 1063. Citeseer, 1973.

Dimitris Kalimeris, Smriti Bhagat, Shankar Kalyanaraman, and Udi Weinsberg. Preference amplification in recommender systems. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 805–815, 2021.

Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. The challenge of understanding what users want: Inconsistent preferences and engagement optimization. *arXiv preprint arXiv:2202.11776*, 2022.

Robert Kleinberg and Nicole Immorlica. Recharging bandits. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 309–319. IEEE, 2018.

Karl Krauth, Sarah Dean, Alex Zhao, Wenshuo Guo, Mihaela Curmei, Benjamin Recht, and Michael I Jordan. Do offline metrics predict online performance in recommender systems? *arXiv preprint arXiv:2011.07931*, 2020.

Yu-Kang Lee, Chun-Tuan Chang, You Lin, and Zhao-Hong Cheng. The dark side of smartphone usage: Psychological traits, compulsive behavior and technostress. *Computers in human behavior*, 31:373–383, 2014.

Liu Leqi, Fatma Kilinc Karzan, Zachary Lipton, and Alan Montgomery. Rebounding bandits for modeling satiation effects. *Advances in Neural Information Processing Systems*, 34:4003–4014, 2021.

Masoud Mansoury, Himan Abdollahpouri, Mykola Pechenizkiy, Bamshad Mobasher, and Robin Burke. Feedback loop and bias amplification in recommender systems. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pp. 2145–2148, 2020.

O'Dhaniel A Mullette-Gillman, Ruth LF Leong, and Yoanna A Kurnianingsih. Cognitive fatigue destabilizes economic decision making preferences and strategies. *PloS one*, 10(7):e0132022, 2015.

Mark Muraven and Roy F Baumeister. Self-regulation and depletion of limited resources: Does self-control resemble a muscle? *Psychological bulletin*, 126(2):247, 2000.

Sarah Perez. Apple unveils a new set of 'digital wellness' features for better managing screen time. *TechCrunch, June*, 4, 2018. URL https://techcrunch.com/2018/06/04/apple-unveils-a-new-set-of-digital-wellness-features-for-better-managing-screen-time/. [Online; accessed 23-September-2022].

Jennifer Rauch. *Slow media: Why slow is satisfying, sustainable, and smart*. Oxford University Press, 2018.

Stephen E Robertson. The probability ranking principle in ir. *Journal of documentation*, 1977.

Hayden A Ross, Paul N Russell, and William S Helton. Effects of breaks and goal switches on the vigilance decrement. *Experimental brain research*, 232(6):1729–1737, 2014.

Tom Ryder, Andrew Golightly, A Stephen McGough, and Dennis Prangle. Black-box variational inference for stochastic differential equations. In *International Conference on Machine Learning*, pp. 4423–4432. PMLR, 2018.

Larry Samuelson. *Evolutionary games and equilibrium selection*, volume 1. MIT press, 1998.

Francesco Sanna Passino, Lucas Maystre, Dmitrii Moor, Ashton Anderson, and Mounia Lalmas. Where to next? a dynamic model of user preferences. In *Proceedings of the Web Conference 2021*, WWW '21, pp. 3210–3220, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450383127. doi: 10.1145/3442381.3450028.

Sven Schmit and Carlos Riquelme. Human interaction with recommendation systems. In *International Conference on Artificial Intelligence and Statistics*, pp. 862–870. PMLR, 2018.

Maarten P. Scholl, Anisoara Calinescu, and J. Doyne Farmer. How market ecology explains market malfunction. *Proceedings of the National Academy of Sciences*, 118(26):e2015574118, 2021. doi: 10.1073/pnas.2015574118.

Hans Henrik Sievertsen, Francesca Gino, and Marco Piovesan. Cognitive fatigue influences students' performance on standardized tests. *Proceedings of the National Academy of Sciences*, 113 (10):2621–2624, 2016.

Yasuhiro Takeuchi. *Global dynamical properties of Lotka-Volterra systems*. World Scientific, 1996.

Emmanuel Trélat and Enrique Zuazua. The turnpike property in finite-dimensional nonlinear optimal control. *Journal of Differential Equations*, 258(1):81–114, 2015.

Mengting Wan and Julian McAuley. Item recommendation on monotonic behavior chains. In *Proceedings of the 12th ACM conference on recommender systems*, pp. 86–94, 2018.

Mengting Wan, Rishabh Misra, Ndapa Nakashole, and Julian McAuley. Fine-grained spoiler detection from large-scale review corpora. *arXiv preprint arXiv:1905.13416*, 2019.

Jyun-Cheng Wang and Juo-Ping Lin. Are personalization systems really personal?-effects of conformity in reducing information. In *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the*, pp. 10–pp. IEEE, 2003.

Romain Warlop, Alessandro Lazaric, and Jérémie Mary. Fighting boredom in recommender systems with linear reinforcement learning. *Advances in Neural Information Processing Systems*, 31, 2018.

Jörgen W Weibull. *Evolutionary game theory*. MIT press, 1997.

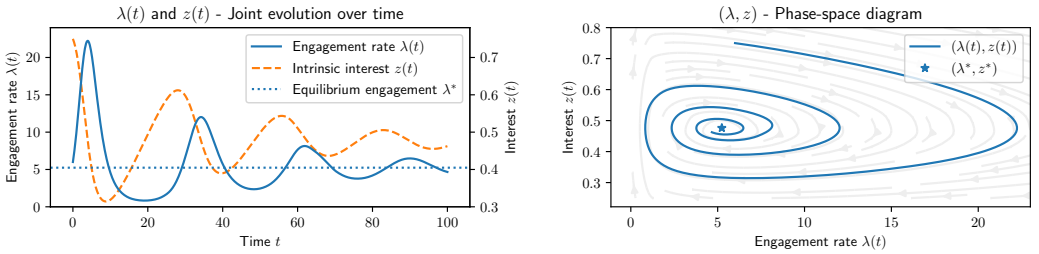

Figure 3: Temporal relations between *rate* $\lambda(t)$, *interest* $z(t)$, and equilibrium λ^* in the continuous limit (Eq. (10)). Note how $\lambda(t)$ drops only some time after z has depleted.

A Related work

Recommendation ecosystems. Our work pertains to a growing literature that studies recommendation systems as complex systems in which learning plays a distinctive role. Some works aim to connect micro-level actions to emerging macro-level phenomena, such as homogenization via confounding (Chaney et al., 2018), heterogenization via social learning (Schmit & Riquelme, 2018), diversity via strategic behavior (Hron et al., 2022), feedback amplification (Mansoury et al., 2020), accessibility and stereotyping (Guo et al., 2021), and the relation between online and offline metrics (Krauth et al., 2020). To the best of our knowledge, our work is novel in considering breaks. Due to the counterfactual nature of recommendation, most works in this field provide either theoretical analysis, or simulation studies (Ie et al., 2019; Chaney, 2021). We follow suit, and aim for both.

User dynamics and feedback. Several recent works aim to capture time-varying user behavior by modeling users as acting based on dynamic latent states. These differ from ours in two important ways. First, they model changes in the *affinity* of users towards items, and over discrete time steps; this makes them inappropriate for expressing *temporal* variations in user behavior, which is needed for discussing engagement. Second, most works consider either positive-only feedback (Jiang et al., 2019; Kalimeris et al., 2021; Sanna Passino et al., 2021; Dean & Morgenstern, 2022), or negative-only feedback (Wang & Lin, 2003; Warlop et al., 2018; Kleinberg & Immorlica, 2018; Cao et al., 2020; Leqi et al., 2021); we simultaneously consider both types of feedback, which we believe is more realistic—as well as necessary for explaining breaks. Closest to ours is Kleinberg et al. (2022), who also model bi-directional feedback, but in a very different setup (a stylized model of short-term user churn in discrete time) and towards different aims (characterizing equilibira, rather than learning).

Lotka-Volterra dynamics. The study of ecosystem dynamics and their conservation has a long and rich history, in which LV analysis is integral (see Hofbauer et al. (1998); Takeuchi (1996)). LV systems are used primarily for modeling biological ecosystems, but are also used in economics (Weibull, 1997; Samuelson, 1998), finance (Farmer, 2002; Scholl et al., 2021), and behavioral modeling (e.g., drug addiction and relapse (Duncan et al., 2019)). We believe our work is novel in its use of LV modeling in recommendation. In terms of learning, Gorbach et al. (2017) and Ryder et al. (2018) propose variational techniques for dynamical systems, but do not consider control. Our work aims to directly learn optimal policies, for which we draw on recent advances in turnpike optimal control (Trélat & Zuazua, 2015).

B Engagement in continuous time

One challenge in optimizing Eq. (2) is that empirical rates $\frac{1}{T}|S_u|$ exhibit variation that may be difficult to account for using observed data. As an alternative, we will aim for optimizing individualized *limiting rates*, defined as:

$$\lambda_u^* = \lim_{T' \to \infty} \frac{1}{T'}|S_u| \tag{9}$$

This abstracts away 'everyday' variation in behavior, and focuses instead on habits—which are easier to anticipate. When empirical rates are 'well behaved' in the sense that they concentrate around the limiting behavior, then we can expect λ_u^* to be a good proxy for engagement. In appendix D we make this notion precise. To understand the possible effects of breaks on limiting

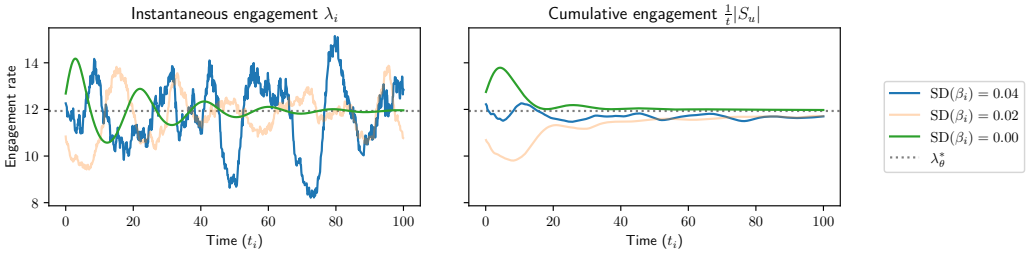

Figure 4: Discrete engagement dynamics for varying levels of dispersion in the distribution of β_i. **(Left)** When the variance of β_i is low, the instantaneous engagement λ_i approaches smooth LV dynamics. **(Right)** Cumulative engagement rates $\frac{1}{T}|S_u|$ converge towards the equilibrium λ^*_θ even under strong stochasticity.

behavior, we will analyze a continuous-time analog of our dynamic model in Sec. 3. This will allow us to employ powerful tools from dynamical systems and control theory, and establish preliminary results that will form the basis of our learning approach.

B.1 Continuous engagement dynamics

Mapping Eqs. (6, 7) to continuous time requires three steps. First, we define $\lambda(t)$ and $z(t)$ as the continuous analogs of λ_i and z_i. Next, we define $\beta = \mathbb{E}_\pi[\beta_i]$ as the 'average effect' of recommendations on behavior. Lastly, we account for breaks: Consider some breaking policy ψ, and denote by p the expected breaking rate, namely $p = \mathbb{P}[\psi = 1]$. Under ψ, the expected values of β and δ become $(1-p)\beta$ and $(1-p)\delta$, respectively. This gives our final continuous model:[5]

$$\begin{aligned} \frac{\mathrm{d}\lambda}{\mathrm{d}t} &= -\alpha\lambda + \beta z\lambda(1-p) \\ \frac{\mathrm{d}z}{\mathrm{d}t} &= \gamma z(1-z) - \delta z\lambda(1-p) \end{aligned} \tag{10}$$

Eq. (10) describes a system of Lokta-Volterra (LV) differential equations, characterized by the set of parameters $\theta = (\alpha, \beta, \gamma, \delta)$, and p. LV systems, also known as *predator-prey dynamics*, have been popularized by and studied extensively in the fields of theoretical ecology and population dynamics (Hofbauer et al., 1998; Takeuchi, 1996). We now proceed to overview some useful properties of LV systems.

