# Evaluating the Effectiveness of Stereotype User Models for Recommendations on Mobile Devices

Béatrice Lamche, Enrico Pollok, Wolfgang Wörndl and Georg Groh

Technische Universität München,
Boltzmannstr. 3, 85748 Garching, Germany
{lamche,pollok,woerndl,grohg}@in.tum.de
http://www.11.in.tum.de

**Abstract.** Mobile recommender systems have been proven as a promising approach in mobile scenarios to support the decision making process of users by suggesting beneficial items in a certain mobile context. The main goal of this paper is to examine whether a stereotype user model leads to better recommendations as part of such a system. For this purpose, we developed and tested a prototype for a shopping scenario. Research on fashion stereotypes led to a user model containing ten different stereotypes. The stereotype classification is performed by computing the proximity of each stereotype to the user's properties. Results of a user study show that a user model based on stereotypes generates better results than a recommender system without a stereotype-based user model. Moreover, stereotype-based user models allow personalized recommendations right away thus contributing to alleviating the cold start problem.

**Keywords:** mobile recommender systems, stereotypes, user modeling

## 1 Introduction

Mobile recommender systems support the decision making process of users by providing suggestions for items that are of potential use for them in a certain mobile context [1]. Stereotype user modeling was one of the earliest approaches to user modeling and personalization in general [2]. A stereotype-based system maps the individual features for the recommendation process to one of several equivalence classes, whose profiles are then used for computing the recommendations. Stereotypes are usually organized in a directed acyclic graph to allow for generalizations. Each stereotype corresponds to a certain set of features characteristics. If the characteristics of users change they may be reassigned to a different stereotype. In order to match a stereotype to a person, the system needs to have specific *triggers* - events that signal the appropriateness of a particular stereotype and in turn activate it. For one person, several stereotypes can be active. Once activated, the characteristics of the stereotype are incorporated into the user model [2]. Several approaches for constructing user models exist. One approach are *keyword user profiles* which usually extract several keyword vectors from a specific source (e.g. the browser history of the user) using different weighting schemes or algorithms. The user's explicit and implicit feedback is used in order to build the user profile [3].

Examining related research, most mobile recommender systems do not explicitly state the user model used behind their recommendation algorithm (e.g. [4]). It may be simple or implicitly part of the recommendation algorithm. In order to provide personalized mobile recommendations even in the cold start phase, a user modeling approach based on stereotypes is suitable. most people can be associated with a specific style that barely changes (e.g. *casual* vs. *elegant*), so that stereotypes can be easily predefined and an already existing user data base is not required. Moreover, the use of a stereotypical user model allows for a quick characterization of users, particularly important for a mobile scenario. So far, no research was found which tried to combine a stereotypical user model with a recommender system on a mobile device. This work will therefore examine the effectiveness of a mobile recommender system with a user model based on stereotypes. The main goal of this paper is to examine whether a stereotype user model leads to better recommendations as part of a mobile recommender system. The rest of the paper is organized as follows: We first introduce important foundations of user modeling and summarize related work. Next, we explain our prototype. We then present the results of our user study that showed that the recommender system using a stereotype user model performed overall better. We close by suggesting opportunities for future research.

## 2 Designing the Prototype

The scenario in mind is that of a fashion recommender system on a mobile device. Going shopping is an exploratory scenario and the users most often do not have a specific item in mind. We therefore develop a system that delivers recommendations right from the beginning without having to specify a search query. Since the system is used in a mobile environment, the mobile context such as the user's current location and time should be considered for the calculation of personalized recommendations. To be more precise: When users are going to town to look for clothing items and start the application, the system should recommend items of open stores nearby suitable to the user's taste right away and provide information about these items and corresponding stores.

There is little academic research on stereotypes for fashion styles. Therefore our knowledge on publicly perceived stereotypes was limited to information found on the world-wide web, e.g. [5]. We compared the most frequently classified stereotypes based on their given definition and finally identified the following ten fashion stereotypes: *Indie/Hipster*, *Emo*, *Preppy*, *Gothic*, *Urban*, *Athlete/Jock*, *Skater*, *Girly*, *Classy* and *Mainstream.* For the allocation of items to stereotypes we use a *weighted keywords* approach (see *section 1*). Out of the features identified for the various stereotypes, a limited set of attributes consisting of colors, brands and general descriptions for the clothing was identified. Each stereotype has manually been given a rating on a scale of 0 to 10 for each attribute, representing the *weight* to which the feature is related to the stereotype.

