Conducting User Experiments in Recommender Systems

Bart P. Knijnenburg
Department of Informatics
University of California, Irvine
bart.k@uci.edu

ABSTRACT
There is an increasing consensus in the field of recommender systems that we should move beyond the offline evaluation of algorithms towards a more user-centric approach. This tutorial teaches the essential skills involved in conducting user experiments, the scientific approach to user-centric evaluation. Such experiments are essential in uncovering how and why the user experience of recommender systems comes about.

Categories and Subject Descriptors
H.1.2. **[Models and principles]**: User/Machine Systems–*software psychology*; H.5.2 **[Information Interfaces and Presentation]**: User Interfaces–*evaluation/methodology*; H.4.2. **[Information Systems Applications]**: Types of Systems–*decision support*

General Terms
Measurement, Experimentation, Human Factors, Standardization.

Keywords
User experiments, recommender systems, user experience, user-centric evaluation.

1. INTRODUCTION
From a methodological perspective, the evaluation of recommender systems has undergone an interesting development [8]. The recent focus on "user-centric" evaluation [4, 11] is inspired by the suggestion that higher accuracy does not always mean higher user satisfaction [9], and that the algorithm accounts for only a small part of the real-world relevance of a recommender system. Other aspects such as the presentation and interaction have a signification impact on the user experience [2, 5, 10].

To make inferences about the users' experience, we need to move beyond measuring their behavior, and measure their *subjective valuations* as well [6]. Moreover, as users' interaction with recommender systems is highly context-dependent [1, 3], personal and situational characteristics also need to be taken into account.

In Knijnenburg et al. [6] we present a framework for the user-centric evaluation of recommender systems that takes all these aspects into consideration (Figure 1). This framework can be used as a guideline for *user experiments* to reveal how and why the user experience of recommender systems comes about. However, conducting such experiments is a complex endeavor. How does one test whether a certain system aspect has a significant influence on e.g. users' satisfaction with the system? How does one measure a subjective concept like "user satisfaction" to begin with?

As recommender systems evaluation is becoming more user-centric, an increasing number of recommender systems researchers have to deal with these tricky questions. We thus often find papers without clearly defined hypotheses, lacking proper experimental manipulations, and/or testing a large number of seemingly unrelated effects using simple t-tests. Although it is encouraging to see user-centric evaluation efforts bloom, these evaluations are with notable exceptions not up to par with state-of-the-art research methods and statistical analyses. Whereas our RecSys 2011 short paper provides a pragmatic yet curtailed approach to user-centric evaluation [7] that fits such budding research efforts, this tutorial provides a more thorough treatment of user experiments as a mature scientific approach to the user-centric evaluation of recommender systems.

For the intended audience of recommender systems researchers wanting to get serious about user-centric evaluation, the tutorial covers all aspects involved in conducting user experiments: developing testable hypotheses, sampling participants from the right population, constructing useful experimental manipulations, robustly measuring behavior and subjective valuations, and analyzing the results using modern statistical methods.

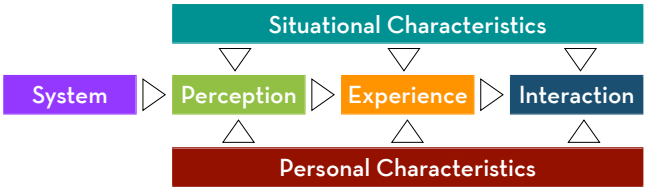

Figure 1. Framework for the user-centric evaluation of recommender systems, adapted from Knijnenburg et al. [6]

2. TOPICS
2.1 Hypotheses
Trying to find out whether your recommender system is "good" for users is not a workable goal in user-centric evaluation. Instead, researchers need to operationalize a set of hypotheses: testable predictions about how a recommender system influences the user. This part of the tutorial teaches the principle of *ceteris paribus* as a way to single out the effects of specific system aspects on the user experience.

2.2 Participants
Many researchers believe that a recommender system can be evaluated with a handful of willing colleagues or students. This part of the tutorial instead makes the case that systems should be

RecSys'12, September 9–13, 2012, Dublin, Ireland, UK.

tested with participants sampled from the target population (i.e. the target audience of the system). It also discusses the typical sample size needed to allow for statistical inferences (which is usually much larger than a handful).

2.3 Testing A vs. B

Although it may make intuitive sense to test the user experience of a recommender system in a holistic fashion, such "test everything at once" evaluations cannot discern the specific causes of the user experience. Good user experiments instead try to single out the effects of specific aspects of the system. To single out the effect of an aspect, one needs to *manipulate* that aspect by creating two or more *conditions* (versions of the aspect). The tutorial covers adequate manipulations, and discusses the pros and cons of between-subjects (i.e. each participants gets to see only one condition) and within-subjects (i.e. each participant gets to see all conditions) experiments.

