RichVSM: enRiched Vector Space Models for Folksonomies

Rabeeh Abbasi
abbasi@uni-koblenz.de

Steffen Staab
staab@uni-koblenz.de

ISWeb - Information Systems and Semantic Web Research Group
Institute for Computer Science, University of Koblenz - Landau
Universitätsstraße 1, 56070 Koblenz, Germany

ABSTRACT

People share millions of resources (photos, bookmarks, videos, etc.) in Folksonomies (like Flickr, Delicious, Youtube, etc.). To access and share resources, they add keywords called tags to the resources. As the tags are freely chosen keywords, it might not be possible for users to tag their resources with all the relevant tags. As a result, many resources lack sufficient number of relevant tags. The lack of relevant tags results into sparseness of data, and this sparseness of data makes many relevant resources unsearchable against user queries.

In this paper, we explore two dimensions of semantic relationships between tags, based on the context and the distribution of tags. We exploit semantic relationships between tags to reduce sparseness in Folksonomies and propose different enriched vector space models. We also propose a vector space model *Best of Breed* which utilizes appropriate enrichment method based on the type of the query. We evaluate the proposed methods on a large dataset of 27 million resources, 92 thousand tags and 94 million tag assignments. Experimental results show that the enriched vector space models help in improving search, especially for the rare queries which have few relevant resources in the sparse data.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Retrieval models

General Terms

Algorithms, Experimentation, Performance

Keywords

Folksonomies, Information Retrieval, Search, Smoothing, Sparseness Reduction, Tagging, Vector Space Model, Vector Space Models

1. INTRODUCTION

Folksonomies or *Collaborative Tagging Systems* provide a good way to share and store resources. Users can share their resources with other users in a folksonomy. Users can share images (Flickr), bookmarks (del.icio.us, bibsonomy), citations (citeULike, bibsonomy), and many other types of resources in different folksonomies. Users can also add *tags* (keywords) to the resources. Later on, resources can be searched and retrieved using these tags. For example a user can upload a funny image of 1970s and add the tags *funny* and *seventies* to it. Users can search this image by giving the tags attached to it.

The users may tag resources with keywords of their choice. They might not add many relevant tags to the resources. This results into sparseness of data and makes it difficult to search relevant resources. Especially when there are only few resources in the folksonomy relevant for a combination of query tags. For example if a user is searching for funny pictures of 1970s using the tags *funny* and *seventies*. He will get only the images tagged with *funny* and *seventies* or ones that contain one of the two tags. The user will be unable to get resources that are tagged with *1970s* and *funny* but are not tagged with *seventies*, although the resources tagged with *1970s* and *funny* might be of interest to him. Our hypothesis is that there are many resources in folksonomies which are not searchable because they do not contain most of the relevant tags.

In this paper we show that one can find meaningful relationships between tags and then use these relationships to reduce the sparseness in folksonomies. We find the relationships between tags based on two dimensions, first the context of the tags and second the distribution of tags. We consider two types of tag contexts, the resource context (which resources are assigned a particular tag), and the social context (which users have used a particular tag). The resource context of tags helps in finding tags which are mostly used in similar kind of resources, whereas the social context finds broad relationships between tags based on the users' interests (represented by the tags they use). We also exploit two kinds of tag distributions, 1) similar tags and 2) generalized tags. We find relationships between similar tags by using the existing *cosine* similarity measure and propose a modified overlap coefficient to exploit generalization relationships between tags. We hypothesize that the statistic description of resources that use common tags exhibits different behavior than the statistic description of resources with uncommon tags. To test this hypothesis, we split the queried tags into three sets; having 1-10 search results, 11-50 and more

than 50 search results respectively and perform experiments on these sets of queries. We also propose a method *Best of Breed* (BB), which selects appropriate enrichment model based on the number of relevant resources related to the queried tags. Experimental results based on a large scale evaluation (150 queries evaluated on a dataset of ˜27 Million resources by 18 expert users) show that the enrichment of existing data by exploiting semantic relationships among tags helps in improving the search results, particularly for the queries which have a few relevant resources in the original data.

The remainder of this paper is structured as follows, in section 2 we define the basic structure of folksonomies. Section 3 describes the methods used for enriching the vector space model of folksonomies. Section 4 contains the description of evaluation setup and dataset used for evaluation. Section 5 describes the results. Section 6 presents the related work and in section 7 we conclude our research.

2. FOUNDATIONS

We start with the Vector Space Model representation of Folksonomies.

2.1 Formal Representation

Mainly, a folksonomy consists of three elements Users, Tags, and Resources and relationships between these three elements. Users can use tags, tags are associated to resources, and resources are associated to users. In this paper, we only consider relationships between tags and users, and relationships between tags and resources. More specifically, which tags are used by which users and which tags are associated to which resources. We represent these relationships in the form of two matrices U and R. Let us define the relationships between tags and users using the matrix U as follow

$$U = [u_{ij}] \tag{1}$$

where u_{ij} is equal to 1, if user j has used the tag i, otherwise u_{ij} is equal to 0. Each row vector u_{i*}[1] of the matrix U represents a tag vector, whose non-zero elements represent the users that have used this tag. Each column vector u_{*j} of the matrix U represents the users. The matrix U is a sparse matrix, but denser than the R matrix, because the column vector u_{*j} has non-zero value for all the tags which are used by the user j and it is more likely that the set of tags used by a user is bigger than the tags used in a resource.

Similar to the matrix U, we represent the relationships between tags and resources using the matrix R as follows

$$R = [r_{ij}] \tag{2}$$

where r_{ij} denotes how many times the tag i appeared with the resource j. Each row r_{i*} of the matrix R is a tag vector, whose non-zero elements represent how many times these elements (resources) have been annotated with this tag (i). Each column vector r_{*j} of the matrix R represents a resource, which has non-zero values for the associated tags, and zero for the tags it does not use. As there are millions of tags and resources, but each resource is assigned only a few tags, therefore the matrix R is a very sparse matrix.

