

The Impact of Multifaceted Tagging on Learning Tag Relations and Search

Fabian Abel, Nicola Henze, Ricardo Kawase, and Daniel Krause

IVS – Semantic Web Group & L3S Research Center, Leibniz University Hannover,
Germany

{abel,henze,kawase,krause}@L3S.de

Abstract. In this paper we present a model for multifaceted tagging, i.e. tagging enriched with contextual information. We present TagMe!, a social tagging front-end for Flickr images, that provides multifaceted tagging functionality: It enables users to attach tag assignments to a specific area within an image and to categorize tag assignments. Moreover, TagMe! maps tags and categories to DBpedia URIs to clearly define the meaning of freely-chosen words. Our experiments reveal the benefits of these additional tagging facets. For example, the exploitation of the facets significantly improves the performance of FolkRank-based search. Further, we demonstrate the benefits of TagMe! tagging facets for learning semantics within folksonomies.

1 Introduction

Tagging systems like Flickr or Delicious organize and search large item collections by utilizing the Web 2.0 phenomena: users attach tags to resources and thereby create so-called tag assignments which are valuable metadata. However, imprecise or ambiguous tag assignments can decrease the performance of tagging systems regarding search and retrieval of relevant items.

For example a tag assigned to an image may describe the image with respect to different dimensions referring to the owner, an opinion or the usage context (task) and thus is only valid for a user-specific point of view [1]. Not all tags can be used for search, but most often tags describing the content of a document are applicable for search [2]. However, even those descriptive tags might characterize just a small part of an image and hence cannot be used to derive the overall topic of the image correctly. Finally, tag assignments suffer from ambiguity in natural languages. For disambiguation, approaches like MOAT [3] exist, which support users to attach URIs describing the meaning of a tag to a particular tag assignment analogously to semantic tagging in Faviki¹.

In this paper, we extend the common folksonomy model by flexible, contextual tagging facets. We present the TagMe! system² that introduces novel tagging facets: tag assignments are enriched with a DBpedia URI [4] to disambiguate the meaning of a tag. So-called *area tags* enable users to tag a specific part of an image (spatial tagging). Furthermore, a *category* dimension is offered to categorize

¹ <http://faviki.com>

² <http://tagme.groupme.org>

tag assignments. We propose different ranking strategies that exploit the additional context by extending the FolkRank [10] algorithm. In our evaluation we show how the different strategies improve search significantly (see Section 4.2) and analyze the impact of the additional context on learning semantic relations among tags (see Section 4.3).

2 Context in Folksonomies

Folksonomies evolve over time when users annotate resources with freely chosen keywords. Research in the area of folksonomy systems most often focusses on lightweight models where folksonomies are basically considered as collections of tag assignments. In this section we introduce approaches for modeling context in folksonomy systems.

2.1 Folksonomy Models

Traditional folksonomy models describe the relations between users, tags and resources. According to [5], a folksonomy can be defined as follows.

Definition 1. *A folksonomy is a quadruple $\mathbb{F} := (U, T, R, Y)$, where U , T , R are finite sets of instances of users, tags, and resources. Y defines a relation, the tag assignment, between these sets, that is, $Y \subseteq U \times T \times R$.*

Some systems imply a folksonomy model that incorporates additional information indicating in which context a tag was assigned to a resource. In particular, such context might be formed by groups, which are finite sets of resources [6]. In Definition 2 we introduce a more generic folksonomy model that considers arbitrary type of context.

Definition 2. *A context folksonomy is a tuple $\mathbb{F} := (U, T, R, Y, C, Z)$, where:*

- U , T , R , C are finite sets of instances of users, tags, resources, and context information respectively,
- Y defines a relation, the tag assignment that is, $Y \subseteq U \times T \times R$ and
- Z defines a relation, the context assignment that is $Z \subseteq Y \times C$

Given the context folksonomy model, it is possible to attach any kind of context to tag assignments. For example, the context folksonomy allows for tagging tag assignments. TagMe! [7], a tagging and exploration front-end for Flickr pictures, introduces three types of context: (i) spatial information describing to which part of a resource a tag assignment belongs to, (ii) categories for organizing tag assignments, and (iii) DBpedia URIs that describe the semantic meaning of a tag assignment. Such contextual information is simply assigned to a tag assignment by the relation Z .

The spatial information as well as the categories are explicitly provided by the end-users via the tagging interface of TagMe! (see Figure 1). For each tag assignment a user can enter one or more categories that classify the annotation. While typing in a category, the users get auto-completion suggestions from the pre-existing categories of the user community (see bottom in Figure 1). When

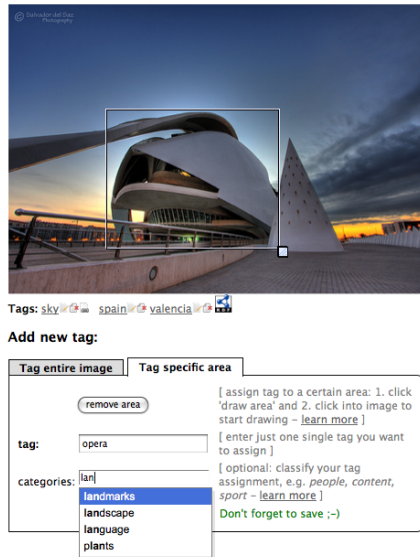


Fig. 1. User tags an area within an image and categorizes the tag assignment with support of the TagMe! system.

