Short-Term Satisfaction and Long-Term Coverage: Understanding How Users Tolerate Algorithmic Exploration

Tobias Schnabel
Cornell University
Ithaca, NY, USA
tbs49@cornell.edu

Paul N. Bennett,
Susan T. Dumais
Microsoft Research
Redmond, WA, USA
[pauben,sdumais]@microsoft.com

Thorsten Joachims
Cornell University
Ithaca, NY, USA
tj@cs.cornell.edu

ABSTRACT

Any learning algorithm for recommendation faces a fundamental trade-off between exploiting partial knowledge of a user's interests to maximize satisfaction in the short term and discovering additional user interests to maximize satisfaction in the long term. To enable discovery, a machine learning algorithm typically elicits feedback on items it is uncertain about, which is termed *algorithmic exploration* in machine learning. This exploration comes with a cost to the user, since the items an algorithm chooses for exploration frequently turn out to not match the user's interests. In this paper, we study how users tolerate such exploration and how presentation strategies can mitigate the exploration cost. To this end, we conduct a behavioral study with over 600 people, where we vary how algorithmic exploration is mixed into the set of recommendations. We find that users respond non-linearly to the amount of exploration, where some exploration mixed into the set of recommendations has little effect on short-term satisfaction and behavior. For long-term satisfaction, the overall goal is to learn via exploration about the items presented. We therefore also analyze the quantity and quality of implicit feedback signals such as clicks and hovers, and how they vary with different amounts of mix-in exploration. Our findings provide insights into how to design presentation strategies for algorithmic exploration in interactive recommender systems, mitigating the short-term costs of algorithmic exploration while aiming to elicit informative feedback data for learning.

CCS CONCEPTS

• **Information systems** → **Recommender systems**; • **Human-centered computing** → *Interaction design theory, concepts and paradigms*;

KEYWORDS

recommender systems; algorithmic exploration; user experience; interactive systems; human-in-the-loop

WSDM'18, , February 5–9, 2018, Marina Del Rey, CA, USA

ACM ISBN 978-1-4503-5581-0/18/02…$15.00
https://doi.org/10.1145/3159652.3159700

ACM Reference format:
Tobias Schnabel, Paul N. Bennett, Susan T. Dumais, and Thorsten Joachims. 2018. Short-Term Satisfaction and Long-Term Coverage: Understanding How Users Tolerate Algorithmic Exploration. In *Proceedings of WSDM'18, February 5–9, 2018, Marina Del Rey, CA, USA, ,* 9 pages.
https://doi.org/10.1145/3159652.3159700

1 INTRODUCTION

Recommender systems are an integral component of many websites including e-commerce destinations, news content hubs, and movie streaming portals. Recommender systems help people navigate a large space of options [33] and can both increase user satisfaction as well as achieve other business goals [16]. For the continuous improvement of recommender systems, *exploration* forms a key component of how learning algorithms obtain coverage of a user's possible interests. More specifically, algorithmic exploration describes a concept in *online learning* [35], where algorithms learn sequentially, and need to try out new actions or options – explore them – to achieve good coverage of the learning domain in the long run. Algorithmic exploration not only helps with respect to a single user, but can also improve other users' recommendations, for example, through exploration of newly listed items [1, 2]. Moreover, the feedback collected through exploration can be used to counterfactually evaluate new recommender systems [4, 28, 39].

Although exploration is of great importance for many algorithmic goals, exploration has the potential to lead to user dissatisfaction, since the items chosen for exploration frequently turn out to not match the user's interests. This raises the following key question: what is the overall negative impact of exploration on user satisfaction and how can it be mitigated while ensuring a high level of quality and quantity of the resulting feedback data?

In this paper, we study the question of how to best explore from two perspectives. First, we study in a systematic way how the amount of exploration affects short-term user satisfaction. In particular, a certain number of items to explore are mixed into each impression of personalized recommendations – a process we call *mix-in exploration*. This allows us to examine different levels of exploration among the set of recommendations, starting with a base strategy that does no exploration and then gradually moving to a strategy that recommends items purely for the purpose of exploration. By conducting a user study on Amazon Mechanical Turk with over 600 participants, we find that people are able to perceive changes from different levels of exploration, but that changes affect user-reported measures in a non-linear way. In particular, we find that mixing only a few exploratory items into each personalized recommendation impression does not change reported

user satisfaction significantly. However, exceeding a threshold of exploration then sharply impacts user perception. Surprisingly, we could not find any significant impact of exploration on peoples' final choices, indicating that people are quite resilient to irrelevant recommendations and that they find ways to work around them.

Second, as a step towards learning, we examine quantity and quality of feedback signals under different levels of exploration. We find that larger amounts of exploration drives down feedback quantity. Overall, we find that among the signals we studied both shortlisting events as well as hovering events provide the most informative feedback signals.

In summary, we find that people are tolerant with respect to smaller amounts of exploration, and that limiting exploration is also important for ensuring good feedback quantities. These findings indicate that for improving recommendation systems in practice, it is preferable to mix a limited amount of exploration into every impression – as opposed to having a few impressions that do pure exploration. Also, our study of feedback signals can further guide the design of learning algorithms. To the best of our knowledge, this is the first study to measure the user-centric cost of exploration in a personalized setting.

2 RELATED WORK

Our work is related to ideas from various areas, such as user-centric evaluation of recommender systems, understanding of implicit feedback, and interactive recommender systems.

We mainly follow the user-centric paradigm [26, 30, 36] for the evaluation of recommender systems in this paper. We divide user-centric evaluation into rating-based and task-based studies. Studies in the first category present a list of recommendations to users that they need to rate, whereas studies in the second category ask people to actually use a recommender system as part of a user task. The study in our paper falls into the second category.

