Tag as you like…, we can understand you!

Samia Beldjoudi[1], Hassina Seridi[2], Catherine Faron-Zucker[3],

[1, 2] Laboratory of Electronic Document Management LabGED, Badji Mokhtar University Annaba, Algeria
[3] KEWI Group, I3S Laboratory, University of Nice, France
[1]bsamia_2008@yahoo.fr, [2]Seridi@labged.net,
[3]catherine.faron-zucker@unice.fr

Abstract. Nowadays, the approaches that combine semantic web ontologies and web 2.0 technologies constitute a significant search field that attracted the interest of a best part of searchers. We will present in this paper an original approach concerning a technology that has recognized a great popularity in these recent years, we talk about folksonomies. Our aim in this contribution is to give birth to a new kind of reasoning concerning the Social Semantic Web technologies in order to see how we can overcome the problem of tags' ambiguity in folksonomies automatically even when we choose representing these latter with ontologies. In this work we'll also see how we can enrich any folksonomie by a set of pertinent data that can improve and facilitate the resources' search in these systems; all this with tackling another problem from it suffer this technology, we speak about the problem of spelling variations.

Keywords: Folksonomies, Ontologies, Web 2.0, Semantic Web, Tags' Ambiguity, Spelling Variations, Enrichment.

1 Introduction

Web 2.0 and Semantic Web constitute today a new line of search, its aim is the combination between the techniques and the principles of these two fields in order to constitute a real Social Semantic Web. These days the Web 2.0 is imposed as the web of users; with it the Net users become also creators of data and not only consumers like in the case of Web 1.0. Moreover Web 2.0 offers to users more possibilities of expression for achieving their goals. These possibilities allowed each one to create, annotate, share and make public what he found interesting. Among the powerful technologies of Web 2.0, we find folksonomies, this term has recently appeared on the net to describe a system of classification derived from the practice and method of collaboratively creating and managing tags to annotate and categorize content; this practice is also known as collaborative tagging, social classification, social indexing, and social tagging [12]. The basic principle of this concept focused on three main elements: the user, the resource and the tag, where the combination of these three notions gives delivery for a search tool based on the annotation of web resources by users whom employed a set of words called tags.

The Semantic Web in turn plays an important role in the development of resources retrieval on the web. Ontologies that constitute the backbone of this tendency contribute significantly in solving the problems of semantics during the definition and the search of information. However even with the strengths and the benefits of folksonomies and ontologies; except that their combination does not guarantee the resolution of any problem from it they suffer all the both. As an example we cite the main problem in folksonomies which is the problem of tags' ambiguity (Polysemy). Also the variations in writing the same concept (Spelling variations or Synonymy) can pose some problems during the search phase. Therefore the 'resources' search within folksonomies needs some techniques of inference and reasoning to improve more and more the quality of the results obtained in these systems. Also the use of ontologies in the majority of the previous approaches requires an expert who must control the relations between the different elements of ontology. And even in the cases where this procedure is done automatically; we find in general that the problem of ambiguity was solved for a specific domain or according to a set of constraints previously given depending on the relations existing in the ontology; and not for any situation. Also the lack of an automatic enrichment of folksonomies is a major challenge that we will try to tackle it in our approach SSFO (Semantic Social Folksonomy with Ontology). So our goal in this contribution is to show how we can exploit the power of social interactions between the members in folksonomy in order to extract the meaning of terms and overcome the problems of tags' ambiguity and spelling variations. Also we will try to show how we can use the principle of rules-based systems with ontologies for helping our system to enhance automatically the folksonomy by facts can increase the amount of data available within our system with relevant information which can facilitate the resources retrieval and optimize the time expended during this process.

Our paper is organized as follows: Section 2 presents a quick overview about the main contributions attached to our search field; in Section 3 we will detail the design of our approach. After in Section 4 we move to the experimental phase in order to measure the performance of the SSFO approach and discuss the obtained results. Conclusion and future works are discussed in Section 5.

2 Related Work

In this section we will show the principals contributions concerning the subject of folksonomies. After a profounder study we have saw that the majority of propositions are oriented toward the search of semantic relationships among the folksonomy's terms in order to surmounting the problems of semantics' lack in these systems. Because the article's space doesn't allow us citing all these contributions, we will put the point on the famous works which try to reduce tags' ambiguity problem and especially those aimed to extract the semantic links between folksonomy's terms using ontologies.

Mika [8] proposed to extend the traditional bipartite model of ontologies to a tripartite one: that of folksonomies, where instances are keywords used by the actors of the system in order to annotate web resources. In this article, Mika focuses on social network analysis in order to extract lightweight ontologies and the exploitation of the strength of these ontologies in order to explicit semantics between the terms used by the users (actors).

In another work, Gruber [4] argued that there is no contrast between ontologies and folksonomies, and therefore recommended to build an "ontology of folksonomy".

According to Gruber, the problem of the lack of semantic links between terms in folksonomies can be easily resolved by representing folksonomies by ontologies.

Specia and Motta [11] in their turn have preferred the use of ontologies to extract the semantics of tags. Their approach consists in building tags clusters, and then trying to identify the possible relationships between tags in each cluster. The authors have chosen to use ontologies available on the semantic web in order to express the correlations which can exist between tags. A more detailed attempt to mechanize this method is described in [1].

