Folksonomy Resources as a Data Source for the Social Data in Semantic Web

F. Echarte, J.J. Astrain, A. Córdoba, J. Villadangos

Dpt. de Ingeniería Matemática e Informática
Universidad Pública de Navarra
Campus de Arrosadía. 31006 Pamplona (Spain)
patxi@eslomas.com, {josej.astrain, alberto.cordoba, jesusv}@unavarra.es

Abstract. The increasing popularity of folksonomies has made them interesting as a tool to bridge the gap between Web 2.0 and the Semantic Web. This paper describes a modular and generic method (ACoAR) for the automatic classification of the resources tagged in a folksonomy, using semantic measures and a reduced number of relevant tags. Generic since it applies to both narrow and broad folksonomies, and modular since it allows the integration of different techniques and algorithms. ACoAR updates this classification as the folksonomy evolves. ACoAR is validated using a *del.icio.us* sample set to obtain a set of classification concepts. The accuracy of ACoAR to classify new resources is analyzed obtaining a well classification rate of 93% of the analyzed resources.

Keywords: Social data, folksonomies, ontologies, automatic classification

1 Introduction

Early this century, two substantial changes in the evolution of the Web appeared: i) Tim Berners-Lee advocated his vision of a Semantic Web [1]; and ii) Social Networks emerged as a useful and widely used tool which enable collaborative relationships among users (social data). Web goes through a significant evolution in the so-called Social Sites (Web 2.0), where user roles have been notably modified. Users lived a relevant evolution: from just information consumers to be content generators, defining in a great extent the usefulness and evolution of very different kind of webs. Users have agreed in the use of a great number of different collaborative tools such as Blogs, Wikis, and social networks like Flickr, Facebook, etc. These tools require a powerful, simple and easy-to-use mechanism to classify the information managed. This mechanism must be simpler and more flexible than those based on taxonomies, and less formal than those mechanisms used on the semantic web that are based on ontologies. The mechanism more widely used in current webs is the content tagging based on a no-controlled vocabulary. The term *Folksonomy* [2] is frequently used for this kind of collective annotation process in Web 2.0.

Several works focus on *bridging the gap* between the Semantic Web and Web 2.0. Semantic information from tags can be obtained using subsumption strategies [3,4], concentrating in finding groups of highly related tags [5, 6] or relating folksonomies with ontologies [7, 8]. The semantic information obtained allows the navigation among the tags of the folksonomy and then, the resource accessibility. Those proposals build tag classification systems often considering tag co-occurrences, and in some cases, external sources that allows obtaining either information from tags, either ontologies in order to describe their semantic. However, those tag classification systems do not consider fully the resource's semantics, because they only consider the tags assigned to a resource, but not the frequency which each tag is associated to the resource.

Some works [4, 9] deal with narrow folksonomies, where the related frequency is always the unit. While in a narrow folksonomy (like *flickr*) only the owner of a resource can tag it, in a broad folksonomy (like *del.icio.us*) anyone can tag anything. In most cases, the process of harvesting the information semantics is not automatically performed. Moreover, as folksonomies are dynamic systems which evolve as users introduce new annotations, some mechanisms are required to accommodate this evolution to the proposed classification systems.

We propose a resource centric method called ACoAR (*Automatic Classification of Annotated Resources*) [10] which ensures the automatical classification of resources. It uses a set of classification concepts -CCs- (a set of resources with similar semantics). These concepts are automatically obtained from the resources annotations taking into account both tags and the frequency with which a resource is tagged. This resource classification is performed using a relatively small subset of the tags of a folksonomy. We prove that this subset is valid enough to semantically group the resources of the folksonomy under classification concepts which represents the main topics of the folksonomy. Additionally, ACoAR is a dynamic and automatic method which updates the classification concepts as the folksonomy evolves with new annotations.

ACoAR is a generic and modular method. Generic, in the sense of it applies to both narrow and broad folksonomies, and modular since it allows the integration of different techniques and algorithms.

In order to validate our proposal we have obtained information about web pages from the broad folksonomy of *del.icio.us,* since *del.icio.us* is considered one of the world's leading social bookmarking service and contains a large set of resources and annotations.

