# Tag Similarity in Folksonomies

**Hatem Mousselly-Sergieh[1,2], Elöd Egyed-Zsigmond[2], Mario Döller[3], Gabriele Gianini[4], Harald Kosch[1], Jean-Marie Pinon[2]**

*[1] Universität Passau, Germany*
*[2] Université de Lyon, France*
*[3] FH Kufstein, Austria*
*[4] Università degli Studi di Milano, Italy*
*hatem.mousselly-sergieh@insa-lyon.fr, elod.egyed-zsigmond@insa-lyon.fr,*
*mario.doeller@fh-kufstein.ac.at, gabriele.gianini@unimi.it,*
*harald.kosch@unipassau.de, jean-marie.pinon@insalyon.fr*

ABSTRACT. *Folksonomies - collections of user-contributed tags, proved to be efficient in reducing the inherent semantic gap. However, user tags are noisy; thus, they need to be processed before they can be used by further applications. In this paper, we propose an approach for bootstrapping semantics from folksonomy tags. Our goal is to automatically identify semantically related tags. The approach is based on creating probability distribution for each tag based on co-occurrence statistics. Subsequently, the similarity between two tags is determined by the distance between their corresponding probability distributions. For this purpose, we propose an extension for the well-known Jensen-Shannon Divergence. We compared our approach to a widely used method for identifying similar tags based on the cosine measure. The evaluation shows promising results and emphasizes the advantage of our approach.*

RÉSUMÉ. *Les folksonomies sont des collections d'annotations créés de manière collaborative par plusieurs utilisateurs. Afin d'améliorer la recherche d'information à l'aide de folksonomies, leur classification permet d'apporter des solutions à certains des problèmes inhérentes au caractère libre et collaboratif des folksonomies. Ces problèmes sont les fautes de frappe, des séparations-collage d'expressions, le multilinguisme, etc. Dans ce papier nous proposons une nouvelle méthode de classification d'annotations basé sur la prise en compte de la cooccurrence des mots avec un ensemble de mots fréquents. Nous utilisation la distribution de cooccurrences des mots peu fréquents avec l'ensemble de mots fréquents afin de créer une fonction de similarité entre les mots. L'approche a été évaluée sur des ensembles d'annotations provenant de Flickr associés à des images prises dans quelques grandes villes. Nous avons comparé notre méthode à un algorithme plus classique basé sur la distance de cosinus entre des vecteurs de mots.*

KEYWORDS: *Folksonomies, Tag Similarity, Tag Clustering, Semantic Web*

MOTS-CLÉS : *Folksonomies, similarité de tags, Tag Clustering, Semantic Web*

## 1. Introduction

With the emergence of Web 2.0, users become able to add contents to the web by themselves. To alleviate the semantic gap when retrieving web resources tagging (Voss, 2007) was proposed as a simple and efficient solution. Tagging allows users to annotate any kind of web resources, such as images, video, text, etc. with keywords called "tags". A collection of tags that are created by different users is called a folksonomy (Vanderwal, 2010). Currently, many web portals (e.g. Flickr[1], delicious[2]) allow users to tag their resources as well as to collaborate with other users to perform the annotation task. Folksonomies have been used with success by different applications. They enable the identification of user communities and allow efficient search and browsing (Diederich et Iofciu, 2006) (Hotho *et al.*, 2006). Additionally, many recommendation system benefit from the user-generated tags in creating their recommendations (Sergieh *et al.*, 2012) (Li *et al.*, 2009)

Tags can be easily created since they do not require any special knowledge and user can freely use whatever they think suitable to describe web resources. However, this freedom poses a real challenge for applications that use folksonomies. Tags are ambiguous (e.g. use of different synonyms), they can be in different languages (e.g. English: *Bridge* and German: *Brücke* ) and contain inflections (e.g. *book* vs. *books*). Additionally, tags which consist of word combinations are expressed differently. Some users use the space characters (e.g. *Eiffel tower*), other users use hyphens or underscores to indicate that the words represent a single unit (e.g. *Eiffel-tower*); while another group of users may connect all the words together (e.g. *Eiffeltower*).

In recent years, research suggested that using clustering techniques enables the discovery of hidden structures in folksonomies and grouping related tags of different users together (Begelman *et al.*, 2006) (Papadopoulos *et al.*, 2010) (Specia et Motta, 2007). The identified tag clusters showed to be helpful for the generation of topic hierarchies in tagging system and for improving content retrieval and browsing (Begelman *et al.*, 2006) (Brooks et Montanez, 2006).

In line with that, we propose a novel approach for identifying similar tags in folksonomies (Figure 1). First, a tag subset $G$ is created from the most occurring tags in the folksonomy. Next, for each tag $g \in G$ a co-occurrence probability distribution is created. This is done as follows: First, we count the number of times in which $g$ was used with each of the other tags in $G$ to annotate the same resources in the folksonomy. Second, the co-occurrence counts of $g$ with the other tags are converted to a probability distribution by normalizing over the total occurrence of $g$ in the folksonomy. The most frequent tags are then clustered according to the distance between their corresponding probability distributions.

