Interaction-Grounded Learning for Recommender Systems

Jessica Maghakian
Stony Brook University (USA)

Kishan Panaganti
Texas A&M University (USA)

Paul Mineiro
Microsoft Research NYC (USA)

Akanksha Saran
Microsoft Research NYC (USA)

Cheng Tan
Microsoft Research NYC (USA)

ABSTRACT

Recommender systems have long grappled with optimizing user satisfaction using only implicit user feedback. Many approaches in the literature rely on complicated feedback modeling and costly user studies. We propose online recommender systems as a candidate for the recently introduced Interaction Grounded Learning (IGL) paradigm. In IGL, a learner attempts to optimize a latent reward in an environment by observing feedback with no grounding. We introduce a novel personalized variant of IGL for recommender systems that can leverage explicit and implicit user feedback to maximize user satisfaction, with no feedback signal modeling and minimal assumptions. With our empirical evaluations that include simulations as well as experiments on real product data, we demonstrate the effectiveness of IGL for recommender systems.

CCS CONCEPTS

• **Information systems** → **Information retrieval**; • **Computing methodologies** → **Machine learning**.

KEYWORDS

recommendation systems, interaction-grounded learning, contextual bandits, reinforcement learning

ACM Reference Format:
Jessica Maghakian, Kishan Panaganti, Paul Mineiro, Akanksha Saran, and Cheng Tan. 2022. Interaction-Grounded Learning for Recommender Systems. In *Proceedings of Workshop on Online Recommender Systems and User Modeling (ORSUM '22).* ACM, New York, NY, USA, 6 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The last decade has seen unprecedented growth in e-commerce, social media and digital streaming offerings, resulting in users that are overwhelmed with content and choices. Online recommender systems offer a way to alleviate this information overload and improve user experience by providing personalized content. Unfortunately, optimizing user satisfaction is challenging because explicit feedback indicating user satisfaction is rare in practice [4]. To resolve the problem of data sparsity, practitioners rely on implicit signals such as clicks [7] or dwell time [25] as a proxy for user satisfaction. However, designing an optimization objective using implicit signals is nontrivial, and many modern recommender systems suffer from the following challenges.

Challenge 1: No one implicit signal is the true user satisfaction signal. User clicks are the most readily available signal, and the Click-Through Rate (CTR) metric has become the gold standard for evaluating the performance of online recommendation systems [18]. Yet there are many instances when a user will interact via clicks and be unsatisfied with the content. The most familiar of these is clickbait, where poor quality content attracts user clicks by exploiting cognitive biases such as caption bias [5], position bias [6] or the curiosity gap [15, 16]. Optimization of the CTR will naturally promote clickbait items that provide negative user experiences and cause distrust in the recommender system [21]. Recent studies show that clicks may even be a signal of user *dissatisfaction.* In laboratory studies of online news reading [11] and Spotify listening sessions [22], half of the clicked on content was actually disliked by users.

Challenge 2: Incorporating multiple implicit feedback signals requires manual feature engineering. In addition to clicks, user implicit feedback can include dwell time [25], mouse movement [8], scroll information [14] and gaze [19]. One popular approach uses dwell time to filter out noisy clicks, with the reasoning that satisfied users stay on pages longer [25]. Although the industry standard is 30+ seconds of dwell time for a "meaningful" click, this number actually varies depending on the page topic, readability and content length [9]. It is equally challenging to incorporate other signals, for example, behaviors such as viewport time, dwell time and scroll patterns have a complicated temporal relationship and represent preference in different phases [11]. There is an extensive body of work on modeling different implicit feedback signals [10, 20], however these niche models may not generalize well across a diverse user base, or stay relevant as recommender systems and their users evolve.

To tackle these challenges, we propose online recommender systems as a candidate for Interaction-Grounded Learning (IGL) [23]. IGL is a learning paradigm where a learner optimizes for latent rewards by interacting with the environment and associating observed feedback with the unobservable true reward. Although IGL was originally inspired by brain-computer interface applications, in this paper we demonstrate that the framework, when utilizing a different generative assumption and augmented with an additional latent state, is also well suited for recommendation applications. Existing approaches such as reinforcement learning and traditional contextual bandits suffer from the choice of reward function. However IGL resolves the 2 above challenges while making minimal assumptions about the value of observed user feedback. Our new approach is able to incorporate both explicit and implicit signals, leverage ambiguous user feedback and adapt to the different ways in which users interact with the system.

