

Organizing Resources on Tagging Systems using T-ORG

Rabeeh Abbasi¹, Steffen Staab¹ and Philipp Cimiano²

¹ISWEB, University of Koblenz-Landau, Germany

<http://isweb.uni-koblenz.de>

{abbasi,staab}@uni-koblenz.de

²Institute AIFB, University of Karlsruhe, Germany

<http://www.aifb.uni-karlsruhe.de/WBS>

pci@aifb.uni-karlsruhe.de

Abstract. Tagging systems (or folksonomies) like Flickr or Delicious are expanding tremendously. More and more resources are being added to them. As the resources present on these system increase in amount, it becomes difficult to explore these resources. For this purpose, we present a system T-ORG, which provides a mechanism to organize these resources by classifying the tags (or keywords) attached to them into predefined categories. Supervised classification in this case seems infeasible; therefore we also propose a new classification algorithm T-KNOW that does not require training data. For our experiments, we have downloaded images and their tags from groups present on Flickr website and then classified these tags into different categories. We have used Cohen's Kappa and F-measure to evaluate the classification results of T-KNOW. Results are encouraging and show that T-ORG can be used to explore resources in an effective manner.

Keywords: Tags Classification, Tagging Systems, Folksonomies, Semantic Web, Cohen's Kappa

1 Introduction

More people are being attracted to tagging systems like Flickr¹ or Delicious² because of the number of benefits they provide. For example, they are easy to use and do not require any specific skills. Users can search and browse resources using the tags (keywords) attached to the resources. They also provide "Tag Clouds" to browse resources. In a "Tag Cloud", frequently used tags are displayed in large text. Despite of all these benefits, sometimes it might become difficult for a user to browse particular types of resources. Just consider the scenario in which a user wants to explore vehicle images. Considering current searching and browsing facilities

¹ <http://www.flickr.com/>

² <http://del.icio.us/>

provided by these systems, it seems difficult to browse only a particular kind of resources. This problem of exploring resources of a particular type can be solved by additional classification of resources. Classifying resources into predefined categories can provide a mechanism to explore a particular type of resources present on a tagging system. It can improve the user experience and can add more benefit to existing tagging systems. Manually classifying the resources available on a tagging system is not feasible, because of the tremendous amount of data present and being added to it. Therefore some kind of system is required which can classify resources on a tagging system into some categories without any supervision.

We have explored means to automatically organize tags into hierarchies in order to explore resources in tagging systems and to provide better browsing experience to the user. For this purpose, we have developed a system T-ORG (Tag-ORGanizer), which classifies the resources of a tagging system into predefined categories and helps in browsing a particular type of resources available on tagging system. The classification of resources is based on the classification of tags attached to these resources. If a resource has two tags having two different categories, then the resource is classified as both of these categories. For example, if a resource has tags “Paris” and “Peugeot” and these tags are classified as “Location” and “Vehicle” respectively, then the resource is placed in both of these categories (i.e. Location and Vehicle). Tag classification can help a user to use tags on a tagging system in a more organized way. For example, instead of representing different tags in a tag cloud, sometimes it could be more useful, if a “Tag Cloud” displays the abstract tags (i.e. categories) and when a user clicks an “abstract tag”, its subsequent tags are displayed. In such way, a user can explore different type of tags (and hence resources) available on a tagging system, which might not be possible with a simple “Tag Cloud”.

The core of T-ORG is its classification method T-KNOW (Tag classification using KNowledge On the Web). It is based on an unsupervised mechanism for classifying tags in folksonomies. T-KNOW uses Google for finding categories of tags; therefore it does not require any training and can be used for unsupervised classification of tags (like [3]). It classifies the tags into categories using its pattern library, categories extracted from a given ontology and Google search results. As there might be several results returned by Google against a query posed by T-KNOW, a method is required to select best results on the basis of the similarity between tagging and search results. T-KNOW uses the context of the tag to measure the similarity between Google search results and the tag. We also propose four methods of selecting the context of a tag.

2 Process of T-ORG (Tag-ORGanizer)

The purpose of T-ORG is to organize resources by classifying their tags into categories. This process is done by selecting concepts from single or multiple ontologies related to the required categories and then pruning and refining these ontologies. These concepts are considered as categories into which the tags are classified. Figure 1 shows the overall process of T-ORG while each step is described below.

