Multi-points of view semantic enrichment of folksonomies

Freddy Limpens

UNIVERSITÉ DE NICE - SOPHIA ANTIPOLIS

ÉCOLE DOCTORALE STIC
SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

T H È S E

pour obtenir le titre de

Docteur en Sciences

de l'Université de Nice - Sophia Antipolis
Mention : INFORMATIQUE

Présentée et soutenue par

Freddy LIMPENS

Multi-points of view semantic enrichment of folksonomies

Thèse dirigée par Fabien GANDON & Michel BUFFA

préparée à l'INRIA Sophia Antipolis, Projet EDELWEISS

soutenue le 25 octobre 2010

Jury :

President :	Jean-Paul Rigault	-	Université Nice - Sophia Antipolis
Rapporteurs :	Monique Grandbastien	-	LORIA, Nancy
	Nathalie Aussenac-Gilles	-	IRIT, Toulouse
Examinateurs :	Asunción Gómez-Pérez	-	Universidad Politécnica, Madrid
	Lora Aroyo	-	Vrije Universiteit, Amsterdam
	Cécile Bothorel	-	Telecom Bretagne, Brest
Directeurs :	Fabien Gandon	-	INRIA Sophia Antipolis
	Michel Buffa	-	I3S, Université Nice - Sophia Antipolis

à Cécile

Acknowledgements

I would like to first thank my advisors, Fabien Gandon and Michel Buffa, who initiated this project and without whom this work would simply not exist. I am grateful to their constant support, their patience, and all the manifold things I learned from both of them.

I am also grateful to Rose Dieng-Kuntz, who was leading the Edelweiss team when I arrived, from whom I have had glimpses of what a great person of science and heart look like. I wish I had the chance to know her better.

I want to thank also all the members of the Edelweiss team for being workmates that I will sincerely miss, and for contributing to a warm and great atmosphere that I had the chance to benefit from all along these three years spent in their company. In particular, I want to thank: Olivier Corby, for teaching me, among other things, fundamentals of knowledge engineering; Alain Giboin, for rich and fruitful conversations about the crucial human aspects in technological systems; Khaled Khelif and Noureddine Mokhtari, for being nice and helpfull office mates; Guillaume Erétéo, for his kindness and good mood, but also for fruitful collaborations; Emmanuel Jamin, for many insightful discussions; Nicolas Delaforge and Sébastien Comos, for the precious knowledge on diverse aspects of programming they taught me; Isabelle Mirbel and Martine Collard, who visited Edelweiss for some time, for their kindness and advice; all the former members of Edelweiss I met and that also contributed to make this three years a memorable experience : Stéphanie Péron, Corentin Follenfant, Priscille Durville, Leila Khelif, Amel Yessad, Amira Tifous, Mohamed Bennis, Hacene Cherfi, Birahim Sall, Cheikh Anta Diop, Ibrahima Diop, Gaoussou Camara, Abdoulaye Guisse, Reda Boucid, Bassem Makni, Aroua Hedhili, Phuc-Hiep Luong, Virginie Bottolier, Adil El Ghali; and of course all the members recently arrived in Edelweiss, Nicolas Marie, Oussama Cherif, Jerome Maraninchi, Sada Kalidou Sow, Cheikh Mbacké Thiam, Guillaume Husson, and Pavel Arapov.

I thank also Alexandre Monnin and David Laniado who became sort of remote workmates with which I had a genuine pleasure to work, discuss and share some nice moments.

I am as well grateful to Mireille Bossy for her really helpfull advice.

I also want to thank Anne Merle from Ademe for a much appreciated support and collaboration.

I shall not forget to thank the assistant of Edelweiss that have all been extremely helpfull and each contributed to the human warmth of Edelweiss: Angela Ouvrier, Laurie Vermeersch, Claire Senica, Patricia Maleyran, and Christine Foggia.

I also thank the ANR who funded this work through the ISICIL project ANR-08-CORD-011-05.

Enfin, je n'aurai jamais de mots suffisament justes pour remercier tout particulièrement Cécile, ma femme, ainsi que toute ma famille, et tous mes amis pour leurs encouragements et leur précieux soutien.

Freddy Limpens,
Sophia Antipolis, Friday the 3rd of September, 2010

Abstract

This thesis is set in the research effort to bridge Social Web (also called Web 2.0) with Semantic Web. In particular, we looked for ways of bridging Social tagging-based systems with structured representations such as thesauri or ontologies (in the informatics sense). Social tagging platforms allow their users to associate freely chosen signs to their favorite resources. These platforms have recently become very popular as a means to classify large sets of resources shared among on-line communities over the social Web. However, the folksonomies resulting from the use of these systems revealed limitations: tags are ambiguous and their spelling may vary, and folksonomies are difficult to exploit in order to retrieve or exchange information. The goal of this thesis is to overcome these limitations and to support the use of folksonomies with formal languages and ontologies from the Semantic Web, while proposing an approach to take the benefits of social dynamics found on the Web 2.0 for the elaboration of thesauri or ontologies.

This thesis present our multi-points of view approach to the semantic enrichment of folksonomies. We propose a socio-technical system, grounded on a usage analysis, and combining automatic processing of tags and users' contributions through user-friendly interfaces. Automatic processing of tags allows bootstrapping the process by using a combination of a custom method analyzing tags' labels and adapted methods analyzing the structure of folksonomies. The contributions of users are described thanks to our model SRTag (Semantically Related Tag) that allows supporting diverging points of view, and captured thanks to our user friendly interface allowing the users to structure tags while searching the folksonomy. Conflicts arising between individual points of view are then detected and temporarily solved by an automatic agent, whose outcome is then exploited to help a referent user maintain a global and coherent structuring of the folksonomy. Each individual point of view can then be enriched with the others' contributions, with the global point of view serving as a reference to guaranty a local coherence for all users. The result of our method allows enhancing the navigation within tag-based knowledge systems, but can also serve as a base for building thesauri or ontologies fed by a truly bottom up process, providing therefore a solution to the bottleneck effect of knowledge acquisition.

Keywords

Social tagging, Folksonomies, Ontologies, Thesauri, Social Web, Semantic Web

Résumé

Cette thèse s'inscrit dans un effort de convergence entre approches Web Social (appelé aussi Web 2.0) et Web Sémantique. A cet égard, nous nous intéressons en particulier au rapprochement entre folksonomies et représentations structurées de connaissances tels que les theasauri ou les ontologies informatiques. Les folksonomies résultent de la collection de tags partagés au sein d'utilisateurs de plateformes de *social tagging.* Ces plateformes permettent à leurs utilisateurs d'organiser leurs resources favorites en leur associant de manière libre des signes appelées tags. Cependant, ces tags ne présentent aucune structure, et constituent, *in fine*, des listes de termes non organisés qu'il est difficile d'exploiter efficacement pour la naviguation. L'objectif de cette thèse est de fournir des solutions pour améliorer les usages liées au plateformes de *social tagging* tout en proposant une approche mettant à profit la dynamique participative constatées sur le Web 2.0 pour l'élaboration de thesauri ou d'ontologies.

Cette thèse présente notre approche multi-points de vue de l'enrichissement sémantique des folksonomies. Nous proposons un système sociotechnique combinant, à partir d'une analyse des usages de nos communautés cibles, traitements automatiques et contributions des utilisateurs via des interfaces ergonomiques. Les traitements automatiques permettent d'extraire des relations sémantiques entre tags et sont assurées par la combinaison d'une méthode que nous avons mise au point et analysant les labels de tags, et de méthodes que nous avons adaptées et analysant la structure des folksonomies. Notre solution permet à chaque utilisateur d'organiser les tags selon son propre point de vue, tout en bénéficiant des contributions de ses pairs. Ceci est permis à la fois par notre modèle, SRTag (*Semantically Related Tag*), qui permet de representer des relations entre tags tout en supportant les points de vue divergents, et par une interface intégrant des fonctionnalités de structuration des tags dans les taches de navigation au sein de la folksonomie. Les éventuels conflits entre points de vue des utilisateurs sont détectés et temporairement solutionnés par un agent automatique dont les résultats sont ensuite exploités pour aider un utilisateur référent à maintenir une structuration globale et cohérente de la folksonomie. Ce point de vue cohérent est alors exploité pour enrichir chaque point de vue individuel avec les autres contributions tout en garantissant une cohérence locale. Nous montrons de plus comment le résultat de notre méthode permet d'améliorer la navigation dans les systèmes de connaissances à base de tags, mais aussi comment il sert de base à des ontologies ou thesauri incluant la participation des membres de la communauté, proposant ainsi une solution au problème de goulet d'étranglement lors de l'acquisition de connaissances.

Mot-clés

Tagging Social, Folksonomies, Ontologies, Thesauri, Web Social , Web Sémantique

Contents

Le désordre d'une bibliothèque n'est pas en soi une chose grave; il est de l'ordre du "dans quel tiroir ai-je mis mes chaussettes?": on croit toujours que l'on saura d'instinct où l'on a mis tel ou tel livre; et même si on ne le sait pas, il ne sera jamais difficile de parcourir rapidement tous les rayons.

A cette apologie du désordre sympathique, s'oppose la tentation mesquine de la bureaucratie individuelle: une chose pour chaque place et chaque place à sa chose et vice versa; entre ces deux tensions, l'une qui provilégie le laisser-aller, la bonhomie anarchisante, l'autre qui exalte les vertus de la *tabula rasa,* la froideur efficace du grand rangement, on finit toujours par essayer de mettre de l'ordre dans ses livres: c'est une opération éprouvante, déprimante, mais qui est susceptible de procurer des surprises agréables, comme de retrouver un livre que l'on avait oublié à force de ne plus le voir, et que, remettant au lendemain ce qu'on ne fera pas le jour même, on redévore enfin à plat ventre sur son lit.

Georges Perec, *Penser/Classer,*
Notes brèves sur l'art et la manière de ranger ses livres

L'art technique porte in fine sur le fait humain et culturel, pour être pleinement technique, et pas seulement scientifique. Ainsi, l'enjeu n'est certainement pas de mettre l'homme au cœur de la technique, ou d'avoir une technologie centrée sur l'homme, ou enfin d'introduire la dimension humaine au cœur de la technique. Car la technique, si elle doit aller au bout de sa logique, doit intégrer les sciences de la culture sans changer sa nature, mais en l'accomplissant. La technique est et a toujours été humaine.

Bruno Bachimont, *Arts et sciences du numérique:*
Ingénierie des connaissances et critique de la raison computationnelle

CHAPTER 1

Introduction

Presentation and context of the thesis

The Web has now become a global socio-technical space of communication and exchange of information that gives access to an ever-growing number of resources of many different types. However, it has also become paradoxically more and more difficult to cope with the mass of information that floods our conscience in today's knowledge streams aired on the Web. Bachimont (2005) theorizes this problem as *a symbolic disorientation* that we witness at our digital age. According to him, the principles at stake in the interpretation of content shall not be found within the content itself, and if computers help us deal with the manipulation of contents at an ever increasing scale (as evidenced by the Web infrastructure itself), they do not necessarily help us deal with the complexity of the interpretation of these contents. Computers have thus a contradictory result of helping us to process an increasing mass of content, but letting us alone in front of the mass of results of these computations. And this problem is well illustrated with the mass of information that we have to interpret when navigating the Web.

It is in this context that knowledge engineering, as a discipline, and the Semantic Web, as a technical framework, can help us interpret content and overtake the symbolic disorientation. The Web is not a cause of our disorientation but an amplifier that enlarges the amount of information to interpret. Furthermore, interpretation strongly relies on representations of the world that are formed by our conscience. Representations are mediations to the real world which should provide a meaning to our actions. Thus, the problem is not the trueness of these representations, but their intelligibility, and the goal of knowledge engineering is precisely to provide tools and methods to help us build more comprehensible representations capable of guiding interpretation (Bachimont, 2005). The Semantic Web is an evolution of the Web where these representations are exploited to better organize the mass of information and data available on the Web.

This is where structured knowledge representations such as ontologies, topic maps, or thesauri have been proposed by research work in knowledge engineering to guide users within massive amounts of content. These symbolic representations can be seen as conceptualizations of a field of knowledge expressed with the help of a series of formalisms and languages. However, they remain costly to build and relie on knowledge acquisition, which is a difficult task, as it requires processing large amounts of information covering the whole spectrum of the community's expertise.

In parallel to the development of the Semantic Web, the Web has evolved towards what we call *Web 2.0* or *Social Web*. This evolution has opened up new ways of interacting and has brought novel platforms where users are able to contribute to the creation of collaboratively edited contents, or to tag or comment the contents published online. These platforms also allow users to gather around their center of interest to form *online communities of interests* that are at the origin of the creation of popular information resources such as, for example, the WikiPedia, or other online forums devoted to specific topics[1].

Social Web and Semantic Web have for sometimes been opposed as incompatible paradigms to the elaboration of knowledge-based platforms, the first being seen as a bottom-up approach, based on the contributions of the communities' members, and the second as a top-down approach, based on the intervention of experts. As an attempt to challenge this opposition, the approach we detail in this dissertation is aimed at helping online communities of interest better handle their knowledge based systems by combining the social and participative dynamics found in Web 2.0 platforms with Semantic Web formalisms. In this regard, we particularly focus on enhancing tagging-based systems that constitute one of the most representative technologies of typical Web 2.0 platforms.

Tagging consists in associating freely chosen strings of characters[2] to a resource. When tags are shared among a community, social tagging becomes a powerful means to allow users to share their personal and unconstrained classification of their favorite resources. However, folksonomies resulting from the collection of users' tags suffer from a lack of precision and a lack of explicit links between tags that bring significant obstacles to their full exploitation.

This thesis thus proposes an approach for cross-fertilizing the richness of folksonomies and the possibilities brought by Semantic Web formalisms. In this respect, we will show that folksonomies can be better exploited by using semantic metadata and by semantically linking tags in order to guide users when navigating among tagging-based platforms. Moreover, the usage of this type of system can serve as an opportunity to capture the knowledge of the community in order to build structured representations. Furthermore, we strive to involve users in the process of semantically enriching folksonomies without overloading them while allowing diverging points of view in order to make the structured knowledge representation yielded by this process as representative as possible of the community's richness and diversity.

This dissertation presents my work during the three years I spent as a PhD candidate within the Edelweiss team of INRIA - Sophia Antipolis[3]. Edelweiss[4],

[1] The reader is likely to be already accustomed to a number of them, as these online forums cover a broad variety of topics such as health (www.doctissimo.fr), technical support (ubuntuforums.org), diy (forum.doityourself.com), etc.

[2] Indeed, it does not have to be a word to be called a tag, it could even be any kind of sign (Monnin, 2009)

[3] http://www-sop.inria.fr/

[4] http://www-sop.inria.fr/edelweiss/

previously known as Acacia[5], has much experience on knowledge management (Dieng *et al.*, 2005) and methodologies and tools, anchored in the Semantic Web, for supporting collaborative exchange and production of knowledge in the context of communities of interest. The evolution of the Web towards a participative space of co-creation of knowledge made up a natural extension of the field of investigation for the Edelweiss team, which provided us with a fruitful context for our research.

Moreover, I participated in the ISICIL[6] project that illustrates the current research interests of Edelweiss as it is an attempt to reconcile Web 2.0 paradigms and Semantic Web methods and tools in the context of science and technology monitoring in organizations. This project, and in particular the close collaboration with one of its partner, the Ademe agency, has brought a rich and concrete terrain that helped us challenge our proposals all along this thesis. Our approach is thus aimed at helping communities of interest that can be found on the web or within organizations, and that rely strongly on online tools to communicate, collaborate, and exchange knowledge.

To conclude this introduction, we can summarize our research goal with the following question: *How can we help online communities of interest semantically enrich their folksonomy in order to both enhance the use of tagging based systems and to obtain a rich and structured representation of the knowledge of all their members ?*

Organization of the dissertation

This thesis dissertation is divided in 7 chapters plus this introduction and a global conclusion, and it is organized as follows.

Chapter 2, Motivating scenario and context of the thesis, on page 7

We will start by detailing the background and the motivations that lead us to do this thesis. This chapter will also provide a typical motivating scenario that illustrates with a concrete example the expected outcome of this thesis. It will then present the main technical and scientific challenges this thesis addresses.

Chapter 3, State of the art on bridging folksonomies, thesauri and ontologies, on page 17

This chapter will give the reader a thorough review on the current approaches for bridging folksonomies with structured knowledge representations such as thesauri and ontologies. This research work focuses on the nature of folksonomies, analyze their usage, but also their network structures. Then a substantial part of this chapter is devoted to methods that automatically extract the semantics emerging from folksonomies, or to methods that semantically enrich folksonomies by

[5]http://www-sop.inria.fr/acacia/
[6]http://isicil.inria.fr/

describing them with the help of ontologies or by extending them with external termino-ontological resources. We then give a brief overview of systems that make use of these improvements, and detail our positioning regarding the state of the art. This chapter ends with a summary of the definitions of the main concepts used in this dissertation.

Chapter 4, Modeling tags and folksonomy enrichment, on page 81

This chapter will then present NiceTag, our model of tagging. We will present the motivations to propose another tagging model that is aimed at overcoming the lack of pragmatics in current models. One of NiceTag's goals is to serve as a pivot model in order to improve the interoperability while maximizing the expressivity of the modelisation of tag actions. The last part of this chapter will then introduce our approach to folksonomy enrichment that should be seen as a complement to the enrichment brought by NiceTag. We will hence present the lifecycle of the enriched folksonomy whose steps will be further detailed in the remaining chapters.

Chapter 5, Combining methods to infer tag semantics, on page 103

The first step of the folksonomy enrichment we propose consists in automatically computing semantic relations between tags. To this end, we will introduce in detail the heuristic string-based method we designed that extract semantic relations between tags by analyzing their labels. This heuristic is based on the combination of standard string-based metrics, which we selected after a benchmark that evaluated the ability of such metrics to detect a variety of semantic relations. The second part of this chapter is then devoted to our adaptation of two other state-of-the-art methods that analyze the structure of folksonomies.

Chapter 6, Allowing diverging points of view on the semantic structuring of folksonomies, on page 147

This chapter will cover in detail our motivations for supporting multiple points of view in the process of folksonomy enrichment and the SRTag model we propose for that purpose. We will also detail the specificity of the notion of point of view that we consider and how we translated it to our problem. We then detail the principles of SRTag and give examples of its use to describe semantic relations between tags while allowing diverging points of view.

Chapter 7, Combining and exploiting individual points of view, on page 167

After the introduction of our multi-points of view model, we will move on to describing how we can exploit these possibly diverging points of view. The first step in this regard is the detection of conflicts that may arise when several relations are proposed for the same pair of tags. We will detail the strategy we propose to deal with this situation and also how we can help a referent user maintain a global

and coherent point of view after this step. This chapter also features the report on the experiment we conducted at Ademe with a set of users. We present there the analysis of the results and the interpretations we can draw from them. Finally, the last part details the strategy we propose to enrich each user's point of view with others' points of view while preserving its local coherence.

Chapter 8, Implementation of a semantic tagging-based system fostering multi-points of view enrichment of the folksonomy, on page 193

The second to the last chapter of this dissertation will cover the implementation of the enhanced tagging-based system that we envision in our approach. The goal of this chapter is to show the readers how such a system has been implemented in the context of the development of the ISICIL solution. We will present the specifications of the back-end and front-end part, paying particular attention to the design of the interface that captures individual contributions.

The organization of this dissertation is aimed at first giving a broad view of the context of our thesis and the relevant literature; the next chapter introduces our model for tagging and the lifecycle of semantic enrichment of folksonomies that is then detailed by chapters 5, 6, and 7, chapter 8 giving an illustration of how our system has been implemented. After having concluded this dissertation, we give the perspectives we envision for improvements and extensions to this present work. In particular, we discuss the challenges posed by knowledge based systems in the light of the improvements brought by bridging Social Web and Semantic Web approaches that, we believe, will constitute the future of the Web.

CHAPTER 2

Motivating scenario and context of the thesis

Abstract. This chapter's goal is to detail the motivation at the origin of this thesis and the context in which it took place. The Social Web, also called Web 2.0, and the Semantic Web consist in two major evolutions of the Web. The Social Web met an indisputable success, and the Semantic Web promises to help overcome the limitations in the exploitation of the wealth of data produced and exchanged over the Web. This thesis took place in the context of the ISICIL project that is aimed at combining the best of both Semantic and Social Web to help organizations address their knowledge management issues. The problem thus stated, we illustrate it with a detailed scenario that will allow the reader to have a broad view on the problems and challenges that this thesis aims to address.

Contents

2.1 Introduction

The Social Web, also called Web 2.0, consists in one of the major recent evolution of the Web. This notion denotes the shift from a read-only web to an online space of exchange, debate, and collaborative creation of knowledge repositories (such as Wikipedia, or numerous forums providing useful practical information *e.g.)*. This phenomenon can be partly linked to the development of new kinds of online tools that enabled usually passive users to become active participants and producers of content. Typical tools that came along with this evolution of the Web include Wikis, blogs, and forums, which all help users easily post new contents online

without the burden of classical web sites development. Social tagging is another example of such tools involving producers and consumers in the indexing of the contents they share.

The success of the Web 2.0 found some echo within organizations that are concerned with monitoring technological and scientific advancements. Thus, these organizations are willing to upgrade their information systems and to import these tools in order to foster new dynamics of knowledge exchange and discovery among their members.

However, the tools that made the success of the Social Web have some limitations. Their information infrastructures still lack efficient means to interoperate, and freely contributed tags revealed difficult to exploit for navigation and search purposes.

It is in this context that the tools, methods, and formalisms brought by the Semantic Web seem to offer some potential solutions that are worth investigating. Indeed, the Semantic Web is aimed at providing means to help share and reuse data within the Web that could improve dramatically the interoperability of Social Web's platforms. Furthermore, this thesis has had for context the ISICIL project whose main objective is to combine social Web tools and practices with Semantic Web approaches to provide novel tools for managing knowledge within organizations.

This chapter aims at describing the context and the motivating scenario of this thesis. We first detail in section 2.2 the scientific background of our study that is at the crossroads of Social and Semantic Web, and knowledge management issues within organizations. Then we present a detailed motivating scenario in section 2.4, and in section 2.5 the scientific and technical challenges that this thesis addresses before concluding in section 2.6.

2.2 Recent evolutions of the Web and knowledge management

2.2.1 Social and Semantic Web

The recent evolution of the structure and practices observed on the Web is often called Web 2.0 or Social Web, and this can be seen as a result of a social dynamics coupled with simple and intuitive interfaces that allowed users to evolve from simple consumers to producer, tagger, commenter, etc. of the contents aired on the Web. Social tagging plays a particular role in this context, and consists in letting users associate freely chosen character strings to the resources they thus tag. Tags are nowadays a key feature of the Social Web and a new form of expression that can serve many purposes: categorizing or classifying content, comment, vote, react, express, share, identify, etc. Social tagging and the resulting folksonomies can be seen as a new opportunity to involve users in a novel relationships with content they exchange, read or publish online.

The Semantic Web consists in a novel paradigm that aims at leveraging knowledge exchange at the scale of the Web thanks to richer metadata enabling a better interoperability of the informational structures of the Web. The official semantic web group[1] of the World Wide Web consortium defines it in the following way : "The Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries". The core motivation of the Semantic Web is thus oriented towards assisting collaboration and exchange of knowledge at the scale of the Web. Ontologies, in the informatics sense, are at the core of the Semantic Web infrastructure. They consist in a formalization of concepts and relations aimed at representing the knowledge of communities in order to gain access to the data from conceptual descriptions rather than from the addresses of the location where they are stored. But ontologies are costly to build, and this has prevented the Semantic Web to meet a success similar to that of the Social Web.

Thus, we are now facing a situation in which, on one side, the Social Web brought a wealth of exchange and production of knowledge, and on the other side we have a set of Semantic Web technologies that are ready to be used by online communities to enhance their practices, but only miss a better formalization of the shared knowledge, the ontologies, in order to be fully usable. One of our goal in this respect will be to first show that opposing Social and Semantic Web is counterproductive, and that, on the contrary, both of these concepts refer to different dimensions of the Web that have the potential to mutually enrich each other and make the Web an open space of sharing and collaborative elaboration of knowledge. Following from this, our goal is to set up synergetic processes where the users are helped to elaborate knowledge representations that, in return, help them better exploit the potentials of the knowledge bases they directly use or that are relevant to them but remained unexploited due to a lack of interoperability.

2.2.2 Evolutions in knowledge management within organizations

In parallel to the recent evolution of the Web, a growing number of organizations, which rely heavily on monitoring scientific and technological changes (public sector institutions, big companies, etc.), are trying to apply the paradigm of Web 2.0 by importing the tools and practices that contributed to the success of this evolution. These organizations are setting up blogs and wikis to foster knowledge exchange among their members, such as Motorola, just to mention an example, which deployed 4400 blogs and 4200 wikis.

The challenge for these organizations is to overcome classical knowledge management issues. Already known attempts in this respect consisted in (1) developing exhaustive knowledge-based systems available via intranets and dedicated portals in order to externalize and share explicit knowledge. But this systemic approach of knowledge management revealed limited, and therefore managers

[1] http://www.w3.org/2001/sw/

(2) set up communities of practice (Wenger *et al.*, 2002) consisting in semi-formal groups sharing virtual collaborative workplaces, but also meeting at regular intervals. This approach did not meet the expected success since employees did not feel so enthusiastic about exchanging knowledge with persons they had never met (Vaast *et al.*, 2006).

This is why organizations looked towards the Social Web. For example, in social networks, one can observe people having rich and frequent online exchanges with persons of their informal social network, *i.e.* with people they know in the real life. But we also observe very efficient exchange and collaborative construction of knowledge, such as *e.g.*, in online forums where people share valuable information with people they, in some cases, never met or will probably never meet. Hertzum & Pejtersen (2000) also showed in this respect that, within organizations, individuals who are seeking information on a given topic first contact people they know prior to exploring documents databases. Thus, it seems that organizations have a lot to benefit from allowing their members to maintain their informal social network, and to exchange their knowledge in flexible manners, for instance through tagging based platforms (with social bookmarking tools), or micro-blogging tools (such as Twitter.com), etc. Tags in this regard have, we believe, a potential that is still underestimated in linking people with objects and objects (or resources) with objects (id.), tags having the advantage over classical classification systems to be freely chosen by users, and thus closer to the rapid evolution of most recent topics.

2.3 Context of the thesis

This thesis took place in the context of the ISICIL[2] project, which is focused on transferring the use of Web 2.0 tools to communities who have special needs in monitoring current innovations and information within their field of expertise. This type of activity is also called competitive intelligence, but in our case it also concerns organizations wishing to keep up with the state of the art of their domain, such as sustainable development in the case of the Ademe agency, one of the end-users of the ISICIL project.

Our targeted end-users are communities exchanging knowledge online and who may work together or not, but who share strong common interests. We refer to this type of social group as *online communities of interest*. For example, in the Ademe agency, expert-engineers do not directly work together, but they all share common interests and access the corpus of Ademe's documents via the same tools online.

One of the objectives of ISICIL is to promote the use of tagging-based systems enhanced by semantic technologies in order to make explicit tacit knowledge of experts. This type of systems typically allows their users to share, to comment, to index, or to edit any kind of document (photos, bookmarks, wiki pages, etc.). As a kind of resource, bookmarks can also be seen as an opportunity to index shared

[2]http://isicil.inria.fr/

documents as long as they are not kept private but are shared with other community members, just as delicious.com could be seen as a collaborative indexing tool for the Web. In this regard, we believe social tagging practices observed on the Social web can be transferred to the smaller scale of organizations and communities of interests, and that semantic technologies can successfully both help organizations challenge the shortcomings of current knowledge management approaches, but also contribute more generally to knowledge exchange in the Social Web.

2.4 Motivating scenario

In order to illustrate the goal of the ISICIL project, and more specifically of our approach to semantic enhancement of tagging-based systems, we describe in this section a typical motivating scenario in technology and science monitoring from one of the ISICIL end user, the Ademe agency. The scenario-based approach we use here follows the method proposed by Giboin *et al.* (2002), and this particular scenario has been inspired by interviews with members of Ademe who explained the workflow of the producing and indexing of documents and reports.

Context and objectives. Paul is an expert-engineer for Ademe and his position leads him to often produce reports and expertise about latest advancements in the field of renewable energies. For the Info-Energie network, Paul is in charge of delivering a strategic report on offshore wind turbines.

Current scenario and usages. In order to collect information, Paul uses a set of different search engines that come each with their own interface and principles. He also uses Caddic, the internal repository of documents produced by other experts at Ademe and external documents. Documents in Caddic are indexed by the archivists' service, which maintains for this purpose a list of controlled keywords. As a complement, Paul looks up on the Web through different search engines, several other intranet applications, his bookmarks, and documents stored in his personal computer. When using all these different sources, he reformulates each time his query without any suggestions to build or to refine his search. In addition, Paul contacts some acquaintances working at Ademe or outside, and submits a request to the archivists who also have some technical and scientific watch activities on their own in order keep the Ademe's corpus up to date. Paul then proceeds to write the report creating a plan from scratch and copy-pasting elements from the relevant documents he has collected. Upon completion, the report is sent merely to the Info-Energie network, and Paul keeps a personal copy on his machine. When Paul submits his report to the archivists, he is asked to suggest some keywords in order to enrich the list of controlled keywords, but this is done informally with a note bound to the report.

Targeted scenario and usages. Paul owns an account at the ISICIL platform, an internal web application that automatically connects when he starts his web browser. His personal space on ISICIL gives him the latest news tagged with the tags he has subscribed to, but also related tags resulting from the process of semantic enrichment of the folksonomy. In this way, he is informed on the updates of the web sites he has bookmarked, on posts submitted by members of his social network, and new documents or posts (blogs, micro-blogs, wiki pages) related to his topics of interest. When he searches for information, a unified interface allows him to select the sources into which he wants to tap, but also guides him by suggesting tags semantically linked to the terms of his query. This interface enables him also to organize tags with thesaurus relationships through a user-friendly interface so that he can, for example, state a hyponym relation between the tag "wind turbine" (the *broader* term) and the tag "offshore wind farms" and the tag "offshore wind energy" (the *narrower* terms) in order for him to be warned about news on these two narrower topics, or to enrich his search. These semantic links are managed in a multi-points of view fashion so that Paul can maintain his own points of view independently of the other users. Then the system collects each individual point of view, strive to solve the conflicts, and submit this structured folksonomy to the Ademe's archivists review. The archivits maintain a global and coherent point of view used for the Ademe's thesaurus. Along with the search tools, ISICIL provides Paul with a wiki that allows him to edit his report, possibly starting with a draft automatically generated from the notes he added while bookmarking resources he found during his search. This wiki supports collaborative editing features and enables Paul to share his report with members of his social network so that they can leave comments, draft up some parts, or suggest additional relevant resources. Once completed, this report becomes available for further searches to all members of Ademe in addition to being sent to the Info-Energie network. To enhance its visibility, Paul tags this report, and while doing so, he is suggested a list of semantically related tags to help him enriching his tagging. This new report is also automatically available to the archivists that are also warned about the semantic links Paul has proposed between the tags of the folksonomy. The archivists can then validate these semantically linked tags and include them in the global thesaurus they build with the help of the contributions of all members of Ademe. The thesaurus, in return, helps users navigating the Ademe's corpus and is included in the structured folksonomy in order, among other things, to enrich tag suggestion.

Further details on the analysis of the activities of knowledge exchange and elaboration in both of ISICIL end-users, Ademe and Orange Labs, can be found in (Giboin *et al.*, 2009)

2.5 Scientific and technical challenges

Let us now give an overview of the scientific and technical challenges current information and tagging-based systems are facing regarding the envisioned scenario.

Modelling heterogeneous tagging data. As illustrated in the scenario, current content management systems and knowledge databases do not interoperate well, which results in the users having to reiterate the same queries several times. This problem exists in intrawebs, but is even more important on the Web, and is also true for tagging-based systems. All information and tagging content remain "isolated" within "information silos" and unreachable to other sites, which hinders the potential of cross discoveries. For example, when checking out blog posts tagged with "wind turbine", we currently have no means to query several blogs at a time with a precise tag-based query. Furthermore, if social tagging brought solutions to the scarcity of annotations, current implementations or models of tagging did not cover all the potential and diversity of use tags can take on, and in the previous example, different models can be used to link the term "wind turbine" with resources.

One way of overcoming this consists in using standard schemes to describe tagging data that should be followed by all the administrators of tagging-based systems. However, several models already exist (SCOT[3], NAO[4], CommonTag[5]), and targeting a single model would overlook the manifold forms and uses that tags can take on. Indeed, tags can sometimes easily be linked to unambiguous meanings, or follow a given syntax that is to be recognized by some APIs (such as Flickr machine tags). **One of the scientific objectives of this thesis is to propose a flexible model to represent tagging data that allows both querying across tagging repositories while respecting the diversity of tags.**

Enhancing folksonomies for search and navigation purposes Folksonomies resulting from social tagging practices have some limitations. In particular, the spelling variations of similar tags and the lack of semantic relationships between tags hinder significantly the possibilities of navigation within tagged corpora.

One way of tackling the limitations of folksonomies is to semantically structure them with languages from the Semantic Web. 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) hierarchically organizing tags (using SKOS[6] e.g) to guide novice users in a given domain more efficiently than with flat lists of tags or occurrence-based tag clouds. Navigation within tagging data spaces can also be enhanced by linking tags to external termino-ontological resources, such as thesauri or lightweight ontologies in order to gain, in return, the semantic links in these structures, or to help disambiguate tags. **One of the scientific objectives of this thesis is to provide a method to enrich folksonomies by semantically structuring tags or by linking tags to termino-ontological resources.**

[3] http://scot-project.org/scot/ns#
[4] http://www.semanticdesktop.org/ontologies/nao/
[5] www.commontag.org
[6] http://www.w3.org/TR/skos-reference/

Knowledge acquisition for the construction of structured knowledge representations. Another challenge of the semantic enrichment and the construction of shared and structured knowledge representations lies in the coverage of the whole community's field and the integration of the expertise of all users. This is the well-known bottleneck effect of knowledge capture processes where the amount of knowledge to be integrated exceeds the capacity of the system to acquire it. And this is a key problem in the context of organizations, especially in domains of activities that evolve quickly, like the domain of environmental issues in the case of Ademe. This is why folksonomies have been praised to be a solution to this old problem since they are grounded on a bottom-up principle that allows all users who tag to contribute to the final result. However, folksonomies alone are not sufficient, and involving users in their semantic structuring should be as unobtrusive as possible and take the benefit of already existing tasks, such as, for instance, when submitting documents, as Paul does in the end of the scenario. The structuration of folksonomies can then directly be injected into the process of ontology or thesaurus construction, thus lowering down their costs. **One of the scientific objectives of this thesis is to propose an approach to folksonomy enrichment that integrates the point of view of all users without overloading them in order to help our target communities build structured knowledge representations.**

2.6 Conclusion

This thesis is anchored at the crossroads of Social and Semantic Web. These two recent and major evolutions of the Web are complementary aspects of what the future of the Web can look like. The Social Web brought promising technologies such as social tagging that, however, suffer from some limitations that the Semantic Web can help overcome. In parallel to these recent evolutions, organizations are trying to ground their knowledge management methods on Social Web paradigms by importing tools and by encouraging internal dynamics similar to those that made the success of Web 2.0. The ISICIL project, which sets the context of this thesis, is aimed at proposing a novel approach and original tools for assisting members of organizations in technological and scientific monitoring tasks. The goal is to help them to search for, collect, and organize information relevant to them, and to navigate across their networks of acquaintances thanks to unified interfaces that are able to guide them and to suggest related notions to both broaden and refine their search.

Thus, our targeted end-users are both members of organizations for whom exchanging knowledge consists in one of their core activities, but also members of online communities of interest such as, *e.g.*, open-source software developers communities, or contributors to collaboratively edited knowledge bases such as EkoPedia[7]. In particular, the Ademe agency is a typical targeted end-user of our approach, and we will refer to their practice and organization all along this thesis.

[7] http://en.ekopedia.org

The Ademe agency is characterized by a network of expert-engineers, scattered around different antennas, and bound by common interests and the use of the same online tools to access the Ademe's knowledge bases.

We have proposed in this chapter a scenario that illustrates typical situations we want to help improve thanks to the outcome of this thesis. In this scenario, Paul, a member of Ademe, has to produce a report synthesizing the latest advancements on offshore wind turbines. Currently, Paul has to make his path through a maze of different knowledge sources in the intraweb and among its favorite web resources. In this regard, current social tagging solutions provide a way to foster the discovery of sources of information based on Web 2.0 paradigms, but they are still not sufficient, because current tagging-based platforms do not interoperate efficiently. One of the goal of this thesis is to provide a model of tagging that can serve as a pivot to represent tag data in various situations and across scattered tagging platforms. Another shortcoming of current tagging applications is the lack of semantic links between tags that makes them ambiguous, but also difficult to exploit for search purposes. One of the goal of this thesis is to provide a method to semantically enrich folksonomies in order to enhance search and navigation within tagging data. Paul, as an expert on his domain, is also interested in being warned on new information not only about the tags he has subscribed to, but also about semantically related tags *according to him.* Indeed, this situation could be improved if the Ademe's indexing base included all the points of view of Ademe's members, but this is difficult to achieve because of the well-known problem of the bottleneck effect that prevents from easily including all the diversity and breadth of the knowledge of a community. On the other hand, the Ademe's archivists, who helped Paul in his search, are willing to semantically structure their flat list of controlled terms used for the indexing of the corpus. The end result would consist in a thesaurus that requires, however, a significant effort to be constructed. One of the objectives of this thesis is to provide solutions to allow all users to contribute, without overloading them, to the semantic enrichment of the shared folksonomy, which in return, helps administrators of knowledge bases build structured representations.

To conclude, the purpose of this thesis is to bring solutions to improve, with semantic technologies and models, tagging-based systems used by communities of interest or by members of organizations. The social structure of our targeted end-users plays a crucial role in the design of our solution, in particular the fact that all these groups have administrators for their knowledge platforms, such as the archivists at Ademe, who can play an important role regarding the monitoring of the process and the animation of the community. The expected results consist both in an improvement of the experience of each user, but also in a method to help build structured knowledge representations such as a thesauri from folksonomies.

CHAPTER 3

State of the art on bridging folksonomies, thesauri, and ontologies

Abstract. In this chapter we give a detailed presentation of the current approaches to bridging folksonomies, thesauri, and ontologies. Generalier, these research works propose a novel vision of knowledge exchange that strives to reconcile the openness and profusion characteristic of the Social Web with the methods and formalisms of the Semantic Web. In this regard, ontologies are used to represent folksonomy data in order to improve the interoperability of the tagging-based platforms. Then, the ambiguity of tags and their lack of semantic links hinder dramatically the navigation within folksonomies and lowers down their potential for organizing information. Hence, a lot of works propose methods to extract the semantics that can emerge from folksonomies, or to link tags with well-defined concepts from other termino-ontological structures. Some applications already making use of these improvements are presented in the end of this chapter.

Contents

3.1 Introduction

3.1.1 Social tagging and its limitations

To share and index the large number of resources available on the Web raises several issues that systems based on folksonomies (Vanderwal, 2004), such as del.icio.us for sharing bookmarks, have recently tried to address. On the other hand, the Semantic Web aims at supporting the exchange of information by developing the interoperability between applications available on the Web. To this end, several methods, tools and principles are proposed, among which formal ontologies play a central role. Generally speaking, ontologies are knowledge representations aiming at "specifying explicitly a conceptualization" (Gruber, 1993). More specifically, formal ontologies use formal semantics to specify this conceptualization and make it processable by machines. The obstacles to a generalization of ontologies lie mainly in their cost of design and maintenance.

The problem we address here is the need for the users of social Web platforms to find an agreement about the knowledge representations that support their collaborative use of the system. To this regard, folksonomies are often seen as the bottom-up approach, while formal ontologies of the Semantic Web are considered to be necessarily a top-down approach. In this thesis we try to show that opposing folksonomies and ontologies in this way is counterproductive, and the work we present here shows the potential of combining both approaches in order to collaboratively build up solid knowledge representations that are both representative of the communities of users, and at the same time allow for better retrieval or exchange of information.

