Evaluating Stereotype and Non-Stereotype Recommender Systems

Nourah A. ALRossais
University of York, UK
nar537@york.ac.uk

Daniel Kudenko
University of York, UK
National Research University Higher School of Economics,
St Petersburg, Russia
JetBrains Research, St Petersburg, Russia
daniel.kudenko@york.ac.uk

ABSTRACT

Stereotype-based user modeling was proposed by Elaine Rich in 1979 and has been applied to recommender systems on numerous instances since its conception. The key motivations for application of stereotyping in user modeling are resolution of the new user problem and space efficiency. Several claims have been made in the literature related to the effectiveness of stereotyping but only a few studies have validated them empirically. Furthermore, to the best of our knowledge, there has been no empirical study of item-based stereotype models for recommender systems. Our research empirically substantiates the efficacy of using stereotypes in item modeling and user modeling when compared with non-utilization of stereotypes. The empirical evaluation was performed with a state-of-the-art machine learning algorithm (gradient boosted decision forests) applied to two datasets integrating MovieLens, IMDb and TMDB movie data.

KEYWORDS

Recommender Systems; Stereotypes; New Item; User-Item Modeling; Performance Evaluation

ACM Reference Format:
Nourah A. ALRossais and Daniel Kudenko. 2019. Evaluating Stereotype and Non-Stereotype Recommender Systems. In *Proceedings of Knowledge-aware and Conversational Recommender Systems (KaRS) Workshop 2018 (co-located with RecSys 2018)*. ACM, New York, NY, USA, 6 pages.

1 INTRODUCTION

Elaine Rich was the first to propose the utilization of stereotypes in user modeling and recommender systems as a method for the resolution of the new user problem [14]. A stereotype depicts a collection of attributes that are relevant for a collection of users [12] and represent the "frequently occurring characteristics of users" [14]. A stereotype may or may not be a precise representation of the user group or any specific group member, but it may simply be an estimation of certain characteristics of the group.

The fundamental motivation for applying stereotyping is to provide personalization while having insufficient information about new users, by assigning them to a stereotype. In this context, stereotypes are usually regarded to be similar to other models [12].

Knowledge-aware and Conversational Recommender Systems (KaRS) Workshop 2018 (co-located with RecSys 2018), October 7, 2018, Vancouver, Canada.

The objective was that recommendations could be presented to new users without the requirement to gather a set of ratings from the users for the purpose of user model training. Moreover, Rich mentioned that an additional benefit of stereotyping is its space-efficiency, because the characteristics that are applicable to several users are required to be stored only once and can be applied to all members belonging to a stereotype. Stereotyping has been deployed in user modeling (see e.g. [3, 5, 10, 12–15]), and to the best of our knowledge there has been no application of the concept of stereotypes to item modeling, apart from our previous study where we developed techniques for an item-based recommender system employing stereotypes based on item characteristics [1].

Furthermore, several claims have been made in the literature related to the effectiveness of stereotyping in user modeling, but only a few studies have validated them empirically [12]. A performance comparison between item modeling with and without stereotypes has not been carried out to date.

1.1 Contribution

In previous work [1], we proposed a technique for utilizing stereotypes in item modeling. However, that study did not include a performance comparison of item modeling with and without the application of stereotypes. This paper, for the first time, provides a comparative analysis of the performance of stereotype-based item modeling with non-stereotype-based item modeling. Furthermore, this paper contributes to the literature by presenting experimental results comparing the effectiveness of user models with and without the application of stereotypes.

Specifically, the presented research addresses the following research questions:

(1) Using a state-of-the-art machine learning method, do recommendations based on stereotype-based user modeling have better accuracy than those not utilizing it in user modeling?
(2) Do item-based recommender systems have improved accuracy when they employ stereotypes for item modeling when compared to non-utilization of stereotypes?

For evaluation, we use two integrated datasets combining MovieLens, IMDb and TMDB information.

The remainder of this paper is organized as follows: Section 2 summarizes related work. A preliminary design of the prediction algorithm for building manual stereotype-based item model offline is presented in Section 3. Experimental results are presented in Section 4 along with discussion in Section 5. Lastly, Section 6 presents concluding remarks alluding to future work.