In some folksonomies (called *Narrow Folksonomies* [17] like Flickr), a resource cannot be tagged with a tag more than once, while in other folksonomies (called *Broad Folksonomies* [17]) a single resource can be tagged with a tag multiple times (for example from different users). In case of *Narrow Folksonomies*, the value of r_{ij} will always be equal to zero or one.

[1]We will denote row vectors as $i*$ and column vectors as $*j$ in the subscript of a vector throughout the text

2.2 Querying and Retrieval

To retrieve resources from a vector space model by a query, we represent the tags in a query as a resource vector. If we represent the query vector as q, then similarity between the query vector q and the resource vector r can be computed using the cosine similarity as follows

$$cosine(q, r) = \frac{q \cdot r}{\|q\| \cdot \|r\|} \tag{3}$$

If the query vector q is equal to the resource vector r, then their cosine similarity will be equal to 1 and if vectors q and r have no common term, then their cosine similarity will be equal to 0.

3. ENRICHING THE VECTOR SPACE

The standard vector space (eq. 2) of a folksonomy is very sparse. Many resources are not tagged with relevant tags. For example if a picture of a broken radius[2] is tagged with *broken* and *radius* but not with *fracture*, and a user searches for the images using tags *fracture* and *radius*, he will not discover the picture because this picture is not tagged with *fracture*. However if we associate the tag *broken* with the tag *fracture*, then it would be possible to retrieve the resources which are tagged with *radius* and *broken* against the query *radius* and *fracture*. We enrich the standard vector space model (eq 2) by associating relevant tags to the resources.

We consider several methods to enrich the vector space model. The basic idea behind all these methods is to find semantic relationships between tags and then enrich the standard vector space using these semantic relationships.

3.1 Semantic Relationships Between Tags

We define two dimensions of semantic relationships between tags and propose enriched vector space models based on these dimensions. Figure 1 shows the two dimensions of tags relationships, x-axis shows the context of the tag, and y-axis represents the distribution of semantic relationships between tags. SR_G, SR_C, and SU_C are the different semantic relationship matrices created based on the two dimensions. These matrices are described in section 3.1.2. Following sections describe the dimensions in detail

3.1.1 Tag Distribution

We define the following three types of tag distributions

Similar Tags: are like synonyms or equally related to each other. If two tags appear together in most of the resources, then they are considered to be similar. For example the tags *Brazil* and *Brasil* appear often together, therefore they are considered to be similar tags. Similarly *Notre* and *Dame* can also be considered similar tags. To find similar tags we use the cosine measure defined in equation 3. By using cosine measure, we identify symmetric relationships between tags. [5] gives good qualitative insights of cosine similarity and other methods for identifying semantic relationships between tags.

[2]The bone of the forearm on the thumb side

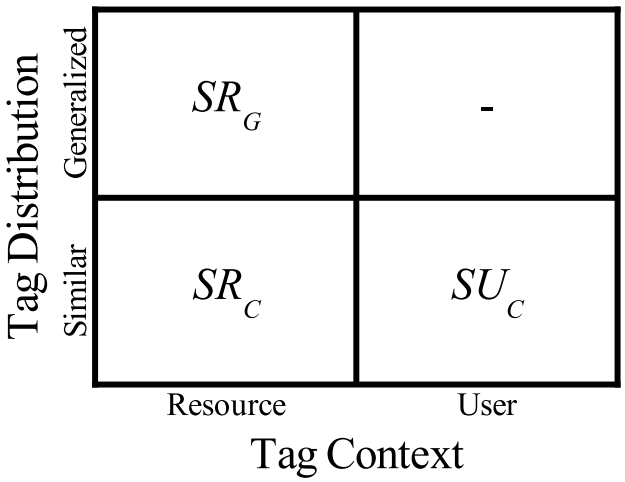

Figure 1: Dimensions of Semantic Relationship between Tags. X-Axis represents the context of the tag and Y-Axis represents the distribution of the tag. User context with generalized tag distribution is not considered.

Generalized Tags: A generalized tag has a parent relation with another tag. If a tag t_1 is assigned to more resources than the tag t_2, and all or most of the resources having tag t_2 also have the tag t_1, then the tag t_1 generalizes the tag t_2. For example, the tag *Paris* is assigned to more resources than the tag *Eiffel Tower*, and most of the resources having the tag *Eiffel Tower* also have the tag *Paris*; therefore the tag *Paris* generalizes the tag *Eiffel Tower*.

To enrich the vector space model, we hypothesize that it is useful to enrich tags with generalized tags, instead of enriching tags with specialized tags, because by adding the generalized tags to existing resources, we do not add any incorrect information. For example, adding the generalizing tag *Spain* (of the tags *Madrid* and *Barcelona*) to all the resources having the tags *Madrid* and *Barcelona*. But it would not be meaningful, if we add the tag *Madrid* (a specialized tag of the tag *Spain*) to all the resources tagged with *Spain*.

To find the generalized tags, we define a modified overlap coefficient for two tags t_1 and t_2 as follows

$$gen(r_{1*}, r_{2*}) = \begin{cases} \dfrac{|r_{1*} \cap r_{2*}|}{|r_{1*}|} & \text{if } |r_{1*}| \leq |r_{2*}|; \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

where r_{1*} and r_{2*} represent the tag vectors of the tags t_1 and t_2. $|r_{1*} \cap r_{2*}|$ represents the number of times t_1 and t_2 appear together (for narrow folksonomies $|r_{1*} \cap r_{2*}| = r_{1*} \cdot r_{2*}$), $|r_{1*}|$ represents the number of times t_1 is used and $|r_{2*}|$ represents the number of times t_2 is used. If t_1 is a tag that appears only with the tag t_2, then the value of $gen(r_{1*}, r_{2*})$ will be 1, it does not matter how many times the tag t_2 is used with other tags. If $|r_{2*}| \geq |r_{1*}|$, then t_2 is considered as a generalized tag for t_1 and if $|r_{2*}| < |r_{1*}|$, then t_2 is a specialized tag of the tag t_1 and we do not consider such relationship and set the similarity coefficient $gen(r_{1*}, r_{2*}) = 0$.

