

Context Determines Content - An Approach to Resource Recommendation in Folksonomies

Thomas Rodenhausen
Multimedia Communications
Lab
Technische Universität
Darmstadt, Germany
thomas.rodenhausen@
kom.tu-darmstadt.de

Mojisola Anjorin
Multimedia Communications
Lab
Technische Universität
Darmstadt, Germany
mojisola.anjorin@
kom.tu-darmstadt.de

Renato Domínguez
García
Multimedia Communications
Lab
Technische Universität
Darmstadt, Germany
renato@kom.tu-
darmstadt.de

Christoph Rensing
Multimedia Communications
Lab
Technische Universität
Darmstadt, Germany
rensing@kom.tu-
darmstadt.de

ABSTRACT

By means of tagging in social bookmarking applications, so called folksonomies emerge collaboratively. Folksonomies have shown to contain information that is beneficial for resource recommendation. However, as folksonomies are not designed to support recommendation tasks, there are drawbacks of the various recommendation techniques. Graph-based recommendation in folksonomies for example suffers from the problem of concept drift. Vector space based recommendation approaches in folksonomies suffer from sparseness of available data. In this paper, we propose the flexible framework VSScore which incorporates context-specific information into the recommendation process to tackle these issues. Additionally, as an alternative to the evaluation methodology LeavePostOut we propose an adaptation LeaveRTOut for resource recommendation in folksonomies. In a subset of resource recommendation tasks evaluated, the proposed recommendation framework VSScore performs significantly more effectively than the baseline algorithm FolkRank.

Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—*Information filtering, Retrieval models, Search process*; H.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing—*Linguistic processing*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

4th ACM RecSys Workshop on Recommender Systems and the Social Web 2012 Dublin, Ireland
Copyright 20XX ACM X-XXXXX-XX-X/XX/XX ...\$15.00.

General Terms

Algorithms, Human Factors, Experimentation, Measurement

Keywords

Recommender Systems, Context, Social Media, Collaborative Tagging, Folksonomy, Ranking, Vector Space

1. INTRODUCTION

The assignment of tags by users to resources forms what is called a folksonomy. For this work, the formalized definition of a folksonomy adapted from [12] is given in Definition 1.

Definition 1. A folksonomy is a tuple $F = \{U, R, T, Y\}$ where U , R and T are finite sets, whose elements are called users, resources and tags respectively. Y is the ternary tag assignment relation between them, $Y \subseteq U \times R \times T$. $E = U \cup R \cup T$ is the finite set of the so called entities in the folksonomy.

The information shared with other users by means of tagging can help to retrieve resources via search, navigation, or to give an overview about their content. With a high number of users, folksonomy applications allow to deliver relevant and authoritative results in these tasks. This is where recommendation algorithms can be of benefit to users. Resource recommendation algorithms usually rank resources according to a certain criterion, which often times, and also for this work is relevance towards an information need.

Resource ranking in folksonomies is however not a trivial task as folksonomies are not designed for search [16]. The information about the resources to rank may be sparse e.g. because the content of a multimedia resource is unknown and hence semantic information missing. Additionally, the criterion to rank may be dependent on the user or the user's context. It is therefore necessary to utilize such information in order to create a high-quality ranking and hence recommendation of resources.

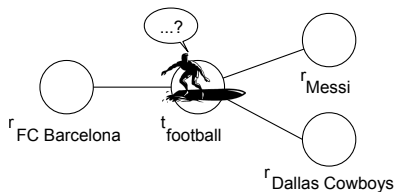


Figure 1: As the probabilistically traversing surfer arrives from a resource about the soccer club FC Barcelona he actually intends to visit nodes related to the semantic concept of soccer. However, he may drift off to nodes related to the semantic concept of american football, as the tag *football* is ambiguous and connected to a resource about the Dallas Cowboys, which is a popular american football team.

2. CONCEPT AND REALIZATION

In this section VSScore is described. We hypothesize that context-specific information can be leveraged to overcome drawbacks of existing resource recommendation algorithms in folksonomies.

2.1 Challenges of Recommendation in Folksonomies

The nature of folksonomies poses challenges to recommendation tasks. Users may suffer from the cold-start problem, which means they might not yet have built up a profile of tags and resources, which describes their interests well. Tags are usually very sparse compared to the resources' content. Hence extensive information about resources, as for example used in traditional web search, might not be (readily) available. Moreover, tags may be ambiguous or describe users' opinions instead of resources' content. These reasons can cause concept drift making the recommendation of resources misleading.

Graph-based recommendation techniques have been frequently applied to folksonomies [2, 3, 12, 17]. In order to measure relevance, graph-based approaches make Assumption 1 about the folksonomy's content and structure, which is extended from [2]. Other recommendation algorithms and technologies in folksonomies, which limit their information sources to information stored in folksonomies, often times make similar assumptions, e.g. [1].

Assumption 1.

- (i) Tags of a resource describe the resource's content well.
- (ii) Resources of a tag describe the tag's semantic well.
- (iii) Tags of a user describe the user's interests well.
- (iv) Users of a tag describe the tag's semantic well.
- (v) Resources of a user describe the user's interests well.
- (vi) Users of a resource describe the resource's content well.