2 RELATED WORK

The Grundy system developed by Rich [14] is a pioneering work in the field of stereotype-based recommender systems. It is the first system of its kind to recommend items to users and uses a hierarchical structure for the creation of stereotypes to make recommendations to users. The results of experiments conducted by Rich revealed that users were more satisfied with recommendations made by Grundy when compared with ones generated randomly. However, the empirical evaluation did not provide ample evidence to support the case of a stereotype-based user model over individualized user modeling as the latter type of modeling was not implemented and evaluated [12].

Even though resolution of the new user problem and achieving space efficiency [14] are the key objectives of stereotyping, authors in [5] stated that stereotyping offers yet another advantage as it renders "*knowledge acquisition and debugging to occur in a highly modular and incremental way, thus facilitating the job of the knowledge engineer (which turns out to be especially hard in the particular domain of user modeling)*".

An evaluation of Personal Program Guide (PPG) by Ardissono et al. [3] showed that overall user modeling in PPG consisting of three user modeling components displayed better performance but performance of stereotypical user model was poor which might be attributed to lack of completeness of knowledge base in stereotypes and inappropriate assignment of stereotypes. Gena and Ardissono [8] noted that "*stereotypical knowledge does not correctly handle users matching different lifestyles in different aspects of their behaviours, because of the major selectivity of the personal data in the classification of users, in spite of interests*". Thus, Ardissono et al. believed that stereotype-based user models are useful when interacting with users [3], even when they lead to weaker recommendation performance.

Kurapati and Gutta proposed in the domain of TV personalization that a stereotype-based approach to recommendations displays similar performance to the individualized recommender system developed previously by them [11], even though the comparison was performed only for a single user (User K). The estimated error rate for the individualized user model of User K was 22% while it was 13% for the stereotype-based model. However, the study did not include a detailed and direct comparison between a stereotype-base user modeling and a single component approach, neither at individual user level nor averaged over all users [12].

Krulwich while evaluating LIFESTYLEFINDER found that a random recommendation approach was outperformed by a stereotype-based system. Krulwich noted that "*the ability to operate on a small amount of innocuous information comes at the expense of the accuracy that the system is able to achieve*" [10]. Yet, this claim is unsupported as no direct comparisons between stereotype-based and individualized approach was made. Lock [12] in a rare empirical study on the performance of a stereotype-based approach (in the context of the development of the stereotype-based recommender system GERMANE) found that, on average, stereotype-based user modeling is comparable to a single component approach. However, Lock also noted that stereotyping can be effectively employed in recommender systems for known users from whom relevance feedback has been collected and this enhancement in flexibility will not lead to lower performance. It was added by Lock [12] that online user evaluation is required to substantiate the advantages of stereotype-based user modeling over single component user models.

The shortage of empirical studies into the performance advantages of stereotype-based recommendations is a key gap in this field. To support this, the authors of [15] stated that "*experiments must be conducted to compare the results with and without the use of stereotypes for the same users and data*". However, they added that "*such experiments are not easily carried out*".

3 STEREOTYPE BASED RECOMMENDATION ALGORITHM

To address the research questions, in previous work [1] we proposed an algorithm that assigns users and items to stereotypes through our user and item preference model (i.e. P_u, P_i) as well as a clustering technique. In this work, we will validate the algorithms empirically.

The stereotypes in this study are of a 'double stereotype' nature which implies that the information that was used to recommend an item to a target user is also influencing the allocation of items to stereotypes. Double Stereotypes were suggested by Chin [6] in terms of user-based stereotypes for information filtering where the information that a user chooses to view is also examined as a way to allocate users to stereotypes. We apply a similar concept for item-based stereotypes. Sections 3.1 and 3.2 explain our proposed algorithm in more detail.

