Rating items by rating tags

Fatih Gedikli
Department of Computer Science
44221 Dortmund
Germany
fatih.gedikli@tu-dortmund.de

Dietmar Jannach
Department of Computer Science
44221 Dortmund
Germany
dietmar.jannach@tu-dortmund.de

ABSTRACT

Different proposals have been made in recent years to exploit Social Web tagging data to improve recommender systems. The tagging data was used for example to identify similar users or viewed as additional information about the recommendable items.

In this work we propose to use tags as a means to express which features of an item users particularly like or dislike. Users would therefore not only add tags to an item but also attach a preference or rating to the tag itself, expressing, for example, whether or not they liked a certain actor in a given movie. Since rating data is in general sparse in commercial recommender applications we also present how to infer the user opinion regarding a certain feature (tag) for a given item automatically. In contrast to previous works, we not only infer the user's general preference for a tag but rather determine this preference in the context of a certain item.

An evaluation on the MovieLens data set reveals that our new tag-enhanced recommendation algorithm is slightly more accurate than a recent tag-based recommender even when the explicit tag rating data is 100% sparse, that is, if only derived information can be used.

1. INTRODUCTION

User-contributed tags are today a popular means for users to organize and retrieve items of interest in the participatory Web. Social Tagging plays an increasingly important role both on Social Web platforms such as Delicious[1] and Flickr[2] as well as on large-scale e-commerce sites such as Amazon.com.

Different ways of exploiting these additionally available pieces of information to build more effective recommender systems have been proposed in the last years. For example, tags can be seen as item descriptions and used by a content-based recommender. The set of tags a user attaches to resources also provides valuable information about the user. Thus, the relationship between users and tags can be used to find similar users in neighborhood-based collaborative filtering systems.

The goal of these tag-based approaches is to exploit the existing interactions between users, items and tags to improve the effectiveness of the recommender system, measured in terms of the predictive accuracy or the coverage of the algorithm [4, 5, 6, 18, 21, 23, 26].

In their recent work, Sen et al. [18] explore another way of leveraging tagging information to generate more precise recommendations. The strategy of their so-called "tagommenders" is to automatically infer the user's *preference for individual tags*. In the movie domain, the first task thus consists of determining if and to which extent the user Alice likes movies that are, for example, annotated with the tag "animated". After that, the rating prediction for an item is based on the aggregation of the inferred user preferences for the tags assigned to that item. An analysis of several algorithms and preference inference metrics on a tag-enhanced MovieLens data set showed that more precise recommendations can be made when the user's tag preferences are taken into account.

In the work by Sen et al., the inferred preferences or ratings for tags are "global" in a sense that a tag is either liked or disliked, independent of a specific item. Thus, a particular user Alice either likes movies annotated with the tag *animated* or not. In our work we explore whether *allowing users to give ratings for a tag in the context of an item* can help to further improve the accuracy of recommendations. The intuition behind this idea is that the same tag may have a positive connotation for the user in one context and a negative in another. For example, a user might like *action movies* featuring the actor *Bruce Willis*, but at the same time this user might dislike the performance of Bruce Willis in *romantic movies*.

Our general goal is to explore a new Social Web recommendation approach, in which *users rate items by rating the corresponding tags*. This corresponds to a multi-criteria or multi-dimensional recommendation approach as described in [1] or [2]. In order to analyze the potential value of such multi-criteria ratings and the corresponding richer user models we measure whether we can increase the accuracy of a tag-based recommender by using *inferred* tag ratings only. We aim to demonstrate that rating items by rating tags works on principle in this work.

The paper is organized as follows. In the next two sections, we will outline the overall preference inference and

[1] http://www.del.icio.us
[2] http://www.flickr.com

.

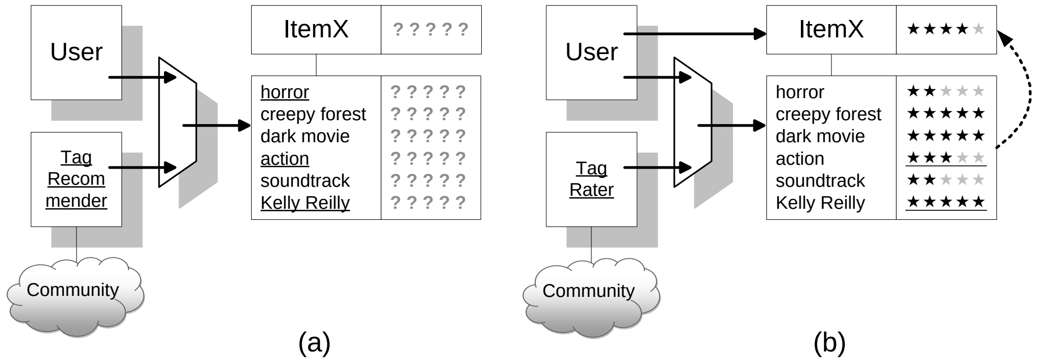

Figure 1: (a) Tagging items (automatically). (b) Rating items by rating tags (automatically) and recommending items based on tag ratings.

recommendation process and give details of our new user- and item-aware method for tag preference prediction and recommendation. In Section 4, the results of a comparative evaluation of the method on the tag-enhanced MovieLens data set are discussed. The paper ends with a discussion of related approaches and an outlook on future work.