Cycling behavior. Eq. (10) describes user behavior as a *cycle*: when interest $z(t)$ is high, engagement rate $\lambda(t)$ increases, resulting in positive feedback; conversely, when $\lambda(t)$ is high, $z(t)$ decreases, which expresses negative feedback. In general, λ grows until interest is too low to sustain consumption, at which point consumption drops sharply, allowing interest to recover—and the cycle repeats. The cycling behavior exhibits oscillations in λ and q, with one lagging after the other. A typical trajectory is illustrated in fig. 3. Note how the drop in λ occurs only some time after z is depleted; hence, anticipating (and preventing) the collapse of λ requires conservation of z. Thus, z serves as a resource: necessary for engagement, and of limited supply.

Equilibrium. Over time, and if no interventions are applied, the magnitude of oscillations decreases, and the system naturally approaches a *stable equilibrium*, denoted $(\lambda^*_\theta, z^*_\theta)$, determined by system parameters θ. Our first result shows that λ^*_θ has a convenient closed form.

Lemma B.1. *Let $\theta = (\alpha, \beta, \gamma, \delta)$ define a controlled LV system as in Eq. (10). Then for any $p \in [0, 1]$, we have:*

$$\lambda^*_\theta(p) = \max\left\{\frac{\gamma}{\delta}\frac{1}{1-p}\left(1 - \frac{\alpha}{\beta}\frac{1}{1-p}\right), 0\right\} \tag{11}$$

Proof in appendix H.1, and illustration in fig. 2 (Left). Eq. (11) is useful as it depicts λ^* as a simple function of p, parameterized by θ. This will prove useful for optimization.

[5] Eqs. (6, 7) can be obtained from Eq. (10) via the Euler method.

B.2 BREAKING AS OPTIMAL CONTROL

Eq. (10) suggests a natural approach for optimizing breaks: given θ, find p that maximizes the limiting rate λ^*. Essentially, this casts p as a *control variable*, and learning to break becomes a problem of optimal control. Note how p mediates the relations between $\lambda(t)$ and $z(t)$: when $p > 0$, it decelerates engagement rate λ, and at the same time, lets z recover.

Our goal is now to solve the optimal control problem:

$$p^*(\theta) = \text{argmax}_{p\in[0,1]}\, \lambda^*_\theta(p) \tag{12}$$

Towards, this, our next result derives a closed form solution for the optimal p^*. Note that Eq. (11) shows $\lambda^*_\theta(p)$ is piece-wise polynomial in $q = 1/(1-p)$. Solving for q, we get:

Lemma B.2. *Let $\theta = (\alpha, \beta, \gamma, \delta)$ define an LV system as in Eq. (10). Then the optimal p^* is given by:*

$$p^*(\theta) = \max\left\{1 - 2\frac{\alpha}{\beta}, 0\right\} \tag{13}$$

Proof in appendix H.1. See illustration in Fig. 2 (Center).

B.3 LEARNING TO BREAK, REVISITED

In continuous space, breaking manifests as a control variable $p \in [0, 1]$, continuous and fixed in time. To apply this idea back in our original discrete problem setting, we can interpret p as determining the probability to break on any given input. This defines a class of stationary breaking policies:

$$\Psi = \{\psi_u(p) = \text{break w.p. } p \,:\, p \in [0,1], u \in \mathcal{U}\} \tag{14}$$

Using this, our general approach for learning to break will be to: (i) associate with each user u a set of LV parameters θ_u, and then (ii) compute p^*_u which maximizes $\lambda^*_{\theta_u}(p)$, and apply breaks using the individualized policy $\psi_u = \psi(p^*_u)$.

One useful property of optimal LV policies is that they suggest breaks only when this is deemed necessary. Note that by Eq. (13), $p^*(\theta)$ exhibits a *phase transition* at $\frac{\alpha}{\beta}{=}\frac{1}{2}$, below which $p^*{=}0$, and above which $p^*{>}0$. When considering individualized θ_u, we get the following result:

Corollary B.3. *In LV space, users are partitioned by their θ_u to those who benefit from breaks, and those who don't.*

We further explore this idea in our experiments in Sec. E.

C LEARNING OPTIMAL BREAKING POLICIES

We now turn to presenting our learning algorithm. As noted, the algorithm consists of two steps: (i) *embedding*, which fits for each user some $\hat{\theta}_u$ from data, and (ii) *optimization*, which computes an optimal $\hat{p}_u$ from $\hat{\theta}_u$. One benefit of our approach is that it operates entirely on predictions of future engagement. Our procedure is illustrated in fig. 2 (Right).

C.1 EMBEDDING USERS IN LV SPACE

Our first task is to choose a suitable θ_u for u, which we think of as embedding users in 'LV space'. A natural first attempt would be to fit θ_u to λ^*_θ in Eq. (11) from data. However, the crux is that the observed S_u come from the default policy π_0, which does not include breaks, i.e., has $p = 0$. But our ultimate goal is to optimize over *all* p—for this, data that is representative of a single p (e.g., $p = 0$) will likely be biased.

Equilibrium curves. Ideally, what we would like to do is fit θ_u to the entire *equilibrium curve* of $\lambda^*_\theta(p)$. Let $\bar{\lambda}_u(p)$ be the *expected empirical engagement rate*, defined as:

$$\bar{\lambda}_u(p) = \mathbb{E}_{S_u\sim\mathcal{S}(\psi(p);u)}\left[\tfrac{1}{T}|S_u|\right] \tag{15}$$

As a function or p, $\bar{\lambda}(p)$ gives the *true* engagement rate for any choice of p. Using this notation, we seek θ for which $\lambda^*_\theta(p)$ closely aligns with that of $\bar{\lambda}_u(p)$ across all $p \in [0, 1]$:

$$\bar{\theta}_u = \text{argmin}_\theta \, \|\bar{\lambda}_u - \lambda^*_\theta\| \tag{16}$$

for some function norm $\|\cdot\|$, and for which $\lambda^*_{\bar{\theta}_u}$ and $\bar{\lambda}_u$ have similar maximizing p. Unfortunately, $\bar{\lambda}_u$ is a theoretical object, and solving Eq. (16) requires access to all p-s (whereas our original data offers just one). Our solution will be to make due with a small set of p-s, in the form of *predictions*.

The role of prediction. As any policy problem, learning to break requires some form of exploration or experimentation. Here we aim for experimentation to be simple and minimal. Specifically, we will allow the system to collect some additional data: for a small set of N distinct breaking rates $p_j > 0$, we assume the system can allocate some bandwidth to experiment using compound policies $\pi_j = \pi(p_j) = \psi(p_j) \circ \pi_0$, and obtain data $D^{(j)} = \{(u_k, S_{u_k})\}_{k=1}^{m_j}$ for small m_j.

Denoting by $D^{(0)}$ the original data for the base policy $\pi_0 = \pi(0)$, our approach will be use the gathered $D^{(0)}, \ldots, D^{(N)}$ to learn individualized *policy-specific predictors*, $f_j(u) = f_{p_j}(u)$, trained to predict for each user u her engagement rate $y = \frac{1}{T}|S_u|$ under the policy π_j. For example, if we train f_j to minimize the squared error $\sum_k (f_j(u_k) - y_k)^2$ on pairs $(u_k, y_k) \in D^{(j)}$, then $f_j(u)$ should be a reasonable estimator of the expected $\bar{\lambda}_u(p_j)$. Hence, for a given u, a finite set of pairs $\{(p_j, f_j(u))\}_{j=1}^N$ gives points to which we can fit θ to λ^*_θ. Our final criterion for choosing $\hat{\theta}_u$ is:

$$\hat{\theta}_u = \text{argmin}_\theta \sum\nolimits_{j=1}^{N} \left(f_j(u) - \lambda^*_\theta(p_j)\right)^2 \tag{17}$$

where $\boldsymbol{f}(u) = (f_1(u), \ldots, f_N(u)) \in \mathbb{R}^N_{>0}$, and given here with the ℓ_2 vector norm. From lemma B.2, optimizing over p requires only the ratios α/β and γ/δ, which appear as polynomial coefficients. Hence, Eq. (17) can be efficiently solved using a polynomial Non-Negative Least Squares (NNLS) regression solver (Chen & Plemmons, 2010).

The role of experimentation. In the realizable case, Eq. (16) has a zero-norm minimizer, and the goodness of fit for $\hat{\theta}_u$ is controlled by two parameters: the number of experimental datasets, N, and their sizes, m_j for $j \in [N]$. In general, increasing N provides more data points for solving Eq. (17), and increasing each m_j reduces noise for that point (i.e., $f_u(p)$ should be closer to $\bar{\lambda}_u$). However, in reality experimentation is costly, and so N and the m_j may be small. As motivation, we next show that under realizability and for accurate predictions, $N = 1$ suffices. Our result applies to more general base policies $\pi_0 = \pi(p_0)$ using any $p_0 \geq 0$.

Proposition C.1. *Fix $N = 1$, and let $p_0, p_1 \in [0, 1 - \alpha/\beta]$. For a user u, if (i) exists θ_u s.t. $\bar{\lambda}_u = \lambda^*_{\theta_u}$, and (ii) $f_i(u) = \bar{\lambda}_u(p_i)$ for $i = 1, 2$, then solving Eq. (17) recovers the true expected rate, i.e., $\hat{\theta}_u = \bar{\theta}_u$, and is therefore optimal.*

Proof provided in appendix H.1, and relies on lemma B.1. Next, we discuss how to obtain $\psi(u)$ from θ_u.

C.2 From predictions to optimal policies

One useful property of our approach is that it circumvents the need to learn a global policy: Once the $\{f_j(u)\}_j$ have been learned, the policy problem decomposes over users, and optimal individualized policies ψ_u are determined independently for each user. I.e., by relying on predictions, the solution to Eq. (2) is immediately obtained, and at test time we simply use predictions to compute ψ_u for new users u.

Our final procedure is as follows: given some user u, we (i) compute predictions $\boldsymbol{f}(u)$; (ii) find $\hat{\theta}_u$ by solving Eq. (17); (iii) obtain $\hat{p}_u$ by solving Eq. (13); and (iv) apply the policy:

$$\psi_u = \psi(\hat{p}_u), \quad \text{where } \hat{p}_u = p^*(\hat{\theta}_u) \tag{18}$$

Notably, for $N = 1$, $\psi(u)$ has a closed-form formulation as a function of predictions (Appendix H.2). In this case:

Corollary C.2. *In the realizable case of proposition C.1, $\psi(\hat{p}_u)$ idempotently improves over the myopic $\pi(0)$.*

Thus, the optimal policy can be interpreted as suggesting breaks only when it deems them necessary. fig. 2 (Left, Center) illustrates $\lambda^*(p)$ curves and policies for various user types.

D THEORETICAL GUARANTEES

Our main theoretical result bounds the expected long-term engagement obtained by our global learned policy, $\hat{\psi} = \psi(\hat{p})$. Our bound shows that the gap between $\hat{\psi}$ and the optimal policy, ψ^{opt}, is governed by three additive terms, each relating to a different aspect of our approach: modeling error (ε_{LV}), predictive error ($\varepsilon_{\text{pred}}$), and deviation from expected behavior (ε_{dev}). A description and interpretation of each term follows shortly. For simplicity, we focus on $N = 1$.

Theorem D.1 (Informal)**.** *For any π_0, let $p_0, p_1 \in [0, 1]$, and denote by $\psi^{\text{opt}} \in \Psi$ be the optimal stationary policy. Then for the learned breaking policy $\hat{\psi}$, we have:*

$$\text{LTER}(\psi^{\text{opt}}) - \text{LTER}(\hat{\psi}) \leq \frac{\eta_{\text{TPP}}}{|p_1 - p_0|}(\varepsilon_{\text{LV}} + \varepsilon_{\text{pred}} + \varepsilon_{\text{dev}})$$

where η_{TPP} is an $\mathcal{S}$-specific constant scale factor.

Formal statement, precise definitions, and proof are given in Appendix H.3. The proof consists of three main steps: Starting with a clean LV system at $T = \infty$, we quantify the downstream effects of perturbing the optimal policy. Then, we plug in the learned policy, and bound the gap due to predictive errors and finite T. The final step makes the transition from continuous dynamics to our discrete dynamic model.

We now proceed to detail the role of each of the five terms in the bound, and how they may be controlled.

Predictive error: Since targets $y = \frac{1}{T}|S_u|$ are predicted, $\varepsilon_{\text{pred}}$ is simply the expected regression error over users, measured in RMSE. As is standard, $\varepsilon_{\text{pred}}$ can be reduced by increasing the number of samples m, or by learning more expressive predictors f (e.g., larger neural nets).

