Rating Prediction with Contextual Conditional Preferences

Aleksandra Karpus[1], Tommaso di Noia[2], Paolo Tomeo[2] and Krzysztof Goczyła[1]
[1]*Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, Gdańsk, Poland*
[2]*Department of Electrical & Electronics Engineering, Polytechnic University of Bari, Bari, Italy*

Keywords: Recommender Systems, Context Awareness, Conditional Preferences, Rating Prediction, Cold-start Problem.

Abstract: Exploiting contextual information is considered a good solution to improve the quality of recommendations, aiming at suggesting more relevant items for a specific context. On the other hand, recommender systems research still strive for solving the cold-start problem, namely where not enough information about users and their ratings is available. In this paper we propose a new rating prediction algorithm to face the cold-start system scenario, based on user interests model called contextual conditional preferences. We present results obtained with three publicly available data sets in comparison with several state-of-the-art baselines. We show that usage of contextual conditional preferences improves the prediction accuracy, even when all users have provided a few feedbacks, and hence small amount of data is available.

1 INTRODUCTION

Exploiting contextual information is considered a good solution to improve the quality of recommendations, aiming at suggesting more relevant items for a specific context (Lombardi et al., 2009). During last decade many context-aware approaches were proposed. However, they usually consider the situation where a lot of data is available. On the other hand, recommender systems research still strive for solving the cold-start problem, namely where we do not have enough information about users and their ratings. For example, matrix factorization methods do not work well in cold start scenarios (Kula, 2015).

Different situations described in the literature are called cold-start problem. Two of them are well-known and have also another name, respectively: new item and new user cold-start problem. Both occur when recommender system is well-established and a lot of ratings are available. When we introduce a new item into such system, in many recommendation algorithms it will not be recommended to users, because of the lack of its history, i.e. user ratings. The same happens when a new user registers into the recommender system. He will not receive interesting recommendations just because the system does not know his preferences yet (Jannach et al., 2010).

However, to the best of our knowledge, a little work was done on the third kind of the cold-start problem, i.e. a new system cold-start problem. The lack of interest in this particular problem could be justified by the facts that it is rather rare and that a company, when runs a new system, does not have enough resources to support research. Nevertheless, this scenario when we do not have may users, items and ratings is very interesting and deserves further consideration.

Besides the well-know cold start problem, we could distinguish the continuous cold-start problem which is characteristic for some specific domains such as tourism or job recommendations (Bernardi et al., 2015). It was noticed that in some domains users never become warm, i.e. we never have many adequate ratings, because a user searches for items very rarely and changes his preferences over time. For example, for young people it is better to sleep in a cheap hostel than in an expensive hotel during a trip, while older people could think the opposite.

In this paper we introduce an algorithm for the rating prediction task in a new system cold-start situations. It is based on a user model called contextual conditional preferences (Karpus et al., 2016) which represents user interests in items in a compact way. We run our experiments on three context-aware data sets publicly available in the Web, i.e. `LDOS-CoMoDa`, `Unibz-STS` and `Restaurant & consumer data`, which are quite small and fit well for our purposes. We confirmed that our algorithm works well in the cold-start scenarios. Because the similarity in the continuous cold-start problem and a new system cold-start problem characteristics, we be-

Karpus, A., Noia, T., Tomeo, P. and Goczyla, K.
Rating Prediction with Contextual Conditional Preferences.
DOI: 10.5220/0006083904190424
In *Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2016) - Volume 1: KDIR*, pages 419-424
ISBN: 978-989-758-203-5

lieve that this method could help to resolve the continuous cold-start problem. However, this is the next step in our research.

The remainder of the paper is constructed as follows. Section 2 briefly introduces contextual conditional preferences. In Section 3 a rating prediction algorithm is presented. The data sets used are described in Section 4. Section 5 provides our evaluation approach and obtained results. Related works are presented in Section 6. Conclusions close the paper.

