Improving Recommendation Accuracy Based on Item-Specific Tag Preferences

FATIH GEDIKLI and DIETMAR JANNACH, TU Dortmund, Germany

In recent years, different proposals have been made to exploit Social Web tagging information to build more effective recommender systems. The tagging data, for example, were used to identify similar users or were viewed as additional information about the recommendable items. Recent research has indicated that "attaching feelings to tags" is experienced by users as a valuable means to express which features of an item they particularly like or dislike. When following such an approach, users would therefore not only add tags to an item as in usual Web 2.0 applications, but also attach a preference (*affect*) to the tag itself, expressing, for example, whether or not they liked a certain actor in a given movie. In this work, we show how this additional preference data can be exploited by a recommender system to make more accurate predictions.

In contrast to previous work, which also relied on so-called tag preferences to enhance the predictive accuracy of recommender systems, we argue that tag preferences should be considered in the context of an item. We therefore propose new schemes to infer and exploit context-specific tag preferences in the recommendation process. An evaluation on two different datasets reveals that our approach is capable of providing more accurate recommendations than previous tag-based recommender algorithms and recent tag-agnostic matrix factorization techniques.

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 therefore 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.

Beside the usage of tags for improved item retrieval, various ways of exploiting these additionally available pieces of information have been proposed in recent years to build more effective *recommender systems*. In general, the main task of such a recommender system is to provide its users with personalized buying proposals. Historically,

---

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

Authors' address: Department of Computer Science, TU Dortmund, 44221 Dortmund, Germany; email: firstname.lastname@tu-dortmund.de.

the main knowledge sources in the recommendation process are the users' past buying and rating behavior as well as information known about the available items, see Jannach et al. [2010] for a recent overview. Tagging information can be exploited in the recommendation process in different ways. Tags can for example be seen as item descriptions and used by a content-based recommender. However, the set of tags a user attaches to resources also provides valuable information about the user him- or herself. Thus, the relationship between users and tags can be used to find similar users in neighborhood-based collaborative filtering systems. Overall, the goal of many tag-based recommendation approaches is to exploit the existing interactions among users, items, and tags to improve the effectiveness of the recommender system, measured for example in terms of the predictive accuracy or the coverage of the algorithm [Diederich and Iofciu 2006; Hotho et al. 2006; de Gemmis et al. 2008; Tso-Sutter et al. 2008; Zhen et al. 2009; Bogers and van den Bosch 2009; Wang et al. 2010; Sen et al. 2009b].

In most existing approaches to tag-based or tag-enhanced collaborative item recommendation, the main assumption is that preference information provided by the user community is only available for the items. Only recently, first ideas have been put forward that consider the possibility to attach preferences also to the tags themselves and use this information to improve different quality aspects of the recommendation process.

In Sen et al. [2009b], for example, the goal is to leverage information about the users' estimated preference for individual tags to generate more precise recommendations. Assuming that no explicit tag preference information is available, the first step in their "tagommenders" is therefore to estimate the user's attitude toward the different tags. In the movie domain, this would for example mean to estimate if and to which extent 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 dataset showed that more precise recommendations can be made when the user's *estimated* tag preferences are taken into account.

Vig et al. [2010] later on report on a first study on using explicit tag preferences, which they called *tag expressions*. In their field study, the users of the MovieLens recommender were allowed to share tags and the associated *affect*[3]—like, dislike or neutral—to the tags attached to the movies. This way, users could express which features of a movie they particularly liked or disliked. Among other aspects, their study revealed that users particularly appreciated this new feature, a fact that was measured in increased user satisfaction. Above that, allowing users to express affect associated with tags also helped to increase the volume of the contributed tags as well as their quality.

While the work of Vig et al. for example shows how the users' satisfaction with the system can be increased, they do not propose any algorithms for improving the recommendation accuracy based on explicit tag preferences. Note that in contrast to Vig et al.'s "tag expressions," the tagommender algorithms proposed by Sen et al. rely on "global" tag preferences, which means that a tag is either liked or disliked by a user, independent of a specific item. Thus, a particular user Alice either likes movies annotated with the tag *animated* or not.

In our own previous work [Gedikli and Jannach 2010], which was developed independently of and in parallel with Vig et al.'s study, we already proposed first methods aimed at exploiting item-specific tag preferences to compute more precise recommendations. The intuition behind this idea was that the same tag may have a positive

---

[3]We use the notion of *preference* in this work.

Table I. Tags and Overall Ratings of *User1*

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,...	**?**

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. At the same time, being used to see this actor mainly in action movies, the user might dislike the performance of Bruce Willis in a romantic movie. First experiments on a tag-enhanced dataset and a neighborhood-based method for estimating item-specific tag preferences revealed that the predictive accuracy can be improved when compared with a similar method that only takes global tag preferences into account [Gedikli and Jannach 2010].

This article extends our previous work in different ways. First, we show how a new metric to derive user- and item-specific tag preferences can help us to produce more accurate recommendations when incorporated in a method based on Support Vector Machines (SVM), which showed superior performance in previous work. Beside comparing our algorithm with the best-performing method from Sen et al., we also compare our work with a recent tag-agnostic matrix factorization method and our own previously presented method. In addition to experiments with the tag-enhanced MovieLens dataset that was also used by Sen et al., we conducted experiments in which we varied the density of the tagging data. Above that, accuracy measurements were taken based on a new dataset consisting of *real* item-specific tag preferences, which were collected in a user study [Gedikli et al. 2011b] focusing on the explanatory power of tag preferences in the sense of [Vig et al. 2009]. This way, a more precise picture of the potential benefits of item-specific tag preferences could be obtained.

The article is organized as follows. In the next section, we outline the overall preference inference and recommendation process on an illustrative example. In Section 3, we present a scheme for automatically inferring user- and item-specific tag preferences. Afterwards, we show how the additionally available tag preference information can be exploited to make more accurate predictions. In Section 5, the results of the comparative evaluation of the different methods on two different datasets are discussed. The article ends with a discussion of related approaches and an outlook on future work.