Specialized Tags: are opposite to generalized tags. If a tag t_1 generalizes the tag t_2, then the t_2 is a specialized tag of the tag t_1. In the example given for generalized tags, the tag *Eiffel Tower* is the specialized tag for the tag *Paris*. In this paper we enrich the vector space model using similar and generalized tags only.

3.1.2 Contexts of the Tags

In the following sections, we describe the two different kinds of contexts of the tags to enrich the vector space models.

Resource Context: The resource context of t_1 consists of all the resources that have the tag t_1. We can formally represent the resource context of the tag t_1 as r_{1*}. To discover the semantically similar tags t_1 and t_2 based on the resource context, we compute the cosine similarity between the two tag vectors using Equation 3. We build a similarity matrix SR_C based on the cosine similarity between the tags using resource context as follows

$$SR_C = [cosine(r_{i*}, r_{j*})] \quad (5)$$

where r_{i*} and r_{j*} are the tag vectors of the tags t_i and t_j of the vector space R respectively.

We also exploit the resource context to discover the generalized tags. We use the following equation to define the matrix SR_G to identify generalized relationships between the tags using the resource context

$$SR_G = [gen(r_{i*}, r_{j*})]^T \quad (6)$$

where r_{i*} and r_{j*} are the tag vectors of the tags t_i and t_j respectively of the vector space R.

Table 1 shows some examples[3] of the tags having the tag *hibiscus*[4] as a generalized tag. First five tags in the table 1 are the translations of hibiscus in different languages. *Rosemallow* and *gumamela* are different names of hibiscus. It is important to note that although the tags *rosemallow*, and *gumamela* are semantically similar tags to the tag *hibiscus*, but they are used more rarely than the tag *hibiscus* and computing cosine similarity between these tags and the tag *hibiscus* will give a very small similarity value due to the normalization in the cosine measure. *Rose of Sharon* is specie of hibiscus and is correctly identified.

Finding generalized tags using the equation 6 might also incorrectly identify a generalized tag as a specialized tag, if the generalized tag is assigned to fewer resources than the specialized tag. For example, the specie *malvaceae* of plants comes into higher hierarchy than *hibiscus*, but because it is assigned to fewer resources than the resources assigned the tag *hibiscus*, therefore *hibiscus* is considered as a generalized tag for the tag *malvaceae* (which is incorrect). Such kind of anomalies can be discovered by using external data sources, which is out of scope of this paper.

Social Context: We hypothesize that tags represent the interests of the users. For example if we consider a person who normally takes pictures of his interest, he add tags to describe the images he takes, therefore these tags show the interests of the user. The social context of a tag means the users which have shared the tag with other users, we can use this social context for enriching the vector space models. For example if many users share the tags *Brazil* and *Brasil*, but they do not necessarily use these tags in their resources

[3]These relationships are discovered based the images and their tags uploaded to Flickr between Jan 2004 and Dec 2005. Details in section 4.1.

[4]http://en.wikipedia.org/wiki/Hibiscus

Table 1: Semantic relationships of tags having the generalized tag *hibiscus* by exploiting the resource context and generalized tag distribution SR_G (see Eq. 6).

Tag	Generalized Tags
hibiscus	hibiscus(1.00), flower(0.61),flowers(0.25),red(0.17),macro(0.15)
hibiskus	hibiskus(1.00), hibiscus(0.67),flower(0.47),blume(0.33),garten(0.29)
ibisco	ibisco(1.00), flower(0.65),hibiscus(0.52),flor(0.43),red(0.30)
ibiscus	ibiscus(1.00), flower(0.55),hibiscus(0.45),nature(0.41),flowers(0.41)
hibisco	hibisco(1.00), flor(0.66),flower(0.65),hibiscus(0.39),macro(0.21)
rosemallow	rosemallow(1.00), hibiscus(0.52),flower(0.48),malvaceae(0.38),garden(0.24)
gumamela	gumamela(1.00), flower(0.58),philippines(0.38),hibiscus(0.35),red(0.23)
roseofsharon	roseofsharon(1.00), flower(0.66),macro(0.28),flowers(0.28),hibiscus(0.21)
malvaceae	malvaceae(1.00), flower(0.73),hibiscus(0.62),flowers(0.28),macro(0.22)

together, it would be still possible to find relationships between these tags by considering the number of users that shared both of these tags. Initial observations have showed that the social context of the tags is not appropriate for discovering generalized tags. However social context helps in finding similar tags. For this reason, we only exploit social context of the tag to find similar tags.

To define the semantic relationships between similar tags t_i and t_j based on social context, we compute the semantic relationship matrix using cosine similarity as follows

$$SU_C = [cosine(u_{i*}, u_{j*})] \quad (7)$$

where u_{i*} and u_{j*} are the tag vectors of the tags t_i and t_j respectively of the vector space U.

3.1.3 Filtering the Semantic Relationship Matrices

The semantic relationship matrices defined in section 3.1.2 might have a lot of weak relationships in some cases and they might also have too many relationships in other cases. To make the semantic relationship matrices more accurate, we prune these matrices using the following two approaches.

Pruning Weak Semantic Relationships: There could be many weak semantic relationships between tags, which, when used for enriching the vector space model, might add noise to the enriched vector space model. The value of semantic relationship computed between two tags using eq 3 or eq 4 can be between 0 and 1. We ignore all the values which are less than .1, hence pruning the weak relationships from the semantic relationship matrices.

Confining Enrichment: To avoid over-enriching the vector space model, we limit the number of semantic tag relationships for each tag to five. In each of the semantic relationship matrix (SR_C, SR_G, and SU_C), each tag is associated with the top five most semantically related tags. As a result, a maximum of five new tag associations per tag and per resource will be possible in the enriched vector space model.