In the same trends; Buffa et al. [2] presented a semantic web application baptized SweetWiki reconciling two trends of the web: a semantically augmented web and a web of social applications where every user is an active provider as well as a consumer of information. The goal here is to exploit ontologies and semantic web models to improve the notion of social tagging. According to the authors, tagging remains easy and becomes both motivating and unambiguous.

The niceTag project of Limpens et al. [5] is focused on this same principle: the use of ontologies to extract semantic links existing between tags in a system. In addition, this project has introduced the idea of exploiting interactions between users and the system to validate or invalidate automatic treatments carried out based on tags. The authors have proposed methods to build lightweight ontologies that can be used to suggest terms semantically close during the search of documents guided by tags.

Pan et al. [9] aimed at reducing the problem of ambiguity in tagging. They proposed to extend the search of tags in a folksonomy by using ontologies. They defended this principle of extension of the search in order to avoid bothering the users with the rigidity of ontologies. More precisely, they concatenated ambiguous terms with other ones in order to increase the precision of the results of a keyword-based search.

Scott and Hubermann [10] analyzed the structure of collaborative tagging systems as well as their dynamic aspects. Specifically, they discovered regularities in user activity, tag frequencies, kinds of tags used, bursts of popularity in bookmarking and a remarkable stability in the relative proportions of tags within a given URL. They also discussed why it is difficult to retrieve contents (which in our case play the role of resources) in a folksonomy. In particular they highlighted some problems like synonymy and polysemy.

Markines et al. [6] discussed how to extend and adapt traditional notions of similarity to folksonomies, and which measures are best suited for applications such as navigation support, semantic search, and ontology learning. The authors builded an evaluation framework to compare various general folksonomy-based similarity measures derived from established information-theoretic, statistical, and practical measures.

We also investigated the works relative to the recommendation of relevant resources. De Meo et al. [3] proposed to recommend a set of resources to enrich user profiles, a user profile being represented by the list of tags involved in his query. They expand queries to recommend resources to users performing a keyword-based search; in order to enhance their profiles. A user query is enriched with tags discovered through the exploration of the two graphs TRG (Tag Resource Graph) and TUG (Tag User Graph). According to the authors, this enrichment improves both the strategy of recommender systems and that of collaborative filtering and content-based filtering systems.

To sum up, most of the works relative to folksonomies aim to bring together ontologies and folksonomies as a solution to the tags' ambiguity problem and that of the lack of semantic links between tags. The approaches summarized in this section showed that the social nature of resources sharing is not in contradiction with the possibilities offered by ontology-based systems. But the rigidity which characterizes ontologies and the need of an expert who must control and organize links between terms as in [4] seems a little cumbersome and too much expensive. Even the structures extracted automatically as in [7] still suffer from the ambiguity of concepts.

Regarding the work of [11], the use of semantic web ontologies for extracting relationships between terms is not sufficient, because the semantic web does not include enough specific domain ontologies and this will push the problem further.

Also the expertise of users which was introduced in [5] is characterized by the complexity of its exploitation when we try as much as possible to avoid a cognitive overload, to limit the necessary effort for the formalization of this expertise.

Based on these observations, we started our trial to improve a little this technology and give a new view concerning the combination between folksonomies and ontologies.

3 The Approach: Semantic Social Folksonomy with Ontology (SSFO)

The newness presented via this approach appears from the fact that our idea is not limited to a specific folksonomy or a particular ontology. But the challenge here is to present a new technique can be applied to any folksonomy represented by any type of ontology (we want say that all the contributions which have supported the use of ontologies for emerging the semantics in folksonomies; can use our technique behind their approach for improving some points in those latter).

Also the new which is offered in this work can be summarized by the fact that we have based our proposal on the strength of the social aspect of folksonomies in order to extract the words' semantics and overcome the problem of tags' ambiguity. Note that the aim of our approach SSFO is to introduce both the semantics and the social aspects in folksonomies in order to enable any user in the system to retrieve relevant web resources close to his preferences.

Moreover our objective via this contribution is to provide a simple and optimal method can be adapted to any user profile in order to give to each member in the community pertinent resources when he makes a search by keywords. This can let the results be modified according to the preferences of each user in the system. Therefore we have suggested assigning to each proposed resource a percentage of recommendation as follows: 'strongly recommended', 'recommended' or 'weakly recommended ' according to the degree of similarity among users.

Here we will not limit or specify a particular way to represent a folksonomy by an ontology, but the essential for us is to show how we can produce a technique can help any ontology already proposed for representing a folksonomy to overcome the problem of tags' ambiguity automatically without the need of an expert who must do the control of different relations in the ontology.

In addition we want show how we can enrich our folksonomy without human intervention with relevant data in order to help optimizing the time of search and enormously reduce the problems of spelling variations and the lack of semantics within folksonomies focusing on the rules-based systems.