a user categorizes a tag assignment $y = (u, t, r) \in Y$ into category c then this is modeled as relation $(y, c) \in Z$ where $c \in C$ can actually be an arbitrary tag, i.e. $c \in T$. Additionally, users are enabled to perform *spatial tag assignments*, i.e. to attach a tag assignment to a specific area, which they can draw within the picture (see rectangle within the photo in Figure 1) similarly to *notes* in Flickr or annotations in LabelMe [8]. In the context folksonomy a spatial tag assignment is simply modeled via a relation $(c_a, y) \in Z$ where c_a refers to the context information that describes the area that is tagged. TagMe! automatically assigns DBpedia URIs to tag assignments by exploiting the DBpedia lookup service³ (cf. Section 4.1). Hence, tag assignments have well-defined semantics so that applications, which operate on TagMe! data using the Linked Data[9] infrastructure, can clearly understand the meaning of the tag assignments. In particular, RDF data can be obtained by accessing resources via an alternative URI, e.g. <http://tagme.groupme.org/TagMe/resource/220/rdf>, where each RDF representation links to other RDF representations of users, tags, resources, and DBpedia concepts.

3 Ranking Algorithms

In this section we present the ranking algorithms that we apply in our experiments (Section 4) to reveal the benefits of exploiting context information. We first outline the FolkRank [10] algorithm, which we use as baseline strategy for ranking resources in our search experiments. Later we present different al-

³ <http://lookup.dbpedia.org>

gorithms (based on FolkRank) that leverage context folksonomies to improve ranking quality.

3.1 FolkRank

The FolkRank algorithm [10] operates on the folksonomy model specified in Definition 1. FolkRank transforms the hypergraph that is spanned by the tag assignments into a weighted tripartite graph $\mathbb{G}_{\mathbb{F}} = (V_{\mathbb{F}}, E_{\mathbb{F}})$, where an edge connects two entities (user, tag, or resource, i.e. $V_{\mathbb{F}} = U \cup T \cup R$) if both entities occur together at a tag assignment within the folksonomy: $E_{\mathbb{F}} = \{\{u, t\}, \{t, r\}, \{u, r\} | (u, t, r) \in Y\}$. The weight of an edge corresponds to the amount of the entities' co-occurrences. For example, the weight of an edge connecting a tag t and a resource r is defined as $w(t, r) = |\{u \in U : (u, t, r) \in Y\}|$ (cf. Definition 1) and thus corresponds to the number of users, who have annotated r with t . The constructed graph $\mathbb{G}_{\mathbb{F}}$ serves as input for an adaption of the Personalized PageRank [11]: $\mathbf{w} \leftarrow dA_{\mathbb{G}_{\mathbb{F}}}\mathbf{w} + (1-d)\mathbf{p}$, where the adjacency matrix $A_{\mathbb{G}_{\mathbb{F}}}$ models the folksonomy graph $\mathbb{G}_{\mathbb{F}}$, \mathbf{p} allows to specify preferences (e.g. for a tag) and d enables to adjust the influence of the preference vector. FolkRank applies the adapted PageRank twice, first with $d = 1$ and second with $d < 1$ (in our evaluation we set $d = 0.7$ as done in [10]). The final vector, $\mathbf{w} = \mathbf{w}_{d < 1} - \mathbf{w}_{d=1}$, contains the *FolkRank* of each folksonomy entity.

3.2 Category-based FolkRank

The category-based FolkRank algorithm operates on a context folksonomy (see Definition 2) where the context is given by categories that are attached to tag assignments. The algorithm relates folksonomy entities via the category assignments and the main hypothesis is that entities sharing the same category are related to each other. Similarly to GFolkRank, the category-based FolkRank introduces an alternative approach for creating the weighted folksonomy graph $\mathbb{G}_{\mathbb{F}}$ (cf. Section 3.1). Categories are treated as tags ($c \in T_C$ where $T_C \subseteq T$) so that the set of nodes is extended with T_C : $V_{\mathbb{F}_{new}} = V_{\mathbb{F}} \cup T_C$. For each category assignment $(y, c) \in Z$, new edges are created to connect the given category c with the resource and tag of the tag assignment y : $E_{\mathbb{F}_{new}} = E_{\mathbb{F}} \cup \{\{c, r\}, \{c, t\} | c \in T_C, t \in T, r \in R, ((u, t, r), c) \in Z\}$. The weight of an edge (c, r) corresponds to the frequency the category c is assigned to a tag assignment that refers to r : $w(c, r) = |\{(u, t, r) \in Y : (u, t, r) \in Y, ((u, t, r), c) \in Z\}|$. Weights of (c, t) -edges are accordingly computed by counting the tag assignments that refer to t and are categorized using c .

3.3 Area-based FolkRank

While the categories are used to enrich the folksonomy graph with further edges and possibly also with further vertices, the area-based FolkRank merely modifies the weights of edges in $\mathbb{G}_{\mathbb{F}}$ (cf. Section 3.1). In particular, it emphasizes the weight of an edge between a tag t and a resource r , i.e. (t, r) -edges, whenever t and r occur within a tag assignment $(u, t, r) \in Y$ to which spatial context information

is attached to. The amplification is based on the size of the corresponding area as well as on the distance of the midpoint of the area to the center of the resource (in our experiments we examine pictures).