The application was written for the Android API version 19 and supports all devices running Android API version 8 or higher. The first screen of the application is a form in which users are asked to provide the data necessary for determining their stereotype (such as age, gender, profession and music taste). The music taste is taken into account because studies found out that it is highly related to the individual fashion style [6]. The user profile thus contains a user ID and a stereotype that is based on the users gender, age, job and music taste. After filling out the form, the application computes the three most relevant stereotypes based on the information that has previously been provided by the user. Each stereotype has been given a weight for all available age groups, jobs and music styles. The stereotype algorithm iterates through all stereotypes available and adds up the likelihood that this stereotype has for each of the properties age, job and music. The resulting three stereotypes are presented to the user in a picture. An extract of the two corresponding user interfaces can be seen in *figure 1*. As soon as the user selects the preferred stereotype, stereotype-based recommendations are calculated and shown in a grid view.

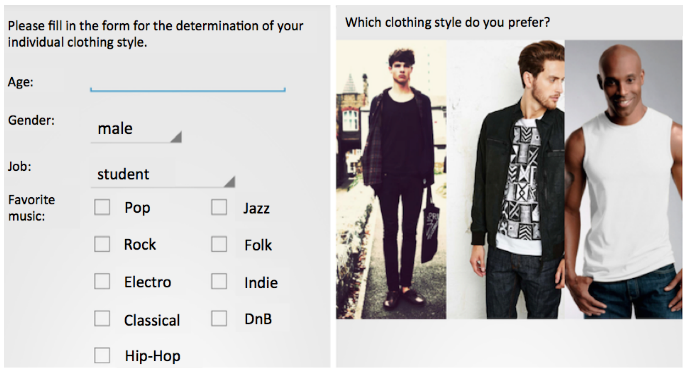


**Fig. 1.** The stereotype determination interfaces.

The *recommendation* algorithm sorts the items by their expected interest for the user. It first gets all attributes and their values for the active stereotype and then scans each item for the attributes color, brand and description. If the checked item contains one of the stereotype attributes, the specific *attribute weight* is added to the proximity measure. All weight values for the found attributes are thus added up and then divided by the number of found attributes. The result is a value for each clothing item which indicates the expected interest for a user with the selected stereotype. These values will be added to a map, sort in descending order and then presented as clothing recommendations to the user. It is worth noting that we give found brand names double the weight compared to other attributes, as we found out during testing that they provide the most reliable indicator for the attractiveness of a clothing item to a person. The user can scroll down in the recommendations view as long as necessary.

An implemented drop-down menu allows for filtering the results. Users can select or exclude specific values of features such as type of clothing, color, brand or price. A text field above the results is always visible, listing all the filters that have already been set. Clicking on a recommendation opens a new screen and details about price, colors, brand, images and the stores that sell them are listed. Once users have found an item they would like to purchase they can select it and the recommendation process terminates.

## 3 Evaluation

The main goal of the evaluation is to find out whether personalized recommendations can be improved through the use of stereotypes. To keep the number of testers at a reasonable size the study was designed as *within-subject*, one group of people tested both variants. The first system used in the user study is our developed mobile recommender system that uses a stereotypical user model. The user's stereotype is determined as described above. To successfully test the developed system, we need to establish a *baseline* to compare against. The second system that is tested is therefore a mobile recommender system without this stereotypical user model, or more clearly without any other algorithm supporting it but the users can still filter the results based on preferred features. The complexity of the experiment is kept low by asking users to choose one item they like for each approach. A potential user bias by using one approach before the other and thereby being aware of the choices available, is reduced by not making the user aware which approach is currently used and randomly switching the order of execution. All variables other than the algorithm used are kept fix. After having performed the task for each approach, candidates are asked to fill out a demographic questionnaire and to rate statements about the system design, the perceived ease of finding information and effort required to use the system, the usefulness of the system, the perceived accuracy of the suggestions, the satisfaction with the user's choice and intention to actually buy the product and reuse the system.

For the study, *participants* using mobile applications or showing an interest in using the described application were recruited. The user study finished with 32 participants, 27 male and 5 female, with an average age of 28 years, ranging from 22 to 54.

The *data set* used for this study was extracted from the now deprecated Google API for Shopping to retrieve the clothing item data and the Google Places API to retrieve information about shopping stores. The raw information from the API was rather limited with most information having to be extracted from the item and store description. To generate the data set of clothing items, the Shopping Search API was queried for keywords associated with types of clothing (e.g. simply 'dress') without any adjectives, to avoid leaning into a particular style as much as possible. The dataset built contains 668 different clothing items of 263 different brands. Items were associated with the following features: an id, one of 13 types of clothing, one of 15 colors, the price, the sex, a description and the link to an image of the item.

The *analysis* of the results is based on the user evaluation framework as described by Chen and Pu [7]. The data was analyzed using averages, standard deviations and student's t-test for determining distribution differences. A one-tail paired t-test was performed to calculate the p-value. *Table 1* shows the means for the most important metrics of the two systems, the standard deviation, as well as the p-value.