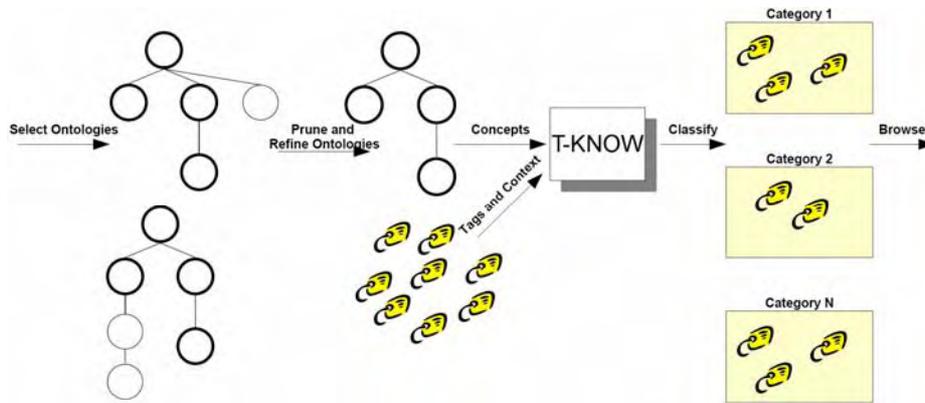


Fig. 1. Process of T-ORG

Selecting Ontology

The user of T-ORG has to decide about the categories into which the resources are to be classified. The user selects ontologies relevant to the required categories. Concepts from these ontologies are used as categories. For example to browse through the images of vehicles at Flickr, one would select a vehicle ontology. Currently this step is done manually in T-ORG.

Pruning and Refining Ontology

After selecting ontologies, they must be pruned and refined for the desired categories. Only those concepts from these ontologies are considered which have some relation to the required categories. Unwanted concepts are pruned. Redundant and conflicting concepts are refined. Missing concepts are also added into the given ontology. For example to include the images of a “draisine”, one might have to add this concept into a given vehicle ontology. Once the ontology is pruned and refined, its concepts are used as categories. Currently this step is also done manually in T-ORG.

Applying T-KNOW for Classifying Tags

Classifying the tags is a major step in the process of T-ORG. Once the ontology is selected, pruned, and refined, and categories are extracted from this ontology, then these categories and the context of tags are used for classification. Once all tags are classified into categories, each category is subsumed by its parent category, for example, every tag classified as Train, Bulldozer or Bus is finally classified as Vehicle. Section 3 describes the detailed process of classifying tags using T-KNOW.

Browsing the Resources

After classifying each tag, resources may be browsed according to the categories assigned to their tags. The browser may use information of resources to display them in categories, so that the user can browse particular type of resources present in these categories.

3 Tag Classification using knowledge On the Web

T-KNOW uses lexico-syntactic patterns and Google APIs for finding the appropriate categories of the tags. Given a list of tags and categories, T-KNOW classifies these tags into categories. It builds queries by combining linguistic patterns (Hearst Patterns [7] and a few more [3]) and the categories and then searches these queries on Google using Google API. The process of classifying tags using T-KNOW is shown in Figure 2. In what follows, we describe in more detail the steps shown in Fig. 2.

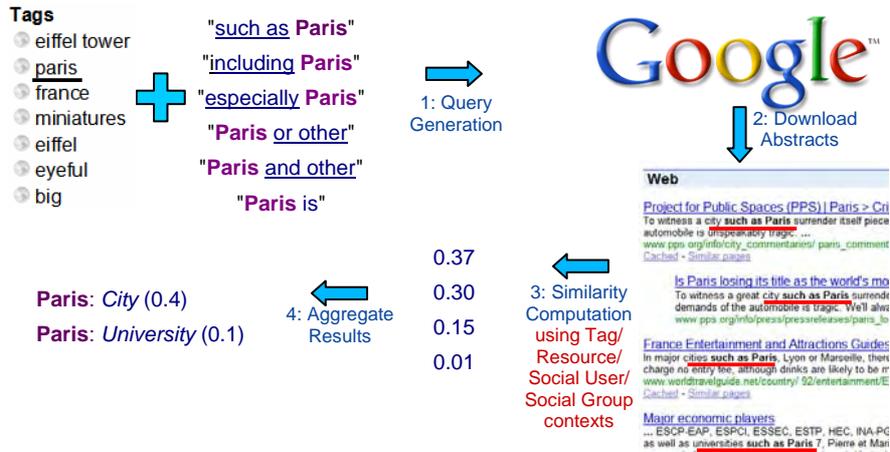


Fig. 2. Process of T-KNOW

Assume a tag like “Paris” is to be classified in a context as depicted in Figure 4