ORSUM '22, September 2022, Seattle, WA

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

Our Contributions. We introduce IGL for recommender systems, allowing us to leverage implicit and explicit feedback signals and mitigate the need for reward engineering. We present the first IGL strategy for context-dependent feedback, the first use of inverse kinematics as an IGL objective, and the first IGL strategy for more than two latent states. Using simulations and real production data, we demonstrate that recommender systems require at least 3 reward states, and that IGL is able to address two big challenges for modern online recommender systems.

2 BACKGROUND ON INTERACTION-GROUNDED LEARNING

Problem Statement. Consider a learner that is interacting with an environment while trying to optimize their policies without access to any grounding or explicit reward signal. At each time step, the stationary environment generates a context $x \in \mathcal{X}$ which is sampled i.i.d. from a distribution d_0. The learner observes the context and then selects an action $a \in \mathcal{A}$ from a finite action set. In response, the environment jointly generates a latent reward and feedback vector $(r, y) \in \mathcal{R} \times \mathcal{Y}$ conditional on (x, a). However, the learner is only able to observe y and not r. Since the latent reward can be either deterministic or stochastic, let $R(x, a) := \mathbb{E}_{(x,a)}[r]$ denote the expected reward after choosing action a for context x. In the IGL setting, the context space $\mathcal{X}$ and feedback vector space $\mathcal{Y}$ can be arbitrarily large. Let $\pi \in \Pi : \mathcal{X} \to \Delta(\mathcal{A})$ denote a stochastic policy, with corresponding expected return $V(\pi) := \mathbb{E}_{(x,a)\sim d_0 \times \pi}[r]$. In IGL, the learner's goal is to find the optimal policy $\pi^* = \text{argmax}_{\pi \in \Pi} V(\pi)$, while only able to observe context-action-feedback (x, a, y) triples.

In the recommender system setting, the context x is the user, the action a is the recommended content and the feedback y is the user feedback. Unfortunately existing IGL approaches ([23], [24]) leverage assumptions designed for classification and control tasks which are a poor fit for recommendation scenarios: (i) context-independence of the feedback and (ii) binary latent rewards.

Feedback Dependence Assumptions. It is information theoretically impossible to solve IGL without assumptions about the relation between x, a and y [24]. In the first paper on IGL, the authors assumed full conditional independence of the feedback on the context and chosen action, i.e. $y \perp x, a|r$. For recommender systems, this undesirably implies that all users communicate preferences identically for all content. In the following paper, Xie et al. [24] loosen full conditional independence by considering context conditional independence, i.e. $y \perp x|a, r$. For our setting, this corresponds to the user feedback varying for combinations of preference and content, but remaining consistent across all users. Neither of these two assumptions are applicable in the setting of online content recommendation because different users interact with recommender systems in different ways. This is evidenced by our production data from a real world image recommendation system (see Sec. 4.3) along with existing results in the literature [1, 17]. By assuming user-specific communication rather than item-specific communication, we allow for personalized reward learning.

Number of Latent Reward States. Prior work shows the binary latent reward assumption, along with an assumption that rewards are rare under a known reference policy, is sufficient for IGL to succeed. Specifically, optimizing the contrast between a learned policy and the oblivious uniform policy is able to succeed when feedback is both context and action independent [23]; and optimizing the contrast between the learned policy and all constant-action policies succeeds when the feedback is context independent [24].

Although the binary latent reward assumption (e.g., satisfied or dissatisfied) appears reasonable for recommendation scenarios, it fails to account for user indifference versus user dissatisfaction. This observation was first motivated by our production data, where a 2 state IGL policy would sometimes maximize feedback signals with obviously negative semantics. Assuming users ignore most content most of the time [13], negative feedback can be as difficult to elicit as positive feedback, and a 2 state IGL model is unable to distinguish between these extremes. Hence, we posit a minimal latent state model for recommender systems involves 3 states: (i) $r = 1$, when users are satisfied with the recommended content, (ii) $r = 0$, when users are indifferent or inattentive, and (iii) $r = -1$, when users are dissatisfied.

