Rating Support Interfaces to Improve User Experience and Recommender Accuracy

Tien T. Nguyen[1] Daniel Kluver[1] Ting-Yu Wang[1] Pik-Mai Hui[1]
Michael D. Ekstrand[1] Martijn C. Willemsen[2] John Riedl[1]

[1]GroupLens Research
Computer Science and Engineering
University of Minnesota
Minneapolis, MN 55455
{tien,kluver,twang,hui,ekstrand,riedl}@cs.umn.edu

[2]Human-Technology Interaction Group
School of Innovation Sciences
Eindhoven University of Technology
m.c.willemsen@tue.nl

ABSTRACT

One of the challenges for recommender systems is that users struggle to accurately map their internal preferences to external measures of quality such as ratings. We study two methods for supporting the mapping process: (i) reminding the user of characteristics of items by providing personalized tags and (ii) relating rating decisions to prior rating decisions using exemplars. In our study, we introduce interfaces that provide these methods of support. We also present a set of methodologies to evaluate the efficacy of the new interfaces via a user experiment. Our results suggest that presenting exemplars during the rating process helps users rate more consistently, and increases the quality of the data.

Categories and Subject Descriptors

H.1.2 [**Information Systems**]: User/Machine Systems—*Human information processing*; H.5.2 [**Information interfaces and presentation**]: User Interfaces—*Evaluation / methodology*

Keywords

preference elicitation; user experience; human factors; interface design; experimentation; cognitive load; recommendation accuracy

1. INTRODUCTION

Herlocker et al. coined the term *magic barrier* to refer to the supposition that there may be a lower bound on the minimum error that can be achieved by any ratings-based recommender system [9]. The idea is that user ratings include some level of noise, as evidenced by inconsistencies observed when asking users to rerate previously rated items [10]. One interpretation of the magic barrier is that it is unlikely that new algorithms can produce large decreases in recommender error. Another interpretation, and the one that motivates this research, is that if we can understand the cognitive processes that drive the magic barrier then perhaps we can create new ways for users to rate items, reducing the minimum achievable error in our systems.

Psychological research on preference construction and expression provides insight into how users go about rating items. This research suggests that most users do not form a rating at the time of consumption and store that rating in their memory [7]. Rather, they store a complex set of thoughts, feelings, perspectives and insights, and map those complex stored values into a rating when they are asked to provide one. This process is labeled preference construction [15]. There are many ways this process can go poorly, potentially increasing the noise, and raising the magic barrier.

In this work we address two likely sources of errors in the rating process. The first is that users may not clearly remember items, especially if the users experience these items a long time prior to rating them. The second is that users may struggle to accurately and consistently map their constructed preferences to the rating scale.

It seems plausible that modifying the rating interface may mitigate these problems. To that end, we study four interfaces inspired by these two key methods of support: one that is a baseline, one to help users *remember movies*, one to help users *understand their personal rating scale*, and one combined interface to help *with both* simultaneously.

We explore a novel set of methodologies to examine the effectiveness of the interfaces by performing a re-rating experiment involving 386 users and 38,586 ratings in MovieLens. We apply the information theoretic framework developed in our previous work [13], and the magic barrier framework developed by Said et al. [21] to measure the effect of our interfaces on the consistency of, and noise in, a user's ratings. Furthermore, we investigate the impact of these interfaces on cognitive load and user satisfaction via a user survey.

Noise in user rating is one of the main challenges in recommender systems. We argue that we can improve the quality of rating data by implementing interfaces that help users map their constructed preference to the rating scales better. With this in mind, our contributions are novel interface designs to collect user data with less noise, and a set of methodologies to evaluate the effectiveness of these interfaces.

RecSys'13, October 12–16, 2013, Hong Kong, China.

http://dx.doi.org/10.1145/2507157.2507188.

2. BACKGROUND

There is noise in user ratings. Such noise can occur because of inconsistencies in ratings over time [10]. There are two reasons for these inconsistencies. First, user preferences are unstable and therefore vulnerable to external influences such as anchoring effects [1] or manipulated predictions [5]. Second, users may have difficulty mapping their preferences into provided rating scales. Mapping their preferences accurately to the rating scales can be demanding for users, resulting in inconsistencies [20].

To address the noise and to increase the prediction quality, several researchers investigate the mapping process. From the algorithm perspective, reducing noise and improving predictions is possible using multiple ratings on the same user item-pair. [2]. From the interface design perspective, changing to more granular rating-scales can help recommender systems collect and produce more information about user preference [13]. However, it has been shown that more granular rating scales demand more cognitive effort from users [23], and that gaining more accurate user models sometimes can lead to decrease in user experience and user loyalty [16].

Preference elicitation research.

To inspire our interface design we turn to research of preference construction and elicitation. The field of decision making shows an 'emerging consensus' [22] towards the idea that decision makers do not have well defined preferences [3, 7]. Preferences are not easily retrieved from some master list in memory, but rather are constructed while making a decision [15] and depend strongly on the situational context and task characteristics. Researchers have recently linked these phenomena to memory processes [27]. When a person is asked to evaluate an item, a memory query will be activated to recall information that is needed for the evaluation. This query will activate knowledge and associations of the particular item that will be used to form a preference of that item. In other words the evaluation of the item will depend on how the preference for that item is (re)constructed from memory and from information provided in the interface.