3.1 User-Based Stereotype Recommendation

Let $P_u(i)$ be the preference function of user $u \in U$ in item $i \in I$ and $P_{US}(i)$ the preference function of User Stereotype (US) in which user u is a member, in item $i \in I$ then:

$$P_u(i) = P_{US}(i) \qquad \forall i \in I$$

Since user u can belong to multiple User Stereotypes (USs), the recommendation setting is the sum of weighted preference functions of User Stereotype (USs) to that item i, given by

$$P_u(i) = \sum_{S \in US(u)} w_S . P_{US}(i) \tag{1}$$

Where US(u) is the set of user stereotypes for which user u is a member, w_S is weight of preferences functions as defined by an expert in the field and

$$\sum_{S \in US(u)} w_S = 1,\ w_S \in [0, 1] \qquad \forall S \in US(u)$$

3.2 Item-Based Stereotype Recommendation

Let $P_i(u)$ be the preference function of item $i \in I$ in user $u \in U$ and $P_{IS}(u)$ the preference function of Item Stereotype (IS) in which item i is a member, in user $u \in U$ then:

$$P_i(u) = P_{IS}(u) \qquad \forall u \in U$$

Since item i can belong to multiple Item Stereotypes (ISs), the recommendation setting is the sum of weighted preference functions of Item Stereotype (ISs) to that user u, given by

$$P_i(u) = \sum_{S \in IS(i)} w_S . P_{IS}(u) \tag{2}$$

Where IS(i) is the set of item stereotypes for which item i is a member, w_S is the weight of preference functions as defined by an expert in the field and

$$\sum_{S \in IS(i)} w_S = 1, \; w_S \in [0, 1] \qquad\qquad \forall S \in IS(i)$$

4 EXPERIMENTAL EVALUATION

This Section details the two datasets used in our investigation into the performance of stereotype-based user/item models. They differ in terms of the number of stereotypes assigned to users/items. We run a direct comparison between user/item and stereotype based-user/item models as the same datasets are used to construct and evaluate both model types. The main concern of this paper is to compare the single user/item model and stereotype-based approaches. The performance levels reported for both experiments are for known users (i.e. users from which training feedback has been obtained).

4.1 Dataset

The MovieLens dataset is quite popular among the research community. GroupLens Research has collected and made available different versions of the MovieLens dataset. For the purpose of this study, experiments were conducted on two different versions of MovieLens: (1) MovieLens 1 Million dataset and (2) MovieLens 20 Million dataset [9].

Demographic features (e.g. age, gender, occupation) of users were extracted from the MovieLens 1 Million dataset and supplementary item features were extracted from kaggle https://www.kaggle.com/ that is based on the TMDB dataset. The combined dataset contains 6,040 users, 3,827 movies, 1,000,209 ratings, 35,052 cast members, and 28,541 crew members along with other movie data and user generated features like keywords. We refer to this dataset as *Dataset 1* in the remainder of this paper.

Unlike the MovieLens 1 Million dataset, the MovieLens 20M dataset does not contain demographic features. Instead, we interpreted a user's average rating per item feature as a user feature. Precisely, in a previous work [2], we integrated the MovieLens 20M dataset and the IMDb dataset and generated a dataset from this integrated data. This dataset included a feature vector that represent useful information about users and movies that is not explicitly contained in the raw data. More specifically, our dataset contains information about user interest in movie genres, actors, etc. The dataset is different from other data in that the interest of users in movie features are calculated implicitly from their overall ratings rather that explicitly asking user his or her preferences.

A total of 20M ratings applied to 27,242 movies by 138,000 users, where each user rates at least 20 movies, were extracted. In our experiment, we applied our algorithm to 150,567 ratings applied to 9734 movies by 1000 users. There is a wide variance in performance between the users as each user has a different set of interests. We refer to this dataset as *Dataset 2* in the remainder of this paper.

4.2 Evaluation Procedure

In this Section, we present the results of experimental studies and investigate the answer of the following research question:

"Does the use of stereotypes help to improve accuracy over not using stereotypes?"

We first have to build a user model and an item model. These models can be computed automatically by applying machine learning techniques to the ratings given by the user to the items viewed.

A machine learning algorithm (gradient boosted decision trees [7]) was deployed to build a user/item and a user/item-based stereotype models. In our experiments, the baseline for evaluating stereotyping is a single user/item model constructed using the same machine learning algorithm. This makes it possible to directly compare the individualized and stereotype-based models.

Experiments were conducted offline by considering two different predictive accuracy measures: (1) Mean Absolute Error (MAE) and (2) Mean Squared Error (MSE). MAE and MSE are appropriate metrics for assessing models that output scores with similar range and distributions and have been used in previous studies [3, 16]. User satisfaction is not measured at all in our experiments.