2.4 Measurement

Measuring user behavior is insufficient to make inferences about the user experience. Behavior is highly context-dependent and difficult to interpret. Subjective valuations, gathered through questionnaires, typically provide a more robust measurement of the users' experience with the recommender system. Moreover, subjective evaluations are better predictors of longer-term system goals such as adoption and user retention.

This part of the tutorial teaches the art of creating questionnaire items, typically presented as statements to which users can agree or disagree on a 5- or 7-point scale. Currently, researchers typically use one such questionnaire item for each concept (e.g. satisfaction, perceived control, understandability) that they want to measure. This tutorial instead makes the case for creating multi-item measurements for each concept. It presents *factor analysis* as a statistical method to turn such multi-item measurements into robust unidimensional scales.

2.5 Analysis

Statistical analysis of user-centric research typically involves correlations, t-tests and linear regressions. This part of the tutorial presents *structural equation models* as a more sophisticated and modern statistical method to make causal inferences. Structural equation models can test complex causal structures, such as whether a certain manipulation (e.g. a different algorithm) has a significant influence on users' perceptions (e.g. perceived recommendation quality), and whether this perception in turn influences their experience (e.g. system effectiveness), and behavior (e.g. item ratings).

2.6 Evaluation framework

The tutorial concludes by returning to the Knijnenburg et al. [6] evaluation framework. This framework for the user-centric evaluation of recommender systems can be used to develop causal hypotheses, to select and construct subjective measures, and to integrate new and existing user-centric research on recommender systems.

3. CONCLUSION

If you work on recommender systems—as a system developer, an algorithms researcher, or a user interface designer—user-centric evaluations are the way to go. This tutorial presents user experiments as an essential skill in uncovering *how* and *why* the user experience of recommender systems comes about.

4. REFERENCES

[1] Adomavicius, G. and Tuzhilin, A. 2011. Context-Aware Recommender Systems. *Recommender Systems Handbook*. F. Ricci, L. Rokach, B. Shapira, and P.B. Kantor, eds. Springer US. 217–253.

[2] Bollen, D., Knijnenburg, B.P., Willemsen, M.C. and Graus, M. 2010. Understanding choice overload in recommender systems. *Proceedings of the fourth ACM conference on Recommender systems* (Barcelona, Spain, 2010), 63–70.

[3] Knijnenburg, B.P., Reijmer, N.J.M. and Willemsen, M.C. 2011. Each to his own: how different users call for different interaction methods in recommender systems. *Proceedings of the fifth ACM conference on Recommender systems* (Chicago, IL, 2011), 141–148.

[4] Knijnenburg, B.P., Schmidt-Thieme, L. and Bollen, D.G.F.M. 2010. Workshop on user-centric evaluation of recommender systems and their interfaces. *Proceedings of the fourth ACM conference on Recommender systems* (New York, NY, USA, 2010), 383–384.

[5] Knijnenburg, B.P. and Willemsen, M.C. 2009. Understanding the effect of adaptive preference elicitation methods on user satisfaction of a recommender system. *Proceedings of the third ACM conference on Recommender systems* (New York, NY, 2009), 381–384.

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

[7] Knijnenburg, B.P., Willemsen, M.C. and Kobsa, A. 2011. A pragmatic procedure to support the user-centric evaluation of recommender systems. *Proceedings of the fifth ACM conference on Recommender systems* (New York, NY, USA, 2011), 321–324.

[8] Konstan, J. and Riedl, J. 2012. Recommender systems: from algorithms to user experience. *User Modeling and User-Adapted Interaction*. 22, 1 (2012), 101–123.

[9] McNee, S.M., Riedl, J. and Konstan, J.A. 2006. Being accurate is not enough. *CHI '06 extended abstracts on Human factors in computing systems - CHI '06* (Montreal, Quebec, Canada, 2006), 1097–1101.

[10] Pu, P., Chen, L. and Hu, R. 2012. Evaluating recommender systems from the user's perspective: survey of the state of the art. *User Modeling and User-Adapted Interaction*. 22, 4 (2012), 317–355.

[11] Willemsen, M., Bollen, D. and Ekstrand, M. 2011. UCERSTI 2: second workshop on user-centric evaluation of recommender systems and their interfaces. *Proceedings of the fifth ACM conference on Recommender systems* (New York, NY, USA, 2011), 395–396.