3 DERIVATIONS

We now address the first of the previously mentioned challenges from Sec. 1. For the recommender system setting, we use the assumption that $y \perp a|x, r$, namely that the feedback y is independent of the displayed content a given the user x and their disposition toward the displayed content $r \in \{-1, 0, 1\}$. Thus, we assume that users may communicate in different ways, but a given user expresses satisfaction, dissatisfaction and indifference in the same way.

The statistical dependence of y on x frustrates the use of learning objectives which utilize the product of marginal distributions over (x, y). Essentially, given arbitrary dependence upon x, learning must operate on each example in isolation without requiring comparison across examples. This motivates attempting to predict the current action from the current context and the currently observed feedback, i.e., inverse kinematics.

3.0.1 Inverse Kinematics. In this section we motivate our inverse kinematics strategy using exact expectations. When acting according to any policy $P(a|x)$, we can imagine trying to predict the action taken given the context and feedback; the posterior distribution is

$$\begin{aligned}
P(a|y,x) &= \frac{P(a|x)P(y|a,x)}{P(y|x)} && \text{(Bayes rule)}\\
&= P(a|x)\sum_r \frac{P(y|r,a,x)}{P(y|x)}P(r|a,x) && \text{(Total Probability)}\\
&= P(a|x)\sum_r \frac{P(y|r,x)}{P(y|x)}P(r|a,x) && (y \perp a|x,r)\\
&= P(a|x)\sum_r \frac{P(r|y,x)}{P(r|x)}P(r|a,x) && \text{(Bayes rule)}\\
&= \sum_r P(r|y,x)\frac{P(r|a,x)P(a,x)}{\sum_a P(r|a,x)P(a|x)}. && \text{(Total Probability)}
\end{aligned} \tag{1}$$

We arrive at an inner product between a reward decoder term $P(r|y, x)$ and a reward predictor term $\frac{P(r|a,x)P(a|x)}{\sum_a P(r|a,x)P(a|x)}$.

3.0.2 Extreme Event Detection. Direct extraction of a reward predictor using maximum likelihood on the action prediction problem with equation (1) is frustrated by two identifiability issues: first, this expression is invariant to a permutation of the rewards on a context dependent basis; and second, the relative scale of two terms being multiplied is not uniquely determined by their product. To mitigate the first issue, we assume $\sum_a P(r = 0|a, x)P(a|x) > \frac{1}{2}$, i.e., nonzero rewards are rare under $P(a|x)$; and to mitigate the second issue, we assume the feedback can be perfectly decoded, i.e., $P(r|y, x) \in \{0, 1\}$. Under these assumptions we have

$$r = 0 \implies P(a|y, x) = \frac{P(r = 0|a, x)P(a|x)}{\sum_a P(r = 0|a, x)P(a|x)} \leq 2P(r = 0|a, x)P(a|x) \leq 2P(a|x). \quad (2)$$

Equation (2) forms the basis for our extreme event detector: anytime the posterior probability of an action is predicted to be more than twice the prior probability, we deduce $r \neq 0$.

Note a feedback merely being apriori rare or frequent (i.e., the magnitude of $P(y|x)$ under the policy $P(a|x)$) does not imply that observing such feedback will induce an extreme event detection; rather the feedback must have a probability that strongly depends upon which action is taken. Because feedback is assumed conditionally independent of action, the only way for feedback to help predict which action is played is via the (action dependence of the) latent reward.

3.0.3 Extreme Event Disambiguation. With 2 latent states, $r \neq 0 \implies r = 1$, and we can reduce to a standard contextual bandit with inferred rewards $\mathbb{1}(P(a|y, x) > 2P(a|x))$. With 3 latent states, $r \neq 0 \implies r = \pm 1$, and additional information is necessary to disambiguate the extreme events. We assume partial reward information is available via a "definitely negative" function $\mathrm{dn} : \mathcal{X} \times \mathcal{Y} \to \{-1, 0\}$ where $P(\mathrm{dn}(x, y) = 0|r = 1) = 1$ and $P(\mathrm{dn}(x, y) = -1|r = -1) > 0$. This reduces extreme event disambiguation to one-sided learning [2] applied only to extreme events, where we try to predict the underlying latent state given (x, a). We assume partial labelling is selected completely at random [3] and treat the (constant) negative labelling propensity α as a hyperparameter. We arrive at our 3-state reward extractor

$$\rho(x, a, y) = \begin{cases} 0 & P(a|y, x) \leq 2P(a|x) \\ -1 & P(a|y, x) > 2P(a|x) \text{ and } \mathrm{dn}(x, y) = -1 \\ \alpha & \text{otherwise} \end{cases}, \quad (3)$$

equivalent to Zhang and Lee [26, Equation 11] scaled by α. Note setting $\alpha = 1$ embeds 2-state IGL.

