Merging Latent Factors and Tags to Increase Interactive Control of Recommendations

Tim Donkers
University of Duisburg-Essen
Duisburg, Germany
tim.donkers@uni-due.de

Benedikt Loepp
University of Duisburg-Essen
Duisburg, Germany
benedikt.loepp@uni-due.de

Jürgen Ziegler
University of Duisburg-Essen
Duisburg, Germany
juergen.ziegler@uni-due.de

ABSTRACT

We describe a novel approach that integrates user-generated tags with standard Matrix Factorization to allow users to interactively control recommendations. The tag information is incorporated during the learning phase and relates to the automatically derived latent factors. Thus, the system can change an item's score whenever the user adjusts a tag's weight. We implemented a prototype and performed a user study showing that this seems to be a promising way for users to interactively manipulate the set of items recommended based on their user profile or in cold-start situations.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*information filtering, search process*

Keywords

Recommender Systems; Interactive Recommending; Matrix Factorization; Tags; User Interfaces

1. INTRODUCTION

Optimizing the objective accuracy of algorithms that generate recommendations has led to considerable advances, but does not necessarily increase user satisfaction [3]. Hence, letting users influence the recommendation process is increasingly considered an important goal in *Recommender Systems* (RS) research. Interactive RS have been proposed that use metadata such as user-provided tags for this purpose [5]. This has the advantage of using concepts that are meaningful to users without requiring explicit item descriptions. Using tags to express user preferences thus seems promising to improve user control and comprehension. However, attempts to increase interactivity (e. g. [5]) are typically independent of conventional *Collaborative Filtering* (CF) techniques and consequently do not consider existing user profiles based on e. g. previous ratings. Moreover, the availability of precise and efficient algorithms such as *Matrix Factorization* (MF) [4] is not exploited. What is lacking, thus, are techniques that combine the accuracy-related benefits of model-based RS with the easy-to-understand semantics of tags. We therefore propose an interactive recommending approach that integrates latent factors derived by standard SVD-like MF with tags users provided for the items.

Copyright is held be the author(s).
RecSys 2015 Poster Proceedings, September 16-20, 2015, Vienna, Austria.

2. CONCEPT & PROTOTYPE

A range of techniques is available for integrating supplemental data into MF which has been shown to increase accuracy. These techniques, however, are typically very limited regarding user control. In addition, after being learned, the factors often exhibit no interpretable association with the supplemental information, which thus cannot be accessed by the user. In contrast, in [1], the data is explicitly used to establish a content-related association: Using a regression-constrained formulation, the factors are considered as functions of content attributes. We initially follow this approach closely for incorporating item-specific tag relevance information: For a set T of tags we define ${}^{\mathbf{u}}\mathbf{A} \in \mathbb{R}^{|U|\times|T|}$ and ${}^{\mathbf{i}}\mathbf{A} \in \mathbb{R}^{|I|\times|T|}$ representing their relationship with users U and items I, and redefine the original MF model:

$$\mathbf{R} \approx \mathbf{P}\mathbf{Q}^{\mathrm{T}} = {}^{\mathbf{u}}\mathbf{A}\,{}^{\mathbf{u}}\boldsymbol{\Theta}({}^{\mathbf{i}}\mathbf{A}\,{}^{\mathbf{i}}\boldsymbol{\Theta})^{\mathrm{T}}, \tag{1}$$

with ${}^{\mathbf{u}}\boldsymbol{\Theta}$ and ${}^{\mathbf{i}}\boldsymbol{\Theta}$ being the factor-tag matrices corresponding to users and items, respectively. However, explicit supplemental information may only be available either for users or for items. Generally, we act on the assumption that tag-item relevance scores have been calculated separately and ${}^{\mathbf{i}}\mathbf{A}$ is known a priori. Specifically, we exploit tag relevance scores [5]: $a_{it} \in [0, 1]$ describes the extent to which tag t is relevant for item i. In contrast, the corresponding matrix for users, ${}^{\mathbf{u}}\mathbf{A}$, is considered to be unknown. Thus, we treat the whole term ${}^{\mathbf{u}}\mathbf{A}\,{}^{\mathbf{u}}\boldsymbol{\Theta}$ implicitly at this step by just learning the user-factor matrix $\mathbf{P}$ as known from standard MF. With this constrained equation, we formulate the minimization problem as in [1] and apply gradient descent.