The Web 2.0 consists essentially in a successful evolution of web application design supported by some principles and technologies. Social tagging and the resulting folksonomies can be seen as two of those principles that have emerged and met a growing success within Web 2.0 applications. The simplicity of tagging combined with the culture of exchange allows the mass of users to share their annotations on the mass of resources. However, the exploitation of folksonomies raises several issues highlighted by Mathes (2004) and by Passant (2009):

1. synonymy of tags where several tags may refer to the same concept due to (a) the variability of the spelling, as with “nyc”, ”new_york” and “newyork” which all refer to the city of New York, USA, or due to (b) the use of tags coming from different languages (not explicited at tagging time) such as the tag “music” and its french translation “musique”, or regional variants like “synchronize” and “synchronise”, or (c) genuine synonyms like “cab” and “taxi”.

2. homonymy of tags, for one tag may refer to several concepts, as with “paris” may refer to the city of Paris, France or to the city of Paris, Texas

3. polysemy of tags, where a single tag, for example “rabbit”, may refer to different related entities, such that the “fur of the rabbit” or the “meat of the rabbit”.

4. the lack of explicit representations of the knowledge contained in folksonomies where the semantic relations that may exist between tags are not represented, as for example with the tags “car” and “vehicle” where it is possible to state that a “car” “is a type” of “vehicle”.

5. and finally, the difficulties to deal with tags from different languages, since this information is generally not provided at tagging time, and several languages can be mixed in a open web platform, and even for an individual user who uses several languages to communicate. This problem is different than the one in the first point of this enumeration. It concerns the lack of explicit specification of the language of a given tag, which can raise issues when attempting to structure them. For example, if several languages are used to

tag a given resource, it is hard to guess whether some tags are translation of other tags or different concepts.

Another challenge is the need to assist the life-cycle of the folksonomies and the ontologies that support the knowledge bases of social Web applications. Our hypothesis is that the synergy of both folksonomies and ontologies may bring great benefits. Research has been undertaken to tackle the problems posed by the annotation and the exchange of the resources on the Web. The systems or methods they propose strive to reconcile ontology-based models and folksonomy-based models.

3.1.2 The need for a shared vocabulary: tidying up on-line communities

Most of the research works we present in this chapter take place within the social Web that includes all types of groups of people communicating on-line. These communities range from groups of people who do not know each other in the real life but contribute to the same sharing platform (as in Wikipedia or delicious.com where users contribute to an encyclopedia or a social bookmarking database), to collaborators who work together and exchange knowledge on-line.

One of the most commonly cited notions about communities with respect to knowledge sharing issues is probably the notion of Community of Practice (CoP) proposed by Lave & Wenger (1991). The notion of CoP defines a group of people gathered by a commitment to a common activity and sharing common interests, proficiencies, and knowledge. However, other notions have emerged to describe the specificity of on-line communities because the criterion of sharing a common commitment is not always fulfilled in communities communicating on-line.

Tardini & Cantoni (2005) tried to apply the concepts of semiotics (Saussure, 1916; Hjelmslev, 1963) to describe and characterize on-line communities. They distinguish two main types of communities. (1) Paradigmatic communities are groups of people simply having something in common, such as, *e.g.,* the fact of using the same website for the "Wikipedia visitors" community. It is possible to belong to several paradigmatic communities at the same time, and these communities can be embedded in each other such as the community of "eye specialist surgeons" in the surgeons' community. Paradigmatic communities are defined, *a minima,* by the fact that their members share a common point without necessarily being aware of it. (2) On the contrary, syntagmatic communities consist in groups of persons who are aware of belonging to a specific community and are characterized *a minima* by their complementarities rather than by the fact they have something in common. Members of such communities usually collaborate together. This type of community is also very close to the concept of Community of Practice (CoP) proposed by Wenger *et al.* (2002), but is less constrained concerning the commitment to a common activity. For example, members of an online forum about ecological housing construction can be considered to belong to a syntagmatic community, as each member brings his/her own point of view or return on experience, but they are not necessarily involved in a common project as is the case with CoP.

The next step consists in finding criteria to evaluate whether a group of on-line users form a syntagmatic or a paradigmatic community, as this distinction has some consequences on the characterization of the type of knowledge structure that will better fit their needs. For instance, the visitors of a web site form a paradigmatic community, which can evolve into a syntagmatic community as soon as the visitors start exchanging more and realize they have a lot of things in common. To this respect, Tardini *et al.* give five conditions which should be fulfilled for a group of users to form a syntagmatic community: (1) a shared environment of communication, (2) a reasonable level of wealth of exchange, which allows for the discovery of common interests, (3) the arousal of a feeling of belonging to a group, (4) the development of a common symbolic space called the "semio-sphere", and (5) the development of a group identity.

The development of a semio-sphere is particularly relevant to the scope of this chapter in that shared ontologies should depict as closely as possible these semio-spheres, and also in that it seems irrelevant to start building collaboratively an ontology if the community is still at the paradigmatic stage. To this respect, the authors have analyzed several on-line communities (from users of search engines to on-line video-game players) and came to the conclusion that out of the five conditions mentioned above, the common interests, the feeling of belonging, and the development of a common identity are the most important to constitute a syntagmatic community.

The feature of the semio-sphere of a syntagmatic community tells also a lot about the features of the community itself: the more complex the semio-sphere, the more closed the community; on the contrary, the simpler and the more affordable to newcomers the semio-sphere is, the more open the community. This description of the semio-spheres is also close to the distinction between broad and narrow folksonomies (see section 3.2.1).

These insights about the nature of on-line communities is of special interest to us since the purpose of our study is to both (a) leverage the exchange of knowledge by making explicit the tacit links between the tags of each member of our target community, and (b) to help the users of a social tagging platform to form syntagmatic communities by consolidating their semio-sphere and feeling of belonging to the same group thanks to the collaborative process of semantically structuring their folksonomies. Indeed, one of the most recurrent request from Ademe members is to help them constituting and recognizing groups of expertise and interest. Currently, each expert at Ademe maintains a profile in which he or she describes herself or himself with a list of keywords. Since these keywords remain scattered around and unstructured (very much like folksonomies, which remain flat lists of unrelated keywords), it is very difficult to exploit these profiles to form groups of common interests. By helping them to structure the vocabulary they share, we aim at enhancing the mutual recognition of similar interests within the members of Ademe agency.

3.1.3 Comparison of different types of knowledge representations used to index resources

Before presenting the different attempts to overcoming the gap between folksonomies and ontologies, let us recall briefly the main types of structured knowledge representations traditionally used to classify or index resources or documents. These knowledge representations are also called "termino-ontological resources" in the literature and differ mostly from each other in their level of formal structuring, or in their purpose, or in the way in which they are elaborated.

1. **Epistemic classifications** (such as Dewey's classification (Dewey, 1876) used for classifying books in libraries) consist in defining a vocabulary that can be universally shared. This type of classification (but in a more flexible flavor than Dewey's classification scheme) is met for instance in the Dmoz[1] initiative to build a directory of Web pages where specialists debate about categories which should be used to classify all the Web pages.

2. The origins of **thesauri** go back to the 4th century, but the first modern thesaurus is attributed to the British Peter Mark Roget[2]. Modern thesauri and other types of controlled vocabularies, such as taxonomies, consist in notions or concepts that are defined and hierarchically structured. They provide descriptors used to index documents and are aimed mostly at navigation purposes. The notions composing thesauri can be contrasted with the concepts of formal ontologies in that they are oriented towards the descriptions of resources, and are not aimed at describing "what something is", but rather "what something is about" according to the SKOS[3] (an RDF schema for thesauri) definition of the `skos:Concept` class. Moreover, the types of semantic relations linking the concepts of thesauri are usually limited to "broader", "narrower", or "related".

3. Along the expansion of the web, semi-formal and shared knowledge representations have been proposed to organize the information on the Web. Such approaches include **Topic maps[4] (Park & Hunting, 2002),** or, with a greater stress on dealing with conflicting views within the communities of users, "semiotic ontologies" Cahier *et al.* (2005). Primarily, semiotic ontologies and Topic Maps can be used for themselves. In some other cases they can also be considered as an intermediary representation to formal ontologies, in that they are not extended by a "referential formalization"[5] but are based on "semiotic expressions", or "Topics", dealing with a type of semantics tightly

[1] http://www.dmoz.org/

[2] for an historical review of Roget's thesaurus, see Dolezal (2005)

[3] http://www.w3.org/2004/02/skos/core#Concept

[4] http://topicmaps.org/xtm/

[5] in the sense that their semantics is "referential" (Rastier, 1994), that is, based on objective and measurable features of the objects to which the concepts refer.

bound to human interpretation. These approaches differ from formal ontologies in their purpose, which is not to obtain a formal and operational scheme, but rather "description networks" used by humans to navigate a corpus of documents and resources.

4. **Formal ontologies** consist in a specification of the conceptualization of a domain of knowledge with the help of formal concepts and properties linking these concepts (Gruber, 1993). They are at the core of the original vision of the Semantic Web proposed by Berners-Lee *et al.* (2001): "The Semantic Web is an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation". Thus, ontologies are at the interface between humans and machines, and can be seen as the formalization of a field of knowledge, given for a specific problem or task. Bachimont (2000) gives the following definition of an ontology, or more precisely, of the "modeling of an ontology": "Defining an ontology for knowledge representation tasks means defining, for a given domain and a given problem, the functional and relational signature of a formal language and its associated semantics". The definition of this formal mechanisms and the translation of the knowledge of a domain in these formal languages allow in turn for making inferences and expand greatly the possibility of querying when looking for resources annotated with formal ontologies. In addition, we retain from ontologies, in contrast to thesauri, the important notions of *classes* and *instances.* Formal ontologies allow describing entities by instanciating them thanks to sophisticated structures of formal classes. The distinction between classes and instances is irrelevant for a thesaurus, as a thesaurus deals only with notions aimed at describing what a resource *is about*, while classes of a formal ontologies are aimed at describing what a resource *is.*

In comparison with all the above-mentioned structures of knowledge representation, folksonomies can be seen as semiotic representations of the knowledge of a community, but they do not include any semantic structure. They are not either truly elaborated collaboratively, since they consist merely in a social aggregation of individual knowledge. However, their indisputable advantage over the other types of representations we mentioned above is their simplicity (they require a minimal cognitive cost of elaboration (Sinha, 2005)) that made them adopted by a mass of users. We can also note that ontologies in general (formal or semiotic), thesauri, and taxonomies can be grouped under the term "termino-ontological resources". However, this type of resources may be utilized in conjunction with folksonomies in several different ways as we explain below.

3.1.4 Different ways of considering the link between folksonomies and thesauri and ontologies

The aim of this chapter is to present the current approaches to reconcile folksonomy-based and ontology-based or thesauri-based approaches to support social interactions. Bridging termino-ontological resources and folksonomies can be done in different ways:

Extracting the emergent semantics from folksonomies. A first type of approach consists in considering folksonomies as a material to build ontologies by extracting semantic relationships between tags. It is possible indeed to take into account the multiple dimensions of folksonomies as they consist in a triadic structure where tags are associated to resources by people ("who tags what"). This is what Mika (2005), for instance, does in order to extract broader and narrower relationships between tags and to build what he calls "lightweight ontologies", that is, ontologies which consist in an set of terms connected with a limited set of semantic relationships (broader, narrower, related for example).

Linking tags with concepts from ontologies or thesauri. Even if ontologies and folksonomies may remain different entities, several approaches have been proposed to semantically enrich folksonomies by linking tags to precisely defined concepts. Indeed, ontologies and thesauri are characterized by the fact they explicitly define the notions or concepts of their structure. Linking tags and concepts can be done *a posteriori*, that is, when considering an existing folksonomy that one tries to link with termino-ontological resources. For instance, Specia & Motta (2007) have developed a system that applies several semantic treatments to a folksonomy, such as finding equivalent tags or grouping similar tags based on similarity measures computed according to the structure of the folksonomy. Then, they query ontologies on the Semantic Web and try to match the tags from these clusters with classes from ontologies. Another possibility is to ask users to link tags with precisely defined concepts represented by an online resource at tagging time, as Passant & Laublet (2008) propose. The main limitation of both of these types of approach is the limited coverage of currently available termino-ontological resources.

Ontologies as an interoperability framework for social tagging data. Another and rapidly evolving way of considering the link between ontologies and folksonomies consists in exploiting the formalism of the Semantic Web to describe and interlink tagging data. The Linking Open Data project [6] consists in extending the Web with data sources semantically interconnected and which publish varied open data sets in RDF format and following a set of ontologies describing the different types of resources. Ontologies from

[6] http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData/

the Linking Open Data initiative include SIOC[7] used to describe on-line communities exchange, or SKOS[8] used to describe thesauri.

3.1.5 Organization of the chapter

This chapter is organized as follows. In section 3.2, we present the different approaches that analyze the nature of folksonomies and tags. Section 3.3 deals with the analysis of the semantics inherent to the folksonomies and the relationships between the tags that can be extracted in order to build ontologies. Section 3.4 will cover methods which semantically enrich folksonomies or which integrate tagging practices in ontology maturing processes. Section 3.5 will give an overview of different types of usages of knowledge sharing platforms, and section 3.6 will conclude this chapter with a discussion.

3.2 Nature and structure of Folksonomies

In this section we focus on research works that analyze the nature and structure of social tagging systems and folksonomies in order to better understand their dynamics and their semantics.

3.2.1 Folksonomies as collaborative classification means

According to Golder & Huberman (2006), social tagging can be seen as a cognitively lighter alternative system of classification to controlled vocabularies and hierarchical systems, which can be seen in a hierarchy of file system folders for instance. Social tagging is also about sense making since the goal of a tag for its author is to organize its knowledge sources with labels that are a way of making sense of the resources he tags. Tags are then an important sign of what matters for the users and how he describes it.

But social tagging is also about collaborative sense making, and as such, has the potential of revealing the fuzziness of the manifold individual categories merged under the same tag. In the same trend of ideas, Veres (2006) says that tags are the results of ad hoc categorizations, that is, categories which interface between each user's "world model" in order to achieve a goal. But their linguistic properties reveal that tags can also be similar to standard categories in taxonomies.

Golder & Huberman (2006) detailed seven functions that tags may perform for bookmarks in the context of a typical application of social tagging: (1) "Identifying What (or Who) it is About", that is, the topic of the item tagged; (2) "Identifying What it Is", for example an "article", a "blog" or a "book"; (3) "Identifying Who Owns It", or also to whom this bookmark may be forwarded (see also the "network tags" in delicious.com social bookmarking service); (4) "Refining Categories", that

[7] http://sioc-project.org/
[8] http://www.w3.org/2004/02/skos/

is, tags which refine or qualify existing categories, such as numbers; (5) "Identifying Qualities or Characteristics" such as adjectives characterizing the opinion of the author; (6) "Self Reference", such as tags beginning with "my"; (7) "Task Organizing" which correspond to a particular type of *ad hoc* categories, oriented towards a specific task such as "to_read".

Folksonomies have also been characterized by Vanderwal (2004) who distinguishes "narrow folksonomies", in which the personal use of tags is predominant, and "broad folksonomies" in which the use of tags is oriented towards more collective and social purposes (which may correspond in some cases to the first three functions given by Golder & Hubermann). Folksonomies are thus a combination of terms that can serve collaborative categorization, and other terms that are only useful for their authors. This is both an opportunity and a challenge from a knowledge management point of view.

3.2.2 Formal definition

In order to further analyze the structure of folksonomies, we have to model them formally. Hotho *et al.* (2006) thus proposed a formal definition of a folksonomy that they model as a tuple $F := (U, T, R, Y)$ where U, T, and R are finite sets, whose elements are called users, tags and resources, respectively. Y is a ternary relation between them such that $Y \subseteq U \times T \times R$, and is called tag assignment or restricted tagging in the Tag Ontology of Newman *et al.* (2005). A tag assignment is a ternary link between a user, a tagged resource, and a tag. A post is a set of restricted tagging assignments made on a single tagged resource, such as a bookmark in delicious.com.

As a collection of data provided by a group of individuals, a folksonomy can be seen as the collection of the "personomies" of all the users. Let us call Pu the personomy of a given user $u \in U$, where Pu is the restriction of F to u, i. e., $Pu := (Tu, Ru, Yu)$, with $Yu := (t, r) \in T \times R | (u, t, r) \in Y$ that is, the set of all the tag assignments of user u.

Mika (2005) also proposed a formalization of the graph structure of folksonomies. In his approach, a folksonomy is seen as tripartite hypergraph $H(F) = \langle V, E \rangle$ where the vertices are given by $V = U \cup T \cup R$ and the edges by $E = u, t, r | (u, t, r) \in F$ (see the graphic representation of a folksonomy given by Halpin *et al.* (2007) in figure 3.1).

3.2.3 Structure and dynamics of social tagging

Golder & Huberman (2006) proposed one of the earliest quantitative analysis of social tagging in which they discuss its nature as well as the dynamics that can be uncovered with statistical analysis lead on the multi-dimensions structure of folksonomies. Golder & Hubermann give some trends in the use of tags in a social bookmarking system (delicious.com). They remark that users have a tendency to use first more general terms when tagging, the first tag having the greatest fre-

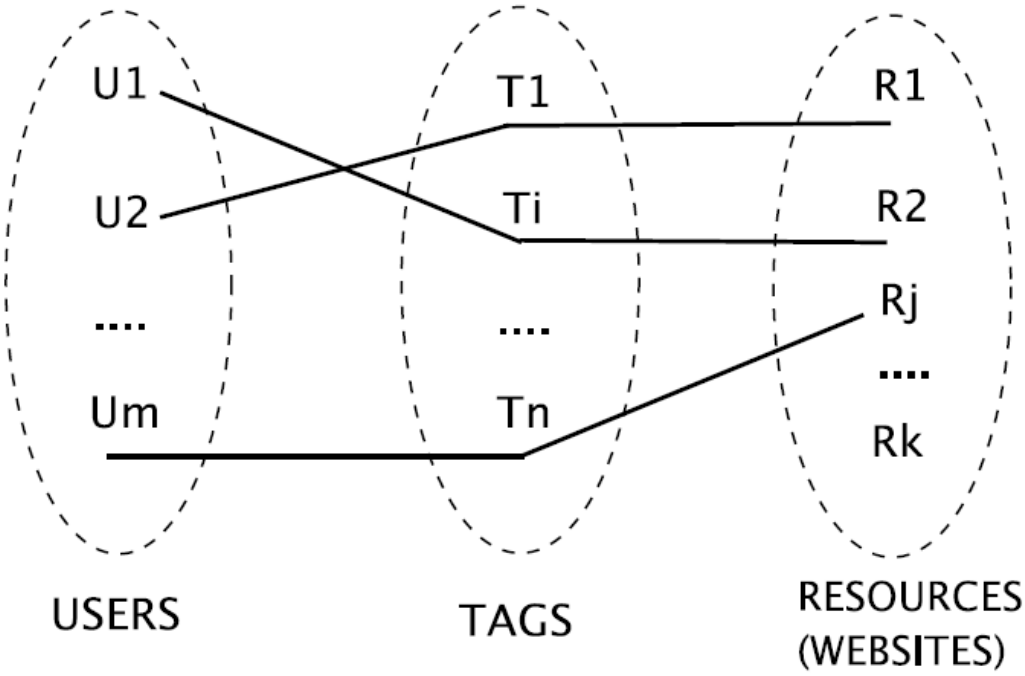


Figure 3.1: Tripartite graph structure of a tagging system. An edge linking a user, a tag and a resource (website) represents one restricted tagging instance or tag assignment (Halpin *et al.*, 2007)

quency of occurrence among all the user's tags, and successive tags having generally a smaller frequency. They also observed stable patterns in the distribution of tags for a given resource (URL in delicious.com). Empirically, once a URL has been bookmarked more than 100 times, each tag's frequency remains in a stable ratio of the total frequency of all the other tags used for this URL.

Halpin *et al.* (2007) pursued this analysis of the dynamics of folksonomies and looked for distribution laws in the frequency of use of the tags. They borrow the hypothesis of Golder & Hubermann and suggest that the most used tags to annotate a resource remain the same after a certain amount of time, and they show that this distribution follows a power law. They verify that hypothesis for the seven to ten tags that are most often associated to popular Web resources posted on delicious.com. These observations may be explained by the theory of preferential attachment, also known as "the rich get richer" principle. This phenomenon, which tends to reinforce, for a given resource, the most often used tags, is even augmented, *e.g.* in the case of delicious.com, by popular tags suggestions while tagging.

But, as Golder & Hubermann suggest, the stability observed in the distribution of the most popular tags persists even for less common tags, which are not shown as suggestions. The choice of the same tags may also be explained by the fact that users share some of the knowledge they express individually when tagging bookmarks. Golder & Hubermann add that this stability in the characterization of

some items is linked with the stability of the ideas and characteristics symbolized by the tags; and that, likewise, this stability may no longer persist when a new concept emerges for describing the same items. This was the case, for example, when the concept "ajax" emerged within the realm of Web designers to describe a set of technologies that were all previously known but not named under a single term.

It is also interesting to look at the distribution of tags for smaller folksonomies as, for instance, Passant (2009) did in the context of a corporate folksonomy. In this folksonomy, Passant (2009) shows that tags follow a distribution in which a lot of tags are used a few times. For example, out of the 12257 tags used to annotate 21614 blog posts, 68% are used at most twice, and only 10% are used more then 10 times. As Hayes *et al.* (2007) showed, it is more difficult to apply classical clustering techniques on this type of distribution in which tags do not neatly partition the annotated data. Indeed, in these cases one should include the content of the annotated data in the analysis of the folksonomy structure.

In order to provide a visual representation of the relationships between tags in a folksonomy, Halpin *et al.* proposed building inter-tag correlation graphs. Each node of these graphs represents a tag and can be seen as a circle whose diameter is weighted by the frequency of occurrence of this tag. The length of the edges of these graphs is weighted by their degree of co-occurrence. The degree of co-occurrence $CoocDegree(T_i, T_j)$ of a pair of tags T_i, T_j is given by :

$$CoocDegree(T_i, T_j) = \frac{N(T_i, T_j)}{\sqrt{N(T_i) * N(T_j)}}$$

Where $N(T_i)$ and $N(T_j)$ denote the number of times each tag T_i and T_j is used individually to tag all pages, and $N(T_i, T_j)$ denotes the number of times two tags are used to tag the same page, summed over all pages. This visualization (shown in figure 3.2) can be seen as a tool for assisting the construction of ontologies out of folksonomies by helping identify visually the most related tags to a given tag.

3.2.4 Looking for common associations in folksonomies

Other works proposed to apply data mining methods to the tripartite model of folksonomies in order to retrieve information in their structure. Jäschke *et al.* (2008) proposed to use formal concept analysis techniques in order to discover the subsets of users sharing the same conceptualizations on the same resources. To do so, they build triples of sets ({R}, {U}, {T}) called tri-concepts where each user of the set {U} has tagged each resource of the set {R} with all the tags of the set {T}. According to the authors, extracting tri-concepts from folksonomies is a first step to build more structured ontologies from folksonomies. Ontologies built in such a way can be seen as shared knowledge representations where each concept is described by a set of tags which belongs to a set of users and are used to characterize a certain kind of resources.

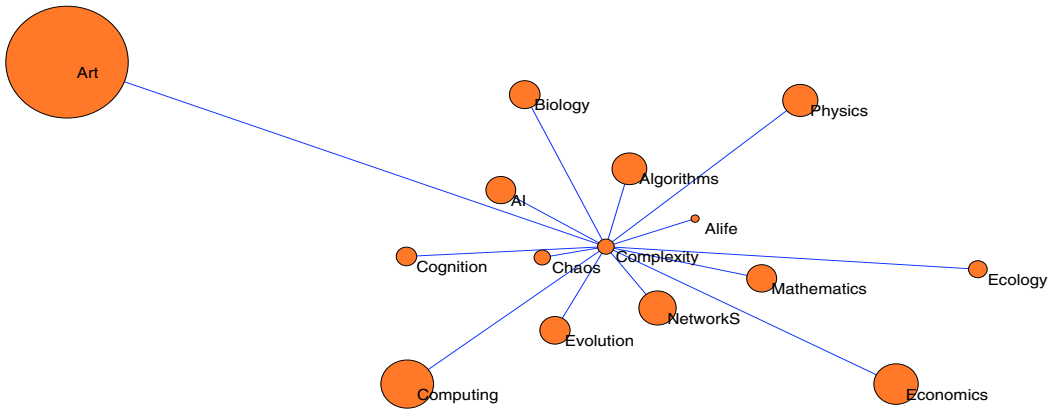


Figure 3.2: Visualization of a tag correlation network (Halpin *et al.*, 2007), considering only the correlations corresponding to one central node "complexity" (data source: delicious.com). The size of the nodes corresponds to the frequency of occurrence of the tags, and the length of the edges corresponds to the co-occurrence degree between the tags.

Other data mining techniques have been applied by Schmitz *et al.* (2006) to extract association rules from folksonomies. The first step is to project the tripartite model (Resources, Users, Tags) onto a two-dimensional structure called a context in formal concept analysis (Wille, 1982). For instance, one can consider all the tuples (Users, Resources) associated to a set of tags T_x. Then Schmitz *et al.* (2006) apply classical rule mining techniques as proposed by Agrawal & Swami (1993). An example of association rule that may be derived from this projection is: all the users associating tags from the set of tags T_A to a set of resources R, often associate the tags from the set of tags T_B to the same set of resources R. This kind of association rule may be exploited for example in a recommendation system. Other types of association rules may be powerful means to identify sub-groups of users sharing the same tagging practices or interested in the same topics, and Schmitz *et al.* (2006) also mention using association rules to learn taxonomic structures of tags.

3.2.5 Comparison and intermediary conclusions

In table 3.1, we compare the different approaches presented in this section. We divided these contributions in two categories.

First, we can mention the qualitative studies conducted on folksonomies. Golder & Huberman (2006) have analyzed the usages of folksonomies and have proposed seven functions that tags may perform for bookmarks in the context of a typical application of social tagging. Vanderwal (2004) distinguished broad folksonomies, where tags tend to be understandable by numerous users, from narrow folksonomies, where tags are more user-centered. Veres (2006) tried to define the linguistic nature of tags and showed, similarly to Golder & Huberman (2006), that

	Qualitative study	Quantitative study
Golder & Huberman (2006)	usages of folkso.	
Vanderwal (2004)	broad/narrow folkso.	
Veres (2006)	linguistic nature of tags	
Mika (2005)		graph structure of folkso.
Hotho *et al.* (2006)		formal definition
(Halpin *et al.*, 2007)		power law distribution of tags
Schmitz *et al.* (2006)		association rules mining
Jäschke *et al.* (2008)		formal concept analysis

Table 3.1: Comparison table of the approach of section 3.2 analyzing folksonomies

some tags correspond to taxonomic categories, while other tags correspond to ad hoc categories serving user's purposes.

Second, we distinguished the contributions that focus more on a quantitative analysis of folksonomies. Mika (2005) and Hotho *et al.* (2006) proposed a formal definition of folksonomies, and Mika (2005) pointed out their graph-like properties and defined them as tripartite hypergraphs. Halpin *et al.* (2007) pursued this analysis of the dynamics and usages of folksonomies initiated by Golder & Huberman (2006) and showed that the distribution of most frequent tags of popular web pages on delicious.com follow power laws. Schmitz *et al.* (2006) applied classical rule mining techniques to discover association rules within folksonomies, and Jäschke *et al.* (2008) used formal concept analysis methods to unveil similar conceptualizations in the tagging of resources shared by groups of users of a social bookmarking site.

3.3 Extracting the semantics of folksonomies

In this section we focus on methodologies and systems aimed at uncovering the emergent semantics from folksonomies. Since usually no explicit semantic relationships are given when users tag, tag semantics have to be first computed by analyzing either tag labels (see section 3.3.1) or the tripartite structure of folksonomies as proposed by Cattuto *et al.* (2008) or Mika (2005) (see section 3.3.2). The semantic interpretation of these measures can be grounded on a third party terminological resource, such as WordNet as proposed by (Cattuto *et al.*, 2008), while others directly inferred semantic relationships out of the analysis of the structure of folksonomies (see section 3.3.3). Another type of approach consists in clustering tags with close similarity measures in order to organize them in bundles or to further process these clusters for ontology maturing processes (see section 3.4.1.1 for the details of this application of clustering)

3.3.1 Dealing with spelling variations

The goal here is to detect and group tags that are equivalent in their meanings or in the topic they describe but are spelled with some variations, such as in "newyork" and "newyork", or "folksonomy" and "folksonomies". In this part, we do not consider the structure of folksonomies but merely focus on the morphological similarity of tags two by two. The main types of methods are the following:

- **String-based methods:** In this type of method, we measure the difference between the string of characters of the tags. This type of method has been used, for instance, by Specia & Motta (2007) to group spelling variants tags.
- **Linguistic methods:** These methods seek to exploit some linguistic or semantic properties of the words to draw comparison between them. For instance, stemming algorithms consist in extracting roots from words (e.g. "links" and "linked" become "link") and grouping tags sharing the same roots. It is also possible to exploit additional resources. For example, (Specia & Motta, 2007; Van Damme *et al.*, 2007) suggest using on-line resources (such Wikipedia, or on-line dictionaries) to check the correct spelling of tags or to find an appropriate representative for a cluster of equivalent tags (grouped together thanks to string-based method for instance).

Euzenat & Shvaiko (2007) also give a detailed overview of these two types of methods when utilized for matching similar concepts from different ontologies.

The detailed presentation of these methods is beyond the scope of this chapter, and we will just present some of the main string-based methods which can be found in the SimMetrics java package[9] that is used in this thesis. A first distinction among the different metrics to be used to compare tag labels is the difference between distance functions and similarity functions. Distance functions associate a real number d to a pair of strings (s_1,s_2), where the smaller the value of d, the closer the strings. Similarity functions associate a real number σ to a pair of strings (s_1,s_2), where the greater the value of σ, the closer the strings. In the SimMetrics package, all measures are implemented so that they can be considered as similarity metrics, even though they can make use of distances, like edit distances, to compute a similarity. The similarity metrics of this package 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, such as *e.g.* Levenshtein, or Gotho; (b) token-based methods, which decompose strings into sets of substrings, *i.e* in our case tokens separated by white spaces, such as Overlap Coefficient or Monge-Elkan ; (c) token-based methods using vector representations of strings such as the cosine similarity; and finally (d) other types of metrics such as QGram or Soundex metrics that compare different features of strings (Soundex *e.g.* associates an arbitrary code to letters composing a string so that string that sound similar have the same code, as *e.g.* "robert" and "rupert").

[9] http://www.dcs.shef.ac.uk/~sam/stringmetrics.html

tag1	tag2	Lev.	Got.	ME
informatique	information	0.75	0.82	0.81
commerce	e-commerce	0.8	1.0	1.0
blog	blogs	0.8	1.0	1.0
Climat/changement	changement climatique	0.14	0.59	0.88
écologie	ecology	0.62	0.71	0.71
ecologie	ecology	0.75	0.86	0.86
developpement-durable	developpement_durable	0.95	0.92	0.92
pollution	pollution des sols	0.5	1.0	0.81
energie	energies	0.63	0.83	0.83
ville	veille	0.83	0.8	0.8
parution	apparition	0.7	0.95	0.95

Table 3.2: Similarity of a set of pairs of tags computed using different metrics from SimMetrics package. Lev. correspond to the Levenhstein metric, Got. to the Gotho metric, and ME to the Monge-Elkan metric.

A simple way to detect equivalent tags using these distance metrics, consists in choosing a threshold value above which two tags are considered equivalent. To illustrate this scenario with real examples, we show in table 3.2 and in figure 3.3 the similarity values of a set of pairs of tags from our application domain at Ademe for three metrics of the SimMetrics package.

3.3.2 Measuring the similarity between tags

Cattuto *et al.* (2008), and later Markines *et al.* (2009), proposed different ways of measuring the similarity between tags and resources in a folksonomy. These approaches can be seen as a generalization of several previous approaches (Mika, 2005; Specia & Motta, 2007) to computing tag similarities. The computation of similarity of tags is often the first step to further process the folksonomy data and to infer semantic relationships between tags (see 3.3.3), or to cluster similar tags (see 3.3.4). In this section we present first a simple method based on co-occurrence count, and then give some details on a method based on an adapted version of the PageRank algorithm proposed by Hotho *et al.* (2006). The next two subsections will present the two main steps of the methods exploiting ternary associations of folksonomies to compute a tag similarity, namely: (1) the aggregation of three-mode tagging data into two-mode data on which (2) several similarity measures can then be applied.

3.3.2.1 Simple co-occurrence counting

The simplest approach consists in counting the coocurrence of tags on a post. Given a folksonomy $F(U, T, R, Y)$ (see section 3.2.2) and given a post $p = (u, T_{ur}, r)$, that is, a subset of the folksonomy corresponding to an annotation of

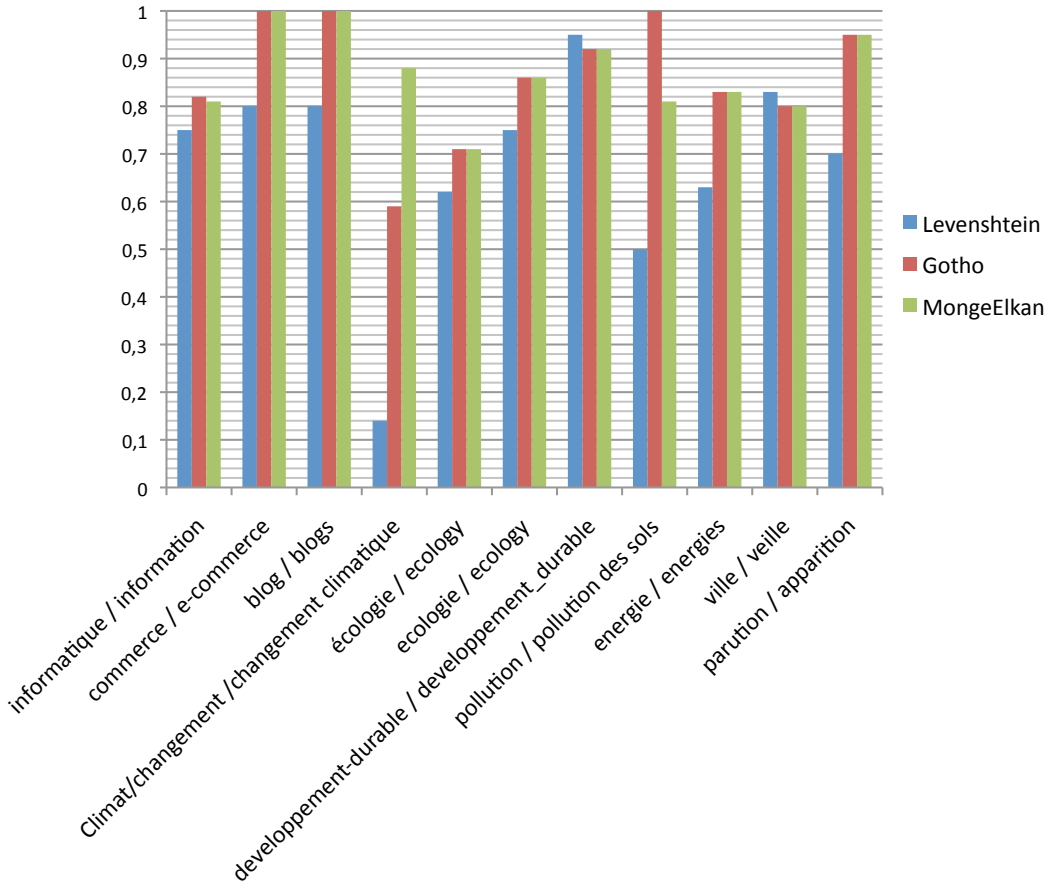


Figure 3.3: Similarity values for a set of pair of tags and for three different metrics from SimMetrics.

a user u of a resource r with a set of tags T_{ur}. The similarity measure given by the simple co-occurrence method counts, for a couple of tags t_1 and t_2, belonging to the folksonomy F with $t_1 \neq t_2$, the number of posts that contain both t_1 and t_2. The complexity of this method is estimated by Cattuto *et al.* (2008) as $O(\frac{|Y|^2}{|P|}\log(\frac{|Y|^2}{|P|}) + |T|^2\log(|T|^2))$, with Y, P, and T the set of ternary relations (as defined in section 3.2.2), the set of posts, and the set of tags.

3.3.2.2 FolkRank based measure of similarity

Hotho *et al.* (2006)developed the FolkRank algorithm, which is an adapted version of the PageRank algorithm used for ranking query results and associating a weight to the folksonomy elements (tags, users or resources). Following the main idea of the PageRank algorithm (Brin & Page, 1998), the idea behind the FolkRank algorithm is that a resource tagged by important users with important tags becomes important itself. The same type of relationships being, conversely, true for tags and users, the aim of the FolkRank algorithm in our case is to compute a ranked list of "relevant" tags for a given tag, the most relevant being the most closely related.

The weight spreading computation of the PageRank algorithm cannot be applied directly to the folksonomy since it is a hypergraph (see section 3.2.2). Thus, the first step is to convert the folksonomy into an undirected graph G_F, where the vertices V consist of the disjoint union of the sets of tags, users and resources so that $V = U \oplus T \oplus R.$, and the edges correspond to all the co-occurrences between the users, tags, or resources (for instance, an edge is drawn between the node corresponding to a user and all the tags he has used at least once). Hotho *et al.* (2006) then apply the weight propagation mechanisms between all the nodes of this undirected graph in order to compute the weight factor $R(v)$ of all the nodes v of the folksonomy graph such that:

$$R \leftarrow c(\alpha R + \beta AR + \gamma P)$$

Where A corresponds to the adjacency matrix of G_F, P is a preference vector where the elements of G_F are given a specific weight, α, β, and γ are constants, and c is a normalization factor such that $\|R\| = 1$. α is a damping factor used to avoid oscillation and speed up convergence, while β and γ control the influence of the preference vector P.

In the case of the computation of related tags for a given tag t, belonging to the set of tags T of the folksonomy $F(U, T, R, Y)$, Cattuto *et al.* (2008) applied the above weight propagation with a high weight for t in the preference vector P and computed the vector R_t for all the other tags. Then, the resulting vector is compared to the case where the weight propagation computation is performed without a preference vector P (which corresponds to the case where $\gamma = 0$). Like this, one computes the winners (and losers) that arise when giving preference to a specific tag in the preference vector P. The tags that, for a given tag t, obtain the highest

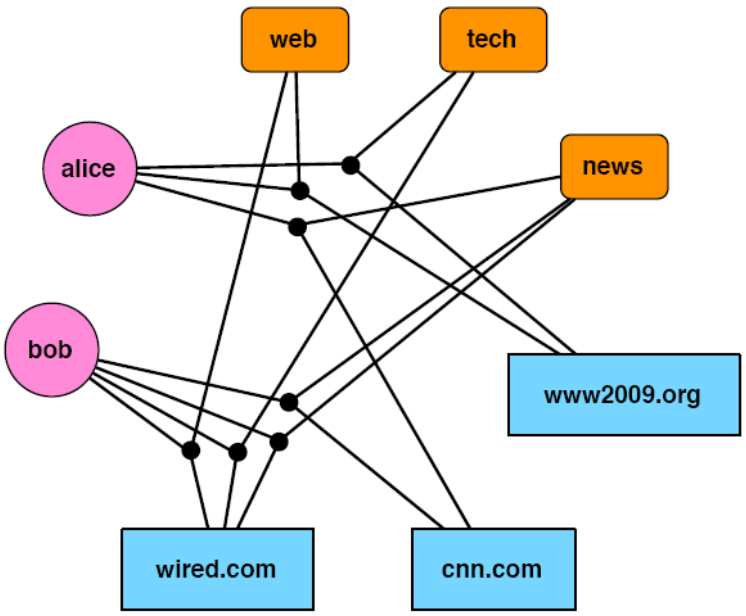


Figure 3.4: Example folksonomy proposed by Markines *et al.* (2009). "Two users (alice and bob) annotate three resources (cnn.com, www2009.org, wired.com) using three tags (news, web, tech). The triples (u; r; t) are represented as hyper-edges connecting a user, a resource and a tag. The 7 triples correspond to the following 4 posts: (alice, cnn.com, {news}), (alice, www2009.org, {web, tech}), (bob, cnn.com, {news}), (bob, wired.com, {news, web, tech})."

weight are considered to be the most related to t. This measure has a complexity of $O(i\,|Y|)$ with i the number of iterations (a typical value is 30 *e.g.*) and Y the set of ternary relations (see section 3.2.2). An example of the results obtained with this method and other similarity metrics is given in figure 3.6.