2 CONTEXTUAL CONDITIONAL PREFERENCES

Contextual conditional preferences (CCPs) introduced in (Karpus et al., 2016) are a compact representation of user interests in items in different situations. This model describes the relations between the context related to the user's ratings and the item content, and consists of a set of conditional preferences.

We define the Contextual Conditional Preference (CCP) as an expression of the form:
$(\gamma_1 = c_1) \wedge \ldots \wedge (\gamma_n = c_n) \mid (\alpha_1 = a_1) \succ (\alpha_1 = a'_1) \wedge \ldots \wedge (\alpha_m = a_m) \succ (\alpha_m = a'_m)$
with γ_i being contextual variables and α_i item attributes, and $c_1, ..., c_n, a_1, a'_1, ..., a_m, a'_m$ being concrete values of these parameters.

The above preference is read as *given the context* $(\gamma_1 = c_1) \wedge \ldots \wedge (\gamma_n = c_n)$ *I prefer* a_1 *over* a'_1 *for* α_1 *and* a_m *over* a'_m *for* α_m. An example of the CCP is shown below.

$$\begin{gathered} dominant\_emo = 7 \wedge decision = 1 \\ \wedge\, end\_emo = 7 \wedge physical = 1 \\ \mid\; genre \in \{1,3,7,8,10,19,21\} \succ genre \in \\ \{6,13,14,17\} \\ \wedge\, actor \in \{1,6,8,15\} \succ actor \in \{2,7,9,10,13\} \end{gathered}$$

It means that for a given context (e.g. decision is 1 - it was a user's decision to watch a certain movie) a user prefers genres with id 1, 3, 7, 8, 10, 19 or 21 to those with 6, 13, 14 or 17 and actors from clusters 1, 6, 8 and 15 to those from clusters 2, 7, 9, 10 or 13 etc.

We distinguish two types of CCPs: individual and general. Individual CCP (ICCP) represents preferences of a single user, while general CCP (GCCP) catches a general trend of interests for all users in a certain contextual situation, i.e. we treat ratings from all users like they were made by one person.

During our experiments we automatically generated CCPs. For more details about the algorithm of the preferences extraction please refer to (Karpus et al., 2016).

3 RATING PREDICTION WITH CONTEXTUAL CONDITIONAL PREFERENCES

Having a concrete user and his context, and wanting to predict his rating for some item, first we need to find the best CCPs (his or general ones) that will be used during a prediction process. In this case, the best preferences are those which are the most similar to the considered context. In order to count a contextual similarity between a CCP p and a current user context $ctx(u)$ we used the following metric:

$$sim(p, ctx(u)) = \sum_{(\gamma_i, c_i) \in p} overlap(ctx(u), (\gamma_i, c_i)).$$

We also used the overlap function defined as:

$$overlap(ctx(u), (\gamma_i, c_i)) = \begin{cases} 1 & (\gamma_i, c_i) \in ctx(u); \\ 0.5 & c_i = -1; \\ 0 & otherwise. \end{cases}$$

The overlap function returns 1 when we are sure that the pair (γ_i, c_i) is contained both in the contextual part of p and in the current user context $ctx(u)$. When it is uncertain, i.e. when the value c_i for the dimension γ_i is equal -1 (the unknown value), it returns 0.5. Otherwise 0 is returned. Please notice, that the current user context $ctx(u)$ is also a set of pairs (γ'_i, c'_i), i.e. the name of the contextual variable and its value.

For an item that we want to predict a rating, we construct a list containing this item and items seen by the user in the context similar to current context in at least some percentage (this value is configurable and could depend on the data set - the configuration for each data set that we used is presented in Table 1).