However, as described, these assumptions are easily contradicted by the characteristics (inherent nature) of a folksonomy. Graph-based approaches are frequently based on surfer models which probabilistically traverse a graphical representation of the folksonomy. Resources are represented as nodes of a graph and the more often they are visited by the surfer, the higher they are ranked for recommendation. As can be seen in Figure 1, the ambiguity of the tag *football* introduces concept drift which harms the measurement of relevance.

Hence it is necessary to research recommendation algorithms that are more robust to the described challenges. In

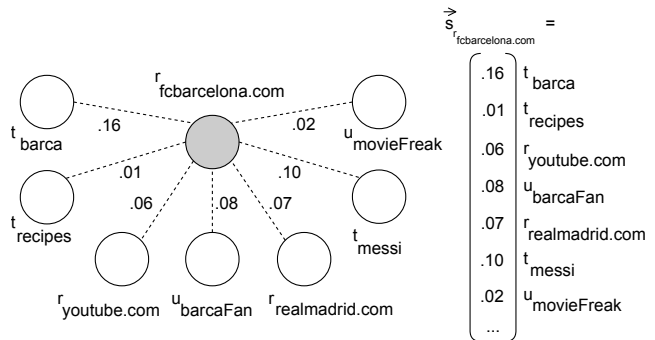


Figure 2: Example of the context of a resource in a folksonomy. The relationship between the resource about the soccer club FC Barcelona and other entities of the folksonomy are differently strong depending on the semantic concepts they represent. The context can be represented by a vector. Note: No direct connection via a tag assignment between the entities is necessary to determine their relationship strength.

this paper, we hypothesize that by incorporating context-specific information into the recommendation process a more robust resource recommendation can be achieved

2.2 Context-specific Information

There has been extensive research on leveraging sources of context-specific information additionally to information stored in folksonomies, e.g. in [2, 3, 8]. However, in this work the focus is on context-specific information inherently available in folksonomies which can thus be leveraged without external knowledge sources.

In a folksonomy, the entities are only described by their name or in the case of resources, e.g. by a URL. Hence, in isolation, they are usually hard to characterize. The interests of a user, the content of resources, or the semantic of a tag are generally unknown. However, Assumption 2 can be made, especially in large and dense folksonomies:

Assumption 2. The context of an entity describes the entity well.

In this work, the context of an entity $e \in E$ of a folksonomy is modeled as follows and illustrated in Figure 2. The context of e is given by the strength of relation between e and other entities in the folksonomy. These relation strengths can be given by a set of scored entities. The set of scored entities can for example be computed by a ranking algorithm, as a ranking describes the relation strength between the ranked entities and the query entity well. In the case of a graph-based ranking algorithm, the scores can be given by the visit probabilities to entities' nodes. Hence, the set of scored entities can be viewed as a vector \vec{s}_e in a vector space, where each entity of the folksonomy represents a single dimension of this space. Algorithm 2.1¹ formalizes the modeling of an entity's context in this work.

Pragmatics is the level of linguistic knowledge concerned with "the study of the relation between language and context-of-use" [14]. E.g. the utterance "FC Barcelona played a stunning game of football" is used in the context of a soccer game. Moreover, context can be used to determine the

¹Might be adapted to compute a context of a set of entities

Algorithm 2.1 Entity’s context computation

Input: Folksonomy $F = (U, R, T, Y)$, Entity $e \in E$

- 1: **procedure** CREATEENTITYCONTEXT(F, e)
- 2: $Q = \{e\}$ \triangleright Use entity as query entity
- 3: $\vec{s}_e = \text{SCORE}(F, Q)$ \triangleright Create scoring vector for query entity

Output: Scoring vector \vec{s}_e
 \triangleright Scoring vector models entity’s context

semantic concept of a term [14]. Hence, the context of the utterance above allows to identify the semantic concept of the term *football* as the semantic concept of soccer due to its appearance with *FC Barcelona*, which is a well known soccer club.

Tags in a folksonomy can be viewed as representations of a semantic concept a user associates with the tag [7], e.g. a user may associate the semantic concept of soccer with the tag *football*. In this paper, we hypothesize that context can not only be used to determine the semantic concept of a tag, but additionally to determine the semantic concept of resources and users. Because the entities of a folksonomy can be described by their context, this representation can uniformly be leveraged as representations of their semantic concepts. A resource may for example be a representation of the semantic concept of american football, e.g. the Wikipedia article on american football. A user may for example be a representation of the semantic concept of a hobby soccer player. Hence in Figure 2, not only a strength of the relation between $r_{FCBarcelona}$ and other entities of the folksonomy is given, but the strength of relation between the semantic concept represented by $r_{FCBarcelona}$ and semantic concepts represented by other entities in the folksonomy.

Note, that this is a generic definition of an entity’s context. $\text{SCORE}(Q)$ is implemented by a so called *scoring algorithm* (in this paper FolkRank [12] is chosen as scoring algorithm). A scoring algorithm however does not necessarily have to be limited to information contained in a folksonomy, e.g. a friendship relation between users may additionally be leveraged. Neither does it have to include all the available information, e.g. users may not be part of the scored entities. The modeling of the context hence depends on the implementation of $\text{SCORE}(Q)$. However, it should be such that Assumption 3 is reasonably justified.