Modeling error: LV dynamics permit tractable learning; but as any hypothesis class, this trades off with model capacity. Here, ε_{LV} quantifies the error due limited expressive power. Further reducing ε_{LV} can be achieved by considering richer dynamic models—a challenge left for future work.

Deviation from expectation: The learned $\hat{p}_u$ rely on predicted equilibrium, but are trained on finite-horizon data. In expectation, ε_{dev} captures how finite sequences deviate from their mean. As a rule of thumb, we expect larger T to reduce this form of noise, but this cannot be guaranteed.

Sensitivity : For $N = 2$, the term $|p_1 - p_0|$ quantifies the added value of exploring beyond the default breaking policy of p_0. Intuitively, when the points are farther away, fitting the equilibrium curve is easier. Thus, for $p_0 = 0$, p_1 should be chosen to balance between performance gain and overexposure of experimental subjects to breaks.

E EXPERIMENTS

We conclude with an empirical evaluation of our approach on semi-synthetic data. Here we include results for the MovieLens 1M dataset. See appendix I for additional details, appendix I.6 for similar results on the Goodreads dataset, and appendix J for additional experiments.

E.1 EXPERIMENTAL SETUP

Data. The MovieLens 1M dataset (Harper & Konstan, 2015) includes 1,000,209 ratings provided by 6,040 users and for 3,706 items, which we use to obtain features, determine the dynamics, and emulate ϕ. We sample and hold out 30% of all ratings r_{ux} via user-stratified sampling, to which we apply Collaborative Filtering (CF) to get user features u and item features x that approximate $u^\top x \approx r_{ux}$ ($d = 8$, RMSE $= 0.917$, $r \in [1, 5]$). This mimics a process where ϕ is based on items recommended by π and rated by users. We then take the remaining data points and randomly assign 1,000 users to the test set, on which we evaluate policies. The remaining users are assigned to the train sets $\{D^{(j)}\}_j$.

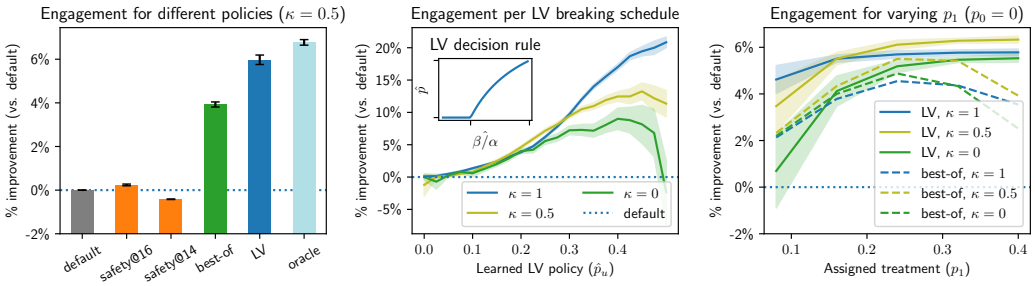

Figure 5: Results on the MovieLens 1M dataset. **(Left)** Performance gain of different approaches (relative to `default` policy). **(Center)** Performance of `LV` by user group, partitioned by learned policies $\hat{p}_u$. **(Right)** Sensitivity to an increasingly aggressive experimental p_1 ($N = 2, p_0 = 0$).

Recommendation policy and user behavior. As defined in Sec. 2, π_0 is set to recommend as $\mathrm{softmax}_x(\hat{r}_u)$, and user behavior as simulated in accordance to the discrete dynamics in Sec. 3. This enables us to evaluate and compare counterfactual outcomes under different policies. Note this entails variation in the β_{ui}, meaning there is no single β_u that underlies the dynamics: even in the limit ($\Delta t \to 0, T \to \infty$), user behavior cannot be described by a continuous LV system, which implies $\varepsilon_{\mathrm{LV}} > 0$ (see Fig. 7). Since the baseline RMSE is high, we set $\beta_{ui} \propto \tilde{r}^2_{u,x_i}$, where $\tilde{r}_{u,x_i} = \kappa r_{u,x_i} + (1-\kappa)u^\top x$, so that κ interpolates between predicted ratings ($\kappa = 0$) and true ratings ($\kappa = 1$). This allows us to (indirectly) control $\varepsilon_{\mathrm{pred}}$. For all experiments we use $T = 100$, and so expect a roughly fixed $\varepsilon_{\mathrm{dev}} > 0$.

Methods. We compare our approach (`LV`) to several baselines: (i) a `default` policy which myopically optimizes for immediate engagement (and so does not break); (ii) a 'safety switch' policy (`safety@`τ) that breaks once consumption surpasses a threshold τ; (iii) a prediction-based policy (`best-of`) that chooses the best observed $p_u = \mathrm{argmax}_{p_j} f_j(u)$ (rather than optimizing over $p_u \in [0, 1]$); and (iv) an `oracle` benchmark which directly optimizes the (generally unknown) true underlying dynamics. We measure mean long-term engagement rate (LTE) for each approach, and report averages and standard errors computed over 10 random splits. Performance is measured relative to the `default` baseline as it represents no change in policy (typical absolute values are LTE ≈ 10).

E.2 Results and Analysis

Main results. Figure 5 (left) compares the performance of our method to other policies. Here we set $p_0 = 0$, use $N = 3$ with $p_j \in \{0.05, 0.1, 0.15\}$, and consider $\kappa = 0.5$ (note κ affects all policies). As can be seen, our approach significantly improves over `default` (+5.98%). For `safety@`τ, improvement over the optimal $\tau = 16$ (+5.74%; chosen in hindsight) shows the importance of being preemptive; for the slightly smaller $\tau = 14$, breaks are harmful. The gap from `best-of` (+2.05%) quantifies the gain from the optimization step in Eq. (13), and the close performance to `oracle` (-0.797%) suggests that optimizing the empirical curve (Eq. (17)) works well as as proxy.

User types. Figure 5 (center) shows for our approach how gain in LTE varies across learned breaking policies $\hat{p}_u > 0$. For increasingly-accurate predictions ($\kappa \in \{0, 0.5, 1\}$), the main plot shows performance gains for each group of users, partitioned by their $\hat{p}_u$ values (binned; plot shows average and unit standard deviation per bin). Gains until $\hat{p}_u \leq 0.15$ are mild, but for $\hat{p}_u > 0.15$, the general trend is positive: users who are deemed to require more frequent breaks, benefit more from breaking. Gains until $\hat{p}_u \leq 0.3$ increase for all κ, but for $\hat{p}_u > 0.3$, extrapolation becomes difficult: note the higher variation within each κ, as well as significant differences across κ. This highlights the importance of accurate predictions for inferring optimal $\hat{p}_u$ when the experimental p_i are small. The inlaied plot shows that, in line with our theory, $\hat{p}_u$ exhibits an empirical phase shift in the estimated $\hat{\theta}_u$.

Treatments. Figure 5 (right) shows the effects of experimental treatments on performance. Focusing on $N = 2$, we fix $p_0 = 0$, and consider increasingly aggressive experimentation by varying $p_1 \in (0, 0.4]$. For our approach, increasing p_1 helps, which is anticipated by our theoretical bound.

For the `best-of` approach, larger p_1 also helps, but exhibits population-level optimum ($p_1 \approx 0.24$), which is easy to 'overshoot'. Note that when prediction accuracy is low ($\kappa = 0$), experimentation is essential: if p_1 is not sufficiently large, then performance can sharply deteriorate.

F Discussion

Our paper studies the novel problem of learning optimal breaking policies for recommendation. We posit a tight connection between long-term engagement and user well-being, and argue that optimizing the former requires careful consideration of the latter. Viewing user interest as a limited we study the role of breaks in facilitating sustainable habits, and propose an efficient algorithm for learning breaking policies that optimize long-term engagement. Our approach relies on LV models at its core, but incorporating more elaborate dynamic models is appealing as future work.

The recommendation setting we study is simple, but offers what we believe is a plausible perspective on the dynamics of user behavior—with emphasis on the importance of bi-directional feedback in shaping outcomes for the system, and for its users. Nonetheless, further work is necessary to establish the degree to which our stylized model is valid in reality. Our hopes are that our work takes one step towards establishing recommendation system as *ecosystems*—requiring active, planned, and regulated conservation.

G Broader perspective

G.1 Ethics statement

As any policy task that involves humans, care must be taken regarding potential risks. While our experiments show that optimizing engagement also improves well-being, this need not always be the case; in fact, merely measuring well-being in reality is challenging. Breaks, as interventions, are presumably 'safe', in the sense that at worst they may lead to suboptimal performance (for system) or satisfaction (for users). But breaks can also be used nefariously, e.g., by enabling temporally-varying rewards (Eyal, 2019). As such, break suggestion decision should be made transparently.

G.2 Recommendation as conservation

At a high level, our work argues for viewing recommendation as a task of *sustainable resource management*. As other cognitive tasks, engaging with digital content requires the availability of certain cognitive resources—attentional, executive, or emotional. These resources are inherently *limited*, and prolonged engagement depletes them (Kahneman, 1973; Muraven & Baumeister, 2000); this, in turn, can reduce the capacity of key cognitive processes (e.g., perception, attention, memory, self-control, and decision-making), and in the extreme—cause ego depletion (Baumeister et al., 1998) or cognitive fatigue (Mullette-Gillman et al., 2015). As a means to allow resources to replenish, 'mental breaks' have been shown to be highly effective (Bergum & Lehr, 1962; Hennfng et al., 1989; Gilboa et al., 2008; Ross et al., 2014; Helton & Russell, 2017).

Nevertheless, traditional approaches to recommendation remain agnostic to the idea that recommending takes a cognitive toll: they simply recommend at each point in time the item predicted to be most engaging (Robertson, 1977). As an alternative, our approach explicitly models recommendation as a process which draws on these resources, and therefore—must also conserve them. The subclass of 'Predator-Prey' LV dynamics which we draw on are used extensively in ecology for modeling the dynamics of interacting populations, and demonstrate how over-predation can ultimately lead to self-extinction by eliminating the prey population—but also show how enabling resources to naturally replenish ensures sustainable relations. As such, here we advocate for studying recommendation systems as human-centric *ecosystems*, and take one step towards their sustainable design.

H Deferred proofs

In this section, we formalize the model presented in section 2, and prove the claims presented in appendix B. The section ends with a formal proof of theorem D.1.

H.1 Properties of Lotka-Volterra systems

Definition H.1 (Static policy equilibrium)**.** Let $\lambda(t), q(t)$ denote a Lokta-Volterra model characterized by parameters $\theta = (\alpha, \beta, \gamma, \delta) \in \mathbb{R}_+^4$, as defined in Eq. (10). Let $p \in [0, 1]$, and denote by π_p the static policy corresponding to p. For $\lambda(0), q(0) > 0$, the static equilibrium of the system is defined as:

$$\lambda^*(p; \theta) = \lim_{t \to \infty} \lambda(t)$$
$$q^*(p; \theta) = \lim_{t \to \infty} q(t)$$

We denote $\lambda^*(p) = \lambda^*(p; \theta)$ when θ is clear from the context. We denote $\lambda^*(p; u) = \lambda^*(p; \theta_u)$ when a user $u \in \mathcal{U}$ characterized by parameters θ_u is given and clear from the context.

Proposition H.2 (Global stability)**.** *$\lambda^*(p; \theta)$ exists and uniquely defined for all $\theta \in \mathbb{R}_+^4$, $p \in [0, 1]$ and for all initial conditions $\lambda(0), q(0) > 0$.*

Proof. See (Takeuchi, 1996, Section 3.2). □

Lemma H.3 (Equilibrium of LV behavioral model. Formal proof of lemma B.1)**.** *Assume a Lokta-Volterra model characterized by $\theta = (\alpha, \beta, \gamma, \delta) \in \mathbb{R}_+^4$, and let $p \in [0, 1]$ denote the proportion of interactions in which a forced break is served. The static equilibrium of the model under static policy π_p is given by:*

$$\lambda^*(p) = \begin{cases} \frac{\gamma}{\delta} \frac{1}{1-p} \left(1 - \frac{\alpha}{\beta} \frac{1}{1-p}\right) & p \in \left[0, 1 - \frac{\alpha}{\beta}\right] \\ 0 & \text{otherwise} \end{cases}$$
$$q^*(p) = \begin{cases} \frac{\alpha}{\beta} \frac{1}{1-p} & p \in \left[0, 1 - \frac{\alpha}{\beta}\right] \\ 1 & \text{otherwise} \end{cases}$$

Proof. The LV dynamical system is given by Eq. (10):

$$\frac{\mathrm{d}\lambda}{\mathrm{d}t} = -\alpha\lambda + \beta(1-p)\lambda q$$
$$\frac{\mathrm{d}q}{\mathrm{d}t} = \gamma q(1-q) - \delta(1-p)\lambda q$$

when $p \in \left[0, 1 - \frac{\alpha}{\beta}\right]$ we equate $\frac{\mathrm{d}\lambda}{\mathrm{d}t} = 0$, $\frac{\mathrm{d}q}{\mathrm{d}t} = 0$ and obtain the result. The solution is guaranteed to be valid, as both $\lambda^*(p) > 0$ and $q^*(p) \in [0, 1]$.