The rest of the paper is organized as follows: Section 2 is devoted to describe the ACoAR method, explaining the creation of the classification concepts from an existing folksonomy and their evolution when new annotations arrive to the folksonomy; Section 3 describes the experimental results obtained; and finally, conclusions, acknowledgements and references end the paper.

2 Method Description

ACoAR is intended to provide the automatic classification of annotated resources. As depicted in Fig. 1 ACoAR has an initial task called CCs *Creation*, which classifies the folksonomy resources under a set of CCs that represent the main topics of the folksonomy. The task uses a *Dictionary* in order to improve the performance of the classification. The *Dictionary* contains the subset of tags which represent the semantics of the resources. Although folksonomies use to have a very large number of tags, some works [11, 12] show that annotations follow a *power law* distribution such that a reduced subset of tags represents (and preserves) the semantics of the resources.

The information of the folksonomy concerning tags and resources is used to build the initial set of CCs and to classify the folksonomy resources under them. The CCs *Evolution* task updates the classification as the folksonomy evolves with new annotations (new or existing resources, users and tags). While the initial task runs only once (at startup), the second one runs whenever a buffer is fulfilled with new annotations. Both tasks, CCs building and evolution, are performed using representations of the resources according to the *Dictionary* tags assigned by users.

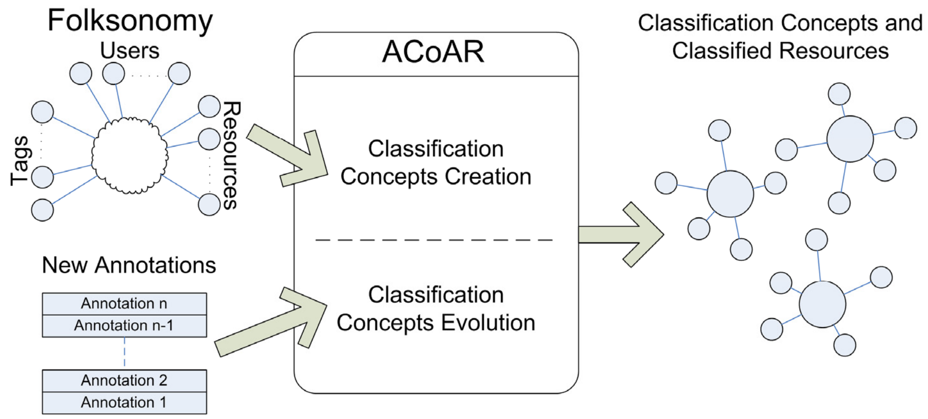

Fig. 1. Method Description: it shows the creation of the classification concepts and their evolution when new annotations are created in the folksonomy

Let *F* be a folksonomy *F:=* <*T,R,U,Y*> where *T* represents the set of tags, *R* is the set of resources, *U* is the set of users and $Y \subseteq U \times T \times R$ is a ternary relation representing the set of annotations, can be extended to represent the dictionary, the classification concepts and the relations among resources and concepts, using a tuple $Model_{ACoAR}$:= <*T,R,U,Y,D,C,Z*>, where *T, R, U* and *Y* define the folksonomy. $D \subseteq T$ is a dictionary which contains the most relevant tags of the folksonomy, *C* is the set of classification concepts (CCs) in which folksonomy resources are classified. And finally, $Z \subseteq R \times C$ is a binary relation between a resource *R* and a classification concept *C*, representing that a resource is classified under a concept.

We consider that a resource has converged when its distribution of tags (tags belonging or not to the dictionary) converges rapidly to a remarkably stable heavy-tailed distribution. In order to evaluate this convergence we define a threshold, based on the annotations number. Resources that have converged are encoded using the vector space model (VSM). The corpus consisting of R converged resources and D dictionary tags is represented by a matrix $A=(a_{ij}) \in \Re^{RxD}$. Each row vector a_i corresponds to a $resource_i$ and each column vector corresponds to a tag_j of the dictionary. Each a_{ij} represents the number of annotations that relates tag_j to $resource_i$. Although there exist many other representation methods that could also be considered, such as TF-IDF, the election of this non normalized method is due to its simplicity since resources can be directly encoded according to their annotations without requiring any additional calculus, and also due to the easiness of the CCs' representation, since it only requires a summatory for each tag of the dictionary over the resources classified by the concept.