Assumption 3. A vector \vec{s}_e , created with a scoring algorithm for a single query entity e , describes the relationship between e and other entities (in the folksonomy) well.

2.3 VSScore

VSScore, which is short for vector space score, is a framework based on the vector space model [15], which is well-known in text retrieval. There, the terms of a text usually represent semantic concepts. VSScore ports the intuition of the vector space model to folksonomies. In the vector space model, queries and resources are represented as vectors, and, for scoring, their distance is measured in a vector space. Therefor the content of resources has to be known. In folksonomies, however, this content is mostly unknown. Additionally, a query entity does not necessarily have to be textual, e.g. it could be a user entity. However, as described above, the context of an entity in the folksonomy can be used to describe the entity well. Therefore, VSScore

makes use of Algorithm 2.1 to establish a representation of an entity’s context. As both the query entities and resources are described extensively by their context, VSScore may alleviate concept drift and improve the relevance measure of e.g. its underlying implementation of $\text{SCORE}(Q)$. As a vector representation of the set of semantic concepts of query entities and resources can be computed, it can for recommendation be proceeded as for ranking with the vector space model. In text retrieval, a standard measure to quantify query-resource-similarity is cosine-similarity of the respective vector representations [15]. As the dimensions of the vector space in text retrieval usually represent semantic concepts, and we regard all entities of the folksonomy to be representations of semantic concepts, VSScore may for example employ the cosine-similarity (which is chosen to be used in this paper) to calculate the similarity between vectors:

$$\text{cosine}(\vec{s}_q, \vec{s}_r) = \frac{\vec{s}_q \cdot \vec{s}_r}{\|\vec{s}_q\| \cdot \|\vec{s}_r\|} \quad (1)$$

Note, that the computation of the vector representation of the semantic concepts of entities may be of different kinds. One possibility is the computation described in Algorithm 2.1. However, e.g. in case the set of query entities is limited to tags, it may also be realized such that it returns a vector representation of the semantic concept of the query tag in a reference corpus, e.g. Wikipedia. In this case, the vector representations of query entities and resource need to be reduced to their common space.

3. EVALUATION

In this paper, we evaluate the following resource recommendation scenarios. Assume an e-Learning platform in which users share resources and tags, such as CROKODIL². In a *user-based recommendation scenario* a list of resources relevant to a user are to be recommended once the user signs in. Additionally, in a *tag-based resource recommendation scenario*, a list of resources relevant to a tag are to be recommended once the user shows interest in a tag. The resource ranking tasks for these recommendation scenarios are defined as follows, adapted from [5]: *Interests match* uses a user and *guided search* a tag as query entity.

The resource recommendation approaches evaluated are FolkRank, VSScore and Popularity. The popularity of a resource is simply computed by the number of tags and users of a resource. A comparison to other resource recommendation algorithms that leverage context-specific information is not possible. GFolkRank and GRank [2], Category-based FolkRank and Area-based FolkRank [3] and the contextualization approach by Cantador et al. [8] all require folksonomy-external context-specific information. Furthermore, the ContextWalk approach presented in [6] by Bogers is specific for a movie recommendation scenario. When transferred to a folksonomy recommendation scenario, the difference to other graph-based approaches like FolkRank becomes very limited. To the knowledge of the authors, this is the first work that leverages context-specific information inherently available in a folksonomy for resource recommendation in folksonomies.

3.1 Evaluation Methodology

²<http://crokodil.de/>, retrieved 07/06/12

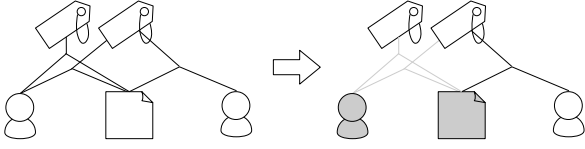


Figure 3: The LeavePostOut methodology eliminates all tag assignments between a user and a resource. No information remains in the folksonomy that directly connects the user with the resource.

LeavePostOut is introduced in [13] for the task of tag recommendation in folksonomies.

$$P_{u,r} = \{(u, r, t) | (u, r, t) \in Y\} \quad (2)$$

$$P = \bigcup_{\forall u,r} P_{u,r} \quad (3)$$

A post $P_{u,r}$ is composed of all tag assignments made by user u to resource r and P is the set of all posts independent of user or resource. *LeavePostOut* removes one post at a time from the folksonomy. A subset of the entities of the post are used as query entities to create a ranking. In [13], the recommendation algorithm comes up with the tags t that appear in tag assignments of $P_{u,r}$ given u and r as query entities. To use the methodology for the task of resource ranking, resource r of $P_{u,r}$ has to be ranked at the top or as high as possible. Therewith the Assumption 4 is made:

Assumption 4. The assignment of a tag by a user to a resource indicates relevance of the resource towards the information need represented by the assigned tag, and represented by the user.

The historical information, however, indicates relevance of a resource towards a tag and user at a point in the past. It is, thus, only a proxy measure, an estimation of the true relevance a resource may actually have at the time the ranking is actually used, e.g. what is relevant for a user may change over time.

Additionally, as the task of resource ranking requires the assessment of relevance of each resource in the ranking, but with *LeavePostOut* the ranking’s quality can only be assessed with regard to the relevance of resource r , the assessment of the ranking’s overall quality is limited. E.g. given a ranking, it is only known that r is of relevance towards the information need. However, this does not mean that none of the other highly ranked resources are not of relevance. This problem is described as the incompleteness problem in [9].