3.1.4 Few Examples of Semantic Tag Relationships

Table 2 shows some examples of semantically related tags using the methods descried in section 3.1.2. The different types of relationships in different dimensions can be observed from the examples. The semantically similar tags based on resource context (SR_C) gives more close relationships to the tags, for example, the tag *brick* is similar to the tag *wall*. The tags *bromelia* is similar to the tags *airplant*, *bromeliad*, and *tillandsia* which belong to the same family of flowers. It is also interesting to note that simple associations like *bromelia* and *flower* are not identified using resource context of similar tags. However such a generalized relationship is obvious in generalized relationship based on resource context (SR_G). Other such generalized relationships are also obvious from other examples like the tag *brick* is associated to a more general tag *building*, *bromelia* to *flower* and *nature* etc. If we consider the similar tags based on the social context, we observe a wide range of tag associations. For example, we find the tags *tibouchina* and *strelitzia* associated to the tag *bromelia* which are different kinds of flower plants. Social context also associate the tags *seventies* to the tags *sixties*, *eighties*, and *forties*, which shows the interest of users in old pictures.

In next section, we exploit the semantically related tags for enriching the vector space model of a folksonomy.

3.2 Enriched Vector Space Models

After discovering the semantic relatedness of the tags and representing them in the form of one of the matrices, i.e. SR_C, SR_G, and SU_C, we can exploit these matrices to enrich the original vector space model R. We transfer the original vector space into enriched vector space by multiplying the semantic relatedness matrix to the original vector space. After the transformation of original vector space into the enriched vector space, the missing relevant tags identified by one of the semantic relatedness matrices are assigned to the resources in the original vector space. If table 3 shows the original vector space model, we can discover the semantically related tags using one of the equations 5, 6, or 7 (with exception of equation 7, where we need the U (eq. 1) vector space).

The matrix SR_G computed using resource contexts of the tags and modified overlap coefficient (eq 6) is shown in table 4 (the semantic relatedness values are computed using the dataset described in section 4.1). After transforming the original vector space R (table 3 into enriched vector space using the formula $SR_G \times R$, we get the enriched vector space shown in table 5.

We can observe that some of the missing relevant tags are now added to the enriched vector space model. For example, in table 4, the tags *1970s* and *70s* are semantically related to the tag *seventies* and are assigned to the resources r_{*1} and r_{*2} in the enriched vector space (table 5). Similarly the tag *broken* is assigned to the resource r_{*4}, because it

Table 2: Semantically related tags based on different tag distributions and contexts.

Tag	SR_C	SR_G	SU_C	Tag	SR_C	SR_G	SU_C
brick	brick(1.00) wall(0.11)	brick(1.00) wall(0.19) building(0.13) red(0.11)	brick(1.00) wall(0.37) fence(0.36) window(0.36) rust(0.35)	pub	pub(1.00) crawl(0.17)	pub(1.00) beer(0.11) bar(0.11)	pub(1.00) beer(0.25) bar(0.24) sign(0.22) London(0.22)
bromelia	bromelia(1.00) airplant(0.32) bromeliad(0.17) tillandsia(0.15)	bromelia(1.00) bromeliad(0.35) tillandsia(0.30) flower(0.27) nature(0.12)	bromelia(1.00) lirio(0.18) tibouchina(0.15) soneca(0.15) strelitzia(0.15)	seventies	seventies(1.00) 70s(0.16) entertainers(0.14) sixties(0.13)	seventies(1.00) 70s(0.33) party(0.18) 1970s(0.11)	seventies(1.00) sixties(0.19) 70s(0.17) eighties(0.16) forties(0.13)
designs	designs(1.00) desktops(0.29) wallpapers(0.22) backgrounds(0.21)	designs(1.00) wallpapers(0.28) gallery(0.25) backgrounds(0.25) Chris(0.16)	designs(1.00)	Spain	Spain(1.00) Espana(0.37) Barcelona(0.25) Andalucia(0.20) Madrid(0.19)	Spain(1.00) 2005(0.13)	Spain(1.00) Barcelona(0.36) Espana(0.32) Madrid(0.31) Gaudi(0.27)
Madrid	Madrid(1.00) Spain(0.19) Zarzuela(0.13) Hipodromo(0.13) Carreras(0.12)	Madrid(1.00) Spain(0.38)	Madrid(1.00) Spain(0.31) Espana(0.24) Segovia(0.22) Toledo(0.21)	style	style(1.00) crave(0.14) Arian(0.13) fashion(0.11) Persians(0.10)	style(1.00) fashion(0.26) beauty(0.11)	style(1.00) fashion(0.16) hair(0.13) woman(0.12) man(0.12)

is semantically related to the tag *fracture* in the semantic relatedness matrix.

Now we define the different enriched vector space models exploiting different dimensions of semantic relatedness between tags. We define the enriched vector space model TR_C which exploits similar tags based on the resource context of the tags as follows

$$TR_C = SR_C \times R \tag{8}$$

Similarly, we define the enriched vector space model TR_G based on generalized tags and exploiting resource context as follows

$$TR_G = SR_G \times R \tag{9}$$

To exploit similar tags based on social context of the tags, we define the enriched vector space model TU_C using the following equation

$$TU_C = SU_C \times R \tag{10}$$

Ranking Relevant Resources against a Query: We use a two step procedure to rank relevant resources against a query q. If there are N resources in the vector space model, we first compute a common terms vector C as follows

$$C_{i=1..N} = |q \cap \hat{r}_{*i}| \tag{11}$$

where $\hat{r}_{*i}$ is the i^{th} resource in a vector space model and $|q \cap \hat{r}_{*i}|$ is the number of common tags between the query and the resource $\hat{r}_{*i}$. We rank the resources retrieved against the query q in descending order of the values of the vector C. In case of a tie, when relevance of two or more resources against a query have the same value in the common terms vector C, we resolve the tie using cosine similarity D_i (eq 12)

$$D_i = cosine(q, \hat{r}_{*i}) \tag{12}$$

In next section, we describe experiments using different vector space models.

Table 3: Sample original vector space model.

	r_{*1}	r_{*2}	r_{*3}	r_{*4}
1970s	**1**	0	0	0
70s	0	0	0	0
broken	0	0	**1**	0
fracture	0	0	0	**1**
funny	**1**	**1**	0	0
radius	0	0	**1**	**1**
seventies	**1**	**1**	0	0

Table 4: Sample semantic relatedness matrix computed using resource context and generalized tag distribution SR-G (See Eq. 6).