2. ILLUSTRATIVE EXAMPLE AND OVERVIEW OF THE APPROACH

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

A method that automatically infers global preferences or ratings for tags such as the one described in Sen et al. [2009b] would probably derive a relatively high value for the tags "Bruce Willis" and "action." At the same time, the tag "romance" would receive a lukewarm 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 preferences 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 II (assuming that we acquired this information directly from the user). In Table II, we can see, that—perhaps among other reasons—the user did

Table II. Tags and Detailed Ratings of *User1*

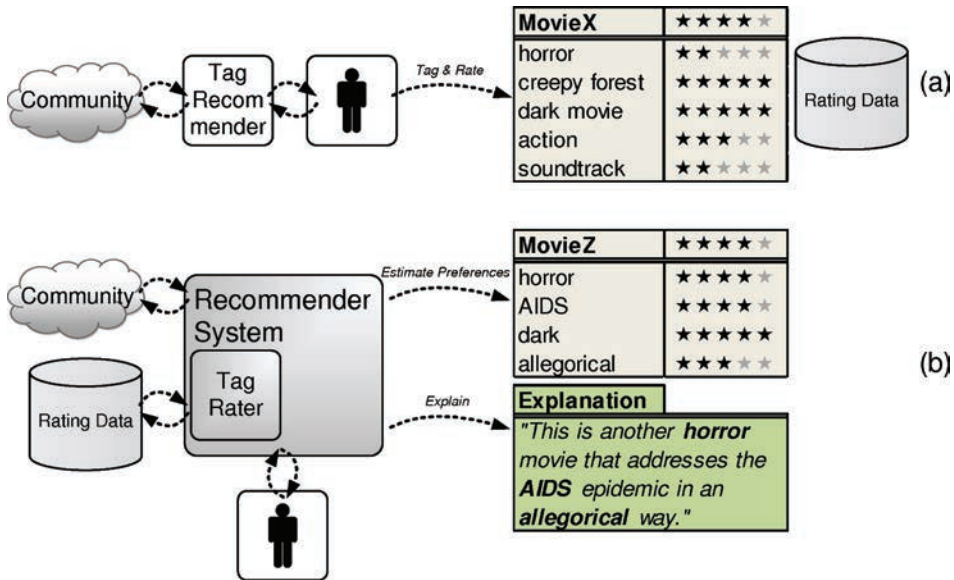| Movie	Tags	Rating
M1	Bruce Willis (5), action (5),...	5
M2	Bruce Willis (1), romance (2),...	2
M3	Bruce Willis (5), action (5),...	5
M4	Sandra Bullock (4), romance (5),...	5
M5	Bruce Willis **(?)**, romance **(?)**,...	**?**

Fig. 1.   (a) Tagging and rating items. (b) Recommending items and explaining recommendations by exploiting tag preference data.

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.

We therefore propose a method that is capable of making recommendations based on more detailed rating data for tags. We assume that such data will be available also in future systems, given the findings of the study of Vig et al. [2010], who showed that users enjoy sharing their affect by expressing their feelings with respect to certain features of the recommendable items. In addition, we develop a method to infer this detailed tag preference data automatically for cases in which such information is not available or the data is 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 preferences 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 preferences describing the user's opinion about the item features represented by these tags. Figure 1(a) shows the process of rating items by rating tags. The user assigns or chooses one or more tags for the item to be rated. The user can either create new tags or select existing quality tags in the sense of Sen et al. [2007] from a list of recommended tags, which can be provided by a tag recommender system such as the ones proposed in Zhang et al. [2009] or

Fig. 2.   Rating tags of a given movie on a 5-star scale with half-star increments.

Jäschke et al. [2007]. After assigning a tag, each individual tag can be given a rating, that is, the user rates selected tags and assigns an overall rating to the movie. Figure 2 shows a tag rating user interface that was used in our own study on the power of tag-based recommendation explanations in Gedikli et al. [2011b].

Figure 1(b), on the other hand, shows scenarios where the augmented rating matrix is exploited by a recommender system. The recommender uses, among others, tag preference data to estimate the prediction for the active user u on the item i. Tag preferences can either be acquired explicitly or derived automatically using the "Tag Rater" component in Figure 1(b) in case no such explicit information is available. Figure 1(b) also provides another compelling application scenario in which tag preferences are used for explaining recommendations, which, however, is not the focus of this work, but tackled by our current and future work (see Gedikli et al. [2011a] or Gedikli et al. [2011b]).

In the next section, we will present different methods to derive tag preferences from the overall ratings automatically, see the "Tag Rater" component in Figure 1(b). Afterwards, schemes to derive an overall rating prediction for a not-yet-seen item based on the ratings of its tags are proposed and evaluated.

In general, our methods extend the "Tagommender" algorithms and metrics proposed in Sen et al. [2009b] in a way that they can take into account item-specific tag preferences in the recommendation process.

3. ESTIMATING UNKNOWN TAG PREFERENCES

If no explicit tag preferences are given, Sen et al. [2009b] propose different algorithms for estimating global tag preferences from the given item ratings. According to their evaluation, the algorithm movie-ratings is both effective and at the same time relatively easy to implement. The algorithm is based on *tag relevance weighting*, a concept that is also used in the work on tag-based explanations described in Vig et al. [2009].

In our evaluation, we use a simple counting metric $w(i, t)$ to measure the relevance of a tag t for an item i.[4] The metric gives more weight to tags that have been used by

---

[4]Note that in a setting in which users rate tags, the same tag can be applied many times for the same resource.

users more often to characterize the item and is defined as follows:[5]

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

In the `movie-ratings` algorithm, the prediction $\hat{r}_{u,t}$ of the general interest of a user u in the concept represented in a tag t, that is, the tag preference, is estimated as follows:

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

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:

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

Instead of iterating over all items that received a certain tag as done in Sen et al. [2009b], 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 most similar items from I_t.

The similarity of items is measured with the adjusted cosine similarity metric. Note that 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.

