Using Explainability for Constrained Matrix Factorization

Behnoush Abdollahi
Knowledge Discovery and Web Mining Lab, CECS
Department, University of Louisville
Louisville, KY 40292
b.abdollahi@louisville.edu

Olfa Nasraoui
Knowledge Discovery and Web Mining Lab, CECS
Department, University of Louisville
Louisville, KY 40292
olfa.nasraoui@louisville.edu

ABSTRACT

Accurate model-based Collaborative Filtering (CF) approaches, such as Matrix Factorization (MF), tend to be black-box machine learning models that lack interpretability and do not provide a straightforward explanation for their outputs. Yet explanations have been shown to improve the transparency of a recommender system by justifying recommendations, and this in turn can enhance the user's trust in the recommendations. Hence, one main challenge in designing a recommender system is mitigating the trade-off between an explainable technique with moderate prediction accuracy and a more accurate technique with no explainable recommendations. In this paper, we focus on factorization models and further assume the absence of any additional data source, such as item content or user attributes. We propose an explainability constrained MF technique that computes the top-n recommendation list from items that are explainable. Experimental results show that our method is effective in generating accurate and explainable recommendations.

1 INTRODUCTION

Machine learning (ML) models are being increasingly used in many sectors, ranging from health and education to e-commerce and criminal investigation. Hence, these algorithmic models are starting to affect the lives of more and more human beings. Examples include risk modeling and decision making in insurance, education (admission and success prediction), credit scoring, healthcare, criminal investigation and predicting recidivism, etc. These models are susceptible to bias that stems from the data itself (attribute or labels are biased) or from systemic social biases that generated the data (e.g. recidivism, arrests). As such, models that are learned from real world data can become unethical if their outputs discriminate, albeit unintentionally, against a certain group of people. While building ethical and fair models seems like the ultimate and ideal goal, the minimum and urgent criterion, that ML models should satisfy, is transparency, and this could be the first step in the direction toward fair and ethical models. Therefore, designing explainable intelligent systems, that facilitate conveying the reasoning behind the results, is of great importance.

When the machine learning model is used in a recommender system, it has been shown that explanations can help users make more accurate decisions; hence, improving user satisfaction and acceptance of recommendations [4, 9, 17]. CF recommender systems provide recommendations to users based on the rating-based similarity between users or items, giving rise to neighborhood-based CF approaches, which can be user-based or item-based. MF methods are accurate CF approaches that map users and items to low-dimensional feature vectors [12]. In most MF-based techniques, predictions are not interpretable and cannot be justified to the user as easily as in neighborhood-based CF methods. One way to communicate explanations would be based on identifying similar users and/or items in the latent space and presenting the most similar users and/or items as the explanation. The drawback of this method is that the way the explanation is generated does not necessarily comply with the learned ML model. This is because the solution to the MF optimization problem does not guarantee that the most similar users to an active user, who have liked the system's suggestion, are necessarily the active user's neighbors in the latent space. It would be very desirable and beneficial to design recommender systems that can give accurate suggestions, which, at the same time, facilitate conveying the reasoning behind the recommendations to the user. However, a main challenge in designing a recommender system is whether to choose an explainable technique with moderate prediction accuracy or a more accurate technique (such as MF) which does not give explainable recommendations.

Our research question is: can we design a Matrix Factorization model for a CF recommender engine that suggests items that are explainable, while recommendations remain accurate? Our current scope is limited to CF recommendations where explanations for recommended items can be generated from the ratings given to these items, by the active user's neighbors or the ratings given by the active user to other items that are similar to the recommended item (known, respectively, as user and item-based neighbor style explanation), as shown in Figure 1.

This article's contribution is a probabilistic formulation that extends our previous Explainable-Matrix Factorization (EMF) model [1], for providing explainable recommendations that can leverage the accurate predictions of MF and the transparency of neighborhood-based CF algorithms. In our method, explainability can be directly formulated based on the rating distribution within the user's or item's neighborhood. If many neighbors have rated the recommended item, or if the user has rated many items that are similar to the recommended item, then this provides a basis upon which to explain the recommendations. We encode the user-item explainability relationship in a graph. While other methods in the literature have used graph structures to find a better representation of data points in lower spaces, we further incorporate the explainability graph in the design of the MF model to be able to generate *explainable* recommendations.