3.3.2.3 Aggregations of tagging data

The first step before measuring the similarities from the analysis of the tri-partite structure of folksonomies is to aggregate this three-mode view of folksonomies onto two-mode views. For instance, in figure 3.4 we see an example of a small folksonomy where two users annotate three resources with three tags. Each link in a folksonomy is made of three parts : one user associates one tag to one resource. The idea is to project these tri-partite links of folksonomy into bi-partite representations by aggregating the data according to a given context. If we want to look at similarities between tags, there are three such contexts:

- the Tag-Tag context, where we consider the co-occurrence of tags on posts,
- the Tag-Resource context, where we consider the associations of tags via the resources on which they are used,

- the Tag-User context, where we consider the associations of tags via the users who use them.

In the following we are going to review the different methods of aggregation used by the approaches that sought to measure tag similarity, namely the projection aggregation first used by Mika (2005), distributional aggregation introduced by Cattuto *et al.* (2008), and the macro and collaborative aggregation later proposed by Markines *et al.* (2009).

Projection aggregation.

This type of aggregation was first investigated byMika (2005) and consists in projecting the tripartite hypergraph of a folksonomy onto different kinds of two-modes graph corresponding each to the contexts described above. The hypergraph of a folksonomy is given by $H(F) = \langle V, E \rangle$, with the set of vertices $V = U \cup T \cup R$ and the set of edges $E = u, t, r | (u, t, r) \in F$, and where R is the set of resources, U the set of users, and T the set of tags.

Mika (2005) focused in his study on the Tag-Resource and on the Tag-User contexts, but more generally, the projection aggregation consist in building the co-affiliation matrix of the tags with one of the other elements of the context we consider. So, in the Tag-Tag context, this co-affiliation corresponds to the co-occurrence of tags. This matrix will be made of $card(T)$ lines and $card(T)$ columns, and each cell (i, j), with $i\varepsilon[0, card(T)]$ and $j\varepsilon[0, card(T)]$, has a value of 1 if the tag t_i co-occurs at least once with tag t_j, and a value of 0 in the contrary. Likewise, the co-affiliation matrix in the Tag-Resource context represents the affiliation between each tag and each resource, and between each tag and each user in the Tag-User context. In table 3.3 we give the example of the projection aggregation in the Tag-Resource context for the example folksonomy given in figure 3.4. The matrix given by this table corresponds to the co-affiliation matrix of the tri-partite folksonomy hypergraph in the Tag-Resource context. Following Cattuto *et al.* (2008) and the formal definition of a folksonomy (given in 3.2.2), the complexity in the Tag-Tag context comes from the creation of the list of tags that co-occur and is given by $O(\frac{|Y|^2}{|P|} \log(\frac{|Y|^2}{|P|}))$; in the Tag-Resource and Tag-User context the complexity comes from scanning the list of ternary relations and is given by $O(|Y| \log(|Y|))$.

	cnn.com	www2009.org	wired.com
news	1	0	1
web	0	1	1
tech	0	1	1

Table 3.3: Example of a projection aggregation in the Tag-Resource context corresponding to the folksonomy example of Markines *et al.* (2009) given in figure 3.4.

Then, Mika extracts from the Tag-Resource projection a weighted one-mode graph connecting tags based on resource associations, and from the Tag-User pro-

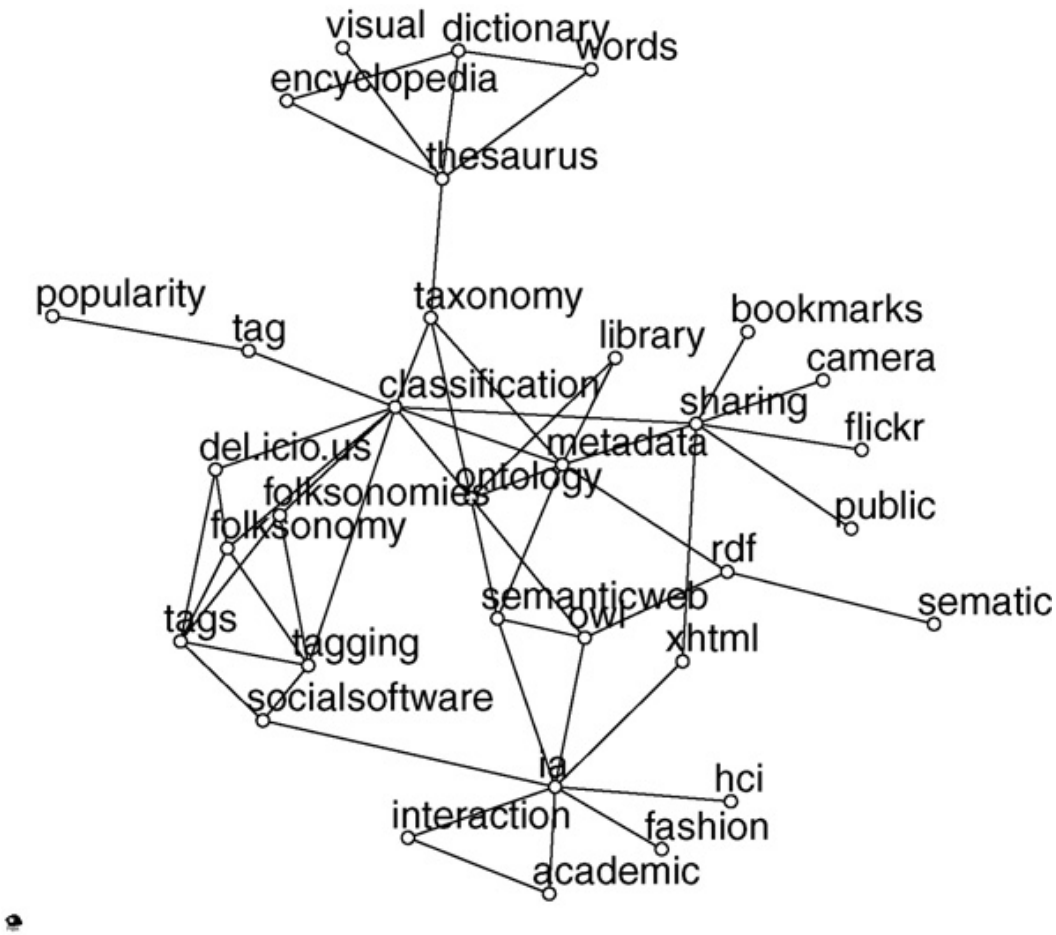


Figure 3.5: del.icio.us tags linked thanks to a projection of the folksonomy based on the Tag-user context (Mika, 2005)

jection a one-mode weighted graph connecting tags based on user associations. In the case of the user-based association of tags, for a given pair of tags, the weights of the graph are given by the number of users who used both tags and normalized by the number of users, thus corresponding to the Jaccard similarity between the tags (the detail of this similarity measure is given below in section 3.3.2.4). Figure 3.5 shows an example of the latter type of graphs that link tags together. This graph was built from an excerpt of delicious.com tags linking two tags when the weight of the link between these tags was above an arbitrary threshold.

Distributional aggregation

More elaborate methods of aggregating tagging data make use of the distributional hypothesis that states that words used in similar context tend to be semantically related (Firth, 1957). This hypothesis was exploited for instance by Cimiano (2006) for ontology learning from texts purposes. This way of aggregating tagging data

considers each element of the context of aggregation as a dimension for a vector representation of the other elements. For instance if we consider the vector representation of tags in the Tag-Resource context, each dimension will correspond to one of the resources of the folksonomy. Thus, distributional aggregation consists in computing the components of the vectors v_t representing each tag t and given for each context by:

- **Tag-Tag Context** : each entry of the tag vector v_t corresponds to the co-occurrence of the tag t with all the other tags, except for the tag t with itself where a weight of 0is given. This is to avoid to consider two tags related when they merely occur together, but rather when they have similar patterns of co-occurrence, that is, when they co occur with the same other tags.
- **Tag-Resource Context** . For a tag t, the vector v_t is constructed by counting how often a tag t is used to annotate a certain resource r.
- **Tag-User Context .** For a tag t, the vector v_t is constructed by counting how often a tag t is used by a certain user u.

If we pick the Tag-Resource context, the matrix representation corresponding to the distributional aggregation for the example folksonomy given in figure 3.4 will look like what we give in table 3.4. For example, the vector of the tag "news" in the Tag-Resource context will be $v_{news} = (2, 0, 1)$. The algorithmic complexity of this aggregation in the Tag-Tag context is given by $O(\frac{|Y|^2}{|P|}\log(\frac{|Y|^2}{|P|}) + |T|^2\log(|T|^2) + |T|^2 2\,|T|)$, in the Tag-Resource context by $O(|Y|\log(|Y|) + |T|^2 2\,|R|)$, and in the Tag-User context by $O(|Y|\log(|Y|) + |T|^2 2\,|U|)$ (Cattuto *et al.*, 2008).

	cnn.com	www2009.org	wired.com
news	2	0	1
web	0	1	1
tech	0	1	1

Table 3.4: Example of a distributional aggregation in the tag-resource context of the folksonomy example of Markines *et al.* (2009).

Macro and collaborative aggregations

The projection and distributional aggregations are considered by Markines *et al.* (2009) as non-incremental, since the whole similarity matrix has to be recalculated after each user add a new annotation. Thus, in cases of web-scale folksonomies, these types of aggregation may not be scalable, since their computation time does not grow constantly with the growth of the folksonomy.

To overcome this limitation, Markines *et al.* (2009) proposed another type of aggregation, called "macro-aggregation" (in contrast with the distributional measures which can be seen as "micro-aggregations") which consists in (1) considering and computing the aggregation and corresponding similarity of each user

separately, and then (2) aggregate across users, that is, to sum the local similarity calculated for each user's data set. Like this, when a user u provides a new annotation, it is not necessary to recompute the similarity for the whole folksonomy but only for this user.

In addition, and in order to take into account the similarity of two resources tagged by the same users but with no tags in common, Markines *et al.* (2009) proposed another way of calculating local similarities, called "collaborative aggregation". The objective of the collaborative aggregation method is achieved by adding a special "user tag" (respectively "user resource") to all resources (respectively tags) of user u. Let us take the example of the tags "news" and "web" for the user "alice" taken from the folksonomy of figure 3.4. If we add the virtual resource "alice_R" to the binary matrix representing alice's tagging (see table 3.5) , we will have a non-zero local similarity between the tags "news" and "web" for the user "alice" since these two tags "co occur" on the virtual resource "alice_R". Then the similarity measure is calculated as in the case of macro-aggregation by summing local similarities across users.

The complexity in the macro and collaborative aggregations can be considered similar to the complexity of the distributional aggregation since similar computation are performed and merely reported in different data structure. The scalability is however not equivalent as explained above.

	cnn.com	www2009.org	wired	alice_R
news	1	0	0	1
web	0	1	0	1

Table 3.5: Binary matrix representation for the collaborative aggregation method for the tags "news" and "web" for the user "alice". The last column is the "virtual resource" added to account for the fact that "news" and "web" are used by the same user, but without being co-occurrent. (Markines *et al.*, 2009)

3.3.2.4 Similarity measures

Different similarity measures can be applied on the 2-modes data resulting from the four types of aggregation methods we described above (projection, distributional, macro, and collaborative aggregations). Markines *et al.* (2009) has applied and evaluated six types of similarity measures that can be performed: matching similarity, overlap similarity, dice coefficient, jacquard similarity, cosine similarity, and mutual information similarity. The detail of the computation of the first three measures being given in Markines *et al.* (2009), we will briefly introduce here the Jaccard similarity used by Mika (2005) (but only in the projection aggregation case), the cosine similarity used byCattuto *et al.* (2008) (but only in the distributional aggregation case), and the mutual information similarity measure introduced and evaluated by Markines *et al.* (2009) who shown that it outperformed the other measures for most of the aggregation methods mentioned above. The

complexities of these measures, which are to be added to the complexities of the aggregation method chosen, are discussed below in section 3.3.2.5.

Let us take two tags x_1 and x_2, with X_1 and X_2 their vector representations. Each entry of these vectors is written $w_{1,xy}$ and $w_{2,xy}$ (for X_1 and X_2 respectively) where y corresponds to the context of the aggregation. For example, in the Tag-Resource context, y will span over the different resources, each taken as one dimension of the tag x. For projection aggregations, the binary vector X can be seen as a set, and $y \in X$ means $w_{xy} = 1$ and $|X| = \sum_y w_{xy}$. In the distributional aggregation case, each resource element w_{xy} of a vector X corresponds to the value on one of the dimension y. For example, in the Tag-Resource context, w_{xy} correspond to the number of times that the tag x is used on the resource y. Similarly, in the macro and collaborative aggregation, for a single user u, $y \in X^u$ is equivalent to $w_{u,xy} = 1$ and $|X^u| = \sum_y w_{u,xy}$.

Jaccard similarity

The Jaccard similarity is a similarity measure between two vector representations that is written as the following for the projection aggregation :

$$\sigma(x_1, x_2) = \frac{|X_1 \cap X_2|}{|X_1 \cup X_2|}$$

And in the distributional aggregation :

$$\sigma(x_1, x_2) = \frac{\sum_{y \varepsilon \{X_1 \cap X_2\}} \log p(y)}{\sum_{y \varepsilon \{X_1 \cup X_2\}} \log p(y)}$$

where $p(y)$ is given by $N(x,y)/N(y)$ where $N(x,y)$ is in the Tag-Resource (resp. Tag-Tag) context the number of times x is used for resource y (resp. the number of times tag x co-occur with tag y), and $N(y)$ is the total number of resources (resp. the total number of tags).

In the macro and collaborative aggregations, one consider the similarity for each user u, and the expression evolves a little bit:

$$\sigma(x_1, x_2) = \frac{\sum_{y \varepsilon \{X_1^u \cap X_2^u\}} \log p(y \mid u)}{\sum_{y \varepsilon \{X_1^u \cup X_2^u\}} \log p(y \mid u)}$$

where $p(y \mid u)$ is the local value of $p(y)$ for user u, *i.e.* $p(y \mid u) = N(x,y)_u/(N(y)_u + 1)$ where $N(x,y)_u$ is in the Tag-Resource (resp. Tag-Tag) context the number of times x is used for resource y by user u (resp. the number of times tag x co-occur with tag y for user u), and $N(y)$ is the total number of resources annotated by user u (resp. the total number of tags used by user u).

Cosine similarity

The cosine similarity σ between tag x_1 and tag x_2 is given by the value of the cosine distance between the two vector representations X_1 and X_2. For the projection

aggregation method it is written as the following:

$$\sigma(x_1, x_2) = cos(X_1, X_2) = \frac{|X_1 \cap X_2|}{\sqrt{|X_1| \cdot |X_2|}}$$

For the distributional aggregation, it is written :

$$\sigma(x_1, x_2) = \frac{X_1.X_2}{\|X_1\|_2 . \|X_2\|_2}$$

And in the macro and collaborative aggregation method, the computation is based on local values for each user u :

$$\sigma(x_1, x_2) = \frac{|X_1^u \cap X_2^u|}{\sqrt{|X_1^u| \cdot |X_2^u|}}$$

Mutual information similarity measure

Markines *et al.* (2009) proposed a new measure of similarity called mutual information. The mutual information similarity $\sigma(x_1, x_2)$ of two tags x_1 and x_2 is defined for the projection and distributional aggregation as:

$$\sigma(x_1, x_2) = \sum_{y_1 \in X_1} \sum_{y_2 \in X_2} p(y_1, y_2) \log \frac{p(y_1, y_2)}{p(y_1)p(y_2)}$$

where, in the projection aggregation, $p(y)$ is the fraction of tags annotating resource y, and the joint probabilities $p(y_1, y_2)$ the fraction of tags annotating both resources y_1 and y_2 given by $p(y_1, y_2) = \frac{\sum_x w_{xy_1} w_{xy_2}}{\sum_x 1}$ where $\sum_x w_{xy_1} w_{xy_2}$ counts the number of tags that annotate both y_1 and y_2, and $\sum_x 1$ correspond to the total number of tags. In distributional aggregation the normalization for $p(y)$ and $p(y_1, y_2)$ is done across the whole matrix rather than across the columns, and we have $p(y) = \frac{\sum_x w_{xy}}{\sum_{r,t} w_{rt}}$ and p(y_1,y_2)=$\frac{\sum_x min(w_{xy_1}, w_{xy_2})}{\sum_{r,t} w_{rt}}$ with $\sum_{r,t} w_{rt}$ the sum of all the entries of the matrix.

In the case of macro and collaborative aggregation, the local mutual information similarity for a user u is given by:

$$\sigma_u(x_1, x_2) = \sum_{y_1 \in X_1^u} \sum_{y_2 \in X_2^u} p(y_1, y_2 \mid u) \log \frac{p(y_1, y_2|u)}{p(y_1|u)p(y_2|u)}$$

where the local simple probabilities $p(y|u)$ are given by $p(y|u) = N(u, y)/(N(u) + 1)$ where $N(u, y)$ is the number of tags used by u to annotate resource y, while $N(u)$ is the total number of tags of u. The joint probabilities are normalized similarly to the projection aggregation, but in this case for the binary representation of each user. The global value of this similarity is obtained by summing across all users these local similarities.

rank	tag	measure	1	2	3	4	5
13	web2.0	*co-occurrence*	ajax	web	tools	blog	webdesign
		folkrank	web	ajax	tools	design	blog
		tag context	web2	web-2.0	webapp	"web	web_2.0
		resource context	web2	web20	2.0	web_2.0	web-2.0
		user context	ajax	aggregator	rss	google	collaboration
15	howto	*co-occurrence*	tutorial	reference	tips	linux	programming
		folkrank	reference	linux	tutorial	programming	software
		tag context	how-to	guide	tutorials	help	how_to
		resource context	how-to	tutorial	tutorials	tips	diy
		user context	reference	tutorial	tips	hacks	tools
28	games	*co-occurrence*	fun	flash	game	free	software
		folkrank	game	fun	flash	software	programming
		tag context	game	timewaster	spiel	jeu	bored
		resource context	game	gaming	juegos	videogames	fun
		user context	video	reference	fun	books	science
30	java	*co-occurrence*	programming	development	opensource	software	web
		folkrank	programming	development	software	ajax	web
		tag context	python	perl	code	c++	delphi
		resource context	j2ee	j2se	javadoc	development	programming
		user context	eclipse	j2ee	junit	spring	xml
39	opensource	*co-occurrence*	software	linux	programming	tools	free
		folkrank	software	linux	programming	tools	web
		tag context	open_source	open-source	open.source	oss	foss
		resource context	open-source	open	open_source	oss	software
		user context	programming	linux	framework	ajax	windows
1152	tobuy	*co-occurrence*	shopping	books	book	design	toread
		folkrank	toread	shopping	design	books	music
		tag context	wishlist	to_buy	buyme	wish-list	iwant
		resource context	wishlist	shopping	clothing	tshirts	t-shirts
		user context	toread	cdm	todownload	todo	magnet

Figure 3.6: Examples of most related tags for different measures and different contexts of aggregations (Cattuto *et al.*, 2008)

3.3.2.5 Evaluation and results of tag similarity measures

Example results

Figure 3.6 provides examples given by Cattuto *et al.* (2008) of most related tags using some of the similarity measures explained above. For each tag, the following similarity are computed (from top to bottom): *co-occurrence* corresponds to the simple count of the most co-occurring tags; *folkrank* corresponds to the similarity based on the folkRank algorithm; then the last three measures make use of cosine similarity computed in different contexts of distributional aggregation as explained above, namely the Tag-Tag context for *tag context*, Tag-Resource context for *resource context*, and Tag-User context for *user context*.

Comparing the different aggregation methods for different similarity measures

Markines *et al.* (2009) conducted an evaluation aimed at comparing the performances in terms of accuracy of the similarity measures presented above for each of the four types of aggregations. This evaluation was led on the dataset of Bib-

sonomy.org[10], a social bookmarking service devoted to the annotation of academic works and in which users can define semantic relations between tags. The measure they compare for tag similarity are computed in the Tag-Resource context.

A first evaluation directly compared computed similarity with user-provided relations in Bibsonomy.org by using different threshold values above which a user-provided similarity relation is predicted by the computed similarity. Each similarity measure is thus evaluated by calculating the number of good predictions (true positive) for different values of the threshold. The result of the evaluation showed that mutual information outperforms the other types of similarity measures for the case of distributional aggregation, whereas for collaborative aggregation, none of the measures compared gave significantly better results. However, the number of relations provided by the users is scarce compared to the number of tags in bibsonomy.org (142 relations for 2000 tags), and the choice of a threshold is problematic due to the great order of magnitude of the values of computed similarities hence the low confidence in this first evaluation.

To overcome these limitations, Markines *et al.* (2009) chose to use Wordnet as a reference to evaluate the accuracy of the computed similarities. To avoid the problems of the choice of a threshold, they compare the ranking of the most similar pairs of tags according to computed similarities and according to a WordNet based similarity which computes the Jiang-Conrath distance (Jiang & Conrath, 1997) between the same tags present in the WordNet dataset. The level of agreement between the computed similarity and the WordNet reference is calculated by the Kendall's τ correlation as implemented by Boldi *et al.* (2004). The results of this second evaluation are shown in figure 3.7. The mutual information similarity is the best measure, outperforming clearly the other similarities in all aggregation methods except for the collaborative one where it comes second after matching similarity. However, the mutual information similarity (see section 3.3.2.4) is the most costly due to its quadratic complexity, whereas the cosine measure has linear complexity. Interestingly, the collaborative aggregation leads better results than other aggregation, except for mutual information that performs best in projection and distributional aggregation. This shows the positive influence on the quality of inferred semantics of looking first at individual users tagging data and the corresponding similarity, and then aggregating these individual-based similarities across all users (we will see below another study (Koerner *et al.*, 2010) that examined the influence of the choice of subset of users on the quality of inferred semantics). Another advantage of collaborative aggregation is its scalability regarding the growth of the folksonomy since, in this case, one only needs to update the similarity computed for the user who adds new annotations before summing it to the other users similarities. However, we should remark that the data structure gains in complexity in this case, since one needs to maintain a separate table for each user, and this seems to be the price in terms of memory management to pay in exchange of the time saved with this method of aggregation. Finally, we should

[10]http://www.bibsonomy.org/faq#faq-dataset-1

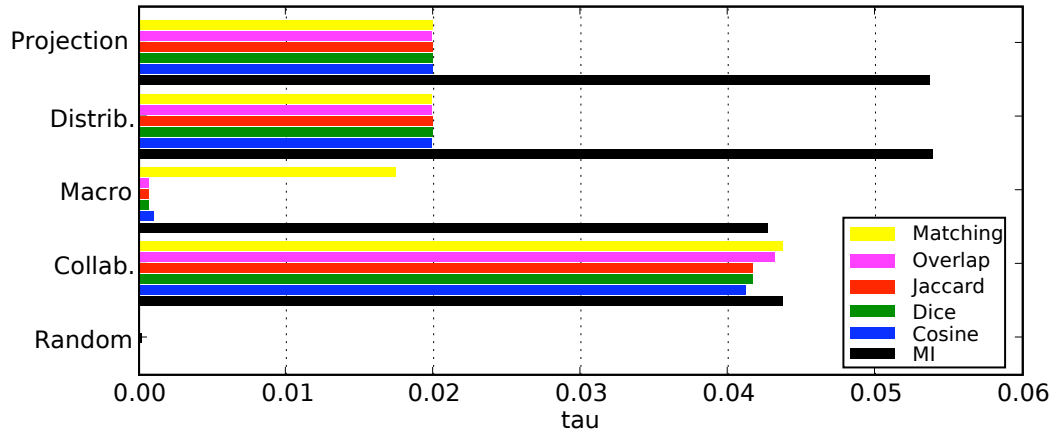


Figure 3.7: Performance in terms of accuracy of several tag similarity measures (from top to bottom : Matching, Overlap, Jaccard, Dice, Cosine, Mutual Information) computed in the Tag-Resource context for several aggregation methods (from top to bottom : Projection, Distributional, Macro, Collaborative) by Markines *et al.* (2009). The performance is given by the Kendall's τ correlation between ranked set of pairs of similar tags according to computed similarities and according to a WordNet-based similarity. All measures are compared with a random set of similar tags.

note that Markines *et al.* (2009) compared these tag similarity measures only in the Tag-Resource context, whereas Cattuto *et al.* (2008) and latter Koerner *et al.* (2010), many of whom were co-author of Markines *et al.* (2009), praised the cosine similarity computed for the distributional aggregation in the Tag-Tag context which brings good quality semantics for an affordable computational cost.

Grounding the relatedness of tags using a generic hierarchy of concepts (Wordnet)

Cattuto *et al.* (2008) have proposed a method to semantically ground the relatedness between two tags in order to better interpret the type of semantic relation that these measures bring. To do so, for each tag they (1) use different types of measures, as defined above, to collect similar tags; then (2) they map these tags into Wordnet (Fellbaum, 1998) synsets; and (3) they measure the distance in the Wordnet hierarchy between these terms using Jiang-Conrath distance (Jiang & Conrath, 1997).

In the example depicted in figure 3.8 (sample data extracted from the 10 000 most frequent tags of del.icio.us), the original tag is "java". If we look at the table shown in figure 3.6, according to the simple co-occurrence measure ("freq" in the figure) and the FolkRank measure, the most related tag to "java" is "programming", and according to the distributional cosine measures computed in the Tag-Tag context, the most related tag is "python". Then, when we look at an excerpt of the Wordnet synset hierarchy containing the original tag and its related tags (see

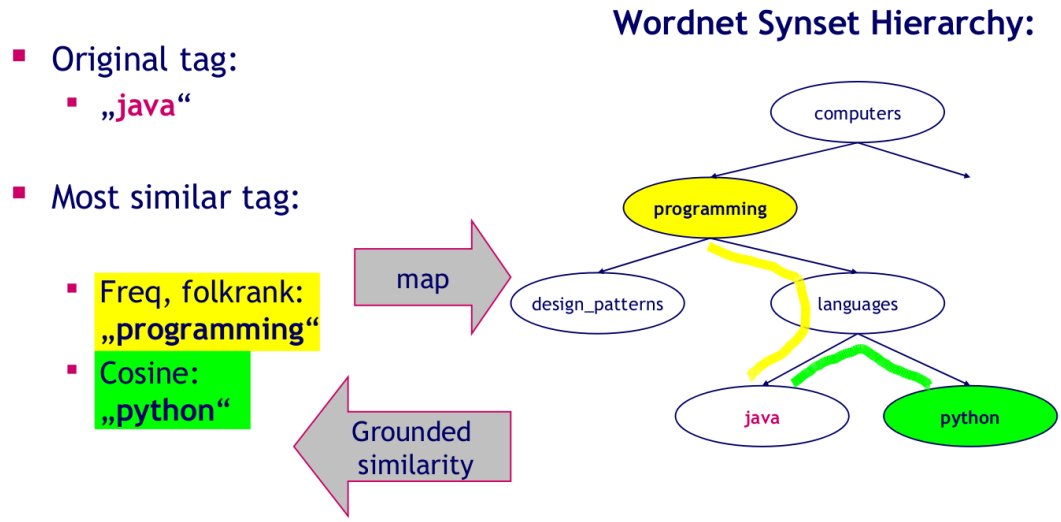


Figure 3.8: Semantic grounding of the relatedness of tags using Wordnet (Cattuto *et al.*, 2008)

figure 3.8), we observe (1) that tags given by the co-occurrence and the FolkRank measure corresponds to concepts higher in the hierarchy, and (2) that tags given by distributional measures tend to have the same level in the hierarchy. Cattuto *et al.* (2008) repeated this experiment for all of the delicious.com tags which were present in Wordnet, and they draw some qualitative remarks about the semantic relationships each type of measure brings:

- similarity measure in the Tag-Tag context and in the Tag-Resource context tend to give siblings in some suitable concept hierarchy, i.e. tags that can be considered *related* in thesauri terms (as “java” and “python”), or to give synonyms (such as “java” and “jee”) or spelling variants (such as “opensource” and “open_source”). The Tag-Tag context similarity, however, seems to be the most capable of identifying sibling tags.
- the FolkRank and co-occurrence similarity measures tend to give more general tags, *i.e.* tags that can be considered *broader* in thesauri terms.

Influence of the choice of sub-folksonomies in the quality of emerging semantics

Koerner *et al.* (2010) investigated the factors that may influence the semantics extracted from folksonomies. The main idea is that if the semantic relationships between tags are inferred from the analysis of the tri-partite structure of a folksonomy, then these inferences might change if we pick up a subset of this folksonomy. The author investigate here the case of a subset corresponding to different types of users, but one could also imagine selecting samples corresponding to a particular set of tags or resources.

The similarity measure used here is the cosine distance computed on distributional aggregation in the Tag-Tag context as defined above (section 3.3.2.3 and 3.3.2.4). Then, this similarity measure is compared to the distance given in WordNet between the words corresponding to the considered pair of tags using Jiang-Conrath distance (Jiang & Conrath, 1997). This gives a quality measurement of the inferred semantics.

The criteria taken into account to distinguish the types of user are their way of tagging. Koerner *et al.* (2010) distinguish two main and prototypical types of tagger:

- Categorizers typically use a small controlled set of tags, their goal is an ontology-like categorization, and they see tags as replacement for folders.
- Describers typically use verbose and freely chosen tags, and thus, they do not necessarily take care of maintaining a strict consistency and may introduce spelling variants and synonyms in their tags that they mainly see as describers of content.

Koerner *et al.* (2010) then introduce several measures to distinguish between the two types of tagger. The vocabulary size is given by the total number of tags used by a user, and is typically higher for describers. The tag over resource ratio is given by the total number of tags divided by the total number of resources and is typically higher for describers. The average number of tags per post is also higher for describers. The orphan ratio is given by the number of tags used very rarely divided by the total number of tags and is typically higher for describers since they tend to introduce very often new tags that they won't necessarily reuse afterwards. One should also be aware that these categories of taggers have some limitations and represent extreme cases of usages. These categories are mixed in the real world, and the graphical interfaces of tagging tools have also a great influence on tagging behavior.

The question now is to know whether we obtain better semantics with a subset of describers or with a subset of categorizers. To answer this question Koerner *et al.* (2010) conducted an experiment on a dataset extracted from delicious.com dating from 2006 until now. They kept from this sample the top 10000 tags and kept users sharing at least 100 posts. Then they computed the metrics on this set of users to detect the categorizers and describers, and created different subsets of the original folksonomy by incrementally adding users ordered either from the most extreme categorizers to the most extreme describers or in the reverse direction. They then computed the similarity between tags with the Tag Context Similarity measure on each subset and assessed the quality of inferred semantics with Wordnet-based distances. When incrementally adding users starting with the most extreme categorizers, and with a ratio below 60% of the total number of users, the corresponding subfolksonomies performed worse than random sampling. On the contrary, subfolksonomies made by describers outperformed random sampling all the way through. Moreover, 40% of the describers are sufficient to beat the full dataset

folksonomy, whereas more than 60% of categorizers are necessary to achieve similar performance. This means that, subset made mostly of describers yield better quality semantics than subset made mostly of categorizers. The explanation of this is that describer, as they use more tags per post, favor co-occurrence of tags, and thus maximize the chances of leading to a relation between tags according to the tag context similarity measure. However, the most extreme describers have detrimental effect too, as they correspond most probably to spammers.

3.3.3 Inferring subsumption relations

Several approaches have been proposed to directly infer subsumption relationships between tags from the analysis of the tri-partite structure of folksonomies.

3.3.3.1 Exploiting similarity graphs

The algorithm proposed by Heymann & Garcia-Molina (2006) takes as input the list of tags in descending order of their centrality in the similarity graph computed with cosine similarity in the Tag-Resource context. The hierarchy of tags is built starting from the root node, and each tag, taken in order of centrality, is added either as a child of one of the nodes or the root node (depending on a threshold value of its similarity with these nodes).

This algorithm has been lately extended by Benz *et al.* (2010) with synonym identification and a mechanism to disambiguate tags. Synonym tags are detected using a cosine similarity measure computed in distributional aggregation in the Tag-Tag context, following the results of Cattuto *et al.* (2008) who showed that this type of similarity measure yielded synonyms or related terms in the thesaurus sense. To retrieve mostly synonyms, the authors used an experimentally chosen threshold. To detect the different possible meanings of a tag, they clustered its ten most co-occurring tags, taken as the context of each tag. These clusters grouped together tags that were very similar exploiting the same similarity measure with a different threshold. Tags with several clusters in their context-tags are set to be ambiguous, and a *preference-tag* is chosen among the tags of each cluster. Benz *et al.* (2010) compared the inferred hierarchy with a reference ontology derived from a combination of WordNet and Wikipedia, and they showed that these extensions of the algorithm of Heymann & Garcia-Molina (2006) provided better results and reflected better the diversity of knowledge contained in folksonomies.

3.3.3.2 Association rule mining

Several approaches proposed different methods which can be seen as association rule mining, similarly to Schmitz *et al.* (2006), the idea being to exploit some properties of the structure of folksonomies, such as overlap and inclusion of some sets, to infer subsumption between tags.

Mika (2005) grouped similar communities of interest (as described above in section 3.3.2.3) to derive subsumption properties between the tags thanks to the

inclusion of communities of interest. In this case, a community of interest may be represented by all the actors who used the tag “fishing”. If the communities of interest “fishing” and “nautic activities” have a number of actors in common, the tags “fishing” and “nautic activities" will be considered as semantically related. Furthermore, if the group of actors using the tag “fishing” is a subset of the group of actors using “nautic activities”, “nautic activities” will be set as a broader term than “fishing”. Mika also shows, thanks to a qualitative analysis lead by asking experts to evaluate these inferences, that subsumption relations are more relevant when derived from inclusions of communities of interest, than when similarly derived from the association of tags via their common use on resources.

Schmitz (2006) used conditional probability to detect subsumption relationships between tags. Given a tag pair (Ti, Tj), let us call the frequency of occurrence of each tag $N(Ti)$ and $N(Tj)$, and the frequency of co-occurrence of both tags $N(Ti \cap Tj)$. The conditional probability $P(Ti|Tj)$ of having Ti given Tj is calculated as follows:

$$P(Ti|Tj) = \frac{N(Ti \cap Tj)}{N(Tj)}$$

And conversely

$$P(Tj|Ti) = \frac{N(Ti \cap Tj)}{N(Ti)}$$

By comparing both values with each other, we can deduce which of the tags of the pair is more dependent on the other tag. In order to induce a hierarchy from flickr.com tags, Schmitz (2006) have adapted the method proposed by Sanderson & Croft (1999), integrating new statistical thresholds to account for the specificity of folksonomies. Thus, tag Ti potentially subsumes tag Tj if :

$$P(Ti|Tj) \geq t \; and \; P(Tj|Ti) < t$$

with

$$N(Ti) \geq Tmin,\; N(Tj) \geq Tmin,\; U(Ti) \geq Umin,\; U(Tj) \geq Umin$$

Where t is a given co-occurrence threshold, $N(Ti)$ and $N(Tj)$ are greater than a minimum value $Tmin$, and $U(Ti)$ is the number of users who use tag Ti at least once with $U(Ti)$ greater than a minimum value $Umin$.

Schwarzkopf *et al.* (2007) also proposed building taxonomies out of folksonomies for user profiling purposes. They pointed out the limitations of the algorithm proposed by Heymann & Garcia-Molina (2006), noting that the cosine similarity measure used in this algorithm does not take into account the popularity of tags. Indeed Mika (2005) reported that relationships between tags established via users are more suitable to infer narrower/broader relationships since they take into account the popularity of tags, because a tag can subsume another tag only if it is more often used. Schwarzkopf *et al.* (2007) also remarked that, more generally, association rules mining proposed by Schmitz *et al.* (2006), of which Mika’s

approach is a particular case, allow to infer taxonomic relations that are more representative of the communities knowledge than mere cosine similarity measures. Schwarzkopf *et al.* (2007) thus used association rules mining methods in order to infer subsumption relationships between tags, such that: "If resources tagged with t_0 are often also tagged with t_1 but a large number of resources tagged with t_1 are not tagged with t_0, t_1 can be considered to subsume t_0". Schwarzkopf *et al.* (2007) also addressed the transitivity problem of the subsumption relations inferred with this method and noticed that the "similarity context" of tags is not taken into account when adding a child-tag to a parent-tag, such as in `design > web > howto > productivity > business`, where each link makes sense but the whole chain does not. Thus they combine the association rules mining technique and a cosine-based similarity measure, so that a child-tag is added to a branch only if its similarity with all the other tags of that branch is above a given threshold.

3.3.3.3 Clustering-based approaches

Zhou *et al.* (2007) proposed an approach to learning hierarchies from folksonomies based on clustering algorithms. The clustering is applied on the set of tags, knowing for each tag on which resource it has been used, thus aggregating tagging data in the Tag-Resource context. They propose an adapted version of the Deterministic Annealing Algorithm of Rose (1998) to cluster tags until all clusters are *effective* clusters, or a maximum number of clusters is reached. An effective cluster is detected when one tag of this cluster can be chosen as a leading tag. A leading tag is a tag that has a maximum coverage, *i.e.*, that maximizes the number of resources on which it co-occurs with each other tag of the cluster. The coverage of each tag is computed at each clustering steps, and a tag that has the maximum coverage for a given cluster is detected as a leading tag only if its coverage reaches a threshold value. If not, the clustering process is repeated on this cluster. When the clustering process is over, each leading tag is considered as a local root node, and the remaining tags of the cluster as the child of that node. This method has been applied on a sample of delicious.com and flickr.com and the authors remarked that the hierarchical relations detected by this algorithm actually mix different types of semantic relations, namely subsumption ("videogame" is a subnode of "game"), related ("travel" is a subnode of "hotel"), and sibling ("WMA" is a subnode of "DVD").

3.3.4 Clustering tags

Clustering tags may be useful to help taggers group tags into bundles of related tags that can be used in further stages of ontology building or ontology maturing. Here we briefly describe different ways of clustering similar tags.

First, we can mention the work of Bothorel & Bouklit (2008) who proposed to apply a clustering algorithm on a bipartite hypergraph of tag co-occurrence in order to detect community structures in the use of tags in folksonomies. This type

of hypergraph connects tags with sets of tags coocurring frequently on a given resource. They have then adapted the algorithm of Newman & Girvan (2004) to detect community structures in the case of hypergraphs. The authors applied this method on a sample of Flickr's dataset and the result consists in clusters corresponding each to a sub-hypergraph linking tags around a tag with the highest centrality for each cluster. Tags with the highest centrality are tags used by many different users and in different contexts and, thus, evidence the emergence of consensuses in the folksonomy precisely around them. This study thus proposes a way to cluster closely related tags and to identify, for each cluster, the tag that reached a maximum consensus. However, the complexity of this type of computations on hypergraph structures call for further improvements according to the authors.

Specia & Motta (2007) applied clustering techniques to group tags according to the similarity measure within the Tag-Tag context (according to the terminology of Cattuto *et al.*, 2008). During the computation, each cluster starts with a seed tag, and a tag is added only if it has a similarity value above a given threshold with all the other tags of the cluster. Then they apply different heuristic techniques to merge very similar clusters based, for instance, on the percentage of equivalent tags contained in similar clusters. These clusters are then used to enhance the mapping between tags and ontology concepts (see section 3.4.1).

