Leveraging Publication Metadata and Social Data into FolkRank for Scientific Publication Recommendation

Stephan Doerfel
University of Kassel
doerfel@cs.uni-kassel.de

Robert Jäschke
L3S Research Center
jaeschke@l3s.de

Andreas Hotho
University of Würzburg
hotho@informatik.uni-wuerzburg.de

Gerd Stumme
University of Kassel
stumme@cs.uni-kassel.de

ABSTRACT

The ever-growing flood of new scientific articles requires novel retrieval mechanisms. One means for mitigating this instance of the information overload phenomenon are collaborative tagging systems, that allow users to select, share and annotate references to publications. These systems employ recommendation algorithms to present to their users personalized lists of interesting and relevant publications.

In this paper we analyze different ways to incorporate social data and metadata from collaborative tagging systems into the graph-based ranking algorithm FolkRank to utilize it for recommending scientific articles to users of the social bookmarking system BibSonomy. We compare the results to those of Collaborative Filtering, which has previously been applied for resource recommendation.

Categories and Subject Descriptors

H.3.5 [**Information Systems**]: On-line Information Services—*Web-based services*; H.2.8 [**Information Systems**]: Database Applications—*Data Mining*

General Terms

Design, Experimentation, Measurement

Keywords

Collaborative Tagging, FolkRank, Recommender

1. INTRODUCTION

One of the most noticeable innovations that emerged with the advent of the Web 2.0 are *collaborative tagging systems*. They allow users to annotate arbitrary resources with freely chosen keywords, so called *tags*. The tags are used for navigation, finding resources, and serendipitous browsing and thus provide an immediate benefit for the user. Examples of collaborative tagging systems are Delicious[1] for sharing web links and BibSonomy[2] for sharing publication references.

Of particular importance for every researcher are *scientific publications*. Especially during the last years, the ever faster growing number of published articles has led to the well-known phenomenon of *information overload*. It has become harder and more time-consuming for researchers to keep track of the important publications in their respective fields or to assemble comprehensive "related work" sections for a new article. The search for previously published material is often conducted on the web, using specialized search engines, editorially controlled scientific databases, or systems of user-generated content on the matter of interest. To the latter belong collaborative tagging systems like BibSonomy. They allow their users to annotate and share metadata about scientific articles and thereby help to mitigate information overload. The users (presumably) post mostly papers and scholarly work they found interesting and additionally categorize them with the tags. Still, the number of resources posted by users to these systems makes it more and more difficult to find or stumble upon interesting articles. One solution for this problem are *recommender systems* that try to suggest interesting and relevant content to the user. They employ data mining methods to leverage the wisdom of the crowds for personalized suggestions of resources.

In this paper we focus on the recommendation of scientific publications to users of the social bookmark and publication sharing system BibSonomy [1]. That is, given a user we aim to provide a ranked list of publications that might be of relevance to him. In particular, we investigate how the incorporation of additional knowledge about publications and users can improve the recommendation quality of the *FolkRank* algorithm [5]. FolkRank was found to be a well performing algorithm for tag recommendation [7] and therefore is a favored candidate for recommending resources. Further, it is relatively easy to adapt the underlying graph structure or to change the preference vector to add additional information. This paper presents research in progress that shall pave the way for a comprehensive integration of metadata into FolkRank that collaborative tagging systems often provide. In this respect its contributions are: (i) a validation of previous results on the performance of Collaborative Fil-

RSWeb'12, September 9, 2012, Dublin, Ireland.

[1] http://delicious.com/
[2] http://www.bibsonomy.org/

tering, (ii) first results on the influence of different types of metadata on the recommendation quality, and (iii) improved FolkRank results by pushing similar users, recent resources, and high-ranked resources.

This paper is structured as follows: We start with a review of related work in Section 2. Then, in Section 3, we describe the used algorithms and in Section 4 we introduce the datasets underlying our analysis. The setup of the experiments and some preliminary investigations are described in Section 5 and the results are presented in Section 6. We conclude with an outlook on future work in Section 7.

2. RELATED WORK

Resource recommendation in collaborative tagging systems has previously been discussed in the literature. Different tasks (e.g., producing recommendations given the active user – like in this work – or given a query) as well as different challenges (the coldstart problem of recommending resources to new users, the unrestricted and ambiguous vocabulary (tags), the sparsity of the data, etc.) have been addressed in various ways. The methodology for the evaluation of resource recommenders also varies greatly in the literature. This applies to the experimental setups, where different methods for splitting the data or cross-validation procedures are common, as well as to the evaluation measures. An overview on common measures is given in [9].

Parra and Brusilovsky [12], for instance, have a 3-point relevance scale (*relevant*, *somewhat relevant*, and *not relevant*) and hence use normalized discounted cumulative gain (nDCG) as evaluation measure which is particularly designed for this kind of scale. They also measure the precision in the top k recommended items for a fixed number k. They evaluate Collaborative Filtering (CF) [14] (using Pearson correlation as similarity measure) and BM25 [9] using data from CiteULike. They asked seven users to manually judge the relevance of their recommended articles. Cantador et al. [3] apply tag similarity measures to build tag context vectors for users and items which they in turn use for item recommendation. As evaluation measures they are using precision/recall at k, MAP, and nDCG. The best results are achieved using BM25, Collaborative Filtering is not considered. Similar to their approach, we are employing the tag similarity measures evaluated in [10]. Another approach making use of tag clusters to personalize recommendations is presented by Shepitsen et al. [15] where a user is not only represented as a tag vector, but as a vector of a (personalized) set of tag clusters. The authors give evidence that a user-specific choice of the set of clusters (compared to only one global clustering) yields better results on sparse data. A similar approach is presented by Wartena and Wibbels [16] with the goal of producing more diverse, topic-based recommendations. For each cluster they employ item-based CF (with items being represented by the tags that have been assigned to them) and two approaches that use the similarity between user and resources in the tag vector space. It turns out that the clustering step indeed improves the recommendation performance of each of the three methods and additionally enables more diverse recommendations.