The key observation of *LeavePostOut* is, that after the removal of $P_{u,r}$, there is no information left in the folksonomy anymore, that connects the user u of $P_{u,r}$ directly with resource r . However, there still remains the possibility, that information in the folksonomy exists that connects resource r or user u directly to the tags of the tag assignments that appear in $P_{u,r}$. This is illustrated in Figure 3. Hence, the methodology provides a substantially harder problem for the task *interests match* than *guided search*. This is because for the task *interests match*, u is used as query entity, which in the folksonomy, is no longer related to r . *Guided search*, however uses a subset of the tags of the post as query entity, which are potentially still connected to r . Carmel et al. point out, that to overcome the incompleteness problem, re-

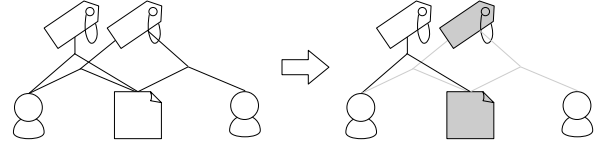


Figure 4: The LeaveRTOut methodology eliminates all tag assignments between a tag and a resource. No information remains in the folksonomy that directly connects the tag with the resource.

sults obtained from *LeavePostOut* should be validated with alternative evaluation methodologies [9]. Therefore, for this work, *LeavePostOut* is complemented with *LeaveNPostsOut*, *LeaveRTOut* and *LeaveNRTsOut*. A possibility to alleviate the incompleteness problem for *interests match* is to use the variation *LeaveNPostsOut*, which, instead of removing one post $P_{u,r}$, removes n random posts. Hence, $\frac{|P|}{n}$ posts of each user u are taken out on average. The ranking algorithm, then, has to rank resource r of any removed post $P_{u,r}$ of user u , for *interests match*, at the top of the ranking, or as high as possible. For *guided search*, using t as query entity, the ranking algorithm has to rank resource r of any removed post $P_{u,r}$, in which tag t appears in a tag assignment, at the top of the ranking, or as high as possible. This allows for a trade-off between how much data of a corpus can be used as information to create a ranking, and alleviating the incompleteness problem.

We propose *LeaveRTOut* as an evaluation methodology inspired by *LeavePostOut*. In *LeavePostOut*, after $P_{u,r}$ is removed, u and r are considered unconnected. However, a tag t in a tag assignment of $P_{u,r}$ may still be connected to r . An alternative is thus *LeaveRTOut*, which instead of eliminating the connection in the folksonomy between a user u and a resource r , eliminates the connection in the folksonomy between a tag t and a resource r as illustrated in Figure 4 and described in Algorithm 3.1.

$$RT_{r,t} = \{(u, r, t) | (u, r, t) \in Y\} \quad (4)$$

$$RT = \bigcup_{\forall r,t} RT_{r,t} \quad (5)$$

The procedure *CREATEQUERYENTITIES*($RT_{r,t}$) in Algorithm 3.1 selects a subset of the involved user and tag entities of the set of tag assignments $RT_{r,t}$. Similar to *LeavePostOut*, *LeaveRTOut* makes Assumption 4. In contrast to *LeavePostOut*, in *LeaveRTOut*, the task *guided search* is substantially harder to solve than *interests match*. Similar to *LeaveNPostsOut*, *LeaveNRTsOut* can be used to alleviate the incompleteness problem. Instead of removing one resource tag connection $RT_{r,t}$ n random resource tag connections are removed. Hence, $\frac{|RT|}{n}$ resource connections of each tag t are taken out on average. *LeaveNRTsOut* can alleviate the incompleteness problem for the task *guided search*.

3.2 Corpus

As folksonomy corpus, a dump³ of the publication management system BibSonomy [4] is used. BibSonomy allows to tag scientific publications (bibtex resources) and arbitrary

³Knowledge and Data Engineering Group, University of Kassel: Benchmark Folksonomy Data from BibSonomy, version of July 7th, 2011

Algorithm 3.1 LeaveRTOut evaluation methodology

Input: Folksonomy $F = (U, R, T, Y)$

```
1: procedure LEAVERTOOUT( $F$ )
2:   for all  $r \in R$  do
3:     for all  $t \in T$ , where  $\exists(u, r, t) \in Y$  do
4:        $F = F \setminus RT_{r,t}$ 
5:       ASSESSRANKINGQUALITY(SCORE( $F$ ,
6:       CREATEQUERYENTITIES( $RT_{r,t}$ )))
7:        $F = F \cup RT_{r,t}$ 
```

resources addressable via a URL (bookmark resources). A p-core [13] of level l guarantees a corpus to contain only entities that appear in at least l posts. In this work, $l = 5$ is used to extract a p-core of the corpus and hence reduce noise due to e.g. infrequent tags, and to focus on the dense part of the folksonomy. Before a p-core is extracted, the corpus is reduced to a manageable size, as the computation of VSScore in its current realization unfortunately does not yet scale well with large corpora. Hence, tag assignments for both bookmarks and bibtex resources are added alternately in temporal order, beginning with the oldest, until a manageable size for evaluation is obtained. Hence, before the extraction of a p-core, the reduced corpus consists of as many bookmarks as bibtex tag assignments. The characteristics of the corpus before and after the extraction of a p-core are shown in Table 1.