Begelman *et al.* (2006) proposed a method for avoiding the use of arbitrary threshold. They first establish a method to determine strongly related tags based on a co-occurrence count on the tagged resources and not on posts. Then they calculate the cut-off frequency of occurrence between two tags by looking for a disruption point in the distribution, for each tag, of all the tags co-occurring with it. This method allows to dynamically find, for each tag, the threshold above which its co-occurring tags are strongly related to it. Then they draw a weighted graph connecting these related tags together. The clustering algorithm takes as input this graph and (1) uses spectral bisection (Pothen *et al.*, 1990) to split the graph into two clusters, (2) compares the value of the modularity function[11] Q_0 of the original unpartitioned graph to the value of the modularity function Q_1 of the partitioned graph. If $Q_1 > Q_0$ it accepts the partitioning, otherwise it rejects the partitioning. It then (3) proceeds recursively on each accepted partition. The clustering of tags is used by Begelman *et al.* (2006) to improve search in the tag space by suggesting groups of strongly related tags instead of flat lists of related tags, each group identifying one particular notion, serving also, when needed, disambiguation purposes. Begelman *et al.* (2006)

Giannakidou *et al.* (2008) also proposed an approach to clustering similar tags aimed at enhancing folksonomy navigation. The similarity measure they use couples a measure based on co-occurrence (which they call "social" similarity because it reflects the social usage of tags) with a measure based on the distance between

[11] "which measures the quality of a particular clustering of nodes in a graph"(Newman & Girvan, 2004)

the tags in a hierarchy of concepts such as Wordnet. The similarity between tags they compute is thus made of a "social" component and a semantic one, both having a given proportion (respectively w and $1 - w$) set as a parameter of the computation. The authors propose then a co-clustering, inspired by graphs partitioning approaches (Hagen & Kahng, 1992), which can be applied on the bi-partite graph linking tags and tagged resources. In this regard, a subset of the most frequent tags is chosen as a set of tag-attributes at_j to be later linked to each resource r_i. The weight of this link is given, for a resource r_i and a tag-attribute at_j (one resource can be linked to several tag-attribute), by the maximum similarity computed between each tag used to tag r_i and the tag-attribute at_j. The goal of the co-clustering algorithm is to group the most related resources and their associated tag-attributes, so that two resources are most related when they both have strong link with the same attribute. The result of this approach is a set of clusters containing resources and tags, allowing in this way to identify the most representative tags for a set of resources, and also to identify sets of strongly related tags. Interestingly, they authors get better result when setting w to 0,5 to give the social and the semantic component as much weight in the similarity measure between tags.

3.3.5 Comparison of the approaches and intermediary conclusions

In this section we have presented several approaches that extract semantic relations between tags by analyzing tag labels or the structures of folksonomies, in contrast with other types of methods that use external semantic resources to achieve this task. In section 3.3.2, we presented the main methods to measure the similarity between tags by first aggregating tagging data in 2-mode views and then applying similarity measures. Then these similarity measures or some variants can be used to find subsumption relationships between tags, or to cluster similar tags (see table 3.6). The case of Cattuto *et al.* (2008) is particular in that they characterize different types of similarity measures according to the type of semantic relationships to which they each correspond. Their results show that some methods are better for inferring some specific semantic relations.

In table 3.6 we report the different types of similarity measure proposed by these approaches. Mika (2005) applied and compared different graph projections methods on the tripartite structure of folksonomies. Hotho *et al.* (2006) adapted the PageRank algorithm to the case of folksonomies in order to find not only relationships between tags, but also between users and resources. Schmitz (2006) used conditional probability methods to induce a hierarchy from Flickr tags. Begelman *et al.* (2006) look closely at the distribution of the co-occurring tags for a given tag, and calculated dynamically the threshold above which its co-occurring tags are strongly related to it. Then, several approaches use distributional measures but with different contexts of aggregation of the folksonomy data as explained in section 3.3.2, Heymann & Garcia-Molina (2006) using the Tag-Resource context, Specia & Motta (2007) using the Tag-Tag context of association of tags, Schwarzkopf *et al.* (2007) using a composite measure mixing association rules mining techniques

	Type of similarity	Subsumption rel.	Cluster.
Mika (2005)	graph projection	yes	no
Hotho *et al.* (2006)	FolkRank	no	no
Schmitz (2006)	conditional probability	yes	no
Begelman *et al.* (2006)	co-occurrence	no	yes
Heymann & Garcia-Molina (2006)	distributional (resource context)	yes	no
Specia & Motta (2007)	distributional (tag context)	no	yes
Schwarzkopf *et al.* (2007)	composite	yes	no
Cattuto *et al.* (2008)	distributional (3 contexts)	yes	no
Markines *et al.* (2009)	mutual information	no	no
Giannakidou *et al.* (2008)	composite	no	yes
Zhou *et al.* (2007)	deterministic annealing	yes	yes

Table 3.6: Comparison table of the approaches extracting semantic relations between tags by analyzing the structure of folksonomies

of Schmitz *et al.* (2006) and the cosine similarity measure. Finally Cattuto *et al.* (2008) proposed an analysis of the different context of distributional aggregation, while Markines *et al.* (2009) proposed a new type of similarity measure based on mutual information calculus that performs well in their evaluation but at the cost of an increased complexity. The performance and complexities of the different aggregation methods and the similarity measures are given above, but, to summarize, Cattuto *et al.* (2008) and later Koerner *et al.* (2010) reported that the cosine similarity computed in the distributional aggregation in the Tag-Tag context gave good quality semantics at a reasonable computational cost.

3.4 Semantic enrichment of folksonomies

In this section we present several works that propose to semantically structure folksonomies or to link tags with structured knowledge representations (ontologies, thesauri, etc.). These approaches consider tags either as attributes of the concepts of an termino-ontological resource (additional labels, properties), or as candidates for new concepts to be added. In this regard, tags are similar to "term-candidate" of the approach of Aussenac-Gilles *et al.* (2000a) to build ontologies from texts. Other approaches use ontologies to support the semantic structuring of folksonomies (section 3.4.1), or to map tags with concepts (Good *et al.*, 2007; Tesconi *et al.*, 2008; Passant, 2007) , or to provide a global framework to help interconnect tagging data within the Semantic Web.

3.4.1 Folksonomy enrichment a posteriori using termino-ontological resources

The methods we present below seek to assist the semantic enrichment of folksonomies *a posteriori*, *i.e.* once tagging data is already collected, through the linking of tags with ontologies. Thus, they do not necessarily make use of Semantic Web formats and infrastructure described in section 3.4.3, but focus more on the automatization of the process of semantifying already created tags.

3.4.1.1 Mapping tags with concepts

A simple approach to tag-concepts mapping consist in using string-based metrics to match a tag and a concept with the same label. This approach was used by Gligorov *et al.* (2010) to compare professional vocabularies with tags provided by participants of a *game with a purpose* to tag video contents (see a presentation of this type of approaches in section 3.4.2). Gligorov *et al.* (2010) used GTAA[12], a vocabulary in the domain of Sound and Vision organized as a thesaurus, and Cornetto[13], a lexical database structured like WordNet in synsets and which contain common lexical terms in Dutch language. Gligorov *et al.* (2010) used an exact string based matching to link tags with terms from the professional vocabularies. However, the matching process is ambiguous in the case of the general lexical resources as 45 % of tags can be matched to more than one synset. This ambiguity obviously recalls the inherent ambiguity of tags. Gligorov et al. also considered using stemming algorithms to be able to match misspelled words. However, the use of stemming algorithms requires to know in advance the language of a tag, which is not obvious in an open environment as the Web.

Laniado *et al.* (2007) addressed the fact that one tag can sometimes be mapped to different WordNet synsets, each synset corresponding to one meaning of the tag. To overcome this, Laniado *et al.* (2007) proposed a disambiguation method that considers the context of each tag. The context of a tag consist in the set of the other tags used to annotate the same resources. A semantic similarity based on Wordnet metrics is computed between each word of the synset and each context-tag so that the synset which is the most related to the other context-tags is chosen. This simple approach is thus bound to the specific structure of WordNet synsets, and may not be suited to map tags to regular domain ontologies from the semantic web.

Another approach (Torniai *et al.*, 2008) to tag-concepts mapping when working with domain ontologies not structured in synsets proposed a "context based measure of semantic relatedness" (*CBRM*), which is a measure of the semantic

[12] "This vocabulary, used by the Dutch national public Audiovisual and radio archives for its documentation process, covers a wide range of topics, as it is meant to describe anything that can be broadcasted on TV or radio. It contains approximately 160.000 terms, divided in 6 disjoint facets: Keywords, Locations, Person Names, Organization-Group-Other Names, Maker Names and Genres." http://www.w3.org/2006/07/SWD/wiki/EucGtaaBrowser

[13] http://www2.let.vu.nl/oz/cltl/cornetto/index.html

relatedness between a tag and a concept which takes into account the context of the target concept. The context of a concept is composed of its ascendants or descendants in the ontology hierarchy. The $CBRM$ between a tag T and a concept C is computed as the weighted average of the measures of semantic relatedness $MSR_{(C,T)}$ between the target tag and the set of descendants and descendants of concept C. The Measure of Semantic Relatedness $MSR_{(C,T)}$ between a tag and a concept is based on a similarity measure between words proposed by Cilibrasi & Vitanyi (2006) and exploiting the WikiPedia content. One of the aims of Torniai *et al.* (2008) is to show that taking into account the context of a concept C helps improve the simple MSR measure. Indeed, they conducted an experiment asking experts to evaluate the relatedness of a set of tag-concept pairs computed using the MSR, WMSR, and CBRM methods. The outcome of this experiment shows that CBRM method brings a substantial benefit in comparison with MSR method, especially for concepts from fine-grained ontologies.

Similarly, the TagPedia project proposed by Ronzano *et al.* (2008) and the Tag Disambiguation Algorithm developed by Tesconi *et al.* (2008) aims at connecting tags with unambiguous definition of their meaning taken from WikiPedia pages. Ronzano *et al.* (2008) proposed mining the Wikipedia disambiguation pages to connect tags with a unique definition page representing a concept. The result is a set of "tag" synsets, that is, sets of synonymous terms linked with a concept defined by a Wikipedia article. These tag synsets are then utilized by the Tag Disambiguation Algorithm (TDA) developed by Tesconi *et al.* (2008) to connect each tag of a given delicious.com's user to a unique meaning. To achieve this task, the TDA identifies for each tag t a list of candidate meanings for which it computes a sense-rank SR. Tesconi *et al.* (2008) assume that the meaning given to a tag does not change across all the taggings of a given user u. To calculate the SR of each possible meaning for a tag t, Tesconi *et al.* (2008) exploits the TagPedia synsets, the text of each meaning extracted from the corresponding Wikipedia article, and tagging data given by delicious.com for each bookmark. The relevance of the results of the DTA has been reviewed by humans, and among 2589 polysemous tags, the DTA has chosen the right meaning for 89,15% of them. Once each tag of a user is associated to an unambiguous meaning represented by a WikiPedia article, it is possible to map these tags to semantically rich structures such as YAGO[14], a generic knowledge representation automatically extracted from Wikipedia which uses Wordnet to organize information, or DBpedia[15] (Auer *et al.*, 2007), a publicly available dataset which references each Wikipedia concept with a unique URI and represents the hierarchy of the Wikipedia categories as a thesaurus written in SKOS. The Wikipedia categories structure covers the largest part of the disambiguated tags of a sample of 9 delicious.com users. Thus, if the disambiguated tags are each connected to a DBpedia URI, the method proposed by Tesconi *et al.* allows connecting any user's tag with an unambiguous meaning, identified with a URI and accessible on the

[14]http://www.mpi-inf.mpg.de/suchanek/downloads/yago/
[15]http://wiki.dbpedia.org

Semantic Web, and semantically linked with other concepts from Wikipedia.

3.4.1.2 Integrated approaches

Below we will present integrated approaches that try to combine different flavors of similarity metrics on the folksonomy structure with the use of termino-ontological resources.

A first example of such approaches is proposed byLin *et al.* (2009), who still rely on WordNet but integrate different strategies in order to extract hierarchical structure from folksonomies. They introduce some distinction between tags regarding the mapping problem: standard tags which are to be found in dictionaries such as WordNet (*e.g.*, "semantic", or "web"), compound tags which are non-standard expressions usually mixing standards terms (*e.g.*, "semantic web"), and jargon tags that are terms very specific to a community (*e.g.*, "semweb"). First, Lin *et al.* combine association rule mining (Schmitz *et al.*, 2006) and cosine similarity computed in the Tag-Resource context to get a weighted graph of related tags. Standard tags are detected as those directly mapped to WordNet using the computed similarity for disambiguation, compound tags are treated with heuristic filters before being mapped, and jargon tags are not directly mapped but linked with their most related standard tag. The benefit of this approach is to integrate specific terms absent from termino-ontological resources in the folksonomy enrichment.

The method proposed by Specia & Motta (2007) expands the set of resources exploited and makes use of string-based and structure-based similarity metrics. After solving spelling issues using the Levenshtein metric and disambiguating acronyms using Wikipedia, tags are clustered by grouping similar tags using the cosine measure computed in the Tag-Tag context (see 3.3.2). Then, for each cluster, the system looks for elements from termino-ontological resources that have the same label as the tags. In case of success, the system is able to map the concepts and their properties to the tags. The result is a set of clusters of tags enriched with semantics, but the experimental results show that this type of method requires that the termino-ontological resources used to infer the semantic relations between the tags provide a good coverage of the domain of study.

The system developed by Angeletou *et al.* (2008) is a continuation of the work of Specia and Motta but differs from it by skipping the phase of clustering similar tags, and by integrating a phase of sense definition and disambiguation of the tags with the help of Wordnet and other terminological resources. Indeed, ontologies available on the Semantic Web are still sparse, and the concepts of these ontologies might not be syntactically equivalent to a given tag of a folksonomy, but rather be labeled with, for instance, a synonym of that tag. Thus, after a first phase of lexical processing of the tags (eliminating isolated tags or user-specific tags which cannot be mapped with already known syntactic categories, such as "b&w"), each tag is expanded with synonyms or hypernyms found in generic ontologies such as Wordnet, producing a semantically expanded tagset. The next phase, called semantic enrichment, consists in looking within online ontologies for

concepts matching one of the terms of each expanded tagset. These matching concepts are called "semantic entities" as they may not belong to the same ontology. The next step in this phase of semantic enrichment consists in discovering relationships between the original tags by exploiting ontology matching techniques to establish semantic relationships between the semantic entities linked with the tags. The result of this approach is a set of semantic entities connected, via the tags, to the tagged resources.

Another integrated approach (Van Damme *et al.*, 2007) proposed integrating more online resources (such as Wikipedia) and use each resource in several ways. For instance, Wikipedia is used to check spelling or acronyms, but also to map tags with concepts. Furthermore, Van Damme *et al.* (2007) suggest involving the community of users to validate the semantic information previously inferred. Their project can thus be seen as a wish to integrate and extend semantic enrichment of folksonomies, and to involve the users themselves in an ontology engineering process, as proposed by Braun *et al.* (2007) (see below, in section 3.4.5).

3.4.1.3 Comparison of tag-concepts mapping methods

A first global remark, which can be made for most of the approaches presented above, is that termino-ontological resources often cover a limited set of users' tags. Laniado *et al.* (2007) for instance estimated that out of a sample of 480000 distinct delicious.com tags, only 8% were contained in WordNet lexicon. However, they also observe that the more popular a tag, the greater its probability of being in WordNet, and this phenomenon follows a power law distribution. The complementarity between users' tags and concepts from termino-ontological resources has been noted and evaluated by Gligorov *et al.* (2010) in their attempt to map users' tags with a professional vocabulary in the domain of Sound and Vision, GTAA, and the lexical database Cornetto, similar to WordNet but in the dutch language. Interestingly, around 50% of tags matched the Cornetto synsets, but only 11% where matched with terms from the GTAA vocabulary. Even if the matching with lexical database is higher in the study of Gligorov *et al.* (2010), we still miss half of the tags, and this low rate is due to the common use of tags that do not correspond to real words, and the even lower matching rate for the professional domain GTAA vocabulary can be explained by the fact that users tend to tag with notions that are complementary to those used by professionals. Gligorov *et al.* (2010) asked a professional cataloguer to qualitatively evaluate users' tags. She found that 45% of the tags used for the most tagged video were useful and she noted that users' tags tend to describe the content of the videos rather than its subject, and users tended to focus on objects appearing in small time frame rather than focusing on logical segments like a scene or a sequence. This shows the semantic gap existing between professional descriptions, based on a controlled vocabulary and focused on the subject of the content, and users' tags, more prolific and focused, in the case of video, on lower level information. These remarks are also reinforced by the study of Golder & Huberman (2006) (see section 3.2.1), which

showed that tags where used to fulfill many other purposes than the description of the topic of tagged resources.

On the other hand, some distinctions can be drawn between the different ways of mapping tags with concepts. Integrated approaches (Specia & Motta, 2007; Lin *et al.*, 2009; Van Damme *et al.*, 2007) apply the mapping of tags with semantic resources considering different types of relatedness measure of tags, while Tesconi *et al.* (2008) consider sets of tags belonging to the same user, and Torniai *et al.* (2008) apply their mapping on a single tag at a time but consider the concepts around the hierarchical structure of the target concept.

Finally, these approaches may have different types of application. The semantic enrichment of tags proposed by Specia & Motta (2007) can be used by all the contributors of a folksonomy, and may be useful to a whole community. The tag disambiguation of Tesconi *et al.* (2008) can be applied to different purposes, such as the profiling of the tagging of a user, providing for richer information when consulting the bookmarks database of this user. However, if we apply the algorithm proposed by Tesconi *et al.* (2008) to all the users of a community, we can measure or detect the divergences existing among the users and, for instance, propose them to discuss their points of view in the case of the collaborative construction of an ontology. The method proposed by Torniai *et al.* (2008) seems more appropriate when working with already chosen domain ontologies (such as within an organization or a community of interest who are maintaining their own ontology) and is complementary to the approaches of Passant *et al.* (see section 3.4.4) who proposed an ontology framework to capture tag-concept mapping at tagging time, unlike the methods presented in this subsection. Indeed Van Damme *et al.* (2007) suggested involving users in the semantic enrichment of tags but did not discuss how these contributions can be captured and further exploited. This is a point we are going to see in the next subsections.

3.4.2 Involving users in the semantic structuring of tags

3.4.2.1 Preliminary questions

Weller & Peters (2008) defines the different aspects of folksonomy improvements taken at a collaborative scale. They define different structural levels on which folksonomies may be improved and edited by the contributors to a folksonomy. (a) Whole document collection vs. single document level. Shall we edit the tags as associated to all the documents, or restrain the editing to tags associated to a single document? (b) Personal vs. collaborative level: should we share the edition of tags or should it be personal? (c) Intra and cross-platform level: depending on the platform we are considering, the treatment applied may differ. The collaborative dimension of the process of ontology building in a Web 2.0 environment has been covered by some approaches presented further in section 3.4.5.

Another issue is the incentives of users to participate in the semantic enrichment of folksonomies. The problem is that users may rarely be keen on providing

the effort of structuring tags. This aspect has been addressed by the *games with a purpose* paradigm, which consists in setting up games whose output is utilized to complete tedious tasks such as massive indexing or ontology construction. Using games with a purpose was first introduced by von Ahn & Dabbish (2008) with the ESP Game[16] where players who do not know each others are paired and presented with images for which they have to agree on the label. As users do not know with whom they play, they have to use words as consensual as possible in order to reach an agreement more easily, thus making these games an opportunity to collect shared knowledge. Gligorov et al propose a similar game called "Waisda?" to tag videos[17], where two users are tagging videos at the same time while watching them, and if they use the same tag for the same time frame, that tag is said to be *verified* as it has a higher potential validity.

The principles of *games with a purpose* have also been applied by Siorpaes & Hepp (2008) to acquire ontological knowledge with Ontogame. They proposed a multiplayer game-like framework where players are presented with different tasks and, with no mean to communicate directly, have to agree on the choice they make in order to earn credits. The tasks they have to complete include typical tasks needed to build ontologies, or to match ontologies, or to annotate resources with semantic annotations. In the ontology construction scenario, players have for instance to agree on the label to give to a class definition or on the relation to assign between two classes. Siorpaes & Hepp (2008) have experimented the Ontogame in particular to build an ontology out of Wikipedia pages, and their experiments on different other scenarios have shown that users were willing to participate in such games and provided for good quality inputs for ontology making.

The principle behind *games with a purpose,* that is, to exploit the expertise of the mass of users to perform tasks traditionally assigned to a few experts, is closely related to the concept of *crowdsourcing.* This concept broadens the contexts in which users are integrated as parts of a collective process. For instance, this idea can consist in outsourcing certain tasks to human agents working remotely such as in the Amazon's service Mechanical Turk[18]. This idea can also be applied in non-for-profit contexts as a guiding principle to collect users contributions in a knowledge-based system. For example, Lin & Davis (2010) proposes applying this principle to enhance ontology construction from folksonomies by capturing semantic relations between searched-for tags and tags suggested from computations.

Finally, involving users in folksonomy enrichment may greatly help improve the quality of these shared knowledge structures. For instance, to tackle the problems of ambiguity or misuse of tagging (like spam), Gruber (2007) proposed to "tag the tags". It would then be possible to state that this tag is the synonym of this other tag, or that this tag does not suit this object, integrating mechanisms of regulation like those observed on Wikipedia.

[16] http://www.gwap.com/gwap/gamesPreview/espgame/

[17] http://research.imagesforthefuture.org/index.php/ waisda-video-labeling-game-evaluation-report/

[18] http://mturk.com

3.4.2.2 Augmented tagging

Tanasescu & Streibel (2007) applied the idea of Gruber and extended social tagging systems with the possibility to tag the tags themselves and the relationships between them. Indeed, classical tagging systems allow their users to add a "tagging relationship", that is a "is_tagged_by" link between a keyword and a document or a Web resource. But richer information may be obtained from the tagging activity, like the relationships between the tags. These tagging can easily be expressed with triples, such as "car" - "is_a" - "vehicle", all these tags being freely added by the users. This added semantic data can then be exploited to assist navigation and to suggest to the user other terms semantically related to her query. To prevent irrelevant contributions, the authors proposed solutions based on votes for some tags, in order to appreciate or depreciate them, or solutions based on points that will be granted either to contributors to the tagging task, or to evaluators of the tags of others. Other incentives to contribution could also be provided with *games with a purpose* as seen above.

Huynh-Kim Bang *et al.* (2008) proposed an extension of the social bookmarking tool Scuttle [19] which let the users add semantic relations between tags while tagging. The goal is to provide communities members with a tool to organize the documents they share, and this tool was conceived with the idea of merging the flexibility of social tagging and the possibilities of inference brought by semantic formalisms. Thus, they proposed to use structurable tags, that is, tags which can be linked to other tags with a limited set of semantic relationships (in contrast with the openness of the "extreme tagging" of Tanasescu & Streibel). Two types of semantic relationships are offered to users, each symbolized by a character that users add while tagging : the subsumption of a tag by another tag symbolized by the sign ">" (as in "`plane > airbus`", meaning that tag "`plane`" subsumes tag "`airbus`"), and the synonymy between two tags symbolized by the character "=" (as in "`test = tests`"). Just as all tags are aggregated within a folksonomy, the semantic relationships created by users are also aggregated, meaning that once a user creates a relation between two tags, this relation will be applied to all the users using the same tags.

We should also mention here the "machine tags" in Flickr[20], where users can define enriched tags in the form of predicate:attribute=value, such as `dct:description=New-York` or `geo:lat=42.33`. This type of tags can easily be translated and modeled into RDF triples via the Flickr API[21].

Some other tools, such as Gnizr[22] and Semanlink[23] (Servant, 2006), also propose users structuring tags by specifying subsumption relations later exported in RDF. Gnizr describes tags and semantic relationships between them with on-

[19] http://sourceforge.net/projects/scuttle/
[20] http://www.flickr.com/groups/mtags/
[21] http://librdf.org/flickcurl/
[22] http://code.google.com/p/gnizr/
[23] http://www.semanlink.net

tologies presented in 3.4.3, such as SKOS for the subsumption relation and the TagOntology for the tags. Semanlink proposes its own model, but which inherits from SKOS. However these tools offer very limited sharing features of this semantic metadata, an issue that has been addressed by the works presented below which aims at providing a framework of ontologies to support the sharing of semantic metadata across the web.

3.4.3 Ontology framework for interlinking social data and tags across the web

Gruber (2007) states that there is no opposition between ontologies and folksonomies and proposes constructing an "ontology of folksonomy". The "TagOntology" is a project of an ontology dedicated to formalizing the act of tagging. This model brings in four entities to describe tagging : the tagged object or resource; the term used to tag; the user tagging; and the domain in which the tagging takes place (it can be the service used for instance). Gruber suggests reifying the tagging and to consider each tag as an object as such, and below we will see the different implementation of these ideas.

The Semantically Interlinked On-line Communities (SIOC) project of Breslin *et al.* (2005) provides developers of social Web platforms a formal and technological framework to describe the resources exchanged within and across on-line communities. The formal scheme they propose uses other ontologies like the Simple Knowledge Organization Scheme SKOS[24] which describes the structure of thesauri, and Friend Of A Friend (FOAF[25]) designed by Brickley & Miller (2004) and which describes the multiple identities and acquaintances of a user (see figure 3.9). SIOC describes the most common elements present on Web sites of communities: the concept of "site", the concept of "post" of a Weblog, the concept of "forum", etc. Starting from this vocabulary, the SIOC project proposes tools to automatically annotate the content of some common Web applications (e.g. wordpress.org) according to the SIOC ontology.

The SCOT[26] project proposed by Kim *et al.* (2007) aims at representing a folksonomy model with the help of ontologies. This model of tagging is an extension of the Tagging Ontology proposed by Newman *et al.* (2005). The first and central entity is the reified "tagging" modeled with the class `tags:Tagging`, which, in SCOT, corresponds to a post, *i.e.* a tagging in this sense can link several tags to a single resource and a single user. An additional class, `tags:RestrictedTagging`, has been proposed by Newman *et al.* (2005) to model ternary relations linking one tag to one resource and one user (we will see below how Passant & Laublet (2008) exploited later this class to attach a meaning to a tag). Then, in Newman's tag ontology, the tagger was modeled with the `foaf:Agent` class, and SCOT has extended the model to link a tagging to a `sioc:User`. Tags are modeled with the `scot:Tag`

[24] http://w3.org/2004/02/skos/
[25] http://foaf-project.org/)
[26] http://scot-project.org

class, itself a subclass of the `tags:Tag` class, itself a subclass of the `skos:Concept` class. SCOT provides also for a class (`scot:TagCloud`) to model clouds of tags as the containers for the tags of a user, the resource annotated with tags being modeled as `sioc:Item` (see figure 3.10). SCOT exporter allows mapping content from a given Content Management System (eg. Wordpress) into SCOT ontologies. This offers in turn a better interoperability between different tag spaces and the possibility to form groups of similar or related tag clouds. One of the most direct use case of the SCOT model is the use of meta-search, which allows users to find similar folksonomies. The similarity between two local folksonomies can be for instance based on the number of common tags, that is, the number of tagging using the same `scot:Tag` instance (since all tags spelled the same will be automatically merged in the same instance of the `scot:Tag` class).

Other models of tagging have been proposed, such as the one developed by Echarte *et al.* (2007) or TagOnt[27], but none of them seem to have been as widely adopted as SCOT, or SIOC. The Semantic Desktop project NEPOMUK also proposed a class to describe tags through its ontology NEPOMUK Annotation Ontology[28]: the class `nao:Tag` and a property `nao:has_tag`, but without considering the action of tagging as a core element of the model of a folksonomy. Kahan *et al.* (2002) also proposed *Bookmark*, a model to describe the infrastructure of the social bookmarking platform Annotea[29]. Even if this model does not include the notion of tags, it allows linking a resource with the terms used to annotate it with the class `bookmark:Topic` and the corresponding property. This model also proposed organizing the topics with the property `bookmark:subTopicOf`, similar to the SKOS property `skos:broader`. We should also mention here the microformat[30] `rel:tag`. Microformats are the product of a community initiative which defines structured metadata which can be embedded within Web pages via simple html tags attributes [31]. Thanks to GRDDL (Gleaning Resource Descriptions from Dialects of Languages[32]), which allows transforming XML dialects into plain RDF, we can transform annotations written with the `rel:tag` microformat into RDF triples based on the `scot:Tag` class for instance.

Some more recent works proposed evaluating the conceptualization and the expressiveness of current tagging models. Kim *et al.* (2008b) proposed a review of current ontologies aimed at modeling tagging and folksonomies, and compare them with regards to their ability (1) to represent tagging, as an individual act involving a user, a tagged resource, and a tag, and (2) the features of folksonomies (such as their container, the co-occurrence between tags, etc.). Thus, Kim *et al.* (2008b) compare tagging models according to their coverage of the wealth of data pertaining to folksonomies but do not really discuss the conceptualization of the

[27] http://code.google.com/p/tagont/
[28] http://www.semanticdesktop.org/ontologies/nao/
[29] http://www.w3.org/2001/Annotea
[30] http://microformats.org/
[31] such as `<a href="http://technorati.com/tag/tech" rel="tag">tech</a>`
[32] http://www.w3.org/2001/sw/grddl-wg/doc29/primer.html

tag itself. The NiceTag[33] ontology proposed by Monnin *et al.* (2010) aims at accounting for the diverse nature and uses of tags, by focusing on the modelization of the different possible relations between a tag and the tagged resource. Furthermore, the reification of tagging in the NiceTag ontology is based on the use of named graphs (Carroll *et al.*, 2005; Gandon *et al.*, 2007) mechanisms, which allow capturing assertional intents while keeping the flexibility and simplicity of RDF binary relations. As a result, NiceTag is able to include different models of tagging, thus serving as a pivot representation allowing the bridging of existing models.

These ontologies aim at realizing the "Web of Linked Data" (now named Linking Open Data[34]), which consists in the evolution of the initial vision of the Semantic Web where the sources of data and the schema describing them are located with http URIs and interconnected in a decentralized way. This project can be realized thanks to ontologies describing the infrastructures where data is stored. This is precisely the goal of the Vocabulary of Interlinked Datasets (voiD), but other models such as SIOC, SCOT, and FOAF can also serve this purpose as they describe the actors of the social web and the type of data they exchange. Another fundamental piece of the Web of Data consist in ontologies describing the content or the topics of the data, such as DBpedia (Auer *et al.*, 2007), which publishes the Wikipedia content and its category structure in a publicly available RDF data store[35]. This project aims at enabling users to access content not only via HTML hyperlinks, but also thanks to the concepts that can be attached to them.

3.4.4 Linking tags and concepts at tagging time

In section 3.4.1 we saw some approaches aimed at automatically linking ontologies concepts with already created tags. In this subsection, we are going to see some other approaches that propose linking tags with concepts at tagging time. These approaches focus on the integration of such functionalities within the tagging interfaces, or provide a coherent semantic web framework to enable the interconnection of tagging data with ontological resources and allowing inference mechanisms.

Passant (2007) proposes strengthening the social tagging interface of a corporate Weblog with a centralized ontology. In his approach Passant considers tags as character strings linked with formal concepts with semantic properties. This association of tagging and ontologies is used here to disambiguate the different meanings of tags. While tagging, users are suggested to connect the terms with which they are tagging to a controlled vocabulary. Thus, if a tag corresponds to two different concepts (for instance the tag "RDF" may correspond to "Resources Description Framework" or to "Rwanda Defense Forces"), the system asks the user to choose the appropriate concept. When no existing concept matches the user's concept, users are free to propose a new one to the administrators, who in turn

[33] http://ns.inria.fr/nicetag/2009/09/25/voc
[34] http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData/
[35] http://wiki.dbpedia.org/OnlineAccess

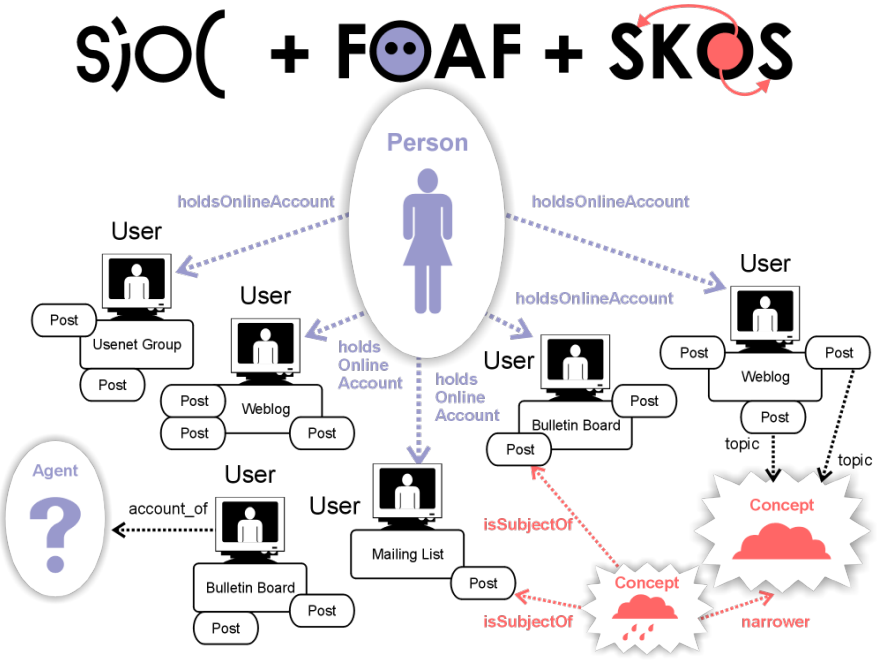


Figure 3.9: Modeling online communities: the SIOC model

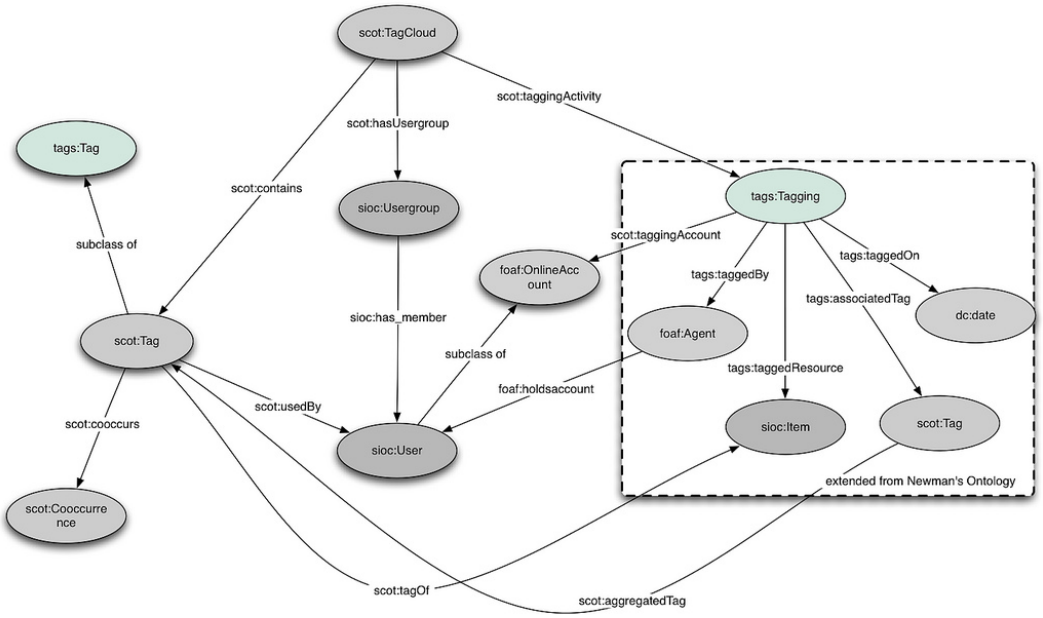


Figure 3.10: Modeling tags and folksonomies: the SCOT (scot:) and TagOntology (tags:) models

will put it in the right place in the ontology. Social tagging is seen here as en empowerment of the construction of an ontology which is used in return to help disambiguating the possible meanings of a tag. While, the approach of Passant (2007) focused on the synchronization of the life-cycle of a folksonomy and a corporate ontology maintained centrally, the approach of Good *et al.* (2007) relies on a set of already existing structured vocabularies used as a source to select tags from. Both of these methods use tagging interfaces and ontologies to annotate resources with unambiguous concepts. However, the system of Passant (2007) lets users first adding tags, and if a tag can be related to one or more concepts, then this tag is linked to the concept the user choose from the list of matching concepts. In the case of Good *et al.* (2007), the situation is a little different since users are provided with a list of concepts in addition to the tags already present in the tagging base. For instance, if a user has typed "hyp", the system first let him choose the appropriate ontology among a list of available ontologies, and then suggest a list of concepts. These concepts are proposed in the manner of an "autocompletion" list of concepts whose labels match with the first characters the user has typed in. Both of these approaches are limited by their dependence on a professional context, either for maintaining a central ontology, or for the source of the vocabularies utilized.

However, with the growth of the Web of Linked Data, some other approaches aim at linking tags and concepts at the scale of the Web. Indeed, as the size and coverage of external resources providing for identifiers for unambiguous concepts grow, it becomes feasible to envision similar systems in an open environment such as the Web. Passant & Laublet (2008) have proposed the MOAT ontology (moat-project.org) that allows users to link the tags they use with a resource's URI representing their meaning. The MOAT ontology reuses other ontologies such as the FOAF (Brickley & Miller, 2004) ontology to represent the users, or the TagOntology (Newman *et al.*, 2005) to represent the tagging activity, and specifically the "restricted tagging" which corresponds to the ternary link in folksonomies between a tag (defined with MOAT's own class `moat:Tag`), a user, and a tagged resource. Restricted tagging corresponds to a tag action, and the aim of MOAT is to allow a user to link this tag action to its intended meaning represented by a meaning's resource (see a graphic representation of MOAT in figure 3.11). The meaning resources can be any Web pages (such as Wikipedia pages), but also concepts of online semantic resources such as ontologies or thesauri. Passant & Laublet (2008) make an important distinction between local and global meaning. The local meaning is the meaning of the tag action, and one tag action can only have one meaning. Then this meaning is also linked to a tag which is considered, in MOAT, as a mere string of characters giving a human readable label for the intended meaning of the tag action. One `moat:Tag` can thus have several meanings. As a continuation and a variation on MOAT principles, CommonTag has been proposed [36] as a semantic annotation framework for tags. The main idea is that users should use unam-

[36] http://www.commontag.org

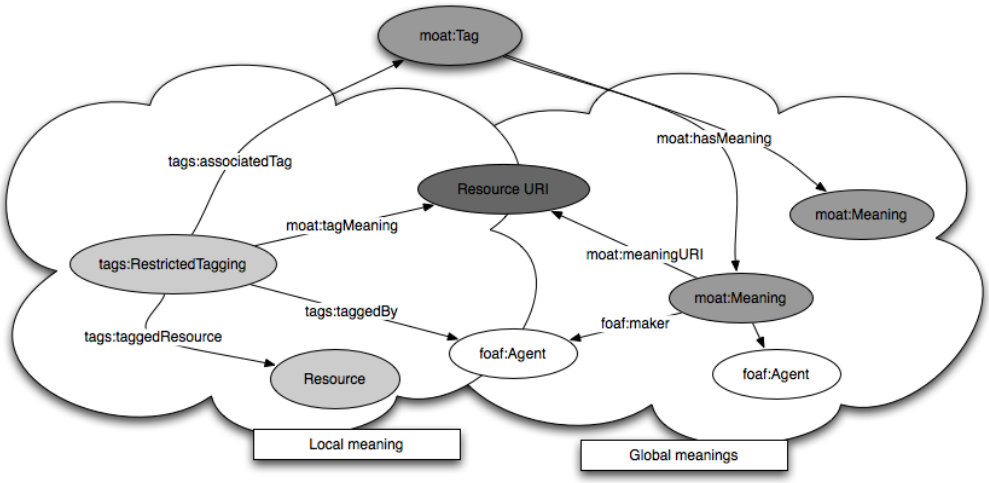


Figure 3.11: Description of the MOAT ontology to link tags with unambiguous meanings (Passant & Laublet, 2008)

biguous concepts to tag, instead of using labels that could possibly be linked a posteriori to such unambiguous concepts as with MOAT. Thus, CommonTag rely heavily on specific designs of the tagging interfaces, which should be capable of helping users choose a concept to tag.

Thus, all of these methods presented here rely on the participation of users who are asked to raise themselves the ambiguity of their annotation by either choosing among possible meanings for a given tag, or by choosing the appropriate ontology concept matching their "tagging intention".