2. ILLUSTRATIVE EXAMPLE

We illustrate the basic rationale of our method in the following example. Let us assume *User1* has attached tags to different movies[3] and given overall ratings on a scale from 1 to 5 as shown in Table 1. *User1* particularly likes action movies featuring Bruce Willis and romantic movies featuring Sandra Bullock, but appears to dislike romantic movies starring Bruce Willis.

Movie	Tags	Rating
M1	Bruce Willis, action, ...	5
M2	Bruce Willis, romance, ...	2
M3	Bruce Willis, action, ...	5
M4	Sandra Bullock, romance, drama, ...	5
M5	Bruce Willis, romance, drama, ...	**?**

Table 1: Tags and overall ratings of *User1*.

A method that automatically infers global preferences or ratings for tags such as the one described in [18] would probably derive a relatively high value for the tags "Bruce Willis" and "action". At the same time, the tag "romance" would receive a luke-warm rating somewhere between 3 and 4 because the user attached the tag both to a highly-liked and a disliked movie. As a result, the rating prediction for movie $M5$ based on the inferred tag ratings would may be around 4, that is, the system would tend to recommend $M5$.

Now let us assume that we knew more about the individual tags and their importance to *User1* as shown in Table 2 (assuming that we acquired this information directly from the user). In Table 2, we can see, that – perhaps among other reasons – the user did not like $M2$ because of Bruce Willis' appearance in a romantic movie. Since movie $M5$ is quite similar to $M2$ with respect to the attached tags, it is somewhat more intuitive *not* to recommend $M5$, which is exactly the opposite decision as in the example above.

Movie	Tags	Rating
M1	Bruce Willis (5), action (5), ...	5
M2	Bruce Willis (1), romance (2), ...	2
M3	Bruce Willis (5), action (5), ...	5
M4	Sandra Bullock (4), romance (5), ...	5
M5	Bruce Willis (**?**), romance (**?**), ...	**?**

Table 2: Tags and detailed ratings of *User1*.

We therefore propose a method that is capable of making recommendations based on more detailed rating data for tags. In addition, we develop a method to infer these detailed tag rating data automatically for cases in which such information is not available or the data are very sparse. In the example above, we would try to approximate the detailed ratings for the items $M1$ to $M5$ as good as possible given only the overall ratings for the movies.

Figure 1 summarizes and visualizes our approach to "rating items by rating tags" for a movie recommendation scenario in which both explicit and inferred tag ratings are exploited. At the core, the usual user-item matrix is extended not only by a set of user-provided tags for the items, but also by tag ratings describing the user's opinion about the item features represented by these tags. Rating items by rating tags consists of two phases. In the first phase, the user assigns one or more tags to the item to be rated. Figure 1 (a) shows the process of assigning tags to items. The user can either create new tags or select existing quality tags in the sense of [16] from the recommendation list of a tag recommender[4]. In the second phase, each individual tag can be given a rating (Figure 1 (b)), that is, the user rates selected tags and assigns an overall rating to the movie. These tag ratings can either be acquired explicitly using some extended recommender system user interface or derived automatically in case no such explicit information is available. In this work we only rely on automatically derived tag ratings as we do not possess real tag rating data yet.

In the next section, we present one possible neighborhood-based metric to derive tag ratings from the overall ratings

[3] These tags can also be derived from the community.

[4] See, for example, the tag recommendation systems of [25] or FolkRank [9].

automatically, see the "Tag Rater" component in Figure 1 (b). Then we propose a metric which is used to derive an overall rating prediction for a not-yet-seen item based on the ratings of its tags (see dotted arrow in Figure 1 (b)).

3. METHOD

In order to infer implicit tag ratings from the data and predict item ratings for a user, we used the following metrics and algorithms.

Similar to [18] and [22], we use a metric $w(m,t)$ that measures the *relevance* of a tag t for an item m. Note that in a setting, where users rate tags, the same tag can be applied many times for the same resource. In our approach, we use the following simple counting metric to determine a tag's relevance, which gives more weight to tags that have been used by users more often to characterize the item[5]:

$$w(m,t) = \frac{\textit{number of times tag t was applied to item m}}{\textit{overall number of tags applied to item m}} \qquad (1)$$

In [18], the prediction $\hat{r}_{u,t}$ of the general interest of a user u in the concept represented in a tag t is calculated as follows (method `movie-ratings`):

$$\hat{r}_{u,t} = \frac{\sum_{m \in I_t} w(m,t) * r_{u,m}}{\sum_{m \in I_t} w(m,t)} \qquad (2)$$

In this equation, I_t corresponds to the set of all items tagged with t. The explicit overall rating that u has given to movie m is denoted as $r_{u,m}$. The general idea of the method is thus to propagate the overall rating value to the tags of a movie according to their importance.