Note that when using the user's explicit overall rating $r_{u,m}$ as in Equation (2), no prediction can be made for the tag preference if user u did not rate any item m tagged with t, that is, if $I_t \cap ratedItems(u) = \emptyset$. In our previous work [Gedikli and Jannach 2010], we therefore also incorporated the recursive prediction strategy from Zhang and Pu [2007] into the tag preference prediction process, which lead to a slight performance improvement. Since such an performance improvement was however also observed for Sen et al.'s original methods, we will not further discuss these generally-applicable technique here in greater depth, see Gedikli and Jannach [2010] for details of the evaluation.

4. PREDICTING ITEM RATINGS FROM TAG PREFERENCES

In Sen et al. [2009b], the best-performing tag-based recommendation algorithm with respect to *precision* was a hybrid that combined the SVM-based method `regress-tag` and the tag-agnostic matrix factorization approach `funk-svd` [Funk 2006]. In our work, we therefore propose to parameterize and evaluate the *regress-tag* method using item-specific tag preferences. Note again, that these tag preferences can be explicitly available or derived as described above in Equation (3). In addition to that, we also report accuracy results when using item-specific tag preferences for Sen et al's `cosine-tag`

---

[5]Further possible metrics to determine the relevance of a tag are described in Sen et al. [2009b].

method, in order to study how this approach, which we proposed in our previous work [Gedikli and Jannach 2010], performs on additional datasets.

The `regress-tag` *algorithm.* The `regress-tag` method from Sen et al. [2009b] is based on determining linear equations—one for each movie—which capture the possibly complex relationship between the user's tag preferences for the tags of a given item and the overall item rating. The prediction function for a user u and an item i is defined as follows, where h_0 to h_n are the coefficients of the linear equations and $\hat{r}_{u,t_i}$ from Equation (2) corresponds to the estimated tag preferences for the tags $t_1, \ldots, t_n$ attached to item i:

$$regress\text{-}tag(u, i) = h_0 + h_1 * \hat{r}_{u,t_1} + \cdots + h_n * \hat{r}_{u,t_n}. \tag{4}$$

In Sen et al. [2009b], the coefficients h_0 to h_n were chosen with the help of regression support vector machines and the *libsvm* library [Chang and Lin 2011] because this led to the most accurate results when compared with other methods for choosing the parameters such as least-squares.

In our work and our evaluations, we used the same approach (as well as the same software library and algorithm parameters), but different values for the tag preferences. In our algorithm variant, which we shall call `regress-tag-ui`,[6] we therefore use the item-specific tag preferences as described in Equation (3):

$$regress\text{-}tag\text{-}ui(u, i) = h_0 + h_1 * \hat{r}_{u,i,t_1} + \cdots + h_n * \hat{r}_{u,i,t_n}. \tag{5}$$

The `cosine-tag` *algorithm.* The `cosine-tag` method is inspired by an analogy to classical item-to-item collaborative filtering methods. The prediction of a user's rating for a given item is based on the weighted combination of the user's preference for the tags of an item. Let T_i be the set of all tags applied to an item i. The rating prediction for a user u and an item i is calculated as follows:

$$cosine\text{-}tag(u, i) = \overline{r_u} + \frac{\sum_{t \in T_i} sim(i, t) * (\hat{r}_{u,i,t} - \overline{r_{u,i}})}{\sum_{t \in T_i} sim(i, t)}. \tag{6}$$

The individual tag preferences are weighted according to the adjusted cosine similarity between items and tags. 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 i (U_i), and their predicted tag preferences for that item, that is,

$$sim(i, t) = \frac{\sum_{u \in U_i} (r_{u,i} - \overline{r_u})(\hat{r}_{u,i,t} - \overline{r_{u,i}})}{\sqrt{\sum_{u \in U_m} (r_{u,i} - \overline{r_u})^2} \sqrt{\sum_{u \in U_i} (\hat{r}_{u,i,t} - \overline{r_{u,i}})^2}}. \tag{7}$$

To illustrate the effect of the approach, consider the following example. Table III shows an example of inferred tag preferences for the MovieLens user Bob[7] and the movie Snatch (2000) on a 5-star scale with half-star increments. Bob is a real user in the MovieLens movie recommendation community. From the inferred rating data, we can see that Bob has particularly liked the British elements and tone in the movie and that Bob probably likes the acting of Jason Statham more than the acting of Brad Pitt, at least in this movie.

When we use Equation (6) to combine the estimated tag preferences into one overall rating prediction for that movie, we predict a rating value of 3.355 from Bob for Snatch. Note that Bob's overall (explicitly given) rating was 3. Our method in that case therefore predicts a rating value that is a little higher and closer to the next higher 3.5-star rating.

---

[6]"ui" stands for *user- and item-specific* tag preferences.

[7]Bob is a pseudonym to protect the user's privacy.

Table III. The Inferred Tag Preferences of the MovieLens User
Bob for the Movie Snatch (2000)

Tag	Rating	Tag	Rating
British	4.938	boxing	3.464
Guy Ritchie	4.328	cynical	3.346
crime	4.241	comedy	3.346
hilarious	4.115	Brad Pitt	3.346
Jason Statham	3.835	fighting	3.346
goofy	3.636	quirky	3.346

5. EVALUATION

In order to measure the predictive accuracy of the presented methods we evaluated our approach on two datasets using common experimental procedures and accuracy metrics. The results of this evaluation are described in this section. The goal of our subsequent evaluation is to analyze in which cases the predictions made by the system are more accurate than previous methods when we rely on user- and item-specific tag preferences.

5.1. Datasets, Tag Quality and Data Preprocessing

We used two datasets in our evaluation. First, we evaluated our methods on the "Movie-Lens 10M Ratings, 100k Tags" (ML) dataset,[8] which was also used in the analysis by [Sen et al. 2009b]. The dataset consists of movie ratings on a 5-star scale with half-star increments. In addition, it contains information about the tags that 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 dataset is the only publicly available dataset that 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.