	1970s	*70s*	*broken*	*fracture*	*funny*	*radius*	*seventies*
1970s	**1**	0	0	0	0	0	**0.1**
70s	**0.1**	**1**	0	0	0	0	**0.32**
broken	0	0	**1**	**0.4**	0	0	0
fracture	0	0	0	**1**	0	0	0
funny	0	0	0	0	**1**	0	0
radius	0	0	0	0	0	**1**	0
seventies	0	0	0	0	0	0	**1**

Table 5: Sample enriched vector space model.

	r_{*1}	r_{*2}	r_{*3}	r_{*4}
1970s	**1.1**	**0.1**	0	0
70s	**0.42**	**0.32**	0	0
broken	**0**	**0**	**1**	**0.4**
fracture	0	0	0	**1**
funny	**1**	**1**	0	0
radius	0	0	**1**	**1**
seventies	**1**	**1**	0	0

4. DATASET AND EVALUATION

In this section we describe the dataset we used for our experiments followed by the evaluation method.

4.1 Data set

Our large-scale dataset[5] was obtained by systematically crawling the Flickr system during 2006 and 2007. The target of the crawling activity was the core elements, namely users, tags, resources and tag assignments. The statistics of the crawled dataset are summarized in table 6.

users	tags	resources	tag assignm.
319,686	1,607,879	28,153,045	112,900,000

Table 6: Flickr dataset statistics

We applied the following strategy to crawl the Flickr dataset. First, we started a tag centric crawl of all photos that were uploaded between January 2004 and December 2005 and that were still present in Flickr as of June 2007. For this purpose, we initialized a list of known tags with the tag assignments of a random set of photos uploaded in 2004 and 2005. After that, for every known tag we started crawling all photos uploaded between January 2004 and December 2005 and further updated the list of known tags. We stopped the process after we reached the end of the list.

We filtered our dataset by removing those tags which were used by less than 10 users. Those users and resources were also removed from the dataset which did not use any tag. In the final dataset, we had data of about 27M photos, 0.3M users, and 92K tags. The exact statistics of the dataset are shown in table 7. We did all our experiments on this dataset.

users	tags	resources	tag assignm.
317,260	92,460	26,801,921	94,499,112

Table 7: Flickr filtered dataset statistics

4.2 Evaluation Method

For evaluation, we used the AOL query log (details in [14]) which originally contained 20M queries from 650K users during three months from March to May 2006. Out of these 20M queries, we selected queries having 2 to 5 words for which the user had clicked on a link to the Flickr website. We split the queries into three sets, each set having 1 to 10, 11 to 50, and more than 50 exact matches (resources having all the queried tags) in the original vector space model. We randomly selected 50 queries from each of these three sets, resulting into 150 total queries for the evaluation.

We evaluated our approach using human based evaluation. The results were evaluated by 18 expert users (mostly PhD students) who were well familiar with search and image search. Each user was shown a search result page similar to the screenshot shown in Figure 2. The query was shown at the top of each evaluation page with images retrieved as a result. The title of the image was shown at the top of the image, tags on the right side, and evaluation options at the bottom of each image. Every user was given a set of queries and results obtained using different vector space models described in section 3.2. Users were unaware of the method used for creating the search result page. Users were asked to mark an image as very relevant or relevant if the image matches the query, mark as don't know if they are not sure about the image, irrelevant or very irrelevant if the image does not match the given query. Queries were randomly distributed among users. The images marked as relevant or very relevant were considered as relevant and others as irrelevant in final evaluation.

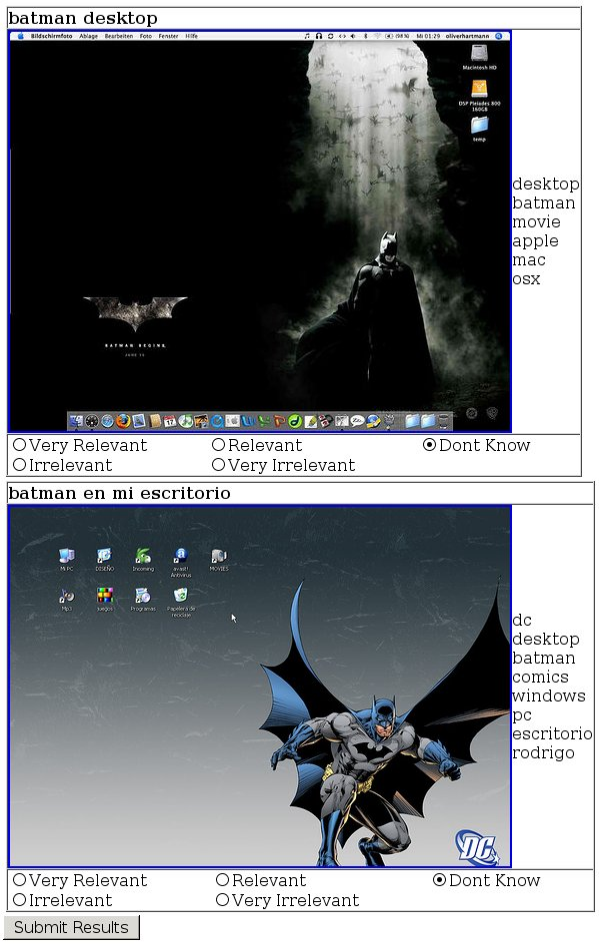

Figure 2: A screenshot of an evaluation page.