Bogers [2] presents a comprehensive evaluation of a variety of recommendation algorithms on four different datasets and investigates the inclusion of metadata to "aid the recommendation process" as well as different hybridizations. Among the chosen algorithms, Collaborative Filtering occurs in several variations. We complement this analysis in evaluating FolkRank on similar datasets, and pointing out ways to aid also this algorithm with metadata as well as social data (user groups) or usage data (recency of posts). Bogers compares algorithms mainly using MAP (*Mean Average Precision*) – a measure for a ranked lists of recommended items – and we follow this example.

Gemmel et al. [4] build a weighted linear hybrid recommender that incorporates four collaborative filtering variants, a recommender suggesting the most popular resources, and an approach that directly recommends resources that are similar to the user in the tag vector space. They compare the hybrid's performance to the pair-wise interaction tensor factorization approach of [13] which had previously been used for tag recommendation. The CF variants are user-based, with similarities between users being computed in the resource and in the tag vector space, and item-based, with similarities between resources being computed in the user and in the tag vector space. In contrast to plain CF this kind of hybridization enables the inclusion of all three dimensions. On all datasets the hybrid outperforms each of the six contributing recommenders. The user-based CF approach using the resource vector space contributes considerably to the hybrid and performs better than or comparable to the other contributing recommenders on their own. In contrast to our approach no additional metadata is included. We can repeat the observation that for user-based CF the user similarities in the resource vector space work better than those in the tag vector space. Similar to our inclusion of group information into FolkRank, Lee and Brusilovsky [8] incorporate information about the user's groups into CF using mixed hybridization. Thereby, they combine user-based CF with (Jaccard) similarity measured in the resource space with recommendations from the group information, which in turn are a fusion of recommendations based on the group's documents and on the group members' documents. Similar to [4] the hybrid outperforms all the baseline approaches. An example for the benefit of metadata in tag recommendations is given by Musto et al. in [11].

These previous findings suggest that the combination of different dimensions and the incorporation of additional metadata can increase recommendation performance. As a crucial next step we therefore evaluate several options for the incorporation of metadata into *FolkRank* which was particularly designed to include all three dimensions of a folksonomy.

3. ALGORITHMS

Here we recall the basics of three algorithms *Collaborative Filtering*, *adapted PageRank* and *FolkRank*. A fourth recommender, that serves as an additional baseline, simply recommends the *most popular resources* of the dataset. To formally describe the algorithms we use the model of a folksonomy (the structure underlying collaborative tagging systems) as introduced in [5]: A *folksonomy* is a quadruple $\mathbb{F} := (U, T, R, Y)$, where U, T, and R are finite sets, whose elements are called *users*, *tags* and *resources*, resp., and Y is a ternary relation between them, i.e., $Y \subseteq U \times T \times R$, whose elements are called *tag assignments*. The *personomy* $\mathbb{P}_u$ of a given user $u \in U$ is then the restriction of $\mathbb{F}$ to u, i.e., $\mathbb{P}_u := (T_u, R_u, I_u)$ with $I_u := \{(t, r) \in T \times R \mid (u, t, r) \in Y\}$, $T_u := \pi_1[I_u]$, and $R_u := \pi_2[I_u]$, where π_i denotes the projection on the i-th dimension.

3.1 Collaborative Filtering

With user-based *Collaborative Filtering* (CF) [14] new resources are recommended to a user based on the preference of like-minded users. Each user u is typically represented by a vector $\vec{x}_u$ that describes the user's rating $x_{u,r}$ for every resource r, if it is known. Since the folksonomy data does not contain explicit user ratings for resources, we interpret the fact that a user bookmarked a resource as (Boolean) expression of the user's interest in that resource. To this end we reduce the ternary relation Y to a lower dimensional space as described in [7]. The vector $\vec{x}_u^R \in \{0,1\}^R$ then represents the resources the user u has bookmarked. For each $r \in R$ we set $\vec{x}_{u,r}^R = 1$ if the user u has bookmarked the resource r, 0 otherwise. We can also represent users by the tags they have used with a vector $\vec{x}_u^T \in \mathbb{N}^T$: for each $t \in T$ we set $\vec{x}_{u,t}^T = |\{r \in R \mid (u,t,r) \in Y\}|$. This variant is called CF_T in the sequel, the resource-minded one is called CF_R.

3.2 FolkRank

FolkRank [5] consists of two steps: an adaptation of the graph structure and a differential approach. First, $\mathbb{F} = (U,T,R,Y)$ is converted into an *un*directed tri-partite graph $G_\mathbb{F} = (V,E)$ where $V = U \dot\cup T \dot\cup R$ and each triple $(u,t,r) \in Y$ yields three edges in E: $\{u,t\}$, $\{u,r\}$, and $\{t,r\}$. Each edge is weighted with the number of triples containing its two nodes. The weights $\vec{w}$ of the vertices of the graph for the *adapted PageRank* (APR) are now iteratively computed as fixpoint of the equation $\vec{w}_{i+1} \leftarrow dA^{\mathrm{T}}\vec{w}_i + (1-d)\vec{p}$, where A is the row-stochastic version of the adjacency matrix of $G_\mathbb{F}$, $\vec{p}$ is a preference vector and $d \in [0,1]$ determines the influence of $\vec{p}$. Choosing $\vec{p} = \mathbf{1}$ yields a global ranking of all folksonomy elements, while a topic-specific or personalized ranking results from an assignment of preference to only certain interesting nodes (entries of $\vec{p}$). In our experiments we set $d = 0.7$ as in [7]. Finally, FolkRank is computed as the difference between the APR result and the fixpoint of the equation when d is set to 1, i.e., between the personalized and the unpersonalized ranking.