Conversely, when $p \notin \left[0, 1 - \frac{\alpha}{\beta}\right]$, there exists $\epsilon > 0$ such that $\frac{\mathrm{d}}{\mathrm{d}t} \log \lambda < -\epsilon < 0$ for all $\lambda > 0$, $q \in [0, 1]$. From this we obtain that $\log \lambda(t)$ tends towards $-\infty$, and therefore $\lambda(t)$ tends towards 0, and $\lambda^*(p) = 0$ as required. When $\lambda(t)$ is close to zero, the interaction terms vanish in the $\frac{\mathrm{d}q}{\mathrm{d}t}$ equation, and $q(t)$ grows logistically towards 1. □

Proposition H.4 (Equilibrium bounds)**.** *For a Lotka-Volterra model, the static equilibrium $\lambda^*(p)$ is bounded by:*

$$0 \leq \lambda^*(p) \leq \frac{\beta\gamma}{4\alpha\delta}$$

Proof. Denote $x = \frac{1}{1-p}$. From lemma H.3, for $x \in \left[1, \frac{\beta}{\alpha}\right]$ the equilibrium consumption $\lambda^*(x)$ is given by:

$$\lambda^*(x) = \frac{\gamma}{\delta} x \left(1 - \frac{\alpha}{\beta} x\right)$$

and is zero otherwise. The equilibrium is a quadratic function of x with roots $x \in \left\{0, \frac{\beta}{\alpha}\right\}$, and therefore attains its maximum at $x = \frac{\beta}{2\alpha}$. Plugging back the maximizing x into λ^* we obtain the upper bound. Lower bound is attained as the equilibrium in lemma H.3 is clipped by 0 from below. □

Lemma H.5 (Optimal static policy. Formal proof of lemma B.2)**.** *The optimal static policy for a Lotka-Volterra system is given by:*

$$p_{\text{opt}} = \begin{cases} 1 - 2\frac{\alpha}{\beta} & \frac{\alpha}{\beta} \leq \frac{1}{2} \\ 0 & \frac{\alpha}{\beta} > \frac{1}{2} \end{cases}$$

And the optimal equilibrium engagement rate is given by:

$$\lambda^*_{\text{opt}} = \begin{cases} \frac{\beta\gamma}{4\alpha\delta} & \frac{\alpha}{\beta} \leq \frac{1}{2} \\ \frac{\gamma}{\delta}\left(1 - \frac{\alpha}{\beta}\right) & \frac{\alpha}{\beta} > \frac{1}{2} \end{cases}$$

Proof. Denote $x = \frac{1}{1-p}$. From proposition H.4, the global maximum of $\lambda^*(x)$ is attained at $x = \frac{\beta}{2\alpha}$. Consider two cases: When $\frac{\alpha}{\beta} \leq \frac{1}{2}$, we obtain that $x_{\text{opt}} = \frac{\beta}{2\alpha} \geq 1$, and therefore $p_{\text{opt}} = 1 - \frac{1}{x} \in [0, 1]$. From this we obtain that in this case the global maximum is attained on the simplex, and given by the formula from proposition H.4. Conversely, when $\frac{\alpha}{\beta} > \frac{1}{2}$, we obtain $p = 1 - \frac{1}{x} < 0$, and therefore x_{opt} translates to a negative value of p. As $\lambda^*(p)$ is uni-modal, the optimal policy restricted to the simplex $[0, 1]$ in this case is attained on the closest boundary point $p = 0$.

fig. 2 (Left, Center) provides graphical intuition for this proof. □

Proposition H.6 (Inference of α/β from two-treatment equilibrium data. Formal proof of proposition C.1)**.** *Let $\lambda(t), q(t)$ be a Lokta-Volterra model, let $p_1, p_2 \in [0, 1]$. Denote by $\lambda^*(p_1), \lambda^*(p_2)$ the static equilibrium rates corresponding to static policies π_{p_1}, π_{p_2}, and assume $\lambda^*(p_1), \lambda^*(p_2) > 0$. The parameter ratio $\frac{\alpha}{\beta}$ is given by the following formula:*

$$\frac{\alpha}{\beta} = \frac{(1-p_2)\lambda^*(p_2) - (1-p_1)\lambda^*(p_1)}{\frac{1}{1-p_1} - \frac{1}{1-p_2}}$$

Proof. From lemma H.3, the equilibrium consumption $\lambda^*(p)$ is given by:

$$\begin{aligned} \lambda^*(p) &= \frac{\gamma}{\delta}\frac{1}{1-p}\left(1 - \frac{\alpha}{\beta}\frac{1}{1-p}\right) \\ &= \frac{\gamma}{\delta}\frac{1}{1-p} - \frac{\alpha}{\beta}\frac{\gamma}{\delta}\left(\frac{1}{1-p}\right)^2 \end{aligned}$$

When $\lambda^*(p_i)$ is observed for different policies $p_1, \ldots, p_m \in \left[0, 1 - \frac{\alpha}{\beta}\right]$, we obtain a polynomial regression problem for the parameters $\frac{\alpha}{\beta}$ and $\frac{\alpha}{\beta}\frac{\gamma}{\delta}$, which can be solved e.g using Non-Negative Least Squares.

When $m = 2$, we obtain a system of two linear equations. Apply Cramer's rule to obtain:

$$\frac{\gamma}{\delta} = \frac{\frac{\lambda^*(p_2)}{(1-p_1)^2} - \frac{\lambda^*(p_1)}{(1-p_2)^2}}{\frac{1}{(1-p_1)(1-p_2)^2} - \frac{1}{(1-p_1)^2(1-p_2)}} = \frac{(1-p_2)^2\lambda^*(p_2) - (1-p_1)^2\lambda^*(p_1)}{p_2 - p_1} \tag{19}$$

$$\frac{\alpha}{\beta}\frac{\gamma}{\delta} = \frac{\frac{\lambda^*(p_2)}{(1-p_1)} - \frac{\lambda^*(p_1)}{(1-p_2)}}{\frac{1}{(1-p_1)(1-p_2)^2} - \frac{1}{(1-p_1)^2(1-p_2)}} = (1-p_1)(1-p_2)\frac{(1-p_2)\lambda^*(p_2) - (1-p_1)\lambda^*(p_1)}{p_2 - p_1} \tag{20}$$

And therefore $\frac{\alpha}{\beta}$ is given by:

$$\frac{\alpha}{\beta} = \frac{\frac{\lambda^*(p_2)}{(1-p_1)} - \frac{\lambda^*(p_1)}{(1-p_2)}}{\frac{\lambda^*(p_2)}{(1-p_1)^2} - \frac{\lambda^*(p_1)}{(1-p_2)^2}} = (1-p_1)(1-p_2)\frac{(1-p_2)\lambda^*(p_2) - (1-p_1)\lambda^*(p_1)}{(1-p_2)^2\lambda^*(p_2) - (1-p_1)^2\lambda^*(p_1)}$$

□

H.2 MODEL FITTING FROM ENGAGEMENT PREDICTIONS

Notations. In this section only, we use the common notation $q = 1 - p$ to denote complementary probabilities.

Definition H.7 (Empirical value of α/β)**.** For single-channel experiments with forced-break probabilities p_1, p_2, denote $\lambda_i = \lambda^*(p_i)$, $f_i = f_{p_i}(u)$, $q_i = 1 - p_i$. The empirical value of the $\frac{\alpha}{\beta}$ parameter is given by the following formula:

$$\frac{\hat{\alpha}}{\beta} = \frac{q_1 q_2 \left(q_1 f_1 - q_2 f_2\right)}{q_1^2 f_1 - q_2^2 f_2}$$

Proposition H.8 (α/β estimation error from prediction errors)**.** *Given a single-channel Lokta-Volterra system with parameter $\frac{\alpha}{\beta} \geq 1$. Let $p_1, p_2 \in \left[1, \frac{\alpha}{\beta}\right]$, denote $\lambda_i^* = \lambda^*(p_i) \in \mathbb{R}_+$, and let $f_i = \lambda_i^* + \varepsilon_i$ be the predicted engagement rates corresponding to p_1, p_2. When $|\varepsilon_1|, |\varepsilon_2| \leq \varepsilon \leq \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}$, the estimation error is bounded by:*

$$\left|\frac{\alpha}{\beta} - \frac{\hat{\alpha}}{\beta}\right| \leq \frac{\varepsilon}{|p_1 - p_2|}\frac{\beta\delta}{\alpha\gamma}$$

Proof. denote $q_i = 1 - p_i$. The value of $\frac{\alpha}{\beta}$ is given by proposition H.6:

$$\frac{\alpha}{\beta} = \frac{q_1 q_2 \left(q_1 \lambda_1^* - q_2 \lambda_2^*\right)}{q_1^2 \lambda_1^* - q_2^2 \lambda_2^*}$$

And the estimator for $\frac{\alpha}{\beta}$ is obtained by replacing the true value with their predictions:

$$\begin{aligned}\frac{\hat{\alpha}}{\beta} &= \frac{q_1 q_2 \left(q_1 f_1 - q_2 f_2\right)}{q_1^2 f_1 - q_2^2 f_2} \\ &= \frac{q_1 q_2 \left(q_1(\lambda_1^* + \varepsilon_1) - q_2(\lambda_2^* + \varepsilon_2)\right)}{q_1^2(\lambda_1^* + \varepsilon_1) - q_2^2(\lambda_2^* + \varepsilon_2)}\end{aligned}$$

The estimation error is given by:

$$\begin{aligned}\left|\frac{\alpha}{\beta} - \frac{\hat{\alpha}}{\beta}\right| &= \left|\frac{q_1^2 q_2^2 (q_1 - q_2)(\varepsilon_2 \lambda_1^* - \varepsilon_1 \lambda_2^*)}{(q_1^2 \lambda_1^* - q_2^2 \lambda_2^*)(q_1^2 \lambda_1^* - q_2^2 \lambda_2^* - (q_1^2 \varepsilon_1 - q_2^2 \varepsilon_2))}\right| \\ &= \underbrace{\left(q_1 q_2\right)^2}_{\equiv\text{(i)}} \underbrace{\left|\frac{q_1 - q_2}{q_1^2 \lambda_1^* - q_2^2 \lambda_2^*}\right|}_{\equiv\text{(ii)}} \underbrace{\left|\varepsilon_2 \lambda_1^* - \varepsilon_1 \lambda_2^*\right|}_{\equiv\text{(iii)}} \underbrace{\left|\frac{1}{q_1^2 \lambda_1^* - q_2^2 \lambda_2^* - (q_1^2 \varepsilon_1 - q_2^2 \varepsilon_2)}\right|}_{\equiv\text{(iv)}}\end{aligned}$$

We now proceed to bound each factor:

- For (i), the term $\left(q_1 q_2\right)^2$ is bounded by 1 since $q_1, q_2 \in [0, 1]$.
- For (ii), the term $\left|\frac{q_1 - q_2}{q_1^2 \lambda_1^* - q_2^2 \lambda_2^*}\right|$ is equal to $\left(\frac{\gamma}{\delta}\right)^{-1}$ by Eq. (19).
- For (iii), from proposition H.4 we obtain the bound $0 \leq \lambda_i^* \leq \frac{\beta\gamma}{4\alpha\delta}$, and therefore the term $|\varepsilon_2 \lambda_1^* - \varepsilon_1 \lambda_2^*|$ is bounded by $2\left(\frac{\beta\gamma}{4\alpha\delta}\right)\varepsilon = \frac{\beta\gamma}{2\alpha\delta}\varepsilon$.