2.1 CCs Creation

In order to create the CCs ACoAR performs several steps, as depicted in Figure 2. Initially ACoAR builds automatically the dictionary for a given folksonomy. The behavior of the method is determined by the size of the dictionary, since a large number of tags in the dictionary implies high computational costs when comparing resources. The objective is to accurately represent the semantics of the resources using the minimum number of tags. Dictionary tags can be selected following different criteria like the tags most frequently used, the tags representing the largest number of resources, etc. Tags can be also be filtered to discard misspellings, to find syntactic variations, etc.

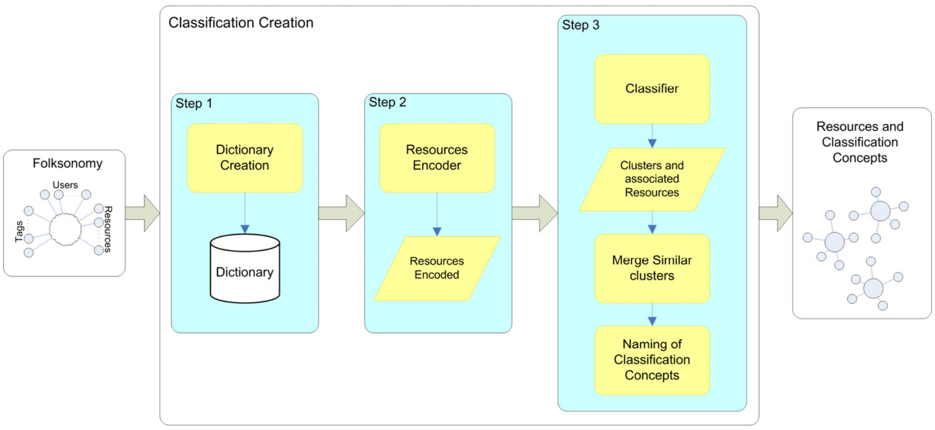

Fig. 2. Classification concepts (CCs) creation, process description

In the second step, ACoAR encodes the folksonomy resources which annotations have converged, obtaining the representative vector of each resource. The rest of resources will be encoded when the folksonomy evolves and their annotations converge. Table 1 shows a folksonomy example. Vector $a_1 \equiv (40,0,0,0,5,0,0,0,32,0,0,0,0,0,2,0,0,0,0,0)$ corresponding to resource r_1 (a_1 is the representation vector of $resource_1$) and indicates that r_1 has been annotated 40 times with the tag *ajax*, 5 with *css*, 32 with *javascript* and 2 times with *programming*.

Table 1. Folksonomy example with dictionary tags and resources

Tag/Resource		r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}
t_1	*"ajax"*	40									
t_2	*"art"*		21				24				
t_3	*"blogs"*			13				34			
t_4	*"blogging"*			4				21			
t_5	*"css"*	5					4				
t_6	*"database"*					5					
t_7	*"design"*		7				8				
t_8	*"java"*			5							
t_9	*"javascript"*	32								45	
t_{10}	*"lisp"*										16
t_{11}	*"museum"*		9								
t_{12}	*"mysql"*				10						
t_{13}	*"oracle"*					8					
t_{14}	*"php"*										
t_{15}	*"programmin"*	2			33	34			25	5	25
t_{16}	*"python"*				15				18		
t_{17}	*"socialweb"*							5			
t_{18}	*"sql"*				12	12					
t_{19}	*"twitter"*							10			
t_{20}	*"xml"*				5				12		

The third step is in charge of the creation of the CCs and the classification of the resources. The classifier provides a set of clusters (each of them corresponding to a CC) and the resources are classified under those clusters. Although many interesting clustering techniques can be considered [13] to create the clusters, k-means technique provides satisfactory results in our experiments (see Section 3). Each cluster consists on a centroid and its associated resources. The classifier compares the resources with the centroids of the clusters in order to obtain the most appropriate cluster for each resource. There exist many similarity measures that can be used to compare resources. For example, the cosine similarity (well known on Information Retrieval [14]) measures the angle among their vector representation without requiring any normalization.