3.0.4 Implementation Notes. In practice, $P(a|x)$ is known but the other probabilities are estimated. $\hat{P}(a|y, x)$ is estimated online using maximum likelihood on the problem predicting a from (x, y), i.e., on a data stream of tuples $((x, y), a)$. The current estimates induce $\hat{\rho}(x, a, y)$ based upon the plug-in version of equation (3). In this manner, the original data stream of (x, a, y) tuples is transformed into stream of $(x, a, \hat{r} = \hat{\rho}(x, a, y))$ tuples and reduced to a standard online contextual bandit problem.

As an additional complication, although $P(a|x)$ is known, it is typically a good policy under which rewards are not rare (e.g., offline learning with a good historical policy; or acting online according to the policy being learned by the IGL procedure). Therefore we use importance weighting to synthesize a uniform action distribution $P(a|x)$ from the true action distribution.[1] Ultimately we arrive at the procedure of Algorithm 1.

Algorithm 1 IGL, Inverse Kinematics and either 2 or 3 Latent States.

Input: Contextual bandit algorithm `CB-Alg`.
Input: Calibrated weighted multiclass classification algorithm `MC-Alg`.
Input: Definitely negative oracle `DN`. # `DN(...)` = 0 for 2 state IGL
Input: Negative labelling propensity α. # $\alpha = 1$ for 2 state IGL
Input: Action set size K.

$\pi \leftarrow$ new `CB-Alg`.
`IK` $\leftarrow$ new `MC-Alg`.
for $t = 1, 2, \ldots;$ **do**
  Observe context x_t and action set A_t with $|A_t| = K$.
  if On-policy IGL **then**
    $P(\cdot|x_t) \leftarrow \pi$.predict$(x_t, A_t)$. # Compute action distribution
    Play $a_t \sim P(\cdot|x_t)$ and observe feedback y_t.
  else
    Observe $(x_t, a_t, y_t, P(\cdot|x_t))$.
  $w_t \leftarrow 1/(KP(a_t|x_t))$. # Synthetic uniform distribution
  $\hat{P}(a_t|y_t, x_t) \leftarrow$ `IK`.predict$((x_t, y_t), A_t, a_t)$. # Predict action probability
  if $K\hat{P}(a_t|y_t, x_t) \leq 2$ **then** # $\hat{r}_t = 0$
    π.learn$(x_t, a_t, A_t, r_t = 0, w_t)$
  else # $\hat{r}_t \neq 0$
    if `DN(...)` = 0 **then**
      π.learn$(x_t, a_t, A_t, r_t = \alpha, w_t)$
    else # Definitely negative
      π.learn$(x_t, a_t, A_t, r_t = -1, w_t)$
  `IK`.learn$((x_t, y_t), A_t, a_t, w_t)$.

4 EMPIRICAL EVALUATIONS

Due to the sensitivity around production metrics and customer segments, most experiments demonstrate qualitative effects via simulation, with simulator properties inspired by production observations. Our final experiment (Sec. 4.3) includes relative performance data from a production real-world image recommendation scenario.

Abbreviations. Algorithms are denoted by the following abbreviations: Personalized IGL for 2 latent states (`IGL-P(2)`); Personalized IGL for 3 latent states (`IGL-P(3)`).

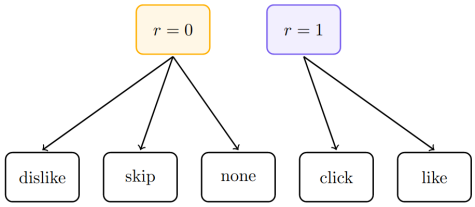
General Evaluation Setup. At each time step t, the context x_t is provided from either the simulator (Sec. 4.1-4.2) or the logged production data (Sec. 4.3). The learner then selects an action a_t and receives feedback y_t. In these evaluations, each user provides feedback in exactly one interaction and different user feedback signals are mutually exclusive, so that y_t is a one-hot vector. In simulated environments, the ground truth reward is sometimes used for evaluation but never revealed to the algorithm.

