Latent Dirichlet Allocation for Tag Recommendation

Ralf Krestel
L3S Research Center
Leibniz Universität Hannover
Germany
krestel@L3S.de

Peter Fankhauser
L3S Research Center
Leibniz Universität Hannover
Germany
fankhauser@L3S.de

Wolfgang Nejdl
L3S Research Center
Leibniz Universität Hannover
Germany
nejdl@L3S.de

ABSTRACT

Tagging systems have become major infrastructures on the Web. They allow users to create tags that annotate and categorize content and share them with other users, very helpful in particular for searching multimedia content. However, as tagging is not constrained by a controlled vocabulary and annotation guidelines, tags tend to be noisy and sparse. Especially new resources annotated by only a few users have often rather idiosyncratic tags that do not reflect a common perspective useful for search. In this paper we introduce an approach based on Latent Dirichlet Allocation (LDA) for recommending tags of resources in order to improve search. Resources annotated by many users and thus equipped with a fairly stable and complete tag set are used to elicit latent topics to which new resources with only a few tags are mapped. Based on this, other tags belonging to a topic can be recommended for the new resource. Our evaluation shows that the approach achieves significantly better precision and recall than the use of association rules, suggested in previous work, and also recommends more specific tags. Moreover, extending resources with these recommended tags significantly improves search for new resources.

Categories and Subject Descriptors

E.1 [**Data**]: Data Structures—*Graphs and networks*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Clustering, Information filtering*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Language models*

General Terms

Algorithms, Experimentation, Measurement

Keywords

social bookmarking system, delicious, tag recommendation, tag search

RecSys'09, October 23–25, 2009, New York, New York, USA.

1. INTRODUCTION

Tagging systems [23] like Flickr[1], Last.fm[2] or Delicious[3] have become major infrastructures on the Web. These systems allow users to create and manage tags to annotate and categorize content. In *social* tagging systems like Delicious the user can not only annotate his own content but also content of others. The service offered by these systems is twofold: They allow users to publish content and to search for content. Thus *tagging* also serves two purposes for the user:

1. Tags help to organize and manage own content, and
2. Find relevant content shared by other users.

Tag recommendation can focus on one of the two aspects. Personalized tag recommendation helps individual users to annotate their content in order to manage and retrieve their own resources. Collective tag recommendation aims at making resources more visible to other users by recommending tags that facilitate browsing and search.

However, since tags are not restricted to a certain vocabulary, users can pick any tags they like to describe resources. Thus, these tags can be inconsistent and idiosyncratic, both due to users' personal terminology as well as due to the different purposes tags fulfill [15]. This reduces the usefulness of tags in particular for resources annotated by only a few users (aka cold start problem in tagging), whereas for popular resources collaborative tagging typically saturates at some point, i.e., the rate of new descriptive tags quickly decreases with the number of users annotating a resource [18].

The goal of the approach presented in this paper is to overcome the cold start problem for tagging new resources. To this end, we use Latent Dirichlet Allocation (LDA) to elicit latent topics from resources with a fairly stable and complete tag set to recommend topics for new resources with only a few tags. Based on this, other tags belonging to the recommended topics can be recommended. Compared to an approach using association rules, suggested previously for tag recommendation, our approach achieves significantly better precision and recall. Moreover, the recommended tags are more specific for a particular resource, and thus more useful for searching and recommending resources to other users [9].

The remainder of this paper is organized as follows. In Section 2, we define the problem of tag recommendation more formally, and introduce the two approaches based on

[1] `http://www.flickr.com`
[2] `http://www.lastfm.com`
[3] `http://delicious.com`

association rules and LDA. In Section 3 we present our evaluation results. In Section 4 we discuss related work, and in Section 5 we summarize and outline possible future research directions.

2. TAG RECOMMENDATION

To evaluate our approach using LDA for tag recommendation we compare our approach to association rules – a state-of-the-art method for tag recommendation proposed e.g. by Heymann et. al. [18]. After a formal problem description we introduce the two approaches in this Section.

2.1 Problem Definition

Given a set of resources R, tags T, and users U, the ternary relation $X \subseteq R \times T \times U$ represents the user specific assignment of tags to resources. A bookmark $b(r_i, u_j)$ for resource $r_i \in R$ and a user $u_j \in U$ comprises all tags assigned by u_j to r_i: $b(r_i, u_j) = \pi_t \sigma_{r_i, u_j} X$[4]. The goal of collective tag recommendation is to suggest new tags for a resource r_i with only a few bookmarks based on tag assignments to other resources collected in $Y = \sigma_{r \neq r_i} \pi_{r,t} X \subseteq R \times T$.

