



HAL
open science

A Complete Life-Cycle for the Semantic Enrichment of Folksonomies

Freddy Limpens, Fabien Gandon, Michel Buffa

► **To cite this version:**

Freddy Limpens, Fabien Gandon, Michel Buffa. A Complete Life-Cycle for the Semantic Enrichment of Folksonomies. Springer. Advances In Knowledge Discovery and Management, Springer, 2013, 978-3-642-35855-5. <10.1007/978-3-642-35855-5>. <hal-01171250>

HAL Id: hal-01171250

<https://inria.hal.science/hal-01171250v1>

Submitted on 3 Jul 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

A Complete Life-Cycle for the Semantic Enrichment of Folksonomies

Freddy Limpens, Fabien Gandon, and Michel Buffa

Abstract. Tags freely provided by users of social tagging services are not explicitly semantically linked, and this significantly hinders the possibilities for browsing and exploring these data. On the other hand, folksonomies provide great opportunities to bootstrap the construction of thesauri. We propose an approach to semantic enrichment of folksonomies that integrates both automatic processing and user input, while formally supporting multiple points of view. We take into account the social structure of our target communities to integrate the folksonomy enrichment process into everyday tasks. Our system allows individual users to navigate more efficiently within folksonomies, and also to maintain their own structure of tags while benefiting from others contributions. Our approach brings also solutions to the bottleneck problem of knowledge acquisition by helping communities to build thesauri by integrating the manifold contributions of all their members, thus providing for a truly socio-semantic solution to folksonomy enrichment and thesauri construction.

1 Introduction

Social tagging is a successful means to involve users in the life cycle of the content they exchange, read or publish online. However, folksonomies resulting from this practice have shown limitations, in particular, the spelling variations of similar tags and the lack of semantic relationships between tags that significantly hinder the navigation within tagged corpora.

Freddy Limpens · Fabien Gandon
Edelweiss - INRIA Sophia Antipolis, France
e-mail: fdy@pl-area.net, fabien.gandon@inria.fr

Michel Buffa
I3S - CNRS / University of Nice - Sophia Antipolis, France
e-mail: buffa@unice.fr

One way of tackling these limitations is to semantically structure folksonomies. This can help navigate within tagged corpora by (1) enriching tag-based search results with spelling variants and hyponyms, or (2) suggesting related tags to extend the search, or (3) semantically organizing tags to guide novice users in a given domain more efficiently than with flat lists of tags or occurrence-based tag clouds, or (4) assisting disambiguation.

We present our approach to design a tagging-based system that integrates collaborative and assisted semantic enrichment of the community's folksonomy. We propose formal models and methods to support diverging points of view regarding the semantics of tags and to efficiently combine them into a coherent and semantically structured folksonomy.

Our end-user is the Ademe agency¹ which seeks to broaden the audience of its scientific work in the field of sustainable development and environmental issues. In this scenario, we can distinguish three types of stakeholders: (1) the expert engineers working at Ademe who are specialists of a given domain, (2) the archivists who take care of the indexing of the documents from Ademe and have transversal knowledge of the thematic covered at the agency, and (3) the public audience who has access to the documents of Ademe from its website. The archivists seek to both enrich their indexing base, which can be seen as a controlled folksonomy, and to upgrade it towards a thesaurus-like structure. The difficulty here comes from the different points of view that may arise from the community of expert engineers, and possibly also from the public, and that have to be turned into a coherent structure by the archivists.

In section two we present current works in folksonomy semantic enrichment, and position our contribution. In section three we give a general presentation of our approach. In section four we present the results of automatic processing of tag data, and detail our method to extract emergent semantics with a combination of string edit distances. Section five will cover the capture and exploitation of users contribution to provide a semantically enriched folksonomy that supports multiple points of view. Section six will conclude and give some insights about possible future developments.

2 Related Work

Folksonomy enrichment has been addressed by numerous research works covering a broad variety of approaches.

2.1 *Extracting the Emergent Semantics*

A first category of work aims at extracting the emergent tag semantics from folksonomies by measuring the semantic similarity of tags. The studies from

¹ French Environment and Energy Management Agency, <http://www.ademe.fr>

[Markines et al., 2009] and [Cattuto et al., 2008] propose an analysis of the different types of similarity measures and the semantic relations they each tend to convey. The simplest approach consists in counting the co-occurrence of tags in different contexts (users or resources). Cattuto *et al.* [Cattuto et al., 2008] showed that this type of measure provided subsumption relations but was not sufficiently accurate. More elaborate methods exploit the network structure of folksonomies making use of the distributional hypothesis that states that words used in similar contexts tend to be semantically related. To apply this hypothesis on tags, [Cattuto et al., 2008] computed the cosine similarity measure in the vector spaces obtained by folding the tripartite structure of folksonomy onto distributional aggregations spanning the associations of tags with either: the other tags (tag-tag context), or the users (tag-user context), or the resources (tag-resources). Their study shows that the tag-tag context performed best at a reasonable cost and that the semantic relation conveyed by this measure was of type “related”. Mika [Mika, 2005] also applied and evaluated different folding of the tripartite structure of folksonomies. Interestingly, he showed according to a qualitative evaluation that exploiting user-based associations of tags yielded more representative taxonomic relations. The principle of this association is that if, *e.g.* the community of users using the tag “biological agriculture” is included in the community of users of the tag “agriculture”, then the tag “agriculture” is broader than the tag “biological agriculture”. Heyman *et al.* [Heymann and Garcia-Molina, 2006] proposed an algorithm that constructs a taxonomy from tags by crawling the similarity graph computed from the cosine distance based on the Tag-Resource context. The hierarchy of tags is built starting from the tag with the highest centrality, and each tag, taken in order of centrality, is added either as a child of one of the nodes or of the root node depending on a threshold value.

2.2 Models and Tools to Structure Tags

Another type of approach consists in letting users semantically structure tags. [Tanasescu and Streibel, 2007] proposed to tag the tags, [Huynh-Kim Bang et al., 2008] proposed a simple syntax to specify subsumption (with “>” or “<”) or synonymy (with “=”) relations between tags. Some tools available online also feature semantic structuring capacities such as Gnizr² and Semanlink³, and even Flickr with machine tags⁴. In the same trend, the Linked Data community seeks to weave together the content of social web sites thanks to a set of formal ontologies not aimed at describing the knowledge of the communities but rather the structure of their knowledge exchange platforms. For instance SCOT⁵ describes tags as parts of sharable tag clouds, and SIOC⁶ describes online

² <http://code.google.com/p/gnizr/>

³ <http://www.semanlink.net>

⁴ <http://www.flickr.com/groups/mtags/>

⁵ <http://scot-project.org/>

⁶ <http://sioc-project.org/>

communities' content. MOAT[Passant and Laublet, 2008] is an ontology aimed at linking each tagging action with a URI representing the meaning of this tag action. These URIs can link to formal ontologies concepts or any web page containing a description of a notion. Once tag actions are formally linked to concepts, it is possible to disambiguate tags when searching, but also to exploit inference mechanisms via the formal concepts and to get a richer browsing experience. NiceTag⁷ is a model that seeks to account for the usages of tags through a finer modeling of the relations between tags and the tagged resources [Limpens et al., 2009]. Its flexibility and the use of a named graphs mechanism allows this model to serve as a pivot model for all other tag models, adding a level of pragmatics. Finally, as we propose to support diverging points of view, let us recall briefly some multi-points of view approaches such as [Rivière, 1999] who proposed multi-points of view knowledge representations grounded on the conceptual graphs formalisms in which the links between concepts can be bound to a given point of view. [Bouquet et al., 2004] does not exactly propose representing concepts according to multiple points of view, but instead suggest contextualizing ontologies thanks to C-OWL, an extension of OWL. The idea of C-OWL is to provide a set of primitives to describe mappings between a series of « local » ontologies that can be each associated to a point of view.