[1] When the number of actions is changing from round to round, we use importance weighting to synthesize a non-uniform action distribution with low rewards, but we elide this detail for ease of exposition.

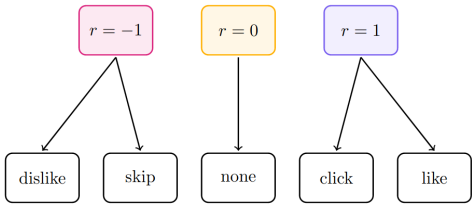
Simulator Design. Before the start of each experiment, user profiles with fixed latent rewards for each action are generated. The users are also assigned predetermined communication styles, so the probability of emitting a given signal conditioned on the latent reward remains static throughout the duration of the experiment. For the available feedback, users can provide feedback using five signals: (1) like, (2) dislike, (3) click, (4) skip and (5) none. The feedback includes a mix of explicit (likes, dislikes) and implicit (clicks, skips, none) signals. Despite receiving no human input on the assumed meaning of the implicit signals, we will demonstrate that IGL can determine which feedback are associated with which latent state. In addition to policy optimization, IGL can also be a tool for automated feature discovery. To reveal the qualitative properties of the approach, the simulated probabilities for observing a particular feedback given the reward are chosen so that they can be perfectly decoded, i.e., each feedback has a nonzero emission probability in exactly one latent reward state. Production data does not obey this constraint (e.g., accidental emissions of all feedback occur at some rate): theoretical analysis of our approach without perfectly decodable rewards is a topic for future work.

4.1 Motivating the 3 State Model for Recommender Systems

We now implement Algorithm 1 for 2 latent states as `IGL-P(2)`. The experiment here shows the following two results about `IGL-P(2)`: (i) it is able to succeed in the scenario when there are 2 underlying latent rewards and (ii) it can no longer do so when there are 3 latent states. Fig. 1 shows the simulator setup used, where clicks and likes are used to communicate satisfaction, and dislikes, skips and no feedback (none) convey (active or passive) dissatisfaction.



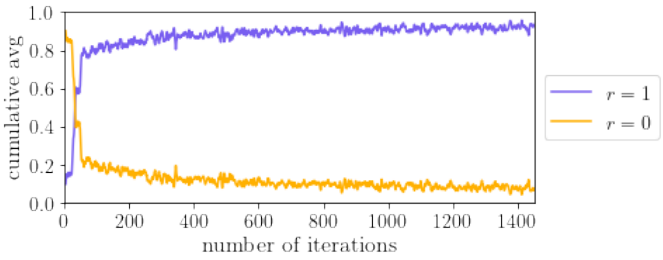
(a) 2 latent state model



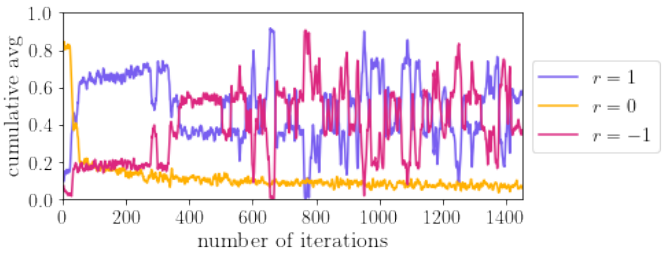
(b) 3 latent state model

Figure 1: Simulator settings for 2 state and 3 state latent model. In Fig. 1a, r = 0 corresponds to anything other than the user actively enjoying the content, whereas in Fig. 1b, lack of user enjoyment is split into indifference and active dissatisfaction.

Fig. 2 shows the distribution of rewards for `IGL-P(2)` as a function of the number of iterations, for both the 2 and 3 latent state model. When there are only 2 latent rewards, `IGL-P(2)` consistently improves; however for 3 latent states, `IGL-P(3)` oscillates between $r = 1$ and $r = -1$, resulting in much lower average user satisfaction. The empirical results demonstrate that although `IGL-P(2)` can successfully identify and maximize the rare feedbacks it encounters, it is unable to distinguish between satisfied and dissatisfied users.