2.2 Association Rules

Association rules have been investigated in [18] for tag recommendation. They have the form $T_1 \rightarrow T_2$, where T_1 and T_2 are tagsets. The three key measures for association rules are support, confidence, and interest. Support is the (relative) number of resources that contain all tags of T_1 and T_2, i.e., an estimate of the joint probability $P(T_1, T_2)$. Confidence is an estimate of the conditional probability $P(T_2|T_1)$, i.e., how likely is T_2 given T_1. Interest (also called lift) is defined as the ratio between the common support for T_1 and T_2, and the individual support of T_1 and T_2 ($\frac{P(T_1,T_2)}{P(T_1)P(T_2)}$), and indicates whether T_1 and T_2 occur more often together than expected, if they were statistically independent. There exist efficient algorithms to exhaustively mine association rules with some minimum support from large datasets (e.g. [1]).

The basic idea of using association rules for tag recommendation is simple: If many resources with tags T_1 (high support) are typically also annotated with tags T_2 (high confidence), then a new resource with tags T_1 may also be meaningfully annotated with tags T_2. More formally, given the tagset from (a few) bookmarks $T = \bigcup b(r, u_i)$ for a resource r by users u_i, and an association rule $T_1 \rightarrow T_2$, all tags in T_2 are recommended, if $T_1 \subseteq T$.

Table 1 gives a selection of association rules with high confidence mined from our dataset. As also observed in [18] these rules cover all sorts of terminological relationships including spelling variants and synonyms (humour → humor; tools, utilities, utility → tool), loose notions of hypernyms (tutorial, resources → reference), and closely related terms (software, mac, apple → osx).

While the mined association rules are very intuitive, they typically recommend rather generic, frequent tags, such as "software" or "web". This is a direct consequence of requiring some minimum support for T_1 and T_2. Such generic tags are not necessarily useful for finding specific resources. Indeed, for personalized tag recommendation Xu et al. [31] explicitly penalize tag co-occurrences, when they have been annotated by different users.

[4]projection π and selection σ operate on multisets without removing duplicate tuples

Conf	Supp	Int	Rule
0.978	0.037	10.13	web, js → javascript
0.921	0.012	6.75	software, macintosh → mac
0.919	0.161	1.36	tools, fun, interesting → cool
0.915	0.086	4.05	web, weblogs → blogs
0.914	0.074	7.71	humour → humor
0.912	0.037	11.57	photography, photos → photo
0.910	0.136	3.81	howto, code, tutorials → tutorial
0.904	0.086	2.42	tools, utilities, utility → tool
0.904	0.111	2.55	tech, tutorial, tutorials → howto
0.902	0.049	1.71	toread, howto, guide → reference
0.902	0.111	1.56	cool, technology, computers → tech
0.902	0.222	2.80	cool, design, blogs → blog
0.900	0.172	1.21	cool, internet, free → web
0.900	0.123	1.38	webdesign, tips → web, design
0.900	0.062	5.40	web, development, web-design → html
0.900	0.124	3.07	design, css → webdev
0.900	0.074	2.13	web, osx → software
0.900	0.099	2.18	design, tutorials, css → development
0.900	0.025	6.75	software, mac, apple → osx
0.900	0.124	1.25	tutorial, resources → reference

Table 1: Selection of tag association rules with confidence $\geq$ 0.9

2.3 Latent Dirichlet Allocation

The general idea of Latent Dirichlet Allocation (LDA) is based on the hypothesis that a person writing a document has certain topics in mind. To write about a topic then means to pick a word with a certain probability from the pool of words of that topic. A whole document can then be represented as a mixture of different topics. When the author of a document is one person, these topics reflect the person's view of a document and her particular vocabulary. In the context of tagging systems where multiple users are annotating resources, the resulting topics reflect a collaborative shared view of the document and the tags of the topics reflect a common vocabulary to describe the document.

More generally, LDA helps to explain the similarity of data by grouping features of this data into unobserved sets. A mixture of these sets then constitutes the observable data. The method was first introduced by Blei et al. [10] and applied to solve various tasks including topic identification [16], entity resolution [7], and Web spam classification [8].

The modeling process of LDA can be described as finding a mixture of topics for each resource, i.e., $P(z \mid d)$, with each topic described by terms following another probability distribution, i.e., $P(t \mid z)$. This can be formalized as

$$P(t_i \mid d) = \sum_{j=1}^{Z} P(t_i \mid z_i = j) P(z_i = j \mid d), \qquad (1)$$

where $P(t_i \mid d)$ is the probability of the ith term for a given document d and z_i is the latent topic. $P(t_i \mid z_i = j)$ is the probability of t_i within topic j. $P(z_i = j \mid d)$ is the probability of picking a term from topic j in the document. The number of latent topics Z has to be defined in advance and allows to adjust the degree of specialization of the latent topics. LDA estimates the topic–term distribution $P(t \mid z)$ and the document–topic distribution $P(z \mid d)$ from an unlabeled corpus of documents using Dirichlet priors for the distributions and a fixed number of topics. Gibbs sampling [16] is

one possible approach to this end: It iterates multiple times over each term t_i in document d_i, and samples a new topic j for the term based on the probability $P(z_i = j|t_i, d_i, z_{-i})$ based on Equation 2, until the LDA model parameters converge.