Some other works seek to integrate one or several of the preceding approaches. For instance [Angeletou et al., 2008] and [Specia and Motta, 2007] make use of similarity metrics to find related tags, and then map these tags to concepts from available online ontologies in order to semantically structure tags with formal properties. [Van Damme et al., 2007] proposed an integrated approach to folksonomy enrichment including as many resources as possible, using each in a tailored way in addition to the validation of the inferences by the users.

Finally, our approach can be related to ontology construction and ontology maturing. Indeed, our approach clearly echoes attempts to build formal ontologies from texts [Aussenac-Gilles et al., 2000] or databases maintained by communities of users [Golebiowska, 2002]. More recently, Braun *et al.* [Braun et al., 2007] addressed the problem of collaborative ontology editing and pointed out the limitations of current ontology engineering tools in that respect. They proposed integrating ontology maturing in common tasks such as information seeking, and they developed a bookmarking service with the possibility for all users to add or edit new “semantic” tags formally structured with SKOS⁸.

2.3 Discussion of Current Approaches

Full automatization of semantically enriching folksonomies is difficult. First the similarity measures used by [Cattuto et al., 2008, Markines et al., 2009], [Specia and Motta, 2007] or other methods for retrieving taxonomical structures

⁷ <http://ns.inria.fr/nicetag/2009/09/25/voc>

⁸ <http://www.w3.org/2004/02/skos/core#>

from folksonomies [Mika, 2005, Heymann and Garcia-Molina, 2006] are useful to bootstrap the process, but their accuracy in reflecting the communities knowledge is limited. The semantic grounding of these measures proposed by [Cattuto et al., 2008] can also help evaluate their accuracy. However, as this evaluation requires that tags be present in Wordnet synsets or in other ontological resources, the validity of these measures can only be evaluated for common knowledge and not really for specific terms that consist in one of the most valuable benefits of folksonomies. The same argument can be used towards other approaches [Angeletou et al., 2008] that make use of ontological resources to formally structure folksonomies.

On the other hand, approaches that rely on user input (to tag the tags, or to link a tag to an unambiguous concept) may induce, without user-friendly interfaces tailored to usages, a cognitive overload that regular users of tagging are not ready to bear. Integrated approaches try to overcome this limit by mixing automatic handling with user validation. However, none of these two types of approaches formally takes into account the multiplicity of points of view within a community, a feature at the core of our approach for which we will now give an overview.

3 Semantic Enrichment of Folksonomies

A generic method to semantically enrich all types of folksonomies in a fully automatic manner seems out of reach today. Our approach to semantically enriching folksonomies consists in creating a synergistic combination of automatic handling, to bootstrap the process, and of users' contributions at the lowest possible cost through user-friendly interfaces. We propose a system that supports conflicting points of view regarding the semantic organization of tags, but also helps online communities build a consensual point of view emerging from individual contributions.

3.1 *SRTag: Using Named Graphs to Keep Track of Diverging Points of View*

In order to model the semantic structuring of folksonomies while supporting conflicting views, we propose an RDF schema, SRTag⁹, which makes use of named graphs mechanisms [Carroll et al., 2005, Gandon et al., 2007]. Named graphs allow to reify the semantic relationship between two tags or two concepts (modeled with SKOS) without the burden of classical RDF reification¹⁰ (see figure 1). The benefits and the reasons for using named graphs to capture assertional intents are given in details in [Limpens et al., 2009], but we can merely recall here that we required a mechanism that allow to encapsulate statements about tags and give a URI to these

⁹ <http://ns.inria.fr/srtag/2009/01/09/srtag.html>

¹⁰ <http://www.w3.org/TR/rdf-mt/#Reif>

statement in order to be able to link them to other entities. For example, we wanted to be able to say that “Kevin agrees with the fact that `soil pollution` is a more specific term than `pollution` but Alex disagrees”. Using a named graph that encapsulate “`soil pollution` is a more specific term than `pollution`” allows us to reuse it with as many other agreement or disagreement relations (or any other type of relation if needed). In addition, these named graphs are typed with our class `srtag:TagSemanticStatement` or with more precise subclasses.

The relationships between tags can be taken from any model, but we chose to limit the number of possible relations to thesaurus-like relations as modeled in SKOS. Then we modeled a limited series of semantic actions which can be performed by users (represented using `sioc:User` class), namely `srtag:hasApproved`, `srtag:hasProposed`, and `srtag:hasRejected`. We are then able to capture and track back users opinions (reject or approve) on the asserted relations, and thus to collect diverging points of view.

We distinguish different types of automatic and human agents according to their role in the life cycle of the folksonomy. We modeled different subclasses of the class `sioc:User` in order to filter statements according to the users who approve it. This includes `srtag:SingleUser` which corresponds to regular users of the system, `srtag:ReferentUser` (e.g. an archivist) who is in charge of building a consensual point of view, `srtag:TagStructureComputer` which corresponds to the software agents performing automatic handling of tags, and `srtag:ConflictSolver` corresponding to software agents which propose temporary conflict resolutions for diverging points of view before referent users choose one consensual point of view.

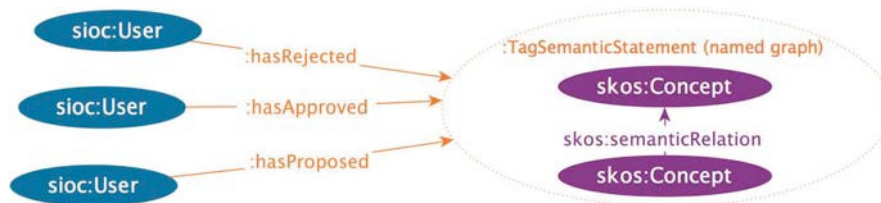


Fig. 1 SRTag RDF schema

3.2 Folksonomy Enrichment Life Cycle

As a result, our model allows for the factorization of individual contributions as well as the maintenance of a coherent view for each user and a consensual view linked to a referent user. Furthermore, by modeling different types of agents who propose, approve or reject tag relations, we are able to set up a complete life cycle of enriched folksonomies. Figure 2 illustrates this life cycle which starts with a “flat” folksonomy (ie. with no semantic relationships between tag) and can be decomposed as follows:

1. Automatic processing is performed on tags using methods based on an analysis of the labels of tags and on the network structure of the folksonomy. `srtag:TagStructureComputer` agents then add assertions to the triple store stating semantic relations between tags. These computations are done overnight due to their algorithmic complexity.
2. Human agents, modeled as `srtag:SingleUser`, contribute through user friendly interfaces integrated into tools they use daily by suggesting, correcting or validating tag relations. Each user maintains his point of view, while benefiting from the points of view of other users.
3. As logical inconsistencies may arise between all users' points of view, another type of automatic agent (`srtag:ConflictSolver`) detects these conflicts and proposes resolutions. The statements proposed are used to reduce the noise that may hinder the use of our system when, for instance, different relations are stated about the same pair of tags.
4. The statements from the conflict solver agent are also used to help the referent user in her task of maintaining a global and consensual view with no conflicts. This view can then be used to filter the suggestions of related tags by giving priority to referent-validated tags over other tags suggested by computers.
5. At this point of the life cycle we have a semantically structured folksonomy in which each user's point of view co-exists with the consensual point of view. Then a set of rules is applied to exploit these points of view in order to offer a coherent navigation to all users.
6. Another cycle restarts with automatic handlings to take into account new tags added to the folksonomy.