Overall, we can summarize our motivations and objectives as follow (see table 1):

Table 1: A summary for the motivations and the objectives of the approach SSFO

• **Motivations:**
1. In the most of the solutions presented to overcome the problems of semantics and those related to the recommendation of resources in folksonomies; the preferences of each user are not taken into account. In other word these approaches aren't characterized by the specification of "display's instability" in folksonomies which is a corollary to the fact that the results of a search procedure vary depending on the interests of each user.
2. The majority of contributions don't take into account the resolution of tags' ambiguity problem.
3. The exactitude of results during the phase of information retrieval and even during the recommendation of resources isn't very precisely.
4. The works that aim to combine the techniques and the principals of the semantic web and the social web still suffer from these problems.
• **Objectives and suggestions:**
1. Recommending relevant resources for each user according to his preferences→ ***suggestion:*** study the profile of each user, compare it with the others and then recommend a set of pertinent resources to this one according to the results of similarities' calculations.
2. Reducing the problem of tags' ambiguity in folksonomies→ ***suggestion:*** see *Hypothesis 1* (Section 3.2).
3. Reducing the problem of tag' spelling variations in folksonomies→ ***suggestion:*** enrichment of folksonomie with a set of information[1] (data) can precise the search result and overcome the problem of spelling variation. (It should be noted that we are referred to the rules-based systems in order to realize this idea).
4. Decreasing the semantics' lacks in the social web → ***suggestion:*** using the principals of semantic web (ontologies) and also the rules language RIF to overcome this problem.

3.1 Formal description of the approach SSFO

In this section we will express how the community's members, the resources, and the tags have been represented, also how we have symbolized the relationships between these three elements.

Formally, a folksonomy is a tuple $\mathbf{F} = <\mathbf{U}, \mathbf{T}, \mathbf{R}, \mathbf{A}>$ where **U**, **T** and **R** represent respectively the set of users, tags and resources, and **A** represents the relationship between the three preceding elements i.e. $\mathbf{A} \subseteq \mathbf{U}\ \mathbf{x}\ \mathbf{T}\ \mathbf{x}\ \mathbf{R}$.

---

[1] We'll use in the remainder of this document the term "fact" instead of "information".

Because this approach is intended to present a technique that can help any folksonomy represented by an ontology to overcome the problems of tags' ambiguity and spelling variations based on the preferences and the interests of each user, and also enrich automatically the system by new relevant data, we suggest here to represent our folksonomy with a simple ontology defined by primitives relations such as "tagged by" and "used by"… etc. (For example: "a resource R is *tagged by* a tag T", "a resource R is *used by* a user U" …etc.).

3.2 The proposed method for treating the problem of tags' ambiguity

The method that we have chosen to treat the problem of tags' ambiguity is simple. It is based on an elementary hypothesis focused on the calculation of similarities between users in the system. Our technique to overcome the problem of tags' ambiguity is not based on the ontologies in an explicit manner, but it is rather based on the strength of the community effect which characterizes the technologies of web 2.0 in general. So the idea is to study the profile of each member in the system and then compare the preferences of this one with other users in order to extract those who are similar to him. The example 1 gives a simple explanation of this idea.

Example 1. In this example we have three users; each one is represented by the list of his tags as shows the Table 2:

Table 2: A matrix represents a set of users with their tags

Users	***Tags***
$\mathbf{U_1}$	computer, mac, web2.0, C#, perl.
$\mathbf{U_2}$	apple, fruit, food, chocolate.
$\mathbf{U_3}$	computer, apple, perl, mac, C++.

Let us suppose that user U_1 wants to retrieve resources relative to the word (i.e. the tag) *'apple'*. In the current folksonomies, the obtained result will contain all the resources tagged with the tag *'apple'* i.e. those relative to food and computer, even with the fact that it is clear for a human reading U_1's tags, that his preferences are relative to computer and not to food. The previous situation can be summarized as follow:

Problem 1. Lack of semantic links between tags leading to a problem of ambiguity: a tag can refer to several concepts, i.e., a tag can have several meanings.

Hypothesis 1. Two resources tagged by the same word (tag) are similar, if they are used by users who share similar interests or when they share some number of tags[2].

Solution 1. Measure the similarity between users, to specify those who have similar preferences.

[2] Of course the degree of similarity between two resources will differ according to the number of the common tags between them.

It should be noted that:

-To make the system flexible, we propose to make it interact with the user to accept or reject the retrieved resources.
-To avoid the "cold start" problem which is generally occur from a lack of the required data by the system in order to make an excellent recommendation; it's proposed to measure the similarity between resources when the users are not similar (see the Example 2).

Example 2: Let us now consider the following situation described in Table 3:

Table 3: The same set of users with another one (U_4); who have not a lot of information indicating his preferences

Users	***Tags***
$\mathbf{U_1}$	computer, mac, web2.0, C#, perl.
$\mathbf{U_2}$	apple, fruit, food, chocolate.
$\mathbf{U_3}$	computer, apple, perl, mac, C++.
$\mathbf{U_4}$	apple.

If the user U_1 searches resources tagged with 'apple'; our system will first propose him the resource corresponding to tag 'apple' which is used by the user U_3 with a 'very strong' level of recommendation because the two users U_1 and U_3 have similar preferences.

On the contrary the resources corresponding to the tag 'apple' which is used by the user U_2 will be given to U_1 with a percentage 'low' level of recommendation because U_1 and U_2 do not share the same interests.

Now how should the system answer U_4 for whom it does not have much information about his interests? For such cases, we propose to measure the similarity between the resources corresponding to the tag 'apple' which is used by U_4 and the resources already proposed to U_1 with a high percentage i.e. those of U_3,. If the resources are similar, the system will propose them to U_1 with a 'very strong' level of recommendation, otherwise with a 'low' level of recommendation.