One cause of rating noise is in the memory processes involved in recalling the item. As our memory is reconstructive, different aspects will come to mind at different times and in different situations [27], resulting in different constructed preferences. Our memories might also become less detailed and precise over time, and the importance of certain aspects might change, leading to our ratings' changes over time [4]. System designers help users form consistent and useful preferences by helping the user recall the item. They do this by presenting descriptions and attributes of the item that will help the user remember the item. We call these features 'memory support' as they help users remember the item.

For memory support to be helpful, we argue that it should be tailored to the user as much as possible. If we are able to predict which aspects are most important for a user, we can use these aspects to provide more effective personalized memory support than general background information. Therefore we will compare memory support interfaces using personalized tags against a minimal support interface.

Noise in ratings can also be caused by difficulty in mapping a formed preference to a rating. Past research shows strong impact of anchors [24] and task characteristics [11, 25] on rating responses, when a user lacks a good internal representation of the rating scale. While it is intuitive that three stars means less preference than four stars, it can be difficult for a user to decide (in an absolute sense) whether to rate an item three or four stars. System designers help users represent their preference by making the rating scale more intuitive. A common way to do this is by providing community guidelines mapping each rating to a simple description such as: (4 stars = 'I liked it'). We call these features 'rating support' as they help convert a preference to a formal rating. However, simple labels won't be sufficient to solve the mapping problem. Tailored rating support is needed to help the user express the evaluation in a consistent way.

Prior work on rating interface design [19] has suggested using the users' past ratings to help them make future ratings. In particular they suggest an interface in which rating options are annotated with past ratings will help people make ratings that are ordered consistently. Indeed, research on preference elicitation has investigated the differences between comparing items on their own (single evaluation) and comparing items against each other (joint evaluation) [11]. When items are evaluated in isolation, the evaluation is absolute and will be predominantly based on aspects that are easy to evaluate. Using joint evaluation allows users to be more sensitive to dimensions on which items vary. We will support joint evaluation in our interface by showing exemplars for each point on the rating scale. These exemplars will be other similar items rated in the past that serve as personalized anchors on the scale, e.g.: 'you previously rated this item 3 stars'. Research on comparative evaluation [17] suggests these exemplars should be similar to serve as good anchors otherwise it might be hard to make any comparison (or even contrast effects can occur). We will compare this interface with exemplars (supporting joint evaluation) to an interface without exemplars (single evaluation).

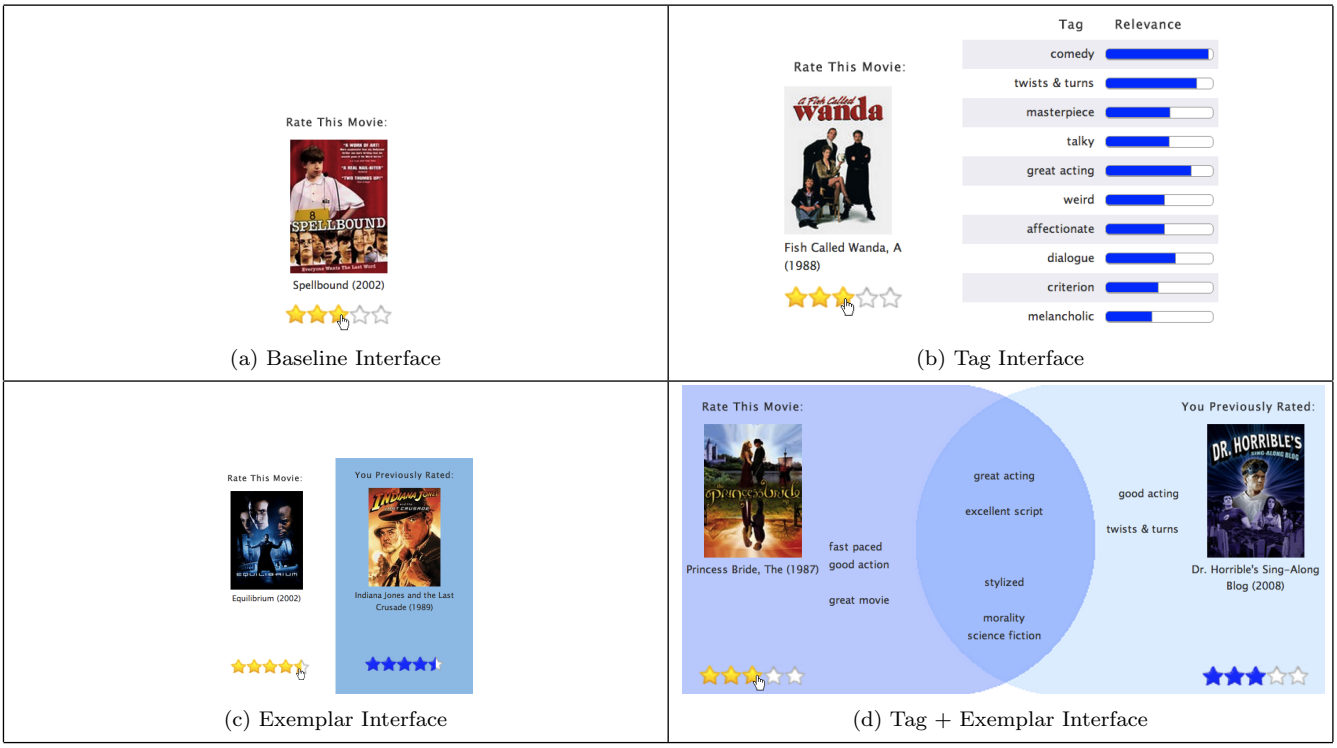
3. INTERFACE DESIGN

To answer the research questions, we develop four interfaces: one with minimalistic support that serves as the baseline, one that shows tags, one that provides exemplars, and another that combines the previous two features (figures 1).