- Step 1: Queries are generated by concatenating the tag and the clues, e.g. “**such as Paris**” is a query generated by combining the clue “such as” and the tag “Paris”
- Step 2: The queries are searched using the Google API and abstracts of search results are downloaded, e.g. “To witness a **city such as Paris** surrendered itself...” is a search result abstract downloaded for the query “such as Paris”
- Step 3: The similarity between each abstract and context (described in Section 3.1) of tag is computed, e.g. between the abstract “To witness a city such as Paris...” and context of the tag “Paris” (eiffel tower, france, miniatures...). If similarity is above a certain threshold value, then depending upon the clue used, the abstract is matched against the pattern, e.g. the abstract “To witness **city such as Paris**...” is matched against the Hearst pattern [7] “**CONCEPT such as (INSTANCE,?)**+ ((and|or) INSTANCE)”, where CONCEPT is the expected category and INSTANCE is the tag. Hence “City” is extracted as an expected category of the tag “Paris” from this abstract.
- Step 4: The results are aggregated and the category having highest similarity with the tag’s context is returned, e.g. for the tag “Paris” the category “City” is

returned, because it has higher similarity than the other category e.g. “University”

The pseudocode for T-KNOW is shown in Figure 3. CN is the total number of clues used. $clue(t,i)$ is a function which returns a *query* string by concatenating the tag t with predefined clues (from 1 to CN). This query is searched on Google using the Google API. The function $download_google_abstracts(query,n)$ takes the query and number of abstracts required as parameters and returns the abstracts of search results found for the given query. The cosine measure is calculated between each abstract (a) and context (ctx) of the tag (t). If the value of the cosine measure is above a certain threshold, then the abstract (a) is considered for further processing. Patterns (find the complete list in [3], and example in step 3) for clue i are matched against the abstract a using the function $pattern_match(a,i)$. If the pattern is matched, then the category of current tag is extracted. The category having the highest similarity with context of the tag is returned.

```
TKNOW(Tag t, Context ctx) {
  for i = 1 to CN {
    query = clue(t,i)
    abstracts = download_google_abstracts(query,n);
    foreach a in abstracts {
      sim = calculate_similarity(a,ctx);
      if (sim > threshold) {
        if (pattern_match(a,i)) {
          c = get_category(a);
          Res[c] = Res[c]+sim;
        }
      }
    }
  }
  return maxarg_ Res[c];
}
```

Fig. 3. Pseudocode of T-KNOW

3.1 Measuring similarity between search results and tags

There can be multiple ways for computing the similarity between the search result and the tag depending upon the context of the tag. We have proposed four methods (3.1.1 – 3.1.4) of selecting the context of the tag. For measuring similarity between Google search result and the context of a tag, the cosine measure is computed between the

bag of word representations of the abstract of the downloaded search result \vec{a} and the context \vec{C} of the tag t . If this cosine measure is above a certain threshold value, the result is considered for further processing. The cosine measure is calculated as

$$\cos(\angle(\vec{C}, \vec{a})) = \frac{\vec{C} \cdot \vec{a}}{\|\vec{C}\| \|\vec{a}\|} \quad (1)$$

Section 4.2 presents different results obtained using different threshold values and different contexts. To understand the different contexts, consider the images in Figure 4. The left most image is of “Eiffel Tower”. The middle image is “Notre Dame”. The right most is the image of a Cow. Table 1 shows the details of each image, its tags, the user who has uploaded this image, and the group in which this image is present.