So we can summarize our methodology as follow:

Step one: Calculation of similarities between users. To calculate this similarity we suggest to use a measure that allows representing each user by a vector v_i designates a series of binary numbers defined the set of his tags or his resources. Thus, to calculate the similarity between two users, for example U_1 and U_2, this measure proposes to calculate the cosines of the angle between their associated vectors v_1 and v_2 as shown in the formula (1):

$$cos(v_1, v_2) = \frac{v_1.v_2}{\|v_1\|^2 \|v_2\|^2} \quad (1)$$

Step two: Calculation of similarities between resources. When the users are not similar we suggest measuring the degree of similarity between resources in order to avoid "cold start" problem which is generally resulted from a lack of the data required by the system in order to make an excellent recommendation.

Step three: Calculate the percentage of recommendation for each proposed resource. We propose here assigning to each resource recommended by the system a factor that indicates the percentage of its recommendation, i.e. the resource is ‘highly recommended', 'recommended' or 'lowly recommended'. To achieve this classification, we propose to calculate the ratio between the number of resources used by the user himself (i.e. the one who does the search) and the number of the resources shared between him and the other users.

We must first select a threshold S ∈ [0, 1] to schedule the results. Such that for each resource recommended by the system we perform the following:

-Calculate $\frac{nb\ resources\ (User\ i \cap User\ j)}{nb\ resources\ (User\ i)}$ (2)

-If $\frac{nb\ resources\ (User\ i \cap User\ j)}{nb\ resources\ (User\ i)} \begin{cases} >= S \text{ then the resource is highly recommended} \\ < S \text{ then the resource is recommended} \\ = 0 \text{ then calculate the similarity between resources} \end{cases}$

In the Figure 1 which is presented below, an activity diagram that will illustrate the search’s procedure followed by our approach is introduced: As it’s shown in this Activity Diagram, the Levenshtein distance[3] is being used when the tag is not found in the folksonomy in order to measure the edit distance between tags in the system, this allows our system to detect spelling variations and so it can offer to user presumably equivalent tags.

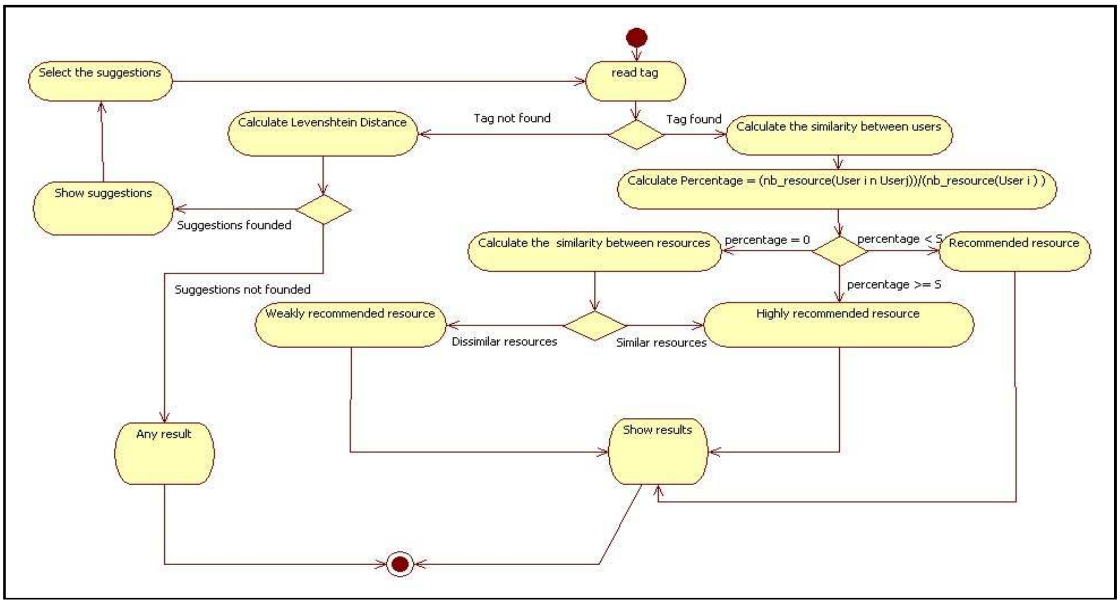

Fig 1: Activity Diagram for the search procedure

---

[3] By following the example of [11] and [5]

3.3 Rules-based systems in Folksonomies

The idea of using rules-based systems in folksonomies and in particular in this approach __which aims to extract the meaning of terms and overcome the problems of tags' ambiguity and spelling variations__ was born from the need to create and save new data from some information available in our folksonomy and others extracted from the similarities' calculations that are computed before (i.e. during the step 3.2).

The purpose here is threefold and it can be summarized in the following points:

a- Avoid the existence of an expert who must control and add the new data. This let us say that our technique is dynamic and automatic.

b- Optimize and reduce the time required for searching relevant resources for each user by avoiding the recalculation of similarities every time.

c- Enrich the folksonomy by a set of data that can help in the process of search and in reducing the problem of spelling variations by adding new facts deduced from the similarities' calculations (which have been made above).

In our approach the folksonomies' enrichment is realized by two categories of data as follows:

1. Enrich our fact base by facts extracted from the similarities' calculations that have already been made (during the step 3.2); and which say that: such resource is similar to such resource.