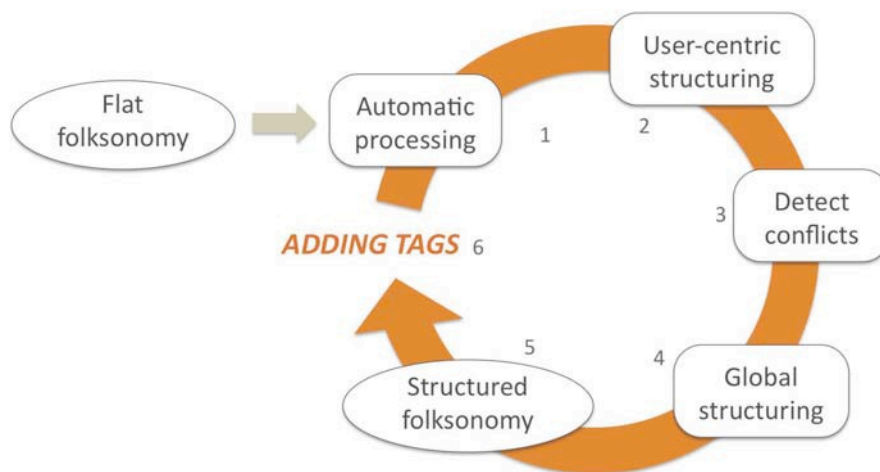


Fig. 2 Folksonomy enrichment lifecycle

4 Automatically Extracting Emergent Semantics

Several types of methods can be applied to folksonomies in order to retrieve semantic relationships between tags. We first present the experiment we conducted with real data from the Ademe agency to evaluate the performance of string-based methods and our proposal to combine them efficiently. Then we present our integration of state of the art algorithms [Markines et al., 2009, Mika, 2005] analyzing the structure of folksonomies.

4.1 Evaluating the Performance of String-Based Metrics

4.1.1 Overview of Existing String-Based Metrics

String based distance measures consider the character strings of the labels of tags to be compared. For instance, the Levenshtein [Levenshtein, 1966] distance metric was used in [Specia and Motta, 2007] to group spelling variant tags such as “new_york” and “newyork”. To go further in the use of these cost effective methods, we conducted a benchmark to evaluate the ability of such metrics to retrieve other types of semantic relations such as *related* relation, or *narrower* or *broader* relation, also called *hyponym* relation. Hyponym relations reflect the relative degree of generality between two notions such as, *e.g.* in: “pollution” is broader than “soil pollution”. Two notions are merely related in the other cases, as for instance “energy” and “electricity”.

We have compared the similarity metrics implemented in the package SimMetrics¹¹ which give, for a pair of strings (s_1, s_2) , a normalized value between 0 and 1, with a value of 1 meaning that both compared strings are most similar. The similarity metrics we compared fall into several categories: (a) edit distance based methods, which consider the set of operations needed to turn string s_1 into string s_2 ; (b) token-based methods, such as overlap coefficient, which decompose strings into tokens separated by white space; (c) methods using vector representations of strings such as the cosine similarity; and finally (d) other types of metrics such as QGram or Soundex metrics.

4.1.2 Benchmarking

We have manually constructed a test sample from the tags used at Ademe to index their documents and resources. This sample, which mixes freely chosen tags and tags chosen by the archivists, was divided into 4 sets of 22 pairs of tags (t_1, t_2) , each set containing tag pairs which correspond to a semantic relation, namely: spelling variant, hyponym, related, and unrelated. These relations have been validated by one member of the Ademe’s archivists team so that it reflects the knowledge of our user’s domain.

¹¹ <http://www.dcs.shef.ac.uk/~sam/stringmetrics.html>

The Monge-Elkan metric is a hybrid metric based on edit distances which also decomposes strings into tokens, and uses a second metric to compare each token with all the others. For our experiment we used a series of 15 metrics and the combination of these 15 metrics with the Monge-Elkan method, which makes a total of 30 different metrics.

Our benchmarking approach consists in using a sample of pairs of tags (mostly in French), manually constructed and validated by a human expert (from Ademe in our case), and which will serve as a reference. This sample was divided into 4 subsets, each subset containing 22 pairs of tags linked with one type of semantic relation, namely: spelling variant, hyponym, related, and unrelated. To evaluate the performance of each metric in retrieving the right relations from our sample, we have computed for each subset the recall, precision and the weighted harmonic mean F_1 (to give as much importance to recall as to precision). These values were computed varying the threshold above which a given tag pair is retrieved or not. Then to count the false positive and true positive pairs that were matched, we applied the following rules: (a) for the related case the true positives are counted from all subsets except the unrelated subset, since spelling variant and broader/narrower pairs can be considered also “related”; (b) for the spelling variant and hyponym case, the true positives were only those from their corresponding subset, and the pairs from all the other subsets were counted as false positives.

We have then looked at the best metric for each type of relation by ranking them according to the mean value and the statistical deviation of F_1 . The outcome of this first evaluation is that the Monge-Elkan_Soundex method outperformed other metrics in the related case. The best in the spelling variant case is the Jaro-Winkler metric, and the best for the hyponym is the MongeElkan_NeedlemanWunch metric. In the latter case however, none of the top metrics clearly outperformed the others. We should also notice the greater deviation in the related case than in the two other cases, and this result was expected since the fact for two notions being related rarely translates to some terminological similarities e.g. "car" and "wheel" are related but don't share any letters. Now we are interested in finding a way, using these metrics, to differentiate between the 3 types of semantic relations.

4.1.3 Identifying Different Types of Relations

Now we are interested in finding a way, using these metrics, to differentiate between the 3 types of semantic relations.

First, we use the MongeElkan_Soundex metric to retrieve all related tag pairs, that is, all pairs sharing a relation which is at least of type “related”, meaning that in this category we'll retrieve also spelling variant and hyponym cases. To do that, we must determine a threshold of the similarity value from the MongeElkan_Soundex metric above which a pair is considered related. To determine this threshold, we looked at the mean similarity value for all related cases (spelling variant, hyponym, related) and for all unrelated cases in the sample set. The results are shown in

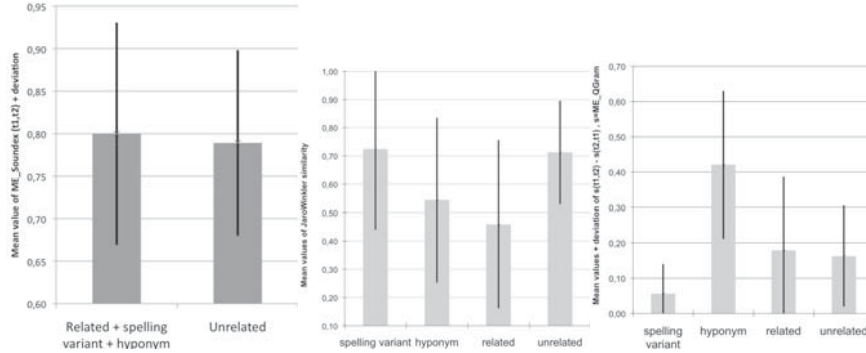


Fig. 3 (left) Comparison of the mean value of the MongeElkan_Soundex metric for all related cases (spelling variants, hyponyms and mere related) and for unrelated cases. (middle) Comparison of the mean value of the JaroWinkler metric for each type of semantic relation. (right) Mean value of the difference $\bar{\delta} = s(t_1, t_2) - s(t_2, t_1)$ with s being the Monge-Elkan_QGram metric for each set of tag pairs.

fig. 3(left). We can see that, considering the deviations, if we choose a threshold value of 0.9 we are able to avoid unrelated pairs.