- **size** Our hypothesis is that the larger the size of an area the more important is also the corresponding tag for the given resource, i.e. the larger the area that is attached to $(u, t, r) \in Y$ is the more relevant t is for r . The size of an area is measured relatively to the size of the resource. For example, if an area is associated to a tag assignment (u, t, r) and the relative size of the area is $s = 0.4$, i.e. the area covers 40% of the resource, then we use s^{-1} to emphasize the weight $w(t, r)$. As different users might attach differently sized areas to (u, t, r) , we use the average size \bar{s} of those areas to finally compute the new weight of (t, r) -edges: $w(t, r)_s = \bar{s}^{-1} \cdot w(t, r)$.
- **distance** The second hypothesis is that tag assignments which are according to the spatial information relevant to the center of a resource are more important for the resource than tag assignments which are associated to the margin of a resource. The distance d is also measured relatively and the weight $w(t, r)$ is emphasized with the average distance \bar{d} of the areas attached to $(u, t, r) \in Y$: $w(t, r)_d = \bar{d}^{-1} \cdot w(t, r)$.

Finally, the weight of the edges (t, r) is simply the average of $w(t, r)_s$ and $w(t, r)_d$: $w(t, r)_{area} = 0.5 \cdot w(t, r)_s + 0.5 \cdot w(t, r)_d$.

3.4 URI-based FolkRank

The URI-based FolkRank operates on meaningful URIs instead of tags. Hence, the construction of the folksonomy graph $\mathbb{G}_{\mathbb{F}} = (V_{\mathbb{F}}, E_{\mathbb{F}})$ is modified as follows. The set of vertices is $V_{\mathbb{F}} = U \cup URI \cup R$, where $URI \subseteq C$ (cf. Definition 2) denotes the set of URIs that describe the meaning of the tag assignments. The set of edges is $E_{\mathbb{F}} = \{u, uri\}, \{uri, r\}, \{u, r\} | u \in U, uri \in URI, r \in R, ((u, t, r), uri) \in Z\}$ whereas there should only exist exactly one URI assignment $(y, uri) \in Z$ for each tag assignment y . The weights of the edges are computed in the same way as done by the traditional FolkRank algorithm.

The URI-based FolkRank algorithm is therewith resistant against ambiguous tags as well as synonymic tags. For example, given two tag assignments $y_1 = (u_1, t_1, r_1)$ and $y_2 = (u_2, t_2, r_2)$ as well as two context assignments (y_1, uri_1) and (y_2, uri_1) , the URI-based FolkRank algorithm would replace the synonymic tags t_1 and t_2 by the unique URI uri_1 that clearly defines the meaning of the tags. It therewith, e.g., relates r_1 and r_2 as it constructs the edges (uri_1, r_1) and (uri_1, r_2) . As the TagMe! system utilizes DBpedia URIs to define the meaning of tags, we denote the URI-based FolkRank as *DBpedia FolkRank* in our search experiments in Section 4.

4 Evaluation

In this Section we analyze the benefits of multi-faceted tagging resulting in context folksonomies (see Definition 2), i.e. contextual information that is embedded in the folksonomy model. In particular, we analyze the three context types introduced by the TagMe! system.

1. *Categories* for organizing tag assignments.
2. *Spatial information* (areas) describing to which part of a resource a tag assignment belongs to
3. *DBpedia URIs* that describe the semantic meaning of a tag assignment.

From the TagMe! data it seems that users appreciate those tagging facets, e.g. 899 of the 1264 tag assignments, which were performed within the three weeks after the launch of the system, were categorized and 657 times the users assigned a tag to a specific area within a picture. In the evaluation period the system was used by 30 students with background in computer science where half of the participants also had a Flickr account and thus were able to tag own pictures while the other half did not had a Flickr account.

In our evaluation we examine the impact of the additional context generated by the multi-faceted tagging on search and mining tag relations. In particular, the key questions we would like to answer are the following.

1. Does the exploitation of the additional context improve the search and ranking performance?
2. Do the different context types offer potential to learn advanced types of relations among tags?

To better understand the context types available in TagMe!, we first present the results of analyses in which we investigated potential benefits of the categories and areas and examined strategies for mapping tags to DBpedia URIs (see Section 4.1. In Section 4.2 we afterwards summarize the results of our experiments that reveal the positive impact of the TagMe! context folksonomy on search. Finally, we examine the possibilities of learning semantic relations between tags in the TagMe! folksonomy.

4.1 Preliminary Analysis

In our preliminary analysis we try to gain first insights into the characteristics of the three TagMe! context types. We target the following questions.

1. How are *categories* used in comparison to tags?
2. What are the benefits of assigning tags to specific *areas* within an image (spatial information)?
3. How accurate can tags (and categories) be mapped to *DBpedia URIs* describing the meaning of the annotations?

Analysis of Category Usage and Benefits Figure 2(a) shows the evolution of the number of distinct tags and categories: Although categories can be entered freely like tags, they grow much less than tags. Further, only 31 of the 87 distinct categories (e.g., “car” or “sea”) have also been used as tags, which means that users seem to use different kinds of concepts for categories and tags respectively.