In our work, however, we are interested in predicting the rating for a tag in the context of the target user u *and* the target item i. Note that the rating prediction in Equation (2) does not depend on the target item i at all. Our tag prediction function, $\hat{r}_{u,i,t}$, for a given user u and an item i is calculated as follows:

$$\hat{r}_{u,i,t} = \frac{\sum_{m \in similarItems(i,I_t,k)} w(m,t) * r_{u,m}}{\sum_{m \in similarItems(i,I_t,k)} w(m,t)} \qquad (3)$$

Instead of iterating over all items that received a certain tag as done in [18], we only consider items that are similar to the item at hand, thereby avoiding the averaging effect of "global" calculations. In Equation (3), the calculation of neighboring items is contained in the function $similarItems(i, I_t, k)$, which returns the collection k of the most similar items from I_t. The similarity of items is measured with the adjusted cosine similarity metric. We also ran experiments using the Pearson correlation coefficient as a similarity metric, which however led to poorer results. As another algorithmic variant, we have tried to factor in the item similarity values as additional weights in Equation (3). Again, this did not lead to further performance improvements but rather worsened the results.

When using the user's explicit overall rating $r_{u,m}$ as in Equation (2) or (3), no prediction can be made for the tag rating if user u did not rate any item m tagged with t, i.e., if $I_t \cap ratedItems(u) = \emptyset$. We therefore apply the recursive prediction strategy as described in [24] and first calculate a prediction for $r_{u,m}$ if it is not given. An additional weight value $w_{rpa}(r_{u,m})$ is applied to the recursively predicted value $\hat{r}_{u,m}$ where $w_{rpa}(r_{u,m})$ is defined as follows:

$$w_{rpa}(r_{u,m}) = \begin{cases} 1, & r_{u,m} \text{ is given} \\ \lambda & r_{u,m} \text{ is not given} \end{cases} \qquad (4)$$

The combination weight threshold λ is a value between $[0,1]$. The RPA strategy can therefore be incorporated in the tagommender approach of [18] by extending Equation (2) leading to the following form:

$$\hat{r}_{u,t} = \frac{\sum_{m \in I_t} w(m,t) * w_{rpa}(r_{u,m}) * \mathcal{R}(r_{u,m})}{\sum_{m \in I_t} w(m,t) * w_{rpa}(r_{u,m})} \qquad (5)$$

The function $\mathcal{R}(r_{u,m})$ either returns $r_{u,m}$ if such a rating exists or an estimated value for $r_{u,m}$ based on the Recursive Prediction Algorithm [24]. Similar to Equation (5), we also applied the RPA strategy to our extended prediction function shown in Equation (3).

The overall rating prediction for an item m for a user u based on the (predicted) tag ratings given the user's average item rating ($\overline{r_u}$) and the user's average tag rating for a given item ($\overline{r_{u,m}}$) is given in Equation (6). Similar to the algorithm `cosine-tag` in [18], we calculate $\hat{r}_{u,m}$ as follows, where T_m is the set of all tags applied to m:

$$\hat{r}_{u,m} = \overline{r_u} + \frac{\sum_{t \in T_m} sim(m,t) * (\hat{r}_{u,m,t} - \overline{r_{u,m}})}{\sum_{t \in T_m} sim(m,t)} \qquad (6)$$

The individual tag ratings are weighted according to the adjusted cosine similarity between items and tags, see Equation (7). The similarity metric given in Equation (7) is used to measure the degree of consistency between the item's overall rating received by all users u who rated item m (U_m), and their predicted tag ratings for that item.

$$\text{sim(m,t)} = \frac{\sum_{u \in U_m} (r_{u,m} - \overline{r_u})(\hat{r}_{u,m,t} - \overline{r_{u,m}})}{\sqrt{\sum_{u \in U_m} (r_{u,m} - \overline{r_u})^2}\sqrt{\sum_{u \in U_m} (\hat{r}_{u,m,t} - \overline{r_{u,m}})^2}} \qquad (7)$$

4. EVALUATION

In order to measure the predictive accuracy of the presented methods we evaluated our approach on the popular MovieLens data set using a common experimental procedure and well known accuracy metrics. The results of this evaluation are described in this section. The goal of our subsequent evaluation is to analyze if the predictions made by the system are more accurate when we use more detailed tag information even for cases where all tag ratings are automatically derived, i.e., the sparsity level of the explicit tag rating data is 100%.