- For (iv), the term $\left|\frac{1}{q_1^2\lambda_1^* - q_2^2\lambda_2^* - (q_1^2\varepsilon_1 - q_2^2\varepsilon_2)}\right|$ is equal to:

$$\begin{aligned}
\text{(iv)} &\equiv \left|\frac{1}{q_1^2\lambda_1^* - q_2^2\lambda_2^* - (q_1^2\varepsilon_1 - q_2^2\varepsilon_2)}\right| \\
&= \frac{1}{|p_1 - p_2|}\left|\frac{q_1^2\lambda_1^* - q_2^2\lambda_2^* - (q_1^2\varepsilon_1 - q_2^2\varepsilon_2)}{p_1 - p_2}\right|^{-1} \\
&= \frac{1}{|p_1 - p_2|}\left|\underbrace{\frac{q_1^2\lambda_1^* - q_2^2\lambda_2^*}{p_1 - p_2}}_{\text{Eq. (19)}} - \frac{q_1^2\varepsilon_1 - q_2^2\varepsilon_2}{p_1 - p_2}\right|^{-1} \\
&= \frac{1}{|p_1 - p_2|}\left|\frac{\gamma}{\delta} - \frac{q_1^2\varepsilon_1 - q_2^2\varepsilon_2}{p_1 - p_2}\right|^{-1}
\end{aligned}$$

Note that $\left|\frac{q_1^2\varepsilon_1 - q_2^2\varepsilon_2}{p_1 - p_2}\right| \leq \frac{2\varepsilon}{|p_1 - p_2|}$. When ε is small enough, and specifically when the bound $\varepsilon \leq \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}$ holds, we obtain:

$$\left|\frac{\gamma}{\delta} - \frac{q_1^2\varepsilon_1 - q_2^2\varepsilon_2}{p_1 - p_2}\right|^{-1} \leq \frac{\delta}{\gamma}\left|1 - \frac{1}{2}\right|^{-1} \leq 2\frac{\delta}{\gamma}$$

and therefore:

$$\text{(iv)} \leq \frac{2}{|p_1 - p_2|}\frac{\delta}{\gamma}$$

Aggregating results (i)-(iv) above, we obtain the overall bound:

$$\begin{aligned}
\left|\frac{\alpha}{\beta} - \frac{\hat{\alpha}}{\beta}\right| &= \underbrace{(q_1 q_2)^2}_{\leq 1}\underbrace{\left|\frac{q_1 - q_2}{q_1^2\lambda_1^* - q_2^2\lambda_2^*}\right|}_{=\frac{\delta}{\gamma}}\underbrace{\left|\varepsilon_2\lambda_1^* - \varepsilon_1\lambda_2^*\right|}_{\leq \frac{\beta\gamma}{2\alpha\delta}\varepsilon}\underbrace{\left|\frac{1}{q_1^2\lambda_1^* - q_2^2\lambda_2^* - (q_1^2\varepsilon_1 - q_2^2\varepsilon_2)}\right|}_{\leq \frac{2}{|p_1 - p_2|}\frac{\delta}{\gamma}} \\
&\leq \frac{\varepsilon}{|p_1 - p_2|}\frac{\beta\delta}{\alpha\gamma}
\end{aligned}$$

□

Proposition H.9 (Cost of α/β estimation error)**.** *Let $\frac{\alpha}{\beta}$ be the engagement ratio parameter of a one-channel Lotka-Volterra system, and let $\left(\hat{\frac{\alpha}{\beta}}\right)$ be an estimate of these parameters. Let λ^*_{opt} be the engagement rate of the optimal static policy, and denote $\lambda^*(x) = \lambda^*\left(\hat{p}(x)\right)$. When $\left|\frac{\alpha}{\beta} - \left(\hat{\frac{\alpha}{\beta}}\right)\right| \leq \min\left\{\frac{\alpha}{2\beta}, 1\right\}$ The price of estimation error is bounded by:*

$$\lambda^*_{\text{opt}} - \lambda^*\left(\left(\hat{\frac{\alpha}{\beta}}\right)\right) \leq \left(\frac{\gamma}{\delta}\right)\min\left\{\left(2\frac{\alpha}{\beta}\right)^{-2}\left|\frac{\alpha}{\beta} - \left(\hat{\frac{\alpha}{\beta}}\right)\right|, \left(4\frac{\alpha}{\beta}\right)^{-1}\right\}$$

Proof. Denote $r = \frac{\alpha}{\beta}, x = \left(\hat{\frac{\alpha}{\beta}}\right)$, and assume without loss of generality that $\frac{\gamma}{\delta} = 1$ and $r \leq 1$. The optimal equilibrium engagement rate is given by:

$$\lambda^*_{\text{opt}} = \begin{cases} \frac{1}{4r} & r \in \left(0, \frac{1}{2}\right] \\ 1 - r & r \in \left(\frac{1}{2}, 1\right] \end{cases}$$

The chosen policy $\hat{p}(x)$ is given by:

$$\hat{p}(x) = \begin{cases} 1 - 2x & x \in \left[0, \frac{1}{2}\right] \\ 0 & \text{otherwise} \end{cases}$$

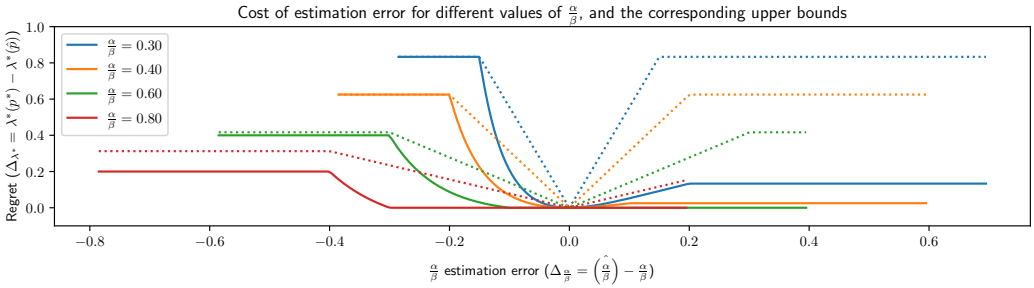

Figure 6: Graphical illustration of proposition H.9. Cost of estimation error for different values of $\frac{\alpha}{\beta}$, and their corresponding upper bounds given by the claim.

Assume without loss of generality that $x \in \left[0, \frac{1}{2}\right]$, as values of x outside the interval can be clipped to its edges without affecting the result. The equilibrium engagement rate of the selected policy is given by:

$$\lambda^*(x) = \lambda^*\left(\hat{p}(x)\right) = \begin{cases} 0 & x \in \left[0, \frac{r}{2}\right] \\ \frac{1}{2x}\left(1 - \frac{r}{2x}\right) & x \in \left(\frac{r}{2}, \frac{1}{2}\right] \end{cases}$$

Denote $\Delta(x) = \lambda^*_{\text{opt}} - \lambda^*(x)$. We obtain:

$$\Delta(x) = \lambda^*_{\text{opt}} - \lambda^*(x) = \begin{cases} \frac{1}{4r} & r \in \left(0, \frac{1}{2}\right], x \in \left[0, \frac{r}{2}\right] \\ \frac{(x-r)^2}{4x^2 r} & r \in \left(0, \frac{1}{2}\right], x \in \left(\frac{r}{2}, \frac{1}{2}\right] \\ (1-r) & r \in \left(\frac{1}{2}, 1\right], x \in \left[0, \frac{r}{2}\right] \\ (1-r) - \frac{1}{2x}\left(1 - \frac{r}{2x}\right) & r \in \left(\frac{1}{2}, 1\right], x \in \left(\frac{r}{2}, \frac{1}{2}\right] \end{cases}$$

Observe that $\frac{1}{4r} \geq 1 - r$ for all $r \in (0, 1]$, and therefore we obtain for all x, r:

$$\Delta(x) \leq \frac{1}{4r} \tag{21}$$

From the convexity of $\Delta(x)$ in the region around $x = r$ we obtain:

$$\Delta(x) \leq \frac{1}{2r^2}|x - r| \tag{22}$$

Finally, combining the two bounds yields the final result. A geometric interpretation of this claim is illustrated in fig. 6. □

H.3 Optimal stationary policy from engagement predictions

Definition H.10 (Expected observable rate)**.** Let $u \in \mathcal{U}$, $p \in [0, 1]$, and $T > 0$. Let $p \in [0, 1]$, denote the corresponding static policy by π_p. The expected observable rate $\bar{\lambda}_u(p; T)$ is defined as:

$$\bar{\lambda}_u(p; u) = \mathbb{E}_\pi\left[\frac{1}{T}\left|\mathcal{S}_{\pi_p}(u; T)\right|\right]$$

where expectation is taken over the stochastic decisions of π_p.

Definition H.11 (Lokta-Volterra approximation of $\mathcal{S}$)**.** Let $u \in \mathcal{U}$, and $T > 0$. Denote by p^* the maximizer of expected observable rate:

$$p^* = \operatorname*{argmax}_{p \in [0,1]} \bar{\lambda}_u(p; u)$$

The LV approximation of $\mathcal{S}(u; T)$ is defined as:

$$\theta^*_u = \operatorname*{argmin}_{\theta} \max_{p \in [0,1]} \left|\bar{\lambda}_u(p; u) - \lambda^*(p; \theta)\right|$$

such that $\operatorname{argmax}_p \lambda^*(p; \theta) = p^*$. The corresponding approximation error is defined as:

$$\varepsilon_{\text{LV},u} = \max_{p \in [0,1]} \left|\bar{\lambda}_u(p; u) - \lambda^*(p; \theta^*_u)\right|$$

Notations. When u is clear from the context, we denote $\theta^* = \theta^*_u$, $\varepsilon_{\text{LV}} = \varepsilon_{\text{LV},u}$. We use $\alpha^*, \beta^*, \ldots$ to refer to the corresponding parts of the Lokta-Volterra parameters vector θ^*.

We are now ready to state and prove the main theorem for this section:

Theorem H.12 (Regret bound for learned static policy. Formal version of theorem D.1)**.** *Let $p_1, p_2 \in [0,1]$ denote two static forced-break policies, and denote by $\mathcal{U}$ the set of users, and assume they remain engaged under the stationary policies $\pi(p_1)$ and $\pi(p_2)$. Assume $S_u(p;T) \sim \mathcal{S}_{\pi_p \circ \psi}(u;T)$, and let $\mu = \left(\max_{u\in\mathcal{U}} \frac{\bar{\gamma}_u}{\bar{\delta}_u}\right) \cdot \left(\max_{u'\in\mathcal{U}} \frac{\bar{\delta}_{u'}}{\bar{\gamma}_{u'}}\right)$, $\nu = \max_{u\in\mathcal{U}} \left(\frac{\bar{\beta}_u}{\bar{\alpha}_u}\right)$.*

Let $f_{p_1}, f_{p_2} : \mathcal{U} \to \mathbb{R}_+$ be functions predicting $\frac{1}{T}|S_u(p_1;T)|$, $\frac{1}{T}|S_u(p_2;T)|$, respectively. Denote the learned policy by $\hat{p}$, and the optimal policy by p^*.

If (i) the expected RMSE of f_{p_1}, f_{p_2} is bounded by $\varepsilon_{\text{pred}}$, (ii) the average absolute deviation of $\frac{1}{T}|\mathcal{S}(u;T)|$ is bounded by ε_{dev}, and (iii) the expected LV approximation error of the system is bounded by ε_{LV}, then the learned policy $\hat{p}$ has bounded regret:

$$\mathbb{E}_{u,\pi}\left[\left|\tfrac{1}{T}|S_u(p^*;T)| - \tfrac{1}{T}|S_u(\hat{p};T)|\right|\right] \leq \frac{\eta_{\mathcal{S}}}{|p_1 - p_2|}(\varepsilon_{\text{pred}} + \varepsilon_{\text{dev}} + \varepsilon_{\text{LV}})$$

where expectation is taken over stochastic choices of policies, and $\eta_{\mathcal{S}} = g(\mu,\nu) \in \text{poly}(\mu,\nu)$.