Second, we used a new dataset containing explicit tag preferences, which we collected in a user study on the usage of tagging data for explanation purposes reported in Gedikli et al. [2011b]. The dataset contains 353 overall ratings for 100 movies provided by the 19 participants of the study. In addition to these overall ratings, the study participants provided 5,295 explicit ratings for the tags attached to the movies using the graphical interface shown in Figure 2. On average, every user rated about 18 movies and each movie had 15 tags assigned.

Limited tag quality is one of the major issues when developing and evaluating approaches that operate on the basis of user-contributed tags. In Sen et al. [2007], for example, it was observed 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, Gemmell et al. [2009], Sen et al. [2007], or Sen et al. [2009a].

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 preference 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 that are not. Still, we believe that this is one key question that was not considered before and that should be taken into account in future approaches to extracting rating information for tags automatically.

---

[8]http://www.grouplens.org/node/73.

Table IV. Quality Parameters

Constraint		Description
Min Users/Tag	(U/T)	Minimum number of users per tag.
Min Items/Tag	(I/T)	Minimum number of items per tag.
Min Tags/User	(T/U)	Minimum number of tags a user has specified.
Min Items/User	(I/U)	Minimum number of items rated by a user.
Min Tags/Item	(T/I)	Minimum number of tags applied to an item.
Min Users/Item	(U/I)	Minimum number of users that rated an item.

Table V. MovieLens DataSet Variations

Name	Minimum constraints						Resulting dataset			
	U/T	I/T	T/U	I/U	T/I	U/I	Ratings	Users	Items	Tags
ML-high	4	4	25	20	5	5	33,973	149	668	19,976
ML-medium	3	3	20	20	5	5	77,578	179	2,952	41,821
ML-low	2	2	1	1	2	1	134,829	3,979	4,713	77,127

Table VI. User Study DataSets

Name	Description
US-real	Only the explicit tag preferences provided by the study participants are used.
US-mixed	50% of the real ratings are randomly removed and replaced by estimates.
US-pred	Existing tag preferences are removed and predicted using Equation (3).

For the MovieLens (ML) dataset, we applied and varied the constraints shown in Table IV in order to remove tags, users or items for which not sufficient data was available. This way, we varied the quality of the existing tag information. We for example only considered movies, for which a minimum number of tags is assigned (Min Tags/Item). This approach was also followed by previous work. Vig et al. [2009], for example, 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 Vig et al. [2009] to detect redundant tags such as "violent" and "violence", in order to replace them by one representative tag. Similar to their approach, we further automatically preprocessed the data in three dimensions by removing stop-words from the tags, by applying stemming [Porter 1997] and by filtering out noisy tags, that is, tags with a certain amount of characters that are not letters, numbers or spaces, for example, elements such as smileys.

We created three different versions from the tag-enhanced MovieLens data with different constraints on data density; see Table V for an overview. Note that our quality and density requirements are relatively weak when compared for example with the work of Vig et al. [2009], who required that a tag has been used by at least five users to be considered in the evaluation. As a result, the MAE values reported in our work are in general slightly higher than those reported in Sen et al. [2009b], who also used similar types of constraints to prune the dataset.

In contrast to the MovieLens dataset, our second dataset from the user study (US) contains explicit tag preferences. Therefore, no estimation of the tag preferences is required. In our experiments, we however varied the amount of explicitly available tag preference data in order to study the differences when using only explicit ratings, only automatically inferred ratings and a setting where one half of the rating data is real and one half only estimated. See Table VI for an overview.

Note that for all variations of the user study dataset, the number of items, users and ratings, that is, the density levels, are the same as no further quality improvement measures were applied.

5.2. Algorithms

We included the following recommendation algorithms in our comparative evaluation. The schemes proposed in this work are:

—`regress-tag-ui`: SVM regression with user- and item-specific tag preferences (Equation (5));
—`cosine-tag-ui`: similarity-based prediction as described in Equation (6).

Beside these two methods, we also experimented with the probabilistic approach to content-based recommendation proposed in Mooney and Roy [2000]. Instead of extracting keywords from documents as in the original approach, we simply viewed the tags of the movie as the keywords and used them to derive an estimate of whether a user will like or dislike a movie. An experimental evaluation on the ML dataset however showed that the predictive accuracy of this simplistic approach is not competitive when compared to other techniques. This observation is in some respect in line with the observation made in Pilászy and Tikk [2009], in which it was also noticed that rating-based collaborative filtering schemes are more accurate than pure content- or metadata-based approaches even if the number of ratings is relatively low.

As a baseline for the comparison, we used the following algorithms:

—`regress-tag`: SVM regression as proposed in Sen et al. [2009b] (Equation (4));
—`cosine-tag`: As proposed in Sen et al. [2009b] (Equation (2)).

Note that in Sen et al.'s work a hybrid algorithm that combined `regress-tag` with a matrix factorization approach performed slightly better than `regress-tag` alone. We experimented with this hybridization strategy also on our datasets, but could not reproduce the findings of Sen et al. Given our data, we consistently observed `regress-tag` to be the best-performing method, which we therefore use as a baseline.

Beside the two tag-aware baseline algorithms, we also measured the predictive accuracy of two tag-agnostic techniques.

—`item-item`. A 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:

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

—`funk-svd`. The recent, highly-accurate matrix factorization algorithm based on Singular Value Decomposition (SVD) proposed in [Funk 2006], which was also used as a baseline for comparison by Sen et al.

We used the following algorithm parameters.

—We chose adjusted cosine as similarity metric for all schemes because experiments with other similarity metrics such as Pearson's correlation coefficient led to poorer results.
—In the user- and item-based scheme `cosine-tag`, the parameter k that 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 varied the parameter systematically. A neighborhood-size of 3 was determined as an optimal choice.
—In the `regress-tag` method, similar to the experiments reported in [Sen et al. 2009b], we set the value of c related to the error penalty to 0.005. For `regress-tag-ui` we additionally determined 10 as a suitable number of neighbors to include when calculating the item-specific tag preferences.