$$P(z_i = j \mid t_i, d_i, z_{-i}) \propto \frac{C^{TZ}_{t_ij} + \beta}{\sum_t C^{TZ}_{tj} + T\beta} \frac{C^{DZ}_{d_ij} + \alpha}{\sum_z C^{DZ}_{d_iz} + Z\alpha} \quad (2)$$

C^{TZ} maintains a count of all topic–term assignments, C^{DZ} counts the document–topic assignments, z_{-i} represents all topic–term and document–topic assignments except the current assignment z_i for term t_i, and α and β are the (symmetric) hyperparameters for the Dirichlet priors, serving as smoothing parameters for the counts. Based on the counts the posterior probabilities in Equation 1 can be estimated as follows:

$$P(t_i \mid z_i = j) = \frac{C^{TZ}_{t_ij} + \beta}{\sum_t C^{TZ}_{tj} + T\beta} \quad (3)$$

$$P(z_i = j \mid d_i) = \frac{C^{DZ}_{d_ij} + \alpha}{\sum_z C^{DZ}_{d_iz} + Z\alpha} \quad (4)$$

2.3.1 Application to Tagging Systems

LDA assigns to each document latent topics together with a probability value that each topic contributes to the overall document. For tagging systems the documents are resources $r \in R$, and each resource is described by tags $t \in T$ assigned by users $u \in U$. Instead of documents composed of terms, we have resources composed of tags. To build an LDA model we need resources and associated tags previously assigned by users. For each resource r we need some bookmarks $b(r, u_i)$ assigned by users $u_i, i \in \{1 \dots n\}$. Then we can represent each resource in the system not with its actual tags but with the tags from topics discovered by LDA.

For a new resource r_{new} where we only have a small number of bookmarks ($i \in \{1 \dots 5\}$), i.e., only one to five users annotated this resource, we can expand the latent topic representation of this resource with the top tags of each latent topic. To accomodate the fact of some tags being added by multiple users whereas others are only added by one or two users we can use the probabilities that LDA assigns. As formalized in Equation 1 this is a two level process. Probabilities are assigned not only to the latent topics for a single resource but also to each tag within a latent topic to indicate the probability of this tag being part of that particular topic. We represent each resource r_i as the probabilities $P(z|r_i)$ for each latent topic $z_j \in Z$. Every topic z_j is represented as the probabilities $P(t|z_j)$ for each tag $t_n \in T$. By combining these two probabilities for each tag for r_{new}, we get a probability value for each tag that can be interpreted similarly as the relative tag frequency of a resource. Setting a threshold allows to adjust the number of recommended tags and emphasis can be shifted from recall to precision.

Imagine a resource with the following tags: "photo", "photography", and "howto". Table 2 shows the top terms for two topics related with the assigned tags. It is interesting to compare these two topics with the corresponding association rules in Table 1. Whereas the association rules indicate only fairly simple term expansions, the latent topics comprise an arguably broader notion of (digital) photography and the various aspects of tutorial material. Given these topics we can easily extend the current tag set or recommend new tags to users by looking at the latent topics. In our example, we can recommend "photos", "images", "photoshop", "tutorial", "reference", and "tips" if we set the threshold for the accumulated probabilities to 0.045 . LDA would assume that our resource in question belongs to 66% to the "photo"-topic and to 33% to the "howto"-topic. Multiplying these probabilities with the individual tag probabilities of the latent topics results in a ranked list of relevant tags for our resource.

Tag	Count	Prob.	Tag	Count	Prob.
photography	16452	0.235	howto	23371	0.219
photo	9002	0.129	tutorial	15519	0.145
photos	7739	0.110	reference	14084	0.132
images	6302	0.090	tips	13955	0.131
photoshop	4825	0.069	tutorials	7320	0.069
graphics	2831	0.040	guide	3430	0.032
image	2769	0.040	toread	2948	0.028
art	1910	0.027	article	2376	0.022
stock	1852	0.026	articles	1498	0.014
pictures	1676	0.024	useful	1442	0.013
design	1666	0.024	learning	1147	0.011
gallery	1386	0.020	tricks	1140	0.011
camera	831	0.012	how-to	1081	0.010
digital	802	0.011	help	1054	0.010

Table 2: Top terms composing the latent topics "photography" and "howto"

3. EVALUATION

To compare the two algorithms we evaluated both on a common dataset.

3.1 Dataset

As a dataset for our evaluations we use a crawl from Delicious provided by Hotho et. al. [19]. The dataset consists of ∼75,000 users, ∼500,000 tags and ∼3,200,000 resources connected via ∼17,000,000 tag assignments of users.

The overlap between tags, resources and users is very sparse. To get a dense subset of the data we computed p-cores [4] for different levels. For $p = 100$ we get enough bookmarks for each resource to split the data into meaningful training and test sets (90%:10%). The test sets differ in the number of bookmarks each resource has assigned to simulate new resources that only have one to five user annotations. For this we removed all tags not belonging to the first n bookmarks, $n \in \{1 \dots 5\}$.