Identified in the previous step best preferences are used to order constructed list. For each pair of items, we choose the one that has the most similar values for the attributes $attr(i)$ (a set of attribute name and value pairs (α_i, a_i)) with the CCP p. For this purpose we used another similarity measure and overlap function defined as:

$$sim_{cont}(p, attr(i)) = \sum_{(\alpha_i, a_i) \in p} overlap(attr(i), (\alpha_i, a_i))$$

$$overlap(attr(i), (\alpha_i, a_i)) = \begin{cases} 1 & (\alpha_i, a_i) \in attr(i); \\ 0 & a_i = -1; \\ -1 & otherwise. \end{cases}$$

The overlap function used here is quite different from the one used above. In the case of item features it is more crucial to have strict matching. This is the reason why we do not reward the unknown value and why we give penalty for unmatched parameter values.

It should be noticed that we need to compare the similarity of the item attributes with both sides of

Table 1: The configuration for experiments with three data sets.

	LDOS	STS	RC
% of sim.	75	95	75
Other algorithms used	Glob. Avg	Item Avg, Glob. Avg	User Avg, Glob. Avg

Table 2: Basic statistics of three data sets.

	LDOS	STS	RC
# of users	121	325	138
# of items	1232	249	130
# of ratings	2296	2534	1161
Max # of ratings / user	275	175	18
Min # of ratings / user	1	1	3
Avg # of ratings / user	18.98	7.80	8.41
Max # of ratings / item	26	282	36
Min # of ratings / item	1	1	3
Avg # of ratings / item	1.86	10.18	8.93

the preference relation in the current preference statement.

The process of reordering is repeated as long as nothing could be changed. Depending on the final place of the considered item in the list, we compute its rating. If the context is new, i.e. if there is no other item in the list, we rate the current item with some baseline algorithm (it is a configurable option, see *other algorithms used* in Table 1). If the item is first or last on the list, we assign to it a rating of the nearest neighbor. Otherwise, we compute the rating as an average of two or four, depending on the size of a list, nearest neighbors' ratings, i.e. the one/two above considered item and one/two below it. We assume that we do not have much data, so we cannot take more than four neighbors.

4 DATA SETS

We performed our experiments with three data sets, i.e. the *LDOS-CoMoDa*[1] data set (LDOS), the *Unibz-STS* data set (STS) and the *Restaurant & consumer* data set[2] (RC). Basic statistics of data sets are presented in Table 2.

The LDOS-CoMoDa (Kosir et al., 2011) contains user interaction with the system, i.e. the rating on a 5-star scale, basic users' information, content information about multiple item dimensions and twelve additional contextual information about the situation when the user consumed the item. According to (Odic et al., 2013) the choice of contextual variables to be used is crucial because of a different amount of information they gain. To eliminate irrelevant variables we computed correlation coefficients between context related attributes. We found only two of them to be strongly correlated, i.e. *city* and *country*. However, values of these attributes remain the same for a single user, so they were not taken into account for further consideration. In (Odic et al., 2013) six variables in the LDOS-CoMoDa were identify as informative. Since we focus on the cold start situation problem in this paper, we want to limit sparsity of the data as much as possible. Therefore, we chose four of six best contextual variables, i.e. *dominant emotion*, *end emotion*, *physical* and *decision*, to use in our further work presented in this paper.

The Unibz-STS (Braunhofer et al., 2013) data set was collected by a mobile application that recommends places of interests (POIs) in South Tyrol in Italy. The recommender is called South Tyrol Suggests (STS). The data set contains ratings on a 5-star scale, information about a users' personality (e.g. *extraversion*, *emotional stability*), a context of visiting a POI (e.g. *weather*, *season*, *companion*) and a POI's category. Like for the LDOS-CoMoDa data set, we did not consider fixed user information as contextual variables. For further consideration we chose the two most informative (according to (Odic et al., 2013)) context parameters, i.e. *weather* and *companion*.