3.3 FolkRank on an Extended Folksonomy

In this work we explore two opportunities to augment FolkRank with further data. The first is the manipulation of the underlying graph through inclusion of another dimension M. The new structure, denoted $\mathbb{F} + M := (U,T,R,M,Y')$, extends the folksonomy $\mathbb{F}$ where Y' is a relation $Y' \subseteq U \times T \times R \times M$ and each triple of Y is extended with those elements of M that one of the elements of the triple is associated with. If, e.g., M is a set of user groups and a user u is member of two groups g and h then each triple $(u,t,r) \in Y$ is extended into two quadruples (u,t,r,g) and (u,t,r,h). If the new dimension M is the set of publication venues, then each triple $(u,t,r) \in Y$ yields a quadruple $(u,t,r,v(r))$ with $v(r)$ being the venue of the publication r. Every time a triple has no corresponding element in M (e.g., missing metadata fields), we insert a new artificial element into that triple and thus into M. The new element will be almost isolated in the graph of $\mathbb{F} + M$ and thus be of little influence. The adaptation of APR and FolkRank to the new graph structure is straightforward: Each quadruple (u,t,r,m) gives rise to six edges: $\{u,t\}$, $\{u,r\}$, $\{r,t\}$, as before, plus $\{u,m\}$, $\{t,m\}$ and $\{r,m\}$. The second way of including further information is the manipulation of the preference vector $\vec{p}$. We simply select users, tags, or resources that should receive preference and assign appropriate values to their entries in $\vec{p}$. Finally, note that both adapted PageRank and FolkRank are computationally more expensive than Collaborative Filtering [7].

Table 1: The datasets, their sizes, and the number of test users.

dataset	users	resources	posts	tags	test
D_{12}	5,132	483,945	543,890	149,034	–
$D_{12,R}$	2,886	29,921	84,176	28,011	590
$D_{12,UR}$	541	25,072	70,382	19,998	541
D_{08}	1,211	71,705	92,545	28,023	–
$D_{08,R}$	729	13,001	32,962	7,084	165
$D_{08,UR}$	150	11,689	29,057	4,652	150

4. DATASETS

The dataset we use for our evaluation is based on the regular dumps of the publicly available data of the social bookmark and publication sharing system BibSonomy. There are several BibSonomy datasets available for research purposes.[3] We use D_{12}, the most recent – and thus larger – one (called "2012-01-01" on the web page) and D_{08}, the one from the ECML PKDD Discovery Challenge 2008 ("rsdc08train"), which was also used by Bogers in [2]. The generation of the dataset dumps is described in [6] including a more in-depth description of the data from 2008.

For our analysis we only use the publication references and ignore the bookmarks as we are especially interested in recommending scientific articles. We restrict each of the two datasets to two subsets using the following procedure: We remove all triples (u,t,r) from Y where the resource r has been bookmarked by less than two users and then remove entities that do not occur in at least one of the remaining triples. The resulting datasets are called $D_{12,R}$ and $D_{08,R}$ and have the property that each resource that might be left out during the evaluation occurs at least once in the dataset and thus can still be selected for recommendation by any of the algorithms. We create even smaller datasets by further removing all triples (u,t,r) from $D_{12,R}$ and $D_{08,R}$ where the user u has less than 20 resources in his personomy. Thus we exclude users with only a short usage history. We repeat both removal procedures iteratively until in the resulting datasets $D_{12,UR}$ and $D_{08,UR}$ each remaining user has at least 20 resources and each resource occurs at least twice. These restrictions are commonly used (e.g., in [2, 7]) and yield a more dense dataset without too many outliers. The difference to the p-core used in [7] is that we do not require that a tag appears in a certain minimal number of posts. The sizes of the datasets can be found in Table 1.

In several experiments we add further data as new dimension M to the folksonomy $\mathbb{F}$ (Section 3.3) denoted by $\mathbb{F}+M$. As in BibSonomy users are required to specify for each resource (besides the title) its authors and its year of publication, these where considered. In the author dimension we used either the first authors, the last authors, or all authors (and editors, if no authors are given). Author names were either normalized to their first name's initial plus lastname or to only their lastname. The according data structures are ($\mathbb{F}$ + publication year), ($\mathbb{F}$ + authors), ($\mathbb{F}$ + authors (lastname)), and so on. One of the most often filled fields are the booktitles of proceedings and the journal for articles and

[3] `https://www.kde.cs.uni-kassel.de/bibsonomy/dumps/`

we use them combined as the "venue" of a publication ($\mathbb{F}$ + venue). Available for all posts is also the year a resource was posted, resulting in ($\mathbb{F}$ + posting year). Choosing the venue and author dimensions is based on the rationale that usually a journal/conference or an author is focussed on a specific subdiscipline of a larger field of science and a researcher who is interested in one article of that area might be interested in the other ones from the same area, too. Selecting the years reflects the idea that often a certain topic is investigated heavily by several researchers during a (short) period in time and thus contemporary articles might be related.