Our final dataset consists of ∼10,000 resources, ∼10,000 users, and ∼3600 tags occuring in ∼3,200,000 tag assignments. We have five test sets containing 10% of the data. The 100-core ensures that each tag, each resource and each user appears at least 100 times in the tag assignments.

On this set, the only preprocessing of the tag assignments performed was the decapitalization of the tags. No stemming or other simplifications were applied. More sophisticated preprocessing might improve the results but would complicate the evaluation of the algorithms.

3.2 Results

In this Section we report results for our LDA-based algorithm and compare these with the numbers we get using association rules for the same task on the same dataset.

Conf	Prec	Rec	FM	Avg TFIDF	Median TFIDF
0.90	0.648	0.077	0.137	0.060	0.029
0.70	0.514	0.167	0.252	0.051	0.021
0.50	0.435	0.244	0.312	0.048	0.018
0.30	0.357	0.319	0.337	0.045	0.016
0.10	0.265	0.408	0.321	0.044	0.015

Table 3: Results for tag recommendation using association rules with different minimum confidences and 5 known bookmarks

#BM	Prec	Rec	FM	Avg TFIDF	Median TFIDF
1	0.741	0.041	0.077	0.054	0.030
2	0.691	0.056	0.104	0.057	0.030
3	0.682	0.066	0.120	0.059	0.029
4	0.663	0.072	0.130	0.060	0.029
5	0.648	0.077	0.137	0.060	0.029

Table 4: Results for tag recommendation using association rules with minimum confidence 0.9 for 1–5 known bookmarks

3.2.1 Association Rules

For mining association rules, we have used RapidMiner [24]. For the 9000 resources in the training set we get almost 550 K association rules with a minimum support of 0.05 and a minimum confidence of 0.1, many of which are of course partially redundant. Table 3 gives the results for 5 bookmarks, at different confidence levels (Conf). Precision (Prec), recall (Rec), f-measure (FM) are measured at macro level, i.e., they are averaged over the individual measures for each resource. The maximum precision (Prec) of 0.648 for confidence ≥ 0.9 is lower than the 0.873 reported in [18], who operated on a bigger dataset (about $60K$ resources, split into $50K$ training and $10K$ testing). Maximum f-measure is reached with association rules above the fairly low confidence of 0.3. The last two columns give the average and median TFIDF for correctly recommended tags. Both values lie in the same range as the corresponding values for the actual tags in the testset (0.054 and 0.018), which indicates that association rules tend to recommend rather general tags. In an attempt to recommend more specific tags, we have also experimented with a smaller support of 0.01. This however only increases recall at the cost of precision; the average and median specificity of recommended tags remains in the same range. For a smaller number of available bookmarks, precision goes up and recall goes down, and the f-measure slightly decreases. Average and median TFIDF remain essentially constant (see Table 4).

3.2.2 Latent Dirichlet Allocation

The tag recommendation algorithm is implemented in Java. We used LingPipe [2], to perform the Latent Dirichlet Allocation with Gibbs sampling. The LDA algorithm takes three input parameters: the number of terms to represent a latent topic, the number of latent topics to represent a document, and the overall number of latent topics to be identified in the given corpus. After some experiments with varying the first two parameters we fixed them at a value of 100.

As described in Section 2.3 we can set a threshold for the probabilities up to which we recommended tags. Table 5 shows precision, recall, f-measure (FM), as well as average TFIDF and median TFIDF of the "correctly" recommended tags. Not surprisingly, precision decreases when lowering the threshold whereas recall increases. We get a maximum f-measure at 0.001 of 0.401

Thresh	Prec	Rec	FM	Avg TFIDF	Median TFIDF
0.01	0.717	0.174	0.281	0.169	0.079
0.005	0.609	0.245	0.349	0.140	0.057
0.001	0.370	0.439	0.401	0.096	0.031
0.0005	0.291	0.527	0.375	0.085	0.026
0.00001	0.168	0.669	0.269	0.071	0.022

Table 5: Results for tag recommendation using LDA with 100 topics with different thresholds to recommend a tag for 5 known bookmarks

#BM	Prec	Rec	FM	Avg TFIDF	Median TFIDF
1	0.680	0.069	0.126	0.233	0.128
2	0.717	0.112	0.193	0.199	0.097
3	0.712	0.139	0.233	0.186	0.089
4	0.711	0.160	0.261	0.174	0.084
5	0.717	0.174	0.281	0.169	0.079

Table 6: Results for tag recommendation using LDA with 100 topics and threshold 0.01 for 1–5 known bookmarks

Table 6 gives detailed results for different numbers of known bookmarks using a threshold of 0.01 to recommend with high precision. Knowing more bookmarks in advance for a resource does not increase precision (2 bookmarks $\rightarrow$ 0.717; 5 bookmarks $\rightarrow$ 0.717) but increases recall significantly. The average TFIDF gives the expected value for the specificity of a tag whereas the median gives the typical specificity. Because the TFIDF values show a power law distribution, the average is of course larger than the median. Both values are significantly higher for tags recommended by LDA than by association rules, but also higher than the average and median TFIDF of the actual tags present in our tag set. As can be seen in Table 4 and Table 6 the TFIDF values are two to four times higher. Recommending resource specific tags with high TFIDF is particularly useful for search as pointed out in [9], fairly infrequent tags are usually used for topical and type annotations.