The Restaurant & consumer data (Vargas-Govea et al., 2011) consists of three types of information: a restaurant data (e.g. *cuisine*, *smoking*, *dress*), a user information (e.g. *smoker*, *dress preference*, *transport*) and a rating that a user gave to a restaurant. In this data set ratings are expressed on a 0-2 scale. Contextual parameters such as information about a user's mood or companion are not available. Thus, for further work we chose most informative features: *smoker*, *drink level*, *dress preference*, *ambience*, *transport*, *personality* and *color*.

5 EVALUATION

To simulate a new recommender system we split the data set twice. Firstly, we generate 5 subsets by putting all ratings of a single user into one subset. Users were picked according to the number of ratings to achieve sets with comparable size. Secondly, we split every of 5 subsets into training and test sets. Because there is no time stamp of the ratings, the assignment of a single rating was made randomly using just one condition: if there are already 8 ratings of con-

[1] The data is available at http://212.235.187.145/spletnastran/raziskave/um/comoda/comoda.php.

[2] The data sets are available at https://github.com/irecsys/CARSKit/tree/master/context-aware_data_sets.

sidered user in the training set, put all other ratings of that user into test set. Consequently, every test set consists of ratings of users that have rated more than 8 items (number 8 was chosen to have reasonable number of test cases). Every test instance contains a user context part, an item content part and a rating that the user gave to the item.

$$ti = (r, (\gamma_1, c_1), ...(\gamma_i, c_i), (\alpha_1, a_1), ..., (\alpha_j, a_j), i),$$

where r is a rating that user gave to an item i, (γ_i, c_i) is part of contextual information (context(ti) which means that contextual parameter γ_i has value c_i, and (α_j, a_j) is part of item content information (content(ti)) which means that content feature α_j has value a_j.

We evaluated our approach by doing hold-out validations on five different training and test subsets described above. We tested our algorithm in three configuration: using both ICCPs and GCCPs, using only ICCPs and using only GCCPs. As measures we used the mean average error (MAE) and the root mean square error (RMSE) which are commonly used for evaluating the accuracy of a rating in the rating prediction task. We compared our approach with nine baseline algorithms, i.e. Random Guess, Item Average, User Average, Item K-Nearest Neighbors (Item KNN), User K-Nearest Neighbors (User KNN), SVD++, Biased Matrix Factorization (Biased MF), Bayesian Probabilistic Matrix Factorization (Bayesian Probabilistic MF) and Probabilistic Matrix Factorization (Probabilistic MF). For this purpose we used the `LibRec library`[3]. The last three algorithms are extensions of Matrix Factorization method, which are available for the rating prediction task in this library. The values of MAE and RMSE measures could be seen in Figure 1.

As seen in Figure 1, our method outperforms known baseline algorithms in most of the cases when considering a median value. The best median results were obtained when we used only ICCPs for `LDOS-CoMoDa` and `Unibz-STS` data sets. An exception is `Restaurant & consumer` data set for which we did not obtain any result in many cases. The reason of this behavior is that we do not have truly contextual data for training the model. Parameters that were used as context variables are fixed for a certain user, thus could not be considered as a user context. Of course they could be used as context information for GCCP, which is confirmed by obtained results. It could lead us to the conclusion that the method is more general and could be used also for quasi-contextual data.

[3]http://www.librec.net/

We performed the Wilcoxon test to prove a statistical significance of presented results. The p-values vary for different data sets, kinds of preferences and pairs of algorithms, and confirm observations from Figure 1 that our method is at least as good as baseline algorithms. The best results were obtained on the LDOS-CoMoDa data set. The Wilcoxon test confirmed a statistical significance of a prediction accuracy improvement on all algorithms for CCP and GCCP with the p-value smaller than 0.05 (and even smaller than 0.01 for some cases). It should be noticed that two of three Matrix Factorization algorithms perform pretty weak in comparison with other baseline methods, e.g. Probabilistic MF returned worse results than Random Guess. It confirms the fact that Matrix Factorization does not work well in the cold-start scenarios.