We exploit social ties among users by including the groups that some are members of ($\mathbb{F}$ + group), usually combining users with similar interests (e.g., from the same institute).[4]

Finally, we make use of the semantic structure among the tags to create sets of similar tags. For that purpose we calculate co-ocurrence-based similarities between tags following the procedure described in [10] and create a graph where each tag is connected to its most similar tag. We then assign to each tag its weakly connected component in that graph as additional metadata ($\mathbb{F}$ + similar tags). In the variation of this scenario ($\mathbb{F}^*$ + similar tags) we completely omit the tag dimension from the folksonomy and replace it by the dimension of the tag-graph's components.

5. EXPERIMENTS

In this section we discuss the setup of the experiments.

5.1 Evaluation Methodology

Since it is difficult to get information on the relevance of recommendations from the users themselves, we treat their history of posted publications as gold-standard. Hence, the relevance of a recommended publication is judged by the fact whether or not the user has posted this publication.

To be comparable to Bogers [2] we evaluate the algorithms using the *LeaveXPostsOut* setup and with MAP (*Mean Average Precision*) as quality score function. That is, one user $u \in U$ is selected and a set X_u of posts out of their personomy is withheld from the dataset. We then train each recommender algorithm on the remaining data (including the chosen user's other posts). Each algorithm's recommendation $\tilde{R}(u)$ is computed as a ranked list of resources $\tilde{R}(u) = (r_{u,1}, r_{u,2}, \ldots, r_{u,n})$ such that the resources that are supposedly of interest to the chosen user are better ranked than others and $r_{u,1}$ is the most highly recommended resource to user u. The number of recommended resources n depends on the algorithm and is bound by the number $|R|$ of all resources within the dataset. It is assumed that a good recommender would rank many of the withheld posts in X_u within the first positions of the ranking.

The complete LeaveXPostsOut procedure is repeated for a set of several users $\hat{U} \subseteq U$. The resulting MAP score for an algorithm is calculated as the mean of the average precisions in each run of the algorithm (one for each selected user). More formally, we have

$$\text{MAP} := \frac{1}{|\hat{U}|} \sum_{u \in \hat{U}} \frac{1}{|X_u|} \sum_{i=1}^{n} \text{precision}(X_u, i) \cdot \delta(X_u, r_{u,i}),$$

where $\delta(X_u, r_{u,i})$ indicates whether the resource ranked at position i is one of the withheld resources of the user u and $\text{precision}(X_u, i)$ is the fraction of withheld resources of user u within the first i positions of the produced ranking:

$$\text{precision}(X_u, i) := \frac{1}{i} |\{r_{u,1}, r_{u,2}, \ldots, r_{u,i}\} \cap X_u|.$$

A nice property of MAP is that one does not have to specify a fixed number of recommended items n. One can simply produce an ordered list of all resources (if the recommender algorithm allows that) in the order in which they would be recommended.

In our scenario we withheld for each user with more than 20 posts in their personomy their 10 most recent posts. The resulting numbers of test users are given in the last column of Table 1. Note that on the datasets $D_{12,UR}$ and $D_{08,UR}$ this means that every user is considered in the evaluation. This setup has several advantages: By withholding the most recently posted resources the setup is closer to the real application, e.g., as opposed to withholding arbitrarily selected posts. Compared to evaluation methods where the dataset is divided only once into a fixed training set and a fixed test set (used in traditional classifier evaluation), the LeaveXPostsOut method is unbiased by the selection of those users in the test set as each user (with enough posts) is considered in the evaluation. This advantage is of importance especially on small datasets where one can not consider an arbitrarily chosen (small) sample of users to be representative for the whole dataset. Finally, by testing only on users with more than 20 posts, we avoid the so called *cold start problem* of having to recommend resources to users, of whom only little is known about their interests.

Our setup is slightly different to that used by Bogers in [2]. He splits the dataset into a test and a training set by arbitrarily selecting 10% of the users (i.e., 15 users) as $\hat{U}$, and then for each such user $u \in \hat{U}$ selects ten arbitrarily chosen resources X_u for testing. The remaining users and the remaining posts of the chosen users form the training set. While parameters of the evaluated algorithms are optimized using a ten-fold cross validation on the training set, the final evaluation of an algorithm's performance is conducted only on the test set $\hat{U}$. In our experiments we found that different selections of 10% of U as test users $\hat{U}$ yield strong fluctuations of the resulting MAP scores (e.g., ranging from 0.0986 to 0.1906 in ten different but arbitrary 10%-splits of the $D_{08,UR}$ dataset) due to the rather small size of the test dataset. We therefore deviate in our setup from [2] in the way described above, yielding higher scores (see Section 6.1).

5.2 Preliminary Observations

Collaborative Filtering as well as the considerations following in Section 6.2 are based on the rationale that users that are (somehow) similar to the user at hand are valuable sources to find resources for recommendation. Therefore, we investigate for different well-known similarity functions how many of the most similar users one needs in order to find many of the withheld items in X_u within their personomies (i.e., to yield a high coverage of those items). Figure 1 shows, for different similarity functions, how many of the withheld resources can be found in the neighborhood of the most similar users for different neighborhood sizes. We tested the Cosine similarity as well as similarities based on Manhattan and on Euclidean distance, both on representations of the user-profiles as resource vectors and as tag

[4]For both datasets we use the group memberships of 2012 as the older ones are not availabe.