In a further step, probability distributions are calculated for the less frequent tags (The set $W$ in Figure 1). Here, the probability distributions are computed based on the co-occurrences of the tag with each of the frequent tag **clusters** (instead of individual tags)

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

which were generated in the previous step.
To find similar tags, the distances between the co-concurrence probability distributions must be calculated. For this purpose, we extend the well-known probability similarity measure *Jensen-Shannon Divergence (JSD)*(Manning et Schütze, 1999) so that it can take into account the sample size from which the co-occurrence distributions are calculated.

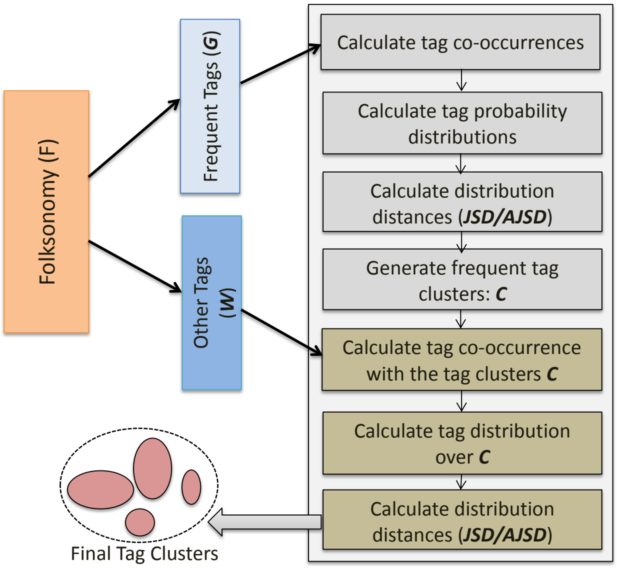

Folksonomy (F)
Frequent Tags (*G*)
Other Tags (*W*)
Calculate tag co-occurrences
Calculate tag probability distributions
Calculate distribution distances (***JSD/AJSD***)
Generate frequent tag clusters: ***C***
Calculate tag co-occurrence with the tag clusters ***C***
Calculate tag distribution over ***C***
Calculate distribution distances (***JSD/AJSD***)
Final Tag Clusters

Figure 1: The different steps followed by our approach for identifying similar tags in folksonomies. JSD correspond to the Jensen-Shannon Divergence and AJSD correspond to our extension of that measure

We evaluated the proposed approach based on real datasets obtained from Flickr. For this purpose, we crawled Flickr tags of images taken in four cities: London, San Francisco, Munich and Venice. We compared our approach to a common solution that identifies similar tags by calculating the cosine similarity between their co-occurrence vectors. The results of the preliminary evaluation are promising. They show that our approach outperforms the traditional approach.

The rest of the paper is organized as follows: in the next section we review related work. In section 3 the proposed tags clustering approach is discussed in detail. Section 4 shows preliminary evaluation results. Finally, we conclude and discuss future work in section 5.

## 2. Related work

Disregarding the application domain, works on tag clustering use in common the graph of tag co-occurrences as input. In the graph, tags are represented as nodes while an edge between two nodes reflects the similarity of the corresponding tags. Furthermore, several similarity measures, which are mainly derived from tag co-occurrence statistics, were proposed in the literature to assign weights to the edges. It is also common to apply a clustering method, such as agglomerative clustering to generate the final tag clusters.

Tag clustering in (Begelman *et al.*, 2006) is based on tag co-occurrence counts. For each pair of tags, their co-occurrence according to the same resource is counted. A cut-off threshold is then used to determine if the co-occurrence of the tags indicate similarity. The cut-off threshold is determined using the first and the second derivatives of the tag co-occurrence curve. However, no clear justification for that choice was given. To determine the final clusters the tag co-occurrence matrix is fed to spectral bisection clustering algorithm.

In (Gemmell *et al.*, 2008b) (Gemmell *et al.*, 2008a) the authors use agglomerative clustering to generate tag groups with a similarity measure based on the term frequency-inverse document frequency (TF.IDF) (Baeza Yates *et al.*, 1999). The generated tag clusters are then used as a nexus between users and their interests.

In (Specia et Motta, 2007) the authors propose a tag clustering approach based on co-occurrence similarity between the tags. First, the tags are organized in a co-occurrence matrix with the columns and the rows corresponding to the tags. The entries of the matrix represent the number of times two tags were used together to annotate the same resource. Each tag is represented by a co-occurrence vector and the similarity between two tags is calculated by applying the cosine measure on the corresponding vectors. A clustering algorithm is then defined over the co-occurrence similarity matrix.

In the work presented in (Simpson, 2008) a normalized tag co-occurrence based on Jaccard measure was initially used to generate a tag graph. After that, tag clusters were generated by applying iterative divisive clustering on the tag graph.

Another algorithm for tag clustering which is based on the notion of $(\mu, \epsilon) - cores$ (Xu *et al.*, 2007) was introduced in (Papadopoulos *et al.*, 2010). For this purpose, a tag graph is built and the edges were weighted according to structural similarity between the nodes. In other words, tags that have a large number of common neighbors to each other are grouped together.