3.3 Metrics

The metrics *Mean Average Precision (MAP)*, *Average Precision* [15] and *Mean Normalized Precision (MNP)* at k are used for the evaluations. MAP and Average Precision are used to determine the overall ranking quality for a set of information needs or a single information need respectively. MNP at k determines the ranking quality in the top k positions of the ranking. *MNP at k* is derived from *Precision at k* [15] to obtain a single measure over a number of information needs I as well as to be more suitable for the evaluation methodology, i.e. respect the maximal achievable *Precision_{max}(k)*. For $k = 10$ using LeavePostOut e.g. *Precision_{max}(k)* = $\frac{1}{10}$ due to the incompleteness problem described previously.

$$MNP(I, k) = \frac{1}{|I|} \cdot \sum_{j=1}^{|I|} \frac{Precision_j(k)}{Precision_{max,j}(k)} \quad (6)$$

3.4 Parameterization

To determine good parameter values for the scoring algorithms under investigation we use LeavePostOut for the task

Table 1: Characteristics of the BibSonomy corpus before and after reduction and extraction of p-core

	Before	After
Users	7243	69
Bookmark resources	281550	9
Bibtex resources	469654	134
Tags	216094	179
Tag assignments	2740834	3269
Bookmark posts	330192	51
Bibtex posts	526691	959

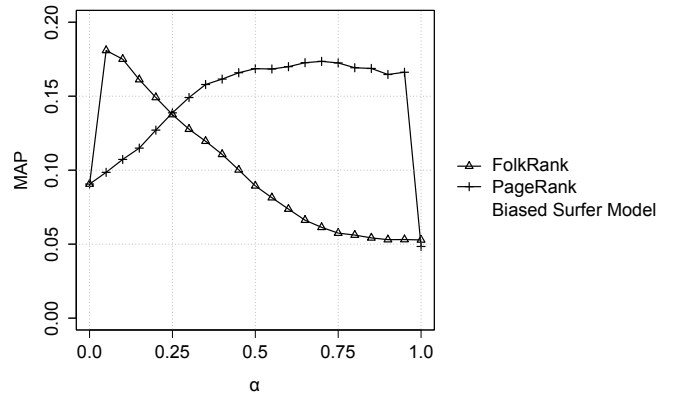


Figure 5: Influence of the bias α of the surfer model in FolkRank and PageRank to visit the nodes of the query entities on MAP. Analysis performed for *interests match* evaluated with LeavePostOut on the BibSonomy corpus. Best MAP is achieved with FolkRank and $\alpha = 0.05$

interests match and the reduced corpus described previously. The ranking effectiveness is measured with MAP.

FolkRank and PageRank with the biased surfer model can be parameterized with parameter α . The parameter determines the bias of the surfer model to visit the nodes of the query entities during its probabilistic graph traversal. In the following, a good value for the parameter α for FolkRank on the BibSonomy corpus is found via sensitivity analysis. A possible overfitting of FolkRank is for the purpose of this work neglected, as the algorithm is not used in a practical application but solely for ranking with the corpus for which it is optimized. With steps of 0.05, α is varied in the interval $\alpha \in [0, 1]$. The results are shown in Figure 5. As can be seen, the maximal MAP for FolkRank is achieved with $\alpha = 0.05$. The maximal MAP for PageRank with the biased surfer model on FolkRank’s folksonomy graph is achieved with $\alpha = 0.70$. FolkRank achieves a higher MAP than PageRank with the biased surfer model, therefore in the remainder of the evaluation FolkRank is used as baseline. On the basis of these results, $\alpha = 0.05$ is set for the remainder of the evaluation with FolkRank.

VSScore creates a vector representation of an entity’s context as described in Algorithm 2.1. Due to its better performance, FolkRank is used as scoring algorithm that implements SCORE(Q) of Algorithm 2.1. However, any algorithm that ranks entities in the folksonomy could principally implement SCORE(Q) and thus be extended by the VSScore framework. The dimensions of the vector space consist of E (all entities U, R, T in the folksonomy). The restriction of the vector space to e.g. T has shown inferior effectiveness in preliminary experiments. Cosine-similarity is used as distance measure in the vector space as it is standard in text retrieval [15]. Other measures such as Spearman’s rank correlation coefficient do not allow to incorporate the exact scores which produce a ranking and are hence not investigated further. A more detailed investigation of the parameter space of VSScore is deferred for later work.

3.5 Evaluation Hypothesis

Firstly, we make the hypothesis, that the inclusion of context into the resource recommendation by VSScore allows for

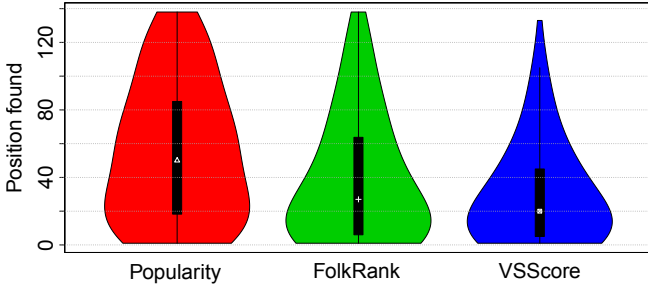


Figure 6: Violinplot for *interests match* evaluated with LeavePostOut

Table 2: MAP for *interests match* evaluated with LeavePostOut

Popularity	FolkRank	VSScore
0.0943	0.1809	0.1972

improved results over FolkRank. This hypothesis is made regardless of the recommendation task, as the entities are treated and described by their context uniformly in VSScore.