3.4.5 Tagging and collaborative ontology maturing processes

Folksonomy enrichment aiming towards termino-ontological structures has some strong connections ontology building. We can mention in this regard general approaches to build ontologies from scratch such as METHONTOLOGY (Fernandez-Lopez *et al.*, 1997), or approaches to build ontologies from texts (Aussenac-Gilles *et al.*, 2000b; Cimiano, 2006) or from databases (Golebiowska, 2002). The main steps of the method proposed by Fernandez-Lopez *et al.* (1997) to build ontologies from scratch are the following:

1. **specification** of the purpose and scope of the ontology, as it is fundamental to know what the ontology will be used for,
2. **elicitation** of the knowledge bases to be exploited in order to acquire the knowledge that is to be formalized in the ontology,
3. **conceptualization** of knowledge acquired thanks to intermediate representations (folksonomies consist in a knowledge base, but they can also be seen as an intermediary representation as soon as it is structured thanks to the unveiling of its emergent semantics)

4. **formalization** of the conceptual mode resulting from the previous step,
5. **integration** of other already existing and relevant ontologies
6. **implementation** of the ontology thanks to one of the available language
7. **evaluation** of the ontology with respect to a reference frame or with a series of standard tests to verify its consistency
8. **documentation** of the ontology to ease its reuse by other ontologists,
9. and finally, **maintenance** of the ontology is to be performed all along its life time.

Indeed, as a folksonomy consist in a list of terms that users chose to index resources, we can look at them as valuable resources from which to build ontologies. Even if the structure of texts and folksonomy largely differs, it is relevant for this study to have a look at the method proposed by Aussenac-Gilles *et al.* (2000a) to build ontologies from the analysis of a textual corpus. The main steps of this methodology are the following:

1. Corpus constitution : this step should be done by an expert of the domain to model and consists in collecting a corpus of texts, or other terminological structures, which cover as broadly as possible the domain.
2. Linguistic analysis: with the help of Natural Language Processing (NLP) tools, such as terms extractor (*e.g.* Lexter (Bourigault & Jacquemin, 1999)) or relation extractor (*e.g.* Caméléon (Seguela & Aussenac-Gilles, 1999)), this step consists in extracting from the corpus a raw set of terms and lexical relations.
3. Normalization: the goal of this step is to first select the terms and lexical relations that will be kept, and then to turn terms into concepts, and lexical relations into semantic relations. The outcome of this step is an *informal* ontology.
4. Formalization: this step consists in building the ontology by turning concepts and relations into formal concepts and formal properties. Then the ontology is validated to be sure of its logical consistency, in particular regarding inheritance constraints.

This approach to ontology design thus proposes a combination of automatic processing and human expertise, but still lacks a collaborative component.

The first type of approach that adressed the development of ontologies involving a community of users consist in distributing this task. For instance, Sunagawa *et al.* (2003) proposed a framework for synchronizing distributed ontologies developped separately. The ontology is divided in component ontologies. For example, an ontology about vehicles is divided into several component ontologies:

one about aircraft, one about vehicles running under or above water, and another about vehicles running on the earth. These component ontologies are connected via some of their concepts with 2 types of relations, "super-sub" and "referring-to". One concept C_A in an ontology O_A can be linked to another concept C_B in an ontology O_B with a "is-a" relation; O_A and O_B would then share a "super-sub" relation. Then, one concept from O_A can refer to another concept from O_B, and then O_A would be considered as the "referring-to" ontology, and O_B the "referred-to" ontology. Sunagawa *et al.* proposed a series of rules to manage the changes that should be made on an ontology when changes are made on another connected ontology according to the type of dependency ("super-sub" or "referring-to") and on the type of change (deletion of a concept, change of a label, specialization of a concept with sub-concepts, etc.). This approach to the management of distributed ontologies has been integrated in the ontology editor "Hozo"(Kozaki *et al.*, 2002). Another project, called DBin (Tummarello *et al.*, 2006) proposed a framework for editing pieces of ontologies, which are then exchanged following peer-to-peer protocols. Each piece of ontology is devoted to a given domain and is administered by a power user who is in charge of detailed and advanced work on the formalization. Then regular users who are interested can join one of these groups and easily contribute through a graphical user interface. The main benefit of this approach is that it lowers the barrier to contribution thanks to the structuring around groups lead by power users. However, this approach still requires a high level of involvement and learning of the interface. In the same trend of sharing the semantic individual actions, Abbattista *et al.* (2007) proposed an approach to assist the construction and the evolution of ontologies using collaborative tagging principles. Each user is thus seen as a "knowledge organizer" which contributes to the construction of a collective knowledge base by sharing his structured data. The tool they developed seeks to assist the users in this organization process by (1) providing, for a selected resource, relevant metadata from several repositories, (2) assisting the user in disambiguating the chosen terms using lexical resources such as, *e.g.*, Wordnet (Miller *et al.*, 1990), (3) suggesting the user to place the terms in relevant location within a personal taxonomy. The user then choose to share parts of his knowledge base, called "binders", that is, groups of annotated resources and the corresponding portion of his personal taxonomy, the result being a shared information space.

Another type of approach to collaborative editing of ontologies consist in involving all the member of a community into the contribution to the shared ontology. To this regard, Braun *et al.* (2007) highlight the lack of integration of the collaborative processes in current ontology engineering tools and suggest using the dynamics of the use of Social Web platforms such as wikis or social tagging systems. For example, semantic wikis are wikis that include semantic functionalities, such as an indexing of pages with formal vocabularies, and that can also be seen as useful tools to collaboratively build ontologies. Indeed the ontologies elaborated in such a context can be extracted from the categories used to organize or index the context of the wiki pages. For example, Auer *et al.* (2007) applied this principle

to build a thesaurus out of the category structure of Wikipedia. Braun *et al.* propose the following description of the ontology maturing process. (1) The first step is the consolidation of the terminology used in the communities, which could be achieved by analyzing the folksonomy to extract the tags that should be included in the ontology, (2) the formalization is performed by identifying the concepts and semantic relationships out of the shared terminology, and (3) the axiomatizing consists in formalizing more semantic relations between the shared concepts. This description is close to the methodology proposed by Aussenac-Gilles *et al.* (2000b), but the main difference in the case of ontology maturing from folksonomies is that the NLP tools are not suited to extract terms from folksonomies. Thus the approach of Braun *et al.* (2007) consists in exploiting the dynamics of social tagging platforms to fuel the process of ontology maturing by involving users from the start, and allowing each user to turn a tag into a more elaborate conceptual entity. This process should also be integrated in daily tasks such as information seeking or distribution. The benefit could be a better motivation from the users to participate in ontology-maturing as they wish to retrieve more accurate content in order to be more efficient, or want to make their own publications more visible. Braun *et al.* (2007) implemented a prototype which consists in a bookmarking service with some extra capabilities such as (1) suggestion of tags from the already existing ontology, (2) possibility for all users to add or edit new "semantic" tags, (3) knowledge representation models based on SKOS which includes narrower, broader, and related semantic relationships. In this regard, we should mention that this particular implementation applied the approach of Braun *et al.* (2007) to the elaboration and maturing of a thesaurus, which involves a lower level of formalization in contrast with ontologies. Similarly, Buffa *et al.* (2008) developed a semantic wiki in which any user can tag the pages and organize globally the tags of the folksonomy, just as they would do for an ontology or a thesaurus. The idea is that each action of a user benefits to all the other users. To this respect, Braun *et al.* (2007) remark that current collaborative tagging systems offer few functionalities to structure the vocabularies, and when they do, the structuring is not shared among users (for instance in delicious.com, the "super tags", which are used to subsume a bundle of tags, are not shared).

Finally, other approaches tend to lower the barrier to participation. In the corporate blog supported by a centralized ontology proposed by Passant (2007), users who tag their posts do not actually directly participate in the ontology maturing process, but merely propose new instances that are then used to populate the ontology, the actual ontology design being let to the systems administrators.

Following the distinctions brought by Weller & Peters (2008) between the individual and the collective level at which folksonomies can be modified, we can distinguish the approaches presented here where the users merely propose new concepts (Passant, 2007), with approaches where users can directly edit the whole shared ontology or thesaurus (Braun *et al.*, 2007; Buffa *et al.*, 2008), or with approaches where individually maintained ontologies are synchronized (Sunagawa *et al.*, 2003; Abbattista *et al.*, 2007). In the latter case, there will be a need to fine-tune

sharing strategies or to use ontology mapping techniques (Euzenat & Shvaiko, 2007) in order to efficiently combine these shared ontologies into coherent structures.

3.4.6 Comparison and intermediary conclusions

In table 3.7 we compare the approaches presented above. This section first covered approaches which proposed automatic methods to link tags to online ontologies. However these approaches suffer from the limited coverage of specific domains due to the scarcity of formal ontologies online[37]. To overcome this limit, Tesconi *et al.* (2008) and Ronzano *et al.* (2008) build sets of terms-meaning by mining Wikipedia, and then link each tag of delicious.com users to a unique meaning. Similarly integrated approaches combine the use of ontological resources with similarity measures, such as Lin *et al.* (2009), Specia & Motta (2007), or Van Damme *et al.* (2007) which also suggested integrating users intervention to build, at a reasonable cost, genuine "folks-ontologies".

The second part of this section covered approaches aimed at involving users in the semantic enrichment of folksonomies. Huynh-Kim Bang *et al.* (2008) proposes the concept of structurable tags where users can define semantic relations between tags, and Tanasescu & Streibel (2007) suggest letting the users tag the links existing between tags. The two latter approaches do not make direct use of semantic Web formalisms as they focus more on the flexibility of the system than on logical consistency of the knowledge structure obtained.

In this regard, Gruber (2007) suggested constructing collaboratively an ontology of folksonomy to support more advanced use of tagging. This idea has been implemented by Newman *et al.* (2005), and further improved by Kim *et al.* (2007) which integrated their SCOT ontology with SIOC Breslin *et al.* (2005), another ontology modeling users' interaction on social Web platforms.

This ontology framework has been exploited by some systems proposing to the users to tag with concepts at tagging time. Passant (2007) developed a semantically augmented corporate blog where users can attach their tags to the concepts of centrally maintained ontology, while Good *et al.* (2007) suggest terms from professional vocabularies fetched online at tagging time. Later, Passant & Laublet (2008) have extended these interconnected schemas with MOAT, an ontology allowing to link tags with online resources, similarly to CommonTag, to define precisely the meaning of tags and to tie them with the "Web of Linked Data"[38], a vision of the Web where resources are linked with each other thanks to the concepts which can be attached to them.

Finally, the approaches presented in section 3.4.5 focus on the ontology maturing processes and exploit Web 2.0 tools to achieve this task like wikis (Buffa *et al.*,

[37] For example Cattuto *et al.* (2008) (p.10, Table 3) evaluated the coverage of delicious.com tags in WordNet and found out that the 500 most frequent tags are covered at 80% by WordNet, but this fraction goes down to 61% for the 10000 most frequent tags.

[38] http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData/

	User intervention	Ext. resources	Automatic	Sem. Web
Gruber (2007)	-	no	no	yes
Newman *et al.* (2005)	-	no	no	yes
Tanasescu & Streibel (2007)	yes	no	no	no
Huynh-Kim Bang *et al.* (2008)	yes	no	no	no
Breslin *et al.* (2005), Kim *et al.* (2007), Monnin *et al.* (2010)	-	no	no	yes
Passant (2007), CommonTag	yes	yes	no	yes
Good *et al.* (2007)	yes	yes	no	yes
Specia & Motta (2007), Angeletou *et al.* (2008), Lin *et al.* (2009)	no	yes	yes	yes
Tesconi *et al.* (2008),Ronzano *et al.* (2008)	no	yes	yes	yes
Van Damme *et al.* (2007)	yes	yes	yes	yes
section 3.4.5	yes	no	no	yes

Table 3.7: Comparison table of the approach enriching folksonomies which (1) exploit users intervention, and/or (2) make use of external semantic resources, and/or (3) seek the automatization of the process, and/or (4) are based on Semantic Web formalisms

2008), blogs (Passant, 2007), e-learning platforms (Torniai *et al.*, 2008), personal knowledge organizers (Abbattista *et al.*, 2007), or social bookmarking sites (Braun *et al.*, 2007)

3.5 Knowledge sharing in the social and semantic Web

In this section we give a brief overview of different cases where online interactions and folksonomies play a central role for the exchange of knowledge on the social and semantic Web. We first cover systems dedicated to the task of experts and based on collaborative annotations similar to folksonomies. Then we focus on knowledge sharing platforms (section3.5.2) and semantic wikis (section 3.5.3), which take the benefit of a combination of semantic formalisms and social tagging.

3.5.1 Collaborative information and experts seeking

A problem often posed by collaborative work is expert seeking: how to know "who does what"? The study and the system proposed by Delalonde & Soulier (2007) address this problem in the context of a big organization. Delalonde & Soulier (2007) developed "DemonD", a system that aims at creating the conditions of social interactions which yields to capitalized knowledge. DemonD is grounded

on personal profiles filled in by the users who state their field of expertise and interests with tags and by attaching relevant documents. Then the process starts when one of the user asks a question to the system, which then selects a list of persons and documents relevant to this question. The selection depends on four main criteria (1) matching tags, (2) connectivity with other resources, (3) participation of the person in past interactions, and (4) the reputation evaluated by other peers. Then the system automatically creates a forum of discussion to which the selected persons are invited to participate. The system also includes a step of knowledge capitalization as soon as the original question is answered and that this answer is validated. Thus, this approach includes a collaborative elaboration of knowledge, which is based on folksonomy-like annotation of the resources. To this respect, Delalonde & Soulier (2007) suggest that the system could be enhanced by suggesting tags when the users build their profiles, and that semi-structured vocabularies could also support the annotation process and help more accurate and more relevant selection of resources.

Regarding the support of expert finding purposes with semantics, Aleman-Meza *et al.* (2007) proposed a guideline to combine efficiently different ontologies for expert findings. Their hypothesis is that as persons are described with standards of the semantic web, it becomes feasible to automatically retrieve experts. This vocabulary framework includes SIOC, to describe contents published within online communities, FOAF for personal details, and SKOS which allow refining the semantic relations between concepts used to describe domains of expertise. However, this approach assumes that these ontologies are used and integrated within the tools that potential experts use. In reality however, a lot of different format are utilized (iCal, vCard, FOAF for contact information, BibTex and all the other formats for academic publications, etc.) A possible workaround to overcome this profusion of vocabularies is to use rules to map equivalent classes, such as vCard:homeTel and foaf:phone for instance.

3.5.2 Sharing social and semantic annotations

Other works propose integrating several ontologies to assist the sharing of data. Hausenblas & Rehatschek (2007) designed "mle", a system that automatically treats mailing lists in order to map the structure of email to appropriate concepts of an ontology (SIOC). These annotations, generated in RDF, allow this database to be queried with the language of the Semantic Web SPARQL[39].

Revyu.com (Heath & Motta, 2007) proposes applying the principles of the "Web of Linked Data" (see section 3.4.3) to organize the sharing of reviews of cultural items (books, movies, etc.). Revyu.com includes these principles by (1) allowing anyone to access data stored on other databases in order to prevent redundancies; (2) utilizing RDF to annotate the resources; and (3) keeping open the field of knowledge which can be covered since Revyu.com uses multiple ontolo-

[39] www.w3.org/TR/rdf-sparql-query/

gies and other types of knowledge bases to categorize items.

Other approaches allow to semantically structure the tags in order to enrich social bookmarking services, like GroupeMe![40] (Abel *et al.*, 2007) or inter.est[41] (Kim *et al.*, 2007). GroupMe! extends the idea of social bookmarking : it allows user to build groups, resources can be re-arranged and tagged and these operations produce RDF metadata (following standard ontologies such as DublinCore[42], FOAF, and GroupMe! Ontology[43]). The graphical user interface of system also features presentation adapted to the type of content (picture, video, rss feed, etc.), and drag and drop operations to include a resource in a group. The group structure is then exploited while searching by indicating the context of a resource according to the group in which it is included.

3.5.3 Semantic Wikis

Semantic wikis were among the first applications to exploit the potential of ontologies to support collaborative practices. Gaved *et al.* (2006) thus proposed to develop wikis supporting physical rather than virtual communities, and aimed at providing local information guides, which could serve as a community memory for a geographical area. The Open Guides project aims at highlighting the different types of usages and future uses, and to provide a theoretical framework about wikis of locality. The Open Guides were developed after an adaptation of generic wiki principles in order to describe items with locative elements : latitude and longitude, address, opening time, name of the area. These wikis make use of semantic formalisms since each entry can be exported in RDF/XML, and all the info of each entry is structured following concepts from several vocabularies devoted to the sharing of online resources (FOAF, DublinCore, ChefMoz[44]). Gaved *et al.* also identified common tasks performed by users of wikis, such as locating, exploring, grazing, monitoring, sharing, and asserting about the information described in each entry of the wiki, leading to truly collaborative semantic processes. The analysis of the usages lead to make some other observations concerning the interface which should empower non-technical experts to contribute, the sustainability of the system which can be enhanced by providing more machine-readable metadata, and the spam of diverse kind which tended to pollute the content of the wikis. This return on experiment is of great usefulness for a designer of collaborative tools and addresses the main problems arising from the use of collaborative semantic tools.

SweetWiki (Buffa *et al.*, 2008) is another example of semantic wikis: users can edit and modify pages, and also tag any document published on the wiki. The tags are tied together in a folksonomy expressed with the languages of the Seman-

[40] http://groupme.org/
[41] http://int.ere.st/
[42] http://dublincore.org/2008/01/14/dcterms.rdf
[43] http://groupme.org/rdf/groupme.owl
[44] http://chefmoz.org/rdf/elements/1.0/

	Type of platform	social context
Delalonde & Soulier (2007)	Expert finding	organization
Aleman-Meza *et al.* (2007)	Expert finding	web 2.0
Hausenblas & Rehatschek (2007)	mailing list	generic
Heath & Motta (2007)	reviews sharing	web 2.0
Kim *et al.* (2007), Abel *et al.* (2007)	social bookmark	web 2.0
Gaved *et al.* (2006)	wiki	city
Buffa *et al.* (2008)	wiki	organization

Table 3.8: Comparison table of the approach of section 3.5.

tic Web. All the new tags are collected as the labels of new classes, which are, by default, subsumed by the class "new concept". All the users are then able to organize the tags of the folksonomy, and to edit them, to add new labels in other languages, to create relations of synonyms, to merge classes, etc. The author of pages can also use tags to keep an eye on the activity of other contributors in a targeted manner: each user can specify in her homepage her topic of interest in the form of tags. For instance, a user interested in wikis will put a tag "wiki" in the field "interested by". Then, whenever a page is tagged with "wiki" or a subclass of "wiki", the user will be notified. This function allows watching content that does not yet exist. By keeping track of created or modified pages, and by analyzing over time the behavior of users, it is possible to detect acquaintance networks or communities of interest. This reveals several possibilities: finding the most active person on a given topic, finding the users using similar tags as others, inferring relationships between tags when they are used by the same users, etc.

3.5.4 Comparison and intermediary conclusions

To conclude this brief overview we can see that, except from Delalonde & Soulier, who propose to assist users in finding experts in the social context of corporate organizations, all the other approaches integrate Semantic Web formalisms to describe their data model. In table 3.8, we can distinguish these approaches with the type of content they organize or with the type of services they offer. While some applications target no specific social context (Hausenblas & Rehatschek, 2007), some others are set in the Web 2.0 by dealing with the sharing of cultural items (Heath & Motta, 2007) or simply by providing semantically enriched social bookmarking services (Kim *et al.*, 2007 and Abel *et al.*, 2007). Finally, semantic wikis have been developed to assist the communities of the inhabitants of cities (Gaved *et al.*, 2006), or to assist the activity of organizations in a broad sense.

3.6 Conclusion

We have seen that it is possible to describe a folksonomy and all the activities occurring on social Web sites with ontologies. In this chapter we have compared different approaches that aim at bridging ontologies and folksonomies to support the exchange of knowledge, and to bootstrap the emergence and collaborative construction of shared knowledge representations. These methods can indeed greatly benefit to the final user's experience by proposing more precise tools to navigate within and across platforms based on social tagging.

3.6.1 Summary

A first category of works is aimed towards extracting the emergent tag semantics from folksonomies by measuring the semantic similarity of tags via the analysis of the tri-partite structure of folksonomies. Mika (2005), and later Cattuto *et al.* (2008) and Markines *et al.* (2009), investigated different methods of aggregating the three-modes view of folksonomies onto two-modes views, in order to be able to apply similarity measures. The studies from Markines *et al.* and Cattuto *et al.* propose an analysis of the different types of similarity measures and the semantic relations they each tend to convey. Cattuto *et al.* proposed using 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.* 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 : 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. They also computed the distance and relative placement in Wordnet hierarchy of the pairs of tags retrieved by this method, and showed that the semantic relation conveyed by this measure was of type "related" in thesauri terms. Mika (2005) also applied and evaluated different foldings of the tripartite structure of folksonomies combined with association rules mining, similarly to Schmitz *et al.* (2006), and he showed after a qualitative evaluation that exploiting user-based associations of tags yielded more representative taxonomic relations. The association rule used by Mika is that if the community of users using tag "wind turbine" is included in the community of users of the tag "renewable energy", then the tag "wind turbine" is broader than the tag "renewable energy". Other approaches inferred subsumption relationships such as Heymann & Garcia-Molina (2006) who 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 node or the root node depending on a threshold value. The outcome of the methods mentioned here is a measure of the similarity between tags or taxonomical structures extracted from folksonomies.

Another group of works seek to semantically enrich folksonomies by automatically mapping tags to ontology concepts. Several approaches investigated tag-concepts mapping, using simple string-based matching (Gligorov *et al.*, 2010), or exploiting the synset structure of WordNet (Laniado *et al.*, 2007) or the structure of the target ontology(Torniai *et al.*, 2008). Some other approaches integrate one or several of the tag similarity measures seen above in the mapping process. For instanceAngeletou *et al.* (2008) and Specia & Motta (2007) use a similarity metric computed in the Tag-Tag context to group together strongly related tags, and then map these tags to concepts from available online ontologies. Van Damme *et al.* (2007) proposed including as many resources as possible, using each in a tailored way, and also the validation from users. Addressing the issue of the incentive of the users to contribute to this process, Lin & Davis (2010) proposed complementing automatic processings with crowdsourcing method to collect users' feedback and proposal on the relations between tags.

Another type of approach consists in letting users semantically structure tags or link tags to unambiguous meanings. We can mention in this category the work of Tanasescu & Streibel (2007) who proposed to *tag the tags*, or the work of Huynh-Kim Bang *et al.* (2008) who proposed a simple syntax to specify subsumption (with ">" or "<") or synonymy (with "=") relations between tags. 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 SCOT describes tags as parts of shareable tag clouds, and SIOC describes online communities content. MOAT (Passant & 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 get a richer browsing experience. NiceTag is a model that seeks to account for the usages of tags through a finer modelization of the relations between tags and the tagged resources (Limpens *et al.*, 2009c; Monnin *et al.*, 2010). Its flexibility and the use of named graphs mechanism allow this model to serve as a pivot model for all other tag models, adding a level of pragmatics.

Finally, many of these approaches to semantically enrich or structure tags clearly echo with older methods to build formal ontologies from texts (Aussenac-Gilles *et al.*, 2000a) or databases maintained by communities of users (Golebiowska, 2002). The DBpedia project (Auer *et al.*, 2007) is an example of a lightweight ontology built from collaboratively created content which exploits the Wikipedia pages and its category structure, however without involving users in this specific task. 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 SKOS. Some other researchers also proposed involving users in ontology construction via different types of systems, spanning from custom clients (Tummarello *et al.*, 2006) to semantic wikis (Buffa *et al.*, 2008).

3.6.2 Discussion

The potential benefits of a synergetic combination of semantic web technologies with the dynamics of social tagging and folksonomies can be summarized by the vision of Gruber (2008) who differentiates collective intelligence from collected intelligence. He gives three characteristics of the current systems which collect knowledge: (1) the production of content performed by the users, (2) a synergy between users and the system, (3) increasing benefit with the size of the domain covered. In order to upgrade this type of system towards a collective intelligence, Gruber proposes adding another feature: the emergence of knowledge beyond the mere collection of each contributor's knowledge. He suggests that this fourth feature directly benefits from the integration of the technologies of the Semantic Web. Thus, the potential of hybrid systems, which exploit the benefit of both the ease of use of folksonomies and the support of the formalisms and the methods of the Semantic Web, opens new perspectives for assisting knowledge exchange on the social Web. But several challenges remain, for the full automatization of semantically enriching folksonomies is difficult.

First the similarity measures used in (Cattuto *et al.*, 2008; Markines *et al.*, 2009; Specia & Motta, 2007) or other methods for retrieving taxonomical structures from folksonomies (Mika, 2005; Heymann & Garcia-Molina, 2006) are useful to bootstrap the process, but their accuracy in reflecting the communities' knowledge is limited. Specia & Motta (2007) and later Angeletou *et al.* (2008) showed the efficiency of combining statistical techniques with external ontological resources, but such resources are still scarce and their limited coverage of specific domains greatly hinders the potential of application of this type of approach. Moreover, the granularity of such ontologies may not always be compatible with all folksonomies, and more generally, the capitalization of domain ontologies for several communities will still be limited by the differences in the conceptualization between these communities.

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. Indeed, the success of folksonomy comes for a big part from its simplicity of use. Sinha (2006) showed in her social and cognitive analysis that tagging requires less cognitive effort than choosing a unique category. Tagging is simpler since it allows picking up all the concepts first activated in the mind. Some approaches (Van Damme *et al.*, 2007; Lin & Davis, 2010) try to overcome this limit by mixing automatic handlings with user validation, integrating crowdsourcing principles. However, the social context may also play an important role: incen-

	Computed tag similarity	Tag-concept mapping	Users' contributions	Sem-Web formalisms	Multi-points of view
Angeletou *et al.* (2008)	✓	✓	-	✓	-
Huynh-Kim Bang *et al.* (2008)	-	-	✓	-	✓
Passant & Laublet (2008)	-	✓	✓	✓	-
Lin & Davis (2010)	✓	✓	✓	✓	-
Braun *et al.* (2007)	-	-	✓	✓	-
Our approach	✓	**partly**	✓	✓	✓

Table 3.9: Positioning with other folksonomy enrichment methods

tives to contribute to an enterprise weblog or to a platform of shared reviews may largely differ in the amount of effort users may put in providing additional data. Workmates may be rewarded by their company for good quality contributions, or members of open social platforms may be motivated to make their contributions more visible. Lastly, some divergences and conflicts may also arise when asking users to contribute to the semantic structuring of folksonomies.

3.6.3 Positioning

In table 3.9, we report our positioning in comparison with the different types of approaches presented in this chapter. Our approach to semantically enriching folksonomies consists in creating a synergistic combination of automatic handling, similarly to Angeletou *et al.* (2008), to bootstrap the process. Our system is devoted to semantically structuring tags with thesauri-like relations, but not primarily devoted to mapping them with existing concepts as in Passant & Laublet (2008) or Angeletou *et al.* (2008). One of our contributions consists in a heuristic string based similarity metric based on a combination of different string-based metrics. String based metrics have been used mostly in the literature (by Specia & Motta (2007) *e.g.*) to merge spelling variant tags or map tags to concepts. Indeed, we conducted systematic benchmark of these metrics to evaluate their ability to detect other semantic relationships such as *related* or *hyponym*, and to be able to combine them efficiently. To overcome the lack of accuracy of purely automatic processing, we propose, similarly to Lin & Davis (2010), to capture the expertise of users by allowing them to contribute through user friendly interfaces. However, we also believe that significant progress can be achieved by carefully analyzing the usages of the target communities of a system. We conducted such an analysis in one of our target community, the Ademe agency, in order to take advantage of the tasks already achieved by users to capture knowledge as a side effect of their daily activity. Similarly to Braun *et al.* (2007), we propose users to structure tags with a limited set of thesaurus-like semantic relationships, in order to limit the complexity of this task.

The interface we designed is embedded as seamlessly as possible in a folksonomy navigation tool, avoiding the need to shift to another editing interface as Braun *et al.* (2007). The approach of Huynh-Kim Bang *et al.* (2008) is one of the few to consider the fact that some divergences may arise between contributors, but they do not formally deal with these diverging points of view unlike our approach. Indeed, we propose a formal model to capture diverging points of view, and a set of rules allowing to present users with a coherent experience where they can benefit from other's points of view, while still maintaining their own and not being disturbed by noise from others' contributions. In addition, we propose a conflict solver mechanism, which allows exploiting the users' contributions for ontology maturing purposes (Braun *et al.*, 2007) by pointing to divergences and consensuses among the user's points of view.

3.7 Definitions

In this section we recall briefly the definitions of the key notions that will be used in the remaining of this thesis.

Tag : A tag is a freely chosen keyword that a user of social-tagging system associates to a tagged resource. Such resources may include web pages, as for instance in the social-bookmarking web service delicious.com, or a picture in flickr.com, or a wiki page in the semantic wiki SweetWiki proposed by Buffa *et al.* (2008).

Tagging: A tagging instance is a ternary link associating the tagger, a tagged resource, and a tag. In the TagOntology[45], proposed by Newman *et al.* (2005) and later included in the SCOT[46] model, a distinction is made between a simple tagging (`tags:Tagging`), in which we consider all the tags associated by one user to one resource, and a restricted tagging (`tags:RestrictedTagging`), in which we consider only a single tag associated by one user to one resource. In this thesis, we well use the term *tagging* for a restricted tagging, as it is the most rigorous definition, and we will use the term *post* for a simple tagging (`tags:Tagging`) in the TagOntology sense.

Folksonomy : A folksonomy is defined as a collection of taggings. In formal term, a folksonomy is defined by Hotho *et al.* (2006) as a tuple $F := (U, T, R, Y)$ where U, T, and R are finite sets, whose elements are called users, tags, and tagged resources, respectively. Y is the set of tagging instances such that $Y \subseteq U \times T \times R$. Mika (2005) also proposed a graph definition where a folksonomy can be seen as tripartite hypergraph $H(F) = \langle V, E \rangle$ where the vertices are given by $V = U \cup T \cup R$ and the edges by $E = u, t, r | (u, t, r) \in F$.

Personomy: As a collection of data provided by a group of individuals, a folksonomy can be seen as a collection of the "personomies" of all the users. Let us call Pu the personomy of a given user $u \in U$, where Pu is the restriction of F to u,

[45] http://www.holygoat.co.uk/owl/redwood/0.1/tags/
[46] http://scot-project.org/scot/ns#

i. e., $Pu := (Tu, Ru, Yu)$, with $Yu := (t, r) \in T \times R | (u, t, r) \in Y$ the set of all the tag assignments of user u.

Ontology : This notion, in computer science, has to be distinguished from the notion of Ontology (often spelled with a capital O) in the philosophical sense where it corresponds to the discipline studying the nature of *being*. In computer science, an ontology consists in a symbolic representation that specifies the conceptualization of a domain of knowledge with the help of concepts and properties linking these concepts (Gruber, 1993). We can however also distinguish different types of ontologies:

- **Formal ontologies,** or heavy ontologies : this type of ontology usually relies on rich formal languages (such as OWL[47]) to define their primitives (Gandon, 2008, p. 26). A rigorous definition of formal ontologies is given by Bachimont (2000): "Defining an ontology for knowledge representation tasks means defining, for a given domain and a given problem, the functional and relational signature of a formal language and its associated semantics". The definition of this formal mechanisms and the translation of the knowledge of a domain in these formal languages allow in turn to make inferences and expand greatly the possibility of querying when looking for resources annotated with formal ontologies.

- **Lightweight ontologies**: these ontologies are less focused on inference mechanisms, and contain less or no formal definitions of their primitives and rely thus on lighter languages (such as RDFS[48]) to usually describe hierarchies of types of entities (Gandon, 2008, p. 26).

Thesaurus: The origins of thesauri go back to the 4th century, but the first modern thesaurus is attributed to the British Peter Mark Roget[49]. Modern thesauri and other types of controlled vocabularies, such as taxonomies, consist in notions or concepts that are defined and hierarchically structured. Concepts in thesauri can be contrasted with concepts in ontologies in that they are oriented towards the descriptions of resources, and are not aimed at describing "what something is", but rather "what something is about" according to the SKOS[50] (an RDF schema for thesauri) definition of the `skos:Concept` class. Moreover, the types of semantic relations linking the concepts of thesauri are usually limited to "broader", "narrower", or "related".

Semantic Web : According to Tim Berners-Lee, the inventor of this notion (Berners-Lee *et al.*, 2001): "The Semantic Web is an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation". This brief definition is not sufficient though to describe the broad variety of academic research and technical applications covered today by the Semantic Web. We should also remark that this early vision of

[47] http://www.w3.org/TR/owl-guide/
[48] http://www.w3.org/TR/rdf-schema/
[49] for an historical review of Roget's thesaurus, see Dolezal (2005)
[50] http://www.w3.org/2004/02/skos/core#Concept

Time Berners-Lee has lately evolved towards the notion of the Web of Linked Data, "a term used to describe a recommended best practice for exposing, sharing, and connecting pieces of data, information, and knowledge on the Semantic Web using URIs and RDF."[51]

Tag similarity metric : this type of metric gives a measure of the similarity between two tags, and provides a value between 0 and 1, 1 meaning that both tags are most similar. We can distinguish in the scope of this thesis, two kinds of tag similarity metrics:

- **String-based similarity metrics** : this type of metrics compares the labels of tags without considering the structure of the folksonomy, and they are often based on string-edit distances such as Levenshtein(Levenshtein, 1966). Implementations of such metrics, together with detailed explanations for each method, can be found in the SimMetrics package[52].

- **Structure-based similarity metrics** : this type of metrics is based on the analysis of the tri-partite structure of folksonomies. Examples of such metrics include folkRank metric (Hotho *et al.*, 2006) , and metrics combining (a) a method to aggregate the 3-mode view of folksonomies onto a 2-mode view in (b) one of the three contexts in which this can be done, namely Tag-Tag context, Tag-Resource context, and Tag-User context. Then several common similarity measures can be applied on these 2 mode-views of tagging data, such as *e.g.*, the cosine similarity or the Jaccard coefficient (the details on this type of metric is given in section 3.3.2)

[51] http://linkeddata.org/
[52] www.dcs.shef.ac.uk/~sam/simmetrics.html

CHAPTER 4

Modeling tags and folksonomy enrichment

Abstract. This chapter addresses the conceptualization of tags as an essential part of folksonomy enrichment. Indeed, the model of tag we present aims at covering the diversity in form and usages of the tags. This model considers tags primarily as a link, typed according to the use of the tag, between a tagged resource and a sign used to tag. Then we propose using named graph to embody this record and type it in order to account for other dimensions of tag actions. The enrichment of the folksonomy with semantic links between tags comes as a complement to the richer descriptions of tag actions. To this regard, we give an overview of our approach that is based on usage analysis in order to propose a synergistic combination of automatic processing and users' contributions, with the support of diverging points of view regarding the semantic enrichment of folksonomy.

Contents

4.1 Introduction

As we have seen it in the previous chapter, folksonomies and tags suffer from a lack of explicit semantics. Indeed, current approaches to solve this problem propose to find semantic relationships between tags or to link, a posteriori or at tagging time, tags with unambiguous concepts. In this chapter we show that the lack of expression of the use of tags is also problematic, as a tag taken in a given meaning can have different relationships with the tagged resource. For instance the tag "blog" can be used to state that a resource *is* a blog or *is about* blogs. Moreover, tags can be expressed with different formal means, ranging from machine tags, which follow a specific syntax to be recognized and processed more precisely, to tags consisting in URIs of resources describing their meaning.

Regarding the specification of tagging assignments, we propose a model, NiceTag, which addresses the conceptualization of tags in order to describe the diversity of form and use they can take on. NiceTag aims at describing tag actions primarily as a link between a tagged resource and a sign used to tag, this link being typed to take into account the diverse uses of a tag. The triple describing such tag actions are then encapsulated within a named graph that allow to type the tag actions and describe complementary dimensions.

The enrichment of folksonomies with semantic relationships between tags comes as a complement to the enrichment of tagging assignments with lightweight semantics as proposed by the NiceTag framework. To this regard, we propose an approach that consists in a synergistic combination of automatic processing of the folksonomy and user's contributions. This approach is grounded on an scenario-based analysis of the usages in order to integrate this process in user's everyday tasks. Unlike other approaches that rely on users' will to specify the meaning of each tag action, we propose to structure tags at the level of the folksonomy with a limited set of thesauri-like semantic relationships, thus minimizing user's involvement. However, we also propose to support multiple points of view from the start in order to let each user maintain his semantic structuring of the tags. Our approach also include automatic processing of the tags in order to help each individual contributors by suggesting related tags, but also in order to assist the construction of a global structuring of the folksonomy from these contributions.

In section 4.2 we present the NiceTag model in details and give some examples. Section 4.3covers our approach to semantic enrichment of folksonomy and gives the main steps of the cycle of this process. Section 4.4 concludes this chapter and briefly introduce the following chapters that deal in details with each module of our approach.

4.2 Modeling tagging and tags with the NiceTag ontology

4.2.1 From annotations to tagging

Tagging systems as we use them now on Social Web platforms can be seen as one type of system for sharing digital annotations that met a growing success thanks to their simplicity of use and the low cognitive effort tagging requires in comparison to more elaborate annoations (Sinha, 2005). This successful implementation of digital and shared annotations was preceded by other attempts to set up frameworks for sharing annotations such as Annotea (Kahan *et al.*, 2002). Technically, Annotea is defined as "a system for creating and publishing shareable annotations of Web documents"[1]. It can be considered as a pioneer Semantic Web application whose goal was to demonstrate the possibilities offered by this technology to help users better collaborate and find information more easily. This framework consisted in four basic objects, namely: annotations attached to web documents, replies to others' annotations, bookmarks used to "recall" a resource, and topics. Two RDF schemas describe these objects: the Annotation Schema that described annotations linked to any kind of resource [2] and another schema for bookmarks [3]. A typical scenario of Annotea is the following. Anne has seen a web page of interest and decides to bookmark it. She creates a bookmark object thanks to the Firefox extension Annozilla [4] and annotates it. Annotations can consist in a text or any other resource that Anne find relevant to annotate her bookmark. Anne can also create topics to classify this bookmark with informal categories, similar to tags as one bookmark can be cataloged under several topics, and topics can be shared among users. Topics in Annotea can also be further described and organized in hierarchies thanks to a `subTopicOf` property. Anne's bookmark and annotations are then stored in an Annotea server and are thus available for other users who can reply to Anne's annotations. Annnotations are seen in this context as a technical mediator for collaborative work where several users can take part in discussions around a web document. We see that the Annotea framework is a kind of social bookmarking and social tagging system created at a time when delicious.com was just released[5], but the minimalism of tagging systems brought them the success that classical annotations systems are struggling to achieve.

Recent studies on social tagging systems (such as Golder & Huberman (2006), see 3.2.1 on page 25) revealed that tags are bearing a lot of different functions and role that annotations were supposed to capture in systems such as Annotea. The roles and functions of annotations has been extensively studied, but we can nevertheless cite here the work presented in (Mazhoud *et al.*, 1996, 1995). The authors consider annotating as an activity linked to the activity of reading, and as a consequence the annotation is tied to the annotated resource in a three-fold fashion

[1] http://www.annotea.org/Annotea/User/AnnoteaProtocol-20051226.html
[2] the Annotation schema namespace http://www.w3.org/2000/10/annotation-ns#
[3] the Bookmark schema namespace http://www.w3.org/2002/01/bookmark
[4] http://annozilla.mozdev.org/
[5] http://en.wikipedia.org/wiki/Delicious_%28website%29

to which tagging also comply with. (1) An annotation is prompted by a resource or document that is being read, just as a delicious.com user decides to tag a resource of which he wants to keep a trace. (2) An annotation is attached to this very resource, just as a tagging can be defined as a link between a resource and a tag. (3) The interpretation of an annotation depends on the annotated document, just as the ambiguity of a tag can be raised by considering the tagged resource; for example, if a user tag a web page about Semantic Web with the tag "RDF", we can infer from this link with the tagged resource that "RDF" here probably means "Resource Description Framework" and not "Rwanda Defense Force", the problem being that this elicitation is rarely made explicit in current tagging systems[6]. Mazhoud *et al.* proposed a list of different functions that annotations can have for the annotated document. Annotations can be used to:

- hierarchize, *i.e.* to identify and organize different fragments according to their relevance,
- architecturize, *i.e.* to highlight fragments according to their linguistic feature (definition, illustration, etc.),
- contextualize, *i.e.* highlight some terms and the fragments of the document that are relevant to this term,
- plan other activity to be done in connection to the reading, such as *e.g.,* isolating a fragment to be read again,
- reformulate parts of the document with synthetic annotations
- comment
- link with other relevant references or resources.