The focus of the introduced works is on finding a clustering approach that provides best grouping of tags. They share the common idea of using tag similarity measures that are based on co-occurrence of **all tag pairs** and they differ according to the applied clustering algorithm.

The work presented here makes a different hypothesis. We assume that providing a good tag similarity/distance measure leads to a better clustering results disregarding the applied clustering algorithm. Therefore, we provide a similarity/distance measure that uses tag co-occurrence more efficiently. This is done by limiting the calculation to the co-occurrence of each tag with a smaller subset of the most frequent tags (instead of considering the co-occurrence with every tag in the folksonomy). Furthermore, we provide a similarity measure that is aware of the fluctuations in the co-occurrence counts caused according to the sampling. Briefly, the paper provides the following contributions:

– A new tag similarity/distance measure based on the distance between co-occurrence probability distributions of the tags.

– An extension for calculating Jensen-Shannon divergence between two probability distributions. The new measure deals with the inherent fluctuations of the estimated distributions.

– Evaluation and comparison of our method to the traditional tag co-occurrence similarity measure.

## 3. Tag Clustering Approach

### 3.1. *Introduction*

Traditionally, a folksonomy $F$ can be defined as a tuple $F = \{T, U, R, A\}$ where $T$ is the set of tags that are contributed by a set of users $U$ to annotate a set of resources $R$. Two tags $t_1, t_2 \in T$ occur together "co-occur" if they are used by one or more users to describe a resource $r \in R$. This is captured by the assignment set, $A = (u, t, r) \in U \times T \times R$.

Tag co-occurrence form the initial input for the majority of works that aims at identifying semantic similarity among tags. In our approach, we defined the following process to identify similar tags in folksonomies:

1) We identify the collection of tags that occur most in the folksonomy.

2) We derive a probability distribution for each tag in the folksonomy based on their co-occurrence with the most frequent tags.

3) After the co-coccurrence proabilty distributions have been aquired, we calculate the distance between them for each tag pair.

4) Finally, two tags are considered similar if the distance between their distributions is under a certain threshold.

Figure 2 shows an example of the co-occurrence distribution of a subset of tags which are used to annotate photos taken in the city of London (more details are provided in the next section). The x-axis corresponds to the subset of most frequent tags. The y-axis shows the co-occurrence distribution of four less frequent tags, namely: "big", "ben", "river" and "thames". It can be seen that the tags "big" and "ben" show a similar co-occurrence behavior. The same holds for "river" and "thames".

Since the set of frequent tags can be correlated, the occurrence of a given tag with one of the correlated frequent tags implies the co-occurrence with the other ones. To consider this case, we extend the above process by first grouping the most frequent tags according to the corresponding co-occurrence probability distributions. Now, instead of using the individual frequent tags, the co-occurrence probability distributions of the

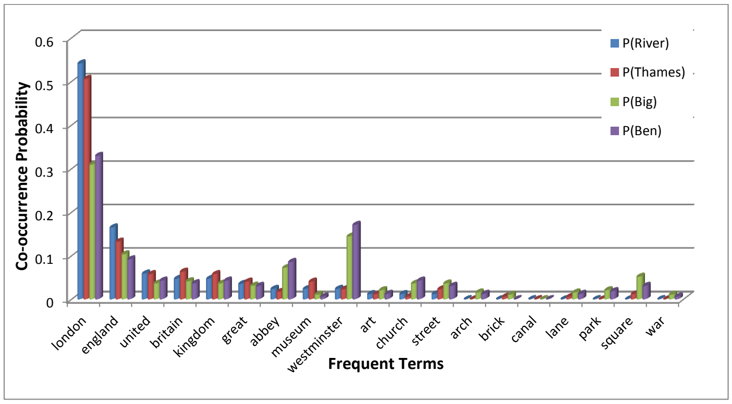


Figure 2: Co-occurrence probability distribution (y-axis) of tags that are used to annotate images in London over a subset of frequent tags (x-axis)

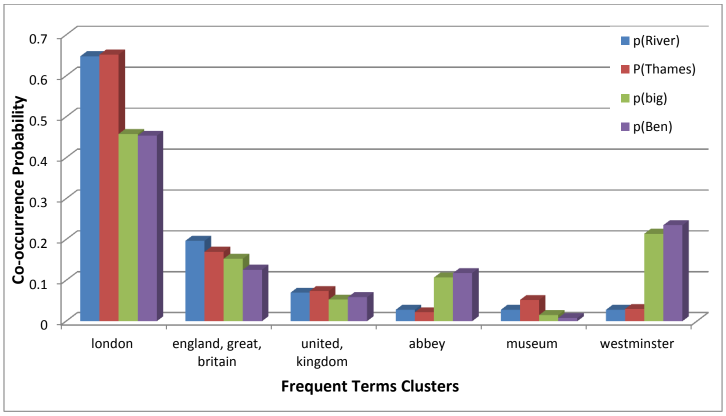


Figure 3: Co-occurrence probability distribution (y-axis) of tags that are used to annotate images in London over a subset of frequent terms clusters (x-axis)

less frequent tags are calculated over the **clusters** of frequent tags. Figure 3 illustrates the co-occurrence probability over frequent tag clusters.