Example 3. If we have already found that a resource R1 is similar to another resource for example R2, then we add this information to our fact base, i.e. we add the following fact: is_similar_to (R1, R2) which express that "R1 is similar to R2".

With this method our system does not recalculate the similarity between the users every time when an actor want search a relevant resources, but it will optimize this time and also the memory space that can be lost in each calculation because with this process; before our system begin the calculation of similarity between users or between resources it will first see in the fact base if there are resources similar to those already proposed to this user __ this can be done by checking if there is a fact (is_similar_to (Rx, Ry)) in the fact base__. If it is the case then our system will not calculate the similarity between the current user (i.e. the one who do the search) and the others who have used this resource or even recalculate the similarity between these two resources (for example the two resources Rx and Ry) because they are already considered as similar, but it will propose this latter resource (we speak in this example about the resource Ry) directly to our user with the same percentage of recommendation of the resource that is similar to it. With this procedure we can guarantee the gain of time and memory space.

Note that the choice of these types of facts was based on resources and not on users because we must be aware that the profiles of users can be changed at any time by adding new tags or new resources and therefore we can't say that "two users have always the same tastes". On the contrary if a large set of users has already agreed that two resources are similar, this information becomes an assertion even if the profiles of these users can be changed in the future. And so we can put our second hypothesis which states that:

Hypothesis 2. Two resources are similar if they are already judged as similar by a group of users.

2. The second type of facts that we have chosen to enrich our folksonomy consist to add facts of the form: "A resource R_Z can have as tags the tag T_Y" or can_tagged_by (R_Z, T_Y). These facts can help us to overcome the problem of ambiguity and especially the problem of spelling variations in folksonomies. The example below (Example 4) gives a simple and a concise explanation.

Example 4. Supposing that we have found two resources R1 and R2 tagged with the same tag (we take for example the case of an ambiguous tag as "apple") as similar, and that R1 have as others tags the following set: computer, mac, software. On the contrary the resource R2 has only the tag apple. So if we add in our fact base the following assertions: can_tagged_by (R2, computer), can_tagged_by (R2, mac) and can_tagged_by (R2, software), the advantage of such facts is twofold:

i. Reduce the problem of tags' ambiguity (because the similarity between resources became more highly).
ii. Reduce the problem of spelling variations.

We can explain the second point (ii) as follows:
Problem 2. The problem of spelling variations when we want associate tags for the same or similar resources. For example a user can use two different tags to describe a same resource or similar resources.
Solution 2. Enhance automatically the fact base by assertions of the form: can_tagged_by (R_Z, T_Y) or "The resource Rz can have as tags the tag Ty".

Example 5. For example: "cat" and "chat" means both the same concept (animal) in English and in French, but when a user searches resources annotated by the tag "cat", the system will not offer him those tagged by the word "chat" because it can't understand that the tag "cat" is equivalent to the tag "chat". In others words, supposing that the user U_X tagged a resource R_1 by the tag *cat* and U_W is the user who tagged the resource R_2 by the tag *chat*. Noting that; the two resources R_1 and R_2 are already considered as similar according to the similarities' calculations that have been made before.

Now if the user U_X wants search resources concerning the animal "cat" by the tag *cat*, the resource R_2 will not be given to him. In order to overcome this problem our approach proposes to add the following facts: can_tagged_by (R_1, chat), can_tagged_by (R_2, cat). And now any user can benefit from the resources of the other and so we have overcame the problem of spelling variation in folksonomies.

Note 1. We must explain the difference between the two properties *tagged_by* and *can_tagged_by:*

The first (i.e. tagged_by (Rx, Ty)) shows that a user has tagged the resource Rx by the tag Ty. In other word it means that is the user who realizes this task and not the system. On the contrary the second relation (i.e. can_tagged_by (Rx, Ty) demonstrates that is the system who confirms that the resource Rx can tagged by the tag Ty after the calculation of similarities. So in our database we can differentiate between all the kinds of properties.

3.4 The relationship between our folksonomy and our rules-based system

Until now we have spoken only about the fact base and we haven't talk about the rules presented in our approach. In this part we will illustrate how our idea can help the approaches based on the representation of folksonomies with ontologies to overcome *automatically* the main problems in folksonomies like the problem of ambiguity and it of spelling variations.

Here the relationship between this solution and these problems appear behind the choice of the rules' language RIF (Rule Interchange Format), which became recently a W3C Recommendation. The choice of this language is motivated by the fact that it can support the import of RDF data and RDFS/OWL ontologies. Also a mapping to RIF from ontologies and the vice versa is possible, and thus we can easily treat our dataset and enrich the folksonomy.

Furthermore the strength of this language is appear from the fact that it can support many kinds of dialects; among them we find the RIF-PRD (the Production Rule Dialect of the W3C Rule Interchange Format) which allows adding, deleting and modifying facts in the fact base. And so we can easily produce updated rules for our fact base as required. In others terms we can modify or assert and also retract a set of facts in our data base according to our needs. In the example 6 we will show how we can assert or retract a set of facts in our database by the language RIF.

