Testing a Recommender System for Self-Actualization

Daricia Wilkinson, Saadhika Sivakumar,
Pratitee Sinha, Bart P. Knijnenburg

Clemson University
Clemson, USA
dariciw@clemson.edu, ssivaku@g.clemson.edu, psinha@g.clemson.edu,
bartk@clemson.edu

Abstract. Traditionally, recommender systems were built with the goal of aiding users' decision-making process by extrapolating what they like and what they have done to predict what they want next. However, in attempting to personalize the suggestions to users' preferences, these systems create an isolated universe of information for each user, which may limit their perspectives and promote complacency. In this paper, we describe our research plan to test a novel approach to recommender systems that goes beyond "good recommendations" that supports user aspirations and exploration.

Keywords: Recommender Systems; Filter Bubble; Choice Overload; Self-Actualization.

1 Introduction

Recommender systems have become ubiquitous in daily user interactions across many e-commerce websites, social networking sites and streaming services. The main purpose of these systems is to provide users with relevant information, and as such, much of the content consumed online is personally tailored [17].

Although personalized content has numerous benefits, presenting items that are only based on some of users' expressed preferences could hinder the effectiveness of the recommender, and trap users in a "filter bubble" [14] that limits their perspectives, discourages exploration and prevents genuine taste development.

Recently, scholars have acknowledged the importance of other user-centered factors beyond accuracy that contribute to the effectiveness of recommender systems [6, 12]. This shift "beyond the accuracy" has spawned investigations into solutions that improve all aspects of the user interaction experience. For instance, to increase understandability, researchers have suggested providing explanations of the recommendations [4]. However, these explanations could increase the already high conformity, as users simply trust the system's explanation rather than engaging in true understanding and exploration [5].

This could have long-term societal consequences as the persuasive nature of recommendations could replace human creativity and understanding [13], turning humans into "input" for systems rather than acknowledging the opportunities for taste development. This paper outlines our research plan to build upon our recent proposal for *recommender systems for self-actualization*- systems that helps users in understanding their unique tastes through development and exploration [9].

2 Algorithmic Features

Previous research on critiquing [2, 3, 15, 16] and diversifying recommendations [20] has already investigated better options for the Top-N suggestions, yet the focus of these alternative methods is to provide "good recommendations". However, our approach is fundamentally different, since it carefully considers the psychology of consumer choice processes, and *supports* (rather than replaces) these processes by featuring new recommendation lists. In addition to displaying a Top-N list, our system also differs from previous studies by simultaneously displaying the following lists that promote exploration and taste development (for more details see [9]): Our alternative lists will address the following issues:

Incorrect negative predictions. In conventional recommenders, items that a system predicts that you may dislike are never shown. While they are mostly correct, it is possible for the system to be mistaken on some. These mistakes are hard to correct, because items with low-valued predictions are never recommended. Presenting users with a list of *things we think you'll hate* will allow users to correct or confirm low-valued predictions.

Our list of *things we think you'll hate* contains items that have a low predicted rating for this user, compared to the average predicted rating. To populate this list, compute the difference between the total average rating of the item and the user predicted rating, with the following formula:

$$items = max(average\ predicted\ rating - user\ predicted\ rating)$$

This allows users to correct mistakes quickly.

Unknown preferences. It is difficult for recommenders to predict items for which there is insufficient information about whether the user will like them or not. As a result, recommender usually tailor to users' *known* preferences only. Rather than only catering to known preferences, we propose to display a *list of things we have no clue about.* This list consists of items with a user predicted rating for which the system has the lowest confidence. Current recommender algorithms do not provide confidence intervals, so for our study we estimate the system's confidence by computing the difference in user-predicted ratings for different algorithms, e.g. matrix factorization (mf) and k-nearest neighbors (knn):

$$items = max(predicted\ rating_{mf} - predicted\ rating_{knn})^2$$

This allows the system to learn information about all of a user's preferences, rather than just a subset of their preferences.

Novel items. New items are an enduring complication in recommender systems. Since users have yet to try them, they rarely show up among the recommendations. Most recommender systems solve this *cold start problem* through content based techniques to approximate predicted ratings. However, this solution ignores the fact that user may at time actually be excited to try new things, even if it does not always fit their preferences [18]. We propose to resolve the cold start problem by simply presenting items with limited rating data to users who are excited to try them. These "hipster" users are likely to appreciate *things you'll be among the first to try*, and their feedback on these items will help to populate the available information and hence, improve the system. We detect these "hipster" users by detecting their high percentage of top-rated items with very few ratings, and then show them more of such items.

$$users = max(\%\ top\ rated\ items\ with\ (\#ratings < threshold))$$

$$items = min(\#ratings)$$

Controversial items. Recommenders usually identify a set of users that are similar to the current user, and then calculate recommendations based on the preferences of these nearest neighbors. This often leads to recommendations that the neighbors unanimously like. These "safe" recommendations do not challenge a user's tastes beyond what is generally agreed upon as "good" among like-minded users. However, these neighbors may not always be an unvaried group, and there may be certain *polarizing* items that some of them really like, but others hate. Our fourth feature will detect these polarizing items. This list of *things that are controversial* can help users to develop their unique tastes.