### 3.2. *Formal Definitions*

In the folksonomy $F = \{T, U, R, A\}$ we divide the tag set $T$ into two subsets:

– the subset of the most frequent tags $G \subset T$ (tags with number of occurrences higher than a predefined threshold) and

– the subset of the remainder tags $W = T \setminus G$.

Hereafter we indicate the most frequent tags by the variable $g$ and the tags in the complement set by $w$ (i.e. $g \in G$ and $w \in W$, whereas a tag, taken from either set will be indicated by $t \in T$).

Now we define the four steps for this first phase of data processing (the most frequent tags processing), namely: co-occurrence measure, empirical probability definition, tag dissimilarity computation and tag clustering:

1) For a tag $t \in T$ we can quantify the co-occurrences with each of the most frequent tags $g \in G$, by counting the number of times $\#(t, g)$ in which $t$ was used together with $g$ to annotate a resource. We can use this set of counts to create an histogram in the variable $g$.

2) Then, normalizing this histogram with the total number of co-occurrences of $t$ with the elements of the set $G$ we obtain the empirical co-occurrence probability distribution $P_t(g)$ for the tag $t$ with the elements $g \in G$:

$$P_t(g) = \frac{\#(t, g)}{\sum_{g \in G} \#(t, g)} \qquad [1]$$

In view of what follows we can consider $\#(t, g)$ as a shorthand for $\#_{tag}^{tag}(t, g)$, which indicates the number of co-occurrences of a tag $t$ with another tag $g$.

3) Given the empirical co-occurrence probability distribution $P_{t_1}(g)$ relative to a tag $t_1 \in T$ and the co-occurrence probability distribution $P_{t_2}(g)$ relative to a different tag $t_2 \in T$ it is possible to define a dissimilarity metrics $D_{AJSD}(t_1, t_2)$ between the two tags, based on information-theoretic divergences, namely an adaptation of the Jensen-Shannon Divergence (AJSD): due to the complexity of this step the definition and the discussion are provided in the next section.

4) This dissimilarity metrics between tags can then be used to classify them into clusters. Indeed we applied this procedure to the elements of the set $G$ of most frequent tags itself: for every pair of tags $g_1, g_2 \in G$ we computed the empirical probability distributions $P_{g_1}(g)$ and $P_{g_2}(g)$ of co-occurrence with the other most frequent tags, according to the above definition (1) and then computed the AJSD dissimilarity metrics $D_{AJSD}(g_1, g_2)$, defined in the following section; finally we applied a clustering algorithm over the set $G$ using $d$ as the dissimilarity metrics and achieved the partitioning of the set $G$ into clusters – in other words, we obtained from $G$ a family of mutually exclusive subsets $\mathcal{G} = \{G_1, ..., G_j, G_k, ...G_n\}$ whose union covers $G$ (i.e. $G_j \bigcap G_k = \emptyset \; \forall G_j, G_k \subset G$ and $\bigcup_k G_k = G$).

At this point, as outlined in the previous section, we enter a second data processing phase: we considered the less frequent tags $w \in W$ and repeated the co-occurrence probability estimate, the dissimilarity calculation and finally the clustering procedure – however we do this adopting an important variant, relating to the definition of co-occurrence and described hereafter – so as to achieve a partitioning of $W$ into a family of subsets: $\mathcal{W} = \{W_1, W_2, \ldots, W_m\}$.

1) This time we did not relate a tag $t$ to another tag, say $g$, but, rather, a tag $t$ to a cluster of tags $G_k$, i.e. the co-occurrence definition this time concerned the co-

occurrence of a tag and a cluster. There are several possible choices for providing a measure of the co-occurrence of a tag and a cluster: we have chosen to use the number of co-occurrences with the term which has the maximum number of co-occurrences. Formally we defined the measure $\#^{tag}_{clus}(w, G_k)$ of the co-occurrence between a tag $w$ and a cluster $G_k$ as

$$\#^{tag}_{clus}(w, G_k) = \max_{g \in G_k}(\#^{tag}_{tag}(w, g)) \qquad [2]$$

2) Consequently the empirical distribution of co-occurrences turned out to be defined by

$$P_w(G_k) = \frac{\#^{tag}_{clus}(w, G_k)}{\sum_{G_k \in \mathcal{G}} \#^{tag}_{clus}(t, g)} \qquad [3]$$

3) Based on the empirical distributions for all the pairs of terms $w_i, w_j \in W$ we computed the AJSD dissimilarities $D_{AJSD}(w_i, w_j)$, defined in the next section, and

4) were able to cluster the less frequent terms into clusters reflecting their co-occurrence behaviour with respect to the high frequency term clusters.