We see that these functions of annotations overlap with some functions of tags identified by Golder & Huberman such as the ability of tags to serve as a means to reformulate briefly what a content is about, to plan other actions such as with the tag "todo", or to comment or evaluate the tagged resource. Other studies analyzed the role of annotations in collaborative works and showed that annotations are a privileged means to articulate the communication between actors dealing with or co-authoring a set of common documents (Zacklad *et al.*, 2007; Boujut, 2005; and Koivunen, 2006).

To summarize, tags are a specific type of annotations that can be distinguished from other types of annotations by their minimalist form and simplicity of use that fosters the emergence of folksonomies as most of the tags are short enough to have a chance to be shared by several users[7]. Finally, tags, as already observed in the case of annotations in a more general perspective, may have a handful of different usages that call for a richer model than current tag models.

[6]This problem of ambiguity of tags has been addressed by Passant & Laublet (2008) with MOAT (cf 3.4.4 on page 62).

[7]see (Monnin *et al.*, 2010) for this point

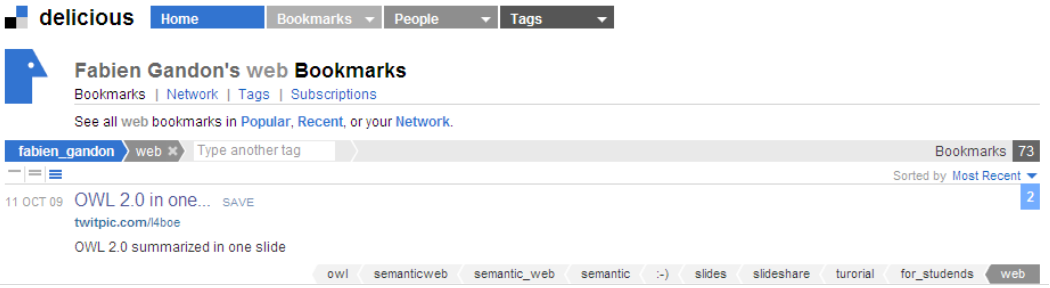


Figure 4.1: Example of a delicious.com bookmark posted by user fabien_gandon on a picture summarizing OWL2.0

4.2.2 Addressing the conceptualization of tags

Tags, in current tag ontologies, are usually modeled with a single "tag class". The uniqueness of tags is thought to be bound to the uniqueness of their label, whereas the same label used as a tag can take on many different uses, and therefore be modeled in many different ways for each act of tagging. Moreover, the relation between the tagged resource and the sign used to tag is often modeled merely with a single property, usually named "has tag" in SCOT , or "tagged" in CommonTag (even if the latter distinguishes between a tag made by the author or by a reader of a Web resource). This choice of modelization has for consequence to overlook the nature of the relation between a tag and the tagged resource.

To illustrate the variety of use of tags, let us take a concrete example of a tagging from delicious.com shown in figure 4.1. In this example the user has tagged a document entitled “Owl 2.0 summarized in one slide”. Each of he tags he has chosen reflects a specific use of tagging that is seldom accounted for in current models. The tag “OWL” is meant to indicate the subject of this information resource. The tags "semanticweb", "semantic_web", "semantic” and “web” are used as topics and synonyms. Tag ":-)" refers probably to the pun of the title and consists in an iconic sign serving a very particular use of tags. Tag “slides” comes to indicate the media or support of the tagged document, whereas “slideshare” tells about its container, and “tutorial” about its genre. Finally “for_students” is very similar to some machine tags, i.e. tags that is meant to be recognized by machines (although in this case, the syntax does not follow Flickr API after which the expression was coined) while still being readable by humans, indicating the target audience. This example shows the variety of relations that exist between the tags and the resources, and this variety is often conflated in current models as a single property.

Kim *et al.* (2008a) proposed a review of current ontologies aimed at modeling tagging and folksonomies, and compare them with regards to their ability (1) to represent tagging, as an individual act involving a user, a tagged resource, and a tag, and (2) the features of folksonomies (such as their container, the co-occurrence between tags, etc.). Thus, Kim *et al.* (2008a) compare tagging models according to their coverage of the wealth of data pertaining to folksonomies but do not re-

ally discuss the conceptualization of the tag itself. The example mentioned above clearly shows a wealth of dimensions embedded in tagging that still remains to be addressed and modeled. This is what the NiceTag framework is about.

4.2.3 Modeling tag assignments with named graphs

The goal of the NiceTag ontology is to allow for modeling tag assignments, or tag actions, without being bound to a unique model of the sign used to tag. To be able to describe tags in the most flexible manner, we propose to consider them primarily as a link between a tagged resource and a sign used to tag, which can take on many different forms and conceptualizations (an image, a literal, an ontology concept, etc.). Regarding the model of tagged resources, Halpin & Presutti (2009) addressed the problem of the "identity crisis" of the Semantic Web, which stems from the fuzziness around the notion of *resource* on the web and its relation to URIs. They proposed the IRW ontology [8] for solving the identity crisis of resources on the Web. Their model is particularly useful to distinguish between taggings of *non-information resources*, as when tagging the Eiffel Tower itself, *i.e.* the physical object, even when doing so through a web page, from taggings of *information resources*, as when tagging a web page about the Eiffel tower. Regarding the sign used to tag, it can be modeled with all the other currently available models of tags such as SCOT, NAO, Newmann's Tag Ontology, or CommonTag, and can also be based on thesauri models (such as SKOS) or concepts from any domain ontologies.

In their paper, Carroll *et al.* (2005) remarked that RDF does not provide any operational means, apart from reification, for making statements about graphs and relations between graphs. As a solution to overcome this limitation, they proposed Named Graphs in RDF to allow publishers to communicate assertional intent and to sign their assertions. The fact that Named Graphs were designed to embody social acts with some record clearly resonates with the scenarios of social tagging.

To model tag actions we defined a subclass of named graphs (modeled as `rdfg:Graph` by Carroll *et al.* (2005)) called `nicetag:TagAction` which embodies one single act of tagging (see figures 4.2 and 4.3). The triples contained in the named graph represent the link, modeled with the property `nicetag:isRelatedTo`, between an instance of the class `nicetag:TaggedResource` and a sign modeled as an instance of `rdfs:Resource`. Starting from this point, our model is able to serve as a pivot-model as the signs used to tag can be modeled with all the other currently available models of tags (see section 4.2.7 for some examples).

More importantly, our paradigm opens up new perspectives on modeling tags by providing for three degrees of freedom: (1) the model of the tagged resource, which can be extended with subclasses of the class `irw:Resource` (to which our class `nicetag:TaggedResource` is an equivalent) to overcome the identity crisis related issues (Halpin & Presutti, 2009); (2) the modeling choice of the sign used to tag is let free; and (3) the relation between the tagged resource and the sign

[8] http://ontologydesignpatterns.org/ont/web/irw.owl

Figure 4.2: TagAction instances are declared as named graphs

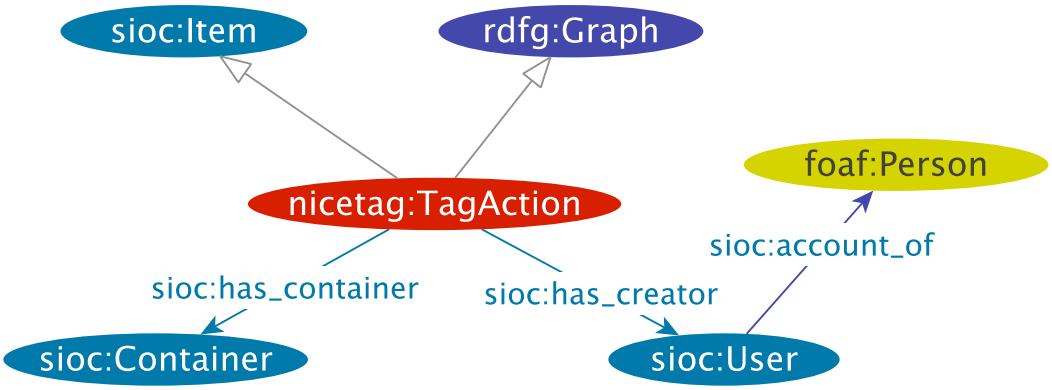


Figure 4.3: TagAction class and its relation to other ontologies

allows for a fine grained account of the semiotics of tagging. Furthermore, the possibilities to capture the intention of a tag action are twofold. (3.a) The relation `nicetag:isRelatedTo` can be readily declined to faithfully model all the possible uses of tags already described in academic literature (see details in subsection 4.2.4). (3.b) The type of tag action can be specified with extra subclasses to capture other dimensions of the tag (as described in subsection 4.2.5).

Finally, the TagAction class is declared as a subclass of `sioc:Item` in order to account for the shareable nature of tags, which can be seen as some sort of post. This, in turn, makes it possible to describe the place where tag actions are stored with the property `sioc:has_container`, and the account (`sioc:User`) of the user (`foaf:Person`) of the tag with `sioc:has_creator`.

4.2.4 Modeling tag usages

Some current models of tags, as MOAT (Passant & Laublet, 2008) *e.g.*, allow one to link a tag to a well defined meaning; this relationship helps to face the problem of tags' polysemy and describe precisely the different acceptations a term can have in different contexts and for different communities. However, polysemy is not the only ambiguity of tags: some meaning resides in the (so far implicit) kind of relationship between the resource and the sign. For example, the use of the tag "blog", one of the most popular in delicious.com, can assume at least two different meanings with respect to the same definition of the word "blog": it can mean that a resource is about blogs, or that a resource is a blog. Moreover, some tags are

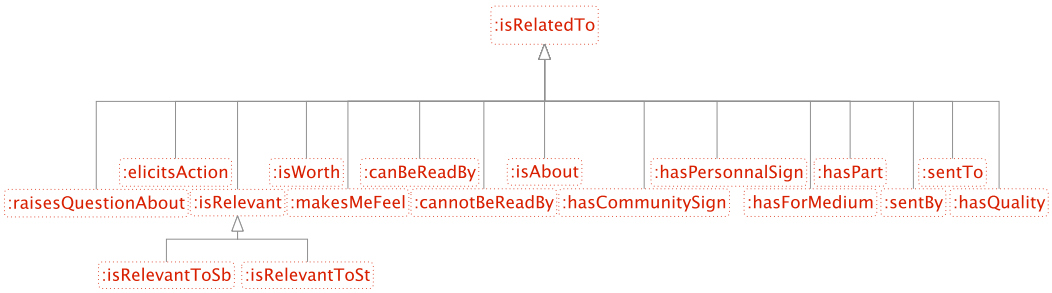


Figure 4.4: `nicetag:isRelatedTo` sub-properties

intended for personal use and to only make sense for the tagger.

Golder & Huberman (2006) proposed 7 classes of tags according to their function. Sen *et al.* (2006) collapsed Golder's tag classes in three broader categories: factual, subjective and personal tags; quantitative studies based on existing popular applications have shown that a significant part of tags tend to fall in the latter classes (Sen *et al.*, 2006; Al-Khalifa & Davis, 2007). Other works proposed a functional classification of tags based on a first distinction between subject related and non-subject related tags, where the latter class can be split into affective, and task and time related tags (Kipp, 2008), whereas subject related tags can be refined into content related and resource related ones (Wolff *et al.*, 2008).

Inspired by previous studies, and in particular by Golder & Huberman, we modeled the different possible uses of tags with sub-properties `nicetag:isRelatedTo` (see figure 4.4). The first possible relationship between a sign and a resource is `isAbout`, which represents the most common use of a tag and identifies the topic of the tagged resource. Most tagging models tacitly assume that this is the relation by default. A second subproperty of `:isRelatedTo` is `:hasForMedium`, intended for all cases in which a tag is used to define what a resource is (e.g.: "forum", "video"). Another wholesome property whose virtue is to limit the inadequate use of `:isAbout` is `:isRelevant` (to someone, with its subproperty `:isRelavantToSb`, or to something, with its subproperty `:isRelevantToSt`). A resource might indeed be said to be relevant to my thesis, my studies, etc. without being even remotely about any of these elements. `:makesMeFeel` is a property intended for tags expressing an emotion stirred up by a resource; typical examples are exclamations and smileys (e.g.: "wow!", "^_^"). The property `:hasQuality` can be used to associate a resource with an adjective or with any kind of sign expressing a quality (e.g.: "nice", "bullshit"). `:isWorth` is meant whenever a resource is evaluated, ranked, etc. (e.g.: "nice", "****"). Another distinction in the intended use of a tag is the one represented by the property `:raisesQuestionAbout` used when the label of a tag indicates that a question is being asked. These last three properties show how it is important to distinguish the relation between a tagged resource and a tag since one can use the same *tag* (*e.g.* "nice") with different intentions, meaning in the case of `:hasQuality` that, for instance, the resource *is* nice, in the case of `:isWorth` that the tagger *judges*

the resource to be nice, and in the case of :raisesQuestionAbout that s/he is not sure *whether* this resource is nice. Other popular uses of tags are those we represent with the subproperties :hasPersonalSign, which is intended to fit Golder & Huberman's class *self reference* (like "mystuff") that just make sense for the applier, and :elicitsAction, which is intended to fit Golder & Huberman's class *task organizing* (like "toread") and more generally whenever a resource elicits an action to be performed. To cover collective uses of tags, we introduced the property :hasCommunitySign for collectively approved tags designed to aggregate resources revolving around a shared event, goal or entity known by all the members of a community or audience. For example, we used the tag "#vocampnice-2009" to share resources about the VoCamp[9] where this paper has been conceived. Still covering the non-topic uses of tags, the pair of properties :canBeReadBy / :cannotBeReadBy is a proposal to allow users to give or to deny access rights to some other users simply by indicating their login as the tags. Another purpose that tags may well serve is the ability to point to a fragment of a resource, as segment of a video or a part of an image, and this is represented in our model with the property :hasPart. Finally, we've added the two properties :sentTo and :sentBy to model networking tasks, as when a tag is used to share a resource with someone else. Some bookmarking systems already have a special syntax for this (e.g.: delicious "for:username" tags).

4.2.5 Typing tag actions

Another way of describing tagging assignments consists in typing the tag action embedded within a named graph by extending the class :TagAction with adequate subclasses. These subclasses can help distinguish, for instance (see figure 4.5), tagging performed automatically by machines (:AutoTagAction) from tagging performed manually by humans (:ManualTagAction). They can also help in accounting for the way in which tags are expressed. The :WebConceptTagAction would be used when signs are computer processable by design, like URIs in MOAT and CommonTag. We intentionally add "by design" because a URI acting as a MOAT "meaning" would be a WebConcept, whose meaning is sometimes construable by a human (a Dbpedia URI) sometimes not (a Geoname one). A MOAT tagging can also be typed with a subclass of the :DisambiguatedTagAction named :Polysemy, as one of the purpose of MOAT is to help disambiguating taggings involving tags that may have different meanings (as "paris" used for the city in France, and "paris" used for thee city in Texas, USA). Systematic polysemy (subclass :SystematicPolysemy) is a more subtle case of ambiguity where a tag refer to the same entity, like "rabbit" for the animal, but with different intended meaning, as when using "rabbit" for the animal's fur or for the animal's meat. Some tags follow a particular syntax(:SyntacticTagAction), and this category includes two more specific types. :MachineTagAction suits tagging involving machine tags, that

[9] http://vocamp.org/wiki/VoCampNiceSeptember2009

is, tags decomposed in three elements that follow a particular syntax that make them processable by the Flickr API[10]. `N-TupleTagAction` could be used with n-tuple tags, that is, tags neither conforming to the syntax used by machine tags nor used on websites that employ it. Another important distinction deals with the status of the author of the tag regarding the authorship of the tagged resource. The `:OwnerTagAction` is used to describe an act of tagging performed by the author of the tagged resource, and the `:VisitorTagAction` is used to describe an act of tagging performed by a person who browsed the Web representation corresponding to the tagged resource.

Tag actions can also be considered as social actions (Reinach, 1983)–or more precisely *speech* acts– mediated through a technical means, the Web. In order to account for the nature of speech acts of tags, we proposed a series of subclasses of the `:TagAction` class. `:Assert` correspond to the broadest class of speech acts that can be accomplished through tagging, and it describes the action that is performed with a tag whenever it is used to assert anything about a resource. Other subclasses can be used to highlight some pragmatic aspects, in the linguistic sense, when a tag action is meant to express feelings (`:ExpressFeelings`), to ask a question (`:Ask`), or to give an evaluation (`:Evaluate`). Then some other subclasses accounts for the specificity of some social acts that correspond to some uses observed or made possible within social tagging web platforms. `:GiveAccessRights` describes the action that is performed with a tag whenever it is used to define to whom access rights to a resource are granted or denied. In the same trend, `:Share` describes the action that is performed with a tag whenever it is used to share the representation of a web resource on various services - Twitter or Delicious for instance - with the owner of a `sioc:UserAccount`[11]. Some other types of tag actions are clearly aimed at aggregating resources under a collectively defined tag (`:Aggregate`). Some other possible uses of tags are already met in current platforms. On YouTube for instance, users now have the possibility to isolate media fragments of videos at will in order to contextualize their comments by pointing at a specific part of a resource, and this type of action (`:Point`) could also be performed with a tag. Finally, tag actions may also well serve organizational purposes by describing a task awaiting performance (`:SetTask`).

4.2.6 Using RDF/XML Source declaration to implement and use named graphs

In SPARQL when querying a collection of graphs, the GRAPH keyword is used to match patterns against named graphs. However the RDF data model focuses on expressing triples with a subject, a predicate, and an object and neither it nor its RDF/XML syntax provide a mechanism to specify the source of each triple. A typical means proposed in the W3C Member Submission "RDF/XML Source Dec-

[10] http://www.flickr.com/groups/api/discuss/72157594497877875/

[11] not necessarily a `foaf:Person` as it might be either a bot, a person or an institution whose representatives may well vary over time

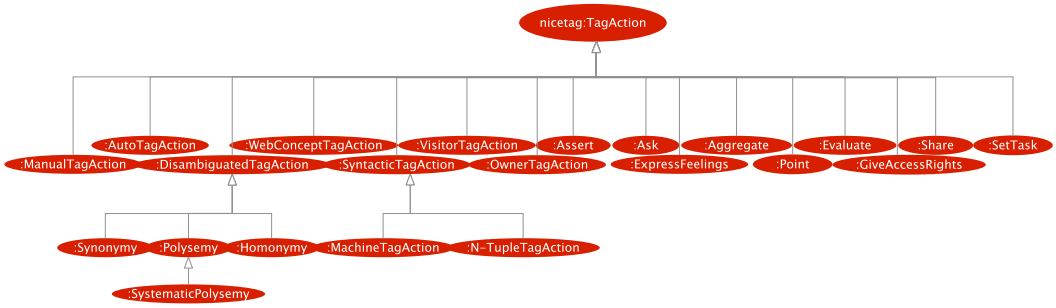


Figure 4.5: `nicetag:TagAction` subclasses

laration" (Gandon *et al.*, 2007) is an XML syntax to associate to the triples encoded in RDF/XML an IRI specifying their origin ; it uses a single attribute to specify for these triples represented in RDF/XML the source they should be attached to. The IRI of the source of a triple is:

1. the source IRI specified by a cos:graph attribute on the XML element encoding this triple, if one exists, otherwise

2. the source IRI of the element's parent element (obtained following recursively the same rules), otherwise

3. the base IRI of the document.

The scope of a source declaration extends from the beginning of the start-element in which it appears to the end of the corresponding end-element, excluding the scope of any inner source declarations. Such a source declaration applies to all elements and attributes within its scope. If no source is specified, the URL of the RDF/XML document is used as a default source. Only one source can be declared as attribute of a single element.

The example in listing 4.1 shows how this applies to declare a tag as a named graph. Line 3 corresponds to the namespace needed for the source declaration. Line 5 declares the tagging of the resource `www.yesand.com`. Lines 6-7 declare the tag as a graph named `http://mysocialsi.te/tag#7182904` and link the resource with the `:isAbout` relation to the tag "improvisation". Lines 9-12 reuse the name of the graph to qualify the tag as a tag created manually by "Fabien Gandon" the 7th of October 2009.

Loading this RDF in a compliant triple store one can then run SPARQL queries like the one in listing 4.2, using, *e.g.*, the Corese RDF engine [12]. Line 2 searches for named graphs and the triples they contain. Line 3 enforces these graphs to be manually generated tags.

[12] http://www-sop.inria.fr/edelweiss/wiki/wakka.php?wiki=Corese

Listing 4.1: Declaration of a tag as a named graph using RDF/XML

```
<rdf:RDF xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cos="http://www.inria.fr/acacia/corese#"
  xmlns:nicetag="http://ns.inria.fr/nicetag/2009/09/25/voc#">
  <nicetag:TaggedResource rdf:about="http://www.yesand.com/"
    cos:graph="http://mysocialsi.te/tag#7182904">
    <nicetag:isAbout>improvisation</nicetag:isAbout>
  </nicetag:TaggedResource>
  <nicetag:ManualTagAction rdf:about="http://mysocialsi.te/tag
     #7182904">
    <dc:creator>Fabien Gandon</dc:creator>
    <dc:date>2009-10-07T19:20:30.45+01:00</dc:date>
  </nicetag:ManualTag>
</rdf:RDF>
```

Listing 4.2: SPARQL query to retrieve tags declared as named graphs

```
SELECT ?t ?a ?g WHERE {
  GRAPH ?tag { ?t ?a ?g }
  ?tag rdf:type nt:ManualTagAction }
```

4.2.7 Examples of Tags

Our model of tag and tagging consists mostly in a link between a tagged resource and a sign used to tag that can be expressed in many different flavors. In figure 4.6 we show some examples of tagging assignments expressed with our model. Tag actions are declared as named graphs as explained in section 4.2.3 and are depicted by a red dotted ellipse surrounding the triples contained in them. Hence, each ellipse represent a tag action and we have adopted a color-code to distinguish the different ontologies we integrated in these examples. Then each tag action is typed with `nicetag:ManualTagAction`, a subclass of `nicetag:TagAction`, since our examples have been taken from actual taggings created manually by two different delicious.com users.

A sign used to tag can be a mere character string, such as ":-)" (in the example using the property `:makesMeFeel`), or it can also be modeled with instances of the Tag class from Common Tag, SCOT, MOAT, or an instance of any concept from already existing ontology; in a word, any `rdfs:Resource` reachable on the Web. Below we illustrate this by going through several examples of tagging using different tag models.

Using CommonTag (ctag) This example uses the tag "semanticweb" and the relation `nicetag:isAbout` to indicate the topic of the tagged resource. The `ctag:Tag` used as a sign is not assigned any URI (*i.e.* it is a blank node) but points to a literal node used as the label with `ctag:label`, and, with `ctag:means`, to a freebase URI

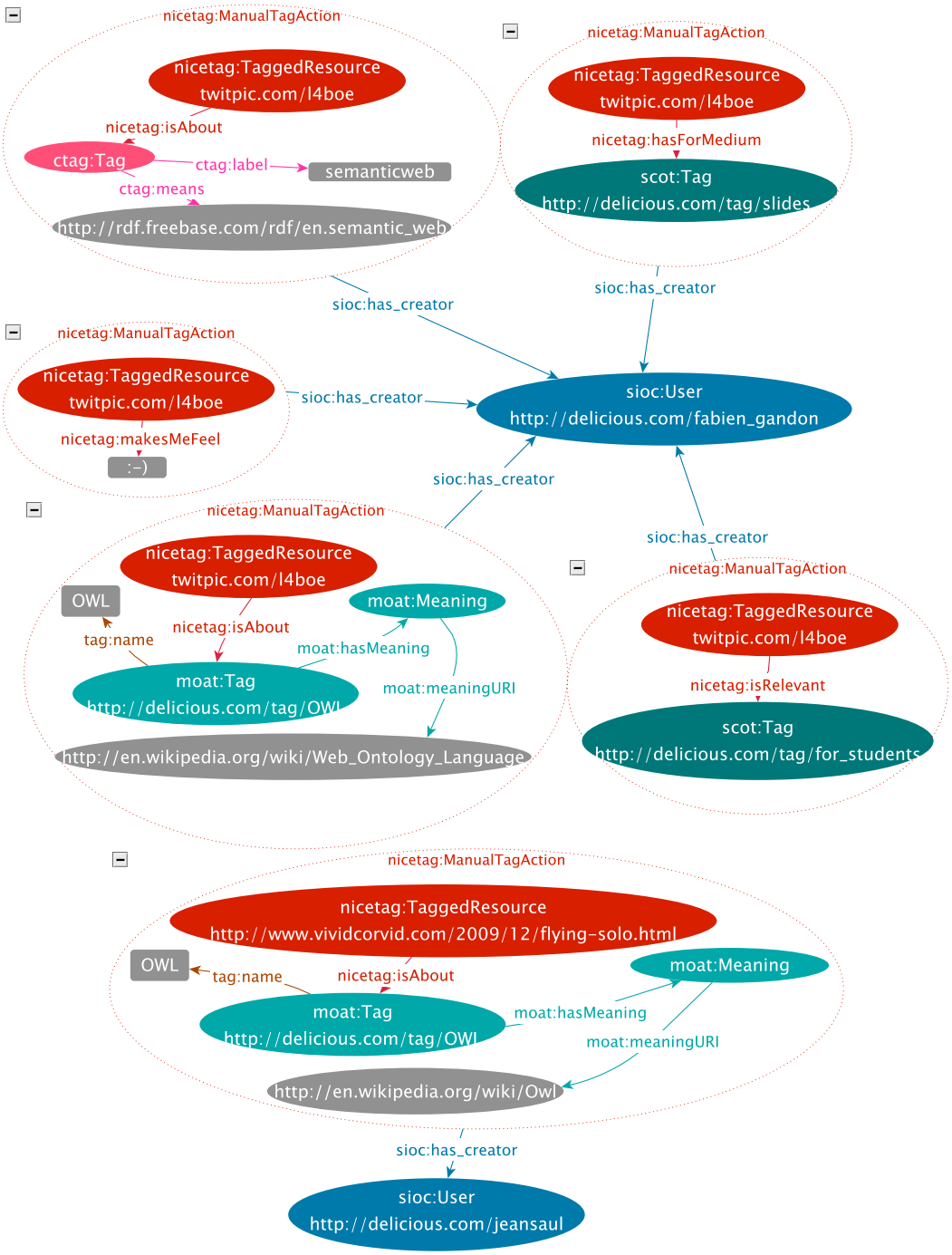


Figure 4.6: Examples of tagging actions expressed with NiceTag and using various models of tags (MOAT in light green, SCOT in dark green, CommonTag in pink, SIOC in blue, and NiceTag in red)

Listing 4.3: SPARQL query to retrieve tag actions by specifying a MOAT meaning for the tag label "OWL"

```
1 SELECT *
2 WHERE {
3   GRAPH ?tagaction {
4     ?doc  nicetag:isAbout ?tag
5     ?tag tag:name  'OWL'
6     ?tag moat:hasMeaning ?m
7     ?m moat:meaningURI <http://en.wikipedia.org/wiki/Owl>
8   }
9 }
```

which defines the intended meaning of the tag.

Using SCOT SCOT is used with the example tag "slides" in conjunction with the `nicetag:hasForMedium` relation indicating the medium of the tagged resource, and also with the tag "for_students" with the `nicetag:isRelevant` relation indicating the targeted audience of the tagged resource (note that if the tag would correspond to an actual user account's login we could have use instead the property `nicetag:isRelevantToSb`) .

Using MOAT Figure 4.6 shows two examples using MOAT with a tag labeled "OWL". The first is used to tag a slide about OWL2.0, and thus refers to OWL as the Web Ontology Language. In this case the tag "OWL" is disambiguated by connecting it to a `moat:Meaning` that is linked to a meaning URI corresponding to the Wikipedia article page that fit the intended meaning. The second example shows a tagging of another resource with the same tag label "OWL" but referring to the animal "OWL". In NiceTag, the locality of the meaning is preserved thanks to the features of named graph. Indeed, in MOAT the meaning URI is attached to a `tag:RestrictedTagging` (according to Newman's TagOntology of Newman *et al.*, 2005), and named graphs in NiceTag have the similar virtue of properly identifying tags as a link between one resource, one sign, and one user. The idea of MOAT is that, even though a tag's label may have several meanings, the disambiguation of a tag action is guarantied by the link between a meaning URI and a `tag:RestrictedTagging`. A similar principle is made possible in NiceTag by encapsulating the tag's label and the meaning URI within the same named graph that identifies the tag action. This is illustrated by the SPARQL query shown in listing 4.3 that allows retrieving a tag action involving a tag labeled "OWL" (lines 3-5) whose meaning conforms to the MOAT meaning URI given as a parameter (lines 6-7), in this case corresponding to the animal *owl*.

The flexibility on the type of signs inherent to NiceTag allows retrieving in a single query all taggings, regardless the model used to describe the sign used to tag. For instance, one can write the SPARQL query shown in listing 4.4 that allows

Listing 4.4: SPARQL query to retrieve tag actions across different tag models

```
1 SELECT * WHERE {
2   GRAPH ?tagaction {?resource nicetag:isRelatedTo ?sign}
3   OPTIONAL{
4     ?sign rdf:type ?signtype.
5     ?sign rdfs:label ?signlabel.}
6   ?resource rdf:type ?resourcetype.
7   ?tagaction rdf:type nicetag:TagAction}
```

retrieving tag actions across different tag models. Line 2 shows that thanks to the inference mechanism, we can retrieve all types of tagging relations expressed with `nicetag:isRelatedTo` and its subproperties. Lines 3-5 show with the OPTIONAL assertion that our model is able to retrieve both typed and untyped signs used to tag, thus retrieving literal tag nodes as well as any type of tag nodes. The same holds for the subtypes of `nicetag:TagAction` (line 7).

4.2.8 Temporary conclusion

The NiceTag model set up the basis of a tagging system by allowing representing tagging assignment in a flexible manner. An important feature for our study is the ability to use different models in the side of the sign used to tag. Indeed, in our scenario, tagging assignments can be made with free tags, but also with controlled tags, and we also envision the possibility to tag directly with concepts from thesauri. Thus, we need to be able to deal with different models on the side of the sign used to tag.

Moreover, the NiceTag model allows to account for the specific relations between a resource and a tag through different subproperties of `nicetag:isRelatedTo`. Indeed, some tags may have purely personal uses (such as the tag "todo") and should not be included in the process of semantic enrichment for which we should favor tags describing topics. Thus, the NiceTag model allows us to filter out the most relevant tags for the process of semantic enrichment of folksonomies that we present in the next section.

4.3 Semantic enrichment of folksonomy lifecycle

4.3.1 Enriching taggings assignments and folksonomies

The NiceTag ontology allows to describe in a flexible yet precise manner the tagging assignments that feed the folksonomy but is not aimed at semantically structuring tags at the level of the folksonomy, *i.e.* it does not aim at organizing tags similarly to thesauri concepts. The problem of the lack of semantics of tags has been addressed by Passant & Laublet (2008) with MOAT, which proposed anchoring the local meaning of tags to instances of the class `RestrictedTagging` from the

TagOntology of Newman *et al.* (2005) that identifies singled out tag actions similarly to NiceTag. However, this solution requires the users to choose a meaning URI from large databases for each tag assignment they make. Regarding this scenario, the NiceTag framework, while still being compatible with MOAT, proposes introducing little steps of semantics in existing interfaces, according to the taggers' needs, thus enriching current tagging systems and keeping their essential simplicity. A few checkboxes about the function of a tag, to let the tagger specify, for example, if a sign is to be considered as a topic or as the genre of the tagged resource, can be integrated in a simple interface, providing a significant added value without overloading users. As a result, in this minimal scenario, *i.e.* NiceTag without MOAT, we obtain tags that are disambiguated regarding their function and use, but that are not yet fully semantically enriched.

To take a step further in the semantic enrichment of folksonomies, we propose to semantically structure tags, or more precisely signs used to tag in NiceTag terminology, relatively to each other instead of linking each tag assignment to a well defined meaning. In order to take into account the different meanings of a tag for different users, we propose in addition to support the multiple points of view that may arise between users. To this regard, we assume that a user will use a tag with the same definition in mind each time, even though he can use it for different purposes that are captured thanks to NiceTag. If someone uses the tag "paris" for the city in France, we assume indeed that he will spell it differently if he happens to tag a resource about Paris, Texas, USA. The semantic structuring we envision consists in stating semantic relationships between tags that can be found in thesauri for instance, such as *broader/ narrower* to state relative levels of generality, *spelling variant* to state that two tags are equivalent in meaning but spelled differently, or *related* to state that two tags are related in some ways.

Structuring semantically tags at the level of the folksonomy presents several advantages. First, it allows for the factorization of each single contribution since tags and their semantic relations to other tags are exploitable by all users. Then, the automatization of the process of finding semantic links between tags is easier than the automatic mapping of a tagging assignment to a URI, as we have seen in chapter 3 that several methods can be used to extract the emergent semantics of tags from folksonomies. Finally, it is possible to synchronize or exploit the semantic structuring of the folksonomy with the elaboration of a thesaurus or lightweight ontology that benefits from the richness of folksonomies. And in this case, we can also take the benefit of tasks that are already achieved by members of the community or by administrators, who may possibly already clean up tags or merge them into similar categories; the monitoring of the enrichment of folksonomies can be directly exploited for the maintenance of knowledge based systems.

To summarize our position regarding the semantic enrichment of tags and folksonomies, we propose the following:

1. capturing lightweight semantics about each tag assignment thanks to NiceTag that allows to account for different uses and forms of tag actions, even

if the sign used to tag has the same meaning in the MOAT's sense.

2. semantically structuring tags with thesauri-like relationships. Tags are taken here at the level of the folksonomy, *i.e.* tags are identified by the feature they share across users and resources, that is to say, their label.

An important thing to note is that our approach is still compatible with the idea of linking tags to unambiguous meanings, as we have shown it with NiceTag whose flexibility makes this possible. The semantic structuring of tags can even helps find ambiguous tags, since these tags are likely to have several other tags as broader tags. For example, the tag "paris" in this scenario may have the tags "USA" and "France" as broader tags, and we can thereafter use this information to ask users of the tag "paris" to provide the meaning they intended in the first place.

4.3.2 Scenario-based analysis for combining machine and human participation in a coherent socio-technical tagging application

A generic method to semantically enrich all types of folksonomies in a fully automatic manner seems out of reach today. We believe that significant progress can be achieved by carefully analyzing the usages of the target communities of a system. Indeed, one may take advantage of the tasks already achieved by users to capture knowledge as a side effect of their daily activity. We conducted such an analysis in one of our target community, the Ademe agency.

Figure 4.7 gives an overview of the different types of users in the Ademe scenario and their role regarding the contribution to the folksonomy. The archivists are in charge of the centralization and the indexing of the documents at Ademe. This indexing is made with a controlled folksonomy in which tags are carefully chosen by the archivists. Experts of Ademe produce reports and internal documents for which they can suggest indexing key words. On the other side, the Ademe agency is currently opening to the public a portion of its documents that external users may tag via a dedicated service. The controlled folksonomy is flat for the moment, but the archivists seek to structure it and enrich it with new terms so as to be able to offer richer search results as well as thematic navigation capabilities within their corpus. To do so, they need contributions in both new tags and semantic structuring from the experts of Ademe and the public.

Regarding the semantic enrichment of the folksonomy, we can take advantage of the time experts take to submit their reports and adequate keywords. Some members of the public may also be keen on providing for contributions in their field of expertise or interest in order to gain access more easily to the documents of Ademe they are looking for. Finally, as the archivists centralize and maintain a controlled vocabulary, their current activity is already in line with the task of monitoring the folksonomy enrichment.

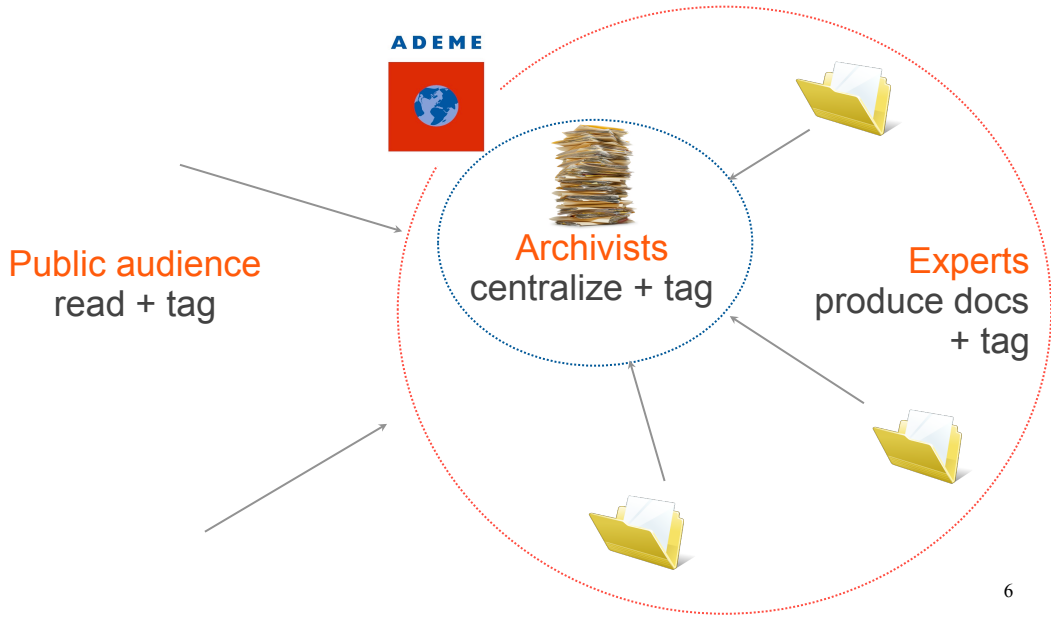


Figure 4.7: Ademe scenario

4.3.3 Folksonomy enrichment life-cycle

In addition to setting the process of semantic enrichment of the folksonomy within the current activity of the members of our target communities, we also propose exploiting automatic processing of tags. Our approach to semantically enriching folksonomies consists thus in creating a synergistic combination of automatic computation of the emergent tags' semantics, to bootstrap the process, and of users contributions at the lowest possible cost through user friendly interfaces. We propose a socio-technical system that supports conflicting points of view regarding the semantic organization of tags, but also helps online communities to build up a consensual point of view emerging from individual contributions.

Figure 4.8 gives an illustration of the different steps of the folksonomy enrichment cycle that can be decomposed as follows:

1. We start from a "flat" folksonomy, IE. with no semantic relationships between tag. Then, automatic agents perform calculation on tags using methods based on an analysis of the labels of tags and on the network structure of the folksonomy. These agents then add assertions to the triple store stating semantic relations between tags. These computations are done during low activity period of time due to their algorithmic complexity.

2. Members of the community can then contribute through user-friendly interfaces integrated in tools they use daily by suggesting, correcting or validating tag relations. Each user maintains his point of view regarding tag relations, while benefiting also from the points of view from other users.

3. As logical inconsistencies may arise between all users' points of view, another type of automatic agent detects these conflicts and proposes conflict resolutions. The statements they proposed are exploited firstly to avoid the noise that may hinder the use of our system when, for instance, several different relations are stated about the same pair of tags.

4. The statements from the conflict solver agent are also used to help a referent user (the archivists in the Ademe scenario) 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 strategies is applied to exploit these points of view to offer a coherent navigation to all users.

6. Then, another cycle restarts with automatic processing in order to take into account the new tags that are added to the folksonomy.