Cosley et al. [7] conducted one of the first rating-based studies in recommendation, finding that user satisfaction decreases if a recommender shows incorrect star ratings. A study in [8] compared a number of popular recommendation algorithms. The authors found that a static popularity-based method was among the methods with highest satisfaction. Ge et al. [12] mix-in diverse items into recommendation lists. They found that inserting these diverse items did not significantly impact user satisfaction – however, the recommendations were manually created and static. Most rating-based studies, however, showed an increase in user satisfaction with an increase in diversity [10, 44]. The problem with rating-based studies is that they provide no insight into how and whether recommendations would impact users in their tasks [41], a shortcoming that is addressed by task-based studies.

One of the first task-based studies was done by Swearingen and Sinha [40]. They found users to be hesitant to purchase new items and hypothesize that having known items among the recommendations increases the chance of users adopting novel items. We were not able to see the latter effect for random mix-ins in our study. Only a few task-based studies also examine feedback signals along with user-centric measures. Cremonesi et al. [9] found different recommender algorithms to have no effect on metrics like time-to-decision or the number of examined items. Jones and Pu studied user satisfaction in music recommendation [24], and found the number of recommended items that were liked to be predictive of preference between systems, similar to shortlisting interactions in our setting.

There has been substantially more work on understanding implicit feedback signals in information retrieval (IR) [6, 11, 23, 31]. Clicks and dwell time are traditionally used as feedback signals for learning algorithms in these settings. However, we found these signals to be insufficient in quantity in our study, and recommend to consider other signals, e.g., hover time. This issue has also spurred some research in IR on alternative signals, such as cursor movements [20] or scrolling [13].

Lastly, we present a general interface useful for many interactive recommendation settings [17, 38]. In contrast, many interactive systems discussed in the literature [17] provide domain-specific controls for people, e.g., tags associated with a movie [3]. Our interface only relies on items as input, making it applicable to a wide range of domains.

In comparison to related work, ours is the first work to systematically vary the amount of exploration in a personalized recommendation setting. Because of our experimental design, we are able to study the impact of exploration on both user satisfaction and feedback signals.

3 METHODOLOGY

We conducted a randomized between-subjects study with over 600 participants on Amazon Mechanical Turk. This allowed us to collect data from a large and diverse set of people, an advantage over traditional laboratory settings [29]. The user task in this study consisted of picking a movie to watch from a large set of movies. This task is an instance of what is often referred to as the *Find Good Items* task [18], or *One-Choice* task [34]. Using movies gave us a task domain that people were already familiar with; it is also of high practical importance, and has a natural visual representation via movie posters.

We evaluate different presentation strategies through an interactive recommender interface that updates automatically while the user is browsing. This is in contrast to many other studies on user-centric evaluation of recommender studies [8, 10, 12] where users are exposed to recommendations that are only computed once. Having the ability to serve recommendations interactively has the following two benefits: (i) the recommendations are not isolated responses, but can be evaluated with respect to the user task, and (ii) we are able to observe feedback signals directly after new recommendations have been made.

3.1 Interactive recommender interface

The general idea behind our interface is to base recommendations on items that are marked as interesting during a session, similar to e-commerce settings where further items are recommended after initial items have been added to the basket. This setting is also often referred to as *session-based* recommendation [19, 34], as opposed to recommendation based on long-term preferences. To enable interactivity, we update recommendations each time a user adds items to a *shortlist* during a session, that is a temporary list of candidates that the user is currently considering [34].

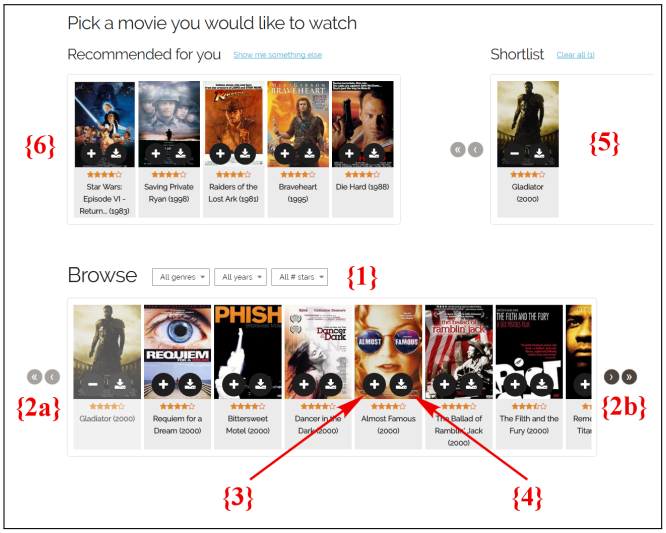

Figure 1: The interactive recommendation interface each user was presented with. The "Recommended for you" panel was updated automatically each time a user added a movie to the shortlist.

As Fig. 1 shows, there are three panels in the interface: the browsing panel (bottom), the shortlist panel {5}, and the recommendation panel {6}. The layout was motivated by the interfaces of common online movie streaming services, such as Netflix, Hulu, or Amazon. In the main panel, people can browse movies by using the paging buttons ({2a}, {2b}), and can also filter the displayed movies using facets {1}. They have the option of examining a movie further by clicking on a poster of the movie, which opens a pop-up with more information. They can add movies to the shortlist panel by clicking on the plus button {3}, and make their final choice by clicking on the download button {4}, and confirming their choice in a prompt afterwards. Every time an item is added to the shortlist, the recommendation panel {6} displays a new set of five recommendations with a brief loading animation. Once a movie is shown as a recommendation, it is blacklisted for the session and never shown again to the user to avoid repetitiveness [43]. We do not update recommendations if a user adds a recommended movie to the shortlist to avoid accidentally dismissing any other interesting recommendations.

3.2 Presentation strategies

In order to systematically test the effects of exploration on people, we created a set of controlled presentation strategies that cover the spectrum from no exploration to full exploration. We used the following strategy for presenting items to explore, which we call *mix-in exploration*: we insert a given number of items to explore from an Explore strategy into the recommendations from a personalized Base strategy. Since there were five presentation slots available, we had six different presentation strategies to cover all cases: Base (B), Mix-1 (M-1), Mix-2 (M-2), Mix-3 (M-3), Mix-4 (M-4), FullExplore (FE).