6 RELATED WORK

A Context-Awareness in Recommender Systems is a well-established research area. Many recommendation techniques were already proposed, so they were classified in three types of them according to the phase in the recommendation process in which the context is incorporated, i.e. pre-filtering technique, post-filtering technique and contextual modeling (Adomavicius and Tuzhilin, 2011).

A multi-agent system for making context and intention-aware recommendations of Points of Interest (POI) was presented in (Costa et al., 2012). The tasks of collecting information about POIs and storing users' profiles data were divided into two kinds of agents. The user's Personal Assistant Agent is responsible for receiving queries, storing user data, computing recommendations and updating user preferences according to his feedback. Authors incorporated not only the context but also a user's goal in visit the POI.

An interesting approach for a context-awareness was proposed in (Baltrunas and Amatriain, 2009). Authors introduced *micro profiles* which split a user profile into partitions depending on the values of context parameters. They showed that usage of such *micro profiles* gives a significant improvement in the prediction accuracy in the movie domain while considering time as a context variable. Contextual Conditional Preferences could be seen as a kind of *micro profiling*, because each preference statement consists of user interests and a context in which it is true.

In (Lee and Lee, 2007) a new context-aware music recommender system was presented. As a main recommendation technique authors used case-base rea-

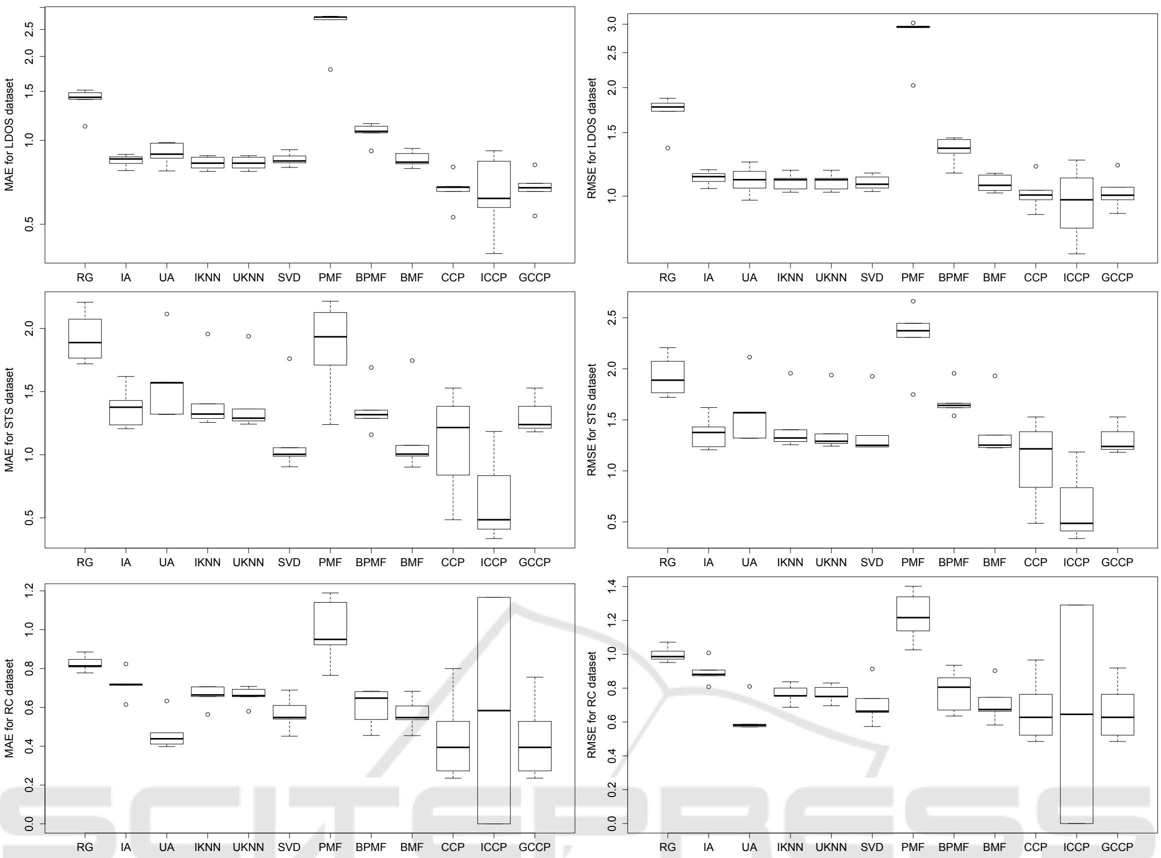

Figure 1: Boxplots of MAE and RMSE values for three data sets: `LDOS-CoMoDa`, `Unibz-STS` and `Restaurant & consumer data`. Algorithms that were used for computation: Random Guess (RG), Item Average (IA), User Average (UA), Item KNN (IKNN), User KNN (UKNN), SVD++ (SVD), Probabilistic MF (PMF), Bayesian Probabilistic MF (BPMF), Biased MF (BMF) and introduced algorithm with: both types of CCPs (CCP), only individual (ICCP) and only general ones (GCCP).

soning (CBR). CBR systems store knowledge in the case base in the form of `cases`. During a recommendation task, the cases are compared to the current case according to some similarity metric. In the paper, 2-step case-based reasoning was used. Firstly, to determine similar context, and then to find similar users to make predictions. Contextual conditional preferences could be seen as cases, but in fact they are something different. We chose active preferences according to two similarity metrics so we could position our work in the CBR research area. However, we do not have iterations or a relevance verification in the recommendation process.