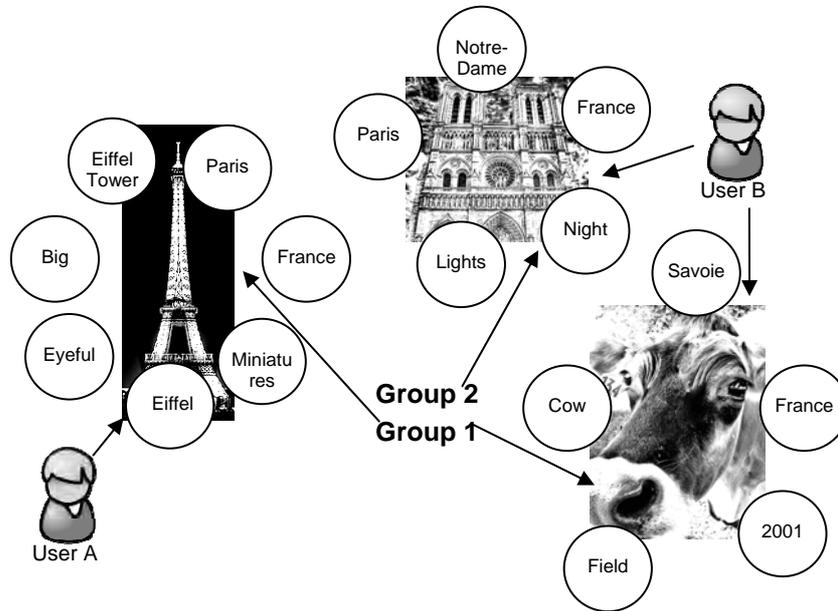


Fig. 4. Sample images with tags

Table 1. Details of images in Figure 4

| Image | Tags | User | Group |
|--------------|--|------|-------|
| Eiffel Tower | Eiffel Tower, Paris, France, Miniatures, Eiffel, Eyeful, Big | A | 1 |
| Notre Dame | Notre-Dame, France, Night, Lights, Paris | B | 2 |
| Cow | Savoie, France, 2001, Field, Cow | B | 1 |

To formally define the context of the tags, we need to formally define the tagging

systems. We use the same formal model of tagging systems (or folksonomies) as defined in [10]. According to [10] a tagging system (or folksonomy) is a tuple

$$F := (U, T, R, Y) \quad (2)$$

where U , T , and R are finite sets representing users, tags, and resources respectively, Y represents taggings by users U , using tags T of resources R , and $Y \subseteq U \times T \times R$. In addition to these sets, we also use the set of groups G that might be found in some tagging systems (like Flickr). Users can post their resources to these groups. Now we formally define different contexts as

3.1.1 Resource Context (R)

In order to represent a tag by its context, we here consider the case of resource context. We choose the tags that belong to the current resource except the tag (to be classified) itself. The Resource Context of tag t for resource r can be defined as

$$C_R(t, r) = \{t' \in T \setminus \{t\} \mid (u, t', r) \in Y \wedge u \in U\} \quad (3)$$

We are also interested in the frequency of t_i in resource r_j (in case of Flickr it is at most 1, because one tag can occur only at most once in a resource) to create a bag of words using this context. $W_R(t, r)$ represents the number of times tag t appears with resource r .

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

We can get the Resource Context of a tag t of resource r using $C_R(t, r)$ and for each tag t_x in the Resource Context of tag t , we can get its number of occurrences in resource r_x using $W_{CL}(t_x, r_x)$.

We can define a bag-of-words resource context representation of a tag t appearing in resource r , i.e. by

$$B_R(t, r) = \{(t', W_R(t', r)) \mid t' \in C_R(t, r)\} \quad (5)$$

Note that B_T , B_{SU} , and B_{SG} can be defined in the similar manner for Tag, Social User, and Social Group contexts respectively. Consider that we want to classify the tag "Paris" of the image Eiffel-Tower in Figure 4, only the tags of the image Eiffel-Tower are selected as the context, i.e. $C_R(\text{"Paris"}, \text{Eiffel-Tower}) = \{\text{"Eiffel Tower"}, \text{"France"}, \text{"Miniatures"}, \text{"Eiffel"}, \text{"Eyeful"}, \text{"Big"}\}$. The bag-of-words representation of the tag "Paris" of Eiffel-Tower will be $B_R(\text{"Paris"}, \text{Eiffel-Tower}) = \{(\text{"Eiffel Tower"}, 1), (\text{"France"}, 1), (\text{"Miniatures"}, 1), (\text{"Eiffel"}, 1), (\text{"Eyeful"}, 1), (\text{"Big"}, 1)\}$.

3.1.2 Tag Context (T)

In case of Tag Context, we select all the tags joint to the resources having the tag t , except the tag t itself. Tag Context can be defined as

$$C_T(t) = \{t' \in T \setminus \{t\} \mid (u, t, r) \in Y \wedge (u', t', r) \in Y \wedge u \in U \wedge u' \in U \wedge r \in R\} \quad (6)$$

For creating a bag of words representation (like (5)) using this context, we define $W_T(t, t')$ that represents the number of times tag t appears with tag t' .

$$W_T(t, t') = \left| \left\{ (u, t, r) \in Y \mid (u, t, r) \in Y \wedge u \in U \wedge u' \in U \wedge r \in R \right\} \right| \quad (7)$$

We can get the Tag Context of a tag t using $C_G(t)$ and for each tag t' in the Tag Context of tag t , we can get its number of occurrences with tag t using $W_T(t, t')$.