4.1 Data set and algorithms

4.1.1 Data set

We evaluated our approach to recommendation of items based on tag ratings on the "MovieLens 10M Ratings, 100k Tags" data set[6], which consists of three files: *ratings*, *movies*

[5] Further possible metrics to determine tag relevance are described in [18].

[6] `http://www.grouplens.org/node/73`

and *tags*. The ratings file contains a list of user ratings on a 5-star scale with half-star increments. The movies file contains information about each movie such as the title and the genre, which are, however, not used in our method. The tags file contains the information about which tags have been assigned by the users to the movies. A tag assignment is a triple consisting of one user, one resource (movie) and one tag. No rating information for the tags themselves is available in the original MovieLens database.

To the best of our knowledge, the 10M MovieLens data set is the only publicly available data set which contains both rating and tagging data. It contains 10,000,054 ratings and 95,580 (unrated) tags applied to 10,681 movies by 71,567 users of the online movie recommender service MovieLens.

4.1.2 Tag quality and data pruning

Limited tag quality is one of the major issues when developing and evaluating approaches that operate on the basis of user-contributed tags. In [16], for example, Sen et al. revealed that only 21% of the tags in the MovieLens system had adequate quality to be displayed to the user. Therefore, different approaches to deal with the problem of finding quality tags have been proposed in recent years, see, for example, [7], [16] or [17].

Note that our approach of rating items by rating tags calls for a new quality requirement to tags: tags must be appropriate for ratings. For example, there is no point in attaching a rating to a tag like "bad movie" because the tag already represents a like/dislike statement. It would therefore not be clear how to interpret a rating for such a tag. In our current work and evaluation, we did not take this question into account yet, that is, we did not distinguish between tags that are appropriate for being rated and those which are not. Still, we believe that this is one key question which was not considered before and which should be taken into account in future approaches to extracting rating information for tags automatically.

We applied traditional data pruning measures in order to improve the quality of the existing tag information. In particular, we defined the following two comparably weak requirements for movies and tags to be taken into account in our evaluation. First, we only consider movies that have at least two tags assigned. Second, only those tags are considered that were assigned by at least two users. Note that these constraints are not as strong as the constraints applied in previous works. In [22], for example, the authors require that "a tag has been applied by at least 5 different users and to at least 2 different items". Additionally, content analysis methods were applied in [22] to detect redundant tags, such as *violent* and *violence*, in order to replace them by one representative tag.

Beside the measures taken to improve the tag quality we further applied a random-based subsampling method to avoid problems with memory limitations. The resulting data set used in our experiments finally contained 4,713 movies, 3,979 users, 134,829 ratings and 77,127 tags.

4.1.3 Algorithms

The goal of our analysis is to determine whether recommending items based on user- and item-specific tag ratings can lead to more precise results than previous approaches that only use item-independent tag ratings, even when the tag ratings are inferred automatically.

We compared the following algorithms:

- **TBR-UI** (tag-based recommender with user- and item-specific ratings): it uses the metric in Equation (3) to predict tag ratings and the function in Equation (6) as well as the similarity metric from Equation (7) to predict item ratings.
- **TBR-UI-RPA**: same as above, but with the additional application of the Recursive Prediction Algorithm as described in Section 3.
- **TBR-U** (user-specific ratings only): corresponds to the `movie-ratings / cosine-tag` algorithm variant from [18].
- **TBR-U-RPA**: same as above, but with the additional application of the Recursive Prediction Algorithm.
- **Item-Item**: the classical item-to-item baseline recommendation scheme that does not exploit tag information at all. Adjusted cosine is used as a similarity function. Rating predictions are calculated as follows:

$$\hat{r}_{u,m} = \frac{\sum_{i \in ratedItems(u)} sim(m,i) * r_{u,i}}{\sum_{i \in ratedItems(u)} sim(m,i)} \qquad (8)$$

Note that we have used adjusted cosine as a similarity metric for all schemes. Experiments with other similarity metrics such as the Pearson correlation coefficient, however, led to poorer results.

In the user- and item-based schemes, TBR-UI(-RPA), also the parameter k which determines the size of the neighborhood containing the k most similar items from I_t can be varied, see Equation (3). In order to find an optimal value we performed 4-fold cross-validation and varied the parameter. A neighborhood-size of 3 was determined as an optimal choice. Besides this, the combination weight threshold parameter λ in Equation (4) is set as 0.5 as suggested as an optimal value in [24].

4.1.4 Accuracy metrics

In our experiments, we measured the predictive accuracy with the help of different metrics. First, we used the usual *Root Mean Squared Error* (RMSE) metric in order to make our results comparable with the results in literature. Note that we also calculated *Mean Absolute Error* values (MAE), but do not report these numbers here because no significant differences to the RMSE values have been observed. Because of the different criticisms on the RMSE measure, we measured the quality of the recommendations produced by the different algorithms also with the standard information retrieval metrics *precision* and *recall*.