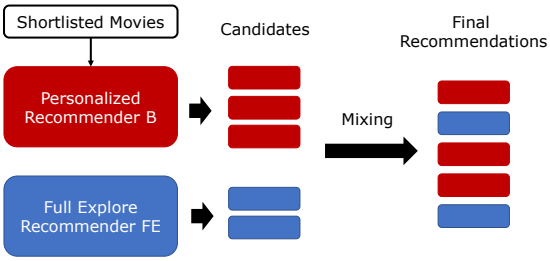

Figure 2: In mix-in exploration, items to be explored are mixed into items from a personalized recommender.

Fig. 2 shows schematically how mix-in exploration is performed in the case of Mix-2. The personalized Base recommender (B) generates a list of candidates based on the set of movies that a user has currently on her shortlist. For the final recommendation impression that is shown to the user, items generated by the FullExplore recommender (FE) are then mixed-into the items of the personalized recommender (B). The details for each strategy are provided below.

Base. This personalized content-based strategy finds the 50 most similar movies for each movie in the shortlist, and re-ranks all candidates by popularity for familiarity. We also explored collaborative filtering techniques, but found them to perform poorly for the small sample sizes available to us. To compute the content-based similarities, we project all tag vectors from the Tag Genome project [42] into a 25-dimensional space via Singular Value Decomposition for efficiency and use cosine similarities between those vectors. Popularity here is defined as the number of ratings of three or more stars for a movie in the MovieLens 1M dataset [15]. Ranking the candidates by popularity showed improved user satisfaction in our preliminary experiments over a method which did not re-rank movies by popularity, a finding that is consistent with other studies [8, 22].

FullExplore. Returns items to explore uniformly at random from the inventory. We chose this method partly because this is how many theoretically-motivated machine learning algorithms explore (e.g., [27]), and partly because it represents the worst-case for frequently missing the user's interests.

Mix-*i*. Interweaves recommendations from Base and the Random strategy, using i slots for FullExplore. It first samples i slots at random to mitigate presentation bias effects and marks them to be filled with recommendations from FullExplore. The remaining slots will be filled with recommendations from Base. The slots are then filled from left to right with the first movie from the corresponding strategy that has also not already been chosen for another slot.

3.3 Movie inventory

To make our system similar to real-world movie streaming websites, we populated it with data from OMDb[1], a free and open movie database. We only kept movies that were released after 1950 to ensure a general level of familiarity. Furthermore, we only selected movies that also appeared in the Tag Genome dataset [42], since the latter dataset was used to make recommendations. After this

[1]http://omdbapi.com/

filtering step, 3470 movies remained in our inventory. In order to have a reasonably attractive default ranking, we sorted all movies first by year in descending order, and then by review score (IMDb score), again in descending order. This means that people were shown recent and highly-rated movies first in the browsing panel by default.

3.4 Participants

We recruited 610 participants from Amazon Mechanical Turk, a crowd sourcing platform. Crowd workers were required to be from a US-based location, and to have an approval rate of at least 95% on previous tasks. Moreover, to ensure that all participants had a consistent experience and saw the same content on the screen, we tested for their browser version and screen resolution. Their median age was between 30 and 40 years as self-reported in the survey. We offered a payment of $1. With an average completion time of 8 minutes, this is an effective wage of $7.50 an hour which is above the US Federal minimum wage. Participants were randomly assigned one of the six presentation strategies as a treatment. From the 610 completed user responses, we filtered out all sessions where people reloaded the page with the interface, or shortlisted more than 15 items since we found these sessions to show overall spamming behavior. After this step, we were left with 577 sessions.

3.5 Study design

We gave participants the following instructions:

> *Imagine you are home alone and want to watch a movie. Use the interface provided next to choose a movie to watch. To make your final choice, click on the button with the download icon next to a movie.*

We chose the scenario above to keep the instructions as simple as possible since any complications were at risk of being ignored by the crowd workers.

After having read the instructions and having seen a short one-minute video about how to use the interface, the participants were taken to the recommendation interface in Fig. 1. After confirming their final choice of movie, they were asked to complete a survey that asked for their experience with the interface.

The survey questions posed to a user after they made their final choice were adapted from existing user-centric frameworks [26, 30] to the crowd sourcing setting [25]. In our survey, we chose questions focusing on the following five properties of the recommendations: quality, helpfulness, transparency, novelty, and choice satisfaction. Except for one binary question, all questions were answered on a 5-point Likert scale from strongly disagree (0) to strongly agree (4).

4 USER SATISFACTION

In this section, we study how varying the amount of exploration affects self-reported user outcome. The following research questions address different aspects of user outcome to the amount of exploration:

(1) Do people perceive differences in the amount of exploration? Besides recommendation accuracy, are any other properties perceived as different?
(2) Can exploration interfere with helpfulness?
(3) How does exploration impact perceived transparency?
(4) Is the cost of exploration linear, i.e. do linear changes in the amount of exploration also lead to linear changes in user outcome?
(5) Does introducing exploration have a significant impact on people's final choices?

Table 1 shows the aggregated results over 577 user responses (93-96 responses for each condition). The last column reports the p-value of a one-way analysis of variance (ANOVA), testing for the equivalence of means among all conditions. Furthermore, an asterisk indicates that a particular exploration strategy is significantly different from Base (t-test, $p < 0.05$, Bonferroni-correction). To test for trends, we also linearly regressed all survey responses with ANOVA significance of $p < 0.01$ against the number of mix-ins and found the slope to be significantly different from zero in all cases with $p < 0.01$. Before turning to the research questions, we will briefly discuss how people perceived their interaction with the interface. We can see from the first two questions that user responses are consistently positive across all conditions, indicating that people did indeed enjoy interacting with the interface, and did not feel stressed during the process. Participants also frequently reported that they made use of the shortlist to keep track of their current choices, independent of which exploration strategy was used (question 3). This overall positive feedback is evidence that the interface is on par with realistic settings and that our results are not biased by a poor interface design.