RecSys '17, August 27–31, 2017, Como, Italy.

DOI: http://dx.doi.org/10.1145/3109859.3109913

2 RELATED WORK

There are different ways of classifying explanation styles. Generally neighbor style explanations can be User-Based Neighbor Style Explanation or Item-Based Neighbor Style, also known as Influence Style Explanation [4]. A user-based neighbor-style explanation is based on similar users, and generally is used when the CF method is also a user-based neighborhood style method. An item-based neighbor-style explanation is generally used in item-based CF methods by presenting the items that had the highest impact on the recommender system's decision [3]. In all styles, the data sources employed in the recommendation task, may be different from the data sources used in generating the explanation [4, 5, 9, 18], leading to explanation generation modules that are separate from the recommender system. However, performing the recommendation task based on the items' explainability (thus *integrating* recommendation and explanation) may improve transparency by suggesting interpretable items to the user, while enjoying the powerful prediction of a model-based CF approach. Zhang et al. [19] proposed a model-based CF to generate explainable recommendations based on item features and sentiment analysis of user reviews, as data sources, in addition to the ratings data. Their approach is similar to our method in that the recommendation and explanation modules are *not* separate. In contrast, our approach does *not* require additional reviews for explanation generation. Vig et al. [18] proposed a content-based explanation approach for a CF recommender system, where explanations are generated from community tags, a form of content-based explanation. Herlocker et al. [9] proposed 21 different explanation interfaces for user-based nearest neighbor CF. Gedikli et al. [6] used the same user-based approach as their base recommender system module and compared 10 different existing explanation types and their impact on the perceived level of transparency and hence the satisfaction of the user. Billsus and Pazzani [5] presented a keyword style and influence style explanation approach for their news recommendation system. Recently, Abdollahi and Nasraoui [2] presented an explainable CF approach using Restricted Boltzmann Machines (RBM).

3 PROPOSED METHOD

3.1 Explainability

In neighbor style explanations, the explanation is either *user-based*, i.e. based on the ratings given by similar users to the recommended item, or item-based, i.e. based on ratings given by the active user to similar items, as shown in Figure 1.

For user-based neighbor style explanation, if we divide the counts for each rating value by the total counts, we will obtain the empirical density distribution of the similar users' ratings on the recommended item i. Equivalently, this is the empirical conditional probability of ratings of item i, given the set of similar users for user u, denoted as N_u. For each rating value k in the set of ratings, κ, we can write this probability as:

$$\mathbf{Pr}(r_{v,i} = k | v \in N_u) = \frac{|N_u \cap U_{i,k}|}{|N_u|} \tag{1}$$

where $r_{v,i}$ is the rating of user v to item i and $U_{i,k}$ is the set of users who have given rating k to item i. Using Eq. 1, for each explanation we can calculate the expected value of the ratings given by the similar users to the recommended item i as follows:

$$\mathbf{E}(r_{v,i} | N_u) = \sum_{k \in \kappa} k \times \mathbf{Pr}(r_{v,i} = k | v \in N_u). \tag{2}$$

Your ratings for similar movies

Movie	Your Rating out of 5
L.A. Confidential	4
Air Force One	5
The Game	5
12 Angry Men	3
Carrie	4

Your neighbors' ratings for this movie

Rating	Number of Neighbors
★	0
★★	0
★★★	3
★★★★	4
★★★★★	2

Figure 1: Examples of user-based neighbor style explanation (left), and item-based neighbor style explanation (right).