(a) Two latent states



(b) Three latent states

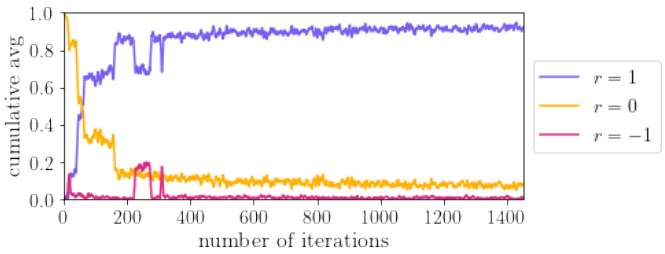
Figure 2: Performance of `IGL-P(2)` in simulated environment. Although `IGL-P(2)` is successful with the 2 state simulator, it fails on the 3 state simulator and oscillates between attempting to maximize r = 1 and r = −1.

4.2 `IGL-P(3)`: Personalized Reward Learning for Recommendations

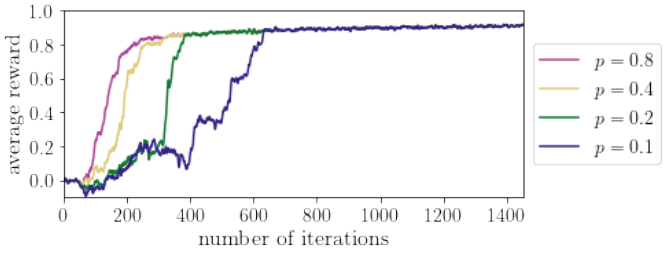
Since `IGL-P(2)` is not sufficient for the recommendation system setting, we now explore the performance of `IGL-P(3)`. Using the same simulator as Fig. 1b, we evaluated `IGL-P(3)`. Fig. 3a demonstrates the distribution of the rewards over the course of the experiment. `IGL-P(3)` quickly converged, and because of the partial negative feedback for dislikes, never attempted to maximize the $r = -1$ state. Even though users used the ambiguous skip signal to express dissatisfaction 80% of the time, `IGL-P(3)` was still able to learn user preferences.

In order for `IGL-P(3)` to succeed, the algorithm requires direct grounding from the dislike signal. We next examined how `IGL-P(3)` is impacted by increased or decreased presence of user dislikes. Fig. 3b was generated by varying the probability p of users emitting dislikes given $r = -1$, and then averaging over 10 experiments for each choice of p. While lower dislike emission probabilities are associated with slower convergence, `IGL-P(3)` is able to overcome the increase in unlabeled feedback and learn to associate the

skip signal with user dissatifaction. Once the feedback decoding stabilizes, regardless of the dislike emission probability, IGL-P(3) enjoys strong performance for the remainder of the experiment.



(a) Ground truth learning curves, P(dislike|r = −1) = 0.2.



(b) Effect of varying P(dislike|r = −1).

Figure 3: Performance of IGL-P(3) in simulated environment. In Fig. 3a, IGL-P(3) successfully maximizes user satisfaction while minimizing dissatisfaction. Fig. 3b demonstrates how IGL-P(3) is robust to varying the frequency of partial information received, although more data is needed for convergence when "definitely bad" events are less frequent.

4.3 Production Results

Our production setting is a real world image recommendation system that serves hundreds of millions of users. In our recommendation system interface, users provide feedback in the form of clicks, likes, dislikes or no feedback. All four signals are mutually exclusive and the user only provides one feedback after each interaction. For these experiments, we use data that spans millions of interactions over a period of days. The current policy implemented in practice is a CB algorithm that utilizes a hand-engineered reward function. The production policy achieves both more click and like feedback than directly optimizing for the number of clicks or directly optimizing for the number of likes. As a result, *any improvements over the production policy imply improvement over any bandit algorithm for click feedback.*

We implement IGL-P(2) and IGL-P(3) and report the performance as relative lift metrics over the production baseline. Unlike the simulation setting, we no longer have access to the user's latent reward after each interaction. As a result, we evaluate the performance of the novel IGL implementations through the implicit and explicit feedback signals. An increase in both clicks and likes, and a decrease in dislikes, are considered desirable outcomes. Table 1 shows the results of our empirical study.