4.1 Do people perceive differences?

One of the key questions of this paper and of the overall design of this study is whether and how people are affected by the amount of exploration. Knowing that people are able to find differences between the conditions also indicates that our methodology is sufficiently sensitive to the differences we aim to study. Moreover, since we know the composition of each exploration strategy by design, we can formulate a set of hypotheses to test.

For example, as the amount of exploration increases, we expect people to perceive presentation strategies as less accurate and less similar to their shortlisted items. Similarly, we expect people to report an increase in novelty. As the scores of questions 4 and 5 in Table 1 show, these expectations indeed hold true – people report both decreases in recommendation accuracy ("The recommendations were a good fit for my taste") as well as decreases in recommendation similarity to the shortlist ("The recommendations were similar to the movies on the shortlist"). Moreover, people also accurately report novelty of the recommendations as shown in the responses to question 6 ("Most of the recommendations were known to me"). Also, question 6 shows that perceived novelty increases as we mix in more exploration items.

Examining the responses to each of the three questions above more closely, we can also see that user responses for Mix-4 and FullExplore are significantly different from the Base strategy. In fact, Likert scores fall off almost monotonically with an increasing number of mixed-in items. Since we know the composition of each presentation strategy by design, the findings above also serve as validation for our study design – we were to able to successfully match up the user responses with our initial hypotheses.

Question	Conditions						p
	B	M-1	M-2	M-3	M-4	FE	
Interaction adequacy							
1. I enjoyed selecting movies with this interface.	3.23	3.30	3.26	3.37	3.17	3.08	0.39
2. I was stressed out while selecting movies.	0.33	0.20	0.33	0.28	0.33	0.33	0.70
3. I used the shortlist to keep track of my current choices.	3.12	3.03	3.01	3.05	3.11	3.03	0.98
Accuracy							
4. The recommendations were a good fit for my taste.	3.05	2.90	2.93	2.77	2.35*	2.03*	<0.01
5. The recommendations were similar to the movies on the shortlist.	3.21	3.01	2.97	3.01	2.39*	2.17*	<0.01
Novelty							
6. Most of the recommendations were known to me.	2.89	2.80	2.69	2.58	2.38*	2.09*	<0.01
7. My final choice is different from the movies I usually watch.	0.85	0.96	0.91	0.89	1.09	1.20	0.31
8. I have not watched my final choice before. *[binary]*	0.28	0.30	0.30	0.25	0.33	0.24	0.76
Helpfulness							
9. I had a good idea of what I wanted to watch when I started.	1.81	1.89	2.03	1.99	2.07	1.57	0.12
10. The recommendations helped me determine what I was in the mood for.	2.76	2.69	2.64	2.62	2.08*	1.99*	<0.01
Transparency							
11. I understand why the system recommended me the movies that it did.	3.27	3.11	3.16	3.05	2.43*	2.37*	<0.01
12. I was able to steer the recommendations into the right direction.	2.95	2.75	2.82	2.92	2.42*	2.07*	<0.01
Choice Satisfaction							
13. If I were to keep browsing, I'd find a movie that I'd prefer to my final choice.	2.44	2.54	2.41	2.35	2.49	2.56	0.82
14. I would watch my finally chosen movie, given the opportunity.	3.58	3.56	3.66	3.80	3.61	3.52	0.04

Table 1: Aggregated answers for all survey questions. Answers were on a likert scale from 0 (strongly disagree) to 4 (strongly agree), except for question 8. The last column has the p-values for a one-way ANOVA.

4.2 Does exploration interfere with helpfulness?

A good recommender system should ultimately help the user with her decision. In fact, people are often not sure what to watch when they start browsing for a movie as the responses to question 9 show ("I had a good idea of what I wanted to watch when I started."). Hence, it is important to understand whether exploration can actually change how helpful people perceive a system to be. As the results for question 10 indicate ("The recommendations helped me determine what I was in the mood for"), helpfulness is negatively correlated with the amount of mixed-in exploration. Compared to the Base strategy, both Mix-4 and FullExplore produced recommendations that people perceived as significantly less useful.

4.3 Does exploration decrease transparency?

Another important property of a recommender system is transparency, i.e. why a particular recommendation was made to the user. For interactive recommendations, transparency may be even more important since people's recommendations are tied to their actions. We asked participants for transparency in two directions. In question 11, we asked for transparency in the traditional setting ("I understand why the system recommended me the movies that it did") [37], whereas in question 12 we targeted transparency in the proactive setting ("I was able to steer the recommendations into the right direction"). For both questions, we find that transparency correlates negatively with the amount of exploration. Again, Mix-4 and FullExplore were perceived to be significantly less transparent than the Base recommender.

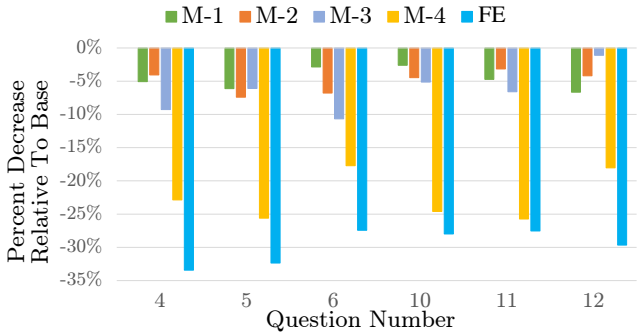

Figure 3: Likert scores fall off in a non-linear way when exploration is increased.