In addition to the vector space models defined in section 3.2, we also defined three other vector space models *Semi Random* (SEMI_RAND) *Random* (RAND) as baselines and *Best of Breed* (BB) as the best model against a particular type of query. For the *SEMI_RAND* vector space model, the semantic relationship matrix was created by associating a maximum for five random tags to each of the tags. The similarity of each tag to itself was explicitly set to 1 (maximum) in the *SEMI_RAND* vector space model. Whereas the semantic relationship matrix in the *RAND* vector space model consisted of totally random values (multiplying the original vector space (R) with a random matrix with a maximum of five random values per tag). The *Best of Breed* (BB) model

[5]The reference data set used for this evaluation is available at `http://www.uni-koblenz.de/~goerlitz/datasets/tas_flickr.zip`

is created based on the type of query. For queries having 1 to 10 or 11 to 50 search results (exact matches) in original data, we use the TU_C vector space model and for queries having more than 50 search results, we use TR_C vector space model. Based on experimental results, we show that it is helpful to use a particular type of vector space model for a particular kind of query for achieving better search results.

5. RESULTS AND DISCUSSION

For evaluating the enriched vector space models, we computed precision at 5, precision at 10, precision at 15, and precision at 20 for each of the methods and each query set. In each of the results, the original vector space (R) in the results, show the average precisions obtained without enriching the vector space model, *TR-C*, *TR-G*, *TU-C* shows the average precisions obtained using the enriched vector space models TR_C, TR_G, and TU_C respectively, SEMI_RAND and RAND shows the baseline results obtained using SEMI_RAND and RAND (described in section 4.2) vector space models respectively and *BB* represents the *Best of Breed* model (described in section 4.2). Please note that the results for RAND vector space model are not visible in any figure due to very low precision values and results in all the figures are shown in this order *R, TR-C, TR-G, TU-C, SEMI_RAND, RAND, and BB.*

The main goal of this research is to develop a method which enables the users to find rare resources by enriching the vector space models. For the (rare) queries which had 1 to 10 exact matches in the original vector space model, we achieve great improvement in results using enriched vector space models, which shows the significance of our proposed models. Figure 3 compares the performance of all methods for queries having varying number of relevant resources. The x-axis represents the types of the queries. We can observe that the enriched vector space models, particularly the best selection *Best of Breed* model, perform better than the baselines and the original vector space model. For rare queries (having 1 to 10 relevant resources), we achieve an improvement of 35%. The improvement decreases for the queries having many relevant resources in the original vector space model (7% for 11-50 resources and 1.5% for more than 50 relevant resources). The fact for the decrease in improvement is the reason that there are sufficient relevant resources in the original vector space model to be ranked in top 20 results. If we consider the results of all the queries together, we still get an improvement of 12% using *Best of Breed* model.

Figure 4 shows the average precisions achieved for the queries which had 1 to 10 exact matches (resources associated with all the queried tags) in the original vector space. The figures display the results in the order from left to right (R, TR_C, TR_G, TU_C, SEMI_RAND and RAND). The results on each of the evaluation page were ranked using the ranking method described in section 3.2. First the resources having exact matches are displayed, afterwards the resources having one tag less than the total number of queried tags and so on. We still achieve .40 to .45 precision @ 15 and 20 for original vector space model, reason is that retrieved resources are still associated with some of the queried tags, hence making these resources relevant. We observe a significant improvement in the precision at all levels using enriched vector space models, specially using the vector space model based on semantically similar tags using social context TU_C,

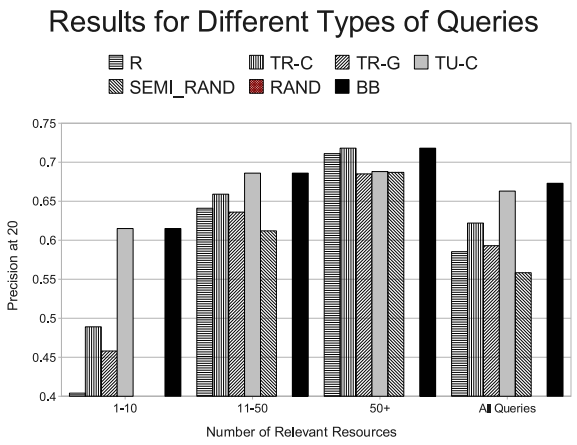

Figure 3: Comparison of methods for different types of queries. X-Axis shows the number of relevant resources for the evaluated queries and Y-Axis shows the results for precision at 20. The results are most significantly visible for rare queries (i.e. having 1 to 10 relevant resources. Results for RAND are not visible for any type of query.

which is also used in *Best of Breed* for queries having 1 to 10 exact matches. The reason for improvement in precision is the retrieval of those resources which do not contain the queried tag(s) exactly, but have some relevant tag(s). If we consider arbitrary tag relationships (SEMI_RAND), then we get even worse results than the original vector space model. That suggests that the tags must be semantically related to improve the resource retrieval. Due to very low precision of the RAND vector space model, its results are not visible.

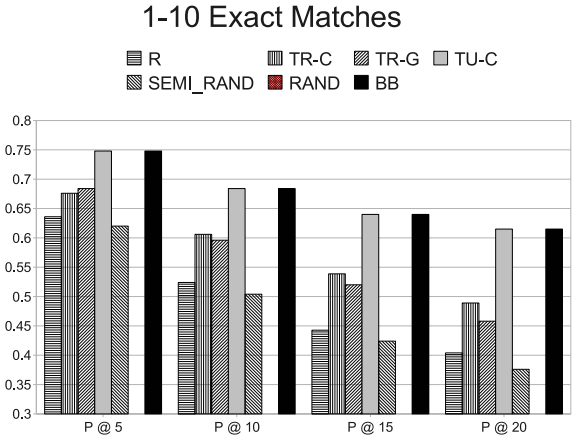

Figure 4: Results of precision at 5, 10, 15 and 20 for evaluation of rare queries having 1 to 10 relevant resources. Results for RAND method are not visible at any precision level.

Figure 5 shows the results of precision values at different levels for the queries having 11 to 50 exact matches in the original vector space model. We observe a slight decrease in the performance of enriched vector space models when compared to the original vector space model (R) for preci-

sion @ 5. But if we consider higher precision levels (15, 20), the results of enriched vector space models are better than the results obtained from original vector space model. Particularly, the TU_C model performed better than all other methods, which is also used in *Best of Breed* for queries having 11 to 50 exact matches.