Proof. By assumption (i), the functions f_{p_1}, f_{p_2} have bounded expected RMSE:

$$\mathbb{E}_u\left[\left(f_{p_i}(u) - \tfrac{1}{T}|S_u(p_i;T)|\right)^2\right] \leq \varepsilon^2_{\text{pred}} \tag{23}$$

Applying Jensen's inequality with the convex function $\varphi(x) = x^2$ yields:

$$\left(\mathbb{E}_u\left[\left|f_{p_i}(u) - \tfrac{1}{T}|S_u(p_i;T)|\right|\right]\right)^2 \leq \mathbb{E}_u\left[\left(f_{p_i}(u) - \tfrac{1}{T}|S_u(p_i;T)|\right)^2\right]$$

Combining with Eq. (23) and taking the square root, we obtain an upper bound on the expected absolute error:

$$\mathbb{E}_u\left[\left|f_{p_i}(u) - \tfrac{1}{T}|S_u(p_i;T)|\right|\right] \leq \varepsilon_{\text{pred}} \tag{24}$$

Let $\Delta_f = |f_{p_i}(u) - \lambda^*(p_i)|$ apply the triangle inequality to obtain:

$$\begin{aligned}\Delta_f &= |f_{p_i}(u) - \lambda^*(p_i)| \\ &\leq \left|f_{p_i}(u) - \tfrac{1}{T}|S_u(u;T)|\right| + \left|\tfrac{1}{T}|S_u(u;T)| - \bar{\lambda}(p_i;u)\right| + \left|\bar{\lambda}(p_i;u) - \lambda^*(p_i)\right|\end{aligned}$$

Denote $\varepsilon_f = \varepsilon_{\text{pred}} + \varepsilon_{\text{dev}} + \varepsilon_{\text{LV}}$. Applying the triangle inequality and using the bounds in Eq. (24) together with assumptions (ii), (iii), we obtain:

$$\begin{aligned}\mathbb{E}_{u,\pi}[\Delta_f] \leq& \mathbb{E}_u\left[\left|f_{p_i}(u) - \tfrac{1}{T}|S_u(u;T)|\right|\right] \\ &+ \mathbb{E}_{u,\pi}\left[\left|\tfrac{1}{T}|S_u(u;T)| - \bar{\lambda}(p_i;u)\right|\right] \\ &+ \mathbb{E}_u\left[\left|\bar{\lambda}(p_i;u) - \lambda^*(p_i;\theta^*_u)\right|\right] \\ \leq& \varepsilon_{\text{pred}} + \varepsilon_{\text{dev}} + \varepsilon_{\text{LV}} = \varepsilon_f\end{aligned} \tag{25}$$

Denote $\theta^*_u = (\alpha, \beta, \gamma, \delta)$. The empirical value $\hat{\left(\frac{\alpha}{\beta}\right)}$ of $\left(\frac{\alpha}{\beta}\right)$ is given by definition H.7. Denote the estimation error by $\Delta_{\frac{\alpha}{\beta}} = \left|\hat{\left(\frac{\alpha}{\beta}\right)} - \left(\frac{\alpha}{\beta}\right)\right|$.

By proposition H.8, the following pointwise upper bound on $\Delta_{\frac{\alpha}{\beta}}$ applies when $\Delta_f \leq \frac{\gamma}{\delta}\frac{|p_1-p_2|}{4}$:

$$\Delta_{\frac{\alpha}{\beta}} \leq \frac{\Delta_f}{|p_1 - p_2|}\frac{\beta\delta}{\alpha\gamma} \tag{26}$$

Plugging in the bound on the expected value of Δ_f into Eq. (26), we obtain in expectation:

$$\begin{aligned}\mathbb{E}_{u,\pi}\left[\Delta_{\frac{\alpha}{\beta}} \mid \Delta_f \leq \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}\right] &\leq \mathbb{E}_{u,\pi}\left[\frac{\Delta_f}{|p_1 - p_2|}\frac{\beta\delta}{\alpha\gamma} \mid \Delta_f \leq \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}\right] \\ &\leq \frac{\varepsilon_f}{|p_1 - p_2|}\max_u \frac{\beta\delta}{\alpha\gamma}\end{aligned} \tag{27}$$

Next, we apply proposition H.9. Denote $\Delta_{\lambda^*} = \lambda^*(p^*) - \lambda^*(\hat{p})$, and define the following probability event:

$$A = \left(\Delta_f \leq \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}\right) \text{ and } \left(\Delta_{\frac{\alpha}{\beta}} \leq \frac{1}{2\nu}\right)$$

Note that the bound in proposition H.9 is represented as a minimum between two functions, one linear in ε and one constant. To leverage this property, apply the law of total expectation:

$$\mathbb{E}_{u,\pi}[\Delta_{\lambda^*}] = \mathbb{E}_{u,\pi}[\Delta_{\lambda^*} \mid A]\mathbb{P}[A] + \mathbb{E}_{u,\pi}\left[\Delta_{\lambda^*} \mid \bar{A}\right]\mathbb{P}\left[\bar{A}\right] \tag{28}$$

Under A, the first term in Eq. (28) can be bounded by the linear term in proposition H.9. Taking $\mathbb{P}[A] \leq 1$ and combining with equation Eq. (26):

$$\begin{aligned}
\mathbb{E}_{u,\pi}[\Delta_{\lambda^*} \mid A]\mathbb{P}[A] &\leq \mathbb{E}_{u,\pi}[\Delta_{\lambda^*} \mid A] \\
&\leq \mathbb{E}_{u,\pi}\left[\frac{\beta^2\gamma}{2\alpha^2\delta}\Delta_{\frac{\alpha}{\beta}} \mid A\right] \\
&\leq \mathbb{E}_{u,\pi}\left[\frac{\beta^2\gamma}{2\alpha^2\delta}\frac{\Delta_f}{|p_1 - p_2|}\frac{\beta\delta}{\alpha\gamma} \mid A\right] \\
&\leq \frac{\nu^3}{2|p_1 - p_2|}\varepsilon_f
\end{aligned} \tag{29}$$

The expectation factor in the second term of Eq. (28) can be bounded by the constant term in proposition H.9:

$$\mathbb{E}_{u,\pi}\left[\Delta_{\lambda^*} \mid \bar{A}\right] \leq \frac{1}{4}\max_u \frac{\beta\gamma}{\alpha\delta} \leq \frac{\nu}{4}\max_u \frac{\gamma}{\delta} \tag{30}$$

Decompose the probability factor $\mathbb{P}\left[\bar{A}\right]$ using the law of total probability:

$$\begin{aligned}
\mathbb{P}\left[\bar{A}\right] &= \mathbb{P}\left[\Delta_f > \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}\right] + \mathbb{P}\left[\Delta_{\frac{\alpha}{\beta}} > \frac{1}{2\nu} \mid \Delta_f \leq \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}\right]\mathbb{P}\left[\Delta_f \leq \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}\right] \\
&\leq \mathbb{P}\left[\Delta_f > \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}\right] + \mathbb{P}\left[\Delta_{\frac{\alpha}{\beta}} > \frac{1}{2\nu} \mid \Delta_f \leq \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}\right]
\end{aligned}$$

Apply Markov's inequality $\mathbb{P}[|X| \geq a] \leq \frac{\mathbb{E}[|X|]}{a}$ on the probabilities to obtain:

$$\begin{aligned}
\mathbb{P}\left[\Delta_f > \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}\right] &\leq \mathbb{E}_{u,\pi}[\Delta_f]\left(\frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}\right)^{-1} \\
&\underset{\text{by Eq. (25)}}{\leq} \varepsilon_f \frac{4}{|p_1 - p_2|}\max_u \frac{\delta}{\gamma}
\end{aligned} \tag{31}$$

$$\begin{aligned}
\mathbb{P}\left[\Delta_{\frac{\alpha}{\beta}} > \frac{1}{2\nu} \mid \Delta_f \leq \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}\right] &\leq \mathbb{E}_{u,\pi}\left[\Delta_{\frac{\alpha}{\beta}} \mid \Delta_f \leq \frac{\gamma}{\delta}\frac{|p_1 - p_2|}{4}\right] \\
&\underset{\text{by Eq. (27)}}{\leq} \frac{\varepsilon_f}{|p_1 - p_2|}\max_u \frac{\beta\delta}{\alpha\gamma} \\
&\leq \frac{\varepsilon_f}{|p_1 - p_2|}\nu \max_u \frac{\delta}{\gamma}
\end{aligned} \tag{32}$$

Plugging back equations Eq. (29), Eq. (30),Eq. (31),Eq. (32) into equation Eq. (28), we obtain bounds for each term:

$$\mathbb{E}_{u,\pi}[\Delta_{\lambda^*}] = \underbrace{\mathbb{E}_{u,\pi}[\Delta_{\lambda^*} \mid A]\mathbb{P}[A]}_{\text{by eq. 29}} + \underbrace{\mathbb{E}_{u,\pi}\left[\Delta_{\lambda^*} \mid \bar{A}\right]}_{\text{by eq. 30}} \underbrace{\mathbb{P}\left[\bar{A}\right]}_{\text{by eqs. 31,32}} \tag{33}$$

we obtain:

$$\mathbb{E}_{u,\pi}[\Delta_{\lambda^*}] \leq \frac{\varepsilon_f}{|p_1 - p_2|}\left(\frac{\nu^3}{2} + \left(\nu + \frac{\nu^2}{4}\right)\mu\right) = \varepsilon_{\lambda^*}$$

To obtain the regret bound on the empirical rates, we apply assumptions (ii), (iii) once again to bound the expected difference between $\lambda^*(p)$ and $\frac{1}{T}|S_u(p;T)|$, and apply the triangle inequality:

$$\mathbb{E}_{u,\pi}\left[\left|\tfrac{1}{T}|S_u(p^*;T)| - \tfrac{1}{T}|S_u(\hat{p};T)|\right|\right] \leq \varepsilon_{\lambda^*} + 2(\varepsilon_{\text{dev}} + \varepsilon_{\text{LV}})$$

Note that $\frac{\nu}{|p_1-p_2|} > 1$, as $\frac{\beta}{\alpha} \geq 1$ since all the users are assumed to remain engaged in the long term, and $|p_1 - p_2| \leq 1$ as $p_1, p_2 \in [0,1]$. Therefore, the function $\eta_{\mathcal{S}} = g(\mu,\nu) = \left(\frac{\nu^3}{2} + \left(\nu + \frac{\nu^2}{4}\right)\eta + 2\nu\right)$ satisfies:

$$\mathbb{E}_{u,\pi}\left[\left|\tfrac{1}{T}|S_u(p^*;T)| - \tfrac{1}{T}|S_u(\hat{p};T)|\right|\right] \leq \frac{\eta_{\mathcal{S}}}{|p_1 - p_2|}(\varepsilon_{\text{pred}} + \varepsilon_{\text{dev}} + \varepsilon_{\text{LV}})$$

□

I Experimental details

I.1 Data

MovieLens-1M We base our main experimental environment on the MovieLens-1M dataset, which is a standard benchmark dataset used widely in recommendation system research (Harper & Konstan, 2015). The dataset includes 1,000,209 ratings provided by 6,040 users and for 3,706 movies. Ratings are in the range $\{1,\ldots,5\}$, and all users in the dataset have at least 20 reported ratings. The dataset is publicly available at: https://grouplens.org/datasets/movielens/1m/.

Goodreads. We validate our results using the Goodreads book recommendations dataset, which is a common benchmark dataset used in recommendation systems research (Wan & McAuley, 2018; Wan et al., 2019). Ratings are in the range $\{1,\ldots,5\}$, and the dataset is filtered to only include users with at least 20 reported ratings. We use the official comic-books genre subset of the dataset, which includes 3,679,076 ratings provided by 41,932 users and for 87,565 books after pre-processing. The dataset is publicly available at: https://sites.google.com/eng.ucsd.edu/ucsdbookgraph/. Pre-processing code is available in the repository: https://github.com/lvml-icml-2023/paper-code. We follow an identical experimental procedure for all datasets.