—For the SVD-based recommender, 30 was empirically chosen as the number of latent features.

All algorithms were implemented in our Java-based framework called *Recommender-Suite*, which also includes components to run offline experiments, do cross-validation and measure various metrics such as accuracy or coverage. The regress-tag methods are based on the libsvm implementation [Chang and Lin 2011]; the implementation of the matrix factorization approach was adapted from the Apache Mahout project.[9]

5.3. Accuracy Metrics

We analyzed the quality of the recommendations generated by the different algorithms with the standard information retrieval metrics *precision* and *recall*. To determine precision and recall, we followed the evaluation procedure proposed in Nakagawa and Mobasher [2003] and converted the rating predictions into "like" and "dislike" statements as described in Sandvig et al. [2007], where ratings above the user's mean rating are interpreted as "like" statements.

In each of the iterations of a cross-validation procedure, the dataset is split into a training set and a test set. We then determined the set of existing "like" statements (ELS) in the 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 $(|PLS \cap ELS|)/|PLS|$ and measures the number of correct predictions in PLS. Recall[10] is measured as $(|PLS \cap ELS|)/|ELS|$ 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 dataset and then averaged. These averaged precision and recall values are then combined in the usual F1-score, where $F = (2 * precision * recall)/(precision + recall)$.

Beside these information retrieval metrics, we also report the usual *Mean Absolute Error* (MAE) numbers in order to make our results comparable with the results in literature, in particular with Sen et al.'s work. Note that we also calculated *Root Mean Squared Error* values (RMSE), but do not report these numbers here because no significant differences to the MAE values have been observed.

5.4. Results and Discussion

5.4.1. Experiments on Real Data. Table VII shows the Mean Absolute Error values for the different algorithms and the datasets from the user study in increasing order,[11] see Table VI for a description of the datasets obtained from our user study. It is important to recall that the density levels of the datasets are fixed, whereas the amount of explicitly available tag preference data is varied as described in Section 5.1.

We can see that the regress-tag-ui scheme proposed in this work to exploit user- and item-specific tag preferences leads to the smallest Mean Absolute Error values for all dataset variations. The improvement over the previous regress-tag method is particularly strong when only real rating data is used. For the situation in which we

---

[9]http://mahout.apache.org.
[10]In Nakagawa and Mobasher [2003], this metric is called *coverage*.
[11]Note that we only report two decimal places in the tables due to standard error of the entries and randomness effects. However, the order of the entries is based on the third decimal place.

Table VII. MAE Results for Real Tag Preference Data

Algorithm	US-real	US-mixed	US-pred
regress-tag-ui	0.46	0.47	0.49
regress-tag	0.48	0.49	0.49
funk-svd	0.58	0.58	0.58
cosine-tag	0.59	0.59	0.60
item-item	0.59	0.59	*0.59*
cosine-tag-ui	0.60	0.59	0.60

Table VIII. F1, Precision, and Recall Results for Real Tag Preference Data

Algorithm	US-real			US-mixed			US-pred		
	F1	Prec.	Recall	F1	Prec.	Recall	F1	Prec.	Recall
regress-tag-ui	84.07	84.66	83.50	84.43	85.03	83.86	83.30	83.90	82.73
regress-tag	83.30	83.90	82.73	83.30	83.90	82.73	83.30	83.90	82.73
funk-svd	72.39	72.99	71.82	71.97	72.57	71.41	72.39	72.99	71.82
cosine-tag	68.90	69.50	68.33	68.90	69.50	68.33	68.90	69.50	68.33
cosine-tag-ui	68.66	69.26	68.10	68.23	68.83	67.67	68.05	68.65	67.49
item-item	68.05	68.65	67.49	68.05	68.65	67.49	67.30	67.99	66.82

Table IX. MAE Results for Tag-Enhanced MovieLens Dataset

Algorithm	ML-high	ML-medium	ML-low
regress-tag-ui	0.64	0.64	0.69
funk-svd	0.64	0.65	0.71
regress-tag	0.65	0.65	0.69
cosine-tag-ui	0.68	0.70	0.85
cosine-tag	0.68	0.70	0.85
item-item	0.68	0.70	0.85

only rely on estimates of item-specific tag preferences (US-pred), our method is minimally better or at least on a par with the previous method. When using only half of the existing ratings, MAE values somewhere in the middle between the two extremes can be observed. As a result, the numbers indicate that the accuracy constantly increases when more real tag preferences are entered into the system.

Another observation on this relatively dense data from the user study is that the cosine-tag-ui method performs slightly worse than the cosine-tag and even the item-item algorithm. For the dataset US-pred, item-item also slightly outperforms cosine-tag.

Table VIII shows the corresponding values for precision, recall and the F1-measure. The values were obtained using a 10-fold cross-validation on the relatively small dataset.

Again, the numbers show that regress-tag-ui slightly outperforms regress-tag and the other algorithms and is significantly better than the other methods. The neighborhood- and similarity-based methods show the poorest results on this metric and are also outperformed by the SVD-based algorithm. Note that there are virtually no differences in the observed numbers across the different datasets, which can be accounted to the characteristics of the dense user study dataset and the chosen evaluation metric.

5.4.2. MovieLens Data. Table IX shows the MAE numbers for the different MovieLens datasets. Note that unlike the three datasets considered before, the different MovieLens datasets have different density levels, as described in Section 5.1.