A user's u (calculated) interest in a particular factor f is numerically expressed by entry p_{uf} of $\mathbf{P}$ while entry q_{if} of $\mathbf{Q}$ describes the extent to which item i possesses this factor. Although in our case tag relevance scores are only known for items, we can establish a relation between users and tags as well: Under the assumption that f reflects a certain characteristic which has the same semantic meaning for users and items [4], we extend the approach of [1] and transfer the learned relationship between tags and latent factors to the user side. In fact, we assume that the regression coefficients stored in ${}^{\mathbf{i}}\boldsymbol{\Theta}$ are equivalent to the implicitly assumed entries of the corresponding matrix ${}^{\mathbf{u}}\boldsymbol{\Theta}$, such that: ${}^{\mathbf{u}}\boldsymbol{\Theta} = {}^{\mathbf{i}}\boldsymbol{\Theta} =: \boldsymbol{\Theta}$. Thus, according to (1) we solve for ${}^{\mathbf{u}}\mathbf{A}$:

$$\begin{aligned}\mathbf{P} = {}^{\mathbf{u}}\mathbf{A}\boldsymbol{\Theta} \Leftrightarrow \mathbf{P} = {}^{\mathbf{u}}\mathbf{A}\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{T}} \Leftrightarrow \\ {}^{\mathbf{u}}\mathbf{A} = \mathbf{P}\mathbf{V}\boldsymbol{\Sigma}^{+}\mathbf{U}^{\mathrm{T}} \Leftrightarrow {}^{\mathbf{u}}\mathbf{A} = \mathbf{P}\boldsymbol{\Theta}^{+}\end{aligned} \tag{2}$$

Since $\boldsymbol{\Theta}$ is generally not a square matrix, we first calculate its pseudo-inverse $\boldsymbol{\Theta}^{+}$ using SVD. Regarding the regression-constrained approximation of $\mathbf{R}$ in (1), this gives us:

$$\begin{aligned}\mathbf{R} &\approx {}^{\mathbf{u}}\mathbf{A}\boldsymbol{\Theta}\boldsymbol{\Theta}^{\mathrm{T}}\,{}^{\mathbf{i}}\mathbf{A}^{\mathrm{T}} \approx {}^{\mathbf{u}}\mathbf{A}\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{T}}\mathbf{V}\boldsymbol{\Sigma}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}}\,{}^{\mathbf{i}}\mathbf{A}^{\mathrm{T}} \\ &\approx {}^{\mathbf{u}}\mathbf{A}\mathbf{U}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}}\,{}^{\mathbf{i}}\mathbf{A}^{\mathrm{T}} \approx \mathbf{G}\boldsymbol{\Psi}\mathbf{H}^{\mathrm{T}}\end{aligned} \tag{3}$$

$\mathbf{G} \in \mathbb{R}^{|U| \times |T|}$ basically stores all vectors for the users and summarizes ${}^{\mathbf{u}}\mathbf{AU}$. Conversely, $\mathbf{H} \in \mathbb{R}^{|I| \times |T|}$ holds the item vectors. $\boldsymbol{\Psi} \in \mathbb{R}^{|T| \times |T|}$ is a diagonal matrix containing positive eigenvalues of $\boldsymbol{\Theta}\boldsymbol{\Theta}^{\mathrm{T}}$. The general interest of a certain user regarding all tags is now expressed by vector a_u of ${}^{\mathbf{u}}\mathbf{A}$, which is basically the counterpart of the tag-item relevance scores. Since they also comprise the latent factors, the g_u vectors can then be used to generate recommendations.

The previously abstract user-factor and item-factor vectors can now both be accessed in a much more comprehensible way. The tag concept is easily understood by users and can be used to actively adjust their own user vector, i. e. their profile. In particular, users can influence the recommendations by searching, selecting and weighting tags, thus indirectly determining their preferences in the latent factor space. A weight vector $w_u \in [0,1]^{|T|}$ therefore holds the user feedback regarding the tags, where 0 means no and 1 very strong interest in a particular tag. Integrating the weights into the calculation of recommendations leads to:

$$\tilde{r}_{ui} = (g_u + \alpha w_u)\boldsymbol{\Psi} h_i, \tag{4}$$

where $\alpha \in \mathbb{R}$ represents the extent to which the weight information should be considered. $\tilde{r}_{ui}$ is a combination of the user's general preference structure g_u, with the operationalization of the user's current mood or interest w_u. Initially, all values of w_u are set to 0. When users start to interact with the system by manipulating the values of w_u, for example by means of sliders, the resulting set of recommendations is continuously adapted in realtime.