The TagMe! system supports users in assigning categories by means of auto-completion (see Figure 1). During our evaluation we divided the users into two groups: 50% of the users (*group A*) got only those categories as suggestion, which

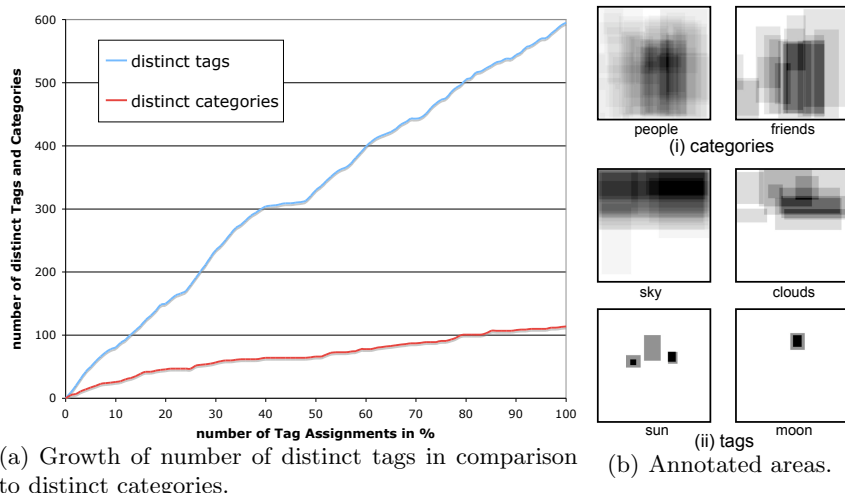


Fig. 2. Annotation behavior in TagMe!

they themselves used before, while the other 50% of the users (*group B*) got categories as suggestions, which were created by themselves or by another user within their group. This small difference in the functionality had a big impact on the alignment of the categories. The number of distinct categories in group A was growing 61.94% stronger than in group B. Hence, the vocabulary of the categories can be aligned much better if categories, which have been applied by other users, are provided as suggestions as well.

Analysis of Spatial Tagging Information Categories can be differentiated according to their usage. For example, some categories have never or very seldomly been used when a specific area of an image was tagged (e.g., “time”, “location”, or “art”) while others have been applied almost only for tagging a specific area (e.g., “people”, “animals”, or “things”).

The areas, can moreover be used to learn relations among categories and tags. Figure 2(b) shows (i) the areas that have been annotated whenever the categories “people” and “friends” have been used (the darker an area the more tags have been assigned to that area). As the areas that have been tagged in both categories strongly correlate and as category “people” was used more often than category “friends” one can deduce that “friends” is possibly a *sub-category* of “people” even if both categories would never co-occur at the same resource. Relations between tags can also deduced by analyzing the tagged areas. Figure 2(b) shows (ii) the areas that were tagged with “sky”, “clouds”, “sun”, and “moon”⁴ and via the size and position of the area it is possible to learn that an entity *is part of* or *contained in* another entity (e.g., “sun, moon, and clouds are contained in sky”). The learned relations among tags and categories can moreover be used

⁴ The visualizations are based on 25 (“sky”), 10 (“clouds”), 6 (“sun”), and 2 (“moon”) tag assignments respectively.

to learn and refine relations between URIs (ontology concepts) as TagMe! maps tags and categories to DBpedia URIs.

Mapping to DBpedia URIs For realizing the feature of mapping tags and categories to DBpedia [4] URIs we compared the following two strategies.

- **DBpedia Lookup** The naive lookup strategy invokes the DBpedia lookup service with the tag/category that should be mapped to a URI as search query. DBpedia ranks the returned URIs similarly to PageRank [12] and our naive mapping strategy simply assigns the top-ranked URI to the tag/category in order to define its meaning.
- **DBpedia Lookup + Feedback** The advanced mapping strategy is able to consider feedback while selecting an appropriate DBpedia URI. Whenever a tag/category is assigned, for which already a correctly validated DBpedia URI exists in the TagMe! database then that URI is selected. Otherwise the strategy falls back the naive DBpedia Lookup.

The mappings of the naive approach result in a precision of 79.92% for mapping tags to DBpedia URIs and 84.94% for mapping categories considering only those tag assignments where a DBpedia URI that describes the meaning properly exists. The consideration of feedback, which is currently managed by the administrators of TagMe!, improves the precisions of the naive DBpedia Lookup clearly to 86.85% and 93.77% respectively, which corresponds to an improvement of 8.7% and 10.4%. As the mapping accuracy for categories is higher than the one for tags, it seems that the identification of meaningful URIs for categories is easier than for tags. Moreover, the precision of the category mappings, which are determined by the strategy that incorporates feedback, will further improve, because—fostered by TagMe!’s category suggestion feature—the number of distinct categories seems to converge (cf. Figure 2(a)). A more detailed evaluation of the DBpedia URI mapping is described in [13] and the implementation of advanced mapping strategies is part of our future work.