Let illustrate the calculus of the centroids with the aid of Table 2. Using a k-means algorithm with a *k* value of four, we obtain the clusters represented by centroids c_1, c_2, c_3 and c_4. Each centroid is obtained, in our case, adding the representation vectors of the resources ($c_1 = a_1 + a_9$). Then, $c_1 \equiv (40,0,0,0,5,0,0,0,77,0,0,0,0,0,7,0,0,0,0,0)$ is the addition of $(40,0,0,0,5,0,0,0,32,0,0,0,0,0,2,0,0,0,0,0)$ and $(0,0,0,0,0,0,0,0,45,0,0,0,0,0,5,0,0,0,0,0)$, the representation vectors of r_1 and r_9 (a_1 and a_9 respectively).

Table 2. Clustering results for the related folksonomy

Cluster centroid	Resources	Cluster Centroid
c_1	r_1, r_9	(40,0,0,0,5,0,0,0,77,0,0,0,0,0,7,0,0,0,0,0)
c_2	r_2, r_6	(0,45,0,0,4,0,15,0,0,0,9,0,0,0,0,0,0,0,0,0)
c_3	r_3, r_7	(0,0,47,25,0,0,0,5,0,0,0,0,0,0,0,0,5,0,10,0)
c_4	r_4, r_5, r_8, r_{10}	(0,0,0,0,0,5,0,0,0,16,0,10,8,0,117,33,0,24,0,17)

ACoAR evaluates the similarity among the clusters in order to merge those clusters with a high degree of similarity. Although many different measures can be used, in this case and due to its simplicity, we use the cosine similarity measure to compare the centroids of the clusters. Table 3 illustrates the semantic similarity among centroids. In this example, the similarity between c_1 and c_4 is very low (0.07405) and then c_1 and c_4 are not merged in a same cluster. Two clusters having a high similarity between them (close to the unit) would be merged producing a new cluster if a certain threshold is exceeded. The centroid of the resulting cluster is the sum of both centroids ($c_{new}=c_1+c_4$). Then c_{new} replaces c_1 and c_4.

Table 3. Semantic similarity among centroids

Similarities	c_1	c_2	c_3	c_4
c_1	1.00000	0.00473	0.00000	0.07405
c_2	0.00473	1.00000	0.00000	0.00000
c_3	0.00000	0.00000	1.00000	0.00000
c_4	0.07405	0.00000	0.00000	1.00000

Finally, the method creates a classification concept for each cluster, and assigns them a name. For such purpose, the method considers those tags having the greater values in the cluster centroid vector. Table 4 shows the CC assigned to each cluster. When the frequency of a certain tag is greater than the frequency of the rest of tags (this tag has a high relevance), then this tag becomes the classification concept associated to the cluster. When the most significant tags have similar frequencies and their frequency is amply high in comparison to other tags, the CC name is obtained by merging those tags. Other alternatives like looking for syntactic variations, prefixes and suffixes, or even looking up external resources like Wordnet or Wikipedia, or even ontologies, could be used to assign more adequate names to the CC*s*.

Table 4. Classification concepts created from clusters

Cluster	Cluster main tags	Classification Concept
c_1	javascript:77, ajax:40, programming:7, css:5	Javascript & Ajax
c_2	programming:117, python:33, sql:24, xml:17, lisp:16, mysql:10, oracle:8, database:5	Programming
c_3	art:45, design:15, museum:9, css:4	Art
c_4	blogs:47, blogging:25, twitter:10, java:5, socialweb:5	Blogs

2.3 CCs Evolution

Once the CC*s* have been created and the resources have been classified under them, is necessary to define how to update these CC*s* when the folksonomy evolve with new annotations. ACoAR also considers that users can make changes in their existing annotations, for example deleting a tag assigned to a resource. New folksonomy annotations are accumulated into a buffer. When this buffer is filled ACoAR processes the annotations like Fig. 3 depicts. A four steps process is applied, processing the annotations and updating the classification information.