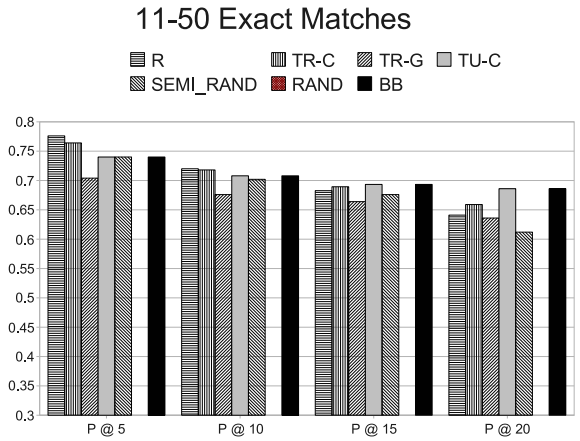

Figure 5: Results of precision at 5, 10, 15 and 20 for queries with 11 to 50 relevant resources. Results for RAND method are not visible at any precision level.

Figure 6 shows the results for the queries having more than 50 exact matches in the original vector space model. The overall performance of all the methods remains almost the same. The slightly higher precision @ 5 value for TR_G is because some images shown to the evaluators were more relevant in their opinion than for R, for example for the query *blue, bedroom*, the 3rd and the 4th images displayed for the vector space model R had the tags *blue, bedroom, plant* and *blue, bedroom, selfportrait* respectively, and the images shown for TR_G at 3rd and 4th positions had the tags *blue, bedroom, home* and *blue, bedroom, house*. This also suggests that using enriched vector space models also helps in ranking relevant resources higher where we already have many exact matches for the query in the original data. Compared to other methods for precision @ 20, the TR_C model performed better and is also used in *Best of Breed* model for queries having more than 50 exact matches.

Figure 7 shows the results of all the queries used for evaluation. The results in Figure 7 also verify the hypothesis that selecting appropriate model for a particular type of query gives an overall improvement in the search results. We represent the appropriate vector space model against a particular type of query as *Best of Breed* (BB) model. This model performs better than other methods and achieves 15% of improvement when comparing its results to original vector space model for precision @ 20.

We also performed statistical significance tests (t-Test) of results achieved through enriched vector space models and original vector space model. When considering search results for all queries, the results are significantly different for precision @ 10, 15, or 20 with p ranging from 0 (P@20) to 0.003 (P@10). However the results are not significantly different for precision @ 5 with $p = 0.11$. This is due to the reason that most relevant results are listed at the top for all

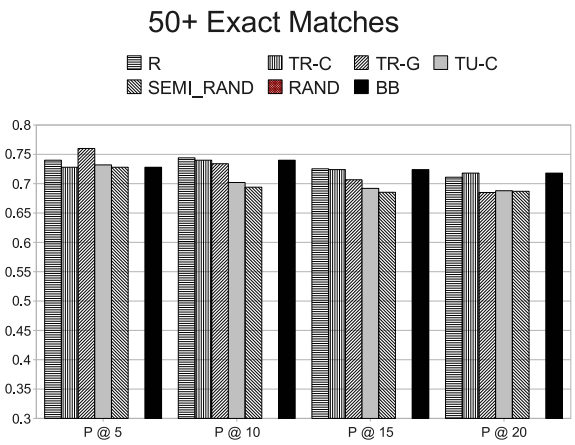

Figure 6: Results of precision at 5, 10, 15 and 20 for queries having more than 50 relevant resources. Results for RAND method are not visible at any precision level.

the methods. But we achieve significantly different results for precision levels higher than 5.

6. RELATED WORK

Folksonomies are different from normal text documents or web pages, because the tags associated with resources are far less than the number of words in text documents. The fewer number of tags associated to resources makes the available data in folksonomies very sparse. To the best of our knowledge, the methods we have proposed are the first in the field of folksonomies to reduce the sparseness in the folksonomy data. The enrichment of folksonomies helps particularly for the queries which have a very few relevant resources.

A lot of research has been done in the last few years in folksonomies or social tagging systems. The research work done on folksonomies related to our research can be divided into two parts. First, getting semantics out of tags and second searching and improving resource retrieval in folksonomies.

Extracting Tags Semantics: [5, 11, 16] present different methods for finding semantic relationships between tags. They [11, 16] recommend tags, and provide semantic analysis of tags [5]. [13, 15] propose methods to extract ontologies from folksonomies. [9, 18] recommend tags using external data sources (e.g. page text, anchor text, Wikipedia etc.). The above mentioned methods look into one of the dimensions of semantic relationships between tags; however in this paper we separated the semantic tag relationships into two different dimensions (see section 3.1). Our goal is also different from the above mentioned research works, as we exploit the semantic relationships between tags to improve search results and reduce sparseness in folksonomies.

Searching and Browsing in Folksonomies: Searching in folksonomies is becoming an interesting research area. Contests are also organized to improve search in Folksonomies [7, 8]. Many researchers have proposed different methods to improve search and browsing experiences in folksonomies. Jaschke *et al.* [10] present an algorithm for ranking tags, resources, and users in folksonomies. In future it might be of interest to compare the ranking results of their algorithm with our simple ranking approach. [1, 3, 4, 12]

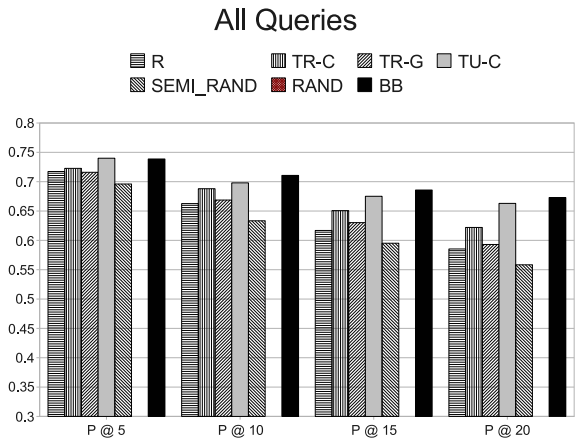

Figure 7: Results of precision at 5, 10, 15 and 20 for all the 150 queries. Results for RAND method are not visible at any precision level. The Best-of-Breed (BB) method performs better than all other methods at all precision levels.

propose algorithms to improve browsing in Folksonomies. They explore semantic and hierarchical relations between tags for browsing resources. Their focus is on improving browsing experience instead of reducing sparseness or improving search results in folksonomies. Yahia *et al.* [19] present a study of network-aware search, in comparison to their work, our focus is at system level without going into preferences of individual users. Zhou *et al.* [20] propose a generative model for social annotations to improve information retrieval in web search. They do not suggest, how one can use their system for improving search in Folksonomies. In our previous work [2], we have used Latent Semantic Indexing (LSI) [6] for improving search results in folksonomies. Due to scalability issues, we were unable to compare our current approach with the previous one or purely LSI. In future, we plan to compare the two approaches on a smaller scale.