Data partitioning. To learn latent user and item features, 30% of all ratings were drawn at random. Stratified sampling was applied to ensure that all users and items were covered, and so that each users have roughly the same proportion of ratings used for this step. These ratings were only used only for learning a CF model, and were discarded afterwards. The remaining 70% data points were used for training and testing. For these, we first randomly sampled 1,000 users to form the test set. Then, the remaining users were partitioned into the main train set $\mathcal{S}$, which included 70% (≈3,528 for ML1M, ≈28,652 for Goodreads) of these users, and the experimental treatment sets $D^{(j)}$, each including 10% (≈504 for ML1M, ≈4,093 for Goodreads) users for $N = 3$. This procedure was repeated 10 times with different random seeds. We report average results, together with 95% t-distribution confidence intervals representing variation between runs.

I.2 Implementation details

- **Hardware**: All experiments were run on a single laptop, with 16GB of RAM, M1 Pro processor, and with no GPU support.
- **Runtime**: A single run consisting the entire pipeline (data loading and partitioning, collaborative filtering, training classifiers, simulating dynamics, learning policies, measuring and comparing performance) takes roughly 33 minutes. The main bottleneck is the discrete LV simulation, taking roughly 70% of runtime to compute, mostly due to bookkeeping necessary for the non-stationary baselines. Simulation code was optimized using the NUMBA jit compiler, which improves runtime.
- **Optimization packages**:
  - **Collaborative filtering (CF)**: We use the SURPRISE package (Hug, 2020), which includes an implementation of the SVD algorithm for CF. All parameters were set to default values.

  - **Regression**: We use the SCIKIT-LEARN implementation of linear regression for predicting long-term engagement from user features (i.e the prediction models $f_j(u)$ in Eq. (17)). All parameters were set to default values.
  - **Non-Negative Least Squares (NNLS)**: We use the SCIPY.OPTIMIZE implementation of NNLS. The algorithm was used with its default parameters.
- **Code**: Code for reproducing all of our figures and experiments is available in the following repository: https://github.com/lvml-icml-2023/paper-code.

I.3 OTHER BASELINES

- **Safety**: In each step of the TPP simulation, look k step back, and calculate the empirical rate $\tilde{\lambda}_i = \frac{k}{t_i - t_{i-k}}$. If this rate exceeds the threshold $\tilde{\lambda}_i > \tau$, the policy enters a 'cool-down' policy state, serving only forced breaks until the next time period. In our experiments, we used thresholds $\tau \in \{14, 16\}$, $k = 10$ look-behind steps, and defined the cool-down period as 0.5 time units.
- **Oracle**: To estimate the effect of perfect predictions, we implement an oracle predictor $f_p^{\text{oracle}}(u)$ which has access to the latent user parameters. For a given u and for each p, the predictor outputs the infinite-horizon LV equilibrium for u, namely $f_p^{\text{oracle}}(u) = \lambda^*(p; \tilde{\theta}_u)$. We define $\tilde{\theta}_u = (\alpha_u, \tilde{\beta}_u, \gamma_u, \delta_u)$, where $\alpha_u, \gamma_u, \delta_u$ are the unobserved parameters for the given user, and $\tilde{\beta}_u$ is the expected value of β_{ux} induced by the distribution over recommended items x induced by the recommendation policy ψ. We view $\tilde{\theta}_u$ as a useful proxy for the otherwise unattainable $\bar{\theta}_u$.

I.4 HYPERPARAMETERS

- **Collaborative filtering**: We used $d = 8$ latent factors and enabled bias terms, which ensured performance is close to the benchmark of $\text{RMSE} = 0.873$ reported in the SURPRISE documentation. We used the vanilla SVD solver, with all hyperparameters set to their default values.
- **Recommendation policy**: Softmax temperature was set to 0.5.
- **Prediction**: We trained regressors $f(u)$ on input feature vectors consisting of three components:

$$u = (\tilde{v}_u, b_u, \hat{\rho}_u) \in \mathbb{R}^{d+2} \tag{34}$$

  The three components are: (i) SVD latent user factors $\tilde{v}_u \in \mathbb{R}^d$, (ii) SVD user bias term $b_u \in \mathbb{R}$, (iii) an additional feature consisting of the average predicted ratings for unseen items $\hat{r}_u$ weighted by recommendation probability, which we found to slightly improve predictive performance:

$$\hat{\rho}_u = \sum_{x \in \text{holdout}(u)} \hat{r}_{ux} \cdot \text{softmax}_x(\hat{r}_u) \tag{35}$$

  Where $\text{holdout}(u)$ is the set of unseen items corresponding to user u, $\hat{r}_{ux} \in [1, 5]$ are the predicted ratings, and $\hat{r}_u \in [1, 5]^{|\text{holdout}(u)|}$ is the vector of all predicted ratings used for softmax recommendation as described in appendix E. We chose to focus on linear models since the treatment datasets are relatively small (each $|\mathcal{S}^{(j)}| \approx 500$), and since other model classes (including boosted trees and MLPs) did not perform significantly better.
- **Engagement dynamics**: I Interaction sequences for each user were generated according to the interaction dynamics described in section 3. We denote this process by $\mathcal{S}^{\text{LV}}(p; u)$, and describe it in detail in the next section. Latent sates were initialized randomly with relative uniform noise around the theoretical LV equilibrium point $(\lambda_0, q_0) = ((1 + \xi_\lambda)\lambda^*, (1 + \xi_q)q^*)$, where $\xi_\lambda, \xi_q \sim \text{Uniform}(-0.1, 0.1)$. Latent states were updated each $B = 10$ recommendations to stabilize noise (see fig. 7). When x is recommended to u at time t, latent states and Δt are set according to $\beta_u(t)$, which depends on ratings r_{ux} (true or mixed with predictions $u^\top x$ via κ). Specifically, we use $\beta_u(t) = r_{ux}^2/100 \in \{0.01, 0.04, 0.09, 0.16, 0.25\}$, which is convex, to accentuate the role of low ratings since they are underrepresented in the data. For $B \geq 1$, we take the effective

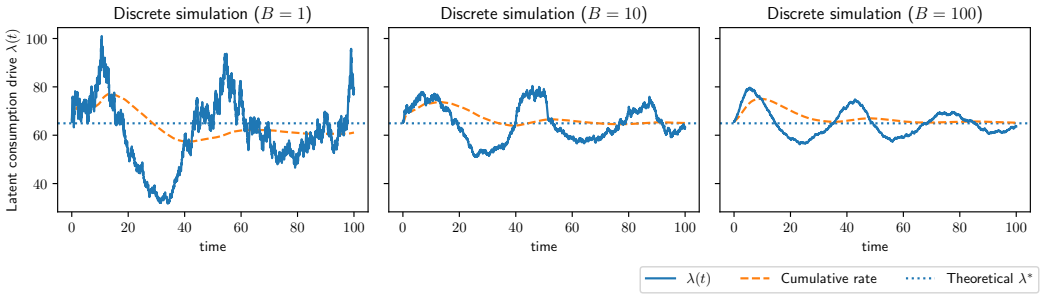

Figure 7: Example discrete sequence $S_u \sim \mathcal{S}^{\text{LV}}(p; u)$, for varying batch sizes. $\mathcal{S}^{\text{LV}}$ captures the general properties of our proposed behavioral model: note how cumulative averaging behavior (orange dashes) exhibits 'habit formation', which our equilibrium approach targets (blue dots). For the same initial conditions $\lambda(0), z(0)$, the figure shows how varying the number of recommended items per step (B) 'smooths' the discrete behavior (left: $B = 1$, center: $B = 10$, right: $B = 100$). As B is increased, $\mathcal{S}^{\text{LV}}$ sequences approach a continuous LV trajectory; in general, and particularly when $\beta_u(t)$ varies by step and per recommended items—this is not the case.

$\beta_u(t)$ to be the average over the B items recommended in that step. We set $\alpha = 0.065$, and chose $\gamma = 0.02, \delta = 0.001$ (which together determine scale) so that typical values for engagement rate $\frac{1}{T}|S_u|$ are on the order of ≈ 10 for the chosen $T = 100$.

I.5 Discrete TPP for Lokta-Volterra simulation

The Temporal Point Process (TPP) we use for simulating user interaction sequences S_u is based on a discretization of the LV system described in Eq. (10), using the forward Euler method with variable step sizes. We denote this process by $\mathcal{S}^{\text{LV}}(u; T)$, and present the sampling procedure in algorithm 1.

Each user is associated with discrete latent states λ_i, q_i, and parameters $\alpha_u, \gamma_u, \delta_u$. Initial states λ_0, q_0 are set randomly. At each step, and in time t_i, the system recommends $x_i = x(t_i)$, which triggers updates in latent states, and determines the next time of interaction t_{i+1}. As noted, these update depend on item-specific parameters β_{u,x_i}.

Under stationary policy $\pi(p)$, the system recommends an item with probability $(1-p)$, and suggests a break with probability p. The simulator considers B recommendation opportunities at each step. For each $k \in \{1, \ldots, B\}$, denote by $I_k \in \{0, 1\}$ the break indicator, equal to 0 when a break is recommended at the k-th slot in the batch. Denote by $x \sim \psi$ the item recommended by the underlying policy ψ, and by $\beta(x)$ the corresponding LV hyperparameter as defined above.

I.6 Goodreads evaluation results

fig. 8 shows results for the Goodreads experiment, in the same format as fig. 5. Results exhibit performance and trends that are qualitatively similar to the MovieLens experiment in appendix E. The LV policy optimization method achieved better performance on this dataset (right pane): The performance of the `LV` is closer to `oracle` (-0.541% in Goodreads compared to -0.797% in ML1M), and the gap from the `best-of` method is slightly larger (+5.14% in Goodreads compared to +2.05% in ML1M). Varying user types (center pane) shows less variation across values of κ, indicating that predictors achieve satisfactory performance even when the κ is low and prediction is harder (see appendix E.1). Varying treatments (right pane) also coincides with the observations: LV policy optimization on the Goodreads dataset exhibits less performance degradation as $p_1 \to 0$, suggesting that less data may be sufficient for optimization. We attribute these results to the larger size of the dataset (approximately 3.6M interactions, compared to 1M interactions in ML1M), and to the possibility that a stronger structure may exist in this recommendation modality compared to general movie recommendation.

Algorithm 1 Sample from $\mathcal{S}^{\mathrm{LV}}(p; u)$

Output: $y = x^n$
Input: Break probability $p \in [0, 1]$
  Stationary content recommendation policy π_0
  Lotka-Volterra parameters $\theta_u = (\alpha, \beta, \gamma, \delta)$
  Time horizon $T > 0$
Output: Interaction sequence $S_u \sim \mathcal{S}^{\mathrm{LV}}(\psi(p) \circ \pi_0; \theta_u)$
  $i \leftarrow 0$
  $t_0 \leftarrow 0$
  $S_u \leftarrow \{\}$
  while $t_i < T$ **do**
    for all $k \in \{1, \ldots, B\}$ **do**
      $I_k \sim \mathrm{Bernoulli}(1 - p)$
      $x_k \sim \psi$
      $\beta_k \leftarrow \beta(r_{u,x_k})$
    end for
    $\Delta t_i \leftarrow \lambda_i^{-1}$
    $\lambda_{i+1} \leftarrow \lambda_i \left(1 - \alpha + \frac{\sum_{k=1}^{B} I_k \beta_k}{B} q_i\right)$
    $q_{i+1} \leftarrow q_i \left(\gamma(1 - q_i) - \frac{\sum_{k=1}^{B} I_k \delta}{B} \lambda_i\right)$
    $t_{i+1} \leftarrow t_i + \Delta t_i$
    $S_u \leftarrow S_u \cup \{(t_i, (x_1, \ldots, x_B), (I_1, \ldots, I_B))\}$
    $i \leftarrow i + 1$
  end while

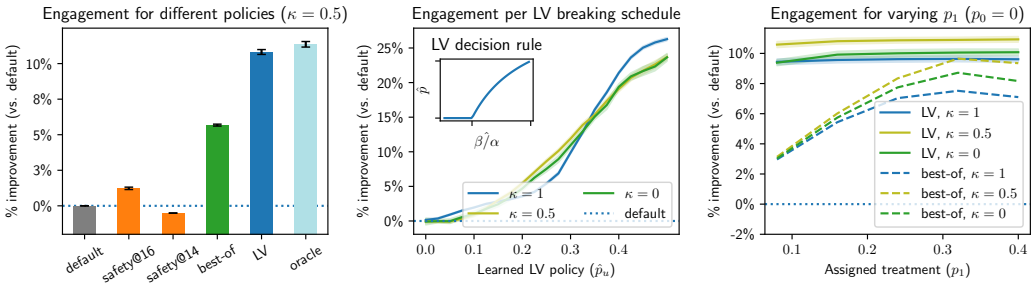

Figure 8: Results on the Goodreads comic-books dataset (compare to Figure 5). **(Left)** Performance gain of different approaches (relative to `default` policy). **(Center)** Performance of `LV` by user group, partitioned by learned policies $\hat{p}_u$. **(Right)** Sensitivity to an increasingly aggressive experimental p_1 ($N = 2, p_0 = 0$). See appendix I.6 for analysis of resutls.