4.4 Are costs linear?

An interesting question is whether a linear increase in the number of mixed-in items will cause a linear change in the Likert scores of user responses. This is particularly important in order to understand *how* an algorithm should explore, i.e. how many potentially poor recommendations people are willing to tolerate. Fig. 3 gives some insight into this question. It shows the relative decrease in

Likert scores compared to Base of all questions that had significant differences under the ANOVA at $p < 0.01$, grouped together by question number, and then sorted by the number of mixed-in items within each group. The bar plot suggests that user responses partition the presentation strategies into two groups – one with Base and Mix-1, Mix-2 and Mix-3; and one with Mix-4 and FullExplore. As the statistical comparisons in Table 1 shows, this is indeed true. Strategies in the last group are significantly different from Base. In other words, scores are not statistically distinguishable from the Base recommender with up to three mixed-in items, and there is a sharp transition going to four or five items.

4.5 Does exploration affect final choices?

The goal of the user task was picking a movie that they would like to watch. A natural question therefore to ask is how much exploration impacts people's final choices. Naively, one would think that since we found decreases in helpfulness and accuracy with increased levels of exploration, these losses would also carry over to choice satisfaction. However, we did not find any significant differences between the six different conditions at the $p < 0.01$ level for choice satisfaction. Users were not more confident that they found the best option as question 13 asked ("If I were to keep browsing, I'd find a movie that I'd prefer to my final choice"). We were also unable to find a significant effect on watch intention in question 14 at the $p < 0.01$ level ("I would watch my finally chosen movie, given the opportunity").

Even though we could not find any direct effects of exploration on choice satisfaction, there could be effects on how novel the choices are. As discussed earlier, we found significant differences with respect to how novel people perceived the recommendations (cf. question 6). To the extent that people actually adopt those novel recommendations or explore more novel options themselves, we should see changes in their final choices. However, we fail to show any significant effects on the novelty of people's final choices as question 7 ("My final choice is different from the movies I usually watch") and question 8 ("I have not watched my final choice before.") show. Note that the options for question 8 were binary, with 0 (no) and 1 (yes). We can also see from question 7 that people are reluctant in general to choose something that would be outside their regular taste profile as Likert scores are consistently low. Question 8 adds even more support to this: Only 24-33% of people chose a movie that they had never seen before.

4.6 Discussion

We found that people were not only aware of the accuracy of a recommender system, but were also able to assess its degree of novelty, and whether it recommends items that are similar to the shortlist. However, even though people were able to report these properties accurately, it had little influence on their reported satisfaction with their final choices. We found neither a significant effect of exploration on overall choice satisfaction, nor on how novel people's final choices were. This implies that people still find ways to be successful at their task, even when random exploration presents less relevant items. This is similar to the finding that people are still able to perform well in information-seeking tasks, despite poorer search results [41].

Perhaps most notable for its implications for exploration cost, the cost of exploration is non-linear. In particular, for *every question* that showed a significant change with respect to the amount of exploration, there are only small statistically insignificant differences in perceived quality up to three mixed-in items. This implies that people are quite forgiving with respect to recommendation quality, as long as they can find at least multiple relevant recommendations. Practically, this suggests that when an algorithm needs to explore, there is a low cost of inserting a limited number of new options. While in our scenario, we found that people needed to have at least two good recommendations among the five, it would be interesting to see how this extends to more general cases. In the case of longer lists, another study [12] showed that inserting four items into a list of twelve recommendations for diversity also did not affect user satisfaction significantly. However, it is unclear when exactly a turning point occurs in this altered setting. Hence, future research should try to gain a better understanding of how people judge lists of recommendations, especially how the overall utility for a list is composed from its individual parts.

5 THE INFORMATIVENESS OF FEEDBACK

In the last section, we looked at how we can minimize the cost of exploration on the user side, and found that cost was minimal for few mixed-in items per recommendation impression. However, it is important to remember why we explore – we want to obtain more information about a user's preferences for future recommendations. Hence, to be effective in this step, we need to ensure a high *informativeness* of feedback signals when exploring. To facilitate a more fine-grained analysis, we define informativeness as the combination of two factors – signal quantity, i.e. how often do people give feedback of a specific type, and signal quality, i.e. how well does the feedback signal reflect the user preferences. So, when striving for highly informative feedback, we need feedback signals to be both abundant as well as indicative of user preferences.

To answer the question of how we can obtain the most informative feedback during exploration, we address the issues of signal quality and quantity in the following two sections separately with the following research questions:

(1) How does the amount of exploration affect feedback quantity? Do different types of feedback behave differently?
(2) Which type of feedback provides the highest quality, i.e. is best able to distinguish between the utility of items from Base and FullExplore?

For the remainder of this section, we will employ the following definitions for different types of feedback signals. An item was *examined* when a user clicked on it, opening a pop-up window with more details about the item. When a user added an item to the shortlist via one of the plus buttons, we refer to this item as *shortlisted*. Similarly, if the mouse cursor was inside an item card (movie poster with the gray title box) for at least 0.5 seconds, we call this item *hovered* over. Finally, if a user picked an item as his or her final choice, we refer to it as *chosen*.

5.1 Feedback quantity

In this section, we look at how the amount of feedback is influenced by exploration. As our goal is to improve learning through

Signal from rec. panel	Interactions	Availability
examined rec.	0.15	9.7%
shortlisted rec	0.94	38.1%
hovered rec.	2.60	62.4%
chosen rec.	0.30	30.0%

Table 2: Feedback quantity varies largely by signal type; with hovering being the most frequent one. Availability is defined as the fraction of sessions in which the signal occurred.

Signal from rec. panel	Conditions						
	B	M-1	M-2	M-3	M-4	FE	p
examined rec.	0.25	0.14	0.09	0.09	0.12	0.19	0.40
shortlisted rec.	1.14	1.15	1.32	1.06	0.58*	0.34*	<0.01
hovered rec.	3.04	2.72	2.90	2.62	2.46	1.87	0.25
chosen rec.	0.39	0.29	0.37	0.33	0.28	0.16*	<0.01

Table 3: Mean number of interactions with recommendations. The last column contains the p-values for a one-way ANOVA.

recommendations, we restrict our analysis to interactions with the recommendation panel. We start our analysis of feedback quantity with overall availability, averaged over all conditions. Table 2 shows the availability and the mean number of interactions with the recommendation panel across all conditions. For the interactions column, we report the average number of interactions that a user had of this type. To compute availability, we report the percentage of sessions in which people had at least one interaction of a specific type with the recommendation panel.