Consider that we want to classify the tag “Paris” of the image Eiffel-Tower. All tags of images having the tag “Paris” are selected as the Tag Context except the tag “Paris” itself. In example of Figure 4, Eiffel-Tower and Notre-Dame have the tag “Paris”, so all the tags of the images Eiffel-Tower and Notre-Dame are added to the context of the tag “Paris” except the tag “Paris” itself, and number of occurrences of each of these tags with tag t can be calculated using W_T . Thus, $B_T(\text{“Paris”}, \text{Eiffel-Tower}) = \{(\text{“Eiffel Tower”}, 1), (\text{“France”}, 2), (\text{“Miniatures”}, 1), (\text{“Eiffel”}, 1), (\text{“Eyeful”}, 1), (\text{“Big”}, 1), (\text{“Notre-Dame”}, 1), (\text{“Night”}, 1), (\text{“Lights”}, 1)\}$ is the bag-of-word representation constructed using Tag Context of the tag “Paris”.

3.1.3 Social User Context (SU)

In case of Social User Context of a tag t , we select all the tags used by a user u , except the tag t itself. Social User Context of tag t of user u can be defined as

$$C_{SU}(t, u) = \{t' \in T \setminus \{t\} \mid (u, t', r) \in Y \wedge r \in R\} \quad (8)$$

For creating a bag-of-words representation (like (5)) using this context, we define $W_{SU}(t, u)$ that represents the number of times tag t is used by the user u .

$$W_{SU}(t, u) = \left| \left\{ (u, t, r) \in Y \mid r \in R \right\} \right| \quad (9)$$

Consider that we want to classify the tag “Paris” of the image Notre-Dame that belongs to user B. All tags of images that belong to the user B are selected as the context except the tag “Paris” itself. In example of Figure 4, the images Notre-Dame and Cow belong to the user B, so all the tags of the images Notre-Dame and Cow are added to the context of the tag “Paris” except the tag “Paris”. Thus, $B_{SU}(\text{“Paris”}, \text{Notre-Dame}) = \{(\text{“Notre Dame”}, 1), (\text{“France”}, 2), (\text{“Night”}, 1), (\text{“Lights”}, 1), (\text{“Savoie”}, 1), (\text{“2001”}, 1), (\text{“Field”}, 1), (\text{“Cow”}, 1)\}$ is the bag-of-word representation constructed using social user context.

3.1.4 Social Group Context (SG)

In case of Social Group Context of tag t that is present in groups g , we select all the tags of all resources present in the same group g , except the tag t itself. The Social Group Context can be defined as

$$C_{SG}(t, g) = \{t' \in T \setminus \{t\} \mid (u, t', r) \in Y \wedge u \in U \wedge r \in R \wedge g \in \text{Group}(u, r)\} \quad (10)$$

where $\text{Group}(u, r)$ is a function which returns the groups that contain the user u and resource r .

For creating a bag-of-words representation using this context (like in (5)), we define $W_{SG}(t, g)$ that represents the number of times tag t appears in the group g .

$$W_{SG}(t, g) = \left| \{(u, t, r) \in Y \mid u \in U \wedge r \in R \wedge g \in \text{Group}(u, r)\} \right| \quad (11)$$

Consider that we want to classify the tag “Paris” of the image Eiffel-Tower that belongs to group 1. All tags of images present in group 1 are selected as the context except the tag “Paris” itself. In example of Figure 4, the images Eiffel-Tower and Cow are present in group 1, so all the tags of the images Eiffel Tower and Cow are added to the context of the tag “Paris” except the tag “Paris” itself. $B_{SG}(\text{“Paris”}, \text{group-1}) = \{(\text{“Eiffel Tower”}, 1), (\text{“France”}, 2), (\text{“Miniatures”}, 1), (\text{“Eiffel”}, 1), (\text{“Eyeful”}, 1), (\text{“Big”}, 1), (\text{“Savoie”}, 1), (\text{“2001”}, 1), (\text{“Field”}, 1), (\text{“Cow”}, 1)\}$ is the bag-of-word representation constructed using social group context.

4 Evaluation

In order to evaluate our system, we have used images, tags, user, and group information from Flickr website. We asked two persons to classify the data into four categories. We have then classified the same data set using T-KNOW in order to evaluate T-KNOW.