A folksonomy resource may belong to one of these three sets: i) *pending* when the resource has not yet received enough annotations to have converged, ii) *converged* when the resource has converged but it is not yet classified, and iii) *classified* when the resources has converged and it is classified under a CC. After the CCs *Creation* phase, all the folksonomy resources belong to *pending* or *classified* sets.

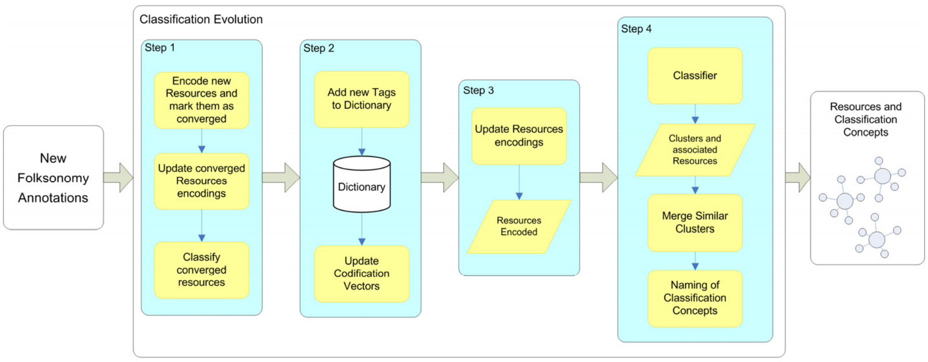

Fig. 3. CCs evolution based in the new annotations on the folksonomy

The first step is intended to classify new resources under the existing CCs. It process annotations and encodes *pending* resources that have reached the minimum annotations threshold, assigning them to *converged* set. It also updates the representation vectors of *converged* resources that have received new annotations.

After that, *converged* resources are processed to find the most similar CC*,* comparing their representation vectors. When the similarity between a resource and a CC reaches a minimum threshold, the resource is classified under CC and it is moved to *classified* set. When the similarity doesn't reach the threshold the resource remains in the *converged* set. Once a resource is assigned to *classified* set, ACoAR doesn't allow moving it to other set.

The second step checks the annotations and the criteria used to create the *Dictionary* and update it when necessary. If the *Dictionary* is modified, ACoAR updates the representation vectors of resources and CCs according to the new situation.

The third step updates the resources representation vectors that belong to the *classified* set with the new annotations they have received.

The fourth step is equivalent to CCs *Creation – Step 3*. Its objective is to update the classification of the *classified* resources taking into account the new information obtained from the annotations. However in this case, the classification is much simpler than in the CCs *Creation,* because the clustering algorithm can initialize the clusters with the previously obtained classification. When using *k-means* the convergence is faster because the initial centroids are not randomly created, they are obtained from the CCs representation vectors.

3 Experimental results

We have used data retrieved from *del.icio.us* to evaluate the proposed method. We have obtained, with the aid of two page scrapers, information about web pages annotated by users using the *Recent Bookmarks* page[1]. The first scraper processes the recent bookmarks page looking for resources bookmarked by a minimum number of 100 users, and stores the url of the resources. We consider this value because it has been empirically proved in [12] that the frequency of each tag in a resource stabilizes after the first 100 annotations. The election of this value as the convergence criteria for the annotations of a resource implies the consideration of the resources annotated by a minimum of 100 users (maybe including more than 100 annotations, if users assign two or more tags to a resource). The selected bookmarks are used by the second scraper to retrieve their information from the "*Everyone's Bookmarks*" pages in *del.icio.us*, that consist of a 40 pages listing with the most recent bookmarks and a summary with the most 30 frequently assigned tag to the resource. These scrapers were used in April 2009 obtaining a total of 25,251 resources, with 93,247,161 annotations, 1,039,796 users and 584,722 tags.