Example 6. Assume an initial state of the fact base that is represented by the following set, w0, of ground atomic formulas, where_u1, _t1, _r1, _u2, _t2 and _r2 denote individuals and where ex1:User, ex1:Tag and ex1:Resource represent classes:

Initial state:

• w0 = {_u1#ex1:User _t1#ex1:Tag _r1#ex1:Resource _u2#ex1:User _t2#ex1:Tag _r2#ex1:Resource
_u1[ex1:tag->_t1] _u1[ex1:resource->_r1] _t1[ex1:user->_u1] _t1[ex1:resource->_r1]
_r1[ex1:user->_u1] _r1[ex1:tag->_t1] _u2[ex1:tag->_t2] _u2[ex1:resource->_r2] _t2[ex1:user->_u2] _t2[ex1:resource->_r2] _r2[ex1:user->_u2] _r2[ex1:tag->_t2]}

1. **Assert (_r1[ex1:similarTo->_r2])** denotes an atomic action that adds to the fact base, a fact that is represented by the ground atomic formula: _r1[ex1:similarTo->_r2]. After the action is executed, the new state of the fact base is represented by:

• w1 = {_u1#ex1:User _t1#ex1:Tag _r1#ex1:Resource _u2#ex1:User _t2#ex1:Tag _r2#ex1:Resource
_u1[ex1:tag->_t1] _u1[ex1:resource->_r1] _t1[ex1:user->_u1] _t1[ex1:resource->_r1]
_r1[ex1:user->_u1] _r1[ex1:tag->_t1] _u2[ex1:tag->_t2] _u2[ex1:resource->_r2] _t2[ex1:user->_u2] _t2[ex1:resource->_r2] _r2[ex1:user->_u2] _r2[ex1:tag->_t2] _r1[ex1:similarTo->_r2]}

2. **Retract (_t1[ex1:user->_u1])** denotes an atomic action that removes from the fact base, the fact that is represented by the ground atomic formula _t1[ex1:user->_u1]. After the action, the new state of the fact base is represented by:

• w2 = {_u1#ex1:User _t1#ex1:Tag _r1#ex1:Resource _u2#ex1:User _t2#ex1:Tag _r2#ex1:Resource
_u1[ex1:tag->_t1] _u1[ex1:resource->_r1] _t1[ex1:resource->_r1] _r1[ex1:user->_u1] _r1[ex1:tag->_t1] _u2[ex1:tag->_t2] _u2[ex1:resource->_r2] _t2[ex1:user->_u2] _t2[ex1:resource->_r2] _r2[ex1:user->_u2] _r2[ex1:tag->_t2] _r1[ex1:similarTo->_r2]}

Note 2. We can use the term *Modify* in order to modify a fact represented in the fact base. And we can also use to realize this task the sequence: *Retract* then *Assert*.

4 Experimentation

After completing the design of the SSFO approach is time now to begin its implementation, so it's in this section where we can reap the fruits of our investigation. We will first describe the dataset used in the test phase and after we'll do the analysis and the discussion of the findings results.

4.1 Data Set

The dataset used in our test phase is described in this section followed by some analysis and discussion of the obtained results. The database of the website del.icio.us has been employed in this experiment representing the most used dataset for overall conducted experiments in folksonomies. Noting that we have selected a random set of data (i.e. a random sets of users, tags and resources) to well demonstrate the validity of our proposal. To fully shown the validity of SSFO, two classes of users are randomly chosen: the first one contains the users who employed ambiguous tags and tags with spelling variations and the second contains those who don't use these words but who can get them in the future. The users' number is equal to 55. For the set of tags used in this experiment we selected 526 different ones, among them there are some tags which are ambiguous and others have spelling variations. The used resources number is equal to 950; every resource can have multiple tags, and even multiple users. And in total we have 1605 tag assignments.

4.2 Data treatment

The first step. First of all, we have constituted a simple ontology from the dataset described in the section (4.1) in order to represent the folksonomy by ontology. It should be noted that we have used a simple properties for describing this ontology in order to avoid losing the meaning and the objective of our approach. So because the approach SSFO is intended to present a technique that can help any[4] folksonomy represented by an ontology to

[4] We try now to validate our approach SSFO with other ontologies proposed already in the above approaches in order to measure its effect to improve these latter.

overcome the problems of tags' ambiguity and spelling variations based on the preferences and the interests of each user, and also enrich automatically the system by new relevant data, we suggest here to represent our folksonomy with a simple ontology defined by primitives relations such as "tagged by" and "used by"…etc. (For example: "a resource R is *tagged by* a tag T", "a resource R is *used by* a user U" …etc.).

The second step. The aim of our contribution is to generate a flexible technique, in other words, a technique that can be adapted to any situation. For this reason, we tried to automate it by using tools that can considerably avoid the effort of the site's administrator.

Therefore after collecting all the data set used in the test phase, a tool of social network analysis called "Pajek[5]" is used, where the purpose is the extraction of the three networks 'Users_Tags', 'Users_Resources' and 'Tags_Resources'. The usage of this tool facilitates greatly the generation of the three graphs and their corresponding matrixes, which makes this step generated automatically. The figures presented below (Figure 2, 3 and 4) show the three networks generated from our data sample.

The results of this step are used in our methodology to calculate now the similarities between users and between resources in order to detect the pertinent resources for each searcher (i.e. each user).