The results for varying the number of latent topics are shown in Table 7. The f-measure is shown for 50, 100, 250, and 500 latent topics. The number of bookmarks (#BM) indicates the number of users that have annotated a resource in the test set. The threshold for our recommendation was set to 0.001. As can be seen in the table, performance decreases with the LDA topic size for the 1 BM case. This effect is reversed when adding more bookmarks. A small number of topics typically leads to fairly general topics that are mixtures of more specific subtopics. Such general topics have a higher chance to be evoked by one of the few tags in one bookmark, leading to a higher recall. With more bookmarks, there are more tags, and it is more beneficial to separate the general topics into more specific topics. 100 LDA topics give the best average results.

Real Tag	TF	TFIDF	LDA Tag	LDA Prob.	AR Tag	AR Conf.
science	0.0906	0.2281	***del.icio.us***	0.1001	***web***	0.912
bookmarks	0.0695	0.1721	***delicious***	0.0478	***reference***	0.760
tags	0.0546	0.1468	***tools***	0.0356	***tools***	0.664
reference	0.0521	0.0407	business	0.0223	***internet***	0.657
social	0.0509	0.1068	***language***	0.0204	cool	0.642
folksonomy	0.0447	0.1306	***bookmarks***	0.0166	tech	0.585
del.icio.us	0.0409	0.1166	***web***	0.0090	***software***	0.541
tools	0.0397	0.0271	***tool***	0.0090	toread	0.515
tagging	0.0360	0.1062	***science***	0.0085	technology	0.467
research	0.0347	0.0714	space	0.0065	interesting	0.417
delicious	0.0248	0.0722	dictionary	0.0064	design	0.398
bookmark	0.0236	0.0770	***bookmark***	0.0059	information	0.395
academic	0.0223	0.0780	english	0.0049	***search***	0.393
search	0.0223	0.0402	environment	0.0040	***blog***	0.391
web	0.0186	0.0084	***reference***	0.0039	–	–
bookmarking	0.0173	0.0717	astronomy	0.0037	–	–
tag	0.0161	0.0496	marketing	0.0033	–	–
socialsoftware	0.0149	0.0387	***tags***	0.0033	–	–
internet	0.0124	0.0124	cool	0.0032	–	–
academia	0.0111	0.0780	startup	0.0031	–	–
collaboration	0.0111	0.0322	words	0.0028	–	–

Table 8: Actual tags with tag frequency and recommended tags with computed probablity for URL www.connotea.org

#BM	# LDA topics			
	50	100	250	500
1	0.313	0.310	0.297	0.268
2	0.353	0.360	0.351	0.328
3	0.367	0.381	0.378	0.356
4	0.371	0.392	0.397	0.386
5	0.378	0.401	0.414	0.403

Table 7: F-measure for different sized test set and different number of LDA topics (threshold 0.001)

Table 8 shows the actual tag distribution for a randomly selected resource (`http://www.connotea.org`), the top tags recommended by LDA with aggregated probabilities, and (all) the tags recommended by association rules based on five known bookmarks. The tags available in the known bookmarks (first column)[5] and the correctly recommended tags (forth and sixth column) are marked in bold. As the actual tags indicate, Connotea is a tagging site focusing on scientists and scientific resources. The tags recommended by LDA come from five latent topics, comprising social systems, tagging, science, business, and language. These tags characterize Connotea quite well, and accordingly among the nine most likely recommended tags, there is only one rather general tag (business) that is not among the actual tags. In contrast, the tags recommended by association rules hardly characterize the site, but are rather non descriptive and general.

3.2.3 Tag Search

To evaluate the effectiveness of our recommended tags for tag search we compared three result lists: The first is based on the testset with only $1-5$ bookmarks per resource, the second uses the tags recommended by our algorithm. These two lists are compared with the list based on all original tags assigned to the test set. For the ranking of the results in each list, we implemented a simple baseline algorithm based on single keyword search. The resources are weighted according to the TFIDF score of the query tag. E.g. a search for the keyword "web" gives a list with resources annotated with the tag "web". The list is ranked according to tag frequency, i.e., how high is the number of "web"-tags compared to the overall number of tags assigned to a resource.

The testset without recommended tags is also ranked by TFIDF, whereas the testset with recommended tags is ranked by the probability assigned by LDA. To compare the three ranked lists, we need to first decide which of the baseline results are considered relevant. We report scores for taking the top 10 and the top 20 resources as relevant results. A well known measure for comparing rankings in information retrieval is Mean Average precision (MAP) [22], computed as follows:

$$MAP(Q) = \frac{1}{|Q|}\sum_{j=1}^{|Q|}\frac{1}{m_j}\sum_{k=1}^{m_j}\text{Precision}(R_{jk}) \qquad (5)$$

with R_{jk} the set of ranked results from the top of the list down to item k in the list, where the set of relevant items is $\{i_1 \dots i_{m_j}\}$. If no relevant document is retrieved, precision is taken to be 0.