Let us now illustrate the lifecycle with a concrete example before we conclude this chapter. John and Paul are two users of our system. While they browse their bookmarks thanks to tag-based search, they are suggested semantically linked tags. For instance, the system has found that the tag "pollution" and the tag "co2" were *related*. John is not really an expert in environmental issues, and he approves this semantic link since it suits its use. Indeed, if the tag is merely *related*, it will be suggested by the system to broaden the search, but will not be included in the results, and if a tag is a *spelling variant* or *narrower* than the searched-for tag, it will be included in the results. Paul, on the contrary, is concerned with environmental issues, and according to him, the tag "co2" should be placed as narrower tag than "pollution" since he believes that "co2" is a type of pollution. Therefore, when he searches for the tag "pollution" he wants the resources tagged with "co2" to be also included in the results. We thus have a case of conflicts between user John and Paul since they each approve a different relation for the same pair of tags "co2" and "pollution". This conflict is then detecting and since there is, so far, no consensus as only two users expressed themselves for this pair of tags, the system propose to stay with the relation *related* as a compromise. However, the system is able to support diverging points of view in the sense that Paul will be able to keep the relation he chose. After that, the team of Ademe's archivists, our "referent user", has a meeting to maintain the global structuring of the folksonomy. They arbitrate the conflict, and finally opt to place the tag "co2" as narrower than "pollution" in order to account for the main field of expertise of Ademe. In the end, we obtain a structured folksonomy in which diverging points of view coexists, while a global and coherent structuring is maintained by a referent user.

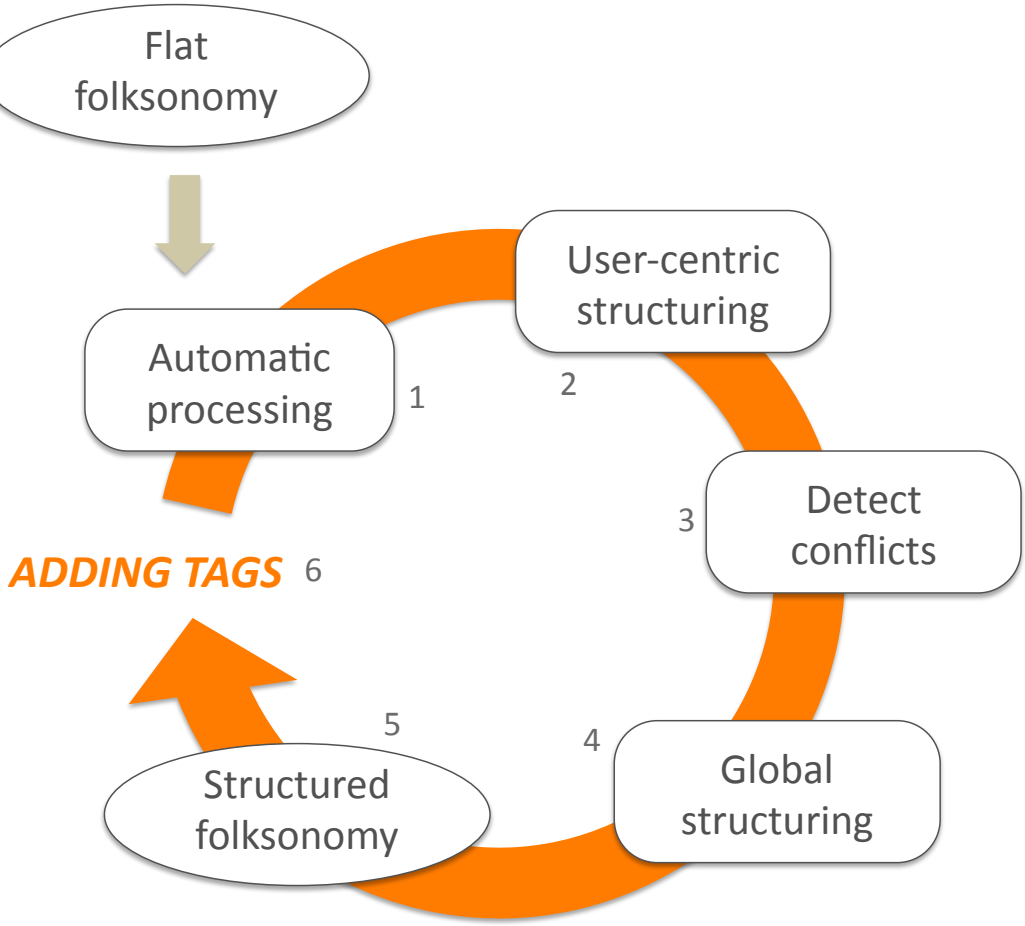


Figure 4.8: Folksonomy enrichment lifecycle

4.4 Conclusion

In this chapter we have presented the NiceTag model that consists in representing tag actions primarily as a link (via the property `nicetag:isRelatedTo`) between a tagged resource and a sign used to tag. This link is then encapsulated within a named graph, thus providing for a way to identify and type this tag action with the class `nicetag:TagAction`. This paradigm allows describing multiple dimensions of tagging through the different subproperties of `nicetag:isRelatedTo`, to account for the different uses of tags (from personal task organization to the description of the topic), and through the different subclasses of `nicetag:TagAction`, in order to account for the different natures of tags as social acts mediated through a technical means, the Web. Its flexibility makes it eligible to serve as a pivot between current tagging models that can be easily integrated on the side of the sign used to tag. This way, thanks to the use of the RDF/XML Source Declaration syntax to assign a URI to a tag action, we obtain a full expressive richness to represent tags from a multiplicity of facets, avoiding the burden of RDF reification.

The benefits of using NiceTag as the foundation of our tagging-based system is twofold. First, by letting a degree of freedom for the model of the sign used to tag, we are able to account for the variety of the vocabulary that can be integrated in our system. For instance in the Ademe's scenario, tags proposed by regular users can be modeled with the SCOT's tag class, tags provided by the archivists can be modeled with a custom class that account for their nature of controlled vocabulary, and tags coming from external thesauri relevant to Ademe's field of knowledge can be modeled with the SKOS schema. Second, by making it possible to precisely define the use of a tag, we are able to filter out tags that would be meaningful only to the tagger (as in the case of "task organization" tags such as "toread") and thus irrelevant for the process of folksonomy enrichment.

The folksonomy enrichment can be seen as a complement to the semantic enrichment of tagging assignment made possible with NiceTag. Indeed, NiceTag allows specifying the use of the tags while the semantic enrichment at the level of the folksonomy allows structuring tags relatively to each other with thesauri-like relationships such as *broader/narrower*, *spelling variants*, or *related*.

The specificity of our approach to folksonomy enrichment lies in a synergistic combination of users contribution and automatic processing, and the support of multiple points of view. To achieve this vision, we ground the design of the life cycle of the enriched folksonomy within an analysis of our target communities' activity to integrate the process as seamlessly as possible within their everyday tasks. Indeed, members of a community usually already perform tasks that should be taken as opportunities to contribute to the enrichment of the folksonomy they share. We have conducted such an analysis in the Ademe agency whose archivists seek to enrich the controlled but flat vocabulary they use to index Ademe's documents with the contributions of both experts and external members. In this scenario, our approach allows each individual contributor to structure tags according to his own point of view thanks to user-friendly interfaces integrated

within every day tools. A first type of automatic agent helps users in this task by suggesting semantically related tags computed thanks to an analysis of the tags and the structure of the folksonomy. Furthermore, each user is able to maintain his point of view while still benefiting from others' point of view. In order to help a referent user (administrators or to build a global and consensual structuring of the tags, a second type of automatic agent detects conflicts arising between user's contributions and proposes solutions. The solutions of the conflict solver serve as suggestions made to the referent user but can be seen also as temporary solutions used until the referent user chooses one. At this step of the folksonomy enrichment cycle, we have a structured folksonomy in which several points of view coexist in addition to a global and logically consistent point of view maintained by a referent user that is assisted by the conflict solver.

In the next chapters of this thesis we are going to give the details for each module of our approach to folksonomy enrichment. In chapter 5 we present in details the three types of methods used to infer semantic relationships between tags and to bootstrap the enrichment process. Chapter 6 covers the model we propose to support diverging points of view regarding individual contributions. This chapter also presents the user-friendly interface that we have implemented to capture these individual contributions and that is integrated within a tool for navigating the folksonomy. Chapter 7 then covers in details our approach to detect conflicts that may arise between the individual contributions in order (1) to propose temporal resolutions that allow for a coherent experience for each user, and (2) to help a referent user maintaining a global and consensual point of view. In chapter 8 we present our implementation of a tag server that includes tagging and semantic enrichment functionalities, as well as the computation of semantic relationships. To this regard, this chapter details our dataset and the results of the computations we obtained. Chapter 9 concludes this thesis and gives some perspectives for future works.

CHAPTER 5

Combining methods to infer tag semantics

Abstract. Several types of computational methods can be applied to folksonomies in order to retrieve semantic relationships between tags. This chapter presents our approach to combine three different types of automatic processing to bootstrap the process of semantic enrichment of folksonomies. The first method we propose is a custom combination of string-based metrics that analyze the labels of tags. This heuristic combination is the result of a systematic benchmark of standard string-based metrics that evaluate their ability to retrieve different types of semantic relations. The second method measures the similarity of tags for the distributional aggregation of tagging data in the Tag-Tag context and allows inferring associative semantic relations, while the third method exploits user-based association rule mining to infer hyponym relations. As a result, these automatic processing methods allow bootstrapping the process of semantic enrichment of folksonomies by proposing a set of relations between tags, or between tags and concepts from thesauri.

Contents

5.1 Introduction

String-based similarity metrics were initially designed to match very similar string patterns, with some important applications in the analysis of sequences of genomes. In the scope of folksonomies enrichment, such metrics are typically used to map tags with ontology concepts, or to merge together spelling variant tags. In this chapter we present the experiment we conducted to evaluate the performance of string-based metrics in detecting other types of semantic relationships between tags. For instance, it is obvious that the tag "pollution" and the tag "pollutant" are related, and our goal in this study was to systematically analyze current standard string-based metrics in order to evaluate and quantify their ability to detect such cases of tag similarity. After this benchmark that we conducted using a sample from Ademe's dataset, we propose a heuristic method to combine them efficiently in order to retrieve three types of semantic relationships, namely *associative*, *hierarchical,* and *mapping relations*. The advantage of such a method is that it does not depend on the structure of the folksonomies, and this has two interesting consequences:

(i) It is possible to semantically link tags across different folksonomies or with other resources (*e.g.*, a lexicon of the organization) as this is the case at Ademe for instance. Indeed, as we are going to see it in details in chapter 8, the dataset of the Ademe agency on which we have applied the methods presented in this chapter is made of three datasets: (a) taggings extracted from delicious.com and dealing with Ademe's content and activity, (b) tags extracted from an internal database and associated to Ademe's funded PhD projects, and (c) *controlled* tags used by the archivists to index Ademe's documents and internal reports. Thus, our heuristic string-based method can operate across the three sub-folksonomies of Ademe's dataset.

(ii) This type of method is incremental since, when new tags are added, we do not need to update the similarity values of all the pairs of tags of the folksonomy, as in the case of structure-based methods, but we only have to compute the string-based similarity for the newly added tags with the other tags.

Both other methods that we present exploit state of the art algorithms analyzing the structure of folksonomies. The first type we consider is a method based on the cosine similarity measure applied on the distributional aggregation of tagging data in the Tag-Tag context. Indeed, Cattuto *et al.* (2008) showed that this type of similarity measure yielded *associative* relations by comparing the semantics inferred in this way with the taxonomic structure of WordNet. Then we give some details on the second method we considered, which is based on mining association rules within the structure of folksonomies. The association rules we look for in this case is the inclusion of community of users of tags to infer *hierarchical* relations between tags.

This chapter is organized as follows. We recall first the different types of semantic relations that we consider and present the model used to represent them in section 5.2. Section 5.3 is devoted to the string-based method: after having presented the standard string-based metrics considered in our study, we give details on the benchmarks we conducted to select the best metrics for each type of semantic relation, and we conclude this section with the presentation of our heuristic string-based method and some example results. Section 5.4 covers both methods based on the analysis of the structure of the folksonomy for which we give details on the computation steps and some example results, and section 5.5 concludes this chapter.

5.2 Models to represent semantic relations

Before going into details on our methods to automatically infer semantic relations between tags or concepts, let us briefly introduce the different types of relations through the presentation of the models we use to represent them. In our approach, we chose to structure tags in the same fashion concepts in a thesaurus are organized. This has the advantage of limiting the number of possible relations, though providing for a precise structuring.

Our approach is aimed at helping our target communities structure tags relatively to each other, but also structure concepts that they define more precisely, such as in thesaurus, relatively to each other, and, finally, map tags with either equivalent tags or concepts corresponding to the meaning of those tags. Semantic relations we infer can thus link tags with tags, or concepts with concepts, or tags with concepts.

Concepts are modeled with SKOS class `skos:Concept`, while tags are modeled with SCOT class `scot:Tag`, which inherits from `skos:Concept` through the TagOntology (Newman *et al.*, 2005) class `tag:Tag`. Semantic relations come from SKOS and can be thus applied between tags or concepts.

The fact for two concepts to be semantically linked is modeled in SKOS with the property `skos:semanticRelation`. In this chapter, we refer to this type of relation as *semantic link* or *semantically linked* . Then, semantic relations in SKOS can be divided into three more precise categories:

1. **Hierarchical or hyponym relations:** this type of relation helps building a hierarchy of concepts by stating different levels of generality between concepts. According to SKOS reference[1] "A hierarchical link between two concepts indicates that one is in some way more general ("broader") than the other ("narrower")". For example, the tag "pollution" is broader than the tag "soil pollution". In SKOS, these relations can be modelled with `skos:broader` and `skos:narrower` properties. These two properties are used to describe direct hierarchical links and are not transitive. To indicate

[1] http://www.w3.org/TR/skos-reference/#semantic-relations

a transitive hierarchical link, one can use the transitive counterparts, namely `skos:broaderTransitve`, and `skos:narrowerTransitive`. **In this chapter we refer to this relation as *hyponym*.**

2. **Mapping relations:** this type of semantic links corresponds to cases where one wants to map equivalent or similar concepts from different thesauri for instance. The `skos:closeMatch` property is a subproperty of `skos:mappingRelation` and we proposed to use this type of relations to describe the spelling variation between tags (such as between "energy" and "energies"), or to map tags with concepts from thesauri. **In this chapter, we refer to this type of relation as *spelling variant*.**

3. **Associative relation:** this type of relation is meant for concepts that are semantically related but do not share any hierarchical links or that are not equivalent in meaning. This type of relation is modeled with the property `skos:related`, and an example is given by the relation between the concepts "electricity" and "battery". **In this chapter, we refer to this type of relation as *related*.**

Finally, following these definitions, we can state that a pair of *semantically linked* tags, which is not a pair of *hyponym* tags, nor a pair of *spelling variant* tags, can be assumed to be a pair of *related* tags. Indeed, the *related* relation has a relatively loose definition, or at least, less precise characterization than the *hyponym* or *spelling variant* relation.

5.3 Evaluating string based methods

5.3.1 Presentation of the study

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 by Specia & Motta (2007) to group *spelling variant* tags such as "new_york" and "newyork". To go further in the use of this type of method that does not depend on the structure of folksonomies, we conducted a benchmark to evaluate the ability of such metrics to retrieve other types of semantic relations such as *related*, *narrower*, or *broader*.

A first distinction among the different string-based metrics concerns the difference between distance functions and similarity functions. Distance functions associate a real number d to a pair of strings, where the smaller the value of d, the closer the strings. Similarity functions associate a real number s to a pair of strings, where the greater the value of s, the closer the strings. In the SimMetrics[2] package, all measures are implemented so that they can be considered as similarity metrics, even though they can make use of distance functions, like edit distances, to compute a similarity.

[2] http://www.dcs.shef.ac.uk/~sam/stringmetrics.html

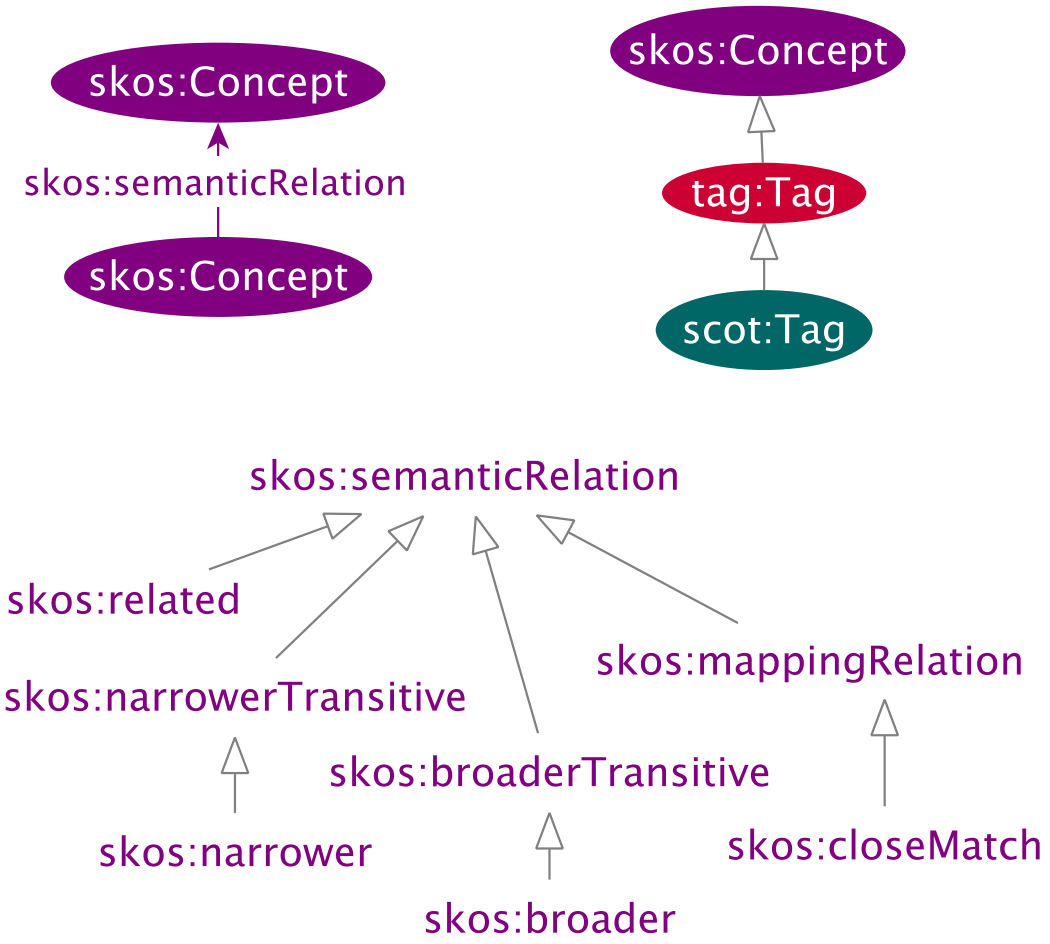


Figure 5.1: Models used to represent semantic relations between concepts or tags: SKOS for thesauri concepts and semantic relations, SCOT, which inherits from Newman's TagOntology's tag class, for tags and spelling variant relation

Below we present the different types of string based metrics implemented in the SimMetrics package. We have compared the similarity metrics implemented in the package SimMetrics, 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.

5.3.1.1 Edit-distance based methods

Levenshtein and Needleman-Wunch metric Edit distances in general consider a set of string operations to turn one string into the other string of the pair to be compared. Such string operations include insertion of a character *c* at position *i, ins(c,i),* substitution of a character *c* by a character *c'* at position *i, sub(c,c',i)*, and deletion of a character *c* at position *i, del(c,i)*. A cost is then attributed to each of these operations. Computing the value of the edit distance between two strings *s* and *t* consists in summing up the costs of the less costly set of operations to turn s into t.

In the Levenshtein distance (Levenshtein, 1966), all operations are assigned a cost of 1, and can be seen as the minimum number of string operations to turn s into t. The Needleman-Wunch distance adds a variable *G* as the cost parameter for a gap, ie. for an insert or deletion operation. The Levenshtein distance can thus be seen as the Needleman-Wunch distance with the parameter G set to 1. The default value for *G* in SimMetrics library for the Needleman-Wunch is 2.0.

SmithWaterman and Smith-Waterman-Gotoh metric The Smith-Waterman metric (Smith & Waterman, 1981) is an extension of the Levenshtein metric that was originally designed to retrieve similar regions between DNA and protein sequences. This method differs from the previous ones by computing the string operation costs as a function of the substring on which the operation is applied. This is achieved by setting specific rules, called a scoring system, to compute the cost matrix which contains the cost value for each operation performed to turn string *s* into string *t* (these matrices have dimensions of $|s| * |t|$). The scoring system introduced by Smith and Waterman has the particularity of making local alignments *(ie* by matching substrings) visible in such cost matrices. The value of the metric is given by the highest value in the cost matrix normalized by the length of the shortest string from the pair to be compared.

The Gotho metric (Gotoh, 1981) is an extension of the Smith-Waterman and introduced an affine model of the gap cost which will vary depending on the length of the gap. The affine model further specifies two types of costs: one which computes a cost corresponding to the start of a gap, and another for the

cost corresponding to the end of a gap. This has the result of favoring cases where there is a little number of big gaps, such as in the pair *"laitage"-"laitier"* which contains one big gap at the end ("lait-age" / "lait-ier") and which scores higher with Smith-Waterman-Gotoh than with Smith-Waterman, over a greater number of small gaps, such as in "reseau intelligent"-"reseaux intelligents" which contains two small gaps ("reseau-" / "reseau-x", and "intelligent-" / "intelligent-s") and which scores higher with Smith-Waterman than with Smith-Waterman-Gotoh.

Monge-Elkan metric The Monge-Elkan approach (Monge & Elkan, 1996) first decomposes strings into substrings so that two strings s and t can be written $s = A_1...A_K$ and $t = B_1...B_L$. Then each substring of both of the compared tags are evaluated using a third party "internal" metric called sim'. By default this metric is the Gotoh distance in the SimMetrics package, and the substrings correspond to tokens delimited by white spaces, so that compound tags, such as "changement climatique", will be decomposed into "changement" and "climatique". The resulting measure is then computed using a recursive matching algorithm. The normalization is done on the length of the first string, making this metric non-symmetric. The Monge-Elkan metric for two strings s and t is given by:

$$MongeElkan(s,t) = \frac{1}{K}\sum_{i=1}^{K}\max_{j=1}^{L} sim'(A_i, B_j)$$

This algorithm has quadratic time complexity and increases greatly the computation time when dealing with a lot of data as in automatic handling of folksonomies, but it presents significant advantages when dealing with compound words.

Jaro and Jaro-Winkler metrics The Jaro metric (Jaro, 1989) aims at treating common spelling deviations by considering the number and order of the common characters between the two strings to be compared. For two strings $s = a_i...a_K$ and $t = b_1...b_L$, a character a_i in s is common with t if there is a character $b_j = a_i$ in t such that $i - H \leq j \leq i + H$ where $H = \frac{\min(|s|,|t|)}{2}$. Now, s' is the set of characters from s which are common with t, and vice-versa for t', and $T_{s',t'}$ denotes the number of transposition to turn s' into t'. The Jaro similarity for s and t is then given by:

$$Jaro(s,t) = \frac{1}{3}(\frac{|s'|}{|s|} + \frac{|t'|}{|t|} + \frac{|s'| - T_{s',t'}}{2|s'|})$$

The Jaro-Winkler extension (Winkler, 1999) computes a weighted Jaro distance to better score string which share a common prefix. The Jaro-Winkler distance is given by:

$$JaroWinkler(s,t) = Jaro(s,t) + prefixLength \cdot PREFIXSCALE \cdot (1 - Jaro(s,t))$$

where *prefixLength* is the length of the common prefix and *PREFIXSCALE* a constant used to control the weights given to common prefixes, which is equal to 0.1

in SimMetrics.

5.3.1.2 Token-based methods

Methods of this kind decompose strings into sets of substrings or words (often called "bag of words"). In the case of tags comparison, these sets are obtained by splitting tags into independent tokens separated by, *e.g.,* a white space or a dash.

Matching Coefficient The matching coefficient simply counts the number of equivalent tokens contained by both strings to be compared. If S and T are two sets of strings, then the matching coefficient is given by:

$$match(S,T) = |S \cap T|$$

Jaccard coefficient This metric, first introduced by Jaccard (1912), is computed as the division of the number of common tokens over the total number of tokens. For two sets of strings S and T :

$$Jaccard(S,T) = \frac{|S \cap T|}{|S \cup T|}$$

Dice's coefficient Similarly to the Jaccard coefficient, the Dice approach considers common substrings. If S and T are two sets of strings, the Dice coefficient is given by:

$$Dice(S,T) = \frac{2 \cdot |S \cap T|}{|S \mid + \mid T|}$$

Overlap coefficient The overlap coefficient is similar to the Dice coefficient but differs slightly by stating that if one of both compared sets of strings S and T is a subset of the other, then the similarity is equal to one. Overlap coefficient is given by:

$$Overlap(S,T) = \frac{|S \cap T|}{\min(|S|,|T|)}$$

5.3.1.3 Token-based methods using vector representations

Some of the token-based methods use vector representations of the strings when comparing two strings s and t. *e.g.,* each token contained in both strings consists in a dimension of the metric space to which the vectors $\overrightarrow{s}$ and $\overrightarrow{t}$ belong. For example, when comparing the strings s = "pollution diffuse" and the string t = "pollution atmosphérique", the dimensions of the vector space V will be (*i*) "pollution", (*ii*) "diffuse", and (*iii*) "atmosphérique", and the vectors will have for coordinates $\overrightarrow{s}(1,1,0)$ and $\overrightarrow{t}(1,0,1)$.

Block (or City-Block) metric If s and t are the two strings to be compared, then the block distance is given by:

$$Block(s,t) = \sum_{i \in V} \mid \overrightarrow{s_i} - \overrightarrow{t_i} \mid$$

where $\overrightarrow{s_i}$ is the coordinate of vector $\overrightarrow{s}$ in the dimension i of the vector space V. In the case of the example where s= "pollution diffuse" and t = "pollution atmosphérique", we have :

$$Block(s,t) = |1-1| + |1-0| + |0-1| = 2$$

The name of this metric comes from the fact that, in 2 dimensions, if we picture the set of points in a grid, then the block distance correspond to the number of edges one has to pass to go from point t to point s. This situation is similar to going from one corner to another corner in a rectilinear city, hence the name "city block"[3].

Euclidean metric This similarity considers both compared strings s and t with the same vector representation as above, and computes the Euclidian distance given by :

$$Euclidean(s,t) = \sqrt{\sum_{i \in V} (\overrightarrow{s_i} - \overrightarrow{t_i})^2}$$

If we compute the Euclidean distance for the same example as above, with the coordinates of strings s and t are $\overrightarrow{s}(1,1,0)$ and $\overrightarrow{t}(1,0,1)$, we have:

$$Euclidean(s,t) = \sqrt{(1-1)^2 + (1-0)^2 + (0-1)^2} = \sqrt{0+1+1} = \sqrt{2} \simeq 1.4$$

If we compare this metric with the Block metric in the 2 dimensions case, this metric gives the length of the segment defined by the two points on a grid.

Cosine similarity This similarity considers the same vector representations of two strings s and t as above and consists in computing the cosine distance between these two vectors given by:

$$cos(s,t) = \frac{\sum_{i \in V} \overrightarrow{s_i} \times \overrightarrow{t_i}}{\sqrt{\sum_{i \in V} \overrightarrow{s_i}^2 \sum_{i \in V} \overrightarrow{t_i}^2}}$$

[3]http://planetmath.org/encyclopedia/CityBlockMetric.html

1	B,P,F,V
2	C,S,K,G,J,Q,X,Z
3	D,T
4	L
5	M,N
6	R

Table 5.1: Mapping of the Soundex codes (1st column) and the corresponding letters.

If we compute the cosine distance for the same example vectors as above for s= "pollution diffuse" and t = "pollution atmosphérique", we have:

$$cos(s,t) = \frac{1 \times 1 + 1 \times 0 + 0 \times 1}{\sqrt{(1^2 + 1^2 + 0^2) \times (1^2 + 0^2 + 1^2)}} = \frac{1}{\sqrt{4}} = 0.5$$

5.3.1.4 Other types of metrics

QGram metric A q-gram of a string consists in a portion of "q" adjacent characters of the string. By sliding a window of size q over the length of the string, we get a series of q-grams for that string. The idea behind the Q-gram similarity measure is to count the number of common q-grams of strings to be compared and to normalize by the total number of q-grams available.

Soundex metric The Soundex approach gives a coarse phonetic index to the strings to be compared. It was designed for matching proper names misspelled. Each term is given a Soundex code which consist in a letter (the first letter of the term) and 3 digits chosen according to the following consonants (vowels are not counted) as given by the table 5.1. If two adjacent consonants are the same, only one number is picked up. If there are less than 3 digits after the conversion, the remaining digits are set to 0. Remaining consonants, once 3 digits are picked up, are omitted. For instance "Robert"and "Rupert" return the same Soundex code "R163"; "Rubin" yields "R150", and "Ashcraft" yields "A261". In the SimMetrics library, a third party metric (the Jaro Winkler by default) is then used to compute the distance between both Soundex codes of the strings to be compared.

5.3.2 Measuring the performance of standard string-based metrics

5.3.2.1 Protocol

An example of previous approaches to compare different string-based methods has been proposed by Cohen *et al.* (2003), who evaluated the relative performances of these metrics when used to match a list of entity names. This problem is closely related to that of finding spelling variant tags, since it consists in matching names in a corpus with named entities from a reference list, the difficulty coming from

the spelling variations that may exist between the corpus and the reference. To compare the different methods used in their experiment, they computed the precision, recall and maximum F_1 measure averaged over the different data sets. To find the best method they plot the recall and precision on the same figure for each value of the threshold used to detect a matching pair, with each point of the curve having as coordinates the value of recall (X axis) and precision (Y axis). The best string-based method according to their study is the Monge-Elkan metric. However they did not evaluated the combination of MongeElkan with other metrics, as we did in our benchmark. Similarly to Cohen *et al.* we used information retrieval indicators such as the precision, recall and F-measure to evaluate the performance of each string based metric.

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). The first three sets contain pairs of tags linked respectively with one of the following semantic relations: *spelling variant*, *hyponym*, and *related*. The fourth set contains pairs of *unrelated* tags. These four sets and the corresponding relations have been validated by one member of the Ademe's archivists team so that it reflects the knowledge of our user's domain.

In our study we have compared the 15 metrics implemented in the SimMetrics package. The choice of this package is motivated by the fact that it is cited in many research works and contains the main state of the art string based methods. Among the 15 metrics that we have compared, the Monge-Elkan metric is a hybrid metric that 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 theses 15 metrics with the Monge-Elkan method, which makes a total of 30 different metrics (in the remaining, these composite metrics are referred to as Monge-Elkan_Soundex for instance when the Soundex metric is used as the Monge-Elkan internal metric).

5.3.2.2 First benchmark

We conducted a first benchmark to evaluate the ability of each metric to retrieve pairs of tags sharing different types of relationships. This benchmark can be seen as an information retrieval problem since, on one side we have 4 sets of pairs of tags for which we know the type of relation they share, and on the other side, we have a set of similarity measures $\{sim\}$ that retrieve a pair of tags (t_1, t_2) when the similarity value for this pair $\sigma_{sim}(t_1, t_2)$ is above a given threshold τ, *ie* when $\sigma_{sim}(t_1, t_2) > \tau$. In this first benchmark, the set of *unrelated* pairs serves as the set of false pairs when retrieved by a similarity metric for all the other three set of semantically linked tags (*i.e.* the sets of *related, spelling variant,* and *hyponym* tags). In this way we evaluate the ability of each metric to retrieve pairs of tags linked with a given semantic relation rather than pairs from the set of unrelated tags. In a first experiment we realized that the similarity values were biased by

stop words, so we removed them before computing the similarity for each tag pair. Then to count the false positive and true positive pairs that were retrieved for a given threshold value τ, $\tau\varepsilon[0,1]$ (because the similarity metrics give a value between 0 and 1, a value of 1 meaning that both tags are most similar), we applied the following rules:

- for each type of relation *rel*, the number of true positives TP is counted by the number of pairs that are retrieved from the set corresponding to the relation *rel*,
- the number of false positives FP is given by the number of pairs retrieved from the set of *unrelated* tags pairs.

To evaluate the performance of each metric in retrieving the correct pairs for each case of semantic relation, we have computed the recall r, and the precision p for different values τ_i of the threshold above which a given tag pair is retrieved, with $\tau_i\varepsilon[0,1]$, $i\varepsilon\mathbb{N}$ and $\tau_{i+1} = \tau_i + 0,01$. At a given threshold τ_i, the value of $p_{rel,sim}(\tau_i)$ and $r_{rel,sim}(\tau_i)$ of the precision and recall for a given semantic relation *rel* and a given similarity metric *sim* are given by the following:

$$\begin{aligned} p_{rel,sim}(\tau_i) &= \frac{|\{relevant\ pairs\}_{rel} \cap \{retrieved\ pairs\}_{sim,\tau_i}|}{|\{retrieved\ pairs\}_{sim,\tau_i}|} \\ &= \frac{TP_{rel,sim}(\tau_i)}{TP_{rel,sim}(\tau_i) + FP_{rel,sim}(\tau_i)} \end{aligned}$$

and

$$\begin{aligned} r_{rel,sim}(\tau_i) &= \frac{|\{relevant\ pairs\}_{rel} \cap \{retrieved\ pairs\}_{sim,\tau_i}|}{|\{relevant\ pairs\}_{rel}|} \\ &= \frac{TP_{rel,sim}(\tau_i)}{|\{relevant\ pairs\}_{rel}|} \end{aligned}$$

with:

$\{relevant\ pairs\}_{rel}$ the number of pairs of tags sharing the relation *rel*,

$\{retrieved\ pairs\}_{sim,\tau_i}$ the number of pairs retrieved by the similarity measure *sim* for the threshold value τ_i

$TP_{rel,sim}(\tau_i)$ the number of true positives retrieved by the similarity *sim* for the threshold value τ_i and relation *rel*

$FP_{rel,sim}(\tau_i)$ the number of false positives retrieved by the similarity *sim* for the threshold value τ_i and relation *rel*.

Then, in order to be able to rank the scores of each metric according to a single value, we computed the weighted harmonic mean $F_\beta(rel, sim, \tau_i)$ for each relation case *rel*, each similarity metric *sim*, and each threshold value τ_i. It is given by :

$$F_\beta(rel, sim, \tau_i) = \frac{(1+\beta^2) \cdot (p_{rel,sim}(\tau_i) \cdot r_{rel,sim}(\tau_i))}{(\beta^2 \cdot p_{rel,sim}(\tau_i) + r_{rel,sim}(\tau_i))}$$

In this study we chose $\beta = 1$ because we wanted to give as much importance to recall as to precision.

In the figures 5.2, 5.3, and 5.4 we report the mean values and deviation of the F_1-measure for the top 10 metrics and for each semantic relation. The results show that the MongeElkan_Soundex metric performed best in each case. 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 that two notions are related rarely translates to some terminological similarities, as *e.g.* "car" and "wheel" are related but don't share any letters. So we can state that this metric is the best we can do with this kind of approaches to retrieve pairs of tags sharing the three types of semantic relations we have tested.

Conclusion of the first benchmark A first conclusion we can draw is that the MongeElkan_Soundex metric can be used to retrieve *semantically linked* tag pairs, *i.e.,* pairs that share one of the semantic relations *broader/narrower* or *spelling variant* or *related*, but conversely we cannot use this metric to differentiate between these three types of relation. Thus, we conducted a second benchmark to find the most discriminative metrics in order to distinguish *spelling variant* and *hyponym* pairs from the pairs that are merely *semantically linked* and retrieved by the MongeElkan_Soundex metric. Our hypothesis is that when two tags of a pair are *semantically linked*, but do not share a *hyponym* nor a spelling *variant* relation, then we consider them to share an associative relation, *i.e.,* we consider them to be *related*, following the SKOS definition of this type of relation.

5.3.2.3 Second benchmark

The goal of this second benchmark was:

1. to verify that the MongeElkan_Soundex metric is the best at retrieving the three types of relation at a time, and
2. to evaluate the ability of string-based metrics to differentiate between the three types of semantic relationships we consider in this study.

Thus, we have applied the same protocol and computed the same performance metrics as in the first benchmark, except that, for the first point, we considered the union of the three sets corresponding to *semantically linked* tags, *i.e.* the sets of *related, spelling variant,* and *hyponym* tags so as to evaluate the ability of each metric to retrieve *semantically linked* tags rather than pairs from the set of *unrelated* tags.

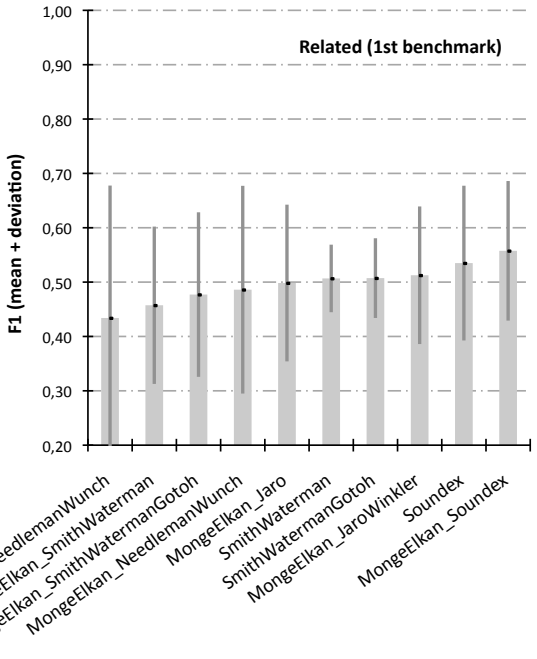


Figure 5.2: Mean values and deviation of the F_1-measure for the related case (1st benchmark)

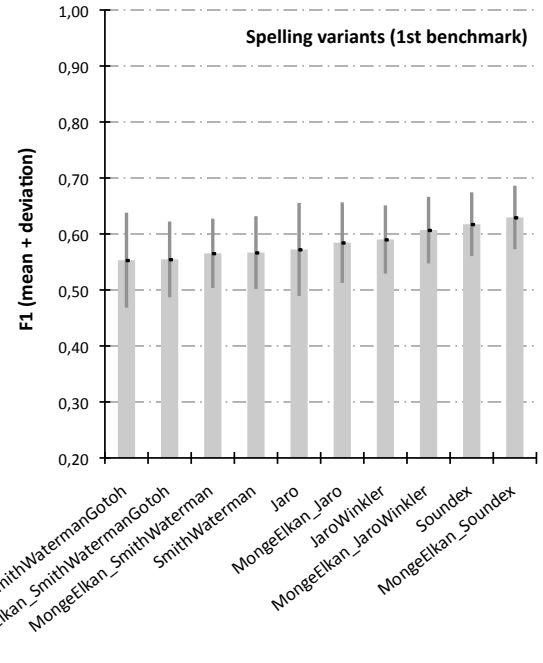


Figure 5.3: Mean values and deviation of the F_1-measure for the spelling variant case (1st benchmark)

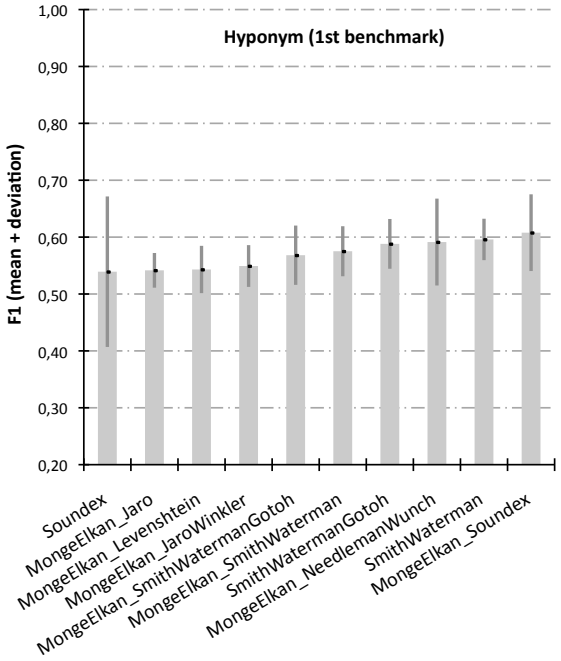


Figure 5.4: Mean values and deviation of the F_1-measure for the hierarchical case (1st benchmark)

For the second point, we wanted to find the metrics that are able to distinguish a pair of *spelling variant* tags or a pair of *hyponym* tags from a pair of merely *semantically linked* tags. Indeed, the two first types of relations are the most specific, and we assume in this study that a pair of *semantically linked* tags that is not *spelling variant* nor *hyponym* can be considered to be *related*. Thus, to translate these two points in the experiment, we used a different method to count the false positive and true positive pairs that were retrieved for a given threshold value τ, $\tau\varepsilon[0,1]$:

(a) for the **semantically linked** case, the number of true positives TP is counted by the number of pairs that are retrieved from the *related*, *spelling variant*, and *hyponym* sets, and the number of false positives FP is given by the number of pairs retrieved from the *unrelated* set;

(b) for the **spelling variant** case, the number of true positives TP are counted by the number of pairs retrieved from the *spelling variant* set, and the false positive FP are counted by the pairs retrieved from all the other sets (namely *related*, *hyponym*, and *unrelated*);

(c) for the **hyponym** case, the number of true positives TP are counted by the number of pairs retrieved from the *hyponym* set, and the false positive FP are counted by the pairs retrieved from all the other sets (namely *related*, *hyponym*, and *unrelated*).