In these experiments, in which only estimated tag preferences are available, the funk-svd method, regress-tag and regress-tag-ui are more or less on a par with respect to the MAE measure. The more traditional neighborhood-based methods are measurably less accurate. These findings are in line with the MAE numbers reported in

Table X. F1, Precision and Recall Results for Tag-Enhanced Movielens Dataset

Algorithm	ML-high			ML-medium			ML-low		
	F1	Prec.	Recall	F1	Prec.	Recall	F1	Prec.	Recall
regress-tag-ui	70.22	70.22	70.22	70.80	70.81	70.79	94.37	94.45	94.28
regress-tag	69.98	69.98	69.98	70.52	70.54	70.51	94.36	94.44	94.27
item-item	68.89	68.89	68.89	68.14	68.15	68.12	90.50	90.59	90.41
funk-svd	67.84	67.84	67.83	67.31	67.32	67.29	93.92	94.00	93.83
cosine-tag	66.42	66.43	66.42	64.32	64.34	64.31	90.72	90.81	90.64
cosine-tag-ui	65.47	65.47	65.46	64.03	64.04	64.01	*90.98*	*91.06*	*90.89*

Sen et al. [2009b], where the `funk-svd` method was reported be even slightly better on the MAE metric by a very small margin. For the dataset ML-low—the largest dataset with the least constraints on the data—the performance of `funk-svd` is slightly worse than the regression-based approaches.

Table X finally shows F1, precision and recall figures for the three MovieLens-derived datasets, in which artificial tag preferences were used. The results show that our new `regress-tag-ui` scheme outperforms the best-performing algorithm `regress-tag` from Sen et al. on the denser datasets (ML-high and ML-medium) and is at least on a par with it on the lower-density dataset ML-low. Quite interestingly, the classical tag-agnostic item-to-item algorithm performs relatively well on this metric and is even better than the matrix factorization approach on the datasets ML-high and ML-medium. A further analysis of this phenomenon and the specific characteristics of the datasets is part of our current work.

Note that the findings of our previous work, in which `cosine-tag-ui` was presented and compared with `cosine-tag` [Gedikli and Jannach 2010], could be reproduced, that is, that `cosine-tag-ui` is slightly better than `cosine-tag` on the F1 metric. On the denser datasets, the `cosine-tag-ui` method however does not perform as well as the other algorithms.

With respect to the absolute recall and precision, we can observe that the measured precision and recall values are mostly very similar to each other for each algorithm and dataset. This is caused by the nature of the particular accuracy metric that we use in our work (see Section 5.3 and Nakagawa and Mobasher [2003] respectively).[12]

The absolute values are also significantly higher for the very sparse dataset ML-low, which is also caused by the characteristics of the chosen evaluation metric, which is not designed for a comparison of the absolute numbers across such different datasets. Note that in our experiments, the data density for the real data from the user study is close to 20%. For the MovieLens datasets, we varied the density levels from 35% for ML-high over 23% for ML-medium to the very low-density dataset ML-low, which has a density of 0.001%.

The main objective of this work is to show that accuracy improvements can be achieved when using tag preference data. Overall, the results achieved showed that the usage of item-specific tag preferences can help to improve the predictive accuracy of recommender systems and that the observed positive effect is stronger, when the tag quality and data density increases. The evaluation on the different datasets demonstrated that especially when using real tag preference data the method `regress-tag-ui` proposed in this work outperforms the best-performing method from previous work. The results show further that even when all tag preferences are artificially derived from the item ratings our method is still able to slightly outperform or at least is on

---

[12]The number of items (N) retrieved by a Top-N recommender during the evaluation is user-dependent and not fixed to a constant value. This can lead to the same precision and recall values because the denominator of the precision and recall values can have the same size. This procedure is good in scenarios where the average number of positive items in the test set varies strongly for each dataset as in our case.

a par with previous best-performing method. The predictive accuracy increases when more real data is entered into the system.

6. RELATED WORK AND DISCUSSION

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 [de Gemmis et al. 2008; Tso-Sutter et al. 2008; Liang et al. 2009; Zhen et al. 2009; Sen et al. 2009b; Kim et al. 2010; Wang et al. 2010; Zhou et al. 2009].

Tag-enhanced recommenders. In de Gemmis et al. [2008], for example, 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 approach with multicriteria 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 [Tso-Sutter et al. 2008; Liang et al. 2009; Zhen et al. 2009; Kim et al. 2010] or [Wang et al. 2010]. Most commonly, tags are considered only as an additional source of information for their proposed methods. In Wang et al. [2010], for example, tags are used for building user and item neighborhoods. The underlying idea of this approach is that neighbors that are determined this way will be better predictors than those that are identified only based on explicit rating data. The evaluation on the tag-enhanced MovieLens dataset shows that such an approach outperforms other algorithms based on nonnegative matrix factorization and Singular Value Decomposition. In particular, the observed improvements in predictive accuracy were comparably strong for sparse datasets. Similar to Sen et al.'s and our work, Zhou et al. [2009] recently proposed a method based on probabilistic factor analysis framework that exploits the information contained in the user-item rating matrix, the user-tag tagging matrix and the item-tag tagging matrix to produce more accurate recommendations. A comparison with two recent tag-agnostic matrix factorization approaches [Funk 2006; Salakhutdinov and Mnih 2008] showed that their approach leads to better RMSE values in particular when only a small part of the available data is used for training.

Note that the idea of allowing the community to rate items is an important topic not only in the area of recommender research but also in the Semantic Web community. Revyu[13] [Heath and Motta 2007], the winner of the Semantic Web Challenge of the year 2007, is a reviewing and rating Web site that 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 metadata to items in the form of Web 2.0 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

---

[13]http://revyu.com.

particularly novel. The real benefit lies in the use of Semantic Web technologies and standards like RDF, SPARQL and the principles of Linked Data [Berners-Lee 2006] 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 that 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.

Relation to multidimensional recommenders. In contrast to works in which tags are only used to build better neighborhoods for classical collaborative filtering systems, we follow a different approach of exploiting tags for recommender systems in this work. In principle, we propose an approach, in which users rate items by rating tags, which to some extent also has a correspondence to a multicriteria or multidimensional recommendation approach as described in Adomavicius and Tuzhilin [2001] or Adomavicius and Kwon [2007]. Adomavicius and Kwon [2007] conjecture that multicriteria ratings will play an important role for the next generation of recommender systems, in particular because multicriteria 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, multicriteria rating information can serve as a valuable source for explaining recommendations. Based on these observations, new user similarity metrics and algorithms were designed in Adomavicius and Kwon [2007] that exploit multicriteria rating information leading to recommender systems of higher quality. The authors show on a small dataset how exploiting multicriteria ratings can be successfully leveraged to improve recommendation accuracy. Our approach of "rating items by rating tags" shares the advantages of these multicriteria recommender systems such as improved accuracy and explanations. The rating dimensions are however not static in our approach and require metrics that are different from those put forward for example in Adomavicius and Kwon [2007]. In this respect, our work is also in line with the ideas of Shirky [2005], 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; in addition, new rating dimensions may emerge over time that were not covered by the predefined and prethought 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.