Table 9 shows the MAP values based on the number of known bookmarks. When considering the top 10 TFIDF ranked results as relevant, extension of the resources with our recommended tags increases MAP by more than 300% for one known bookmark. When considering the top 20 results of the baseline algorithm as relevant, the MAP score for the LDA probabilities weighted ranked list increases by more then 400% in the one bookmark case.

[5]The first five bookmarks contain three more tags with rather low TF: webware, management, and social_software.

#BM	MAP for top 10	
	w/o Extended	w/ Extended
1	0.037	0.137
2	0.058	0.196
3	0.075	0.221
4	0.091	0.241
5	0.105	0.256
#BM	MAP for top 20	
	w/o Extended	w/ Extended
1	0.025	0.105
2	0.039	0.150
3	0.051	0.170
4	0.062	0.186
5	0.072	0.198

Table 9: Mean Average Precision (MAP) for tag search with and without extention of recommended tags

4. RELATED WORK

Tag recommendation has received considerable interest in recent years. Most work has focused on personalized tag recommendation, suggesting tags to the user bookmarking a new resource: This is often done using collaborative filtering, taking into account similarities between users, resources, and tags. [25] introduces an approach to recommend tags for weblogs, based on similar weblogs tagged by the same user. Chirita et al. [11] realize this idea for the personal desktop, recommending tags for web resources by retrieving and ranking tags from similar documents on the desktop. [31] aims at recommending a few descriptive tags to users by rewarding co-occuring tags that have been assigned by the same user, penalizing co-occuring tags that have been assigned by different users, and boosting tags with high descriptiveness (TFIDF). As pointed out in Section 2.2, penalizing co-occuring tags assigned by different users in an effort to recommend *personalized* tags is in contrast to using tag association rules to recommend *general* tags to improve recall for search. Sigurbjörnsson and van Zwol [28] also look at co-occurence of tags to recommend tags based on a user defined set of tags. The co-occuring tags are then ranked and promoted based on e.g. descriptiveness. Jaeschke et al. [20] compare two variants of collaborative filtering and Folkrank, a graph based algorithm for personalized tag recommendation. For collaborative filtering, once the similarity between users on tags, and once the similarity between users on resources is used for recommendation. Folkrank uses random walk techniques on the user-resource-tag (URT) graph based on the idea that popular users, resources, and tags can reinforce each other. These algorithms take co-occurrence of tags into account only indirectly, via the URT graph. Symeonidis et al. [30] employ dimensionality reduction to personalized tag recommendation. Whereas [20] operate on the URT graph directly, [30] use generalized techniques of SVD (Singular Value Decomposition) for n-dimensional tensors. The 3 dimensional tensor corresponding to the URT graph is unfolded into 3 matrices, which are reduced by means of SVD individually, and combined again to arrive at a more dense URT tensor approximating the original graph. Tag recommendation then suggests tags to users, if their weight is above some threshold. An interactive approach is presented in [14]. After the user enters a tag for a new resource, the algorithm recommends tags based on co-occurence of tags for resources which the user or others used together in the past. After each tag the user assigns or selects, the set is narrowed down to make the tags more specific. In [27], Shepitsen et al. propose a resource recommendation system based on hierarchical clustering of the tag space. The recommended resources are identified using user profiles and tag clusters to personalize the recommendation results. Using LDA topic models to recommend resources rather than tags is subject for future work.

An approach to *collective* tag recommendation based on association rule mined from the resource tag matrix has been introduced in [18]. As discussed in Section 3.2, this approach recommends rather general tags with low TFIDF, and achieves smaller recall and precision than the approach based on LDA introduced in this paper. When content of resources is available, tag recommendation can also be approached as a classification problem, predicting tags from content. A recent approach in this direction is presented in [29]. They cluster the document-term-tag matrix after an approximate dimensionality reduction, and obtain a ranked membership of tags to clusters. Tags for new resources are recommended by classifying the resources into clusters, and ranking the cluster tags accordingly.

Tags have been proven to be very useful for search: in case of image search where content based features are very difficult to extract [12], in case of enterprise search where not enough link information is available [13], or in case of web search to optimize results [3]. A large scale evaluation of Delicious regarding search is presented in [17]. They found that 50% of the pages annotated by a particular tag contain the tag within the page's content. Bischoff et al. [9] provide an indepth analysis of a number of tagging systems with respect to to their usefulness for search. They observe that descriptive tags such as topic or type tags are much more frequent than personal tags such as "to read", especially in the mid and low tag frequency range, and that these tags are indeed used in search. Berendt and Hanser [6] argue that tags can be considered content and not just metadata which makes them valuable in a content based document retrieval scenario as well.