Overall, the DBpedia mapping reduces the number of distinct tags and categories within TagMe! by 14.1% and 20.9% respectively, which promises a positive impact on the recall when executing tag-based search. For example, while some users assigned the tag “car” to pictures showing cars other users chose “auto” to annotate other pictures that show cars. As both kinds of tag assignments are mapped to “http://dbpedia.org/resource/Automobile”, TagMe! can simply search via the DBpedia URI whenever users search via “car” or “auto” to increase recall of the tag-based search operations. With the *URI-based FolkRank* we present an advanced ranking algorithm that exploits the DBpedia URIs to improve the ranking performance (see evaluations in Section 4.2).

Synopsis In summary, the context types available in the TagMe! folksonomy have a positive impact on identifying correlations between the folksonomy entities (e.g., identifying similar tags). Further, categories and areas enable the extraction of additional semantic relations between tags. As tags are mapped to DBpedia URIs that describe the meaning of a tag assignment, it is possible

to deduce rich semantics from the context folksonomy available in TagMe!. The results of our preliminary analysis can be summarized as follows.

- The usage of categories differs from the usage of tags: Even for those users, who did not benefit from the category suggestions, the number of distinct categories is growing slower than the number of distinct tags.
- The spatial tag assignments can be used to learn typed relations among tags and categories such as *sub-category*, *sub-tag*, *part-of*, or *contained-in* relations. As tags and category assignments are mapped to meaningful URIs (ontological concepts), it is possible to propagate the learned relations to ontologies.
- A naive DBpedia lookup allows us to map tags and categories to ontological concepts (DBpedia URIs) with a high precision of 79.92% (tags) and 84.94% (categories). The consideration of feedback improves the accuracy of the mapping of tags and categories to 86.85% and 93.77% respectively.

This preliminary analysis already delivers insights into the potential of the context information available in the TagMe! folksonomy. In the next section we will evaluate whether categories, the size and position of areas, and the DBpedia URI assignments can be applied to improve the quality of search and ranking. Hence, we evaluate the ranking algorithms proposed in Section 3.

4.2 Search Evaluation

In our search evaluation we examine the impact of the advanced semantics provided by the TagMe! context folksonomy on search. In particular, we apply the FolkRank algorithm (see Section 3.1) as well as the Category-, Area-, and URI-based FolkRank adaptations to search and rank Flickr images and investigate how the different context types can help to improve the search performance. We evaluate the algorithms with respect to the following task.

Resource Ranking Task. *Given a keyword query (tag), the task of the ranking strategy is to compute a ranking of resources so that resources that are most relevant to the keyword query appear at the top of the ranking.*

Our primary goal is to determine whether the additional context information has a positive impact on the ranking task. We further examine the characteristic strengths and weaknesses of the different context types by comparing the ranking performance of the corresponding FolkRank-based strategies.

Data Set and Test Set We conducted our experiments on the TagMe! data set that evolved during the first month after the launch of the system. In this period the users created 1264 tag assignments where 899 tag assignments were also enriched with a category and 657 tag assignments were attached to a specific area of a Flickr resource. As outlined in Section 4.1, the number of distinct tags was growing faster than the number of distinct categories. Finally, the TagMe! data set contained distinct 610 tags and 118 distinct categories. The distribution

of the usage frequency of tags shows a typical distribution: While some tags are used very often, the majority of tags is used just once.

The DBpedia URI assignments that were automatically attached by TagMe! were validated by hand so that the data set on which we performed the experiments did not contain wrong URI assignments. The cleaned data set finally contained 360 distinct DBpedia URIs referenced by tags and 92 DBpedia URIs referenced by categories. For 17% of the tag assignments there did not exist a correct DBpedia URI mappings.

Regarding relevance assessment, we selected 24 representative tags (according to the usage frequency) as keyword queries and asked TagMe! users to rate the relevance of a picture to a given query on a five-point scale: *yes*, *rather yes*, *rather no*, *no*, and *don't know*. Thereby, we obtained for each of the query all relevant resources in the TagMe! data set. On average, for each query there were nearly 30 resources in the data set that were rated as relevant (*yes*). However, four of the queries had below 10 relevant (*yes*) resources. For all the 24 tag-based queries there was a proper DBpedia URI available in the data set.

Method and Metrics The *ranking task* defined at the beginning of the section requires the ranking strategies to rank those resources at the top of the ranking that are most relevant to the given query. We analyzed the ranking algorithms presented in Section 3 that are applicable to the TagMe! context folksonomy: FolkRank, Category-based FolkRank (CategoryFolkRank), Area-based FolkRank (AreaFolkRank), and URI-based FolkRank (DBpediaFolkRank). Each ranking strategy then had to compute a resource ranking for each of the 24 representative keyword queries. We measured the quality of the rankings using the precision at rank k ($P@K$), which represents the average proportion of *relevant* items within the top k . For our experiment we considered an item as *relevant* iff the average user judgement is at least “yes”.

In addition to the FolkRank-based approaches we also consider a ranking algorithm (denoted as “F+C+A+D”) that combines all four ranking strategies: Given the list of weighted resources as computed by the different algorithms it utilizes the average ranking weight to rank the resources.

We tested the statistical significance of our results with a two-tailed t-Test and a significance level of $\alpha = 0.05$. The null hypothesis H_0 is that some strategy s_1 is as good as another strategy s_2 , while H_1 states that s_1 is better than s_2 .