Similarly, for the item-based neighbor style explanation, the ratings of user u on items that are similar to the recommended item can be used to obtain the empirical conditional probability of user u's ratings on item i. Given the set of similar items to item i, denoted as N_i, we can write this probability as:

$$\mathbf{Pr}(r_{u,j} = k | j \in N_i) = \frac{|N_i \cap I_{u,k}|}{|N_i|} \tag{3}$$

where $I_{u,k}$ is the set of items that were given rating k by user u. Using Eq. 3, the expected value for item-based explanation can be calculated as follows:

$$\mathbf{E}(r_{u,j} | N_i) = \sum_{k \in \kappa} k \times p(r_{u,j} = k | j \in N_i). \tag{4}$$

The expected rating of similar users or similar items gives a reasonable and intuitive measure of goodness or strength of a neighbor style explanation. We furthermore incorporate this value as a soft constraint in a modified cost function whose optimization will favor discovering latent factors that result in recommending items that have higher value for this expected value.

3.2 Explainability Graph

Given a set of users U, a set of items I, and a set of ratings r_{ui} given by user u to item i, we capture the explainability of an item relative to a user in a bipartite graph $G = (V, E)$, with the set of vertices $V = U \cup I$, and the set of edges E from the user nodes $u \in U$ to the item nodes $i \in I$, $E = \{e_{ui} | u \in U, i \in I\}$. The edge weights in the explainability graph are stored in matrix W which represents the explainabilty of the items to the users. Ideally, the edge weights should be higher for items *that can be easily explained* and low in the opposite case. We will try to capture this mutual explainability between an item and a user in an explainability score, $Expl_{u,i}$, which can be either $\mathbf{E}(r_{v,i}|N_u)$ or $\mathbf{E}(r_{u,j}|N_i)$, depending on the particular rationale that is chosen for the explanations. We thus define the *Explainability Matrix*, W, between user-item pairs in the Explainability graph, as follows:

$$W_{u,i} = \begin{cases} Expl_{u,i} & if\ Expl_{u,i} \geq \theta \\ 0 & otherwise \end{cases} \tag{5}$$

where θ denotes a threshold above which we accept item i to be explainable for user u. $W_{u,i}$ thus measures the explainability of item i for user u.

3.3 Explainable-MF

MF is a family of latent factor models that have been used with success in CF recommender system [12]. Using MF, a data matrix,

R, is factored into two lower-rank approximated matrices P and Q, in a joint latent space of dimensionality, f, that is much lower than the typically large number of users or items: $R_{n\times m} \simeq P_{n\times f} Q^T_{f\times m}$. MF algorithms learn the factors $p_u \in \mathbb{R}^f$ and $q_i \in \mathbb{R}^f$, which are the lower-rank representations of user u and item i in dimensionality f [12]. Given the MF definition and the explainability matrix W, we propose an explainable MF, which jointly learns user u and item i's latent vectors, given by p_u and q_i respectively, using as input, the explicit or implicit user interests (ratings or clicks) on the items, as in standard MF models, and using as an additional input, the explainability scores of the items as an additional soft constraint on the reconstruction loss from factorization as input:

$$J = \sum_{u,i\in R} (r_{u,i} - p_u q_i^T)^2 + \frac{\beta}{2}(\| p_u \|^2 + \| q_i \|^2) + \frac{\lambda}{2} \| p_u - q_i \|^2 W_{u,i} \tag{6}$$

where R is the set of user-item pairs for which the ratings are available, $\frac{1}{2}(||p_u||^2 + ||q_i||^2)$ is an $L2$ regularization term weighted by the coefficient β, and λ is an explainability regularization coefficient that controls the smoothness of the new representation and tradeoff between explainability and accuracy. The idea here is that if item i is explainable for user u, meaning $W_{u,i} > \theta$, then their representations in the latent domain should be close to each other ($p_u - q_i$ is close to zero), in order for the objective function to be minimized. Note that the first error term prevents p_u and q_i from vanishing to zero at the same time.