To determine precision and recall, we followed the evaluation procedure proposed in [14] and converted the rating predictions into "like" and "dislike" statements as described in [15], that is, ratings above the user's mean rating are interpreted as "like" statements. In each of the iterations of a four-fold cross-validation procedure, the data set is split into a training set (75% of the data) and a test set (25% of the data). We then determined the set of existing "like" statements (ELS) in the 25% test set and retrieve a top-N recommendation list of length $|ELS|$ with each method based on the data in the training set. The top-N recommendation lists are created based on the prediction score of each method. The set of predicted like statements returned by a recommender shall be denoted as *Predicted Like Statements (PLS)*, where $|PLS| \leq |ELS|$.

Based on these definitions, *precision* can be defined as $\frac{|PLS \cap ELS|}{|PLS|}$ and measures the number of correct predictions in PLS. *Recall*[7] is measured as $\frac{|PLS \cap ELS|}{|ELS|}$ and describes how many of the existing "like" statements were found by the recommender.

In the evaluation procedure, recommendations and the corresponding precision and recall values were calculated for all users in the data set and then averaged. These averaged precision and recall values are then combined in the usual F-score, where

$$F = 2 * \frac{precision * recall}{precision + recall}$$

4.2 Results and discussion

Table 3 shows the average values for the F-score as well as the individual precision and recall values for the different algorithms in increasing order. We can observe that the previous tag-based `cosine-tag` method (TBR-U) slightly outperforms the traditional item-to-item recommendation scheme, a fact, which was already observed in [18]. The usage of the recursive prediction scheme helps to further improve recommendation accuracy.

Scheme	F-score	Precision	Recall	RMSE
Item-Item	80.26	81.09%	79.45%	1,1363
TBR-U	80.70	81.51%	79.90%	1,1521
TBR-U-RPA	80.77	81.59%	79.98%	1,1353
TBR-UI	81.11	81.92%	80.32%	1,1355
TBR-UI-RPA	81.75	82.56%	80.95%	1,1355

Table 3: Average F-score, precision, recall and RMSE values for different algorithms.

Our new (TBR-UI) method proposed in this paper in turn outperforms both the item-based method as well as the recent `cosine-tag` method from [18]. Again, RPA helps to further improve the results. The overall improvements are about 1.5 points on the F-score compared with the tag-unaware method and about 1 point when compared with the Sen et al.'s method.

As an interesting side effect, note that improvements can be consistently observed both in the precision and the recall values, that is, the new user- and item-specific scheme does not introduce a new tradeoff between these general goals.

Our algorithm also shows very light improvements on the RMSE metric compared with the item-based approach, see Table 3. While these improvements are less significant, the results indicate that our technique does also not lead to a deterioration on this metric.

Overall, our results demonstrate the potential value of a new recommendation approach that is based on the principle of rating items by rating tags because quality improvements could be achieved even in situations when the available tag-rating database is 100% sparse and all tag ratings are automatically inferred. Based on these observations, we expect further quality improvements in settings, in which also explicit tag ratings are available and where the quality of the tags is also higher than in our experiments.

5. RELATED WORK

In recent years, many researchers have recognized the value of Social Web tagging information for recommender systems and for example use tagging data as an additional source of information to improve the effectiveness of the recommender system, measured in terms of the predictive accuracy or the coverage of the algorithms [5, 11, 12, 18, 21, 23, 26].

The work that is most closely related to our approach is the recent work of Sen et al. [18]. The authors present tag-based recommender algorithms called "tagommenders" which compute a rating prediction for a target item and a target user by exploiting the user's *inferred tag preferences*, that is, the rating prediction for an item is based on the aggregation of the inferred user preferences of its tags. An analysis of several algorithms and preference inference metrics on the tag-enhanced MovieLens data set revealed that more precise recommendations can be made when the user's tag preferences are taken into account. The concept of *tag preference* was also used by Vig et al. [22] and describes "the user's sentiment toward a tag". In other words, a tag preference determines if and to which extent the user likes or dislikes items that have the feature represented by the tag. Consider again, for example, the tag "Bruce Willis" in the movie domain. In this example, tag preference would measure the degree a user likes or dislikes *all* movies in which Bruce Willis appeared. As stated above, in the approach in [18], the inferred tag preference is "global" in the sense that a user either likes or dislikes movies with that actor, independent of a particular movie.

To the best of our knowledge, the concept of tag preference was first introduced by Ji et al. [10]. The authors present a tag preference based recommendation algorithm for a collaborative tagging system, where collaborative tagging means the process of assigning tags to items by many users which is supported by Social Web platforms such as Delicious[8] or Connotea[9]. The authors first compute the target user's *candidate tag set* which consists of all tags for which a high tag preference value was predicted. Afterwards a naive Bayes classifier is used for making recommendations by exploiting the user's candidate tag set. The proposed algorithm was evaluated on a data set collected from the social bookmarking site Delicious. In contrast to the work of Sen et al. [18] the tag preference predictor in [10] does not make use of item ratings at all because the Delicious data set does not support ratings for items (bookmarks) like the tag-enhanced MovieLens data set.