Disappointingly, people examined detailed information about at least one movie recommendation in less than 10% of sessions. However, in 38% of the sessions, we can observe that at least one recommendation was added to the shortlist. With an average of 0.94 interactions per session, we expect 94 shortlisting interactions with the recommendations panel in total on average for 100 sessions (since some users shortlist multiple items in a session). The signal with highest availability is hovering, which is present in 62% of all sessions. Also, the number of item interactions for it is more than twice as high as for shortlisting (2.60 vs. 0.93). One can also see the general trend that less intrusive feedback signals (i.e. hovering, shortlisting) are more common than more obtrusive signals (i.e. examining). In 30% of all sessions, people chose their final item from their recommendations as opposed to choosing it from the browsing panel as the last row shows. However, this signal obviously comes too late to influence session-based recommendation. It is important though to note that Table 2 averages across all conditions, so the fact that interaction scores are low is also due to suboptimal strategies.

We now analyze how signal quantity varies for different levels of exploration. Table 3 reports the mean number of interactions under each condition. Although not shown in the table, we did not find significant differences in average time-to-decision (session length), average total number of shortlisted movies, or in the average number of recommendations people were exposed to. Again, an asterisk indicates significance with respect to Base using a t-test ($p < 0.05$, Bonferroni correction). As Table 3 shows, the level of exploration has a significant impact on feedback quantity in two cases. There is a significant impact of exploration on both the number of shortlist interactions, and the number of chosen items that came from the recommendations. Both Mix-4 and FullExplore had significantly fewer shortlisting interactions than Base. Interestingly, this grouping is consistent with what we found in our user-centric evaluation in the last section. In the number of recommendations that were also finally chosen by people, we only were able to find significance between Base and FullExplore. However, neither the number of hover interactions nor the number of examine interactions on recommended items showed significant differences, although the number of hover interactions seems to decrease with increasing exploration.

5.2 Feedback quality

We now turn to analyze feedback quality. By construction, and also from the results in Section 4, we know that the utility of the Base strategy to people is higher on average than the utility of the FullExplore strategy which chose items at random. This controlled setting gives us the opportunity to see how this difference in average utility is reflected in implicit signals. This can help guide the design of more sensitive online experiments as well as the design of better learning algorithms. We define signal quality as the ability of a signal to distinguish between the average utility of items that came from Base and the average utility of items that came from the FullExplore strategy. Note that we know how many random items each strategy showed to a user, and also at which positions they were shown. We define the utility $U_\pi(s)$ of a strategy $\pi \in \{\text{Base, FullExplore}\}$ in session s as an average over all individual item utilities $u(i,s)$ of items i in the set $\mathcal{I}(\pi)$ of items that a strategy showed to the user:

$$U_\pi(s) = \frac{1}{|\mathcal{I}(\pi)|} \sum_{i \in \mathcal{I}(\pi)} u(i,s).$$

The individual item utilities can encode various interaction signals, for example, $u(i,s)$ could be a binary function indicating whether a user examined item i in session s. We also hope to find $U_{\text{Base}}(s)$ to be greater than $U_{\text{FullExplore}}(s)$, again, since we know it is by construction and from the user experiments. More formally, we capture this in a win score

$$\text{win}_{\text{Base}}(s) = \begin{cases} 1.0, & \text{if } U_{\text{Base}}(s) > U_{\text{FullExplore}}(s) \\ 0.5, & \text{if } U_{\text{Base}}(s) = U_{\text{FullExplore}}(s) \\ 0.0, & \text{otherwise.} \end{cases}$$

Note that this is closely related to ranking interleaving in information retrieval [31] where the utility functions capture the number of clicks each strategy obtained.

Table 4 reports the average win score for different signal types. Note that Base and FullExplore are omitted because they only showed items from one strategy. We start our discussion by looking at feedback quality in this setting only, ignoring all sessions in

Signal from rec. panel	Conditions			
	M-1	M-2	M-3	M-4
Only sessions where signal available				
examined rec.	0.80	1.00	0.86	0.55
shortlisted rec.	0.91*	0.96*	0.89*	0.70
hovered rec.	0.79*	0.82*	0.74*	0.67*
chosen rec.	0.96*	0.92*	0.87*	0.67
All sessions				
examined rec.	0.53	0.54	0.53	0.51
shortlisted rec.	0.70*	0.72*	0.65*	0.56
hovered rec.	0.72*	0.72*	0.67*	0.62*
chosen rec.	0.63*	0.65*	0.62*	0.55

Table 4: Fraction of sessions where people interacted more with Base items than with FullExplore items. An asterisk indicates significance using a two-sided Bernoulli test ($p < 0.05$, Bonferroni correction).

which no interactions were available. Looking at the upper half of Table 4, we can see that even though Base wins more in the case of examine interactions, there is not enough evidence to conclude that it is statistically different from a random preference between Base and FullExplore. The problem here is feedback quantity, and we will return to this shortly. Using hover interactions allows us to reliably tell apart the utility of Base and FullExplore for all levels of exploration. The signal quality seems lower than for shortlisting, and final choosing, with hovering being able to distinguish correctly between the utility of Base and FullExplore in around 67%-80% of sessions where a hovered recommendation existed or in 62%-72% of all sessions. There also appears to be a decay from Mix-3 to Mix-4 in signal quality; this could indicate that there are also non-linear effects on item-specific feedback signals.