To distinguish spelling variant from related pairs, we look at the mean value and deviation of the best metric in the spelling variant case. In figure 3 (middle) we show the mean value of the JaroWinkler metric for the four types of semantic relations. We see that, taking into account the deviation, if we choose a threshold above 0.9 we are more likely to retrieve spelling variant pairs. This result is confirmed when we look at the threshold value for which F_1 is maximum for the JaroWinkler measure in the spelling variant case.

Next, we want to find a way to tell hyponym pairs from related pairs. The Monge-Elkan metrics are not symmetric, and we have calculated, for each tag pair (t_1, t_2) , the difference $\bar{\delta} = s(t_1, t_2) - s(t_2, t_1)$, with s being one of the 15 combination of MongeElkan with another metric. In figure 3(right) we give the mean value and deviation of $\bar{\delta}$ for each set of tag pairs according to the MongeElkan_QGram metric, which performed best in this respect. We only included in this computation tag pairs that were retrieved thanks to the MongeElkan_Soundex metric and counted “related”. We can see that if we choose a threshold above 0.39 (the highest value for $\bar{\delta}$ when including the deviation), we are able to retrieve tags sharing a hyponym relation. When taking into account the sign of the difference, we are able to tell the direction of this relation, meaning that if we have $\bar{\delta}$ negative and above a certain threshold, then t_1 can be considered narrower than t_2 .

4.1.4 Heuristic String Based Methods (Algorithm 1)

As a result we are able to propose a heuristic (see algorithm14) that combines the best metrics to retrieve different semantic relations between tags. We first look for

pairs of related tags (t_1, t_2) using Monge-Elkan_Soundex with a first threshold τ_a so that we have $s(t_1, t_2) \geq \tau_a$. This first threshold is chosen as explained in 4.1.3, ie $\tau_a = 0.9$ in our case. Then, we compare the JaroWinkler similarity with a second threshold τ_b to see whether the tags are spelling variants, such that $s(t_1, t_2) \geq \tau_b$. The threshold in this case is chosen as explained in 4.1.3, *i.e.* in our case, 0.94. If it's not the case, we use a third threshold τ_c and we compute the difference δ of the MongeElkan_QGram metric $\delta = s(t_1, t_2) - s(t_2, t_1)$, and if δ is such that $\delta \leq -\tau_c$, then we can infer that t_1 is narrower than t_2 , or if $\delta \geq \tau_c$ then t_1 is considered broader than t_2 . The third threshold is chosen after the results shown in figure 3 by picking a value above 0.39. In this process we give priority to the detection of spelling variants since string based methods are better suited for this type of relation, and by checking this case first we make sure to retrieve as many spelling variant cases as possible since those retrieved have statistically more chance to be true positive.

Algorithm 14. Heuristic string based metric to retrieve semantic relations between tags

```

for all distinct pair of tags  $(t_i, t_j)$  from  $S = \{t_1, t_2, \dots, t_n\}$  do
  if  $MongeElkanSoundex(t_i, t_j) > \tau_a$  then
    if  $JaroWinkler(t_i, t_j) > \tau_b$  then
       $t_i$  has spelling variant  $t_j$ 
    else if  $MongeElkanQGram(t_i, t_j) - MongeElkanQGram(t_j, t_i) \leq -\tau_c$  then
       $t_i$  has broader  $t_j$ 
    else if  $MongeElkanQGram(t_i, t_j) - MongeElkanQGram(t_j, t_i) \geq \tau_c$  then
       $t_j$  has broader  $t_i$ 
    else
       $t_i$  has related  $t_j$ 
    end if
  end if
end for

```

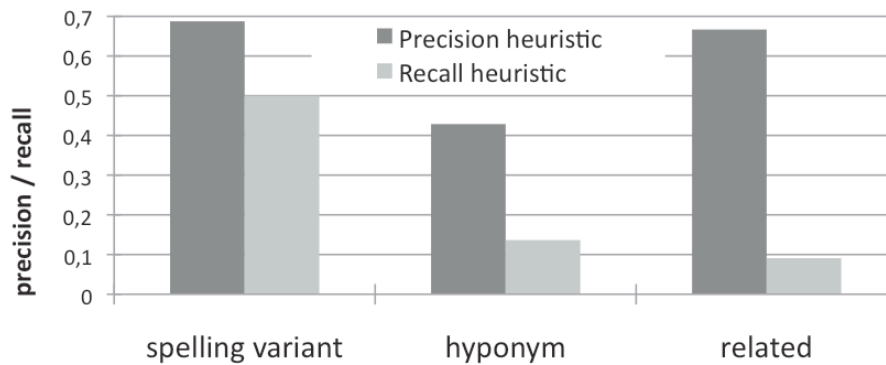


Fig. 4 Performance of the heuristic string-based metric (Algorithm 1)

We have applied our heuristic method to the same sample test. However, this heuristic is not directly comparable to the other metrics as it combines different methods and retrieves 3 types of semantic relations at a time, while in the global comparison experiment each metric was dealing with one type of semantic relation at a time. However, in order to evaluate quantitatively the global performance of this heuristic string-based metric, we show in figure 4 the values of the precision and recall for the 3 types of relations. We can clearly see in this figure that string based metrics perform best in the spelling variant case, which confirms a natural intuition since string-based methods were originally designed to match similar strings. Nonetheless, the noticeable performance in the hyponym case is explained with the ability of string-based metrics to easily detect common tokens such as in “pollution” and “soil pollution” and these cases often correspond to a hyponym relation. The related case is more difficult (hence the low precision) as this relation is the fuzziest and probably the least noticeable in the actual spelling of the tags (“sun” and “energy” *e.g.*). Finally, this indicates the need to use other methods to be able to cover other cases where semantically related tags are not morphologically similar.

4.2 Analyzing the Structure of Folksonomies

In this section we detail our implementation of two methods (named Algorithm 2 and Algorithm 3 in the remaining) to extract emergent semantics that analyze the tri-partite structure of the folksonomy.

Algorithm 2

In order to extract *related* relationships between tags, we use the similarity measure based on distributional aggregation in the tag-tag context [Cattuto et al., 2008]. Cattuto *et al.* compared the different context in which similarity measures can be computed and studied the different type of semantic relationships they bring using the hierarchical structure of Wordnet. This experiment shows that tags associated via similarity measures based on simple co-occurrence tend to share subsumption relationships, whereas tags associated via distributional similarity measures in the tag-tag context tend to be on the same level of a semantic hierarchy, either having the same parents and grand-parents. Cattuto et al. explain that associating tags via their co-occurrence on a single resource accounts for their simultaneous use in the same act of tagging, where the user may have a tendency to span different levels of generality. For instance the tags “java” and “programming” are likely to be used simultaneously, and we can assume that they have, in the user’s mind, different levels of generality. The relationship measured by the distributional measure based on the tag-tag context associates tags which share similar patterns of co-occurrence, but which are not necessarily or rarely used together. This is the case for example of the tags “java” and “python” which may be rarely used together, but each may be often used with the tag “programming”.