Recently a number of papers deal with imroving search in tagging systems. Krestel and Chen [21] propose a method to measure the quality of tags with respect to the annotated resource to identify high quality tags that describe a resource better than others. Hotho et al. [19] propose exploiting co-ocurrence of users, resources, and tags for searching and ranking within tagging systems. "FolkRank" is using a graph model to represent the *folksonomy* and can be used to rank classical keyword search results. In [5], Begelman et al. present a tag clustering algorithm to improve search. The setting is similar to ours: Related tags are identified that can be used for extending existing resource annotations, query expansion or result clustering. The clustering is based on simple co-occurence counts. Unfortunately, the paper does not contain a sound evaluation of the results. Schenkel et al. [26] propose to improve search in tagging systems by expanding a user query with semantically similar tags and rank the result additionally based on a social component, which means that tagging information of friends of a user in the network is taken into account when a user submits a query.

5. CONCLUSIONS AND FUTURE WORK

In this paper we have investigated the use of Latent Dirichlet Allocation for collective tag recommendation. Compared to association rules, LDA achieves better accuracy, and in particular recommends more specific tags, which are more useful for search. In general, our LDA-based approach is able to elicit a shared topical structure from the collaborative tagging effort of multiple users, whereas association rules are more focused on simple terminology expansion. However, both approaches succeed to some degree in overcoming the idiosyncracies of individual tagging practices.

For future work we are interested to see whether it is beneficial to combine association rules and LDA. As we showed in Section 3.2 the tags that are recommended by both algorithms differ significantly from each other. Our hypothesis is that accuracy can be improved by combining the more general tags recommended by association rules with the more specific tags recommended by LDA. Along similar lines, we also plan to investigate combining language models derived from the actual tags annotated to a resource with the latent topic models.

The main contribution of latent topic models is to reduce sparsity of the tag space. This gives rise to several interesting lines of research we will investigate: Mapping resources to their latent topics may result in more robust resource recommendation. Eliciting latent topics from the tagging practices of individual users and combining them with the latent topics for resources is a promising direction for personalized tag recommendation. Finally, we will experiment with using the probability of tags derived from topic models for visualizing tag recommendations in the form of tag clouds.

Regarding data sets, we also want to experiment with datasets from different domains, to check whether photo, video, or music tagging sites show different system behavior influencing our algorithms.

6. ACKNOWLEDGMENTS

This work was supported in part by the EU project IST 45035 - Platform for searcH of Audiovisual Resources across Online Spaces (PHAROS).

7. REFERENCES

[1] R. Agrawal, T. Imielinski, and S. A. Mining association rules between sets of items in large databases. *SIGMOD Record*, 22(2), 1993.

[2] Alias-i. Lingpipe 3.7.0. `http://alias-i.com/lingpipe(accessed:10/2008)`, 2008.

[3] S. Bao, G.-R. Xue, X. Wu, Y. Yu, B. Fei, and Z. Su. Optimizing web search using social annotations. In C. L. Williamson, M. E. Zurko, P. F. Patel-Schneider, and P. J. Shenoy, editors, *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, pages 501–510, New York, NY, USA, 2007. ACM.

[4] V. Batagelj and M. Zaversnik. Generalized cores. *CoRR*, cs.DS/0202039, 2002.

[5] G. Begelman, P. Keller, and F. Smadja. Automated tag clustering: Improving search and exploration in the tag space. In *Proceedings of the WWW 2006 Workshop on Collaborative Web Tagging*, Edinburgh, May 2006.

[6] B. Berendt and C. Hanser. Tags are not metadata, but just more content - to some people. In *Proceedings of the International Conference on Weblogs and Social Media*, 2007.

[7] I. Bhattacharya and L. Getoor. A latent dirichlet model for unsupervised entity resolution. In *SIAM Conference on Data Mining (SDM)*, pages 47–58, April 2006.

[8] I. Bíró, D. Siklósi, J. Szabó, and A. A. Benczúr. Linked latent dirichlet allocation in web spam filtering. In *AIRWeb '09: Proceedings of the 5th International Workshop on Adversarial Information Retrieval on the Web*, pages 37–40, New York, NY, USA, 2009. ACM.

[9] K. Bischoff, C. S. Firan, W. Nejdl, and R. Paiu. Can all tags be used for search? In *CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management*, pages 193–202, New York, NY, USA, 2008. ACM.

[10] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, January 2003.

[11] P. A. Chirita, S. Costache, W. Nejdl, and S. Handschuh. P-tag: large scale automatic generation of personalized annotation tags for the web. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 845–854, New York, NY, USA, 2007. ACM.

[12] R. Datta, W. Ge, J. Li, and J. Wang. Toward bridging the annotation-retrieval gap in image search. *Multimedia, IEEE*, 14(3):24–35, July-Sept. 2007.