Figure 1a presents the baseline design allowing users to map their internal preferences into ratings. To assist our users in the rating process, we provide basic information such as posters and titles. However, posters and titles might not provide sufficient memory support. Therefore, we design the tag interface that provides personalized memory support.

3.1 The Tag Interface

In our design process, we look for a personalizable form of memory support that easily triggers the recall process. While some users remember a movie because it is funny, others remember the movie because it is visually appealing. To provide personalized information about the movie, we use the tag genome [26] combined with users' previous ratings. The tag genome is a tag-movie relevance matrix, inferred from user-supplied tags, ratings, and external textual sources. The relevance of a tag t to a movie m determined by the tag genome is a value between 0 and 1 denoted as $rel(t, m)$. We remove two classes of tags from the genome prior to rendering our interfaces. The first is tags such as 'excellent' and 'great' that indicate a general quality judgement of the movie; these are redundant with ratings and do not provide any information about the movie. The second is binary metadata tags such as directors, actors, 'based on a book', etc. These tags



(a) Baseline Interface

(b) Tag Interface

(c) Exemplar Interface

(d) Tag + Exemplar Interface

Figure 1: Our four interface designs to provide memory-support, rating support and joint-evaluation support.

describe properties that are objectively verifiable and either true or false; it is incorrect to infer them when they are not explicitly added to the item. Future work should build a way to incorporate such metadata tags, but for now we restrict our use of the tag genome to tags for which inference makes sense. Figure 1b shows the tag interface.

To personalize the interface, we select tags that best help users reconstruct their memory about the movies. Due to the limited space on the interface, only a set of 10 personalized tags from 1574 candidates can be displayed. To address this problem, we use Nguyen et al.'s approach [18] to compute how much information about user u's preference a tag t can tell, denoted as $pMI(t,u)$. We hypothesize that the more information a tag can tell about user u's preference, the more likely it is helpful to the user. Moreover, a useful tag must be sufficiently relevant to the movie. Therefore, we require that any selected tag must have a minimum relevance of 0.25.

Similar tags, such as 'political' and 'world politics', diminish the effectiveness of memory support if they are displayed together. To address this challenge, we place a similarity constraint on the tag selection algorithm. Our prototyping suggests that mutual information between tags works well. Formally, let $P(t_1)$ be the probability that tag t_1 is applied to a random movie, $P(t_1,t_2)$ be the joint-probability that tags t_1 and t_2 are both applied to a random movie. Hence, the tag similarity, denoted as $dTagsim(t_1,t_2)$, is defined as $dTagSim(t_1,t_2) = \frac{I(t_1,t_2)}{H(t_2)}$, where $I(t_1,t_2)$ is the mutual information of the two tags, and $H(t_2)$ is the entropy of tag t_2. Note that dTagSim, or directed tag similarity, is asymmetric, i.e. $dTagSim(t_1,t_2) \neq dTagSim(t_2,t_1)$. After several trials, we find that 1% (0.01)to be a reasonable cut-off.

With these concepts laid out, our tag selection approach is a constrained optimization problem described as follow:

$$\begin{aligned} &\underset{T^\star}{\text{argmax}} \sum_{t \in T^\star} rel(t,i) \times pMI(t,u) \\ &\text{subject to } \mid T^\star \mid = 10 \\ &\qquad dTagSim(t_i,t_j) < 1\% \ \ \forall\, t_i,t_j \in T^\star,\ i \neq j \\ &\qquad rel(t,i) > 0.25 \ \ \forall\, t \in T^\star \end{aligned}$$

Since finding sets of perfect personalized tags is computationally expensive, we implement a greedy algorithm to find an approximate solution. Starting with an empty set of tags, the algorithm iteratively selects the tag with the highest $rel(t,i) \times pMI(t,u)$ such that when added to the result the constraints are still satisfied.

3.2 The Exemplar Interface

The exemplar interface improves upon the baseline by showing exemplars. An exemplar for a rating-movie pair is a movie that the user previously gave the same rating. Figure 1c shows the design of the exemplar interface. To support the joint evaluation, we provide anchoring points by annotating rating options with the exemplars. Each exemplar at a specific rating option is the most similar movie to the movie to rate based on the cosine similarity of the movie rating vectors. At any star on the rating scale, if the user has not rated any movie, a blank image is shown.

3.3 The Tag + Exemplar Interface

To investigate the effect of combining 'memory support' and 'rating support', we develop the fourth interface as seen in figure 1d. In this interface, users make use of personalized tags to compare their rated movies to the movie to rate. When users hover over a star, an exemplar is shown. Each tag then

moves either flush left, or flush right to denote which movie the tag is more relevant to (the movie to rate or the exemplar respectively). The tag will move to the center if the tag is equally relevant to both movies (relevance values within 0.1 of each other). To help the users understand the movement of the tags a Venn diagram is shown in the background of the interface. This diagram reinforces the idea that tags on one side or the other best describe the associated movie, and tags in the center describe both movies equally well.

The tag + exemplar interface uses the same exemplar selection strategy as the exemplar interface. Selecting tags, however, is more complicated in this interface. We want tags that not only are relevant to the user and movie, but also may help the user compare between movies. We measure this in two ways. First, for each tag t we measure $\sigma(t)$ the standard deviation of the relevance values of each chosen exemplar to that tag. If a tag's relevance varies greatly between the exemplars then it can help serve to differentiate them.