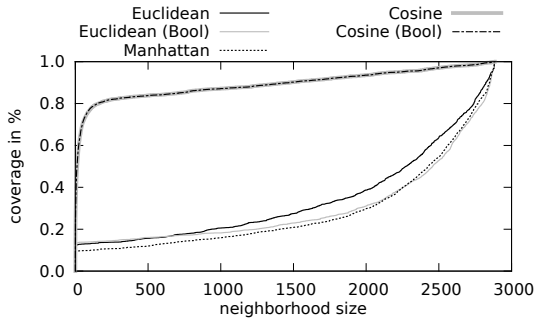
Table 2: The smallest neighborhood sizes to yield a given minimum level of average coverage (cov) of the left out resources. Displayed are the results for Cosine similarity on all datasets based on Boolean resource (res) or tag (tag) vectors.

cov	$D_{12,R}$		$D_{08,R}$		$D_{12,UR}$		$D_{08,UR}$	
of X_u	(2,886 users)		(1,211 users)		(541 users)		(150 users)	
in %	res	tag	res	tag	res	tag	res	tag
30%	3	11	2	6	1	4	1	2
50%	12	114	7	38	5	30	3	11
60%	24	230	14	69	10	54	6	18
80%	154	631	47	165	54	174	18	46
90%	1,473	1104	222	280	173	300	43	87

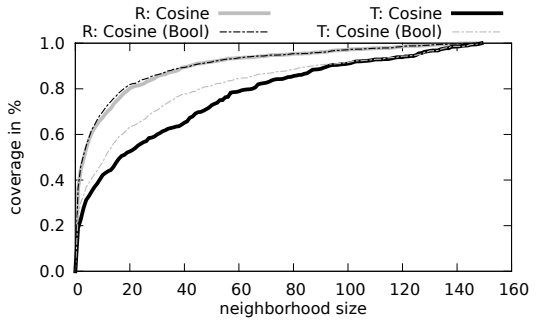
vectors. We also distinguished between Boolean representations (a user has a resource/tag at least once or not at all) and non-Boolean vectors. In the latter, the vector entries are the numbers of tags that a user assigned to a resource or the numbers of resources that a user tagged with a tag, respectively. Note, that in the Boolean case the order of similar users according to Euclidean and Manhattan distance are identical. Also, since we withhold ten items for each considered user, a hypothetical perfect similarity measure would only require neighborhoods of at most ten similar users to cover all ten withheld items – which is an (extreme) upper bound for achieving coverage with as few similar users as possible.

Figure 1(a) examplary shows the resulting coverage curves for the largest considered dataset $D_{12,R}$ using resource vectors to represent users. The results for the other three datasets are similar. For comparison, Figure 1(b) shows the best performing similarities (the four variants with Cosine similarity) for the smallest dataset $D_{08,UR}$. Table 2 displays neighborhood sizes for five coverage levels for the Cosine similarity.

In all cases, the neighborhoods based on Cosine similarity contain (on average) more of the desired resources than those based on other similarities. The resource vector space seems to be more suitable than the tag vector space. The choice of Boolean over non-Boolean vectors does not yield a large gain, but in all cases the Boolean versions of the Cosine similarity yield comparable or slightly higher coverage especially for the smaller neighborhoods. The fraction of covered resources rises quickly to approximately 80% (60%) for the resource (tag) vector space. Adding further users then yields smaller gains in coverage until finally the neighborhoods containing all other users have complete coverage – as a consequence of the dataset construction each resource occurs in at least two user profiles.



(a) $D_{12,R}$ as resource vector space; the lines of Cosine and Cosine (Bool) are almost identical.



(b) $D_{08,UR}$ with all four variants of Cosine similarity

Figure 1: The average coverage of the withheld resources in differently sized neighborhoods of similar users according to Cosine similarity and Manhattan and Euclidean distance in Boolean and non-Boolean user profiles as tag or resource vectors.

6. RESULTS

In the following we present the resulting MAP scores for our algorithms in different parametrizations.

6.1 Comparison of Approaches

We start with an evaluation of differently parametrized versions of the CF_R and CF_T variants of Collaborative Filtering and FolkRank. For CF_R and CF_T we selected – according to the results in Section 5.2 – the Cosine similarity measure and different neighborhood sizes. FolkRank was evaluated in its original version (FolkRank $\mathbb{F}$) and making use of further (social, semantic, or metadata) dimensions M as described in Section 3.2 (FolkRank $\mathbb{F}+M$). The results of these experiments are listed in Table 3. As can be seen, CF_R performs better than FolkRank and FolkRank better than CF_T. FolkRank also yields significantly better results than APR in all experiments (therefore only the baseline is reported in the table). All algorithms have higher MAP scores than the "most popular" baseline. Further, regular FolkRank ($\mathbb{F}$) performs best among the different versions (with the exception of ($\mathbb{F}$ + first authors) on $D_{12,UR}$). On average the included metadata does not improve FolkRank's MAP values. The worst scores result from including the posting and the publication year. Since only few posting years (since BibSonomy's start in 2006) can occur in the dataset and users tend to post publications that appeared recently, these dimensions consist of only few nodes. The thereby induced connections between nodes of the other dimensions seem to be not meaningful for the recommendation scenario at hand. We can further observe that on each dataset the combinations with normalized author names yield better scores than using only the authors' last names. Again this might be due to nodes of the additional dimension connecting too many nodes in the regular three dimensions.