### 3.3. *Distance Measures*

We calculate the semantic similarity between two tags $t_1, t_2 \in T$ based on the similarity between their corresponding empirical co-occurrence probability distributions.

In the literature, different method were used to calculate this distance (Cha, 2007) (Huang, 2008) (Matsuo et Ishizuka, 2004). The Jensen-Shannon Divergence (JSD) (Manning et Schütze, 1999) has shown to outperform other measures (Ljubešić *et al.*, 2008), including the the Kullbak-Leiber divergence, on which it is based. We use a metrics derived from the Jensen-Shannon divergence and able to take into account the statistical fluctuations due to the finitness of the sample: the rationale for using this specific definition and the derivation of our metrics are developed hereafter.

Consider the empirical co-occurrence probability distribution $P_{t_1}(g)$, with $g \in G$, relative to a tag $t_1$ and a set of high frequency tags $G$ as defined in (1), and consider the analogous empirical probability distribution $P_{t_2}(g)$, with $g \in G$, relative to a tag $t_2$ and the same set of high frequency tags $G$. For sake of denotational simplicity we will indicate one value of the first distribution by $P(g)$ and a value of the second by $Q(g)$, whereas we will use $P$ for the first distribution as a whole and $Q$ for the second distribution as a whole.

The most typical metrics for dissimilarity between two probability distributions is the Kullbak-Leiber divergence $D_{KL}$

$$D_{KL}(P||Q) \equiv \sum_g P(g) \log \frac{P(g)}{Q(g)}$$

The expression $D_{KL}(P||Q)$ is not symmetric in $P$ and $Q$ but can be symmetrised as follows:

$$D_{KL}(P,Q) \quad = \quad \frac{1}{2}\left(D_{KL}(P||Q) + D_{KL}(Q||P)\right) \qquad [4]$$

Unfortunately this quantity becomes infinite as soon as either $P$ or $Q$ become null in one point of the support set, due to the denominators in the logarithm arguments of the two terms. In order to avoid this issue the denominators are substituted by $M(g) \equiv (P(g) + Q(g))/2$ giving rise to the Jensen-Shannon divergence:

$$D_{JS}(P,Q) \quad = \quad \frac{1}{2}\left(D_{KL}(P||M) + D_{KL}(Q||M)\right) \qquad [5]$$

$$= \frac{1}{2}\left(\sum_g P(g)\log\frac{2P(g)}{P(g)+Q(g)} + Q(g)\log\frac{2Q(g)}{P(g)+Q(g)}\right)$$

If, as in our case, the probabilities $P$ and $Q$ are not available, but only an estimate of them is available through a finite sample represented in the form of an histogram for $P$ and an histogram for $Q$, then the divergence computed on the histograms is a random variable; this variable, under appropriate assumptions, can be used to compute an estimate of the divergence between $P$ and $Q$ using error propagation under a Maximum Likelihood approach, as illustrated hereafter.

Consider the channel on the value $g$ of the histogram for $P$ and for $Q$, characterized respectively by the number of counts $k_g$ and $h_g$, and define the following measured frequencies

$$x_g \equiv k_g/n \quad y_g \equiv h_g/m$$

where $n = \sum_g k_g$ and $m = \sum_g h_g$ are the total counts for the first and second histogram respectively. Then, for $n$ high enough, the quantities $x_g$ and $y_g$ are normally distributed around the true probabilities $P(g)$ and $Q(g)$ respectively. As a consequence the *measured* Jensen-Shannon divergence $d$ then is a stochastic variable, function of those normal variables according to the following expression

$$d \quad = \quad \frac{1}{2}\sum_g\left(x_g\log\frac{2x_g}{x_g+y_g} + y_g\log\frac{2y_g}{x_g+y_g}\right) \qquad [6]$$

The value of this expression is not in general the maximum likelihood estimate of the Jensen-Shannon divergence, mainly due to the unequal variances of the terms in the sum. In order to find the maximum likelihood estimate $\hat{d}$ of the divergence we need to proceed through error propagation, starting (a) from the Maximum Likelihood (ML) estimate of $P(g)$ and $Q(g)$ based on the variables $x_g$ and $y_g$, (b) considering and propagating their statistical errors to the summation to which they participate and finally (c) proceeding to term weighting according to the term uncertainty.

(a) Thanks to the normality condition stated above, the ML estimates of the probabilities $P(g)$ and $Q(g)$ are the following: the ML estimate of $P(g)$ is $x_g = k_g/n$ with variance given in first approximation by $\sigma^2_{P(g)} = k_g/n^2$; the ML estimate of $Q(g)$ is $y_g = h_g/m$ with variance given in first approximation by $\sigma^2_{Q(g)} = h_g/m^2$

(b) Consider the individual addendum term in the sum expression (6).

$$z_g \equiv x_g \log \frac{2x_g}{x_g + y_g} + y_g \log \frac{2y_g}{x_g + y_g} \qquad [7]$$

If the two variables $x_g$ and $y_g$ are independent, the variance propagation at the first order is

$$\begin{aligned} \sigma^2(z_g) &\simeq \left(\frac{\partial z_g}{\partial x_g}\right)^2 \sigma^2(x_g) + \left(\frac{\partial z_g}{\partial y_g}\right)^2 \sigma^2(y_g) && [8] \\ &\simeq \log^2 \frac{2x_g}{x_g + y_g} \sigma^2(x_g) + \log^2 \frac{2y_g}{x_g + y_g} \sigma^2(y_g) && [9] \end{aligned}$$

Now substituting in this formula the above expressions for the variables and their variances one gets the estimated variance $\sigma^2(z_g)$ of the term (7).