In the simulations, IGL-P(2) exhibited a failure mode of reliable identification of extreme events, with an inability to avoid extreme *negative* events. Our production data shows a similar pathology, where IGL-P(2) receives dramatically more dislikes, at the expense of likes. Although the true latent state is unknown, IGL-P(2) achieved worse performance on explicit feedback signals, directly implying that users had fewer positive interactions and significantly more negative interactions. These results provide evidence for the >2 latent state model in real world recommendation systems.

Although we established that users have more than two latent states, it might not be the case that 3 states is sufficient to capture the recommendation system setting. Our evaluation of IGL-P(3) on our data however, provides evidence that 3 states are enough, and that IGL is able to succeed with the context dependent assumptions. IGL-P(3) was able to achieve performance comparable to the production baseline, with a strong directional improvement in total clicks. This is a notable achievement, because the baseline deployed in production uses a meticulously tuned, hand-engineered reward function generated from an order of magnitude more historical data.

5 DISCUSSION

We presented IGL for recommender systems, an approach to producing personalized recommendations that can leverage rich and diverse types of user feedback signals. In this paper, we showed that IGL can elegantly sidestep complicated manual reward engineering and effectively learn how to maximize user satisfaction with minimal human input. We considered 5 feedback signals in this work, but IGL can easily be scaled to incorporate many more signals with little computational cost.

To complete this work, we want to theoretically investigate the approach presented here in two key directions: first, characterizing finite-sample behaviour; and second, relaxing the assumption of perfectly decodable reward.

One of the open challenges for IGL is developing effective ways of evaluating its performance given the lack of true grounding, especially in situations where explicit user feedback might not be available at all. We speculate that, due to both personalization and the "rewards are rare" prior, the latent reward inferred by IGL could prove superior in casually predicting longitudinal outcomes relative to raw feedback statistics. Because longitudinal outcomes can have facially obvious semantics (e.g., subscription renewals) this could provide an alternative grounding for evaluating IGL.

Another promising future direction is IGL for fair recommender systems. Modern systems optimize for set objectives, often marginalizing user subpopulations that interact with recommender systems in different ways [12]. Since context dependent IGL allows for personalized reward learning, it has the potential to perform consistently and fairly across diverse subgroups of users.

ACKNOWLEDGMENTS

This work is partially supported by National Science Foundation under Grant No. 1650114 (https://www.nsfgrfp.org/).

REFERENCES

[1] Joeran Beel, Stefan Langer, Andreas Nürnberger, and Marcel Genzmehr. 2013. The impact of demographics (age and gender) and other user-characteristics

Algorithm	Clicks	Likes	Dislikes
IGL-P(3)	[0.999, 1.067, 1.152]	[0.985, 1.029, 1.054]	[0.751, 1.072, 1.274]
IGL-P(2)	[0.926, 1.005, 1.091]	[0.914, 0.949, 0.988]	[1.141, 1.337, 1.557]

Table 1: Relative metrics lift over a production baseline. The production baseline uses a hand-engineered reward function which is not available to IGL algorithms. Shown are point estimates and associated bootstrap 95% confidence regions. IGL-P(2) erroneously increases dislikes to the detriment of other metrics. IGL-P(3) directionally improves over the hand-engineered reward function.