To minimize the objective function, we use stochastic gradient descent, which has been used successfully to solve MF for CF with big data sets [11, 13, 16]. For a given training instance r_{ij}, the updates for $p_u^{(t+1)}$ and $q_i^{(t+1)}$ can be shown to be:

$$\begin{aligned} p_u^{(t+1)} &\leftarrow p_u^{(t)} + \alpha(2(r_{u,i} - p_u q_i^T)q_i - \beta p_u - \lambda(p_u - q_i)W_{u,i}) \\ q_i^{(t+1)} &\leftarrow q_i^{(t)} + \alpha(2(r_{u,i} - p_u q_i^T)p_u - \beta q_i + \lambda(p_u - q_i)W_{u,i}) \end{aligned} \tag{7}$$

where α is the step size. With a proper choice of step size, gradient descent converges to a local minimum.

3.4 Explainability Effect in the Latent Space

The Explainability term used in the objective function J, encourages items, that have higher explainability relative to a user, to be projected close to that user in the latent space, while keeping the rating prediction error small. This can be illustrated for the simple case of two factors with an example. Figure 2 shows a test user, along with all the items, projected in a 2D latent space, learned using both EMF and the standard MF that excludes the explainability constraint. A sample user is shown as one point in black. The items shown in green, are items in the latent space, that have a cosine similarity with the user exceeding 0.7. Red items are explainable items, i.e. they have normalized explainability value larger than 0.7, as calculated using Eq. 2, when $|N_u| = 20$. Both EMF and MF techniques have predicted a sufficient set of relevant items for recommendation. However, the main difference is in the factors learned for the explainable items, shown in red. Using EMF, these explainable (red) items are projected such that they are close and relevant to the user, without decreasing recommendation accuracy. However, using standard MF, the red points are spread throughout the entire latent space, making them less favorable to be recommended.

4 EXPERIMENTAL EVALUATION

We tested our approach on the benchmark MovieLens [7] data which consists of 100, 000 ratings, on a scale of 1 to 5. The data is first split into training and test sets such that 10% of the latest ratings from each user are selected for the test set and the remaining 90% of the ratings are used in the training set. Results are reported over the test ratings.

We compare our results with four CF baseline methods that are most related to our approach: A standard latent factor model— Matrix Factorization (MF) [12], Probabilistic Matrix Factorization (PMF) [14], User-Based (UB) top-n CF [8], and Item-Based (IB) top-n CF [15]. We perform the comparison between our approach and the baselines in terms of top-n recommendation list. Therefore, for the MF-based techniques, the top-n recommendation list for each user is generated by selecting the top-n items with highest dot product between the user and the items' representations in the latent space. Standard UB and IB techniques are the state of the art top-n CF techniques that do not require content data. We use cosine similarity to find the similar users/items when generating W. Ratings are normalized between 0 and 1 and parameters are tuned using cross-validation. Figure 3 shows how the objective function, J, and the Root Mean Square Error (RMSE) decrease in each iteration, eventually converging to the optimal solution.

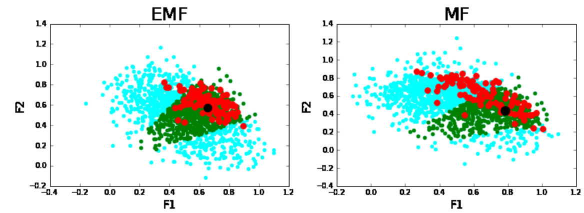

Figure 2: Explainable items (red) and potential recommendations (green) for a sample user (black), along with all the other items (cyan). All are represented in a 2D latent space ($f = 2$ for visualization purposes). Unlike standard MF, EMF succeeded to learn *explainable* latent item factors among the relevant items for recommendation.

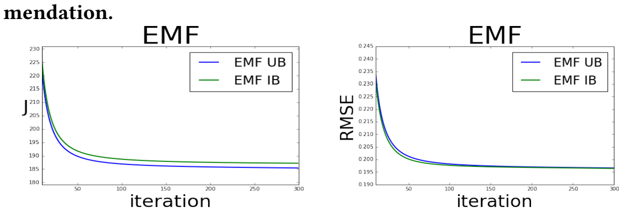

Figure 3: Objective function, J, calculated over the training data in each iteration decreases monotonically (left). RMSE calculated over the test data also decreases in each iteration (right). The setting for this experiment is as follows: $f = 5$, $|N_.| = 20$, $\theta = 0.1$, $\alpha = 0.001$, $\beta = 0.01$, and $\lambda = 0.005$.