For our investigation, we performed two experiments using distinct settings to generalize findings and prove that the proposed stereotype approach is applicable to any user attributes whether subject to change or not. In Table 1, the experimental settings are presented.

Table 1: Experiment Settings

Setting	*Exp on Dataset 1*	*Exp on Dataset 2*
Dataset	MovieLens 1M	MovieLens 20M
User Features	Demographic (not subject to change)	Preferences (subject to change)
Splitting Data	Train/validate/test (no overlap)	k-fold cross validation
Machine Learning algorithm	gradient boosted decision	bagged decision tree
No. of stereotypes assigned to user	2 -age (7 exclusive groups) -gender (2 exclusive groups)	varies between 1 and 477 representing user preferences for: -genres (28 groups) -actors (248 groups) -directors (101 groups) -writers (100 groups)
No. of stereotypes assigned to item	varies between 1 and 647 -genres (23 groups) -cast (132 groups) -crew (192 groups) -keywords (300 groups)	varies between 1 and 477 -genres (28 groups) -actors (248 groups) -directors (101 groups) -writers (100 groups)

4.3 Experiment on Dataset 1

To build the user model, we treat movie ratings of a user as the label of training examples, and the features of a movie (e.g. genre, actor, etc.) form the training example itself. The user model is the output of the applied machine learning method when fed with this training data.

In the case of user-based stereotypes, we train a stereotype model in the same way as the user model, but now using the combined training data from all users that fit the stereotype as indicated in Equation (1). We split *Dataset 1* by items to train the user model (i.e. every movie is in exactly one of train (70%), validate (10%), or test (20%) sets; there is no overlapping). We repeat the process five times using simple random sampling to ensure unbiased results. Performance was averaged over all phases.

Input to the user model is a matrix consisting of the following item features: genres, id, adult, budget, imdb_id, original_language, popularity, production_companies, production_countries, release_date-converted into release_year and release_month, revenue, runtime, spoken_languages, title, vote_average, vote_count, keyword, cast and crew.

Input to the user-based stereotype model is the same matrix used in the user model, but here we combined training data from all users using gender and age to define user-based stereotypes. As Equation (1) uses weighted preferences of user-based stereotypes, we experimented with different weights for the age and gender stereotypes over all five samples to ensure unbiased results. Table 2 summarizes the average results. Overall, there is no significant impact on accuracy as we change weights and the best result is achieved when we assign a weight of age and gender to 0.2 and 0.8 respectively.

Table 2: Different weights for User-based stereotype model

Age weight	Gender weight	MAE	MSE
0.1	0.9	0.79133	1.2236
0.2	**0.8**	**0.79131**	**1.2235**
0.3	0.7	0.79147	1.22420
0.4	0.6	0.79180	1.22570
0.5	0.5	0.79200	1.22629
0.6	0.4	0.79329	1.23012
0.7	0.3	0.79403	1.23183
0.8	0.2	0.79610	1.23720
0.9	0.1	0.79697	1.23951

To build the item model, we treat movie ratings of a user as the label of training examples, and the features of a user (gender, age, occupation) form the training example itself. The item model is the output of the applied machine learning method when fed with this training data.

In the case of the item-based stereotypes, we train a stereotype model in the same way as the item model, but now using the combined training data from all items that fit the stereotype as indicated in Equation (2). We split *Dataset 1* by users to train the item model (i.e. every user is in exactly one of train (70%), validate (10%), or test (20%) sets; there is no overlapping). We repeat the process five times using simple random sampling to ensure unbiased results. Performance was averaged over all phases.

Although cross-validation is used to estimate generalization performance, it is not always appropriate for recommender system evaluation. Random assignment of items to folds was inappropriate as indicated by other authors in their research [4]. Billsus and Pazzani found that a user's ratings of an item is influenced by the items they already saw and rated. Also, the ordering of items is critical. Therefore, we preserved the chronological ordering of the relevance feedback data by sampling every user in either train, validate or test sets.

Input to the item model is a matrix consisting of the following user features: gender, age, occupation and zip code.