4.1 Experimental Setup

To organize tags into predefined categories, we have chosen four categories “Person”, “Location”, “Vehicle”, and “Organization”. To get ontologies related to these categories, we have searched Swoogle¹ [5] for general purpose ontologies and used the ontology OntoSem². For this ontology, we have used concepts and sub-concepts of $p1:vehicle$, $p1:organization$, $p1:place$, $p1:geopolitical-entity$, and $p1:human$ as categories. We have used a total of 932 concepts as categories out of this ontology.

After selecting the categories, we have gathered data from groups present at the Flickr website. Users can post their images to different groups on Flickr. One group usually contains images related to the topic of that group. For example, the vehicles group contains images of vehicles. We have searched for groups related to the topics (i) people, (ii) locations, and (iii) vehicles using the group search facility provided by Flickr, and then selected three groups from each topic. We have selected only those

¹ <http://swoogle.umbc.edu/>

² <http://morphus.cs.umbc.edu/aks1/ontosem.owl> (last accessed Mach 21, 2007)

groups which had at least 100 images and 25 members. The groups selected were candid_celebrity, 35212032@N00 (famous people), politicians, CarDirectory, classic_cars, vehicles, PraiseAndCurseOfTheCity, signcity, and cities. Out of these groups, only the “famous people” group had 27 members and 165 images, all other groups had at least 100 members and more than 500 images. We have then randomly selected 21 images from each of these nine groups. There were a total of 1754 tags in all of these 189 images.

We asked two persons K and S (human classifiers) to classify the tags. They did not have any kind of information about this research and method. They have classified all the tags regardless of the language and spelling mistakes, which has of course affected the results of T-KNOW because T-KNOW uses English patterns for identifying categories. For example, the users have classified the tags “Russia” and “Russland” (German word for Russia) as location, whereas T-KNOW was unable to identify “Russland”, as this is not an English word and hence is not supported by the pattern library used. A spreadsheet was provided to each human classifier with resources, tags, and links to the original Flickr images, Wikipedia, and Google. For example if a user finds a tag "Essen" (a German city as well as the German word for meal) and is unable to decide about its category, he can view the image (in which this tag is present) on Flickr website, if this image is not helpful to identify the tag, he can search it in Wikipedia¹, and still if it unclear, then he can find it in Google². Human classifiers (K and S) agreed upon classification of only 1166 tags out of 1754 tags.

4.2 Results

This section contains the results obtained by classifying tags using T-KNOW with different contexts and threshold values. Table 2 shows the number of tags and resources classified manually (by user K) and using T-KNOW with threshold of 0.0 and Social Group (SG) context.

Table 2. Number of tags and resources classified per category by User K and T-KNOW with Threshold = 0 and Social Group (SG) context

| Category | Resources | | Tags | |
|---------------------|-----------|----------|--------|----------|
| | User K | th=0, SG | User K | th=0, SG |
| Location | 139 | 155 | 519 | 485 |
| Organization | 39 | 54 | 89 | 67 |
| Person | 86 | 107 | 287 | 229 |
| Vehicle | 69 | 64 | 259 | 109 |
| Other | 155 | 177 | 600 | 864 |

¹ <http://en.wikipedia.org/wiki/Special:Search/essen>

² [http://www.google.com/search?num=100&hl=en&lr=&as_qdr=all&q="essen"&btnG=Search](http://www.google.com/search?num=100&hl=en&lr=&as_qdr=all&q=)

We have used F-measure and Cohen's Kappa for evaluation of our method. F-measure is a common measure in information retrieval, in case of tags classification we have computed F-measure as, if

$A = \text{set of correct classification by test}$

$B = \text{set of all classification by Gold Standard}$

$C = \text{set of all classifications by test}$

(In our evaluation, user K is the *gold standard*, and *test* is either user S or T-ORG)

then, we define Precision, Recall, and F-measure as

$$\text{Precision} = \frac{A}{C} \quad (12)$$

$$\text{Recall} = \frac{A}{B} \quad (13)$$

$$\text{F-measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (14)$$

Figure 5 displays the F-measure with user K defining the gold standard and T-KNOW using different threshold values and contexts and it also shows the F-measure of the classification of user K and user S (shown as a constant line).

Due to the possibility of classification that might occur just by chance, we have also calculated the Cohen's Kappa [4] between a user's classification and the system's prediction. Cohen's Kappa is defined as

$$K = \frac{P_0 - P_c}{1 - P_c} \quad (15)$$

where P_0 is the observed agreement between classifiers and P_c is the agreement occurred due to chance. If the two classifiers agree completely, then the value of Cohen's Kappa is 1. Figure 6 shows the Kappa values of the classification of user K and T-KNOW (with different threshold values and contexts) and it also shows the Cohen's Kappa value between the classifications of user K and user S (shown as a straight line).