Secondly, we hypothesize that both algorithms outperform Popularity as Popularity has in related work shown to be outperformed by other recommendation technologies in resource recommendation in folksonomies [5].

3.6 Results

In the presented evaluation, LeavePostOut is used for the task *interests match*. Figure 6 shows the results of positions where relevant resources are found as a violin plot [11]. Table 2 and Figure 7 show the results of the metrics MAP and MNP at k for $k \in [1, 10]$ respectively. As can be seen in Table 2, VSScore can outperform FolkRank, which on the other hand outperforms Popularity. Similarly, the algorithms are effective with ranking in the top positions, as shown in Figure 7. To determine statistical significance of effectiveness of the overall ranking of the algorithms, significance tests based on Average Precision are conducted with a significance level of $p = 0.05$. The pairwise comparisons show that VSScore is significantly more effective than FolkRank, which is significantly more effective than Popularity.

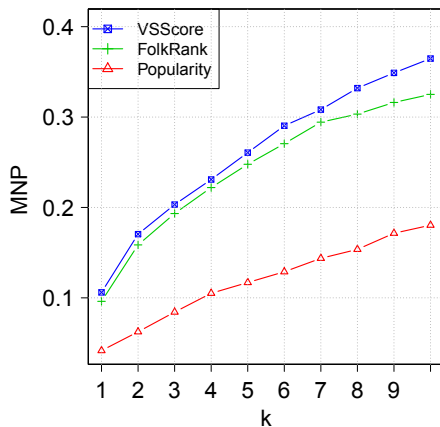


Figure 7: MNP at k for *interests match* evaluated with LeavePostOut

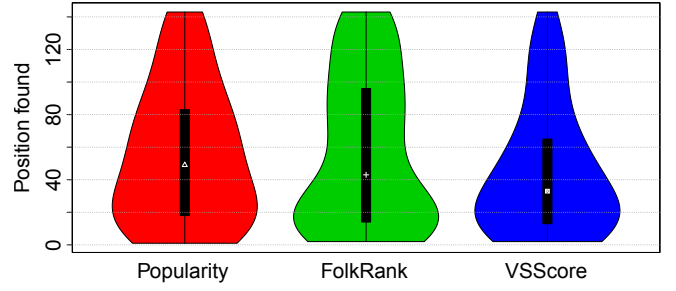


Figure 8: Violinplot for *guided search* evaluated with LeaveRTOut

Table 3: MAP for *guided search* evaluated with LeaveRTOut

Popularity	FolkRank	VSScore
0.0834	0.0529	0.0592

The results show that it is beneficial for VSScore to represent a query and resource extensively by their context, which confirms our first hypothesis. Additionally, both FolkRank and VSScore outperform Popularity, which confirms our second hypothesis.

In the following evaluation, LeaveRTOut is used for the task *guided search*. Figure 8 shows the results of positions where relevant resources are found as a violin plot. Table 3 and Figure 9 show the results of the metrics MAP and MNP at k for $k \in [1, 10]$ respectively. As can be seen in Table 3, Popularity can outperform VSScore, while VSScore can outperform FolkRank with regard to MAP. Similarly, the algorithms are effective with ranking in the top positions, as shown in Figure 9. However, Popularity is superseded by VSScore at $k = 9$ and FolkRank at $k = 10$. The pairwise comparisons of statistical significance based on Average Precision show that VSScore is significantly more effective than FolkRank, while FolkRank is significantly more effective than Popularity. However, Popularity achieves a higher MAP and shows best results for ranking in the very top positions. An explanation for this observation is that Popularity achieves very good results in the very top positions, which allow for a significant increase in MAP. However, most of

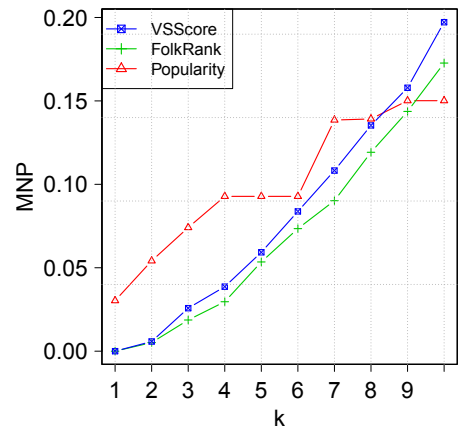


Figure 9: MNP at k for *guided search* evaluated with LeaveRTOut

Table 4: Summary of statistical significance tests of Average Precision results obtained with the different evaluation methodologies for the respective tasks. The algorithms that win most pairwise statistical significance comparisons for a certain evaluation scenario are shown. Note: VSScore performs significantly more effectively than FolkRank in 5 of 6 evaluation scenarios related to resource recommendation.