Our approach is different from all the above mentioned methods as we 1) propose to reduce the sparseness in folksonomies, 2) suggest appropriate vector space model for a particular type of query, and 3) exploit social relationships between tags using social context (see section 3.1.2) which enable us to find the relationships between tags which might not be possible by only considering resource context.

7. CONCLUSIONS

In this paper we show how we can improve search in folksonomies like Flickr or del.icio.us, by reducing the sparsity in the data. We propose methods to find semantically related tags having different types of relationships and contexts. We also describe methods to identify generalized relationships between tags and also exploit social context of the tags to reduce sparsity in the folksonomies. By enriching the folksonomies using semantically related tags, we show that resources which are currently unsearchable can be retrieved. Human based evaluation of enriched vector space models show improvement in search results, especially for the queries where we can not retrieve many relevant resources which lack the queried tags but are relevant to the query using standard methods. We suggest using appropriate vector space model against a query based on the number of relevant resources for that query. Experimental results show that such method (*Best of Breed*) gives overall improvement in search results.

8. ACKNOWLEDGMENTS

This work has been partially supported by the European project *Semiotic Dynamics in Online Social Communities* (Tagora, FP6-2005-34721). We would like to acknowledge Higher Education Commission of Pakistan (HEC) and German Academic Exchange Service (DAAD) for providing scholarship and support to Rabeeh Abbasi for conducting his PhD. We are also thankful to the evaluators who spent their precious time to do the evaluation.

9. REFERENCES

[1] R. Abbasi, S. Chernov, W. Nejdl, R. Paiu, and S. Staab. Exploiting Flickr Tags and Groups for Finding Landmark Photos. In *ECIR'09: Proceedings of 31st European Conference on Information Retrieval / Advances in Information Retrieval*, volume 5478 of *Lecture Notes in Computer Science*, pages 654–661. Springer Berlin / Heidelberg, 2009.

[2] R. Abbasi and S. Staab. Introducing Triple Play for Improved Resource Retrieval in Collaborative Tagging Systems. In *Proceedings of Exploiting Semantic Annotations in Information Retrieval (ESAIR), workshop at ECIR'08*, March 2008.

[3] R. Abbasi, S. Staab, and P. Cimiano. Organizing Resources on Tagging Systems using T-ORG. In *Proceedings of Bridging the Gap between Semantic Web and Web 2.0, workshop at ESWC 2007*, pages 97–110, 2007.

[4] G. Begelman, P. Keller, and F. Smadja. Automated Tag Clustering: Improving search and exploration in the tag space. *Proceedings of the Collaborative Web Tagging Workshop at WWW*, 2006.

[5] C. Cattuto, D. Benz, A. Hotho, and G. Stumme. Semantic Grounding of Tag Relatedness in Social Bookmarking Systems. *The Semantic Web - ISWC 2008*, pages 615–631, 2008.

[6] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[7] J. Gonzalo, P. Clough, J. Karlgren, and U. SICS. Overview of iCLEF 2008: search log analysis for Multilingual Image Retrieval. In *Cross Language Evaluation Forum (CLEF 2008) Workshop Notes, Aarhus (September 2008)*, 2008.

[8] J. Gonzalo, J. Karlgren, and P. Clough. iCLEF 2006 Overview: Searching the Flickr WWW photo-sharing repository. *Proceedings of Cross Language Evaluation Forum (CLEF)*, page 8, 2006.

[9] 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.

[10] A. Hotho, R. Jaschke, C. Schmitz, and G. Stumme. Information Retrieval in Folksonomies: Search and

Ranking. *LECTURE NOTES IN COMPUTER SCIENCE*, 4011:411, 2006.

[11] R. Jaschke, L. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme. Tag Recommendations in Folksonomies. *LECTURE NOTES IN COMPUTER SCIENCE*, 4702:506, 2007.

[12] R. Li, S. Bao, Y. Yu, B. Fei, and Z. Su. Towards effective browsing of large scale social annotations. *Proceedings of the 16th international conference on World Wide Web*, pages 943–952, 2007.

[13] P. Mika. Ontologies are us: A unified model of social networks and semantics. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(1):5–15, 2007.

[14] G. Pass, A. Chowdhury, and C. Torgeson. A picture of search. In *The First International Conference on Scalable Information Systems*, 2006.

[15] P. Schmitz. Inducing Ontology from Flickr Tags. In *Proceedings of Collaborative Web Tagging, workshop at WWW'06*, May 2006.

[16] 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.

[17] T. V. Wal. Explaining and showing broad and narrow folksonomies, 2005. Available at `http://www.personalinfocloud.com/2005/02/explaining_and_.html`.

[18] L. C. Wee and S. Hassan. Exploiting Wikipedia for Directional Inferential Text Similarity. *Information Technology: New Generations, 2008. ITNG 2008. Fifth International Conference on*, pages 686–691, April 2008.

[19] S. A. Yahia, M. Benedikt, L. V. S. Lakshmanan, and J. Stoyanovich. Efficient network aware search in collaborative tagging sites. *Proceedings VLDB Endow.*, 1(1):710–721, 2008.

[20] D. Zhou, J. Bian, S. Zheng, H. Zha, and C. L. Giles. Exploring social annotations for information retrieval. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 715–724, New York, NY, USA, 2008. ACM.