In Vig et al. [22] the authors propose another concept called *tag relevance* which describes "the degree to which a tag describes an item". In the example above, tag relevance would measure how well the tag "Bruce Willis" describes a particular movie. Overall, in previous works *tag preference* was considered a user-specific concept whereas *tag relevance* is considered to be an item-specific concept. In contrast, in our work the proposed concept of a *tag rating* is user- and item-specific which has shown to be a helpful means to capture the user's preferences more precisely and thus

[7] In [14], this metric is called *coverage*.

[8] `http://www.del.icio.us`

[9] `http://www.connotea.org`

produce more accurate recommendations.

In [5], the authors exploit tagging data for an existing content-based recommender system in order to increase the overall predictive accuracy of the system. In their approach, the user interests are learned by applying machine learning techniques both on the textual descriptions of items (static data) and on the tagging data (dynamic data). Tags are therefore only considered as an additional source of information used for learning the profile of a particular user. By conducting a user study with 30 users the authors show that a slight improvement in the prediction accuracy of the tag-augmented recommender compared to the pure content-based one can be achieved. Our work rather represents a collaborative filtering with multi-criteria ratings and is thus better capable to exploit the "wisdom of the crowd" to improve the recommendation accuracy.

In recent times, tags were also used for enhancing the performance of traditional collaborative filtering recommender systems. Tag information was incorporated into existing collaborative filtering algorithms in one or the other way for enhancing the quality of recommendations for example in [11, 12, 21, 23] or [26]. Most commonly, tags are considered only as an additional source of information for their proposed methods. In [23], for example, tags are used for building user and item neighborhoods. The underlying idea of this approach is that neighbors that are determined in this way will be better predictors than those which are identified only based on explicit rating data. The evaluation on the tag-enhanced MovieLens data set shows that such an approach outperforms other algorithms based on non-negative matrix factorization and Singular Value Decomposition. In particular, the observed improvements in predictive accuracy were comparably strong for sparse data sets.

In contrast to works in which tags are only used to build better neighborhoods for classical collaborative filtering systems, we introduce a different way of exploiting tags for recommender systems in this work. We propose a new approach, in which *users rate items by rating tags*. This can be seen as being a sort of a multi-criteria or multi-dimensional recommendation approach as described in [1], [2] or [13]. In [1], Adomavicius and Kwon conjecture that multi-criteria ratings will play an important role for the next generation of recommender systems, in particular because multi-criteria ratings can help to handle situations in which users gave the same overall rating but had different reasons for that (which can be observed in the detailed ratings). Besides this, multi-criteria rating information can serve as a valuable source for explaining recommendations. Based on these observations, new user similarity metrics and algorithms were designed in [1] that exploit multi-criteria rating information leading to recommender systems of higher quality. The authors show on a small data set how exploiting multi-criteria ratings can be successfully leveraged to improve recommendation accuracy. Our approach of rating items by rating tags shares the advantages of these multi-criteria recommender systems such as improved accuracy and explanations; however the rating dimensions are not static in our approach and require metrics that are different to those put forward for example in [1]. In this respect, our work is also in line with the ideas of Shirky [19], who was among the first who argued that using predefined (rating) categories leads to different challenges such as the following. First, professional experts are needed who design the rating dimensions; additionally, new rating dimensions may emerge over time that were not covered by the predefined and pre-thought static rating dimensions designed or foreseen by a domain expert. In collaborative tagging systems, the set of rating dimensions is not limited which allows users to pick their particular way of stating their preferences. Of course, this comes at the price of a less homogeneous and more unstructured set of item annotations.

Finally, rating items is an important topic not only in the area of recommender research but also in the Semantic Web community. Revyu[10] [8], the winner of the Semantic Web Challenge of the year 2007, is a reviewing and rating Web site which aims to aggregate review data of items (resources) on the Web. Revyu allows people to rate items by writing reviews and gives users the opportunity to add meta-data to items in the form of Web2.0-style tags. Based on this relatively unstructured information, stronger semantics are later on derived. As stated by the authors, this functionality in itself is partially not particularly novel. The real benefit lies in the massive use of Semantic Web technologies and standards like RDF, SPARQL and the principles of Linked Data [3] in order to expose reviews in a reusable and machine-readable format.

Note that in the Revyu system, tags are merely used for classifying the reviewed items and for automatically extracting additional information. We believe that our work could complement this approach by exploiting the rating information which is implicitly contained in the tags. That way, by deriving individual preferences for the tags provided by a user, a better "understanding" of the free-text reviews could be achieved.