Input to the item-based stereotype model is the same matrix used in the item model, but here we combined training data from all items using genres, cast, crew and keywords to define different item-based stereotypes. The choice of features on which the stereotypes are based has been made using our domain expertise. Equation (2) included weighted preference functions of item-based stereotypes, however, in our experiment, we used uniform stereotype weights for simplicity as assigning weights manually is not practical. Instead, it should be done automatically and we will leave this for future work.

Table 3 summarizes the accuracy of stereotype and non-stereotype based models for *Dataset 1*. The accuracy of user-based stereotype modeling is promising and in line with findings in literature. As for the item models that represent the "preference" of an item for a user, i.e. a mapping of users to preference values, our expectation that this will provide additional useful information for the recommender system was correct. Moreover, designing stereotypes analogously to user stereotypes was even promising as item-based stereotypes achieve an enhancement in accuracy compared to the raw item model.

Table 3: The Accuracy of Stereotype and non Stereotype models for Dataset 1

Model	**MAE**	**MSE**
User Model	0.794	1.234
User-based Stereotype	0.791	1.223
Item Model	0.878	1.450
Item-based Stereotype	0.876	1.449

4.4 Experiment on Dataset 2

This experiment was run and validated with 5-fold and 10-fold cross validation techniques to avoid over-fitting. The reason for using a different algorithm and different validation methods in this experiment is to demonstrate the impact of stereotypes on recommendation performance irrespective of the methods and algorithms.

To build the user model, we treat movie ratings of a user as the label of training examples, and the features of a movie (e.g. genre, duration, etc.) form the training example itself. The user model is the output of the applied machine learning method when fed with this training data.

In the case of user-based stereotypes, we train a stereotype model in the same way as a user model, but now using the combined training data from all users that fit the stereotype as indicated in Equation (1).

Input to the user model is a matrix that consists of the following item features: genres, release_year and duration.

Input to user-based stereotype model is the same matrix used in the user model, but here we combined training data from all users using preferences of genres, preferences of actors, preferences of directors and preferences of writers. The choice of features used to create the stereotypes was based on our domain expertise. Equation (1) indicates weighted preference functions of user-based stereotypes, however, in our experiment, for simplicity, we used uniform stereotype weights as we assume all preferences are equally important (which may not be the case).

As noted in Section 4.1, the MovieLens 20 Million dataset does not contain any user features. Hence, we implicitly calculated the interest of a users in given movie features, in the form of average rating, and treat this as a user feature. Details are in our previous work [2]. To build the item model, we treat movie ratings of a user as the label of training examples, and the interest of a user in various genres form the training example itself. The item model is the output of the applied machine learning method when fed with this training data.

In the case of the item-based stereotypes, we train a stereotype model in the same way as the item model, but now using the combined training data from all items that fit the stereotype as indicated in Equation (2).

Input to the item model is a matrix consisting of the user features corresponding to the user preferences of various genres.

Input to the item-based stereotype model is the same matrix used in the item model, but here we combined training data from all items using genres, actors, directors and writers for the item-based stereotypes. Equation (2) indicates weighted preference functions of item-based stereotypes. However, in our experiment, for simplicity, we used uniform stereotype weights assuming that all item features are equally important.

Table 4: The Accuracy of Stereotype and non Stereotype models for Dataset 2

k-fold	5-fold		10-fold	
Model	**MAE**	**MSE**	**MAE**	**MSE**
User Model	0.779	0.988	0.778	0.986
User-based Stereotype	0.768	0.970	0.769	0.970
Item Model	0.774	0.978	0.774	0.977
Item-based Stereotype	0.742	0.906	0.742	0.905

In Table 4, the accuracy of stereotype and non-stereotype based models for *Dataset 2* is presented. The user-based stereotype modeling achieves better accuracy than the single user model. The same applies to the item-based stereotype models when compared to the item model. In this experiment, we achieved even better accuracy for item-based stereotype as compared to user-based stereotype. This indicates that stereotype-based item modeling is a promising approach.

5 DISCUSSION

Our research questions have been addressed with the experimental results in Section 4. Two scientific experiments conducted on different datasets in the movie domain demonstrated that the performance levels of stereotype-based user models are slightly better than the single-component models for an existing user.