A hybrid matrix factorization model for the cold start problem was presented in (Kula, 2015). It was shown to work well with cold and warm start scenarios. Similarly to our work, author used both, user and item information.

An interesting approach was introduced in (de Macedo et al., 2015). Authors presented a context-aware system for events recommendation that addresses the new item cold-start scenario. They identified many contextual signals and models, and used them as features for learning to rank events.

The idea of modeling user interests with a preference relation is not new. In (Boutilier et al., 2004) a formalism of CP-nets was proposed. CP-nets are intuitive graphical models for representing conditional preferences under the *ceteris paribus* („all else being equal") assumption. Preferences presented in this paper always contain „conditional part" which consists of contextual parameters only. Another difference is the lack of *ceteris paribus* assumption.

In (Satzger et al., 2006) user preferences, adopted from preference model from database systems, were used for improving collaborative filtering technique.

Contextual preferences were described in (Stefanidis et al., 2011) as database preferences annotated with contextual information, where contextual parameters take values from hierarchical domains, allowing different levels of abstraction. While using CCPs, a generalization of contextual variables is not possible.

7 CONCLUSIONS AND FUTURE WORK

In this paper we introduced an algorithm based on contextual conditional preferences for rating prediction tasks in cold-start situations. For our experiments we used three configurations of the model: only individual preferences, only general preferences and both types of preferences. We performed tests on three publicly available data sets and compared obtained results with those generated with several state-of-the-art baselines. We showed that proposed approach works at least as good as these baselines according to the prediction accuracy measured with MAE and RMSE for all data sets and configurations with one exception for individual contextual conditional preferences and `Restaurant & consumer` data set. However, this result is interesting. It showed that a user information like *drink level* or *dress preference*, is not enough to compute reasonable individual contextual conditional preferences. Nevertheless, proposed algorithm with general contextual conditional preferences decreased a prediction error in comparison with baselines. It could lead us to conclusion that the method is more general and could be used also for a quasi-contextual data.

The next steps that need to be taken are: (I) an automation of a feature selection by using deep learning techniques, (II) testing our approach in the continuous cold start situations, (III) an adaptation of the proposed method for a ranking task, and (IV) a comparison with other cold-start methods. Nevertheless, preliminary results look promising.