Combining $\mathbb{F}$ with only the first authors is better than with the last authors and both are better than combining $\mathbb{F}$ with all authors. Often the first author of a paper is the one contributing most and the last author a supervisor or department head of the other authors. It therefore seems intuitive that papers of the same first author are more in-

Table 3: MAP **scores of the different algorithms in different parametrizations evaluated on the four datasets. The last three lines contain the respective highest scores for each variant of preference manipulation and in braces the respective numbers of elements that got additional preference.**

algorithm / variant		$D_{12,R}$	$D_{08,R}$	$D_{12,UR}$	$D_{08,UR}$
most popular (baseline)		0.0060	0.0129	0.0070	0.0127
CF_R	$k = 4$	0.1103	0.1394	0.1147	0.1406
	$k = 5$	0.1101	0.1382	0.1147	0.1402
	$k = 10$	0.1093	**0.1413**	0.1205	**0.1521**
	$k = 100$	0.1122	0.1296	0.1163	0.1394
	$k = \|U\| - 1$	**0.1142**	0.1365	**0.1215**	0.1395
CF_T	$k = 4$	0.0623	0.0809	0.0605	0.0881
	$k = 5$	0.0621	0.0811	0.0596	0.0811
	$k = 10$	0.0633	0.0728	0.0581	0.0755
	$k = 100$	0.0511	0.0555	0.0516	0.0575
	$k = \|U\| - 1$	0.0489	0.0595	0.0538	0.0646
adapted PageRank (APR)		0.0661	0.0583	0.0702	0.0620
FolkRank	$\mathbb{F}$	0.0900	0.1183	0.0988	0.1289
	$\mathbb{F}$ + authors	0.0850	0.1025	0.0964	0.1151
	$\mathbb{F}$ + authors (lastname)	0.0777	0.0962	0.0881	0.1083
	$\mathbb{F}$ + first authors	0.0895	0.1134	0.1020	0.1262
	$\mathbb{F}$ + first authors (lastname)	0.0768	0.1022	0.0876	0.1139
	$\mathbb{F}$ + last authors	0.0861	0.1076	0.0970	0.1201
	$\mathbb{F}$ + last authors (lastname)	0.0729	0.0989	0.0825	0.1099
	$\mathbb{F}$ + posting year	0.0684	0.0876	0.0694	0.0897
	$\mathbb{F}$ + publication year	0.0716	0.0817	0.0749	0.0847
	$\mathbb{F}$ + venue	0.0801	0.1012	0.0884	0.1150
	$\mathbb{F}^*$ + similar tags	0.0812	0.1080	0.0913	0.1212
	$\mathbb{F}$ + similar tags	0.0782	0.1007	0.0862	0.1134
	$\mathbb{F}$ + group	0.0846	0.1167	0.0927	0.1278
	preference to similar users	(1) 0.0991	(1) 0.1231	(1) 0.1087	(1) 0.1337
	preference to recent resources	(9) 0.1035	(62) 0.1329	(11) 0.1125	(49) 0.1478
	preference to reinforced resources	(50) 0.0917	(17) 0.1223	(15) 0.1020	(38) 0.1358

teresting to a user than papers which have any authors in common.

The inclusion of the users' groups works better on the two datasets $D_{12,UR}$ and $D_{08,UR}$ that are reduced in both the user and the resource dimension. We conjecture that many users with less than 20 resources (which are not in these datasets) use the system less than other users with more resources and are thus less likely to engage in groups with like-minded users. Finally, replacing the tag dimension in ($\mathbb{F}^*$ + similar tags) is better than adding components of similar tags as a fourth dimension ($\mathbb{F}$ + similar tags).

In summary, none of the additional dimensions helps FolkRank to perform better than CF_R. However, several of these dimensions yield comparable results and are thus worth further investigation.

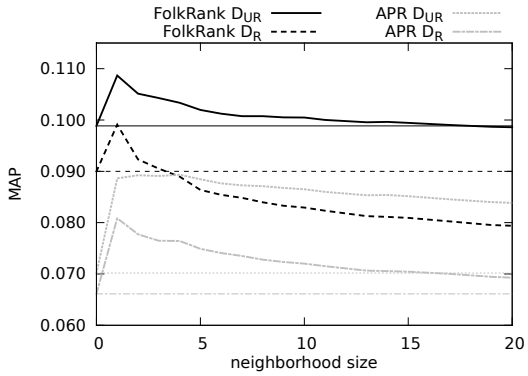
In comparison to the results of [2] we yield higher scores for the same algorithms (e.g., 0.1406 instead of 0.0865 for CF_R with neighborhoods of size 4). We conjecture that this is due to differences in the datasets (as we could not reproduce a dataset with exactly the properties reported in [2]) and the different setups: our scores are based on the whole set of users instead of a random sample of only 15 users.

6.2 Exploiting Similar Users

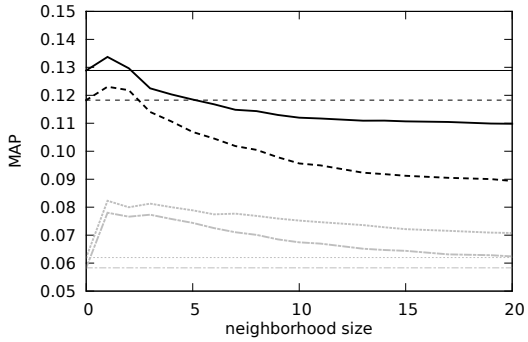
The good results of CF_R compared to FolkRank on all datasets motivated us to integrate the successful strategy of using similar users into FolkRank. We achieve this by modifying the preference vector $\vec{p}$ in FolkRank. For a target user u we select the k most similar users (according to the Cosine similarity measure) and insert their similarity value to u as weight into $\vec{p}$. The results for different neighborhood sizes k are depicted in Figure 2. All scenarios profit from the inclusion of at least small neighborhoods (the top scores are reported in Table 3). On each dataset, FolkRank achieves the best results when only the single most similar user is getting additional preference. This yields better values than FolkRank without additional preference. Increasing the neighborhood size decreases the MAP scores even below the score of the plain FolkRank already for smaller neighborhoods. Although APR can not compete with FolkRank, it is worth noting that it profits even more from the inclusion of similar users, also for larger neighborhoods. This shows that the similarity structure is not already completely captured by the structure of the graph $G_{\mathbb{F}}$ underlying FolkRank. As expected – considering the findings in Section 5.2 – using the Euclidean distance to construct the neighborhoods did only decrease the recommendation quality.