We have built two datasets[2] from the information retrieved in order to evaluate the *CCs Creation* and the accuracy of the method when classifying *converged* resources. The first dataset (ds_1) consists of 24,251 resources and their associated annotations, and it is used to create the classification concepts using k-means. The second dataset (ds_2) consists of 1,021 new resources and their annotations, and it is used to validate the accuracy of the classification, since these resources are classified under the CCs created from ds_1. This validation has been performed manually by ten experts (distinct from authors) checking if the proposed CCs are appropriate to the resources.

We have created the dictionary considering those tags with a minimum number of 500 annotations, obtaining a cardinality of 4,085 tags. It is interesting to note that although these tags represent less than the 0.1% of the retrieved tags, they are used in more than the 95% of the retrieved annotations. In such way, we consider they represent the semantic of the folksonomy resources in a great extent. Once defined the dictionary, the first experiment has been performed in order to evaluate the creation of the classification concepts using the k-means algorithm. The value of *k* has been

---

[1] http://delicious.com/recent/?setcount=100&min=100
[2] http://www.eslomas.com/index.php/publicaciones/sdow09/

determined using the expression $k = \sqrt{n/2}$, where n=24,251 and then k = 110. Nevertheless, many other techniques like [13] can be applied in order to optimize this value. The initial centroids have been defined randomly, since any *a-priori* knowledge is considered. The clustering has been performed using several parallel processes. A total amount of 8 processes have been used over a four *Intel Core 2 Duo* processors and the algorithm has converged in 34 iterations.

The 110 resulting clusters have been analyzed to detect possible equivalent clusters, comparing them and merging those clusters with a similarity greater than 0.75. As result, we obtain 104 clusters.

Table 5. Example of Classification Concepts created based in their most frequent tags

Tag 1	Tag 2	Tag 3	Tag 4	Classification Concept
astronomy (32,683)	space (27,018)	science (24,950)	nasa (10,651)	Astronomy, Space and Science
photography (401,387)	photos (220,623)	photo (199,908)	images (126,910)	Photography and Photos
programming (132,858)	development (38,452)	reference (34,160)	code (24,917)	Programming
social (132,977)	web2.0 (104,364)	community (88,841)	socialnetworking (65,240)	Social, Web 2.0, Community and Socialnetworking

Finally, a name has been assigned to each cluster to create the CCs. Names have been created using the most relevant tag and concatenating the next ones when their weight is greater than the 50% of the most relevant tag weight. So, a cluster which the two most relevant tags *php* (weight 127,427) and *programming* (weight 39,743), has been named as "*Php*". Table 5 shows some examples of the classification concepts created.

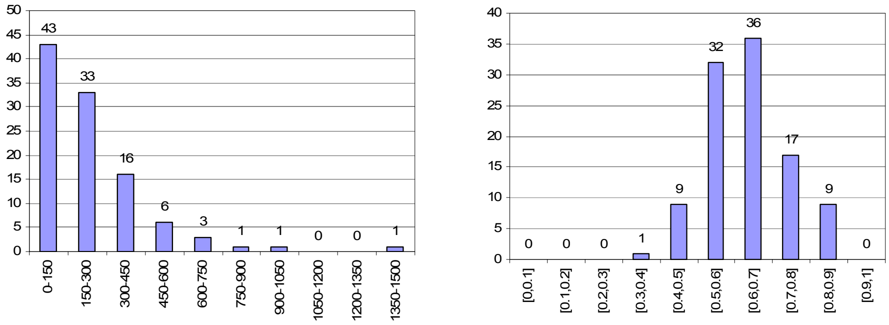

Fig. 4. Number of concepts attending to the number of resources they group (left) and to the average similarity among concepts and resources classified under (right).

Fig. 4 (left) shows the number of classification concepts obtained attending to the number of resources they group. Most of concepts group less than 300 resources, and there exists a concept with a large number of resources (*Art and Design*). Fig. 4 (right) shows the average similarity between resources and concepts. Most of the concepts have an average similarity with their resources greater than 0.5.