Secondly, we measure how well a tag splits between being more relevant to the movie to rate, or the exemplars. If a tag is always more relevant to the movie to rate, or always more relevant to the exemplars, then the tag does not help the user compare the different rating choices. To measure this we define $Z(i,t) = \frac{\max(M_t, L_t)}{\min(M_t, L_t)}$, where M_t is the number of exemplars to which tag t has higher relevance than to the movies to rate, and L_t is the number of exemplars to which tag t has lower relevances than to the movies to rate. When the tag would appear with the exemplar as often as with the movie to rate then $Z(i,t) = 1$, as the balance becomes less equal $Z(i,t)$ becomes larger.

Our algorithm for this tag + exemplar interface is also a constraint optimization problem described as follow:

$$\underset{T^\star}{\text{argmax}} \sum_{t \in T^\star} rel(t,i) \times pMI(t,u) \times \sigma(t) \div Z(i,t)$$

$$\text{subject to } \mid T^\star \mid = 10$$

$$dTagSim(t_i, t_j) < 1\% \ \ \forall\, t_i, t_j, \in T^\star,\ i \neq j$$

Since finding sets of perfect personalized tags is computationally expensive, we implement a greedy algorithm to find an approximate solution. Starting with an empty set of tags, the algorithm iteratively selects the tag with the highest $rel(t,i) \times pMI(t,u) \times \sigma(t) \div Z(i,t)$ such that when added to the result the constraints are still satisfied.

4. EXPERIMENTAL DESIGN

To compare the our interfaces, we ask users to rate 50 movies, and re-rate these movies after at least two weeks. Users are asked to answer 22 survey questions after finishing the second round. To help users understand the interfaces, we provide a tutorial for each interface explaining the interface and walking users through the interface step by step. Our analyses are based on techniques described as below.

4.1 Efficacy Metrics

RMSE.

One of the hopes in improving rating quality is to reduce the magic barrier of a recommendation system. One easy to measure consequence of reducing the magic barrier is to reduce the RMSE of a recommendation system. To measure this effect we estimate the minimum RMSE possible over the ratings using the technique introduced by Said et al. [21]. Said et al. argue that the average variance of ratings (measured at each user item pair) gives an estimate of the minimum possible RMSE over these ratings. If any of our interfaces lower the magic barrier we should see a lower minimum RMSE. To confirm our findings with this method we also build simple item-item recommenders for each interface and compare the RMSE on these interfaces directly.

Although the RMSE quality metrics can suggest that we have lowered the magic barrier, these metrics do not capture the whole picture of rating quality. For example, RMSE can trivially be reduced by influencing users to use a smaller range of the rating scale, which would lead to overall less informative ratings. An absolute measure of the information and noise contained in ratings could be a more useful tool in evaluating the efficacy of the interfaces. Therefore, in the next two sub-sections, we propose approaches to measuring the amount of noise and information about user preferences in ratings collected from each interface.

Preference Bits per Rating.

In prior work [13] we showed how to measure the preference bits of ratings from an interface. This framework can be used to estimate the total amount of information ratings contain about user preferences. An increase in preference bits from any interface would indicate that ratings on that interface tell us more about users' preferences.

We measure preference bits by gathering two ratings for each user-item pair and computing the mutual information between the two sets of ratings. This measures how much the first round of ratings reduces our uncertainty about ratings in the second round, providing a lower bound for the information that one rating contains about the preference that caused it. If ratings on one interface give us more information about user preferences than ratings on another interface, we say that the interface is more effective at eliciting preference.

Rating Noise.

It is possible for ratings from different interfaces to contain the same amount of information with different amounts of noise. Therefore, even if our interfaces yield the same amount of preference information, we may still prefer one with less noise. Information theory gives us a clear way of measuring the amount of noise contained in our ratings.

Let π and R be random variables representing the user's true preference for, and rating on, an item respectively. The total randomness in ratings is measured by the entropy of the ratings $H(R)$. Relative to any other random variable X, the entropy of R can be split into two parts, the mutual information $I(R;X)$ measuring the amount of information R contains about X, and the conditional entropy of R given X, $H(R|X)$. Taking this decomposition with respect to the user's true preference π we get the following identity.

$$H(R) = I(R;\pi) + H(R|\pi) \tag{1}$$

Based on equation 1, we will measure the amount of noise contained in ratings by the conditional entropy of a rating given the user's preference, $H(R|\pi)$. Because true preference π cannot be measured we will have to estimate the true rating noise. In our previous work [13] we showed that we can estimate the true preference bits using two ratings for the same user-item pair taken at different times R_1, R_2,. We prove a similar result about the rating noise. Formally in previous work we showed:

$$I(R;\pi) < I(R_1;R_2) \tag{2}$$

Using equations 1 and 2 we have the following:

$$H(R|\pi) = H(R) - I(R;\pi) < H(R_1) - I(R_1;R_2) = H(R_1|R_2)$$

The conditional entropy between two ratings is therefore an upper limit to the true rating noise. We will use this metric as an estimate of the true rating noise of our interfaces. Because recommendation algorithms are likely sensitive to noise in the ratings, we hypothesize that we can learn better from ratings with less total noise.

4.2 User Experience Metrics

Objective User Experience (Cognitive Load).

Harper et al. [8] investigate motivations behind user rating behaviors in a recommender system domain, and find that users perceive rating-time costs when they provided ratings. Harper et al. also point out that the users keep providing ratings when they perceive that the benefits outweigh the costs they pay. Sparling et al. [23] investigate further to estimate the mental cost and benefits on different rating scales. Since accurately measuring user mental costs is hard, we follow Sparling et al.'s approach to use rating time to estimate cognitive load. We use this metric, as well the user experience metric below, to evaluate the trade-offs between getting more information about user preference and overloading users.