Tag preference. To the best of our knowledge, the concept of tag preference was first introduced by Ji et al. [2007]. 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 that is supported by Social Web platforms such as Delicious[14] or Connotea[15]. 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 naïve Bayes classifier is used for making recommendations by exploiting the user's candidate tag set. The proposed algorithm was evaluated on a dataset collected from the social bookmarking site Delicious. In contrast to the work of Sen et al. [2009b] the tag preference predictor in Ji et al. [2007] does not make use of item ratings at all because the Delicious dataset does not support ratings for items (bookmarks) like the tag-enhanced MovieLens dataset.

---

[14]http://www.del.icio.us.

[15]http://www.connotea.org.

Vig et al. [2009] 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 work *tag preference* was considered a user-specific concept whereas *tag relevance* is considered to be an item-specific concept. In contrast, in our work our proposed concept of *tag preference* is user- and item-specific, which has shown to be a helpful means to capture the user's preferences more precisely and thus produce more accurate recommendations.

Beside the relation of the work in Sen et al. [2009b], which we extend by item-specific tag preferences in this article, our approach is also closely related to the recent work of Vig et al. [2010], who experimented with a recommender system interface that allowed users to assign affect to the tags of a movie. Vig et al. [2010], introduce the idea of "tag expressions," which, at its heart, represents the same idea of rating items by rating tags proposed in our own previous work [Gedikli and Jannach 2010]. Users are able to assign a so-called affect (preference)—like, dislike or neutral—to tags in order to measure a user's pleasure or displeasure with the item with respect to the tag. In contrast to our work the authors are focussing on user interface aspects and how the possibility to express tag preferences affects the tagging behavior of the community. In particular, Vig et al. also analyze the design space of tag expressions and focus on three elements: preference dimensions, affect expression and display of community affect. In our work, however, we present first algorithms that consider tag preferences (tag affects) to generate more accurate predictions, which is one of the main challenges listed in the future work section of Vig et al. [2010]. Additionally we also show how to infer the user opinion regarding a certain feature (tag) for a given item automatically.

7. SUMMARY AND OUTLOOK

The main new idea of our work is to incorporate item-specific ratings for tags in the recommendation process. Based on 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 multidimensional 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 article was to develop and evaluate different recommendation schemes that take item-specific tag preferences into account when generating rating predictions. In addition, we proposed a metric to automatically derive user- and item-specific tag preferences from the overall ratings based on item similarities in order to demonstrate that quality improvements can be achieved even when the tag preference data is not explicitly given. The results of the evaluation on two datasets show that a measurable accuracy improvement can be achieved.

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

—*Further experiments and user interfaces*. We plan to conduct further experiments with real tag preference data. A particular question to be answered in that context is that of an appropriate user interface (see also Vig et al. [2010] or Vig et al. [2009]) 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 5-star scale with half-star increments 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 Tintarev and Masthoff [2007], for example, seven possible advantages of an explanation facility are described. In Vig et al. [2009], the authors have evaluated explanation interfaces that use tag relevance and tag preference as two key components. In our current work [Gedikli et al. 2011b] we examine the role of tag preferences in helping users understand their recommendations. If tags are both user- *and* item-specific, more personalized and detailed, multidimensional explanations can be provided. Based on appropriately designed explanation interfaces, the different aspects of explanations as discussed in Tintarev and Masthoff [2007] (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, Zhang et al. [2009]. We expect that the value of item-specific tag preferences is even higher, when the overall set of tags used in the dataset is more consistent.

—*New tag quality metrics*. We have stated in Section 5.1 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 preferences.

Finally, by making the software used in our experiments publicly available,[16] 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.

REFERENCES

ADOMAVICIUS, G. AND KWON, Y. 2007. New recommendation techniques for multicriteria rating systems. *IEEE Intell. Syst. 22*, 3, 48–55.

ADOMAVICIUS, G. AND TUZHILIN, A. 2001. Extending recommender systems: A multidimensional approach. In *Proceedings of the Workshop on Intelligent Techniques for Web Personalization (ITWP'01)*. 4–6.

BERNERS-LEE, T. 2006. Linked data. http://www.w3.org/DesignIssues/LinkedData.html.

BOGERS, T. AND VAN DEN BOSCH, A. 2009. 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)*. 9–16.

CHANG, C.-C. AND LIN, C.-J. 2011. Libsvm: A library for support vector machines. http://www.csie.ntu.edu.tw/~cjlin/libsvm.

DE GEMMIS, M., LOPS, P., SEMERARO, G., AND BASILE, P. 2008. Integrating tags in a semantic content-based recommender. In *Proceedings of the ACM Conference on Recommender Systems (RecSys'08)*. 163–170.

DIEDERICH, J. AND IOFCIU, T. 2006. 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)*. 288–297.

FUNK, S. 2006. Netflix update: Try this at home. http://sifter.org/~simon/journal/20061211.html.

GEDIKLI, F., GE, M., AND JANNACH, D. 2011a. Explaining online recommendations using personalized tag clouds. *i-com 10*, 1, 3–10.

GEDIKLI, F., GE, M., AND JANNACH, D. 2011b. Understanding recommendations by reading the clouds. In *Proceedings of the 12th International Conference on Electronic Commerce and Web Technologies (EC-Web'11)*.