Results Figure 3 shows the precisions ($P@10$ and $P@20$) of the different ranking strategies. Those algorithms that make use of contextual information embedded in the folksonomy perform better than the traditional FolkRank algorithm that considers only the tag assignments without any additional context. Between DBpediaFolkRank and FolkRank there seems to be no remarkable performance difference. However, as noted in Section 4.2, the DBpediaFolkRank is operating on 215 fewer tag assignments than the other algorithms. It is thus remarkable that DBpediaFolkRank still performs slightly better than FolkRank. The CategoryFolkRank performs good results especially for the precision within the top 20 ($P@20$). Hence, the hypothesis raised in Section 3.2 seems to hold: Category

assignments can be used to relate resources. By exploiting the category context, the algorithm detects relevant resources that are not directly related via tag assignments to the given query. The AreaFolkRank algorithm, which exploits the size and position of spatial information attached to the tag assignments, is—with respect to P@10—the best algorithm among the core ranking strategies (P@10 = 52.9%). However, there is no significant difference between the FolkRank and the Area-, Category-, and DBpedia-based FolkRank.

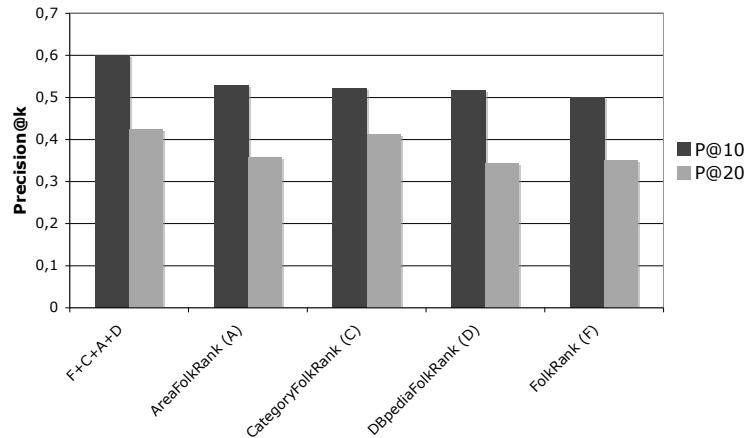


Fig. 3. Precisions of FolkRank-based search algorithms.

The strategy “F+C+A+D”, which combines all four core ranking strategies (i.e., FolkRank, CategoryFolkRank, AreaFolkRank, and DBpediaFolkRank), is the most successful strategy. It performs significantly better than the FolkRank algorithm regarding the P@10 and P@20 metrics. The combined strategy improves the precision of FolkRank by 20.0% and 21.4% with respect to the precision within the top 10 and top 20 respectively.

Figure 4 depicts the precision-recall diagram of the different FolkRank-based ranking algorithms. It underlines that the context-based approach, which combines FolkRank with the strategies that exploit the category, area, and DBpedia context, is the best performing ranking strategy as it results in the best precision-recall ratio. In the low recall interval, i.e. within the very top of the resource rankings, FolkRank can compete with the other algorithms. For example, the probability that a relevant resource appears at the first rank is 75.0% for FolkRank and 79.2% for the combined strategy. However, with higher recall values, the precision of FolkRank drops significantly stronger than the one of the Category-based FolkRank or the combined strategy F+C+A+D: At a recall level of 0.5 the precision of F+C+A+D and CategoryFolkRank is 0.29 and therewith significantly higher (approx. 45%) as the precision of FolkRank.

In summary, the exploitation of context embedded in the folksonomy is beneficial for ranking resources. While the size and position of the area helps to improve the precision particularly at the top of the resource rankings, the DBpedia and category context successfully contribute to improve the recall. And

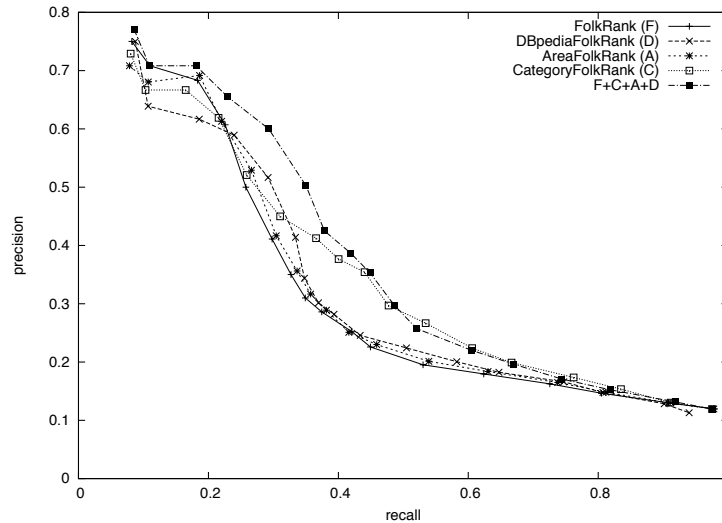


Fig. 4. Precision-recall diagram of FolkRank-based search algorithms.

by combining the different context types we are able to improve the ranking performance of FolkRank significantly.