**Table 1.** A comparison of the user study's results.

| | stereotype mean | stdev | baseline mean | stdev | p value |
|---|---|---|---|---|---|
| objective accuracy | 0.47 | 0.34 | 0.32 | 0.34 | 0.036 |
| perceived accuracy | 3.5 | 0.53 | 2.6 | 0.52 | 0.00036 |
| time consumption | 47.82 s | 35.83 | 64.26 s | 33.68 | 0.001 |
| perceived effort | 57.9 % | - | 42.1 % | - | - |

*Objective accuracy* refers to the estimate of how likely it is that the user will select an item from a ranked list. The system sorts items according to their expected use, so each successive item in a list should be less likely to be selected by the user with an exponential decay. Objective accuracy can be measured using the *R-Score* which is based on the assumption that the value of a recommendation declines exponentially with the position of an item [8]. A higher *R-score* refers to a better ranking of the item. Calculating the *R-score* for the selected items leads to a mean of 0.47 ($\sigma$ = 0.34) in stereotype mode and 0.32 ($\sigma$ = 0.34) in the baseline. So we conclude that the stereotype-based approach is significantly more accurate at a 0.05 level (p-value = 0.036).

To determine the *perceived accuracy*, users were asked whether they would purchase the item they last selected. The answers to the question were put on a five-point Likert scale (from 1, strongly disagree to 5, strongly agree with 3 being neutral). The stereotype iteration was rated better in a median of 3.5 ($\sigma$ = 0.53), compared to the baseline which was rated in a median of 2.6 ($\sigma$ = 0.52), being statistically significant at a 0.05 level (p-value = 0.00036).

*Objective effort* is measured in terms of the time a user needs to find a satisfying item and go through the cycles. On average users took less time to complete the task when supported by a stereotype-based user model, in particular 47.82 seconds ($\sigma$ = 35.83) versus 64.26 seconds for the baseline ($\sigma$ = 33.68). The t-test confirms the difference of the samples at a significance level of 0.05 with a p-value of 0.001.

*Perceived effort* refers to the difficulty a subject has during the performance of the task in terms of information processing. 57.9 % of the participants preferred the stereotype round and 42.1% preferred the baseline.

The analysis of the open questions showed that the participants were overall very satisfied with the design of the application (62% with 28% feeling neutral about it) and 91% understood the usage of the application quickly.

## 4 Conclusion and Future Work

This work investigated the development and effectiveness of recommendations provided by a mobile prototype in the domain of fashion. Recommendations are generated by exploiting a stereotype user model and combining it with a mobile recommender system. The goal of the prototype was to provide the means to measure the effectiveness of its recommendations. 10 fashion stereotypes were identified and included in the user model. The app offers the user the possibility to criticize clothing items by clothing type, color, brand and price. Finally, a user study was conducted among 32 participants. The recommendation system using a stereotype user model performed overall better. Future research could use a more advanced approach to user modeling than the static stereotype user model, e.g. a form of an overlay model or a semantic network. Modeling aspects such as the mobile context may also lead to improved results, as well as a more sophisticated recommendation algorithm based on a dynamic user model which is able to learn how users behave.

## References

1. Lamche, B., Trottmann, U., Woerndl, W.: Active Learning Strategies for Exploratory Mobile Recommender Systems. In: Proc. Decisions@CaRR workshop, ECIR, Amsterdam, Netherlands (2014)
2. Rich, E.: User Modeling via Stereotypes. In: Cognitive science, vol. 3(4), pp. 329–354 (1979)
3. Gauch, S., Speretta, M., Chandramouli, A., Micarelli, A.: User profiles for personalized information access. In Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. Lecture Notes in Computer Science, vol. 4321, pp. 5489. Springer, Berlin Heidelberg (2007)
4. Gavalas, D., Kenteris, M.: A web-based pervasive recommendation system for mobile tourist guides. In: Personal and Ubiquitous Computing, vol. 15(7), pp. 759–770. Springer, London (2011)
5. Angerosa, O.: Fashion stereotypes (2011), `http://people.rit.edu/~ona8039/wf/stage3/index.html`
6. Na, Y., Agnhage, T.: Relationship between the preference styles of music and fashion and the similarity of their sensibility. In: International Journal of Clothing Science and Technology vol. 25(2), pp. 109–118 (2013)
7. Chen, L., Pu, P.: Interaction design guidelines on critiquing-based recommender systems. In: User Modeling and User-Adapted Interaction, vol. 19(3) , pp. 167–206. Springer, Netherlands (2009)
8. Breese, J. S., Heckerman, D., Kadie, K.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the 14th conference on Uncertainty in artificial intelligence, UAI98, pp. 43–52, San Francisco, CA, USA (1998)