Among the four proposed features, identifying polarizing items is arguably the most challenging from an algorithmic perspective. The simplest approach to identify polarizing items is to select items that have the highest rating variability or range (rather than average) among the neighbors:

$$items = max(var(neighbors'\ predicted\ ratings))$$

A more sophisticated approach would be to cluster the identified neighbors based on their ratings, and then select items that best discriminate between clusters.

The proposed features could improve recommenders? ability to support people in life-altering decisions (e.g. choosing an education, a job, an insurance plan, or a retirement fund) where it is important that they develop a strong sense of determination about the chosen path. The features would also improve recommenders? ability help people make lifestyle choices (e.g., about music, movies, or fashion) based on carefully developed personal tastes. Our proposed plan to test and evaluate these features in a movie recommender system are outlined below.

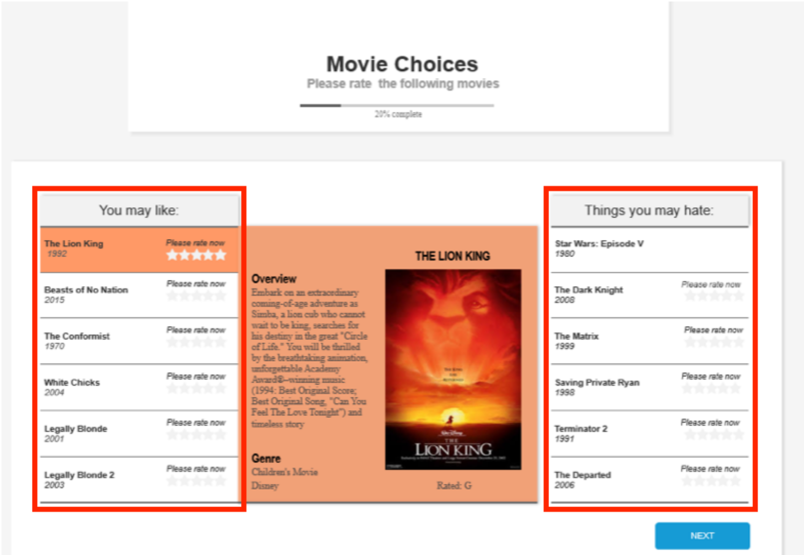

Fig. 1. Mockup of the experiment showing the Top-10 list on the left and "Things you may hate" condition on the right. The list on the right will differ for each of the five conditions and it will be manipulated between-subjects.

3 Research Plan

The goal of our research is to develop, test and evaluate a recommender system that supports rather replaces the decision-making process for users. Our proposed features that were mentioned in the previous section will be tested alongside a traditional Top-N recommender. The system will have the capability to display a Top-N recommendation list, as well as the lists of the four new features. We will train the system using the MovieLens dataset. An online experiment will be conducted to test the RSSA features.

3.1 Online Experiment

The experiment will be conducted on Amazon Mechanical Turk (MTurk) with at least 300 participants. Including the traditional "Top-N only" recommendations, the experiment will also test the four RSSA features in combination with a Top-N list (see Fig. 1).

In our study, participants will see two lists of 10 items: one list will be the traditional Top-10 ("Things you might like"), while the other list will be manipulated between-subjects with the following five conditions:

- "More things you might like"; i.e. the next 10 recommendation (Top-11-20)
- "Things we think you will hate"
- "Things we are not sure about"

- "Things you'll be among the first to try"
- "Things that are controversial"

After being randomly assigned to one of the five experimental conditions, participants will be asked rate 15 movies that they have seen before, to use as a base for their recommendations. Next, we will show them the recommendations the Top-10 list of "things you might like" on the left, while the list on the right will feature 10 items that are based on the randomly selected experimental condition. At this point, we will ask participants to rate the movies from the two lists. After this final round of rating we will update the two lists, and ask participants to select one movie that they would watch right now. Finally, participants will be asked to complete a questionnaire to evaluate their experience with using the system. The behavioral and objective aspects to be evaluated are outlined in Table 1. We will adopt highly validated questionnaire items from previous studies [1, 7, 8, 10, 11, 19] and develop additional scales along the lines of the Knijnenburg et al. user experience framework for recommender systems [10] which will contribute to the theory of recommender systems evaluation.