4.3 Learning Tag Relations

In addition to the exposed benefits in terms of search we hypothesize that the additional context is also helpful for learning fine-grained relations among tags. Different studies showed already that it is possible to learn hierarchical relations among tags by exploiting traditional folksonomy structures. For example, Halpin et al. extract sub-class relations by exploiting correlations between tags induced by co-occurrence of tags [14] and Mika mines narrower and broader concepts by detecting communities and sub-communities within the tripartite graph spanned by traditional folksonomies [15] (cf. weighted tripartite graph in Section 3.1). Others apply association rule mining to detect super-sub-concept relations [16]. Given tags from Flickr, Schmitz detects tag pairs where one tag subsumes the other [17].

In this section we analyze the impact of the additional context types on learning tag relations. Therefore, we again apply the ranking strategies introduced in Section 3. However, instead of ranking resources we analyze their performance regarding ranking tags:

Tag Ranking Task. *Given a tag t , the task of the ranking strategy is to compute a ranking of tags so that tags that are most related to t appear at the top of the ranking.*

We run our experiments on the TagMe! data set as described in Section 4.2. For each of the 610 tags the ranking algorithms had to perform the ranking task defined above. For each ranking algorithm, we then accumulated the 610

rankings of tags to obtain a global ranking of tag pairs (t_a, t_b) , where the ranking score was computed as the sum of the FolkRank of t_b , when searching with t_a , plus the FolkRank of t_a , when searching for t_b (cf. Section 3). We measure the success of the ranking strategies by means of the precision within the top k ranked tag pairs, i.e. the percentage of tag pairs within the top k that are— from our perspective and without considering the tag usage but just the words themselves—truly related to each other.

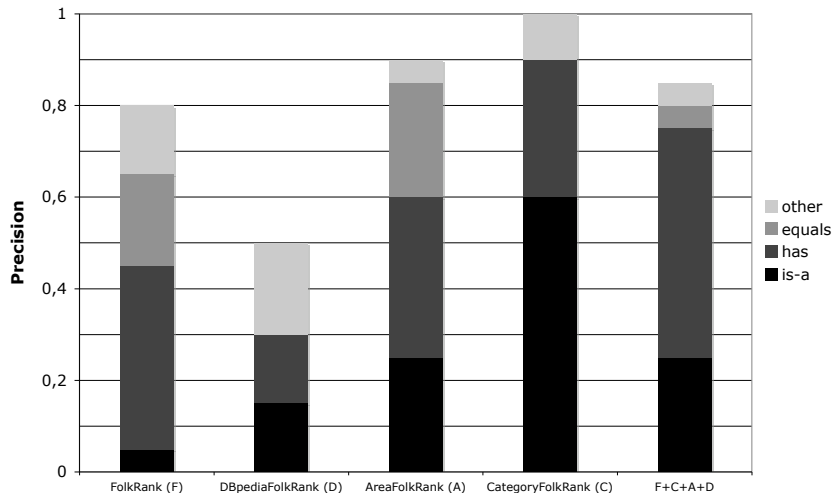


Fig. 5. Precision of tag pairs successfully classified as *related* as well as the distribution of relation types for the different strategies within the top 20.

Results Figure 5 shows the average performance of the different ranking strategies for identifying tag relations. When applying the ranking strategies as described above, the CategoryFolkRank is the most successful strategy as all tag pairs within the top 20 of the ranking are truly related to each other. FolkRank and the URI-based FolkRank (DBpediaFolkRank) perform worse as 20% and respectively 50% of the returned tags are really related. The bad performance of the URI-based FolkRank might be explained by the URI mapping: the actual FolkRank is computed on basis of the URIs to which the tags were mapped and the ranking of tag pairs is constructed by mapping the URIs back to tags. The latter step causes that errors are propagated, for example, the non-related tag pairs $(pink\ rose, bird)$, $(red\ rose, bird)$, $(white\ rose, bird)$, and $(rose, bird)$, which are each mapped to $(dbpedia:Rose, dbpedia:Bird)$, are each wrongly ranked at the same high level whereas FolkRank just ranks $(rose, bird)$ at a high rank. AreaFolkRank as well as the combined approach (F+C+A+D) perform very well with a precision of 90% and 85% respectively. The combined strategy is negatively influenced by DBpediaFolkRank: omitting DBpediaFolkRank (F+C+A) leads to a precision of 95%.

Further, we examined the kind of tag pairs that were identified by the algorithms. Therefore, we considered the following types of relations.

- *is-a*: This relation refers to a sub/super-class or instance-of relation. For example: church is a building, horse is an animal, etc.
- *has*: Denotes that an entity consists of (is part of) or contains (is contained in) another entity. For example: car has wheel, sky contains cloud, etc.
- *equal*: Relation stating that two tags refer to the same thing (cf., *same-As* relation in OWL). For example: spinne equals spider, marienkäfer equals ladybug (German/English).
- *other*: Other types of relations such as instances that belong to the same class or instances that care part of the same concept. For example: batman and superman, rooftop and façade, etc.