The created classification concepts have been used to evaluate the accuracy of the method when the folksonomy evolves and new resources must be classified (*converged* resources). The 1,021 resources in ds_2 have been encoded and the classifier has been applied, obtaining the most similar concept for each resource, the similarity between them, and the similarity with the rest of concepts in order to evaluate the results. It is interesting to note that resources are classified according to the users of the folksonomy knowledge. There exist resources that may correspond to marketing and publishing companies that have been classified under "*Art and Design"*, since *del.icio.us* users bookmark these pages due to their impressive design, or because they may use them as an inspiration for other designs.

Table 6 shows a summary of the results. The number of well classified resources by ACoAR is 951 (93.14%) versus 70 (6.86%) misclassifications. It also shows the average similarity among CCs and the resources classified under them, and the average difference between the similarities of the two most similar CCs (*delta*) to each resource. Considering for example four CCs and a resource r_i to be classified, with a similarity of 0.05 with c_1, 0.50 with c_2, 0.80 with c_3 and 0.00 with c_4, the value of *delta* is 0.30 (0.80-0.50). High *delta* values indicate that the concept suggested by ACoAR has a high similarity with the resource and a low similarity with the next *most similar* concept. Low *delta* values indicate that the resource has a similarity value very similar with the two closest classification concepts, so the classification could be erroneous or may be there is not enough information to classify the resource.

Table 6. Classifications results, including average values for similarity and *delta*

	Results	Average Similarity	Average Delta
Well classified	951 (93.14%)	0.621868	0.242415
Misclassified	70 (6.86%)	0.483300	0.100786

Table 6 shows that the average similarity of the well classified resources is greater than the obtained for the misclassified resources. The same happens with the *delta* value. ACoAR is able to provide classification concepts with a great similarity value for well classified resources and furthermore, the second classification concept has an average similarity of 0.38 (0.62–0.24), that is relatively low.

As described in the method description, we can define a threshold function to adjust the classification and to reduce the classification errors. This function allows defining some minimum conditions that must be fulfilled to classify the resource under the suggested classification concept, like a minimum similarity value, a minimum *delta*, or a combination of both of them.

Fig. 5 (left) depicts the number of resources well classified and misclassified according to the similarity measure between resources and the suggested CCs. Fig. 5

(right) shows the same information based in a threshold function using *delta* values. Both figures show that the election of a threshold considering the *delta value* is a good choice since most of misclassifications belong to the interval [0.0,0.1], while the misclassifications considering a threshold based on the similarity value are more homogenously distributed. Consequently, the selection of a threshold function based on a minimum *delta* value of 0.10 would reduce the classification errors from 70 to 23. As a counterpart, the 238 resources well classified with a *delta* value under 0.1 would be assigned to *converged* set until they could be classified more accurately. Therefore, the results of the classification would be 285 resources not classified (*converged* set), 713 resources well classified and 23 resources misclassified, representing a correct ratio of 96.88% (713 out of 736).

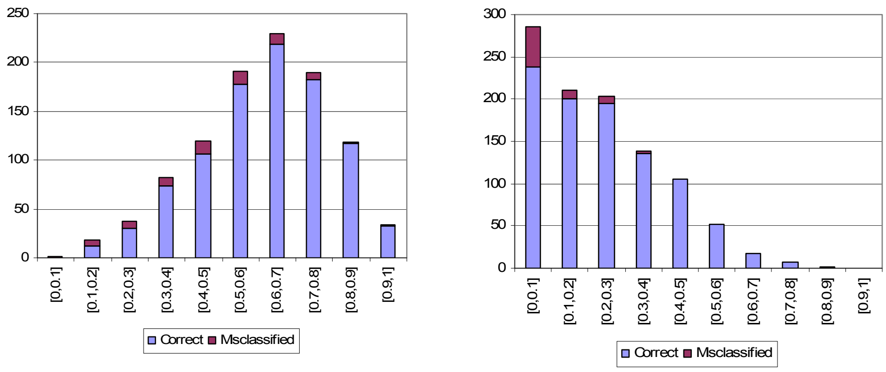

Fig. 5. Results obtained attending to the similarity value between resource and suggested CC (left) and *delta* values (right).