(c) Define the (statistical) precision $w_g$ (to be used later as a weight) as follows:

$$w_g \sim \frac{1}{\sigma^2(z_g)}$$

The maximum likelihood estimate of the quantity (6) is given by the following weighted sum

$$\hat{d} = \frac{\sum_g w_g z_g}{\sum_g w_g} \qquad [10]$$

The corresponding variance is $\sigma^2(\hat{d}) = 1/\sum_g w_g$.

We used $\hat{d}$ as adapted Jensen-Shannon Divergence. Notice that this adapted JSD, due to the statistical fluctuations in the samples, gives in general values greater than zero even when two samples are taken from the same distribution, i.e. even when the true divergence is zero. However by weighting the terms according to their (statistical) precision it provides a ranking for the terms, which is correlated to the true ranking in a stronger way than the raw JSD.

## 4. Evaluation

### 4.1. *Dataset*

The presented tag similarity approach was evaluated using folksonomies obtained from Flickr. For this purpose, we used Flickr API to collect tags of images taken in a specific geographical location. We downloaded images and the associated tags for four cities: London, San Francisco, Venice and Munich. To avoid the influence of redundant tags caused by bulk tagging, we limited our datasets to a single image per user. The number of the acquired images differs from one location to the other. Table 1 shows the number of images, the total number of tags and the number of unique tags corresponding to the four cities.

| **Location** | **# Images** | **# Total Tag Occurrences** | **# Unique Tags** |
|---|---|---|---|
| London | 18803 | 132250 | 3930 |
| San Francisco | 8023 | 53503 | 1390 |
| Munich | 1655 | 9179 | 317 |
| Venice | 3341 | 22814 | 643 |

Table 1: Statistics about the test folksonomies

The tags were then subjected to a light cleaning process to remove meaningless tags, such as technical (typically camera specific EXIF tags) and Flickr-specific tags. We didn't apply stemming on the tags because we wanted to check if our approach is able to handle syntactic variations. Furthermore, we removed tags that were used by less than 5 users.

### 4.2. *Experimental Setup*

For each location-based folksonomy, we calculated tag similarity according to our approach using *JSD* and the proposed extension which we denote as *adapted JSD (AJSD)*. For our algorithm two parameters have to be set: the number of the most frequent tags $k$ and the distance threshold for clustering the most frequent terms $th$. Good results were achieved by choosing the top $k$ frequent tags that account to at least 30% of the total tag occurrences in the folksonomy. Clustering threshold $th = 0.03$ and $th = 0.003$ were used for JSD and the AJSD clustering, respectively. Table 2 shows the number of generated clusters per distance measure and per city-based folksonomy.

|  | JSD | AJSD | CosTag |
|---|---|---|---|
| London | 102 | 106 | 105 |
| SF | 52 | 51 | 44 |
| Munich | 38 | 42 | 45 |
| Venice | 72 | 70 | 75 |

Table 2: User Study: The number of verified clusters per clustering method and per city-based folksonomy

Furthermore, we compared our method to a widely used method which identifies tag similarity based on the cosine similarity between the co-occurrence vectors of the corresponding tags (Specia et Motta, 2007) (Begelman *et al.*, 2006). In that approach, each tag is represented by a vector, the components of which correspond to the individual tags of the folksonomy. The entries of the tag vector are the co-occurrence counts of the tag pairs. Finally, the similarity of two tags is determined using the cosine of the angle between the corresponding tag vectors. We call this method as **COSTag**.

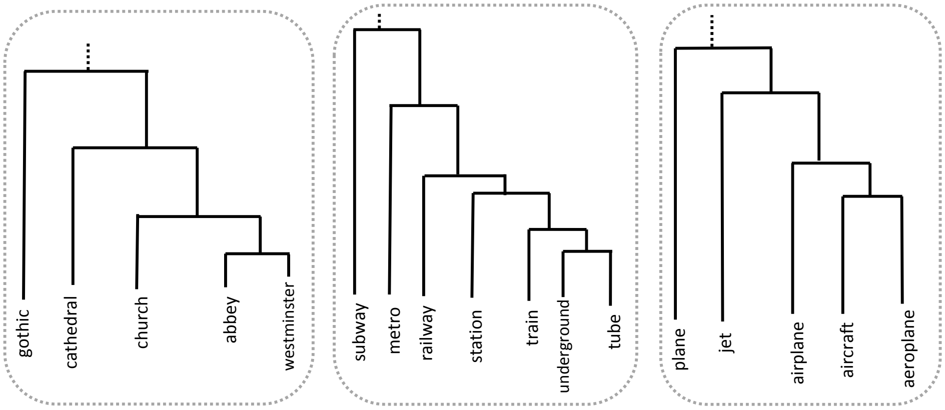


Figure 4: Sample tag clusters from the London folksonomy. The clusters were obtained from a cut-off of a hierarchical cluster that was created based on the AJSD distance measure