4.3 Discussion

The task of organizing resources by classifying tags in a tagging system is not trivial. It is observed that two humans classifying the same data set might not totally agree with each other, as observed in the case of humans classifiers of user K and user S, the kappa value was 0.53, whereas this value would be 1 in case of complete agreement between classifiers.

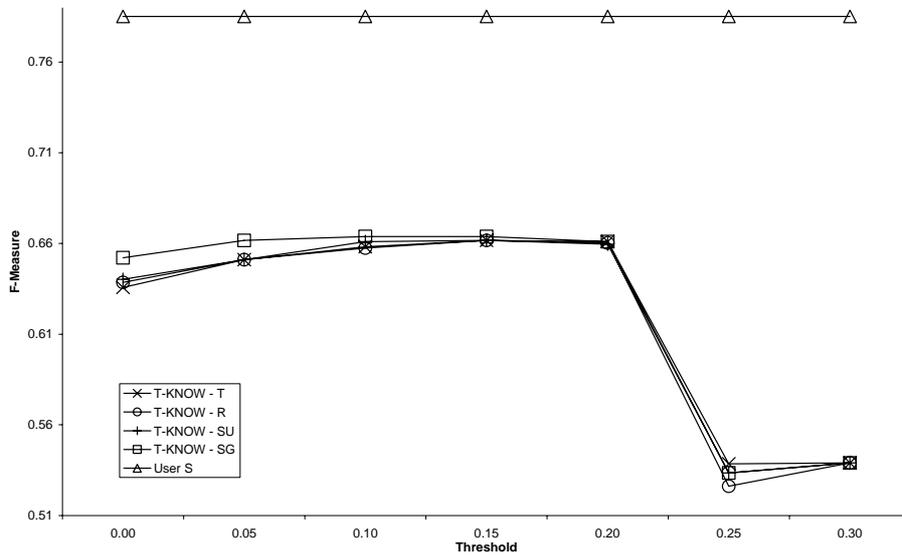


Fig. 5. F-Measure with user K defining the gold standard

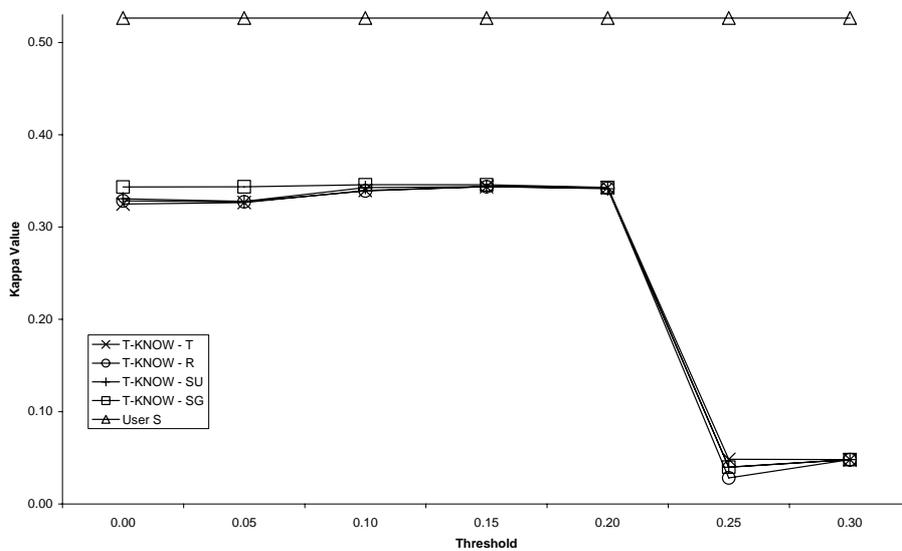


Fig. 6. Cohen's Kappa values for classification of T-KNOW and User S with user K defining the gold standard

Table 2 shows the number of tags and resources per category. The difference between number of resources or tags classified by different classifiers per category is small. As the average resources per user were 1.39 in the data set, the difference between F-measures of Resource (R) and Social User (SU) contexts is hardly visible. We believe

that if there are more resources per user, then the results of classification will be different for these context types. The best F-measure obtained was 0.66 with the context Social Group (SG) at thresholds of 0.10 and 0.15 and this small advantage was stable over other thresholds except 0.25. The F-measure is affected by the problem of classification by chance. Therefore we have calculated Cohen's Kappa [4] to measure the agreement between two users and between T-KNOW and user K. The majority class ("Other" in our case) scores zero in Cohen's Kappa [4]. F-measure lacks this property. The Cohen's Kappa between classification of users K and S was 0.53 (shown as a straight line in Figure 6), which shows the disagreement between the classifications of human users. Best kappa value for gold standard (user K) was 0.35 with Social Group (SG) context and using threshold of 0.10 or 0.15.