6.3 Exploiting Recent Resources

In the next experiment we took into account that a user's interest may vary during the use of the system. Thus it seems reasonable to expect that recently posted resources are an indicator for resources a user might be interested in next. Like in the experiment with similar users in the previous section, we modify the preference vector $\vec{p}$ of FolkRank

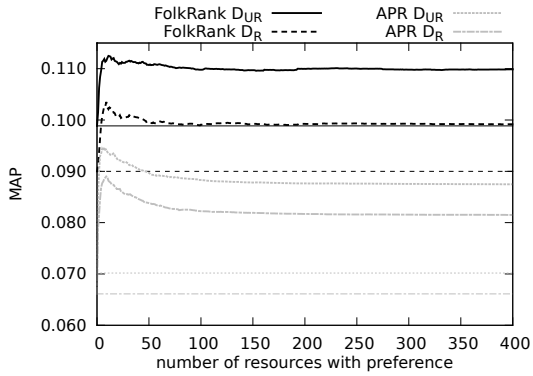


(a) Exploiting similar users on D_{12}.

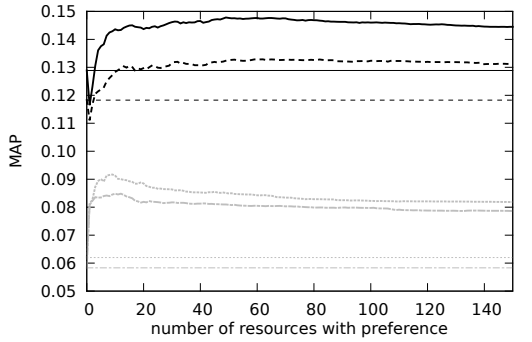


(b) Exploiting similar users on D_{08}.

Figure 2: MAP **scores for FolkRank and APR with additional preference for neighborhoods of similar users using the Cosine similarity on Boolean resource vectors. The straight lines show the according** MAP **score without additional preference.**



(a) Exploiting recent resources on D_{12}.



(b) Exploiting recent resources on D_{08}.

Figure 3: The MAP **scores for FolkRank and APR where the k most recent resources get preference (equally distributed). The straight lines show the according** MAP **score without additional preference.**

by assigning the same weight to all considered recent resources. The diagrams in Figure 3 show the resulting MAP scores, the top values are again reported in Table 3. On the two more recent datasets $D_{12,R}$ and $D_{12,UR}$, the scores rise immediately above the score of the plain FolkRank, while on the datasets from 2008 they first decrease but then also exceed the baseline (for three or more recent resources) and even some of the CF_R results. In general, the scores are comparable to those of CF_R. The optimal values are achieved at very different sizes. Including larger numbers of recent resources yields constant results. This phenomenon can in part be explained by the fact that often users do not even have that many resources to be used in $\vec{p}$. Again APR results also improve significantly but not to the level of FolkRank.

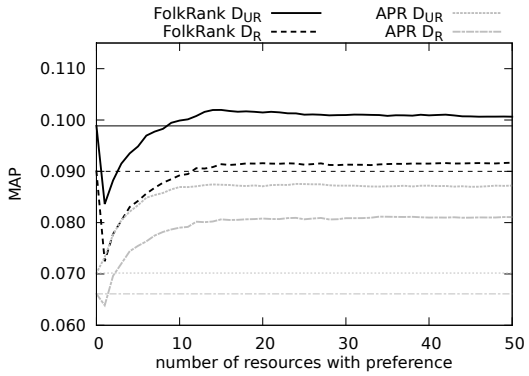
6.4 Exploiting High-Ranked Resources

The last modification of FolkRank in this work follows the idea of reinforcing relevant publications to find further related and thus possibly also relevant publications. We achieve this through two runs: From the first run of the regular version of FolkRank we collect the k best-ranked resources and give them preference (as before in $\vec{p}$) in a second run of FolkRank (or a run of APR). Giving additional preference to only one resource lets the MAP scores drop on average (Figure 4). However, using larger k values lets the score increase and again exceed the score of the regular FolkRank without a second run, although not by as much as when using recent resources.

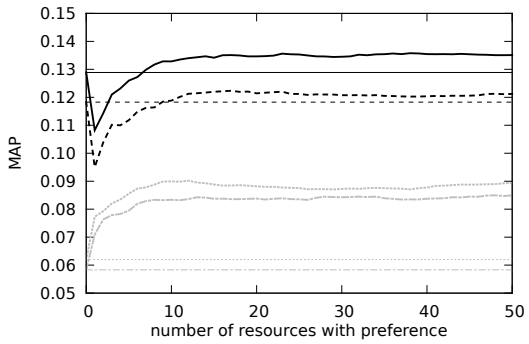
7. CONCLUSION AND FUTURE WORK

In our experiments we yield better results than Bogers in [2] for CF, presumably due to the slightly different setup. For the inclusion of metadata in FolkRank we find that it does not improve the overall recommendation performance, and generally FolkRank results are below those of CF_R but better than those of CF_T. However, some of the additional dimensions (authors or groups) yield comparable results. Like shown in [2], different recommenders perform differently on different datasets. Hence a reasonable next step would be to compare the more successful metadata strategies on other datasets and to investigate whether certain users can benefit more from the inclusion of certain kinds of data than others. For the inclusion of similar users we find that small neighborhoods are suitable to improve FolkRank recommendations. For the selection of users into these neighborhoods as well as for Collaborative Filtering the Cosine similarity is the measure of choice. We also show that the recency of a post is a valuable indicator for the current interests of a user. Including recent resources yields the best results of FolkRank in our experiments.