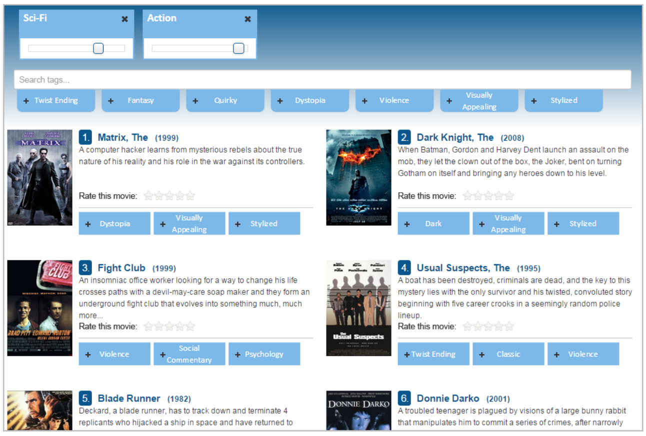

Figure 1: A user has selected and weighted the tags "Sci-Fi" and "Action", and therefore receives matching movie recommendations from our prototype system such as "Matrix" or "Fight Club".

Figure 1 shows a web-based prototype movie RS we have implemented to demonstrate this approach: At the top, an area is shown where the users can place the tags they select and adjust their weight by manipulating the sliders attached to them. Users can also search for tags with the input field underneath. Below, the system shows some suggested tags. Alongside each recommendation the three most relevant tags for this movie are shown. In addition, users could also rate the recommended movies to further adapt their profile.

3. EVALUATION & DISCUSSION

We performed an evaluation using a standard SVD-like MF algorithm[1] as a baseline, and extended this algorithm according to our approach considering a number of the most popular tags as additional training data. We used the well-known MovieLens 10M dataset for ratings and the MovieLens Tag Genome dataset for tag-item relevance scores. Figure 2 shows the results of one of our offline experiments.

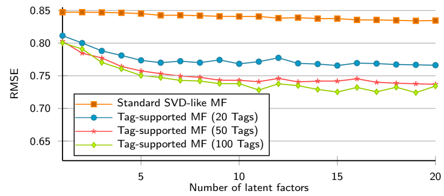

Figure 2: RMSE for different configurations depending on the number of latent factors.

In line with others (e. g. [1]), it seems beneficial to include metadata into MF. However, we also performed a user study with 46 participants (33 female; age: $M = 22.89$, $\sigma = 6.88$) who had to interact with our prototype RS in different conditions, with and without tags. We used a questionnaire comprising among others items from [2]. Results from our and other offline experiments could be confirmed, as subjective perception of recommendation quality was higher with ($M = 3.65$, $\sigma = 0.69$) than without ($M = 3.16$, $\sigma = 0.73$) tags ($t(45) = -3.98$, $p < .001$), also prior to interaction. Outlining some further results, participants were also very satisfied with the movie they finally selected from the recommendations ($M = 4.35$, $\sigma = 0.09$) and stated a good usability (78 on SUS). In general, users liked the interaction via tags while perceiving the interaction effort to be acceptable ($M = 3.64$, $\sigma = 0.74$). In the tag condition, initial preferences were elicited by only selecting a small number of tags instead of rating items first. Since this led to particularly promising results in terms of e. g. perceived recommendation quality, our tag-supported approach seems also to be useful in cold-start situations. In future work, we plan to evaluate the users' perception of differences between conventional MF and our prototype integrating tags in more detail, and to exploit the integration of additional data more extensively.

4. REFERENCES

[1] P. Forbes and M. Zhu. Content-boosted matrix factorization for recommender systems: Experiments with recipe recommendation. In *Proc. RecSys '11*, pages 261–264. ACM, 2011.

[2] B. P. Knijnenburg, M. C. Willemsen, and A. Kobsa. A pragmatic procedure to support the user-centric evaluation of recommender systems. In *Proc. RecSys '11*, pages 321–324. ACM, 2011.

[3] J. A. Konstan and J. Riedl. Recommender systems: From algorithms to user experience. *User Model. User-Adap.*, 22(1-2):101–123, 2012.

[4] Y. Koren, R. M. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42(8):30–37, 2009.

[5] J. Vig, S. Sen, and J. Riedl. Navigating the tag genome. In *Proc. IUI '11*, pages 93–102. ACM, 2011.

[1] Mahout *ParallelSGDFactorizer* (20 fact., 40 iter., $\lambda = .001$)