We can analyze signal quality and quantity together by looking at data from all sessions in the bottom half. We can see now that examined and finally chosen items are less indicative of strategy preference (Base vs. FullExplore) since interactions of that type are only available in a fraction of sessions (cf. Table 2). The best signal types overall are shortlisting and hovering which, for conditions including up to three mixed-in items, are able to correctly order the utility of Base and FullExplore.

Although not reported in this paper, we ran a pilot experiment where we reverse the Base strategy, and inserted one mixed-in item from the reversed ranking. This ensured that the mixed-in item was from a strategy which had higher quality than FullExplore. We found the same effects as above – people still interacted more with items from Base in their sessions.

5.3 Discussion

We saw that feedback quantity, for example the number of shortlisting operations on recommended items, is affected significantly by different levels of exploration. Interestingly, looking at feedback quantity for the latter signal recovered the same grouping of presentation strategies as the user-centric evaluation in the previous section – Base, Mix-1, Mix-2, and Mix-3 were indistinguishable statistically, but different from Mix-4 and FullExplore. Limiting the amount of exploration is not only important for user satisfaction, but also for feedback quantity, as both user satisfaction and feedback quantity are affected in a similar manner. It is plausible that the reduced user satisfaction is at least partly responsible for the decreases in feedback quantity.

As a step towards learning from the collected feedback, we looked at which signals can best tell apart items from Base and FullExplore. We found that both shortlisting as well as hovering correctly rank Base over FullExplore. Clicks, as traditionally used in learning-to-rank [31], were not helpful in this scenario since there were simply too few of them available overall. Finding that shortlisting is among the most informative signals adds further support to the insight that learning algorithms can greatly benefit from improved interface design where users are offered the right incentives to interact [34].

Our results also indicate that an important but often overlooked issue when designing learning algorithms is the interplay of feedback quality and quantity. For example, we found shortlisting feedback to have higher quality than hovering, but hovering to be available in more sessions. Given these difference between feedback signals, learning algorithms could try to combine multiple signal types to further improve overall signal quality and quantity – for example, using hovering in addition to shortlisting feedback.

6 LIMITATIONS & FUTURE WORK

Our user study comes with certain limitations. First, participants were paid to participate in the study, and may had different incentives when interacting with the system, such as maximizing their hourly wage. This might influence certain measures, such as the time-to-decision. We did our best to minimize these effects, but it still would be interesting to connect our results to user behavior in the wild.

Another open question is to what extent interactivity plays a role in how people perceive recommendations. Our results mirror many findings from long-term user studies, however, there is more research needed to study the precise connections between interactive recommendations and traditional static recommendations. Understanding how these two approaches are connected could, for example, allow us to improve evaluation for static recommendation as well. As an intermediate step, future work could combine interactive recommendation with techniques for session-based recommendation [19, 21, 43].

Lastly, we only studied uninformed random exploration in this paper which does not control for concepts like novelty. There is an increasing amount of empirical evidence showing that novelty can be perceived negatively by people [5, 10] and that popular recommendations are often preferred [5]. This motivates the need for research on more constrained presentation strategies, e.g., strategies that expose people only to items they are likely to know [14, 32].

7 CONCLUSIONS

This paper explored the behavioral aspects of minimizing the cost of exploration while ensuring a high quality and quantity of feedback signals. Regarding the cost of exploration, our user study finds that small amounts of mix-in exploration are tolerated well, while

larger amounts lead to a super-linear drop-off in user satisfaction as well as to a decrease the quantity of feedback. This suggests that algorithms should prefer moderate levels of exploration over longer period of time over high levels of exploration over shorter periods. We also examined different types of feedback with respect to their quality, and found both shortlisting and hovering events to provide the best overall signal. Our findings give practical advice to interface designers as well as offer starting points for designing learning algorithms using the right feedback signals.

ACKNOWLEDGMENTS

This work was supported in part through NSF Awards IIS-1513692, IIS-1615706, and a gift from Bloomberg. We thank Pantelis P. Analytis as well as the anonymous reviewers for their valuable feedback and questions.

REFERENCES

[1] Michal Aharon, Oren Anava, Noa Avigdor-Elgrabli, Dana Drachsler-Cohen, Shahar Golan, and Oren Somekh. 2015. ExcUseMe: Asking Users to Help in Item Cold-Start Recommendations. In *RecSys*. 83–90.

[2] Oren Anava, Shahar Golan, Nadav Golbandi, Zohar Karnin, Ronny Lempel, Oleg Rokhlenko, and Oren Somekh. 2015. Budget-constrained item cold-start handling in collaborative filtering recommenders via optimal design. In *WWW*. 45–54.

[3] Svetlin Bostandjiev, John O'Donovan, and Tobias Höllerer. 2012. TasteWeights: a visual interactive hybrid recommender system. In *RecSys*.

[4] Léon Bottou, Jonas Peters, Denis Xavier Charles, Max Chickering, Elon Portugaly, Dipankar Ray, Patrice Y Simard, and Ed Snelson. 2013. Counterfactual reasoning and learning systems: the example of computational advertising. *Journal of Machine Learning Research* 14, 1 (2013), 3207–3260.

[5] Òscar Celma and Perfecto Herrera. 2008. A New Approach to Evaluating Novel Recommendations. In *RecSys*.

[6] Mark Claypool, Phong Le, Makoto Wased, and David Brown. 2001. Implicit interest indicators. In *IUI*.

[7] Dan Cosley, Shyong K Lam, Istvan Albert, Joseph A Konstan, and John Riedl. 2003. Is seeing believing? How recommender system interfaces affect users' opinions. In *CHI*.

[8] Paolo Cremonesi, Franca Garzotto, Sara Negro, Alessandro Vittorio Papadopoulos, and Roberto Turrin. 2011. Looking for "good" recommendations: A comparative evaluation of recommender systems. In *INTERACT*.

[9] Paolo Cremonesi, Franca Garzotto, and Roberto Turrin. 2013. User-centric vs. system-centric evaluation of recommender systems. In *INTERACT*.