Subjective User Experience (Self-report).

To learn how users perceive the usefulness and difficulty of the four interfaces, we conclude our experiment with a survey. We then follow Knijnenburg et al.'s [14] evaluation framework for user experience. The framework models how objective system aspects such as the user interface influence users' subjective perceptions and experience, controlling for other factors such as situational characteristics and personal characteristics. Particularly, we measure how usefulness of the system (experience) is influenced by the perceived difficulty of the system (subjective system aspect), and how each of these are affected differently by the 4 different interfaces, controlling for the self-reported expertise as a personal characteristic. The survey consists of 22 7-likert-scale statements, ranging from 'Strongly disagree' to 'Strongly agree', that query the theoretical constructs of interest: usefulness of the interface, difficulty of the interface and self-reported movie expertise. Some questions are inverted to check users responses for coherence and lack of effort. Participants have to answer all questions. They can change their answers before submission. Participants can provide extra feedback in a text-box. We also ask (but not require) participants' age and gender.

4.3 Design

We conduct our study with users of MovieLens,[1] an online collaborative filtering-based movie recommender system. In order to make our algorithms work on these interfaces, we select only users who have rated at least 100 movies that have both tag genome and Netflix title data available. We also require that the 100 ratings be provided since MovieLens changed to its current half-star rating scales [2]. Finally, we require that users have logged in at least once during the last four years. After preparing data for the experiment (2013-03-01), we randomly invited 3829 users via email invitations. The users were randomly assigned to one of the four interfaces.

[1]http://www.movielens.org/

[2]We estimate that MovieLens switched to a five-rating-star with half-star increment on 2003-02-12

Our participants are asked to rate 50 movies in the first round, and re-rate these 50 movies in the second round. To simulate the real rating experience, the participants are asked to rate the most recent 50 movies that they rated prior to the study. To avoid users recalling recently rated movies, we wait at least 14 days after data preparation before inviting users to participate in our study (round 1). This guarantees that at least two weeks have passed since the user last rated the selected movies in MovieLens. For the same reason, we also wait 14 days before starting round 2. After finishing the second round, participants are asked to complete a survey asking their experience with the interfaces as well as the rating process. For each 20 participants, a randomly selected one receives a $50 Amazon Gift card.

5. RESULTS

Our experiment ran from March 21st 2013 until April 25th 2013, collecting 38,586 [4] ratings from 386 users, of which 103 users were assigned the baseline interface, 100 to the exemplar interface, 98 to the tags interface, and 85 to the tags + exemplars interface. 352 provided information about age (mean: 36, min: 18, max: 81, σ : 11). 304 were male, 72 female, and 10 did not report. Our analyses don't reveal significant differences between the two genders.

5.1 Efficacy Metrics

RMSE.

We follow the procedure described in [21] to estimate the minimum RMSE for each interface. The estimated minimum RMSE for our interfaces are given in table 1. To test these differences for statistical significance we compute an estimated minimum RMSE for each user. Using an ANOVA on the per user minimum RMSE estimates we find marginal significance that the average per-user minimum RMSE differs between interfaces ($p = 0.0574$). Using a TukeyHSD we find marginal evidence that the exemplar interface has a lower average minimum RMSE than the tag and baseline interfaces (adjusted $p = 0.0689$, 0.098 respectively).

To control for possible error in the ANOVA due to violations of the normality assumption we confirm these results with a pairwise Wilcoxon rank sum test over the per user minimum RMSE using the Holm method to correct for multiple comparisons. We find that the minimum RMSE is significantly different between the exemplar interface and baseline($p < 0.005$), however, we find only minimal support for the conclusion that the exemplar interface differs significantly from the tag interface ($p = 0.208$). Therefore we conservatively conclude that the only significant pairwise difference in our minimum RMSE estimates is between exemplar and baseline.

To compare these results against actual predictive accuracy we use the LensKit toolkit [6] to train and evaluate an item-item collaborative filtering recommender over the ratings from the four interfaces. We use 10-fold cross validation with a 10 item holdout set for each test user. Table 1 shows the per-user RMSE of each condition for each round. The exemplar interface has the lowest RMSE in both rounds. The measured RMSEs are significantly different ($p < 0.001$ using ANOVA on the user RMSEs); the exemplar had lower RMSE than the baseline in round 1 ($p < 0.001$ using Tukey HSD

[4]Due to network issues, we had 14 incomplete ratings which were removed from all of the analyses.

Interface	Minimum RMSE	RMSE (round 1)	RMSE (round 2)	Preference Bits per Rating [3]	Rating Noise	Mean Rating Time	Median Rating Time
Baseline	0.232	1.13	1.12	1.26	1.71	4.26	3.71
Exemplar	0.209	0.95	0.95	1.22	1.62	5.53	4.60
Tag	0.237	1.03	1.10	1.22	1.72	4.49	4.07
Tag + Exemplar	0.227	1.04	1.01	1.25	1.70	6.70	5.60

Table 1: Our quantitative results for four interfaces

post-hoc test), and both the baseline and tag interfaces differ significantly in round 2 ($p < 0.05$).

Preference Bits per Rating.

We estimate preference bits using the technique described in [13]. The results are summarized in table 1.