J ADDITIONAL EMPIRICAL EVALUATION

In this section, we provide additional empirical evidence:

- appendix J.1 presents an example use case where richer feedback can be utilized. The experiment shows that utilizing additional feedback improves the breaking policy and increases engagement. We demonstrate how to leverage additional problem structure commonly available in practice, and develop adaptive breaking policies which improve over time.
- In appendix J.2 we evaluate our learning approach under a stateless engagement model. The model is unrelated to Lotka-Volterra and does not promote breaks. The experiment demonstrates how our approach is "safe", in the sense that it does not recommend breaks needlessly, thus extending corollary C.2 beyond the realizable case.

J.1 LEVERAGING ADDITIONAL STRUCTURE USING ADAPTIVE POLICIES

One benefit of our approach in appendix C is that it requires only predictions of long-term engagement. In reality, however, other sources of information may also be available to the learner, which may be useful for improving decisions regarding when to recommend breaks. Here we consider additional information in the form of explicit user feedback, collected by the system over the course of interaction. We model users as providing to the system feedback regarding item quality, namely true ratings r_{ux}, for some of the recommended items x they consume. For simplicity, we assume users report ratings with probability $p_r \in [0, 1]$, independently for each item, where we vary p_r across experimental conditions. Intuitively, we would expect that, over time, user-reported ratings help improve our learned breaking policy; however, over-relying on few data and at and early stage my hinder performance. Here we examine when and how such feedback should be utilized.

Adaptive policy optimization. To extend our approach to incorporate ratings-based feedback, we propose an adaptive explore-then-exploit approach which can leverage sparse explicit feedback. Fixing an *adaptation time* $T_0 \in [0, T]$ (a hyperparameter), the method applies the standard learned breaking policy $\hat{\pi}$ (Eq. (13)) in the time frame $t \leq T_0$, during which it also collects and stores user-reported ratings. Then, at time $t = T_0$, the method updates the learning policy on the basis of the new ratings data, and applies this policy until the eventual time T. In particular, ratings are used to improve the estimated component $\hat{\rho}_u$ in the user feature vector u' (described in Eq. (35)). In this way, the method uses the existing engagement predictors $f_p(u)$ without requiring any retraining.

A formal description of this method is provided in algorithm 2. Here we denote the set of ratings collected until time t by $r_{[0,t]} \in \{1, \ldots, 5\}^*$; the average rating at time t by $\bar{\rho}_{(0,t)} = \frac{1}{|r_{[0,t]}|} \sum_{r \in r_{[0,t]}} r$; and the set of long-term engagement predictions for a feature vector u by $\boldsymbol{f}(u) = \{(p_i, f_{p_i}(u))\}$ (as in appendix C.1).

Algorithm 2 Adaptive policy optimization using sparse rating signals

Input: Initial break probability $p \in [0, 1]$
Time horizon $T > 0$
Adaptation time $T_0 \in [0, T]$
User feature vector $u = (\tilde{v}, b, \hat{\rho})$

1: Collect ratings data $r_{[0,T_0]}$ with breaking policy $\pi(p)$ until time T_0
2: Construct updated feature vector $u' = (\tilde{v}, b, \bar{\rho}_{[0,T_0]})$ using the average rating $\bar{\rho}_{[0,T_0]}$
3: Compute updated long-term engagement predictions $\boldsymbol{f}(u')$
4: Use the LV policy optimization method to obtain updated policy $p' = p^* (\boldsymbol{f}(u'))$
5: Use breaking policy $\pi(p')$ for the remaining time $t \in [T_0, T]$

Evaluation. We evaluate the adaptive policies using the MovieLens-1M experimental setup, as described in appendix E and appendix I. We make use of the same linear regression predictors $f_p(u)$ from the main part of the experiment, and maintain the same time horizon $T = 100$. We vary the ratings density parameter p_r in the range $[0, 1]$, and vary the adaptation time T_0 across three distinct values $T_0 \in \{0.5, 5, 50\}$.

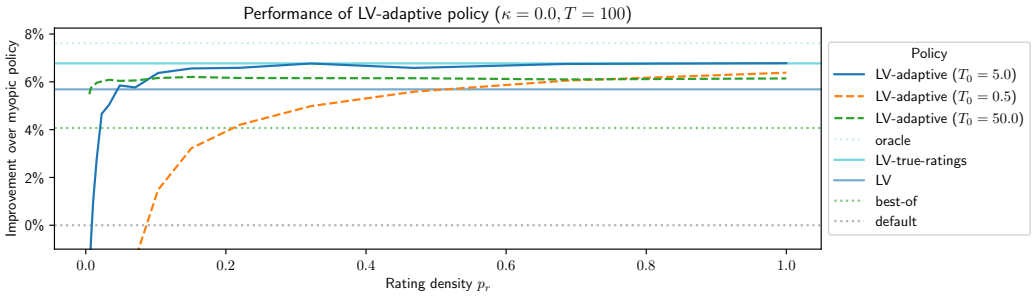

Figure 9: Adaptive policy evaluation for varying rating density $p_r \in [0, 1]$. Solid blue line represents the adaptive policy described in algorithm 2, with adaptation time $T_0 = 5$. Dashed lines represent the performance of adaptive policies under different choices of T_0. Cyan horizontal line represents `LV-true-ratings`, a non-adaptive method with oracle access to the true average rating. The remaining horizontal lines represent the performance of selected non-adaptive policies, as presented in fig. 5 (left pane).

Results. Results are presented in fig. 9. In the results, we observe that `LV-adaptive` with $T_0 = 5$ represents an optimal point within the hyperparameter space. For this choice, results show significant gains over the non-adaptive `LV` method even under moderate rating densities, and quick convergence towards the `LV-true-ratings` upper bound. In contrast, `LV-adaptive` with $T_0 = 0.5$ represents a "premature optimization" scenario, attempting to adapt the policy without allowing the estimated average $\bar{\rho}_{[0,T_0]}$ enough time to converge to its expected value. Similarly, $T_0 = 50$ represents the result of "late optimization", which has accurate estimation of $\bar{\rho}_{[0,T_0]}$ but not enough time to benefit from the updated policy.

J.2 Distinct behavioral model

In this work, we propose the LV model as a behavioral hypothesis class for counterfactual prediction of long-term engagement. As such, using the LV model within the learning-to-break optimization framework is a design choice, to be made at the discretion of the learner; our LV model would be a good choice if it fits the data better than alternative model classes, given the amount of available data. This relation is made precise in our error bound (theorem D.1), which bounds the error when using the LV model for *any* underlying TPP (i.e., we make no assumptions about the true underlying data generation process).

Our main experiments evaluate our approach on data that is not LV dynamics, but nonetheless, bear some resemblance. To complement these results, here we run an additional experiment in which we empirically evaluate our LV-based approach on data that is generated by a behavioral model that is entirely distinct. In particular, we consider a user model in which consumption decisions only depend on the quality of recommended items, without any dependence on internal states—i.e., it is *stateless*. This conforms to behavioral models which are implicitly assumed in conventional recommendation methods such as collaborative filtering.

Stateless content consumption. The data generation process is formalized in algorithm 3. As a means of capturing stateless behavior, we define a "close-range" temporal point process $\mathcal{S}^{\text{CR}}(p; u)$ which generates sequences of user interactions based solely on the average rating of recommended items in a batch. Here we relate rating with utility, and assume that items having higher utility (and therefore higher ratings) induce more frequent interactions; in the same way, we consider break prompts as items having zero utility, and hence zero rating. At time t_i, and given a batch of size B with $k \leq B$ recommended items and $B - k \geq 0$ breaks prompts, users acting according to the stateless behavioral will consume the next batch of content at time:

$$t_{i+1} = t_i + \frac{1}{\tau} \left(\frac{1}{B} \sum_{j=1}^{k} r_{ux_j} \right)^{-1} \tag{36}$$

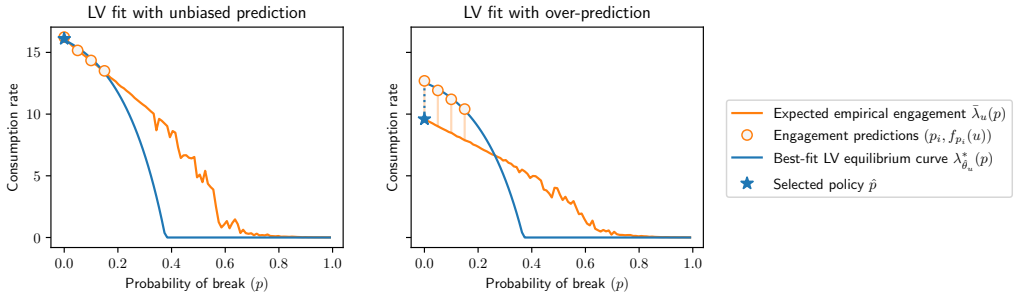

Figure 10: Fitting an LV equilibrium curve on the empirical consumption rates of $\mathcal{S}^{\mathrm{CR}}(p; u)$. The plots are a realization of the schematic diagram in fig. 2 (Right) using data from the MovieLens-1M experiment. **(Left)** Prediction with unbiased engagement predictions. **(Right)** User with extremely low engagement, for which the system tends to over-predict. In both cases, the LV policy optimization method selects the optimal policy $p = 0$.

where $(r_{ux_1}, \ldots, r_{ux_k}) \in \{1, \ldots, 5\}^k$ are the ratings of the items recommended at time k, and $\tau > 0$ is a constant latent parameter to be learned from data. The breaking policy π decides on the number of items k to recommend on each step. Since $r_{ux} \geq 1$ for all user-item pairs, the time difference $\Delta t_i = t_{i+1} - t_i$ in Eq. (36) is minimized by taking $k = B$. This shows that the optimal breaking policy under this behavioral model is the default one, which does not prompt the user to break.

Algorithm 3 Sample from $\mathcal{S}^{\mathrm{CR}}(p; u)$

Input: Break probability $p \in [0, 1]$
  Stationary content recommendation probability ψ
  Scalar parameter $\theta_u = \tau > 0$
  Time horizon $T > 0$
Output: Interaction sequence $S_u \sim \mathcal{S}^{\mathrm{CR}}_{\pi(p) \circ \psi}(p; u)$
  $i \leftarrow 0$
  $t_0 \leftarrow 0$
  $S_u \leftarrow \{\}$
  while $t_i < T$ **do**
    for all $k \in \{1, \ldots, B\}$ **do**
      $I_k \sim \mathrm{Bernoulli}(1 - p)$
      $x_k \sim \psi$
    end for
    $t_{i+1} \leftarrow t_i + \frac{1}{\tau} \left(\frac{1}{B} \sum_{k=1}^{B} I_k r_{ux_k} \right)^{-1}$
    $S_u \leftarrow S_u \cup \{(t_i, (x_1, \ldots, x_B), (I_1, \ldots, I_B))\}$
    $i \leftarrow i + 1$
  end while

Evaluation. We evaluate the stateless behavioral model using the MovieLens-1M experimental setup, as described in appendix E. We set $\tau = 4$ for all users, utilize linear regression for engagement prediction, and maintain all hyperparameters without change. Data processing steps are performed as described in appendix I, and the chosen breaking policies are evaluated.

Results. Despite the difference between the true TPP and our choice of model class, the LV policy optimization method successfully learned the optimal no-breaks—which in this case, is the policy $p = 0$ for all users. Further detail is provided by fig. 10, which illustrates the policy optimization steps under $\mathcal{S}^{\mathrm{CR}}(p; u)$ for typical users with unbiased and biased engagement predictions. In both examples, the optimal points of both curves coincide, and the optimal policy is selected despite poor point-wise fit. Combined, these results show that our approach is "safe", in the sense that when breaking is sub-optimal, the learned breaking policy does not override the default policy.