6. SUMMARY AND OUTLOOK

The main new idea of our work is to incorporate item-specific ratings for tags in the recommendation process. Following such an approach, users are able to evaluate an existing item in various dimensions and are thus not limited to the one single overall vote anymore. In contrast to previous attempts toward exploiting multi-dimensional ratings, our work aims to follow a Web 2.0 style approach, in which the rating dimensions are not static or predefined.

The goal of this paper was to propose a first, comparably simple recommendation method that can take item-specific tag ratings into account when generating rating predictions. In addition, we proposed one particular metric to automatically derive user- and item-specific tag ratings from the overall ratings based on item similarities in order to demonstrate that quality improvements can be achieved even when the tag rating data is not explicitly given. The results of the evaluation on the MovieLens data set shows that a measurable accuracy improvement can be achieved.

In our future work we will not only run experiments with other metrics (incorporating, e.g., *default tag ratings* or hybridization strategies) and more sophisticated methods for estimating tag ratings, but also explore further questions related to tag ratings in the Social Web recommendation process.

- **Further experiments and user interfaces.** First, experiments with real tag ratings are planned to measure the corresponding accuracy improvements. A particular question to be answered in that context is that

[10] http://revyu.com

of an appropriate user interface (see also [22]) because Web users are currently not acquainted to the interaction pattern "providing ratings for tags". Intuitively, interfaces that allow users to rate tags on a scale from 1 to 5 or allow users to classify tags in two or three categories such as "like", "dislike", or "indifferent" seem appropriate. However, we aim to explore different visualizations to stimulate more precise ratings.

- **Better explanations.** Tags can also be a helpful means to generate explanations for the end user. Explanations for recommendations are one of the current research topics in the recommender systems area because they can significantly influence the way a user perceives the system. In [20], for example, seven possible advantages of an explanation facility are described. In [22], the authors have evaluated explanation interfaces which use tag relevance and tag preference as two key components. Further studies could be conducted to examine the role of tag ratings in helping users understand their recommendations. If tags are both user- *and* item-specific, more personalized and detailed, multi-dimensional explanations can be provided. Based on appropriately designed explanation interfaces, the different aspects of explanations as discussed in [20] (such as transparency, trust, effectiveness and satisfaction) can be analyzed in different user studies. Again, also the question of the appropriate end-user visualization has to be answered.

- **Combination with tag recommenders.** Different techniques to *tag recommendation* have been developed in the last years to stimulate users to use a more consistent set of tags in the annotation process, see, for example, [25]. We expect that the value of item-specific tag ratings is even higher, when the overall set of tags used in the data set is more consistent.

- **New tag quality metrics.** We have stated in Section 4.1.2 that our approach of rating items by rating tags calls for a new quality requirement to tags: tags must be appropriate for ratings. Therefore, in our current work, we aim to develop new tag quality metrics in order to improve the overall performance of recommendation algorithms that are based on tag ratings.

Finally, by making the software used in our experiments publicly available[11], we hope to contribute to the comparability of different algorithms since our study revealed that relevant algorithmic details and parameters are often not reported in sufficient details.

7. REFERENCES

[1] G. Adomavicius and Y. Kwon. New recommendation techniques for multicriteria rating systems. *IEEE Intelligent Systems*, 22(3):48–55, 2007.

[2] G. Adomavicius and A. Tuzhilin. Extending recommender systems: A multidimensional approach. In *Proceedings of the Workshop on Intelligent Techniques for Web Personalization (ITWP'01)*, pages 4–6, Acapulco, Mexico, 2001.

[3] T. Berners-Lee. Linked data. `http://www.w3.org/DesignIssues/LinkedData.html`, 2006. Retrieved on July 11, 2010.

[4] T. Bogers and A. van den Bosch. Collaborative and content-based filtering for item recommendation on social bookmarking websites. In *Proceedings of the Workshop on Recommender Systems and the Social Web (RSWEB'09)*, pages 9–16, New York, NY, USA, 2009.

[5] M. de Gemmis, P. Lops, G. Semeraro, and P. Basile. Integrating tags in a semantic content-based recommender. In *Proceedings of the 2008 ACM Conference on Recommender Systems (RecSys'08)*, pages 163–170, Lausanne, Switzerland, 2008.

[6] J. Diederich and T. Iofciu. Finding communities of practice from user profiles based on folksonomies. In *Proceedings of the 1st International Workshop on Building Technology Enhanced Learning Solutions for Communities of Practice (TEL-CoPs'06)*, pages 288–297, Crete, Greece, 2006.

[7] J. Gemmell, M. Ramezani, T. Schimoler, L. Christiansen, and B. Mobasher. The impact of ambiguity and redundancy on tag recommendation in folksonomies. In *Proceedings of the 2009 ACM Conference on Recommender Systems (RecSys'09)*, pages 45–52, New York, NY, USA, 2009.