4.1 Recommender System Evaluation

To evaluate the top-n recommendation results, we use two of the most common top-n metrics: Mean Average Precision (MAP) and Area Under Curve (AUC) at different top-n cutoffs. Table 1 shows the MAP and AUC results when varying the number of factors, f, and also the neighborhood size around users or items, hence collectively denoting this size as $|N_.|$. Other parameters are found using cross-validation: $\alpha = 0.001$, $\beta = 0.01$, $\theta = 0.1$, $\lambda = 0.005$. Both CF techniques, UB and IB, have lower MAP and AUC values compared to other techniques, when varying f. This is because memory based approaches tend to have lower accuracy due to considering only local samples and no latent factors for generating recommendation lists. When increasing f, EMF_{UB} outperforms other approaches. For $f = 5$, MF outperforms both EMF methods. This can be attributed to the impact of the explainability constraint on the learned hidden factors that result in a decrease in accuracy for small f. When varying $|N_.|$, EMF_{IB} outperforms other techniques in most cases, except for $|N_.| = 5$. This is because for small neighborhood size, explainability scores will be very small and the constraint will not have a sufficient impact on the accuracy.

4.2 Explainability Evaluation

To further assess the quality of the proposed approach, it is important to compare the results with other approaches in terms of the explainability

Table 1: Top row: comparison of accuracy when varying f. For EMF_{UB} and EMF_{IB} we set $|N_{.}| = 50$. Bottom row: comparison of accuracy when varying the number of neighbors. For EMF_{UB} , EMF_{IB}, PMF, and MF we set $f = 10$.

MAP@50							AUC						
f	UB	IB	PMF	MF	EMF_{UB}	EMF_{IB}	f	UB	IB	PMF	MF	EMF_{UB}	EMF_{IB}
5	0.009	0.0064	0.0113	**0.0149***	0.0108	0.011	5	0.4988	0.4982	0.5743	**0.7129***	0.5616	0.5745
10	0.009	0.0064	0.0108	0.0145	**0.0157***	0.0112	10	0.4988	0.4982	0.5629	0.7033	**0.7115***	0.5791
20	0.009	0.0064	0.0116	0.0143	**0.0146***	0.0118	20	0.4988	0.4982	0.563	0.6843	**0.6873***	0.5791
50	0.009	0.0064	0.0126	0.015	**0.0165***	0.0138	50	0.4988	0.4982	0.54	0.5697	**0.5984***	0.5019
$\|N_{.}\|$	UB	IB	PMF	MF	EMF_{UB}	EMF_{IB}	$\|N_{.}\|$	UB	IB	PMF	MF	EMF_{UB}	EMF_{IB}
5	0.009	0.0065	0.0108	**0.0145***	0.0102	0.0138	5	0.4759	0.4711	0.563	0.7011	0.7131	**0.7707***
10	0.0087	0.0064	0.0108	0.0145	0.0101	**0.0197***	10	0.4851	0.4835	0.563	0.7011	0.6787	**0.7821***
20	0.0085	0.0071	0.0108	0.0145	0.009	**0.0272***	20	0.489	0.4826	0.563	0.7011	0.6522	**0.7872***
50	0.0081	0.0077	0.0108	0.0145	0.0105	**0.0328***	50	0.4905	0.4991	0.563	0.7011	0.6855	**0.7463***

Table 2: Top row: comparison of accuracy when varying f. For EMF_{UB} and EMF_{IB} we set $|N_{.}| = 50$. Bottom row: comparison of explainability metrics when varying number of neighbors. For EMF_{UB} , EMF_{IB}, PMF, and MF we set $f = 10$.