Methodology	<i>Interests match</i>	<i>Guided search</i>
LeavePostOut	VSScore ^a	VSScore ^c
LeaveNPostsOut	VSScore ^a	FolkRank, VSScore ^c
LeaveRTOut	FolkRank ^b	VSScore ^c
LeaveNRTsOut	FolkRank ^b	VSScore ^c

^aUser-based resource recommendation scenario

^bRanking of user’s resources for their fit to the user’s profile

^cTag-based resource recommendation scenario

the queries evaluated show a worse performance with Popularity than with both other approaches, leading to a statistically significant more effective VSScore and FolkRank. This result can be observed in the violin plot in Figure 8. The results confirm our first hypothesis and observation from the first evaluation, that it is beneficial for VSScore to represent query and resource extensively by their context. However, it can not be said that Popularity is outperformed by VSScore or FolkRank. This is contrary to the second hypothesis. A possible reason for this may be the relatively small corpus used.

3.7 Synopsis

Table 4 summarizes the findings of all evaluation scenarios and their corresponding statistical significance tests. For the respective tasks and evaluation methodologies, the algorithms that win most pairwise statistical significance comparisons are shown. For LeaveNPostsOut, N was chosen such that $\frac{|P|}{n} = \frac{1}{3}$. For LeaveNRTsOut, N was chosen such that $\frac{|RTs|}{n} = \frac{1}{3}$. The results relevant for resource recommendation are highlighted.

For the *interests match* task, the evaluation results from LeavePostOut differ from those obtained with LeaveRTOut. This is due to the fact, that, as already described, they set a differently hard task to solve. In fact, these results are however not obtained for a user-based resource recommendation scenario. LeaveRTOut, on the one hand, is useful to assess the effectiveness for an *interests match* task, in which e.g. the current resources of a user are to be presented in the order of how much they match the user’s interest. This is because a connection between a user and a potential relevant resource may still exist. The results obtained from LeavePostOut, on the other hand, are useful to assess the effectiveness for *interests match* in a resource recommendation task. There, no connection between the user and a potential relevant resource exists, but it is the task of the recommendation algorithm to find such relevant resources.

Our first hypothesis is confirmed for 5 out of 6 evaluation scenarios related to resource recommendation. Our second hypothesis is confirmed for 4 out of 6 evaluation scenarios related to resource recommendation. For a *guided search* task evaluated with LeaveRTOut and LeaveNRTsOut the obtained results are contrary to our second hypothesis. As

shown for LeaveRTOut, neither in LeaveNRTsOut it can be said that Popularity is outperformed by any of its contestants.

One further has to consider that to obtain the results of this evaluation, the parameterization for some of the algorithms was done using an analysis of MAP results obtained for the LeavePostOut methodology and the task *interests match*. Hence, the algorithms may perform better with regard to a metric, or task if parameterized accordingly. Additionally, the statistical significance was computed based on Average Precision, which is a measure of the overall ranking quality. If the statistical significance is to be compared based on the effectiveness of ranking in top positions, a different series of significance tests need to be conducted.

4. RELATED WORK

Resource recommendation in folksonomies has been studied for different recommendation tasks and using techniques that leverage different sources of information. One of the first resource recommendation approaches in folksonomies has been studied by Herlocker et al. In [10], a two-dimensional collaborative filtering approach is discussed that leverages the user and resource dimension of folksonomies. However, in this approach the additional dimension provided by tags is neglected. A popular recommendation variant in folksonomies is FolkRank [12] proposed by Hotho et al. FolkRank is a very flexible algorithm which allows for resource, tag, and user recommendation and is independent of the type of query entity. As the folksonomy is represented as a graph in this approach, it is often referred to as graph-based recommendation or ranking [17]. Ramezani argues that the folksonomy graph representation used in FolkRank does not reflect the flow of information in a folksonomy well [17]. Hence, she introduces a weighted directed graph model for folksonomies, which she evaluates for tag recommendation.

Bogers describes a flexible approach to incorporate context information into graph-based recommendation named ContextWalk [6]. ContextWalk is in this work applied to movie recommendation for a movie database website. In [2], Abel et al. show how user-provided context-specific information in folksonomies, in the form of group structures, can benefit recommendation. Therefore, FolkRank is extended to GFolkRank and GRank is introduced. Abel et al. investigate further how other context-specific and semantic knowledge can have a positive impact on recommendation. To extend FolkRank, a tag category, spatial information from the tagged area within a resource, and from a URL which describes the semantic of a tag are investigated in [3]. In [8], Cantador et al. investigate how the mapping of semantic concepts of an ontology to resources, describing their content, and users, indicating their preferences, can support recommendation in a news recommender system. Cantador et al. spread the information about resources and users by semantic contextualization, a relatedness of semantic concepts in the ontology, thus enriching the initially sparse vector describing resources and users.

To leverage the vector space model in folksonomies and to overcome the sparseness of information in folksonomies, Abbasi et al. propose enriched vector space models [1]. However, the enrichment of the vector space in this work is limited to relationships between tags and users and tags and resources. Additionally, the dimensions of the vector space is limited to the tag space in this approach.