To compute the tag-tag context similarity, we first consider the vector representation v_i of each tag t_i in this context. Each entry of this vector v_i is given by $v_{it_j} = w(t_i, t_j)$ for $t_i \neq t_j$ where $w(t_i, t_j)$ corresponds to the co-occurrence value for the tags (t_i, t_j) , and $v_{it_i} = 0$. We set to zero the value for a tag with itself so that we consider tags to be related when they are found in a similar context, but not when co-occurring together. The similarity value for a pair of tag (t_i, t_j) in the tag-tag context is then given by the cosine distance between the vectors v_i and v_j : $\cos(v_i, v_j) = \frac{v_i \cdot v_j}{\|v_i\| \cdot \|v_j\|}$. When this value is above a given threshold, we create an annotation saying that tag t_i is related to tag t_j .

Algorithm 3

In order to extract subsumption relations, we made use of the method described by [Mika, 2005] which consists in looking at the inclusions of the sets of users associated to a tag. Let S_i be the set of users using tag t_i , and S_j be the set of users using tag t_j . If the set S_i is included in the set S_j , so that we have $S_i \subset S_j$, with $\text{card}(S_i) > 1$ and $\text{card}(S_j) > \text{card}(S_i)$, we can infer that the tag t_j is broader than the tag t_i .

Note that these two algorithms are not incremental since we have to analyze the whole folksonomy to compute the similarity of newly added tags.

4.3 Automatic Processing on a Real-World Dataset

We have performed the three types of calculation described above on a real-world dataset made of the following parts: (a) *delicious* dataset¹² comes from delicious.com and is made of the tagging of users who tagged at least one of their bookmarks with the tag “ademe” as of the 1st of October, 2009. (b) *thesenet* dataset comes from a database of Ademe which lists all the PhD projects funded by the agency. Each keyword has been considered as a tag, each identified project as a tagged resource, and each PhD student as the tagger. (c) *caddic* dataset is made of all entries of the past five year of the documents indexing base of the Ademe’s archivists. Each document corresponds to a tagged resource, and each keyword from the list of keywords associated to each document corresponds to a tag, with the archive service as the only tagger since no trace of the person who validated each entry is kept. In table 1 we detail, for each dataset: the number of distinct tags; the number of restricted tagging, *i.e.* the number of tripartite links between one resource, one tag and one user; the number of posts, *i.e.* the number of set of tags assigned by one user to a single resource (as a bookmark in delicious.com); the number of distinct tagged resources; and the number of users.

In table 2 we give some details on the results we obtained for each of the three methods of computation (Algorithm 1, 2, and 3) when applied to the three datasets. The first thing to notice is that algorithm 1 yields far more results (71034 statements)

¹² This subset of our experimental data is available, as of the time of writing, on the Isicil website <http://isicil.inria.fr>

Table 1 Description of the dataset

	delicious	thesenet	caddic	Full Dataset
Nb. distinct Tags	1015	6583	1439	9037
Nb. Restricted Tagging (1R - 1T - 1U)	3015	10160	25515	38690
Nb. distinct Resources	196	1425	4765	6386
Nb. distinct Users	812	1425	1	2238

Table 2 Description of the results of automatic processing

	Algo. 1	Algo. 2			Algo. 3		Total
	Full dataset	delicious	thesenet	caddic	delicious	thesenet	
Nb. related	59889	8141	206	30	-	-	68633
Nb. Broader/Narrower	10952	-	-	-	106	196	11254
Nb. Spelling variants	3193	-	-	-	-	-	3193
Computation time (s)	20952	4200	180	300	5	10	25647
Total number of statements							83080
Nb. of pairs with overlapping statements between different methods							31
Nb. of pairs with conflicting statements between different methods							22
Total number of statements on distinct pairs							83027

than algo. 2 (8377 statements in total, with 97% from delicious dataset). The results for algorithm2 can be explained because this method looks at the pattern of co-occurrence of tags, and delicious is the dataset in which two tags are more likely to have similar patterns of co-occurrence since, if we look at the ratio between the number of restricted tagging over the number of distinct resources, we obtain 15.38 for delicious, 7.13 for thesenet, and 5.35 for caddic. This says that there are more than twice as much distinct users who tagged the same resource in delicious as in thesenet or caddic. In addition, in delicious, a greater number of users tag the same resource using a smaller set of distinct tags, hence the greater probability for two tags to have similar patterns of co occurrence. For algorithm 3 we obtained a greater number of relations in the thesenet dataset than in the delicious dataset since the thesenet dataset has around 75% more users and even more distinct tags (around 6 times as many), hence a greater probability of having embedded sets of users of common tags.

In the bottom part of table 2 we see that, in total, we obtained 83080 statements from the 3 types of computation applied on our 3 datasets. Few of these statements (31) overlap with each other, i.e. some of them state identical relations between a given pair of tags as other statements established by another method of computation. Likewise, a few statements (22) contradict statements from different methods on the same pair of tags. After removing overlapping and contradictory statements, we obtain a total of 83027 statements.

This automatic handling is performed during low activity periods due to their algorithmic complexity, and each resulting statement is linked to the corresponding type of agent, each modeled as a subclass of `srtag:AutomaticAgent`.

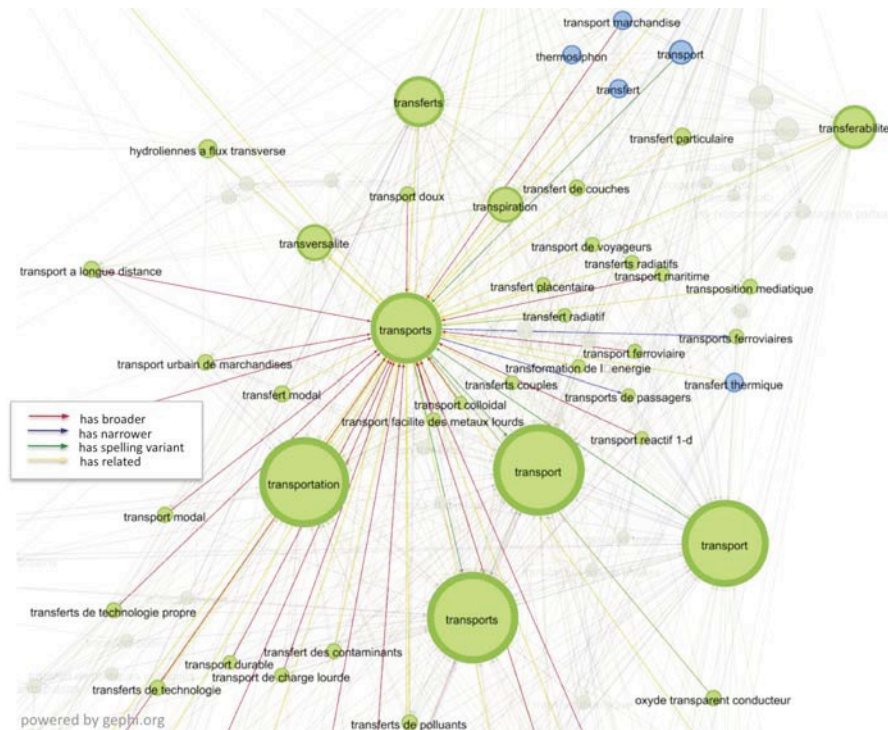


Fig. 5 Example of the results of automatic processing with the String Based method showing tags linked with the tag “transports”. The size of the nodes indicates the number of entering edges (in degree). The green nodes correspond to tags from thesenet and delicious dataset (hence the two nodes “transport”), and blue nodes correspond to tags from caddic dataset.