---

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

GEDIKLI, F. AND JANNACH, D. 2010. Rating items by rating tags. In *Proceedings of the 2nd Workshop on Recommender Systems and the Social Web (RSWEB'10)*. 25–32.

GEMMELL, J., RAMEZANI, M., SCHIMOLER, T., CHRISTIANSEN, L., AND MOBASHER, B. 2009. The impact of ambiguity and redundancy on tag recommendation in folksonomies. In *Proceedings of the ACM Conference on Recommender Systems (RecSys'09)*. 45–52.

HEATH, T. AND MOTTA, E. 2007. Revyu.com: A reviewing and rating site for the web of data. In *The Semantic Web, Proceedings of the 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference (ISWC'07 + ASWC'07)*. 895–902.

HOTHO, A., JASCHKE, R., SCHMITZ, C., AND STUMME, G. 2006. Information retrieval in folksonomies: Search and ranking. In *Proceedings of the 3rd European Semantic Web Conference (ESWC'06)*. 411–426.

JANNACH, D., ZANKER, M., FELFERNIG, A., AND FRIEDRICH, G. 2010. *Recommender Systems An Introduction*. Cambridge University Press.

JÄSCHKE, R., MARINHO, L. B., HOTHO, A., SCHMIDT-THIEME, L., AND STUMME, G. 2007. Tag recommendations in folksonomies. In *Proceedings of the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'07)*. 506–514.

JI, A.-T., YEON, C., KIM, H.-N., AND JO, G.-S. 2007. Collaborative tagging in recommender systems. In *Proceedings of the 20th Australian Joint Conference on Artificial Intelligence (AUS-AI'07)*. 377–386.

KIM, H.-N., JI, A.-T., HA, I., AND JO, G.-S. 2010. Collaborative filtering based on collaborative tagging for enhancing the quality of recommendation. *Electron. Commerce Res. Appl. 9*, 1, 73–83.

LIANG, H., XU, Y., LI, Y., AND NAYAK, R. 2009. Collaborative filtering recommender systems based on popular tags. In *Proceedings of the 14th Australasian Document Computing Symposium (ADCS'09)*.

MOONEY, R. J. AND ROY, L. 2000. Content-based book recommending using learning for text categorization. In *Proceedings of the 5th ACM Conference on Digital Libraries (DL'00)*. 195–204.

NAKAGAWA, M. AND MOBASHER, B. 2003. A hybrid web personalization model based on site connectivity. In *Proceedings of the Workshop on Web Mining and Web Usage Analysis (WebKDD'03)*. 59–70.

PILÁSZY, I. AND TIKK, D. 2009. Recommending new movies: Even a few ratings are more valuable than metadata. In *Proceedings of the ACM Conference on Recommender Systems (RecSys'09)*. 93–100.

PORTER, M. F. 1997. *An Algorithm for Suffix Stripping*. Morgan Kaufmann Publishers Inc., San Francisco, CA, 313–316.

SALAKHUTDINOV, R. AND MNIH, A. 2008. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, MIT Press, 1257–1264.

SANDVIG, J. J., MOBASHER, B., AND BURKE, R. 2007. Robustness of collaborative recommendation based on association rule mining. In *Proceedings of the ACM Conference on Recommender Systems (RecSys'07)*. 105–112.

SEN, S., HARPER, F. M., LAPITZ, A., AND RIEDL, J. 2007. The quest for quality tags. In *Proceedings of the International ACM Conference on Supporting Group Work (GROUP'07)*. 361–370.

SEN, S., VIG, J., AND RIEDL, J. 2009a. Learning to recognize valuable tags. In *Proceedings of the 13th International Conference on Intelligent User Interfaces (IUI'09)*. 87–96.

SEN, S., VIG, J., AND RIEDL, J. 2009b. Tagommenders: Connecting users to items through tags. In *Proceedings of the 18th International World Wide Web Conference (WWW'09)*. 671–680.

SHIRKY, C. 2005. Ontology is overrated. http://www.shirky.com/writings/ontology_overrated.html.

TINTAREV, N. AND MASTHOFF, J. 2007. Effective explanations of recommendations: User-centered design. In *Proceedings of the ACM Conference on Recommender Systems (RecSys'07)*. 153–156.

TSO-SUTTER, K. H. L., MARINHO, L. B., AND SCHMIDT-THIEME, L. 2008. Tag-aware recommender systems by fusion of collaborative filtering algorithms. In P*roceedings of the ACM Symposium on Applied Computing (SAC'08)*. 1995–1999.

VIG, J., SEN, S., AND RIEDL, J. 2009. Tagsplanations: Explaining recommendations using tags. In *Proceedings of the 13th International Conference on Intelligent User Interfaces (IUI'09)*. 47–56.

VIG, J., SOUKUP, M., SEN, S., AND RIEDL, J. 2010. Tag expression: Tagging with feeling. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology (UIST'10)*. 323–332.

WANG, Z., WANG, Y., AND WU, H. 2010. Tags meet ratings: Improving collaborative filtering with tag-based neighborhood method. *In Proceedings of the Workshop on Social Recommender Systems (SRS'10)*. 15–23.

ZHANG, J. AND PU, P. 2007. A recursive prediction algorithm for collaborative filtering recommender systems. In *Proceedings of the ACM Conference on Recommender Systems (RecSys'07)*. 57–64.

ZHANG, N., ZHANG, Y., AND TANG, J. 2009. A tag recommendation system for folksonomy. In *Proceedings of the 2nd Workshop on Social Web Search and Mining (SWSM'09)*. 9–16.

ZHEN, Y., LI, W.-J., AND YEUNG, D.-Y. 2009. Tagicofi: Tag informed collaborative filtering. In *Proceedings of the ACM Conference on Recommender Systems (RecSys'09)*. 69–76.

ZHOU, T. C., MA, H., KING, I., AND LYU, M. R. 2009. Tagrec: Leveraging tagging wisdom for recommendation. In *Proceedings of the 12th IEEE International Conference on Computational Science and Engineering (CSE'09)* Vol. 4. 194–199.