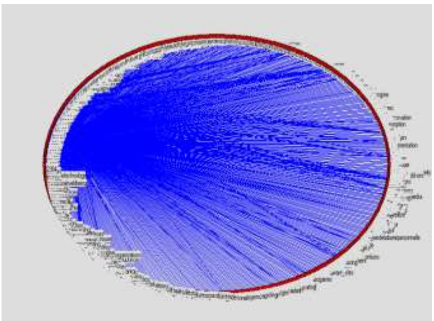

Fig 2: The network Users_Tags of our data sample with Pajek

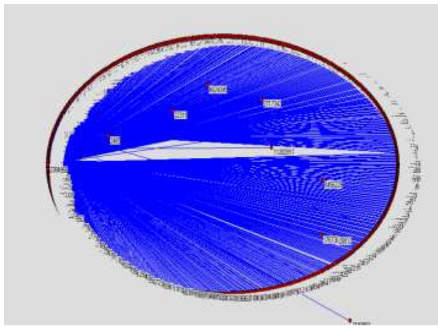

Fig 3: The network Users_Resources of our data sample with Pajek

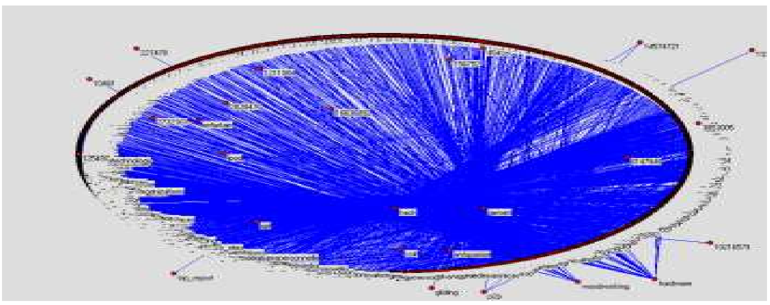

Fig 4: The network Tags_Resources of our data sample with Pajek

[5] It's an analytical tool of social networks, used in [7].

The third step. Once we have extracted the three social networks and calculate the different similarities, now we can turn to the third step in our experiment; it which related to the fact and the rules bases of our system. We have choose to represent the data available in these two bases the rules language RIF because it allow us representing and manipulating our data easily because it can manage RDF data and RDFS /OWL ontologies.

As we have already said in Section 3, the choice of the data used in the fact base is focused on the results obtained after the calculation of similarities, and so every time when a new user does a search by a tag, our system will add a rule specifies that a such resource is similar to a such resource using the RIF language. Concerning the rules base, we have used also RIF to produce rules allow us to add, delete or modify one or a set of facts as required. Note that the enrichment of folksonomies is made after a mapping from RIF to RDF, RDFS or OWL to obtain the new folksonomy represented by a new ontology (i.e. the last ontology modified after adding, modifying or retracting new set of facts).

4.3 Results

Seeing that a system of information retrieval allows giving to the user the documents that will help him to satisfy his need of information, and that a recommender system delivers documents to people basing on their profiles in the long term. Sure enough the system that have been realized combines these two fields, because it allows at the same time to propose resources that will help a user to satisfy his need for information, basing on its long-term profile. And for this reason a set of metrics that are employed in these two areas will be used for the evaluation of our approach:

Precision. It measures the system's ability to reject all not relevant resources to a query. It is given by the ratio of all relevant selected resources and the set of all selected resources.

Recall. It measures the ability of the system to retrieve all relevant resources to a query. It is given by the ratio of relevant retrieved resources and all relevant resources in the database.

Rates of precision and recall are given by the formulations (3) and (4):

$$\textbf{Precision} = \frac{R+}{M} \qquad (3) \qquad\qquad \textbf{Recall} = \frac{R+}{R} \qquad (4)$$

Where *R*: is the number of the relevant resources in the collection, *M*: is the number of the resources selected by the system and *R+*: is the number of the relevant resources selected by the system. It should be noted that the elements of each set (i.e. those of *R*, *M* and *R+*) are selected from our data sample as follow:

-The number of *R* for each user is calculated according to the profile of this one in the folksonomy. For example we take the case of the user who is identified by this list of tags {java, computer, mac}, in our evaluation we have supposed that the preferences of this latter are similar to a computer sciences field and not to the food when he did a search by the keyword apple. And so all the resources that are close to the first domain are considered relevant to this user and they are proposed to him with a highest degree of recommendation. And in order to avoid the case when this user changes his opinion and likes to search resources related to second field (i.e. food) our approach give him these resources at the end of the list in order to avoid this kind of problems. In other words the approach SSFO takes into account the probable changes in the tastes of each user.

- The number of *M* is calculated from the number of resources proposed by our approach.
- The number of *R+* is calculated by computing the number of the resources of R which are highly proposed to the user.

The metric F1. It's a combination of the two previous metrics and is defined by the formula (5):

$$F_1 = \frac{2 \,.\, Precision \,.\, Recall}{Precision + Recall} \quad (5)$$

In order to evaluate our approach; the three metrics listed above are calculated for each user, and then the average of each metric in the system is calculated. The results are shown in Figures 5 and 6.