Moreover, algorithm 2 and algorithm 3 are not incremental since when new tags are added, the structure of the whole folksonomy is modified. This is not the case for algorithm 1 that only compares the labels of newly added tags with all the other tag labels. To give an example of the computation time, the total time to apply this 3 methods on the full dataset is 25647s in our setup, with a machine equipped of a 4 core Intel Core2 Duo processor running at 3.00 GHz with 8Go of RAM. In figure 5 we give an example of the results obtained with the String Based method for the tag “transports”.

5 Capturing and Exploiting Individual Contributions

Up to this point we have presented the different methods of computing tag relations and the model, SRTag, to keep track of the diverging points of view from all users. Now we are going to see how these points of view are first captured, then sorted out and arranged together in a coherent system.

5.1 *Capturing Users Contributions*

Once we are able to support diverging points of view, we want to allow users to contribute to the semantic structuring of the folksonomy while keeping as low as possible the cognitive overhead that this task may involve. To achieve this goal we propose integrating simple and non-obtrusive structuring functionalities within everyday user tasks. For instance, in our target community at Ademe, we want to be able to capture the expertise of the engineers when they browse the corpus of Ademe resources.

The design of the solution we propose is grounded on previous studies and development of collaborative ontology editors conducted in our research team (see [Peron, 2009] for a synthesis). Indeed, these studies set a background of considerations and evaluations regarding the ergonomic aspect of tools allowing the collaborative editing of a shared knowledge representation such as an ontology (ECCO¹³) or a structured folksonomy (SweetWiki by [Buffa et al., 2008]). The ergonomic analysis of the folksonomy editor of SweetWiki revealed several weaknesses that we tried to overcome in our proposal for an interface to capture users contributions regarding the semantics of tags. By taking into account the multiple points of view we make sure that (1) each user is not reluctant to contribute because of a fear to destroy others' contributions, and (2) each point of view is kept in order to obtain a richer knowledge representation in the end.

Our proposal consists in an interface for explaining the computed structure of the folksonomy in which tags are suggested and ordered according to their semantic relations with the current searched-for tag (see figure 6). Related and spelling variant tags are positioned on the right side (respectively top and bottom corner) and broader and narrower tags are positioned on the left side (respectively top and bottom corner). Optionally, users can either merely reject a relation by clicking on the cross besides each tag, or they can correct a relation by dragging and dropping a tag from one category to another.

5.2 *Detecting and Solving Conflicts*

5.2.1 *ConflictSolver Mechanism*

A third type of agent is introduced, modeled with a subclass of `srtag:-AutomaticAgent` named `srtag:ConflictSolver` and which looks for conflicts emerging between all user's points of view. A conflict in the structured folksonomy emerges when different relations have been proposed or approved by different users on the same pair of tags (if a user changes his mind, we simply update his point of view). For instance, the tag "pollution" is narrower than "co2" for a number n_1 of users, but for a number n_2 of users "pollution" is broader than "co2".

¹³ French for Collaborative and Contextual Ontology Editor, see <http://www-sop.inria.fr/edelweiss/projects/ewok/publications/ecco.html>

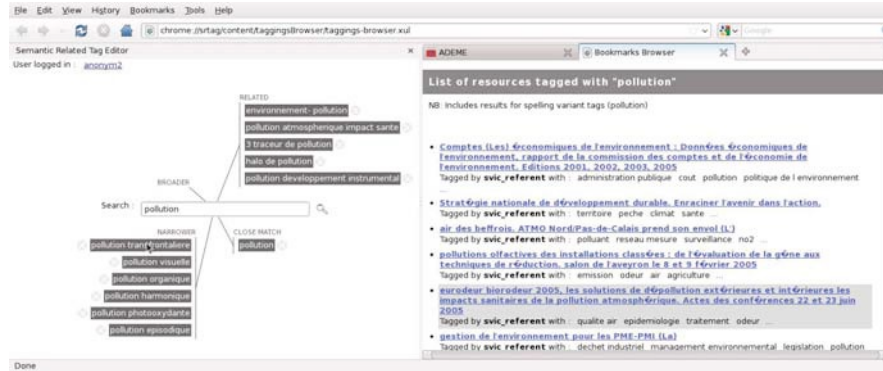


Fig. 6 Firefox extension seamlessly integrating tag structuring capabilities (left part). The user was about to drag the tag “energy” towards the “spelling variant” area to state that the tag “energie” (the tag currently searched for) is a spelling variant of “energy”. On the right side are displayed the resources associated to the current tag.

In addition, other users can say that “pollution” is related to “co2”. In this case the conflict solver first counts the number of approval $nbApp_i$ for each conflicting statement $s_i \in \{s_i\}_n$, n being the total number of statements made on a given pair of tags. Then, it retrieves the maximum $\max\{nbApp_i\}_{i \in [1, n]} = nbApp_{max}$, and compares the ratio $r = \frac{nbApp_{max}}{\sum_i nbApp_i}$ with a given threshold τ_{cs} . If this ratio is above τ_{cs} , then the conflict solver approves the corresponding statement. Otherwise, if r is below τ_{cs} , this means that no strong consensus has been reached yet, and the conflict solver merely says that both tags are related since this relation is the loosest and represents a soft compromise between each diverging point of view. In this case it approves the related statement if it exists, and if not, it proposes its own related statement.

5.2.2 Experiment

Protocol

We have conducted an experiment among 5 members of Ademe. We have presented them with a list of 94 pairs of tags (t_1, t_2) and asked them to choose a semantic relation between t_1 and t_2 among the following: t_1 is a spelling variant of t_2 , t_1 is broader than t_2 , t_1 is narrower than t_2 , t_1 is related to t_2 , or t_1 is not related to t_2 . We have then applied the conflict solver on the set of relations and points of view. When a user chose the fifth possibility, *i.e.* that t_1 is not related to t_2 , we have applied a SPARQL rule to translate this choice into the rejection of all the relations (namely spelling variant, broader, narrower, and related) stated about the same pair of tags. Doing this allows us to consider relations that are debatable, in the sense that some users have approved it, and some other users have rejected it, but none have proposed or approved another relation.

After applying the conflict solver, we are able to distinguish between 4 cases regarding the relation between two tags:

1. Approved statements: when a relation has only been approved.
2. Conflicting statements: when some users have proposed a relation and some other users have approved another relation on the same pair of tags, e.g. some users have approved that “pollution” has broader “pollutant”, and some other users have approved that “pollution” has spelling variant “pollutant”.
3. Debatable statements: when only one relation is stated on a given pair of tags but this relation has been both approved by some users and rejected by some others.
4. Rejected statements: when a relation has only been rejected.



Fig. 7 Result of conflict solving. (a) Distribution of the different cases of conflict solving for all pairs of tags. (b) Distribution of the different cases of conflict solving for each type of semantic relations. (c) Distribution of pairs with compound words compared with pairs with non-compound words for each type of conflict solving cases.

Result analysis

In figure7 we show the detailed results of the conflict solver applied on our dataset gathered from the 5 users who chose one relation for each of the 94 pairs of tags of the dataset. The first chart (a) shows the distribution of the different cases of conflict solving over the 94 pairs of tags. We see that for almost half of the pairs (46%), users proposed several relations for a single pair (Conflicting case).

Then in the second chart (b) we looked at the distribution of conflict solving cases for each type of semantic relation. Since several relations are stated in the conflicting case, we kept only in this chart the relations that were proposed by the conflict solver, i.e. the relations that were supported by a clear majority or proposed as a compromise. We see in this chart that 70% of the close match statements were only approved by users, and that 30% were proposed by the conflict solver. If we look at the broader and narrower case altogether (since these relations are the inverse of each other), we see that they are involved in conflicts in more than 50% of the cases. Lastly, the related relation has never been only approved by users and is either involved in conflicts (48% of the statements) or is debatable (52% of the statements).