MEP@50							MER@50						
f	UB	IB	PMF	MF	EMF_{UB}	EMF_{IB}	f	UB	IB	PMF	MF	EMF_{UB}	EMF_{IB}
5	0.449	0.551	0.6284	0.7079	0.7080	**0.7090***	5	0.054	0.07	0.0706	0.0756	**0.0757***	0.073
10	0.449	0.551	0.5412	0.7085	**0.7089***	0.7187	10	0.054	0.07	0.0622	0.0757	**0.0758***	0.0748
20	0.449	0.551	0.3617	0.7187	0.7224	**0.7242***	20	0.054	0.07	0.0399	0.0778	**0.0785***	0.0755
50	0.449	0.551	0.0843	0.5502	**0.5845***	0.4011	50	0.054	0.07	0.0085	0.0564	**0.0569***	0.0362
$\|N_{.}\|$	UB	IB	PMF	MF	EMF_{UB}	EMF_{IB}	$\|N_{.}\|$	UB	IB	PMF	MF	EMF_{UB}	EMF_{IB}
5	0.4831	0.5895	0.5412	0.708	**0.7081***	0.708	5	0.0583	0.0708	0.062	0.075	**0.0756***	0.0729
10	0.4489	0.5516	0.5412	0.708	0.7083	**0.7099***	10	0.0534	0.0701	0.062	0.075	**0.0757***	0.0732
20	0.4195	0.5423	0.5412	0.708	0.7082	**0.7087***	20	0.0496	0.0668	0.062	0.075	**0.0756***	0.073
50	0.4124	0.5416	0.5412	0.708	0.7083	**0.7096***	50	0.0485	0.0652	0.062	0.075	**0.0757***	0.0731

Table 3: varying θ

	EMF_{UB}				EMF_{IB}			
θ	MAP	AUC	MEP	MER	MAP	AUC	MEP	MER
0.2	0.0113	0.5976	0.3287	0.2448	0.0119	0.6027	0.3074	0.0941
0.4	0.0118	0.6156	0.1714	0.2018	0.0111	0.6257	0.1437	0.082
0.6	0.0115	0.6128	0.0713	0.2001	0.0097	0.5574	0.0433	0.0569
1	0.0117	0.6373	0	0	0.0104	0.5831	0	0
Avg.	0.0115	0.6158	0.1429	0.1614	0.0107	0.5922	0.1236	0.0582

of those items that are actually recommended. Note that in this work, we are not proposing a new explanation format that requires user evaluation. However, we can evaluate the top-n recommendations in terms of the explainability of the suggested list. We measure explainability using the MEP and MER metrics [1]. In top-n recommendation, *Explainability Precision (EP)* is defined as the proportion of explainable items in the top-n recommendation list relative to the number of recommended (top-n) items for each user. Similar to the recall metric, *Explainability Recall (ER)* is the proportion of explainable items in the top-n recommendation list relative to the number of *all* explainable items for a given user. In our experiments, EP and ER are calculated using $\theta = 0.01$. Mean EP (MEP) and Mean ER (MER) are reported in Table 2, which are the average values of EP and ER over all users. EMF_{UB} or EMF_{IB} outperform other approaches in terms of MEP, when varying f or $|N_{.}|$. EMF_{UB} outperforms other techniques in terms of MER, when varying f or $|N_{.}|$. EMF_{UB} and EMF_{IB} have low MEP and MER results for $f = 50$. This shows that for large f, explainability constraint has counter effect on MEP and MER results. When varying $|N_{.}|$, EMF_{UB} has higher MER, but lower MEP compared to EMF_{IB} for $|N_{.}| >= 10$.

To study the effect of the threshold, θ, on the explainability and top-n recommendation accuracy, we trained both EMF models when varying θ, while fixing all the other parameters (Table 3). For both techniques, MEP and MER reach the highest value when the threshold θ is small, and decrease by increasing θ. When $\theta = 1$, almost no user-based or item-based neighbor has high value. This is because users usually tend to rate a small set of items.