To our knowledge significance testing on information theoretic values is not a widely studied field. Therefore, to test for significance in our measured mutual information differences we use a resampling based strategy to test each pairwise difference. Under the null hypothesis ratings from any two interfaces are drawn from the same distribution (and therefore have the same true mutual information). To test for differences between any two conditions we randomly permute the assignments of users to conditions. It is important to permute interface labels over users rather than ratings to control for outlier users who rate with significantly more or less information or noise than the average user. For each permutation we compute the difference in means of measured mutual information between the two conditions. We reject our null hypothesis if fewer than 5% of our random trials have differences of means as large or larger than observed. We perform this test using 10000 permutations and find that no pairwise difference in mutual information is statistically significant. Therefore we do not have sufficient evidence to conclude that our interface modifications had a significant effect on the amount of preference information extracted.

Rating Noise.

The preference noise of an interface is estimated by the conditional entropy of the first rating given the second rating in a rerating dataset. The conditional entropy estimates for each interface can be seen in table 1. The rating noise metric generally agrees with the RMSE based metrics, especially in measuring the exemplar interface as having the least noise. To test these measurements for differences we use the same resampling strategy discussed above. Using 10000 random permutations we find that no pairwise difference in conditional entropies is significant. Therefore we do not have sufficient evidence to conclude that our interface modifications had a significant effect on rating noise.

It is interesting to note that while we are not able to conclude that our interfaces have an effect on rating noise, we are still able to conclude effects on the RMSE of the ratings. This result is contrary to our belief that noise in user ratings is a primary cause of the magic barrier. We expect this finding is largely due to differences in power between our statistical tests for rating noise, and the RMSE based metrics. Because statistics on information theoretic measures is a not-well explored field of statistics, it is possible that analysis with a more powerful technique would allow us to find significance in our rating noise metric that match the results found in the RMSE section. Nonetheless we feel that the rating noise metric is a better metric for future use because it directly measures the noise in ratings, rather than measuring a consequence of this noise.

5.2 User Experience Metrics

Objective User Experience (Cognitive Load).

Cognitive load is estimated by measuring the time it takes users to rate a movie. We assume that it will take users time to process the information when using interfaces. A good interface could decrease the cognitive load and time required.

Before analyzing the rating times, we first exclude long rating times that occur because users are distracted during rating or take a break. Such long times introduce bias in our analysis since these rating times do not reflect real users' cognitive loads. We exclude the top 1% of the data points for each interface, resulting in cut-off points of 41 seconds for the baseline interface, 43 seconds for the tag interface, 55 seconds for the exemplar interface, and 79 seconds for the tag + exemplar interface.

As users made 50 ratings in each round, we take the average rating time per round as an estimated measure of cognitive load. Table 1 shows the statistics of rating time per interface aggregated over both rounds. Our ANOVA analyses suggest that the required cognitive load for the baseline and tags interfaces be the smallest among the four ($p < 0.01$). The difference between the two interfaces is not significant ($p = 0.29$). This suggests that users do not take advantage of the memory support provided by the tags interface. Participants spent the most time on the tag + exemplar interface, followed by the exemplar interface (The difference is significant, $p < 0.0001$). All other pairwise comparison are also significant ($p < 0.0001$). This suggests that users utilize the supports provided by the two exemplar empowered interfaces [5].

Subjective User Experience (Self-report).

Our questionnaire was designed to measure user experience of usefulness and difficulty of the interface and participant's expertise. The items in the questionnaires were submitted to a confirmatory factor analysis (CFA). The CFA used ordinal dependent variables and a weighted least squares estimator, estimating 3 factors. Items with low factor loadings, high cross-loadings, or high residual correlations were removed from the analysis. Factor loadings of included items are shown in table 2, as well as Cronbach's alpha and average variance extracted (AVE) for each factor. The values of AVE and Cronbach's alpha are good, indicating convergent validity. The square roots of the AVEs are higher than any of the factor correlations, indicating good discriminant validity.

The subjective constructs from the CFA (table 2) were organized into a path model using a confirmatory structural equation modeling (SEM) approach with ordinal dependent variables and a weighted least squares estimator. In the resulting model, the subjective constructs are structurally

[5]we also perform multilevel linear regression analysis with random intercepts that also take into account the 50 repeated observations per round, with similar results. We also apply separate ANOVA analyses for two rounds, and observe that in the second round, there is a marginal significant difference between the two exemplar empowered interfaces $p = 0.068$)

Considered Aspects	Items	Factor Loading
Usefulness Alpha: 0.83 AVE: 0.546	MovieLens should use this interface	0.89
	The interface helped me think carefully about how much I liked the movie	0.70
	The interface helped me express my ratings consistently	0.66
	The interface did not make rating easier compared to the usual MovieLens rating interface	-0.71
	Rating movies with this interface was fun	0.71
	The interface helped me to remember what I like and dislike about movies	
	I found the interface helpful in relating a movie to other movies I have seen	
	The interface did not motivate me to think carefully about my ratings	
	I know of other sites with rating interfaces I would rather use than this one	
	I would recommend this interface to other users	
Difficulty Alpha: 0.84 AVE: 0.71	I found the interface confusing	0.86
	The interface was easy to understand	-0.86
	I found the interface difficult to use	-0.82
	Rating with this interface was fast	
	Rating with this interface was easy	
Movie Expertise Alpha: 0.68 AVE: 0.55	I'm a movie lover	0.75
	Compared to people I know, I read a lot about movies	0.85
	Compared to people I know, I'm not an expert on movies	-0.61
	I mostly like popular movies	
	I get very excited about some upcoming movies	
	I know why I like the movies I like	
	I often like to compare a new movie to others I've seen	

Table 2: Items presented in the questionnaires. Items without a factor loading were excluded from the analysis.