For future work we plan to investigate the performance of FolkRank in different parametrizations on other datasets and specifically to analyze which users profit more from which algorithms or from which kind of included metadata. To truly capture the recommendation performance we plan an online evaluation in BibSonomy, since offline evaluations can only determine how well an algorithm can retrieve re-



(a) Exploiting high-ranked resources on D_{12}.



(b) Exploiting high-ranked resources on D_{08}.

Figure 4: The MAP **scores for FolkRank and APR where the k most highly ranked resources (up to** 50**) of a FolkRank run are entered as preferred items into a second run. The straight lines show the** MAP **score for FolkRank with only one run.**

sources a user has already found without the algorithm's help. Another aspect for further experiments is to determine optimal parameters for the inclusion of data, e.g., for choosing the preference weights in $\vec{p}$ or the numbers of included similar neighbors or recent resources. Finally, despite the weaker performance when further dimensions are included, it might well turn out that certain combinations of the here proposed methods yield actually better results. More preprocessing (e.g., normalizing the venues) or densifying the tag dimension further by using other methods to create sets of similar tags are also valuable options for future work.

8. ACKNOWLEDGEMENTS

Part of this research was funded by the DFG in the project "Info 2.0 – Informationelle Selbstbestimmung im Web 2.0".

9. REFERENCES

[1] D. Benz, A. Hotho, R. Jäschke, B. Krause, F. Mitzlaff, C. Schmitz, and G. Stumme. The social bookmark and publication management system BibSonomy. *The VLDB Journal*, 19(6):849–875, Dec. 2010.

[2] T. Bogers. *Recommender Systems for Social Bookmarking*. PhD thesis, Tilburg University, Tilburg, The Netherlands, Dec. 2009.

[3] I. Cantador, A. Bellogín, I. Fernández-Tobías, and S. López-Hernández. Semantic contextualisation of social tag-based profiles and item recommendations. In *E-Commerce and Web Technologies*, volume 85 of *Lecture Notes in Business Information Processing*, pages 101–113. Springer, Berlin/Heidelberg, 2011.

[4] J. Gemmell, T. Schimoler, B. Mobasher, and R. Burke. Resource recommendation in social annotation systems: A linear-weighted hybrid approach. *Journal of Computer and System Sciences*, 78(4):1160 – 1174, 2012.

[5] A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme. Information retrieval in folksonomies: Search and ranking. In *The Semantic Web: Research and Applications*, volume 4011 of *LNCS*, pages 411–426. Springer, 2006.

[6] R. Jäschke, A. Hotho, F. Mitzlaff, and G. Stumme. Challenges in tag recommendations for collaborative tagging systems. In *Recommender Systems for the Social Web*, volume 32 of *Intelligent Systems Reference Library*, pages 65–87. Springer, 2012.

[7] R. Jäschke, L. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme. Tag recommendations in social bookmarking systems. *AI Communications*, 21(4):231–247, Dec. 2008.

[8] D. H. Lee and P. Brusilovsky. Using self-defined group activities for improving recommendations in collaborative tagging systems. In *Proc. 4th Conf. on Recommender Systems*, pages 221–224. ACM, 2010.

[9] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, 2008.

[10] B. Markines, C. Cattuto, F. Menczer, D. Benz, A. Hotho, and G. Stumme. Evaluating similarity measures for emergent semantics of social tagging. In *Proc. 18th Int. Conf. on World Wide Web*, pages 641–641, April 2009.

[11] C. Musto, F. Narducci, P. Lops, and M. de Gemmis. Combining collaborative and content-based techniques for tag recommendation. In *E-Commerce and Web Technologies*, volume 61 of *Lecture Notes in Business Information Processing*, pages 13–23, Berlin/Heidelberg, 2010. Springer.

[12] D. Parra and P. Brusilovsky. Evaluation of collaborative filtering algorithms for recommending articles on citeulike. In *Proceedings of the Workshop on Web 3.0: Merging Semantic Web and Social Web*, volume 467 of *CEUR Workshop Proceedings*, 2009.

[13] S. Rendle and L. Schmidt-Thieme. Pairwise interaction tensor factorization for personalized tag recommendation. In *Proc. 3rd Int. Conf. on Web Search and Data Mining*, pages 81–90. ACM, 2010.

[14] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *Proc. 10th Int. Conf. on World Wide Web*, pages 285–295. ACM, 2001.

[15] A. Shepitsen, J. Gemmell, B. Mobasher, and R. Burke. Personalized recommendation in social tagging systems using hierarchical clustering. In *Proc. 2nd Conf. on Recommender Systems*, RecSys '08, pages 259–266. ACM, 2008.

[16] C. Wartena and M. Wibbels. Improving tag-based recommendation by topic diversification. In *Advances in Information Retrieval*, volume 6611 of *LNCS*, pages 43–54. Springer, Berlin/Heidelberg, 2011.