Figure 5 summarizes the types of relations that exist among the tag pairs returned within the top 20 ranking of the different ranking strategies. For example, regarding CategoryFolkRank 60% of the tag pairs can be typed as *is-a* relation, 30% as *has* relations and 10% denote other types of relations while tags that refer to *equal* concepts do not appear at the top of the ranking. The high frequency of *is-a* relations can possibly explained by the usage style of categories (cf. Section 4.1), i.e. categories are more broader and were often used to classify tags. The AreaFolkRank identifies all types of relations and is the strategy that identifies the most tag pairs that refer to *equal* concepts (25% of the tag pairs are typed as *equals*): different tags assigned to the same area of an image are thus likely to refer to the same concept.

In summary, the strategies that exploit the area and category context of tag assignments perform better (12.5% and 25% respectively) than the strategy that just considers the traditional tag assignments to identify related tags. Further, the exploitation of categories most often induces super/sub-class relations between tags while the area context has potential to infer *sameAs* relations.

Conclusion

In this paper we proposed an approach for multifaceted tagging and a corresponding model for embedding context into folksonomies. We implemented the model into the TagMe! system, a tagging and exploration interface for Flickr pictures that introduces three types of context: spatial information, categories and semantic meanings (URIs). We further developed and evaluated different search and ranking algorithms that exploit the contextual facets of the folksonomy. The algorithm that considered the different contextual facets significantly improved the precision of the baseline FolkRank algorithm by 20.0% and 21.4% with respect to the precision of the search result rankings. Relying on these results we demonstrated that contextual information can significantly improve search. Moreover, we were able to apply the ranking strategies for identifying semantic relations among tags. We showed that the exploitation of spatial information and categories attached to tag assignments improves the precision of inferring related tags by 12.5% and 25% respectively.

Acknowledgments This work is partially sponsored by the EU FP7 project GRAPPLE (<http://www.grapple-project.org/>).

References

1. Golder, S.A., Huberman, B.A.: The structure of collaborative tagging systems. The Computing Research Repository (CoRR) **abs/cs/0508082** (2005)
2. Bischoff, K., Firan, C., Paiu, R., Nejdil, W.: Can All Tags Be Used for Search? In: Proc. of Conf. on Information and Knowledge Management 2008, New York, NY, USA, ACM (November 2008) 193–202
3. Passant, A., Laublet, P.: Meaning Of A Tag: A collaborative approach to bridge the gap between tagging and Linked Data. In: Proceedings of the WWW 2008 Workshop Linked Data on the Web (LDOW2008), Beijing, China. (Apr 2008)
4. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: DBpedia: A Nucleus for a Web of Open Data. In et al., A., ed.: The Semantic Web, 6th International Semantic Web Conference (ISWC), 2nd Asian Semantic Web Conference (ASWC). (November 2007) 715–728
5. Hotho, A., Jäschke, R., Schmitz, C., Stumme, G.: BibSonomy: A Social Bookmark and Publication Sharing System. In: Proc. First Conceptual Structures Tool Interoperability Workshop, Aalborg (2006) 87–102
6. Abel, F., Henze, N., Krause, D., Kriesell, M.: On the effect of group structures on ranking strategies in folksonomies. In Baeza-Yates, R., King, I., eds.: Weaving Services and People on the World Wide Web, Springer (July 2009) 275–300
7. Abel, F., Kawase, R., Krause, D., Siehndel, P.: Multi-faceted Tagging in TagMe! In: 8th International Semantic Web Conference (ISWC2009). (October 2009)
8. Russell, B.C., Torralba, A.B., Murphy, K.P., Freeman, W.T.: LabelMe: A Database and Web-based tool for Image Annotation. International Journal of Computer Vision **77**(1-3) (2008) 157–173
9. Berners-Lee, T.: Linked Data - design issues. Technical report, W3C (May 2007) <http://www.w3.org/DesignIssues/LinkedData.html>.
10. Hotho, A., Jäschke, R., Schmitz, C., Stumme, G.: Information retrieval in folksonomies: Search and ranking. In: Proc. of the 3rd European Semantic Web Conference. Volume 4011 of LNCS., Budva, Montenegro, Springer (June 2006) 411–426
11. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank Citation Ranking: Bringing Order to the Web. Technical report, Stanford Digital Library Technologies Project (1998)
12. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: Dbpedia - a crystallization point for the web of data. Web Semantics: Science, Services and Agents on the World Wide Web (July 2009)
13. Abel, F., Kawase, R., Krause, D., Siehndel, P., Ullmann, N.: The Art of Multi-faceted Tagging – inteweaving spatial annotations, categories, meaningful URIs and tags. In: 6th Int. Conf. on Web Information Systems and Technologies (WEBIST), Valencia, Spain (April 2010)
14. Halpin, H., Robu, V., Shepard, H.: The Dynamics and Semantics of Collaborative Tagging. In: Proc. of the 1st Semantic Authoring and Annotation Workshop (SAAW2006). (November 2006)
15. Mika, P.: Ontologies Are Us: A unified model of social networks and semantics. In: Proc. Int. Semantic Web Conference (ISWC 2005). (November 2005) 522–536
16. Schmitz, C., Hotho, A., Jäschke, R., Stumme, G.: Mining Association Rules in Folksonomies. In: Data Science and Classification. Springer (2006) 261–270
17. Schmitz, P.: Inducing Ontology from Flickr Tags. In: Proceedings of the Workshop on Collaborative Tagging co-located with 15th Int. World Wide Web Conference (WWW '06), Edinburgh, Scotland (May 2006)