We should note here that “related” has been proposed as a compromise without being approved by any user once and gained a clear majority 3 times out of the 43 cases of pairs with conflicts. This means that in most of the cases where “related” is proposed by the conflict solver, this relation serves as a compromise between proposals of other relations. Thus, chart (b) shows that the “close match” is the relation that is the most capable of bringing an explicit consensus, and it is clear that it is easier to agree on the fact that “ecology” and “ecologie” refer to the same notion, than it is to agree on saying that “collective action” is narrower than “collectivity”. Indeed, both tags in the latter case may not directly be related to all users mind, and moreover, the type of relation that these two tags share is disputable and strongly depends on the level of expertise of the user who is to choose a relation (some users with a high level of expertise in the corresponding field will be willing to neatly articulate both notions, maybe opting for broader or narrower, while some other less expert users will simply be willing to account for the fact that there is a relation with “related”, or will even be ready to merge both notions because they are not too concerned about the distinctions that can be made).

In the third chart (c) we examined the influence of another noticeable feature that may distinguish different types of pair of tags. Some pairs of tags consist of a word for the first tag and a compound word for the second tag made of the first tag (as in “pollution” and “soil pollution”) or one of its derivative (as in “pollution” and “pollutants detection”), and this concerns 30 pairs out of 94. In this chart we plotted the distribution between two types of pairs of tags, i.e. pairs with compound words and the rest of the pairs, for each case of conflict solving. The result shows that conflicting pairs are pairs with compound words in the majority of the cases (56%). Likewise, only 18% of the only approved statements and 14% of debatable statements (we recall that in this case only one relation is stated, though it can be approved and rejected) were involving pairs with compound words, and this type of pairs was never at the origin of only rejected statements. This suggests that pairs with compound words are more likely to cause conflicts, and rarely lead to clear consensuses.

5.3 *Creating a Consensual Point of View*

The fourth type of agent we introduced is the `ReferentUser`. The referent user will be able to approve, reject or correct all the relations already existing in the structured folksonomy in order to maintain its own and consensual point of view. The conflict solver mechanism will assist the referent user in her task by pointing out the conflicts already existing in the collaboratively structured folksonomy. Then, all the statements that the referent user has already treated will be ignored in further passes of the `ConflictSolver`. The consensual point of view can be used to generate a hierarchical tag cloud from the folksonomy where broader tags are printed in bigger fonts than narrower tags. This type of tag cloud may be useful to guide the users in giving him a panoramic view of the content of the folksonomy and can be presented at a starting point of the navigation, indicating the broadest

tags, and then, during the search, giving the semantic surrounding of the current tag by showing broader and narrower tags.

5.4 *Exploiting and Filtering Points of View*

At this stage of the process, we obtain a folksonomy semantically structured via several points of view, among which a global and consensual point of view emerges. We present in this section the strategies we propose for exploiting these points of view in order to present a coherent experience to all users of the system.

By keeping track of the type of agents associated to each statement, we are able to give a priority to the suggested tags corresponding to these statements when a user u searches for a tag t . The system issues 5 SPARQL queries looking for statements made on the searched-for tag and each time approved by different types of user but making sure these statements do not conflict with preceding results. All results will then be merged and used to suggest tags semantically related to t . The priority order followed is given below:

1. all statements S_u approved by the user u .
2. all statements S_{ru} approved by the `ReferentUser`, except if they conflict with one from S_u .
3. statements S_{cs} approved by the `ConflictSolver`, except if they conflict with one from S_u or S_{ru} .
4. all statements S_{ou} approved by other users, except if they conflict with one from S_u , S_{ru} , or S_{cs} .
5. all statements S_{tc} approved by the `TagStructureComputer`, except if they conflict with one from S_u , S_{ru} , S_{cs} , or S_{ou} .

This set of rules allows, when suggesting tags to a user during a search, filtering out the conflicting or more general points of view from the other contributions, coming from humans or machines. For example, if the user is searching for the tag “energy”, the system will first suggest tags coming from assertions she has approved, e.g. if current user has approved that “energies” is a spelling variant of “energy”, it will suggest “energies”. We give an example in listing 7.1 of the second query that is issued on named graphs and that looks for statements approved by the `ReferentUser` (line 1 to 4) and that (i) are not rejected by current user (line 5 to 8) and (ii) that do not conflict with the ones approved by the current user (line 9 to 13). For instance if the `ReferentUser` had approved that “energies” has broader “energy”, this assertion will not be included in the results since, in the `SRTag` ontology, the property `skos:closeMatch` (this is the property we use for spelling variants) is declared to be `srtag:incompatibleWith` the property `skos:broader`. The system proceeds with the next queries, following the priority order described above. As a consequence, it allows each user to benefit from the other users contributions while preserving a coherent experience using a referent point of view or, when absent, using the point of view of the conflict solver.

Listing 7.1 SPARQL query used to retrieve statements about the tag “energy” and approved by the `ReferentUser` but not directly rejected by the current user or contradictory with statements he has approved.

```

1 SELECT * {
2   GRAPH ?g {?search-tag ?p ?suggested-tag}
3   FILTER(?search-tag = <http://ex.org/tag/energy>)
4   ?g rdf:type srtag:ReferentValidatedStatement
5   OPTIONAL {
6     ?u srtag:hasRejected ?g
7     FILTER(?u = <http://ex.org/users/me>)}
8   FILTER(!bound(?u))
9   OPTIONAL{
10    GRAPH ?g2 {?search-tag ?p2 ?suggested-tag}
11    ?g2 srtag:approvedBy <http://ex.org/users/me>
12    ?p srtag:incompatibleWith ?p2   }
13  FILTER (!bound(?g2)) }

```

6 Conclusion and Discussion

In this paper, we presented our approach to the semantic enrichment of folksonomies. We propose a socio-technical system in which automatic agents help users in maintaining their personal points of view while still benefiting from others’ contributions, and also helping referent users in their task of building a consensual point of view. Our approach is grounded on a careful usage analysis of our target communities that allows us to include their daily activity in the process.

In order to bootstrap the process, we make use of the automatic handling of folksonomies to extract the emergent semantics. In this regard, we proposed in this paper an evaluation of the main string-based methods. in order to: (a) motivate the choice of the metrics performing best in our context; and (b) evaluate the ability of such metrics to differentiate the semantic relations typically used in thesaurus, *i.e.* to be able to tell when two tags are merely related, or when one tag is broader or narrower than another tag, or when two tags are spelling variants of the same notion. As a result we proposed a heuristic metric that performs this task. This heuristic metric performs best for detecting spelling variants, as expected. The values of the thresholds for this method are chosen after a calibration phase conducted with the help of several Ademe’s agents. Therefore, further studies are required in order to validate the robustness and the sensibility to the threshold values but the objective of this work was to check whether or not string-based distances are relevant to detect other relations that spelling variant, and we have shown here promising results for subsumption relations in cases such as “pollution” which is broader than “soil pollution”.

We have also quantitatively shown that the approaches analyzing the structure of folksonomies are necessary to retrieve semantic relations when tags sharing semantic relations are not morphologically similar, even if they are more costly and not

incremental, and we have presented the results of these three types of method that we obtained on a real world dataset.