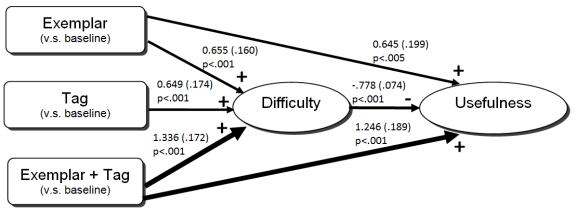


Figure 2: The structural equation model fitted on the Subject constructs difficulty and usefulness and their relations to the 4 interfaces

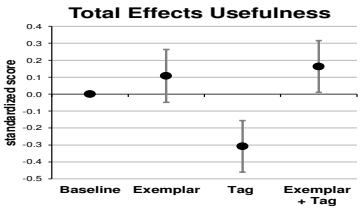


Figure 3: Total effects of the interfaces on usefulness: error bars are 1 standard error of the mean. Comparisons are made against the baseline interface.

related to each other and to the 4 interface types. In the final model, the expertise factor did not relate to any other variable, and was therefore removed from the analysis. The final model had a good model fit [6] ($\chi^2(52) = 3408$, p <.001, CFI = 0.979, TLI = 0.971, RMSEA = .070, 90% CI: [.054, .085]). Figure 2 displays the effects found in this model. Factor scores in the final model are standardized; the numbers on the arrows ($A \rightarrow B$) denote the estimated mean difference in B, measured in standard deviations, between participants that differ one standard deviation in A. The number in parentheses denotes the standard error of this estimate, and the p-value below these two numbers denotes the statistical significance of the effect. As per convention, we exclude effects with p >= .05. The three interfaces are compared to the baseline interface. The SEM model shows that all three interfaces are seen as more difficult than the baseline condition, with the exemplar + tag interface having roughly twice the impact (1.336) as the exemplar (0.655) and tag (0.649) interfaces. Both interfaces with exemplars are perceived more useful than the baseline interface, with the exemplar + tag interface almost twice as useful than the baseline as the exemplar interface. The tag interface is not more useful than the baseline (no significant loading). Reviewing the total structure of the model we observe that difficulty loads negatively on usefulness: the interfaces that are more difficult are also perceived as less useful. We should therefore inspect the total effects on usefulness, which are plotted in figure 3. Compared to the baseline interface the two exemplar-powered interfaces are only slightly more useful (differences not significant) because their difficulty stands in the way of a much better user experience. This suggests that one could increase the user experience of these interfaces by making them less difficult (i.e less confusing, easier to understand), which should increase their perceived usefulness. Note that the total effect on usefulness suggests that the tag interface is actually perceived to be less useful than the baseline (p<.05) mainly because it is more difficult than the baseline. Thus the tag interface is seen as worse than the baseline interface.

[6]Hu and Bentler [12] propose cut-off values for the fit indices to be: CFI > .96, TLI > .95, and RMSEA < .05, with the upper bound of its 90% CI falling below 0.10

6. DISCUSSION

Our user survey reveals that in general providing rating support, as in the exemplar interface, helps users rate more consistently. Although our participants felt that the interfaces providing rating support are more difficult than our baseline interface, they liked these interfaces because they perceived the interfaces to be more useful. Several participants requested to see these interfaces implemented on MovieLens. This is consistent with our quantitative analyses. Between the two interfaces providing rating support, the exemplar one appears to have the lowest RMSE, the lowest minimum RMSE, and the least amount of natural noise.

Our quantitative analyses, however, show that reminding users of movie features in personalized fashion, as in the tag interface, did not help to improve the rating quality or the perceived usefulness of the interface. Our participants perceived the tag-interface as more difficult than the baseline. One potential reason for this is our tag selection algorithm. It is possible that an improved algorithm would select tags that have a more positive effect on the user.

The SEM analysis shows that the perceived usefulness and difficulty of our interfaces is related. This correlation is what led to having only small gains in total usefulness. While our interfaces had a positive effect on usefulness, they also had an opposite effect by being more perceived difficult. This leads us to believe that efforts simplifying our interfaces could significantly improve the user experience.

One of the contributions of this work is an exploration of a set of methodologies to evaluate rating interfaces. It is important that we have techniques for comparing novel recommender interfaces. Future work should continue exploring methodologies for comparing interfaces. Many of our metrics required a large amount of data to give statistically significant results, and metrics such as RMSE should be suspect when too few ratings are used. Likewise, while we feel the information theoretic metrics can be very informative, more work is still needed to develop robust techniques for comparing information theoretic values.

Improving prediction accuracy is one of the main interests in recommender system research, both in academia and in industry. For example, Netflix awarded $ 1 million dollars to a research team for the 10% of improvement in prediction accuracy[7]. Our efforts modifying recommender interfaces to gather high quality data with less noise complement the efforts to improve prediction accuracy. Overall we see the results of this work, and feedback from our participants, as promising. We believe that future work building upon our interfaces, algorithms, and methodologies can help us improve recommender systems.

7. ACKNOWLEDGEMENT

This work is supported by the NSF grants IIS 09-68483, IIS 08-08692, and IIS 10-17697. We also thank Professor Joseph A. Konstan for his helps during the revision.

8. REFERENCES

[1] G. Adomavicius, J. Bockstedt, S. Curley, and J. Zhang. Recommender systems, consumer preferences, and anchoring effects. In *Decisions@RecSys Workshop*, pages 35–42, Chicago, 2011.