[13] P. A. Dmitriev, N. Eiron, M. Fontoura, and E. J. Shekita. Using annotations in enterprise search. In L. Carr, D. D. Roure, A. Iyengar, C. A. Goble, and M. Dahlin, editors, *Proceedings of the 15th international conference on World Wide Web, WWW 2006, Edinburgh, Scotland, UK, May 23-26, 2006*, pages 811–817, New York, NY, USA, 2006. ACM.

[14] N. Garg and I. Weber. Personalized, interactive tag recommendation for flickr. In *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pages 67–74, New York, NY, USA, 2008. ACM.

[15] S. Golder and B. A. Huberman. Usage patterns of collaborative tagging systems. *Journal of Information Science*, 32(2):198–208, April 2006.

[16] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proc Natl Acad Sci U S A*, 101 Suppl 1:5228–5235, April 2004.

[17] P. Heymann, G. Koutrika, and H. Garcia-Molina. Can social bookmarking improve web search? In M. Najork, A. Z. Broder, and S. Chakrabarti, editors, *Proceedings of the International Conference on Web Search and Web Data Mining, WSDM 2008, Palo Alto, California, USA, February 11-12, 2008*, pages 195–206. ACM, 2008.

[18] P. Heymann, D. Ramage, and H. Garcia-Molina. Social tag prediction. In *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 531–538, New York, NY, USA, 2008. ACM.

[19] A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme. Information retrieval in folksonomies: Search and ranking. In Y. Sure and J. Domingue, editors, *The Semantic Web: Research and Applications*, volume 4011 of *Lecture Notes in Computer Science*, pages 411–426, Heidelberg, Germany, June 2006. Springer.

[20] R. Jäschke, L. B. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme. Tag recommendations in folksonomies. In J. N. Kok, J. Koronacki, R. L. de Mántaras, S. Matwin, D. Mladenic, and A. Skowron, editors, *Knowledge Discovery in Databases: PKDD 2007, 11th European Conference on Principles and Practice of Knowledge Discovery in Databases, Warsaw, Poland, September 17-21, 2007, Proceedings*, volume 4702 of *Lecture Notes in Computer Science*, pages 506–514, Heidelberg, Germany, 2007. Springer.

[21] R. Krestel and L. Chen. The art of tagging: Measuring the quality of tags. In J. Domingue and C. Anutariya, editors, *The Semantic Web, 3rd Asian Semantic Web Conference, ASWC 2008, Bangkok, Thailand, December 8-11, 2008. Proceedings*, volume 5367 of *Lecture Notes in Computer Science*, pages 257–271, Heidelberg, Germany, 2008. Springer.

[22] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, UK, July 2008.

[23] C. Marlow, M. Naaman, D. Boyd, and M. Davis. Ht06, tagging paper, taxonomy, flickr, academic article, to read. In U. K. Wiil, P. J. Nürnberg, and J. Rubart, editors, *HYPERTEXT 2006, Proceedings of the 17th ACM Conference on Hypertext and Hypermedia, August 22-25, 2006, Odense, Denmark*, pages 31–40, New York, NY, USA, 2006. ACM.

[24] I. Mierswa, M. Wurst, R. Klinkenberg, M. Scholz, and T. Euler. Yale: Rapid prototyping for complex data mining tasks. In L. Ungar, M. Craven, D. Gunopulos, and T. Eliassi-Rad, editors, *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 935–940, New York, NY, USA, August 2006. ACM.

[25] G. Mishne. Autotag: a collaborative approach to automated tag assignment for weblog posts. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 953–954, New York, NY, USA, 2006. ACM.

[26] R. Schenkel, T. Crecelius, M. Kacimi, S. Michel, T. Neumann, J. X. Parreira, and G. Weikum. Efficient top-k querying over social-tagging networks. In *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 523–530, New York, NY, USA, 2008. ACM.

[27] A. Shepitsen, J. Gemmell, B. Mobasher, and R. D. Burke. Personalized recommendation in social tagging systems using hierarchical clustering. In P. Pu, D. G. Bridge, B. Mobasher, and F. Ricci, editors, *Proceedings of the 2008 ACM Conference on Recommender Systems, RecSys 2008, Lausanne, Switzerland, October 23-25, 2008*, pages 259–266, New York, NY, USA, 2008. ACM.

[28] B. Sigurbjörnsson and R. van Zwol. Flickr tag recommendation based on collective knowledge. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 327–336, New York, NY, USA, 2008. ACM.

[29] Y. Song, Z. Zhuang, H. Li, Q. Zhao, J. Li, W.-C. Lee, and C. L. Giles. Real-time automatic tag recommendation. In *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 515–522, New York, NY, USA, 2008. ACM.

[30] P. Symeonidis, A. Nanopoulos, and Y. Manolopoulos. Tag recommendations based on tensor dimensionality reduction. In *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pages 43–50, New York, NY, USA, 2008. ACM.

[31] Z. Xu, Y. Fu, J. Mao, and D. Su. Towards the semantic web: Collaborative tag suggestions. In *Proceedings of Collaborative Web Tagging Workshop at 15th International World Wide Web Conference*, 2006.