To compare the different tag similarity/distance measure, we generated similarity matrices according to *JSD* our extension of it *AJSD*, as well as according to the *COSTag* measure. After that, we fed the produced similarity matrices into an agglomerative clustering process with complete link as a merge criterion. To generate the final clusters, we used cut-off thresholds for the generated hierarchal clusters that fulfill two conditions: 1) avoid large clusters that contain more than 15 elements and 2) produce almost the same number of clusters from each of the similarity matrices. Note that, the choice of the clustering algorithm is not the focus of our work in this paper. The purpose of the clustering here is to enable comparing different similarity measures.

Figure 4 shows a hierarchal representation of tag clusters generated by applying our algorithm with the AJSD measure on the London folksonomy. This example shows three typical clusters having synonyms and complementary words together. In the cluster: (*gothic, cathedral, church, abbey, westminster*), *westminster abbey* is a named entity, *cathedral* and *church* are synonyms, while gothic is an adjective. The cluster, nevertheless makes sense for a human reader. It is possibe to consider these words as having a similarity $> 0$ in a comparison function. In a situation where there is no available perfect semantic resource, which is the case most of the time when dealing with generic texts or images, such clusters can improve information retrieval.

### 4.3. *Results*

The quality of the produced clusters was evaluated manually by a group of 10 users. To reduce the labor of the manual verification, we limited the evaluation to clusters that contain at least 3 tags. Each user received a list of automatically generated tag clusters

(Table 3) and verified if the tags in each cluster were related (semantically, syntactically, translation in different languages, etc.).

Finally, the clusters were rated according to three quality scales:

– *Good*: meaning that all the tags in the cluster are related.

– *Neutral*: if a subset of the tags in the cluster are related while other tags are not.

– *Bad*: if the cluster contains no or very few related tags.

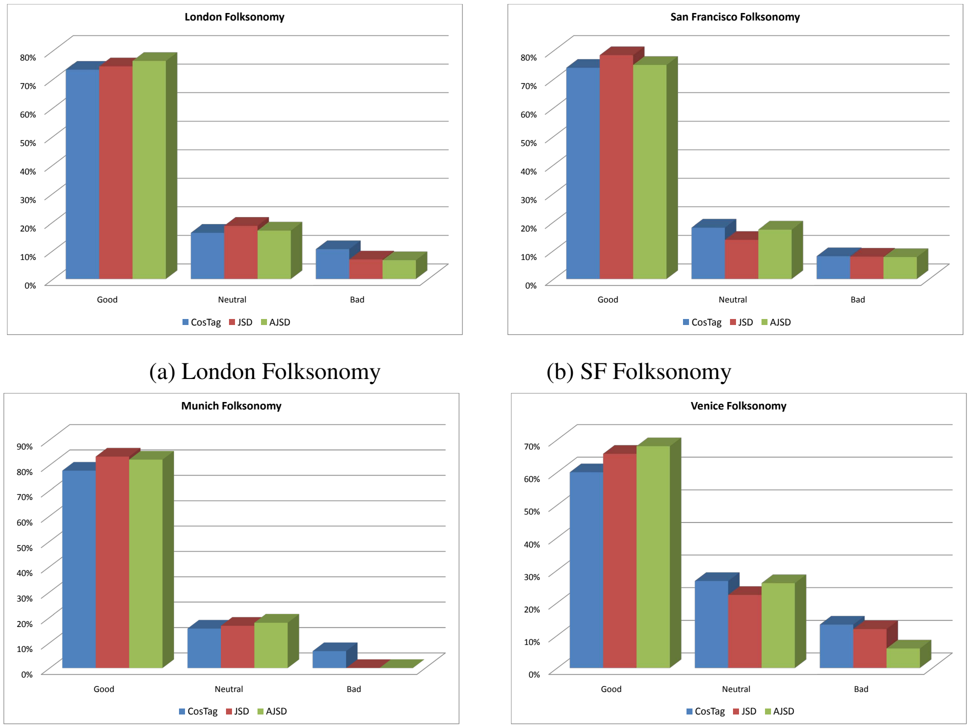
| **Cluster** | **quality** |
|---|---|
| virgin atlantic airlines | Good |
| subway metro railway station train underground tube | Good |
| still dog outdoor | Bad |
| finchley mile east end brick lane | Neutral |

Table 3: Cluster examples from the London dataset and their manual evaluations

The clusters are categorized in given quality scales if more than 50% of the polled users agree on that scale. Figure 5 shows for each city folksonomy the percentage of clusters classified according the proposed quality scales. We notice that the percentage of clusters classified as good exceeds 73% for all four folksonomies when AJSD is used. If we add the neutral ones, we achieve more than 90% good results. This confronts us in our research of discovering relations among words, looking on their use. It can be seen that the user subjective evaluation favors the tag clusters that were generated according to our proposed distance measures: *AJSD*.