In order to capture diverging points of view in the semantic structuring of folksonomies, we proposed a formal ontology that makes use of named graphs to describe semantic relations between tags. The points of view of users are then attached to these asserted relations. By describing the different classes of agents who propose or reject asserted relations, we are able to model a complete life cycle for a collaborative and automatically assisted enrichment of folksonomies. (1) This cycle starts with a flat folksonomy which is first analyzed by automatic agents which propose semantic relations. (2) The users can contribute and maintain their own point of view by validating, rejecting, or proposing semantic relations thanks to a user-friendly interface integrated in a navigation tool. (3) The conflicts emerging from these points of view are detected and (4) utilized to help a referent user to maintain a global and consensual point of view. (5) The result of this process is a folksonomy augmented with semantic assertions each linked to different points of view coexisting with a consensual one. (6) The cycle restarts when new tags are added or when relations are suggested or changed. Semantic assertions are used to suggest tags when navigating the folksonomy, and a set of formal rules allows filtering the semantic assertions in order to present a coherent experience to the users while allowing them to benefit from others' contributions.

Our approach is currently being tested at the Ademe agency to enhance the browsing of its online corpus available to members of the agency and to the public. These tests will also help us to improve the user-friendliness of our interface to browse semantic relationships. In this context the expert engineers of Ademe maintain their points of view so as to reflect on their expertise in a given domain. At the same time, the archivists (our referent users) are assisted in the task of enriching with new tags and semantically structure their global point of view from the collaborative enrichment of the folksonomy.

Our future work includes testing our approach with the users of Ademe. We also plan on exploiting the semantic relations between tags at tagging time to guide and help users provide for more precise tags, and also to provide for additional input material for semantic social network analysis [Ereteo et al., 2009]. We plan in this respect to propose a novel approach to indexing where users and professional indexers, such as the Ademe's archivists, are engaged in a fruitful collaboration leveraged by a tailored automated assistance.

Acknowledgements. This work was funded by French National Agency for Research (ANR) under the reference ANR-08-CORD-011-05.

References

- [Angeletou et al., 2008] Angeletou, S., Sabou, M., Motta, E.: Semantically Enriching Folksonomies with FLOR. In: CISWeb Workshop at European Semantic Web Conference ESWC (2008)

- [Aussenac-Gilles et al., 2000] Aussenac-Gilles, N., Biébow, B., Szulman, S.: Corpus analysis for conceptual modelling. In: 12th International Conference Workshop on Ontologies and Texts at Knowledge Acquisition, Modeling and Management, EKAW 2000 (2000)
- [Bouquet et al., 2004] Bouquet, P., Giunchiglia, F., van Harmelen, F., Serafini, L., Stuckenschmidt, H.: Contextualizing ontologies. *Web Semantics: Science, Services and Agents on the World Wide Web* 1(4), 325–343 (2004); International Semantic Web Conference 2003
- [Braun et al., 2007] Braun, S., Schmidt, A., Walter, A., Nagypál, G., Zacharias, V.: Ontology maturing: a collaborative web 2.0 approach to ontology engineering. In: CKC. CEUR Workshop Proceedings, vol. 273. CEUR-WS.org. (2007)
- [Buffa et al., 2008] Buffa, M., Gandon, F., Ereteo, G., Sander, P., Faron, C.: SweetWiki: A semantic Wiki. *J. Web Sem., Special Issue on Web 2.0 and the Semantic Web* 6(1), 84–97 (2008)
- [Carroll et al., 2005] Carroll, J.J., Bizer, C., Hayes, P., Stickler, P.: Named graphs, provenance and trust. In: WWW 2005: Proceedings of the 14th International Conference on World Wide Web, pp. 613–622. ACM, New York (2005)
- [Cattuto et al., 2008] Cattuto, C., Benz, D., Hotho, A., Stumme, G.: Semantic Grounding of Tag Relatedness in Social Bookmarking Systems. In: Sheth, A.P., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., Thirunarayan, K. (eds.) ISWC 2008. LNCS, vol. 5318, pp. 615–631. Springer, Heidelberg (2008)
- [Ereteo et al., 2009] Erétéo, G., Buffa, M., Gandon, F., Corby, O.: Analysis of a Real Online Social Network Using Semantic Web Frameworks. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 180–195. Springer, Heidelberg (2009)
- [Gandon et al., 2007] Gandon, F., Bottolier, V., Corby, O., Durville, P.: Rdf/xml source declaration, w3c member submission (2007), <http://www.w3.org/Submission/rdfsource/>
- [Golebiowska, 2002] Golebiowska, J.: Exploitation des ontologies pour la memoire d'un projet-vehicule - Methode et outil SAMOVAR. PhD thesis, Universite de Nice-Sophia Antipolis (2002)
- [Heymann and Garcia-Molina, 2006] Heymann, P., Garcia-Molina, H.: Collaborative Creation of Communal Hierarchical Taxonomies in Social Tagging Systems. Technical report, Stanford InfoLab (2006)
- [Huynh-Kim Bang et al., 2008] Huynh-Kim Bang, B., Dané, E., Grandbastien, M.: Merging semantic and participative approaches for organising teachers' documents. In: Proceedings of ED-Media 2008 ED-MEDIA 2008 - World Conference on Educational Multimedia, Hypermedia & Telecommunications, Vienna France, pp. 4959–4966 (2008)
- [Levenshtein, 1966] Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady* 10(8), 707–710 (1966)
- [Limpens et al., 2009] Limpens, F., Monnin, A., Laniado, D., Gandon, F.: Nicetag ontology: tags as named graphs. In: International Workshop in Social Networks Interoperability, Asian Semantic Web Conference 2009 (2009)
- [Markines et al., 2009] Markines, B., Cattuto, C., Menczer, F., Benz, D., Hotho, A., Stumme, G.: Evaluating similarity measures for emergent semantics of social tagging. In: 18th International World Wide Web Conference, pp. 641–641 (2009)
- [Mika, 2005] Mika, P.: Ontologies Are Us: A Unified Model of Social Networks and Semantics. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 522–536. Springer, Heidelberg (2005)

- [Passant and Laublet, 2008] Passant, A., Laublet, P.: Meaning of a tag: A collaborative approach to bridge the gap between tagging and linked data. In: Proceedings of the WWW 2008 Workshop Linked Data on the Web (LDOW 2008), Beijing, China (2008)
- [Peron, 2009] Peron, S.: Etude ergonomique de folkon. Technical report, UNSA, INRIA (2009)
- [Rivière, 1999] Rivière, M.: Représentation et gestion de multiples points de vue dans le formalisme des graphes conceptuels. PhD thesis, Université Nice-Sophia Antipolis (1999)
- [Specia and Motta, 2007] Specia, L., Motta, E.: Integrating Folksonomies with the Semantic Web. In: Franconi, E., Kifer, M., May, W. (eds.) ESWC 2007. LNCS, vol. 4519, pp. 624–639. Springer, Heidelberg (2007)
- [Tanasescu and Streibel, 2007] Tanasescu, V., Streibel, O.: Extreme tagging: Emergent semantics through the tagging of tags. In: Haase, P., Hotho, A., Chen, L., Ong, E., Mauroux, P.C. (eds.) Proceedings of the International Workshop on Emergent Semantics and Ontology Evolution (ESOE 2007) at ISWC/ASWC 2007, Busan, South Korea (2007)
- [Van Damme et al., 2007] Van Damme, C., Hepp, M., Siorpaes, K.: Folksonology: An integrated approach for turning folksonomies into ontologies. In: Bridging the Gap between Semantic Web and Web 2.0 (SemNet 2007), pp. 57–70 (2007)