The results show that, the different approaches for selecting a context are statistically not significantly different. Keeping in view the small difference between different approaches, Social Group (SG) context has given overall better results as compared to other contexts. This is because the tags which are chosen as context belong to the same type of resources/images (as a group mostly contains same type of resources). In case of other contexts, tags of the resources with different subjects might be selected as context, which might affect the similarity measure.

5 Related Work

Tagging systems are becoming popular and more people are using them, especially in a social environment. A general overview of tagging systems can be found in [6]. Schmitz et al. have formalized folksonomies and discuss the induction of association rule mining for analyzing and structuring folksonomies in [10]. A lot of work has been done to extract useful information using natural language patterns. Hearst has used lexico-syntactic patterns to extract hyponyms from large text corpora [7]. Our approach is based on the matching of such lexico-syntactic patterns. These linguistic patterns have been used by other researchers for semantic annotation. T-KNOW is based on the C-PANKOW system (see [2] and [3]), which uses lexical patterns along with Google for semantic annotation of web pages. We have used and formally defined context of tags for measuring similarity between Google search results and the tags, contextual information has also been used by others, like [1] has used contextual information in recommendation process. [9] has used context information and Google search for identifying color of an object, which helps in clustering of images.

6 Conclusion

We have presented T-ORG to organize resources in a tagging system. T-ORG uses T-KNOW for unsupervised classification of tags which exploits Google and linguistic patterns. We have proposed four ways to context of a tag. Experimental results show that the classification accuracy for this unsupervised method is indeed encouraging,

especially in the light of the low agreement between the classifications done by two humans. Subsequent user experiments should show whether such classifications help to improve the user experience in a system like /facet [8].

Acknowledgments. We would like to acknowledge Higher Education Commission of Pakistan and German Academic Exchange Service (DAAD) for providing scholarship and support to Rabeeh Abbasi for conducting his PhD. This work has been partially supported by the European project “Semiotic Dynamics in Online Social Communities” (Tagora, FP6-2005-34721). We also thank Dr. (med.) Saadullah Abbasi and Ms. Katrin Michels for classifying the tags manually.

References

- [1] Adomavicius, G., Sankaranarayanan, R., Sen, S., and Tuzhilin, A. 2005. Incorporating contextual information in recommender systems using a multidimensional approach. *J-TOIS*, 23(1), 103–145.
- [2] Cimiano, P., Handschuh, S., and Staab, S. 2004. Towards the self-annotating web. In *Proceedings of the 13th international Conference on World Wide Web (New York, NY, USA, May 17 - 20, 2004)*. WWW '04. ACM Press, New York, NY, 462-471.
- [3] Cimiano, P., Ladwig, G., and Staab, S. 2005. Gimme' the context: context-driven automatic semantic annotation with C-PANKOW. In *Proceedings of the 14th international Conference on World Wide Web (Chiba, Japan, May 10 - 14, 2005)*. WWW '05. ACM Press, New York, NY, 332-341.
- [4] Cohen, J. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1), 37-46
- [5] Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R. S., Peng, Y., Reddivari, P., Doshi, V. C., and Sachs, J. 2004. Swoogle: A semantic web search and metadata engine. In *Proc. 13th ACM Conf. on Information and Knowledge Management*.
- [6] Golder, S. and Huberman, A. B. 2006. The structure of collaborative tagging systems. *Journal of Information Science*, 32(2):198-208, April 2006.
- [7] Hearst, M.A. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th International Conference on Computational Linguistics*.
- [8] Hildebrand, M., Ossenbruggen, J., Hardman, L. 2006. /facet: A Browser for Heterogeneous Semantic Web Repositories. *International Semantic Web Conference*, 272-285.
- [9] Millet, C., Grefenstette, G., Bloch, I., Moellic, P.A., Hede, P. 2006. Automatically populating an image ontology and semantic color filtering. *International Workshop Ontoimage'2006 Language Resources for Content-Based Image Retrieval*, Genoa, Italy, pp 34-39.
- [10] Schmitz, C., Hotho, A., Jaschke, R., and Stumme, G. 2006. Mining association rules in folksonomies. *Proceedings of the IFCS 2006 Conference*.