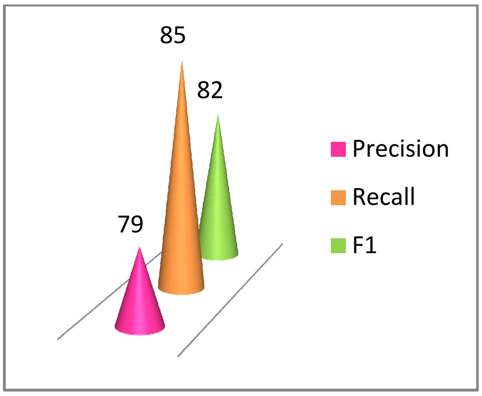

Fig 5: The average of the three metrics concerning the problem of tags' ambiguity

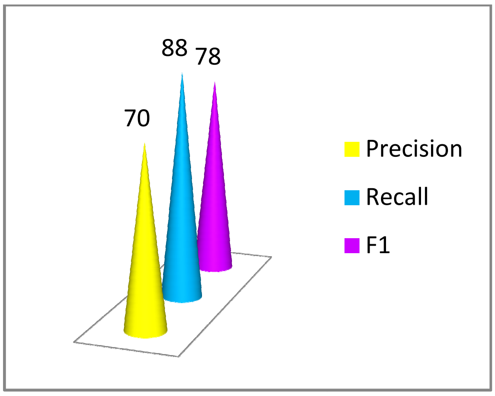

Fig 6: The average of the three metrics concerning the problem of spelling variations

4.4 Discussion

The approach presented in this work consists to find ways to equip in a "dynamic" manner the development of the systems of social sharing. It try to extract the semantics in order to allow users to capture in the course of their daily tasks the social dimension of their use of some terms. It appears from this study highlights that can be summarized as follows:

The consensus among users who have similar interests for using the same tags or the same resources plays an important role in the elimination of the problem of ambiguity.

Also the increase in the weights of these terms or these resources has an influence on the emergence of the semantic even when there are tags that can have several meanings.

Indeed the results which we have obtained concerning our data sample show that the technique SSFO is succeeded in distinguishing between ambiguous tags and also them which have spelling variations.

Comparing the SSFO approach with others trying to discuss the problem of tags' ambiguity; for example the Pan's and al work [8], we can conclude that our results are very optimistic especially when we know that the proposed approach is flexible i.e. the result of the search's procedure will be changed according to interests and the profile of each user in contrary to other approach. In addition the work presented in [8] doesn't tackle the problem of spelling variations.

Concerning the works that aimed to recommending a set of resources for each user like in the approach cited in [3]. We find that the technique that has been designed doesn't take into account the semantic between terms, in particular it can't distinguish between the ambiguous tags and also the spelling variations in tagging and therefore it can provide to a user resources that can rejected by h are not close to his preferences.

5 Conclusion and Future Work

Our investigations in the field of Web 2.0 and especially that of folksonomies have enabled us to make a substantial contribution in which we are interested with surmounting the problems of tags' ambiguity, spelling variations and also enriching our folksonomy automatically by a set of relevant facts.

We have proposed a new technique based on the force of social interactions between the different actors in the system and the rules-based systems in which the objective is creating a consensus among the users of a same system in order to increasing the semantics in folksonomies. We have tested this approach on a small amount of data and we have obtained good results, but this performance still requires a larger sample set to make it with confidence. In order to expand and improve this work we propose to validate our approach on a large amount of data, enrich our database by other relevant facts and rules and we wish also validated our approach on other databases.

6 References

1. Angeletou, S., Sabou, M., Specia, L., Motta, E.: Bridging the Gap Between Folksonomies and the Semantic Web: An Experience Report, Proc. Of ESWC workshop on Bridging the Gap between Semantic Web and Web (2007)

2. Buffa, M., Gandon, F., Ereteo, G., Sander, P. and Fabon, C.: SweetWiki: A semantic Wiki. J. Web Sem., 6(1), 84–97 (2008)

3. De Meo, P., Quattrone, G., Ursino, D.: A query expansion and user profile enrichment approach to improve the performance of recommender systems operating on a folksonomy. User Model. User-Adapt. Interact. 20(1): 41-86 (2010)

4. Gruber, T.: Tag Ontology- a way to agree on the semantics of tagging data, http://tomgruber.org/ (2005)

5. Limpens, F., Gandon, F., Buffa, M. : Sémantique des folksonomies: structuration collaborative et assistée, IC (2009)

6. Markines, B., Cattuto, C,. Menczer, F., Benz, D., Hotho, A., Stumme, G.: Evaluating similarity measures for emergent semantics of social tagging. WWW 2009: 641-650. (2009)

7. Mika, P.: Ontologies are us: A unified model of social networks and semantics, In ISWC, volume 3729 of LNCS, p. 522–536: Springer (2005)

8. Pan, J., Taylor, S., Thomas, E.: Reducing Ambiguity in Tagging Systems with Folksonomy Search Expansion, In Proc. of the 17th International World Wide Web Conference (2009)

9. Specia, L. Motta, E.: Integrating Folksonomies with the Semantic Web, 4th European Semantic Web Conference (2007)
10. Scott A. Golder, Bernardo A. Huberman: Usage patterns of collaborative tagging systems. J.Information Science 32(2): 198-208 (2006)
11. Specia, L. Motta, E.: Integrating Folksonomies with the Semantic Web, 4th European Semantic Web Conference (2007)
12. http://en.wikipedia.org/wiki/Folksonomy [Retrieved 30 June 2011].