Aspect	**Description**
Questionnaire ***(Q)***, Behavior ***(B)***	
Perceived Recommendation quality, diversity, novelty ***(Q)***	Existing scales [1, 7, 8, 10, 19]
System and choice satisfaction ***(Q)***	Existing scales [1, 7, 8, 10, 11, 19]
Choice and tradeoff difficulty ***(Q)***	Existing scales [1, 19]
Perceived taste coverage ***(Q)***	Whether users think the system is able to cover all of their tastes
Objective coverage ***(B)***	Average number of different items that are recommended to each user over the course of the experiment
Fear of missing things ***(Q)***	Average number of different items that are recommended to each user over the course of the experiment
Taste clarification potential ***(Q)***	Whether users think the system helps them understand their own tastes
Taste development potential ***(Q)***	Whether users think the system helps them develop their own tastes
Perceived choice conformity ***(Q)***	Whether users think they are consuming similar things like everyone else
Objective choice conformity ***(Q)***	Average cosine similarity between users' consumption patterns

Table 1. User experience aspects measured in the experiment.

4 Conclusion

In this paper, we describe a new direction for recommender systems that move towards supporting our aspirational selves rather than pushing content to users based on their history. We outline our research plan for developing and evaluating the new interface and algorithmic features. Aside from this, we are also working with several companies and organizations to build these new features into real-life *recommenders*. We believe that our *Recommender Systems for Self-Actualization* acknowledges the multidimensionality and evolving nature of human beings, and can fundamentally change the way recommender systems are used.

Acknowledgments

This research was supported in part by the NSF award IIS 1565809.

References

1. D. Bollen, B. P. Knijnenburg, M. C. Willemsen, and M. Graus. Understanding choice overload in recommender systems. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, RecSys '10, pages 63–70, New York, NY, USA, 2010. ACM.
2. L. Chen and P. Pu. Interaction design guidelines on critiquing-based recommender systems. *User Modeling and User-Adapted Interaction*, 19(3):167–206, Aug. 2009.
3. L. Chen and P. Pu. Critiquing-based recommenders: Survey and emerging trends. *User Modeling and User-Adapted Interaction*, 22(1-2):125–150, Apr. 2012.
4. G. Friedrich and M. Zanker. A taxonomy for generating explanations in recommender systems. *AI Magazine*, 32(3):90–98, 2011.
5. F. Gedikli, D. Jannach, and M. Ge. How should i explain? a comparison of different explanation types for recommender systems. *Int. J. Hum.-Comput. Stud.*, 72(4):367–382, Apr. 2014.
6. C. He, D. Parra, and K. Verbert. Interactive recommender systems: A survey of the state of the art and future research challenges and opportunities. *Expert Syst. Appl.*, 56:9–27, 2016.
7. B. P. Knijnenburg, S. Bostandjiev, J. O'Donovan, and A. Kobsa. Inspectability and control in social recommenders. In *Proceedings of the Sixth ACM Conference on Recommender Systems*, RecSys '12, pages 43–50, New York, NY, USA, 2012. ACM.
8. B. P. Knijnenburg, N. J. Reijmer, and M. C. Willemsen. Each to his own: How different users call for different interaction methods in recommender systems. In *Proceedings of the Fifth ACM Conference on Recommender Systems*, RecSys '11, pages 141–148, New York, NY, USA, 2011. ACM.
9. B. P. Knijnenburg, S. Sivakumar, and D. Wilkinson. Recommender systems for self-actualization. In *Proceedings of the 10th ACM Conference on Recommender Systems*, RecSys '16, pages 11–14, New York, NY, USA, 2016. ACM.
10. 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, Oct. 2012.

11. A. Kobsa, B. P. Knijnenburg, and B. Livshits. Let's do it at my place instead?: Attitudinal and behavioral study of privacy in client-side personalization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 81–90, New York, NY, USA, 2014. ACM.
12. J. A. Konstan and J. Riedl. Recommender systems: From algorithms to user experience. *User Modeling and User-Adapted Interaction*, 22(1-2):101–123, Apr. 2012.
13. J. Lanier. *You Are Not a Gadget: A Manifesto*. Thorndike Press, 2010.
14. E. Pariser. *The Filter Bubble: How the New Personalized Web Is Changing What We Read and How We Think*. Penguin Books, New York, NY, USA, 2012.
15. P. Pu, B. Faltings, L. Chen, J. Zhang, and P. Viappiani. Usability guidelines for product recommenders based on example critiquing research. In F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, editors, *Recommender Systems Handbook*, pages 511–545. Springer, 2011.
16. P. Resnick, R. K. Garrett, T. Kriplean, S. A. Munson, and N. J. Stroud. Bursting your (filter) bubble: Strategies for promoting diverse exposure. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work Companion*, CSCW '13, pages 95–100, New York, NY, USA, 2013. ACM.
17. P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. Grouplens: An open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, CSCW '94, pages 175–186, New York, NY, USA, 1994. ACM.
18. A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '02, pages 253–260, New York, NY, USA, 2002. ACM.
19. M. C. Willemsen, M. P. Graus, and B. P. Knijnenburg. Understanding the role of latent feature diversification on choice difficulty and satisfaction. *User Modeling and User-Adapted Interaction*, 26(4):347–389, Oct. 2016.
20. C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen. Improving recommendation lists through topic diversification. In *Proceedings of the 14th International Conference on World Wide Web*, WWW '05, pages 22–32, New York, NY, USA, 2005. ACM.