The VSScore approach we presented in this work is dif-

ferent from the works above with regard to the following points. VSScore is a framework for recommendation in folksonomies. Any ranking algorithm for folksonomies may be extended by VSScore to overcome the drawbacks discussed in Section 2.1. The vector space model leveraged is flexible and depends on the aforementioned ranking algorithm. Hence, the dimensions of the vector space may encompass any entity type of the folksonomy. Context is regarded as a representation of the semantic concept of an entity in the folksonomy. Hence, query entities and resources are represented by their context to produce recommendations.

5. CONCLUSION

In this work, we described a method to model context-specific information that is inherent in folksonomies. With VSScore we proposed a framework that leverages this context for resource recommendation. The hypothesis that this context-specific information allows to overcome drawbacks of existing resource recommendation algorithms in folksonomies was confirmed in an evaluation setup. The evaluation comprised user-based and tag-based resource recommendation. VSScore outperformed FolkRank in 5 out of 6 evaluation scenarios of these recommendation tasks. Other resource recommendation algorithms which leverage context-specific information could not be compared as they require context information which is not stored in folksonomies.

As VSScore is computationally complex, unfortunately the evaluation of this work could only be performed on a limited corpus. However, the larger the folksonomy, the more detailed the description of an entity can be via its context. Hence, in future work, VSScore will be evaluated with a corpus of larger size as well as with corpora from different domains. Reduction of the high-dimensional vector space to reduce computational complexity provides space for improvements. As a dense vector representation of a user with a sparse profile can be generated, another interesting future work is to investigate VSScore specifically in a cold-start setting. Moreover, the parameter space of VSScore is to be investigated for further improvements. An interesting approach for a tag-based resource recommendation scenario may be to realize $\text{CREATEENTITYCONTEXT}(F, e)$ such that it returns a vector description of the query tag in a reference corpus, e.g. Wikipedia. A reduction of this vector to the vector space of tags in the folksonomy allows then to apply VSScore. Additionally, the vector distance used in VSScore provides possibility for further research.

VSScore may equally be used for recommendation of tags as well as users in folksonomies. Additionally, VSScore may e.g. be used for resource-based resource recommendation or personalized tag-based resource recommendation. This provides further possibilities for future work.

Another approach to improve recommendation in folksonomies is to disambiguate tags. A possibility to automatically disambiguate tags could be by leveraging their context, which was described in this work, with a subset of their tag assignments, e.g. a post.

6. REFERENCES

- [1] R. Abbasi and S. Staab. RichVSM: enRiched Vector Space Models for Folksonomies. In *Proc. of 20th ACM Conference on Hypertext and Hypermedia*, 2009.
- [2] F. Abel, M. Baldoni, C. Baroglio, N. Henze, D. Krause, and V. Patti. Context-based Ranking in Folksonomies. In *Proc. of the 20th ACM conference on Hypertext and Hypermedia*, pages 209–218, 2009.
- [3] F. Abel, N. Henze, R. Kawase, and D. Krause. The Impact of Multifaceted Tagging on Learning Tag Relations and Search. In *Proc. of the 7th ESWC*, 2010.
- [4] 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, 2010.
- [5] T. Bogers. *Recommender Systems for Social Bookmarking*. PhD Thesis, Tilburg University, 2009.
- [6] T. Bogers. Movie Recommendation Using Random Walks Over the Contextual Graph. In *Proc. of the 2nd Intl. Workshop on Context-Aware Recommender Systems*, 2010.
- [7] D. Böhnstedt, P. Scholl, C. Rensing, and R. Steinmetz. Collaborative Semantic Tagging of Web Resources on the Basis of Individual Knowledge Networks. In *Proc. of the 17th Intl. Conference on User Modeling, Adaptation, and Personalization*, pages 379–384, 2009.
- [8] I. Cantador and P. Castells. Semantic Contextualisation in a News Recommender System. In *Proc. of the 1st Intl. Workshop on Context-Aware Recommender Systems*, 2009.
- [9] D. Carmel, N. Zwerdling, I. Guy, S. Ofek-Koifman, N. Har’el, I. Ronen, E. Uziel, S. Yogev, and S. Chernov. Personalized Social Search Based on the User’s Social Network. In *Proc. of the 18th ACM Conference on Information and Knowledge Management*, pages 1227–1236, 2009.
- [10] J. Herlocker, J. Konstan, A. Borchers, and J. Riedl. An Algorithmic Framework for Performing Collaborative Filtering. In *Proc. of the 22nd ACM SIG on Information Retrieval Conference*, pages 230–237, 1999.
- [11] J. Hintze and R. Nelson. Violin plots: A box plot-density trace synergism. *The American Statistician*, 52(2):181–184, 1998.
- [12] A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme. Information Retrieval in Folksonomies: Search and Ranking. In *Proc. of the 3rd ESWC*, pages 411–426, 2006.
- [13] R. Jäschke, L. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme. Tag Recommendations in Folksonomies. In *Proc. of the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 506–514, 2007.
- [14] D. Jurafsky and J. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall, 2008.
- [15] C. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. CUP, 2008.
- [16] I. Peters. *Folksonomies: Indexing and Retrieval in Web 2.0*. De Gruyter Saur, 2010.
- [17] M. Ramezani, J. Gemmell, T. Schimoler, and B. Mobasher. Improving Link Analysis for Tag Recommendation in Folksonomies. In *Proc. of the 2nd Intl. Workshop on Recommender Systems and the Social Web*, 2010.