[2] X. Amatriain, J. M. Pujol, N. Tintarev, and N. Oliver. Rate it again: increasing recommendation accuracy by user re-rating. In *In Proc. of RecSys '09*, pages 173 – 180, New York, NY, USA, 2009. ACM.

[3] J. R. Bettman, M. F. Luce, and J. W. Payne. Constructive consumer choice processes. *Journal of Consumer Research*, 25(3):187 – 217, Dec. 1998. ArticleType: research-article / Full publication date: December 1998 / Copyright ©1998 Journal of Consumer Research Inc.

[4] D. Bollen, M. Graus, and M. C. Willemsen. Remembering the stars?: effect of time on preference retrieval from memory. In *In Proc. of RecSys '12*, pages 217 – 220, New York, NY, USA, 2012. ACM.

[5] D. Cosley, S. K. Lam, I. Albert, J. A. Konstan, and J. Riedl. Is seeing believing?: how recommender system interfaces affect users' opinions. In *In Proc. of CHI '03*, pages 585–592, New York, NY, USA, 2003. ACM.

[6] M. D. Ekstrand, M. Ludwig, J. A. Konstan, and J. T. Riedl. Rethinking the recommender research ecosystem: reproducibility, openness, and LensKit. In *In Proc. of RecSys '11*, pages 133 – 140, New York, NY, USA, 2011. ACM.

[7] B. Fischhoff. Value elicitation: Is there anything in there? *American Psychologist*, 46(8):835–847, 1991.

[8] F. M. Harper, X. Li, Y. Chen, and J. A. Konstan. An economic model of user rating in an online recommender system. In *User Modeling 2005*, pages 307–316. Springer, 2005.

[9] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1):5–53, 2010.

[10] W. Hill, L. Stead, M. Rosenstein, and G. Furnas. Recommending and evaluating choices in a virtual community of use. In *In Proc. of CHI '95*, pages 194–201, New York, NY, USA, 1995.

[11] C. K. Hsee. The evaluability hypothesis: An explanation for preference reversals between joint and separate evaluations of alternatives. *Organizational Behavior and Human Decision Processes*, 67(3):247–257, Sept. 1996.

[12] L.-t. Hu and P. Bentler. Cutoff criteria for fit indexes in covariance structure analysis: Conventional criteria versus new alternatives. *Structural Equation Modeling: A Multidisciplinary Journal*, 6(1):1–55, 1999.

[13] D. Kluver, T. T. Nguyen, M. Ekstrand, S. Sen, and J. Riedl. How many bits per rating? In *In Proc. of RecSys '12*, pages 99 – 106, New York, NY, USA, 2012. ACM.

[14] B. P. Knijnenburg, M. C. Willemsen, Z. Gantner, H. Soncu, and C. Newell. Explaining the user experience of recommender systems. *User Modeling and User-Adapted Interaction*, 22(4-5):441–504, Mar. 2012.

[15] S. Lichtenstein and P. Slovic. *The Construction of Preference*. Cambridge University Press, Sept. 2006.

[16] S. M. McNee, S. K. Lam, J. A. Konstan, and J. Riedl. Interfaces for eliciting new user preferences in recommender systems. In P. Brusilovsky, A. Corbett, and F. d. Rosis, editors, *User Modeling 2003*, number 2702 in Lecture Notes in Computer Science, pages 178–187. Springer Berlin Heidelberg, Jan. 2003.

[17] T. Mussweiler. Comparison processes in social judgment: Mechanisms and consequences. *Psychological Review*, 110(3):472–489, 2003.

[18] T. T. Nguyen and J. Riedl. Predicting users' preference from tag relevance. *User Modeling, Adaptation, and Personalization, UMAP 2013*, pages 274–280, 2013.

[19] S. Nobarany, L. Oram, V. K. Rajendran, C.-H. Chen, J. McGrenere, and T. Munzner. The design space of opinion measurement interfaces: exploring recall support for rating and ranking. In *In Proc. of CHI '12*, pages 2035 – 2044, New York, NY, USA, 2012. ACM.

[20] M. P. O'Mahony, N. J. Hurley, and G. C. Silvestre. Detecting noise in recommender system databases. In *Proceedings of the 11th international conference on Intelligent user interfaces*, IUI '06, pages 109 – 115, New York, NY, USA, 2006. ACM.

[21] A. Said, B. J. Jain, S. Narr, and T. Plumbaum. Users and noise: The magic barrier of recommender systems. In *UMAP 2012*, pages 237–248. Springer, 2012.

[22] I. Simonson. Determinants of customers' responses to customized offers: Conceptual framework and research propositions. *Journal of Marketing*, 69(1):32–45, Jan. 2005.

[23] E. I. Sparling and S. Sen. Rating: how difficult is it? In *In Proc. of RecSys '11*, pages 149 – 156, New York, NY, USA, 2011. ACM.

[24] A. Tversky and D. Kahneman. Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157):1124–1131, Sept. 1974.

[25] A. Tversky, S. Sattath, and P. Slovic. Contingent weighting in judgment and choice. *Psychological Review*, 95(3):371–384, 1988.

[26] J. Vig, S. Sen, and J. Riedl. The tag genome: Encoding community knowledge to support novel interaction. *ACM Trans. Interact. Intell. Syst.*, 2(3):13:1 – 13:44, Sept. 2012.

[27] E. U. Weber and E. J. Johnson. Mindful judgment and decision making. In *Annual Review of Psychology*, volume 60, pages 53–85. Annual Reviews, Palo Alto, 2009.

[7] http://www.netflixprize.com