5 CONCLUSION

We presented a MF-based recommender system that can suggest items that are relevant and explainable, without any major sacrifice in recommendation accuracy. Our scope, in this work, is limited to CF recommendations where no additional source of data is used in recommendations or in explanations. Thus explainability can be directly formulated based on the rating distribution within the active user's neighborhood or the recommended item's neighborhood. Our rationale is that if many similar users (neighbors) have rated the recommended item, or if the user has rated many items that are similar to the recommended item, then this provides a basis upon which one can explain the recommendations.

We focused our research on CF recommender systems which have been shown to perform better than Content Based (CB) filtering methods [10]. We also focused on user-based or item-based neighbor style explanations and did not use any external data source, relying instead only the ratings. This contribution is important because using *no* external source is one of the main challenges for explainable *pure* CF recommender engines. This, in fact, is one of the main differences between our approach and other approaches in the literature. For this reason, these approaches are not comparable on a fair basis. In the future, we plan to extend our evaluation to include user-based testing and to expand our method to other explanation styles and other recommendation domains.

6 ACKNOWLEDGMENTS

This research was partially supported by KSEF Award KSEF-3113-RDE-017.

REFERENCES

[1] Abdollahi, B., and Nasraoui, O. Explainable matrix factorization for collaborative filtering. In *Proceedings of the 25th International Conference Companion on World Wide Web* (2016), International World Wide Web Conferences Steering Committee, pp. 5–6.

[2] Abdollahi, B., and Nasraoui, O. Explainable restricted boltzmann machines for collaborative filtering. *arXiv preprint arXiv:1606.07129* (2016).

[3] Bilgic, M. Explaining recommendations: Satisfaction vs. promotion. In *In Proceedings of Beyond Personalization 2005, the Workshop on the Next Stage of Recommender Systems Research(IUI2005* (2005), pp. 13–18.

[4] Bilgic, M., and Mooney, R. J. Explaining recommendations: Satisfaction vs. promotion. In *Beyond Personalization Workshop, IUI* (2005), vol. 5.

[5] Billsus, D., and Pazzani, M. J. A personal news agent that talks, learns and explains. In *Proceedings of the third annual conference on Autonomous Agents* (1999), ACM, pp. 268–275.

[6] Gedikli, F., Jannach, D., and Ge, M. How should i explain? a comparison of different explanation types for recommender systems. *International Journal of Human-Computer Studies 72*, 4 (2014), 367–382.

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

[8] Herlocker, J. L., Konstan, J. A., Borchers, A., and Riedl, J. An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval* (1999), ACM, pp. 230–237.

[9] Herlocker, J. L., Konstan, J. A., and Riedl, J. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work* (2000), ACM, pp. 241–250.

[10] Herlocker, J. L., Konstan, J. A., Terveen, L. G., and Riedl, J. T. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS) 22*, 1 (2004), 5–53.

[11] Koren, Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining* (2008), ACM, pp. 426–434.

[12] Koren, Y., Bell, R., and Volinsky, C. Matrix factorization techniques for recommender systems. *Computer*, 8 (2009), 30–37.

[13] Paterek, A. Improving regularized singular value decomposition for collaborative filtering. In *Proceedings of KDD cup and workshop* (2007), vol. 2007, pp. 5–8.

[14] Salakhutdinov, R., and Mnih, A. Probabilistic matrix factorization. Citeseer.

[15] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web* (2001), ACM, pp. 285–295.

[16] Takács, G., Pilászy, I., Németh, B., and Tikk, D. Major components of the gravity recommendation system. *ACM SIGKDD Explorations Newsletter 9*, 2 (2007), 80–83.

[17] Tintarev, N., and Masthoff, J. Designing and evaluating explanations for recommender systems. In *Recommender Systems Handbook*. Springer, 2011, pp. 479–510.

[18] Vig, J., Sen, S., and Riedl, J. Tagsplanations: explaining recommendations using tags. In *Proceedings of the 14th international conference on Intelligent user interfaces* (2009), ACM, pp. 47–56.

[19] Zhang, Y., Lai, G., Zhang, M., Zhang, Y., Liu, Y., and Ma, S. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval* (2014), ACM, pp. 83–92.