on evaluating recommender systems. In *International Conference on Theory and Practice of Digital Libraries*. Springer, 396–400.
[2] Jessa Bekker and Jesse Davis. 2020. Learning from positive and unlabeled data: A survey. *Machine Learning* 109, 4 (2020), 719–760.
[3] Charles Elkan and Keith Noto. 2008. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 213–220.
[4] Miha Grčar, Dunja Mladenič, Blaž Fortuna, and Marko Grobelnik. 2005. Data sparsity issues in the collaborative filtering framework. In *International workshop on knowledge discovery on the web*. Springer, 58–76.
[5] Katja Hofmann, Fritz Behr, and Filip Radlinski. 2012. On caption bias in interleaving experiments. In *Proceedings of the 21st ACM international conference on Information and knowledge management*. 115–124.
[6] Katja Hofmann, Anne Schuth, Alejandro Bellogin, and Maarten de Rijke. 2014. Effects of position bias on click-based recommender evaluation. In *European Conference on Information Retrieval*. Springer, 624–630.
[7] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE international conference on data mining*. Ieee, 263–272.
[8] MV Ishwarya, G Swetha, S Saptha Maaleekaa, and R Anu Grahaa. 2019. Efficient Recommender System by Implicit Emotion Prediction. In *Advances in Big Data and Cloud Computing*. Springer, 173–178.
[9] Youngho Kim, Ahmed Hassan, Ryen W White, and Imed Zitouni. 2014. Modeling dwell time to predict click-level satisfaction. In *Proceedings of the 7th ACM international conference on Web search and data mining*. 193–202.
[10] Zhenhua Liang, Siqi Huang, Xueqing Huang, Rui Cao, and Weize Yu. 2020. Post-click behaviors enhanced recommendation system. In *2020 IEEE 21st International Conference on Information Reuse and Integration for Data Science (IRI)*. IEEE, 128–135.
[11] Hongyu Lu, Min Zhang, and Shaoping Ma. 2018. Between clicks and satisfaction: Study on multi-phase user preferences and satisfaction for online news reading. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 435–444.
[12] Nicola Neophytou, Bhaskar Mitra, and Catherine Stinson. 2022. Revisiting popularity and demographic biases in recommender evaluation and effectiveness. In *European Conference on Information Retrieval*. Springer, 641–654.
[13] Tien T Nguyen, Pik-Mai Hui, F Maxwell Harper, Loren Terveen, and Joseph A Konstan. 2014. Exploring the filter bubble: the effect of using recommender systems on content diversity. In *Proceedings of the 23rd international conference on World wide web*. 677–686.
[14] Ladislav Peška and Peter Vojtáš. 2012. Estimating importance of implicit factors in e-commerce recommender systems. In *Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics*. 1–4.
[15] Martin Potthast, Sebastian Köpsel, Benno Stein, and Matthias Hagen. 2016. Clickbait detection. In *European conference on information retrieval*. Springer, 810–817.
[16] Kate Scott. 2021. You won't believe what's in this paper! Clickbait, relevance and the curiosity gap. *Journal of pragmatics* 175 (2021), 53–66.
[17] Donghee Shin. 2020. How do users interact with algorithm recommender systems? The interaction of users, algorithms, and performance. *Computers in Human Behavior* 109 (2020), 106344.
[18] Thiago Silveira, Min Zhang, Xiao Lin, Yiqun Liu, and Shaoping Ma. 2019. How good your recommender system is? A survey on evaluations in recommendation. *International Journal of Machine Learning and Cybernetics* 10, 5 (2019), 813–831.
[19] Ekta Sood, Simon Tannert, Diego Frassinelli, Andreas Bulling, and Ngoc Thang Vu. 2020. Interpreting attention models with human visual attention in machine reading comprehension. *arXiv preprint arXiv:2010.06396* (2020).
[20] Wenjie Wang, Fuli Feng, Xiangnan He, Liqiang Nie, and Tat-Seng Chua. 2021. Denoising implicit feedback for recommendation. In *Proceedings of the 14th ACM international conference on web search and data mining*. 373–381.
[21] Wenjie Wang, Fuli Feng, Xiangnan He, Hanwang Zhang, and Tat-Seng Chua. 2021. Clicks can be cheating: Counterfactual recommendation for mitigating clickbait issue. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1288–1297.
[22] Hongyi Wen, Longqi Yang, and Deborah Estrin. 2019. Leveraging post-click feedback for content recommendations. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 278–286.
[23] Tengyang Xie, John Langford, Paul Mineiro, and Ida Momennejad. 2021. Interaction-Grounded Learning. In *International Conference on Machine Learning*. PMLR, 11414–11423.
[24] Tengyang Xie, Akanksha Saran, Dylan J Foster, Lekan Molu, Ida Momennejad, Nan Jiang, Paul Mineiro, and John Langford. 2022. Interaction-Grounded Learning with Action-inclusive Feedback. *arXiv preprint arXiv:2206.08364* (2022).
[25] Xing Yi, Liangjie Hong, Erheng Zhong, Nanthan Nan Liu, and Suju Rajan. 2014. Beyond clicks: dwell time for personalization. In *Proceedings of the 8th ACM Conference on Recommender systems*. 113–120.
[26] Dell Zhang and Wee Sun Lee. 2005. A simple probabilistic approach to learning from positive and unlabeled examples. In *Proceedings of the 5th annual UK workshop on computational intelligence (UKCI)*. 83–87.