## 5. Conclusion

The method described in this paper enables to make emerge word clusters that have a *meaning*. This is confirmed by the manual evaluations we carried out. These clusters help create word similarity measures that go beyond the purely syntax based ones without using explicit semantic resources such as ontologies. Two words belonging to the same cluster can be considered as having a positive similarity. This can improve similarity measures not only for texts, but also for images having annotations. Our method can improve the quality of user created annotations, where words are not separated, contain errors and descriptions are subjective and diverse. The clusters enable to locate synonyms and named entities.



(a) London Folksonomy

(b) SF Folksonomy

(c) Munich Folksonomy

(d) Venice Folksonomy

Figure 5: User evaluation for clusters generated by our proposal using JSD and AJSD and the adversary approach COSTag

## 6. References

Baeza Yates R., Ribeiro Neto B., others, *Modern information retrieval*, vol. 463, ACM press New York., 1999.

Begelman G., Keller P., Smadja F., others, “Automated tag clustering: Improving search and exploration in the tag space”, *Collaborative Web Tagging Workshop at WWW2006, Edinburgh, Scotland*, 2006, p. 15–33.

Brooks C. H., Montanez N., “Improved annotation of the blogosphere via autotagging and hierarchical clustering”, *Proceedings of the 15th international conference on World Wide Web*, ACM, 2006, p. 625–632.

Cha S.-H., “Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions”, *International Journal of Mathematical Models and Methods in Applied Sciences*, vol. 1, num. 4, 2007, p. 300–307.

Diederich J., Iofciu T., “Finding communities of practice from user profiles based on folksonomies”, *Proceedings of the 1st International Workshop on Building Technology Enhanced Learning solutions for Communities of Practice*, Citeseer, 2006, p. 288–297.

Gemmell J., Shepitsen A., Mobasher B., Burke R., "Personalization in folksonomies based on tag clustering", *Intelligent techniques for web personalization & recommender systems*, vol. 12, 2008.

Gemmell J., Shepitsen A., Mobasher B., Burke R., "Personalizing Navigation in Folksonomies Using Hierarchical Tag Clustering", *Proceedings of the 10th international conference on Data Warehousing and Knowledge Discovery*, DaWaK '08, Berlin, Heidelberg, 2008, Springer-Verlag, p. 196–205.

Hotho A., Jäschke R., Schmitz C., Stumme G., "Information retrieval in folksonomies: search and ranking", *Proceedings of the 3rd European conference on The Semantic Web: research and applications*, ESWC'06, Berlin, Heidelberg, 2006, Springer-Verlag, p. 411–426.

Huang A., "Similarity measures for text document clustering", *Proceedings of the Sixth New Zealand Computer Science Research Student Conference (NZCSRSC2008), Christchurch, New Zealand*, 2008, p. 49–56.

Li X., Snoek C., Worring M., "Learning social tag relevance by neighbor voting", *Multimedia, IEEE Transactions on*, vol. 11, num. 7, 2009, p. 1310–1322, IEEE.

Ljubešić N., Boras D., Bakarić N., Njavro J., "Comparing measures of semantic similarity", *30th International Conference on Information Technology Interfaces, Cavtat*, 2008.

Manning C., Schütze H., *Foundations of statistical natural language processing*, MIT press, 1999.

Matsuo Y., Ishizuka M., "Keyword extraction from a single document using word co-occurrence statistical information", *International Journal on Artificial Intelligence Tools*, vol. 13, num. 01, 2004, p. 157–169, World Scientific.

Papadopoulos S., Kompatsiaris Y., Vakali A., "A graph-based clustering scheme for identifying related tags in folksonomies", *Proceedings of the 12th international conference on Data warehousing and knowledge discovery*, DaWaK'10, Berlin, Heidelberg, 2010, Springer-Verlag, p. 65–76.

Sergieh H. M., Gianini G., Döller M., Kosch H., Egyed-Zsigmond E., Pinon J.-M., "Geo-based automatic image annotation", *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, ACM, 2012, Page 46.

Simpson E., "Clustering Tags in Enterprise and Web Folksonomies", *HP Labs Techincal Reports*, , 2008.

Specia L., Motta E., "Integrating Folksonomies with the Semantic Web", *Proceedings of the 4th European conference on The Semantic Web: Research and Applications*, ESWC '07, Berlin, Heidelberg, 2007, Springer-Verlag, p. 624–639.

Vanderwal T., "Off the Top: Folksonomy Entries", 2010.

Voss J., "Tagging, Folksonomy & Co - Renaissance of Manual Indexing?", *CoRR*, vol. abs/cs/0701072, 2007.

Xu X., Yuruk N., Feng Z., Schweiger T. A., "Scan: a structural clustering algorithm for networks", *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2007, p. 824–833.