REFERENCES

Adomavicius, G. and Tuzhilin, A. (2011). Context-aware recommender systems. In Ricci, F., Rokach, L., Shapira, B., and Kantor, P. B., editors, *Handbook on Recommender Systems*, pages 217–256. Springer.

Baltrunas, L. and Amatriain, X. (2009). Towards time-dependant recommendation based on implicit feedback. In *Proceedings of 1st Workshop on Context-Aware Recommender Systems*.

Bernardi, L., Kamps, J., Kiseleva, J., and Mller, M. J. I. (2015). The continuous cold-start problem in e-commerce recommender systems. In Bogers, T. and Koolen, M., editors, *CBRecSys@RecSys*, volume 1448 of *CEUR Workshop Proceedings*, pages 30–33. CEUR-WS.org.

Boutilier, C., Brafman, R. I., Domshlak, C., Hoos, H. H., and Poole, D. (2004). Cp-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. *Journal of Artificial Intelligence Research*, 21:135–191.

Braunhofer, M., Elahi, M., Ricci, F., and Schievenin, T. (2013). Context-aware points of interest suggestion with dynamic weather data management. In Xiang, Z. and Tussyadiah, I., editors, *Information and Communication Technologies in Tourism 2014*, pages 87–100. Springer International Publishing.

Costa, H., Furtado, B., Pires, D., Macedo, L., and Cardoso, A. (2012). Context and intention-awareness in pois recommender systems. In *Proceedings of 4th Workshop on Context-Aware Recommender Systems*.

de Macedo, A. Q., Marinho, L. B., and Santos, R. L. T. (2015). Context-aware event recommendation in event-based social networks. In Werthner, H., Zanker, M., Golbeck, J., and Semeraro, G., editors, *RecSys*, pages 123–130. ACM.

Jannach, D., Zanker, M., Felfernig, A., and Friedrich, G. (2010). *Recommender Systems: An Introduction*. Cambridge University Press, New York, NY, USA, 1st edition.

Karpus, A., di Noia, T., Tomeo, P., and Goczyła, K. (2016). Using contextual conditional preferences for recommendation tasks: a case study in the movie domain. *Studia Informatica*, 37(1):7–18.

Kosir, A., Odic, A., Kunaver, M., Tkalcic, M., and Tasic, J. F. (2011). Database for contextual personalization. *Elektrotehniski vestnik [English print ed.]*, 78(5):270–274.

Kula, M. (2015). Metadata embeddings for user and item cold-start recommendations. In Bogers, T. and Koolen, M., editors, *CBRecSys@RecSys*, volume 1448 of *CEUR Workshop Proceedings*, pages 14–21. CEUR-WS.org.

Lee, J. S. and Lee, J. C. (2007). Context awareness by case-based reasoning in a music recommendation system. In *Proceedings of the 4th International Conference on Ubiquitous Computing Systems*, UCS'07, pages 45–58, Berlin, Heidelberg. Springer-Verlag.

Lombardi, S., Anand, S. S., and Gorgoglione, M. (2009). Context and customer behavior in recommendation. In *Proceedings of 1st Workshop on Context-Aware Recommender Systems*.

Odic, A., Tkalcic, M., Tasic, J. F., and Kosir, A. (2013). Predicting and detecting the relevant contextual information in a movie-recommender system. *Interacting with Computers*, 25(1):74–90.

Satzger, B., Endres, M., and Kießling, W. (2006). *A Preference-Based Recommender System*, pages 31–40. Springer Berlin Heidelberg, Berlin, Heidelberg.

Stefanidis, K., Pitoura, E., and Vassiliadis, P. (2011). Managing contextual preferences. *in Info. Sys*, pages 1158–1180.

Vargas-Govea, B., Gonzalez-Serna, G., and Ponce-Medellin, R. (2011). Effects of relevant contextual features in the performance of a restaurant recommender system. In *Proceedings of 3rd Workshop on Context-Aware Recommender Systems*.