Moreover, a performance comparison of item modeling with and without stereotypes has been shown for the first time. The generated item-based stereotype models are models of the target-market for a given group of items, i.e. denoting how much an item "likes" a user (rather than the other way round as is done in user modeling). The results are promising for solving the new item problem.

Nevertheless, it may not be effective and efficient for a recommender system to manually define stereotypes from a restricted list of item features such as size, sold quantity, price, etc. Another way could be an automatic and dynamic generation of stereotypes from a collection of features where, for example, in one case feedback, price and similarity are utilized to group products and in another case quantity sold, click-through rate and popularity could be employed. Thus, automated stereotype generation better enhance models that are focused on the requirements of the user in order to increase revenue through identification of items which users may find more interesting.

Therefore, to overcome the limitations of manual stereotypes, we intend to develop an automatic item-based recommender system in the next phase of the project.

6 CONCLUSION AND FUTURE WORK

In this work, we evaluated user and item models with and without stereotypes on two movie recommendation datasets, and the results demonstrate the effectiveness of stereotypes in significantly improving the accuracy of recommendations.

In future work, we aim to evaluate our model on other datasets collected from an online business and an online user study.

Furthermore, we intend to develop a hybrid method that combines stereotype-based user and item models to achieve a higher recommendation accuracy.

REFERENCES

[1] Nourah A. AlRossais. 2018. Integrating Item Based Stereotypes in Recom-mender Systems. In *UMAP'18 Adjunct: 26th Conference on User Modeling, Adaptation and Personalization, July 8-11, 2018, Singapore, Singapore.* ACM, 4.

[2] Nourah A. AlRossais and Daniel Kudenko. 2018. iSynchronizer: A Tool for Extracting, Integration and Analysis of MovieLens and IMDb Datasets. In *UMAP'18 Adjunct: 26th Conference on User Modeling, Adaptation and Personalization Adjunct, July 8-11, 2018, Singapore, Singapore.* ACM, 5.

[3] Liliana Ardissono, Alfred Kobsa, and Mark T Maybury. 2004. *Personalized digital television: targeting programs to individual viewers*. Vol. 6. Springer Science & Business Media.

[4] Daniel Billsus and Michael J Pazzani. 1999. A hybrid user model for news story classification. In *UM99 User Modeling*. Springer, 99–108.

[5] Giorgio Brajnik, Giovanni Guida, and Carlo Tasso. 1990. User modeling in expert man-machine interfaces: A case study in intelligent information retrieval. *IEEE Transactions on Systems, Man, and Cybernetics* 20, 1 (1990), 166–185.

[6] David N Chin. 1989. KNOME: Modeling what the user knows in UC. In *User models in dialog systems*. Springer, 74–107.

[7] Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics* (2001), 1189–1232.

[8] Cristina Gena and Liliana Ardissono. 2001. On the construction of TV viewer stereotypes starting from lifestyles surveys. In *Workshop on Personalization in Future TV*.

[9] F Maxwell Harper and Joseph A Konstan. 2016. The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems (TiiS)* 5, 4 (2016), 19.

[10] Bruce Krulwich. 1997. Lifestyle finder: Intelligent user profiling using large-scale demographic data. *AI magazine* 18, 2 (1997), 37.

[11] Kaushal Kurapati and Srinivas Gutta. 2002. Instant personalization via clustering TV viewing patterns. *IASTEDâĂŹs ASC* (2002).
[12] Zoe Lock. 2005. *Performance and Flexibility of Stereotype-based User Models*. Ph.D. Dissertation. University of York.
[13] Jon Orwant. 1994. Heterogeneous learning in the Doppelgänger user modeling system. *User Modeling and User-Adapted Interaction* 4, 2 (1994), 107–130.
[14] Elaine Rich. 1979. User modeling via stereotypes. *Cognitive science* 3, 4 (1979), 329–354.
[15] Bracha Shapira, Peretz Shoval, and Uri Hanani. 1997. Stereotypes in information filtering systems. *Information Processing & Management* 33, 3 (1997), 273–287.
[16] Manolis Vozalis and Konstantinos G Margaritis. 2004. Unison-CF: a multiple-component, adaptive collaborative filtering system. In *International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*. Springer, 255–264.