[10] Michael D Ekstrand, F Maxwell Harper, Martijn C Willemsen, and Joseph A Konstan. 2014. User perception of differences in recommender algorithms. In *RecSys*.

[11] Steve Fox, Kuldeep Karnawat, Mark Mydland, Susan Dumais, and Thomas White. 2005. Evaluating implicit measures to improve web search. *TOIS* 23, 2 (2005).

[12] Mouzhi Ge, Dietmar Jannach, Fatih Gedikli, and Martin Hepp. 2012. Effects of the Placement of Diverse Items in Recommendation Lists. In *ICEIS*.

[13] Qi Guo and Eugene Agichtein. 2012. Beyond dwell time: Estimating document relevance from cursor movements and other post-click searcher behavior. In *WWW*.

[14] Abhay S Harpale and Yiming Yang. 2008. Personalized active learning for collaborative filtering. In *SIGIR*.

[15] F Maxwell Harper and Joseph A Konstan. 2016. The MovieLens datasets: History and context. *TiiS* 5, 4 (2016).

[16] Gerald Häubl and Valerie Trifts. 2000. Consumer decision making in online shopping environments: The effects of interactive decision aids. *Marketing Science* 19, 1 (2000).

[17] Chen He, Denis Parra, and Katrien Verbert. 2016. Interactive recommender systems: A survey of the state of the art and future research challenges and opportunities. *Expert Systems with Applications* 56 (2016).

[18] Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen, and John T Riedl. 2004. Evaluating collaborative filtering recommender systems. *TOIS* 22, 1 (2004).

[19] Balázs Hidasi, Massimo Quadrana, Alexandros Karatzoglou, and Domonkos Tikk. 2016. Parallel recurrent neural network architectures for feature-rich session-based recommendations. In *RecSys*.

[20] Jeff Huang, Ryen W White, and Susan Dumais. 2011. No clicks, no problem: Using cursor movements to understand and improve search. In *CHI*.

[21] Dietmar Jannach, Lukas Lerche, and Michael Jugovac. 2015. Adaptation and evaluation of recommendations for short-term shopping goals. In *RecSys*.

[22] Dietmar Jannach, Lukas Lerche, and Michael Jugovac. 2015. Item Familiarity Effects in User-Centric Evaluations of Recommender Systems. (2015).

[23] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, Filip Radlinski, and Geri Gay. 2007. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *TOIS* 25, 2 (2007).

[24] Nicolas Jones and Pearl Pu. 2007. User technology adoption issues in recommender systems. In *NAEC*.

[25] Aniket Kittur, Ed H Chi, and Bongwon Suh. 2008. Crowdsourcing user studies with Mechanical Turk. In *CHI*.

[26] Bart P Knijnenburg, Martijn C Willemsen, Zeno Gantner, Hakan Soncu, and Chris Newell. 2012. Explaining the user experience of recommender systems. *User Modeling and User-Adapted Interaction* 22, 4-5 (2012).

[27] Volodymyr Kuleshov and Precup Doina. 2010. Algorithms for the multi-armed bandit problem. *Journal of Machine Learning* 47, 1 (2010).

[28] Lihong Li, Wei Chu, John Langford, and Xuanhui Wang. 2011. Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 297–306.

[29] Winter Mason and Siddharth Suri. 2012. Conducting behavioral research on Amazon's Mechanical Turk. *Behavior Research Methods* 44, 1 (2012).

[30] Pearl Pu, Li Chen, and Rong Hu. 2011. A user-centric evaluation framework for recommender systems. In *RecSys*.

[31] Filip Radlinski, Madhu Kurup, and Thorsten Joachims. 2008. How does clickthrough data reflect retrieval quality?. In *CIKM*.

[32] Al Mamunur Rashid, Istvan Albert, Dan Cosley, Shyong K Lam, Sean M McNee, Joseph A Konstan, and John Riedl. 2002. Getting to know you: learning new user preferences in recommender systems. In *IUI*.

[33] Paul Resnick and Hal R Varian. 1997. Recommender systems. *Commun. ACM* 40, 3 (1997).

[34] Tobias Schnabel, Paul N. Bennett, Susan T. Dumais, and Thorsten Joachims. 2016. Using Shortlists to Support Decision Making and Improve Recommender System Performance. In *WWW*.

[35] Shai Shalev-Shwartz. 2012. Online Learning and Online Convex Optimization. *Found. Trends Mach. Learn.* 4, 2 (Feb. 2012).

[36] Bracha Shapira, Francesco Ricci, Paul B Kantor, and Lior Rokach. 2015. *Recommender Systems Handbook*. (2nd ed.). Springer.

[37] Rashmi Sinha and Kirsten Swearingen. 2002. The role of transparency in recommender systems. In *CHI Extended Abstracts*.

[38] Harald Steck, Roelof van Zwol, and Chris Johnson. 2015. Interactive Recommender Systems: Tutorial. In *RecSys*.

[39] Adith Swaminathan and Thorsten Joachims. 2015. Counterfactual risk minimization: Learning from logged bandit feedback. In *Proceedings of the 32nd International Conference on Machine Learning*. 814–823.

[40] K. Swearingen and R. Sinha. 2001. Beyond Algorithms: An HCI Perspective on Recommender Systems. In *SIGIR Workshops*.

[41] Andrew H Turpin and William Hersh. 2001. Why batch and user evaluations do not give the same results. In *SIGIR*.

[42] Jesse Vig, Shilad Sen, and John Riedl. 2012. The Tag Genome: Encoding community knowledge to support novel interaction. *TiiS* 2, 3 (2012).

[43] Chao-Yuan Wu, Christopher V. Alvino, Alexander J. Smola, and Justin Basilico. 2016. Using Navigation to Improve Recommendations in Real-Time. In *RecSys*.

[44] Cai-Nicolas Ziegler, Sean M McNee, Joseph A Konstan, and Georg Lausen. 2005. Improving recommendation lists through topic diversification. In *WWW*. 22–32.