[8] T. Heath and E. Motta. Revyu.com: A reviewing and rating site for the web of data. In *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference (ISWC'07 + ASWC'07)*, pages 895–902, Busan, Korea, 2007.

[9] R. Jäschke, L. B. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme. Tag recommendations in folksonomies. In *Proceedings of the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'07)*, pages 506–514, Warsaw, Poland, 2007.

[10] A.-T. Ji, C. Yeon, H.-N. Kim, and G.-S. Jo. Collaborative tagging in recommender systems. In *Proceedings of the 20th Australian Joint Conference on Artificial Intelligence (AUS-AI'07)*, pages 377–386, Gold Coast, Australia, 2007.

[11] H.-N. Kim, A.-T. Ji, I. Ha, and G.-S. Jo. Collaborative filtering based on collaborative tagging for enhancing the quality of recommendation. *Electronic Commerce Research and Applications*, 9(1):73 – 83, 2010.

[12] H. Liang, Y. Xu, Y. Li, and R. Nayak. Collaborative filtering recommender systems based on popular tags. In *Proceedings of the 14th Australasian Document Computing Symposium (ADCS'09)*, University of New South Wales, Sydney, Australia, 2009.

[13] N. Manouselis and C. Costopoulou. Analysis and classification of multi-criteria recommender systems. *World Wide Web*, 10(4):415–441, 2007.

[14] M. Nakagawa and B. Mobasher. A hybrid web personalization model based on site connectivity. In *Proceedings of the Workshop on Web Mining and Web Usage Analysis (WebKDD'03)*, pages 59–70, Washington, DC, USA, 2003.

[15] J. J. Sandvig, B. Mobasher, and R. Burke. Robustness of collaborative recommendation based on association rule mining. In *Proceedings of the 2007 ACM*

[11] `http://ls13-www.cs.uni-dortmund.de/rec_suite.zip`

Conference on Recommender Systems (RecSys'07), pages 105–112, Minneapolis, MN, USA, 2007.

[16] S. Sen, F. M. Harper, A. LaPitz, and J. Riedl. The quest for quality tags. In *Proceedings of the 2007 International ACM Conference on Supporting Group Work (GROUP'07)*, pages 361–370, Sanibel Island, Florida, USA, 2007.

[17] S. Sen, J. Vig, and J. Riedl. Learning to recognize valuable tags. In *Proceedings of the 13th International Conference on Intelligent User Interfaces (IUI'09)*, pages 87–96, Sanibel Island, Florida, USA, 2009.

[18] S. Sen, J. Vig, and J. Riedl. Tagommenders: Connecting users to items through tags. In *Proceedings of the 18th International World Wide Web Conference (WWW'09)*, pages 671–680, Madrid, Spain, 2009.

[19] C. Shirky. Ontology is overrated. `http://www.shirky.com/writings/ontology_overrated.html`, 2005. Retrieved on July 11, 2010.

[20] N. Tintarev and J. Masthoff. Effective explanations of recommendations: User-centered design. In *Proceedings of the 2007 ACM Conference on Recommender Systems (RecSys'07)*, pages 153–156, New York, NY, USA, 2007.

[21] K. H. L. Tso-Sutter, L. B. Marinho, and L. Schmidt-Thieme. Tag-aware recommender systems by fusion of collaborative filtering algorithms. In *Proceedings of the 2008 ACM Symposium on Applied Computing (SAC'08)*, pages 1995–1999 , Fortaleza, Ceara, Brazil, 2008.

[22] J. Vig, S. Sen, and J. Riedl. Tagsplanations: Explaining recommendations using tags. In *Proceedings of the 13th International Conference on Intelligent User Interfaces (IUI'09)*, pages 47–56, Sanibel Island, Florida, USA, 2009.

[23] Z. Wang, Y. Wang, and H. Wu. Tags meet ratings: Improving collaborative filtering with tag-based neighborhood method. In *Proceedings of the Workshop on Social Recommender Systems (SRS'10)*, Hong Kong, China, 2010.

[24] J. Zhang and P. Pu. A recursive prediction algorithm for collaborative filtering recommender systems. In *Proceedings of the 2007 ACM Conference on Recommender Systems (RecSys'07)*, pages 57–64, Minneapolis, MN, USA, 2007.

[25] N. Zhang, Y. Zhang, and J. Tang. A tag recommendation system for folksonomy. In *Proceedings of the 2nd Workshop on Social Web Search and Mining (SWSM'09)*, pages 9–16, New York, NY, USA, 2009.

[26] Y. Zhen, W.-J. Li, and D.-Y. Yeung. Tagicofi: Tag informed collaborative filtering. In *Proceedings of the 2009 ACM Conference on Recommender Systems (RecSys'09)*, pages 69–76, New York, NY, USA, 2009.