4 Conclusions

Social tagging systems are nowadays the preferred way to classify information in the Web 2.0 sites. As their popularity increases, many works try to solve some of their intrinsic problems derived from their uncontrolled vocabulary. In this paper we have proposed a modular and generic method, called ACoAR, that automatically: i) creates a classification based in the semantics of the resources of a folksonomy using a relatively small subset of the existing tags, and ii) allows the evolution of this classification when the associated folksonomy evolves with new annotations. Due to its modularity ACoAR allows the use of different clustering algorithms, as well as different similarity measures. ACoAR[3] allows browsing folksonomies by means of the semantics of its resources increasing the chance of finding interesting results. ACoAR can minimize the gap between Web 2.0 and the Semantic Web defining an ontology from the CCs created, and evolving this ontology as the folksonomy does.

[3] http://acoar.eslomas.com/demo

We have performed an evaluation of the proposed method using *del.icio.us*. Experimental results consider an initial set of classification concepts that classify folksonomy resources, and provide a 93.14% well classification rate when the folksonomy evolves and the method classifies new folksonomy resources without using any threshold. This rate may increase using an adequate threshold based on the similarity values between resources and concepts, or based on the difference between the similarities of the CCs (*delta*). ACoAR is a modular method which allows the integration of different techniques and algorithms. Some of these modules are: dictionary generation, convergence with different criteria, cluster algorithms, similarity measures, etc.

Acknowledgements
Research partially supported by the Spanish Research Council under research grants TIN2006-14738-C02-02 and TIN2008-03687.

5 References

1. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American (2001)
2. Vander Wal, T.: Folksonomy, http://vanderwal.net/folksonomy.html
3. Zhou, M., Bao, S., Wu, X., Yu, Y.: An Unsupervised Model for Exploring Hierarchical Semantics from Social Annotations. In: Sixth International Semantic Web Conference (ISWC 2007), LNCS, vol. 4825, pp. 680-693, Springer Verlag (2007)
4. Schmitz P.: Inducing ontology from flickr tags. In: Collaborative Web Tagging Workshop, WWW. Edinburgh, Scotland (2006)
5. Heymann, P., García-Molina, H.: Collaborative Creation of Communal Hierarchical, Taxonomies in Social Tagging Systems. TR Stanford InfoLab 2006-10 (2006)
6 Begelman G., Keller P., Smadja F.: Automated tag clustering: Improving search and exploration in the tag space. In: Collaborative Web Tagging Workshop, WWW (2006)
7. Specia, L., Motta, E.: Integrating Folksonomies with the Semantic Web. In: ESWC 2007. LNCS, vol. 4519, pp. 503-517, Springer Verlag (2007)
8. Passant, A., Laublet, P.: Meaning Of A Tag: A collaborative approach to bridge the gap between tagging and Linked Data. In: Workshop on Linked Data on the Web, WWW. CEUR-WS, vol. 369 (2008)
9. R. Abbasi, S. Staab, Cimiano, P.: Organizing resources on tagging systems using t-org. In 4th European Semantic Web Conferences, pp. 97-110 (2007)
10. Echarte, F., Astrain, J. J., Córdoba, A., Villadangos, J., Labat, A.: ACoAR: a method for the automatic classification of annotated resources. In Fifth International Conference on Knowledge Capture (K-CAP '09), pp. 181-182, ACM, New York (2009)
11. Michlmayr, E.: A Case Study on Emergent Semantics in Communities. In: Workshop on Social Network Analysis, International Semantic Web Conference. Galway, Ireland (2005)
12. Golder, S.A., Huberman, B.A.: The Structure of Collaborative Tagging Systems. Journal of Information Science, vol. 32, no. 2, pp. 198-208 (2005)
13. Maitra, R.: Initializing Partition-Optimization Algorithms. Computational Biology and Bioinformatics, IEEE/ACM Transactions vol.6, no.1, 144-157 (2009)
14. Salton, G.: Automatic Text Processing: the transformation, analysis, and retrieval of information by computer. Addison-Wesley Longman Publishing, Boston, USA (1989)