Note that, for the spelling variant and hyponym cases, we include also the set of unrelated tags to count the false positive so that we make sure that the best metrics in this cases are also good at avoiding unrelated tags that are retrieved by mistake by MongeElkan Soundex metric. Figures 5.5,5.6, and 5.7 show the mean value and the statistical deviation of F_1 for the top 10 metrics for each case of semantic relation, respectively semantically linked, spelling variants, and hyponym.

By looking at the global results, we see that the Monge-Elkan_Soundex method outperformed other metrics in the semantically linked case, and this confirms the results of the first benchmark, which showed that this metric is able to retrieve pairs of tags sharing one of the three types of relations. The best in the spelling variant case is the Jaro-Winkler metric. For the hyponym case, the best metric is MongeElkan_NeedlemanWunch, but in this case it does not clearly outperform the seven metrics that come after, and this means that further investigations are needed for the hyponym case.

Conclusion of the second benchmark The temporary conclusions after the second benchmark is that :

- we can use the MongeElkan_Soundex metric to retrieve pairs of tags sharing one of the three relations, *ie* to detect *semantically linked* tags. We can say *e.g.,* that "energy" and "energies" are semantically linked as a first guess, and then find out that, more specifically, these two tags are, *e.g.,* spelling variant of each other.

- we can use the JaroWinkler metric to distinguish *spelling variants* from merely *semantically linked* tags in a second time.

- we need further investigations to find a way to distinguish *hyponym* pairs of tags from *semantically linked* tags.

5.3.2.4 Distinguishing *hyponym* tag pairs (third benchmark)

The goal of this third benchmark is to find a metric that is best at distinguishing *hyponym* tag pairs from merely *semantically linked* tag pairs. In figure 5.7 we see that the 9 of the 10 best metrics for the *hyponym* case where composite metrics of MongeElkan with another metric. This lead us to look at the specificity of the MongeElkan metric to see whether one of its features could be exploited to retrieve hyponym relations. We should also remark here that the type of hyponymy we are likely to retrieve with string based methods is the one that can be found in "pollution" and "soil pollution" for instance, *ie* when the narrower term contains the broader term or one of its derivative (such as in "pollution" and "pollutant detection").

The MongeElkan metrics are not symmetric, and we have calculated, for each tag pair (t_1, t_2), the difference $\delta(t_1, t_2) = s(t_1, t_2) - s(t_2, t_1)$, with s being one of the

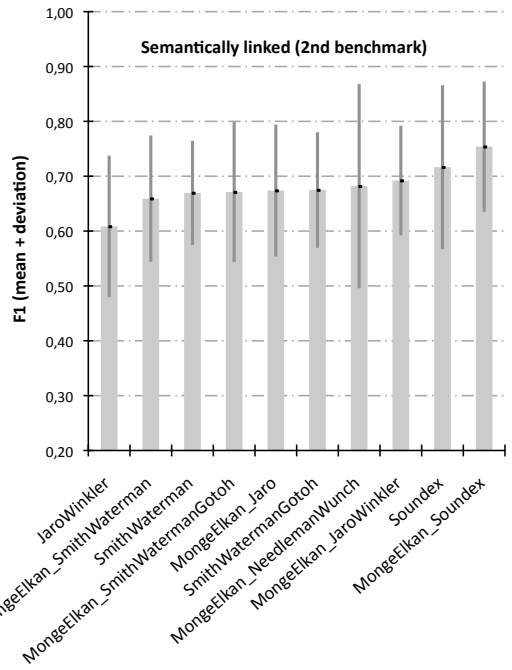


Figure 5.5: Mean and deviation values of F_1 for the top 10 metrics to retrieve pairs of *semantically linked* tags, *i.e.* tags sharing either a *related*, *spelling variant*, or *hyponym* relation (second benchmark)

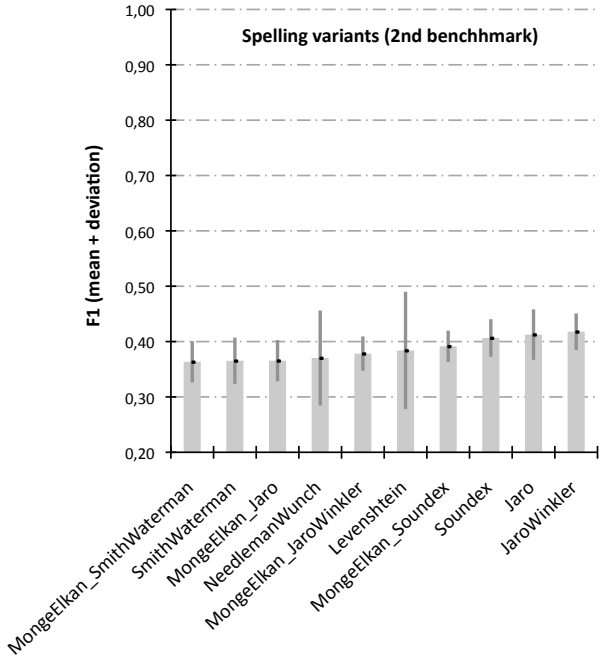


Figure 5.6: Mean and deviation values of F_1 for the top 10 metrics to retrieve the semantic relation *spelling variant* (second benchmark)

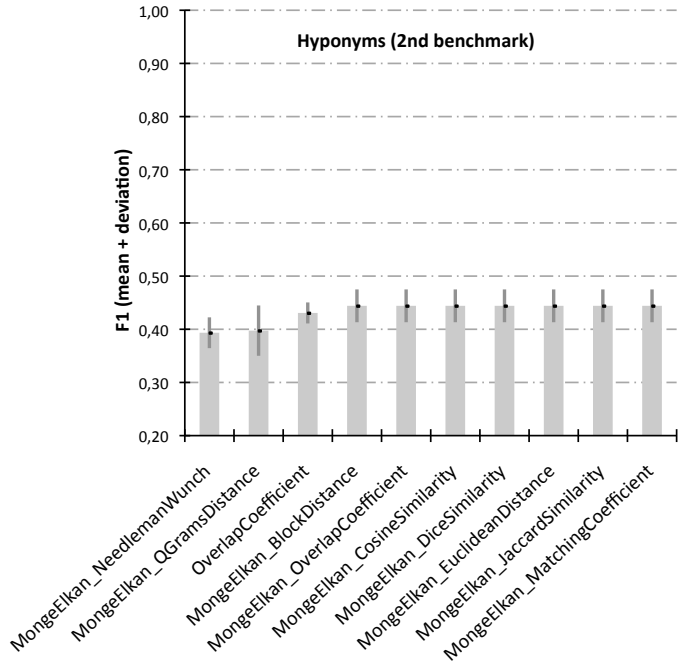


Figure 5.7: Mean and deviation values of F_1 for the top 10 metrics to retrieve semantic relation *hyponym* (second benchmark)

15 combinations of MongeElkan with another metric. Then, in order to check that this feature of the MongeElkan metrics can help us distinguish hyponym pairs from non-hyponym pairs (*ie* in our study, *related*, *spelling variant*, and *unrelated* tag pairs), we have computed the difference between the mean value of δ for the hyponym tag pairs $\delta_{hyponym}$ and the mean value of δ for the non-hyponym tag pairs $\delta_{non-hyponym}$ (the deviation is given by the difference of the deviations for each case, and in this benchmark we compute the absolute value of δ). The results for each composite MongeElkan metric is given in figure 5.8. In this figure we see that the difference between the value of δ for the hyponym and for the non-hyponym tag pairs is significant for most of the MongeElkan-based metrics, and the MongeElkan_QGramDistance is the best metric of this comparison.

Conclusion of the third benchmark. After this third benchmark, we conclude that we can exploit the asymmetry of the MongeElkan_QGramDistance metric to detect that two tags (t_1, t_2) share a hyponym relation. To this end we compute $\delta(t_1, t_2) = s(t_1, t_2) - s(t_2, t_1)$. If $|\delta(t_1, t_2)|$ is above a given threshold τ, we can infer that there is a hyponym relation between t_1 and t_2, and the sign of $\delta(t_1, t_2)$ gives the direction of the relation, so that if $\delta(t_1, t_2) > \tau$ we can infer that t_1 is broader than t_2, and if $\delta(t_1, t_2) < -\tau$, then we can infer that t_1 is narrower than t_2.

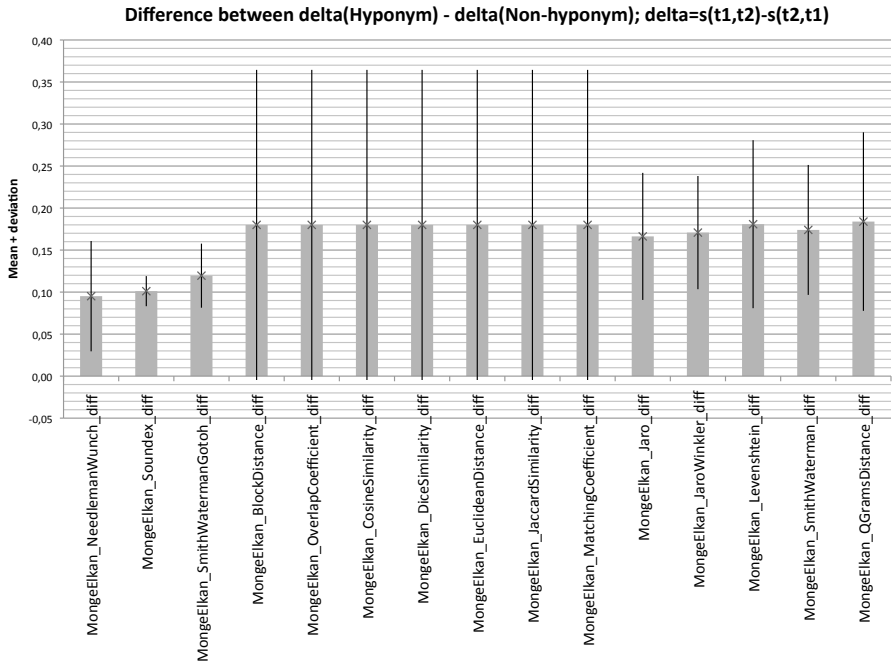


Figure 5.8: Mean value and deviation for $\mid \delta(t_1,t_2)_{hyponym} \mid - \mid \delta(t_1,t_2)_{non-hyponym} \mid$, with $\mid \delta(t_1,t_2) \mid = \mid s(t_1,t_2) - s(t_2,t_1) \mid$, s being one of the composite MongeElkan similarity metric. We computed here δ for all tag pairs of the *hyponym* set and all tag pairs of the *non-hyponym* sets (*ie related, spelling variant,* and *unrelated)*

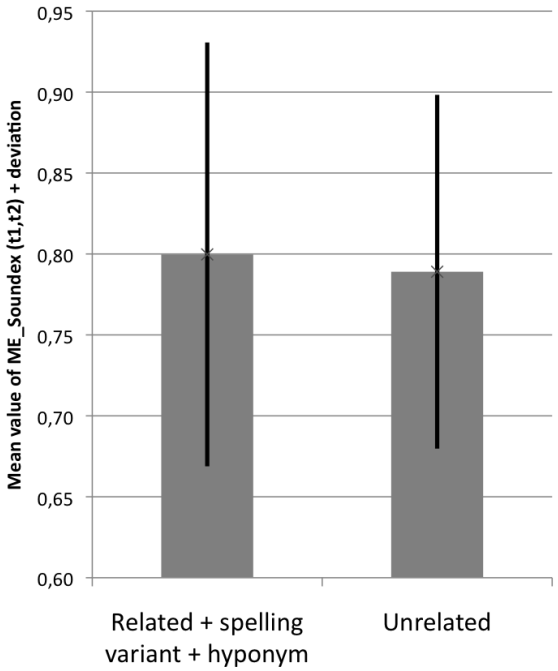


Figure 5.9: Comparison of the mean value of the MongeElkan_Soundex metric for all *semantically linked* cases (*spelling variants*, *hyponym* and *related*) and for *unrelated* cases.

5.3.2.5 Choosing thresholds by looking at mean and deviation of similarity values

First, we use the MongeElkan_Soundex metric to retrieve *semantically linked* tags, meaning that in this category we retrieve *related*, *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 *semantically linked*. To determine this threshold, we looked at the mean similarity value for all *semantically linked* cases (*spelling variant*, *hyponym*, *related*) and for all *unrelated* cases in the sample set. The results are shown in fig. 5.9. We can see that, considering the deviations, if we choose a threshold value of 0.9 we are able to avoid pairs of *unrelated* tags and more likely to retrieve a pair of *semantically linked* tags.

To distinguish *spelling variant* from merely *semantically linked* pairs, we looked at the mean value and deviation of the best metric in the *spelling variant* case and for each tag sets. In figure 5.10 we show the mean value of the Jaro-Winkler 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 than pairs from other tag sets.

Next, we saw above that we can detect if two tags t_1, t_2 share a hyponym relation by looking at the value of $\delta(t_1, t_2) = s(t_1, t_2) - s(t_2, t_1)$ with s being the MongeElkan_QGRamDistance metric. In figure 5.6 we give the mean value and

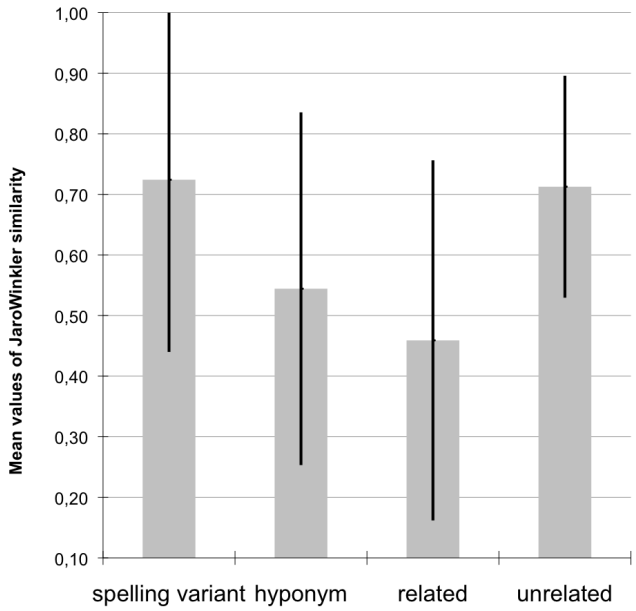


Figure 5.10: Comparison of the mean value of the JaroWinkler metric for each type of semantic relation.

deviation of the absolute value of δ for each set of tag pairs according to the MongeElkan_QGramDistance. We can see that if we choose a threshold above 0.39, the highest value of δ for the non-hyponym tags when including the deviation, we are able to retrieve tags sharing a hyponym relation while avoiding non-hyponym tag pairs.

However, when choosing the threshold values by looking at the mean similarity values we do not have a precise idea of the number of the positive tag pairs we retrieve or of the number of false positive we avoid. Subsection 5.3.2.7 will discuss the choice of thresholds in the light of an analysis of the distribution of the similarity values. But before that, we are going to first have a look at these distributions to check the homogeneity of the similarity values.

5.3.2.6 Homogeneity of the distributions of similarity values

In this subsection, we discuss the analysis of homogeneity of the distribution of the similarity values among the pairs of tags. Indeed, the homogeneity of the similarity values will higher up the confidence in the values of the threshold we obtain and guaranty the validity of this thresholds for other datasets. To evaluate the homogeneity, we plot the percentage of tag pairs that have a similarity value below a given value s, computing in this way the cumulative distribution for each similarity value between 0 and 1. To analyze the homogeneity of these distributions, we have plotted the cumulative distribution for 3 different partitions of equal size of the tag dataset. As an illustration, we show in figure 5.12 the distribution of the JaroWinkler similarity values for 3 different partitions of the spelling variant

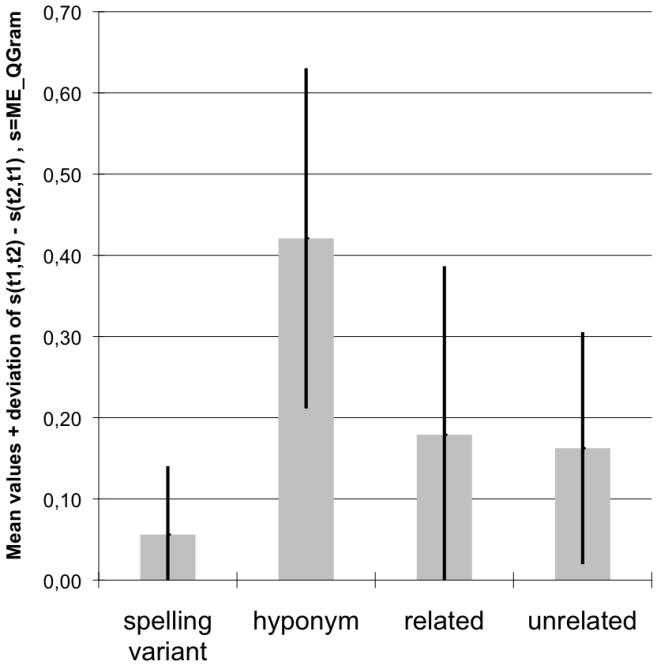


Figure 5.11: Mean value of the difference $\delta = s(t_1, t_2) - s(t_2, t_1)$ with s being the Monge-Elkan_QGram metric for each set of tag pairs.

datasets, and in figure 5.13 we show the same thing but for 3 partitions of the tag datasets that correspond to non-spelling variant relations, *i.e.* the *related*, *hyponym*, and *unrelated* tag datasets. In both cases we notice that the distributions for the 3 partitions follow similar patterns, which means that the similarity values seem to have an homogeneous behavior on our dataset.

We have repeated this comparison for the MongeElkan_Soundex metric used to retrieve pairs of *semantically linked* tags. The results are show in figure 5.14 for the sets of *semantically linked* tags, and in figure 5.15 for the set of *unrelated* tags. The conclusions are similar to the case of JaroWinkler similarity metric, and we see that the MongeElkan_Soundex similarity metric have an homogeneous behavior in both sets of tags. Now we are going to discuss the choice of threshold values with the help of distribution analysis.

5.3.2.7 Distribution analysis for the choice of thresholds

In subsection 5.3.2.5 we have discussed the choice of thresholds by looking at the mean and deviation of the similarity values. However, we know that, in the case of a normal distribution, according to the *three-sigma-rule*[4], around 95% of the values are contained within a range of twice the deviation away from the mean. This rule means that, in the ideal case of a normal distribution, we have a precise idea

[4]http://en.wikipedia.org/wiki/Normal_distribution#Standard_deviation_and_confidence_intervals

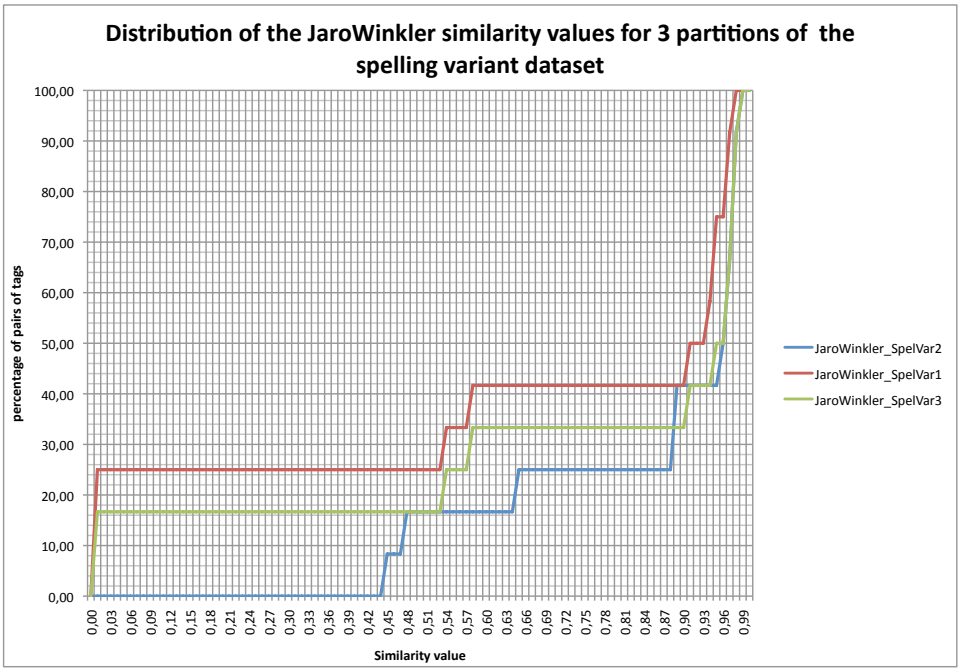


Figure 5.12: Distribution of the JaroWinkler similarity value for different partitions of equivalent size of the spelling variant tag dataset

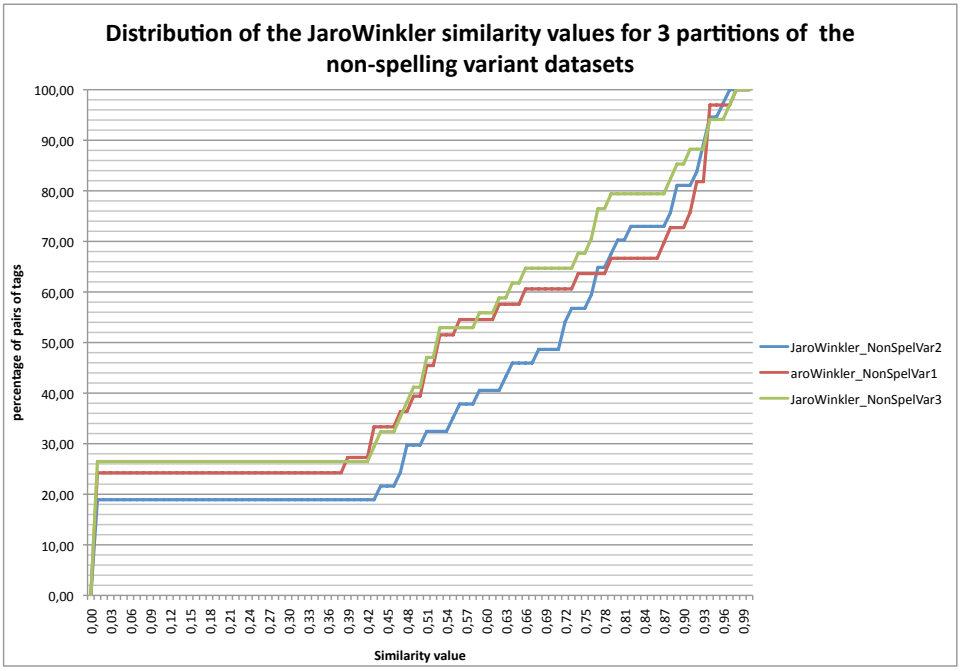


Figure 5.13: Distribution of the JaroWinkler similarity value for different partitions of equivalent size of the non-spelling variant tag datasets, *ie* in our case the *related*, *hyponym*, and *unrelated* tag datasets

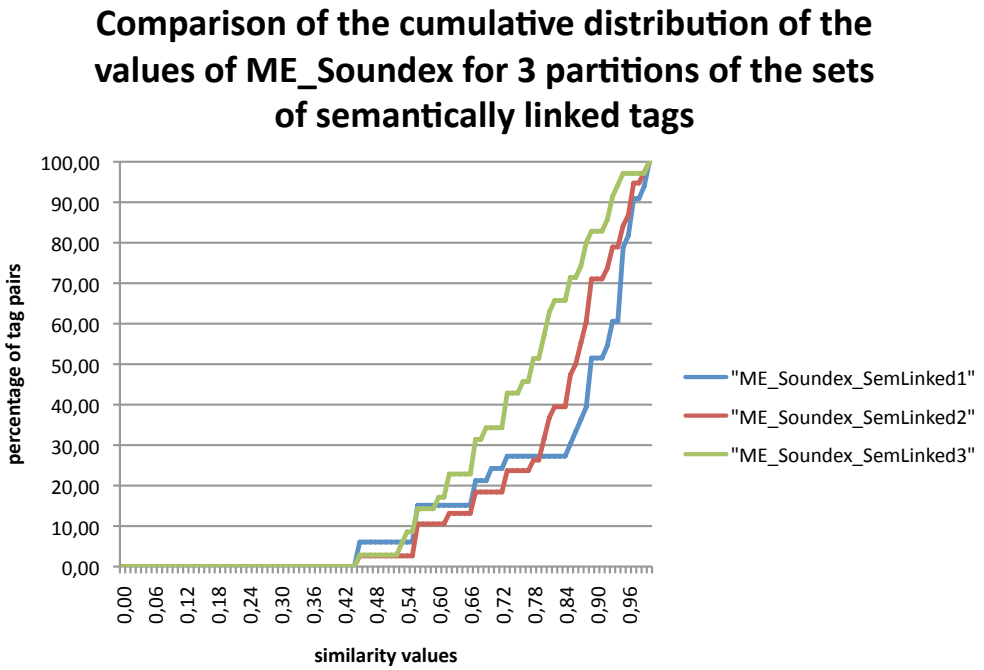


Figure 5.14: Distribution of the MongeElkan_Soundex similarity value for different partitions of equivalent size of the *semantically linked* tag datasets, *i.e.* the union of the *related*, *spelling variant*, and *hyponym* tag datasets

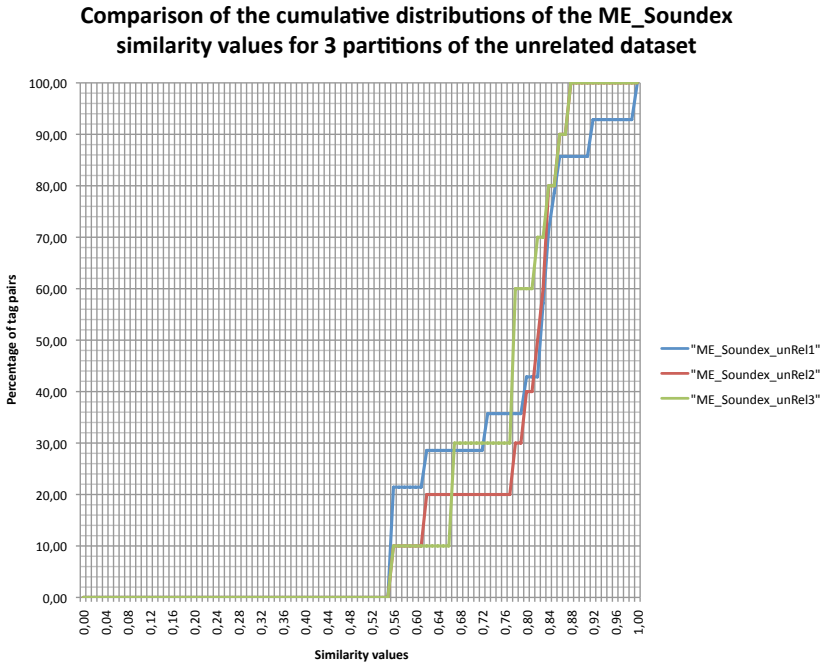


Figure 5.15: Distribution of the MongeElkan_Soundex similarity value for different partitions of equivalent size of the *unrelated* tag dataset

of the percentage of values we retrieve when looking at the value of the mean and deviation. In our case, we are probably not in this ideal situation, but it is nonetheless interesting to look at the distribution of the similarity values to have additional information on the choice of thresholds. Thus, for each case of semantic relation, we plot the cumulative distribution (similarly as explained above) for the set of pairs of tags that correspond to the given relation, and the cumulative distribution for the set of pairs of tags that do not correspond to the given relation.

***Spelling variant* case** We show in figure 5.16 the comparison of the cumulative distributions of the JaroWinkler similarity values for the spelling variant and non-spelling variant tag pairs (*ie related*, *hyponym*, and *unrelated*). We see in this chart that if we choose a threshold of 0.9, as explained in subsection 5.3.2.5, we will filter out 77% of the non-spelling variant pairs of tags, since 77% of these pairs have a similarity value below 0.9. In terms of true positives, we see on this chart that 45% of the spelling variant pairs have a similarity value below 0.9, which means that if we choose this threshold value, we retrieve 55% of true positives. If we want to filter out more non-spelling variant pairs, for instance 95%, then the threshold to choose is around 0.94, and at this threshold value, 55% of the spelling variant pairs will be filtered out. Thus, we see that the analysis of the distribution of the similarity values significantly helps justify the choice of threshold values by providing for valuable information about potential precision and recall we obtain in the end. In our study, we decided to keep the threshold at 0.9 in order to maximize the percentage of positive pairs we retrieve.

***Hyponym* case** In order to detect hyponym tags, we saw that we compute, for each tag pair (t_1, t_2), the value $\delta(t_1, t_2) = s(t_1, t_2) - s(t_2, t_1)$ with s being the MongeElkan_QGRamDistance metric. Then if $\delta(t_1, t_2)$ is above a given threshold, the tag pair (t_1, t_2) is considered to share hyponym relation. In figure 5.17 we show the cumulative distribution of the values of δ for the hyponym tag pairs and for the non-hyponym tag pairs (*i.e., spelling variant*, *related*, and *unrelated* tag pairs). In this chart, if we look at the distribution for the threshold we chose in subsection 5.3.2.5, *i.e.* 0.39, we see that we filter out 87% of the non-hyponym tag pairs and 39% of the hyponym tag pairs. Interestingly, we also see that the percentage of hyponym tag pairs remains the same for threshold values up to 0.44, while the number of non-hyponym tag pairs we filter out ramps up to 90%. So if we choose a threshold value of 0.44 we retrieve as many true positives (61%) and filter out more false-positives (90% instead of 87%) than with a threshold value of 0.39.

***Semantically linked* case** In figure 5.18 we compare the cumulative distributions of the MongeElkan_Soundex similarity values for all the sets of *semantically lin*ked tags (*i.e. spelling variant, hyponym,* and *related* tags sets) and for the *unrelated* tag set. The first thing to notice is that the distributions are less discriminative than for the two other cases of semantic relations, since both curves are closer to each

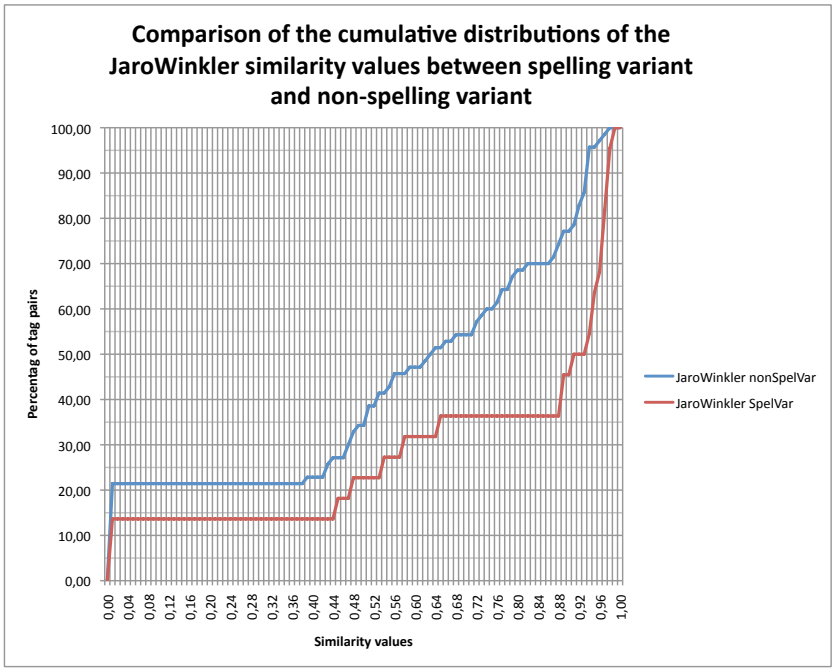


Figure 5.16: Comparison of the distribution of the JaroWinkler similarity value for the spelling variant and the non-spelling variant tags datasets, *ie* in our case the *related*, *hyponym,* and *unrelated* tags datasets.

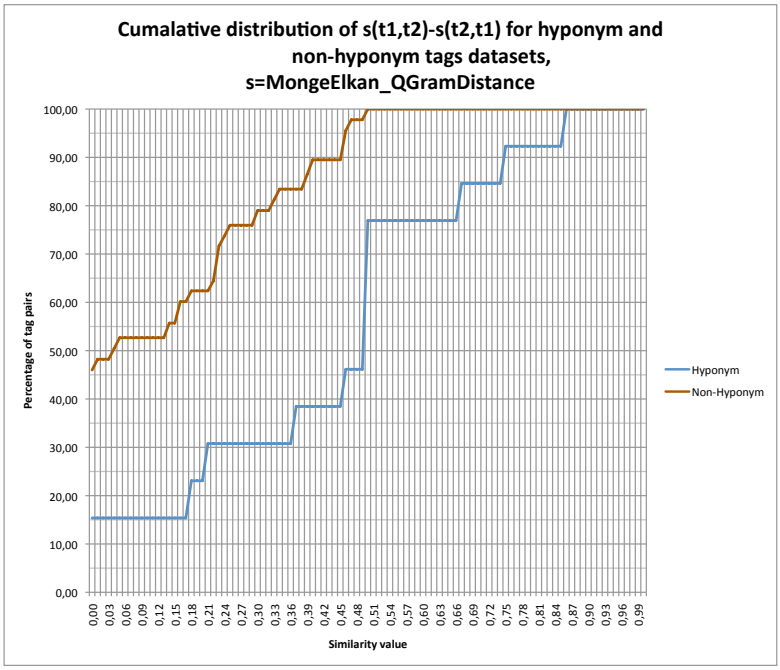


Figure 5.17: Comparison of the distributions of the values of $\delta(t_1, t_2)$ for the hyponym and the non-hyponym tag datasets, *i.e.* in our case the *related*, *spelling variant,* and *unrelated* tags datasets. ($\delta(t_1, t_2) = s(t_1, t_2) - s(t_2, t_1)$ with s the MongeElkan_QGramDistance metric)

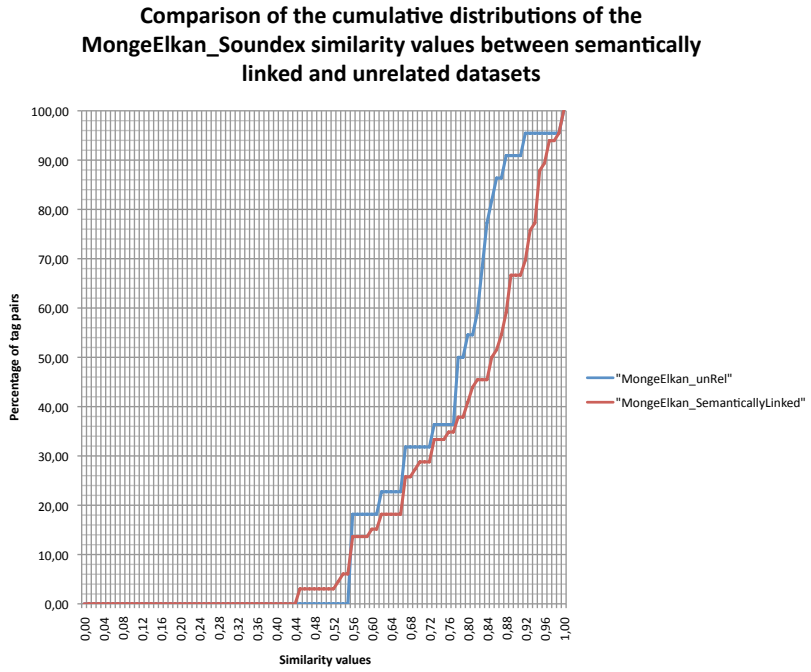


Figure 5.18: Comparison of the distributions of the MongeElkan_Soundex similarity values for all the *semantically linked* tags datasets (*i.e. related*, *spelling variant*, and *hyponym*) and the *unrelated* tag dataset

other, which translates into a lower difference between the number of false positives that are filtered out and the number of true positives that are retrieved for a given threshold value. However, this situation was already predictable when we looked at the higher deviation for the *related* case in the first benchmark, and for the *semantically linked* case in the second benchmark, and this is explained by the fact that for two notions being *related* or merely *semantically linked* rarely translates into morphological similarities, as in "vehicle" and "car" for instance, which are *semantically linked* terms sharing only one letter. If we look now at the distribution we get for the threshold value chosen in subsection 5.3.2.5, *i.e.* 0.9, we get 67% of *semantically linked* pairs that have a similarity value below this threshold and 90% of the *unrelated* pairs. This means that for a threshold value equal to 0.9, we will retrieve 33% of the positive pairs and filter out 90% of the negative pairs. However, we should also remark that the number of *semantically linked* tags in a folksonomy is obviously higher than for the other types of more precise relations, since this relation is meant to include the other types of relation, and thus a lower recall in this case is less problematic.

5.3.3 Heuristic string-based method

5.3.3.1 Algorithm

As a result we are able to propose a heuristic (see algorithm5.1) that combines the best metrics to retrieve different semantic relations between tags. With the first benchmark, we saw that the MongeElkan_Soundex metric was the best to retrieve pairs of *semantically linked* tags, *i.e.* pairs of tags sharing one of the three relations, namely *related*, *spelling variant* or *hyponym*. Then, we conducted a second benchmark that evaluated the ability for each metric to distinguish, for instance, a pair of *spelling variant* tags from a pair of merely *semantically linked* tags retrieved with MongeElkan_Soundex. In this regard, the JaroWinkler metric proved to be the most efficient, while no metric did outperform the others in the *hyponym* case. A third comparison exploited the asymmetry of the MongeElkan composite metrics to distinguish *hyponym* tags, and showed that we can distinguish hyponym tags by computing the value $\delta(t_1, t_2) = s(t_1, t_2) - s(t_2, t_1)$, with s corresponding to the MongeElkan_QGramDistance metric. Finally, if a pair retrieved as *semantically linked* is not a *spelling variant* pair, nor an *hyponym* pair, then we conclude that it is a *related* tags pair.

Algorithm 5.1 Heuristic string based metric to retrieve semantic relations between tags

Require: threshold for *semantically linked :* τ_a
Require: threshold for *spelling variant :* τ_b
Require: threshold for *hyponym :* τ_c
for all distinct pair of tags (t_i, t_j) from $S = \{t_1, t_2, ..., t_n\}$ **do**
  if $MESoundex(t_i, t_j) > \tau_a$ **then**
    if $JaroWinkler(t_i, t_j) > \tau_b$ **then**
      t_i has spelling variant t_j
    else if $MEQGram(t_i, t_j) - MEQGram(t_j, t_i) \leq -\tau_c$ **then**
      t_i has broader t_j
    else if $MEQGram(t_i, t_j) - MEQGram(t_j, t_i) \geq \tau_c$ **then**
      t_j has broader t_i
    else
      t_i has related t_j
    end if
  end if
end for

The heuristic string-based algorithm we propose is shown in algorithm 5.1. We first look for pairs of *semantically linked* 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 subsection 5.3.2.5, i.e. $\tau_a = 0.9$ in our case. Then, we compare the JaroWinkler similarity with a second threshold τ_b to see whether, more specifically, the tags are spelling variants, such that $s(t_1, t_2) \geq \tau_b$. The threshold in this case is

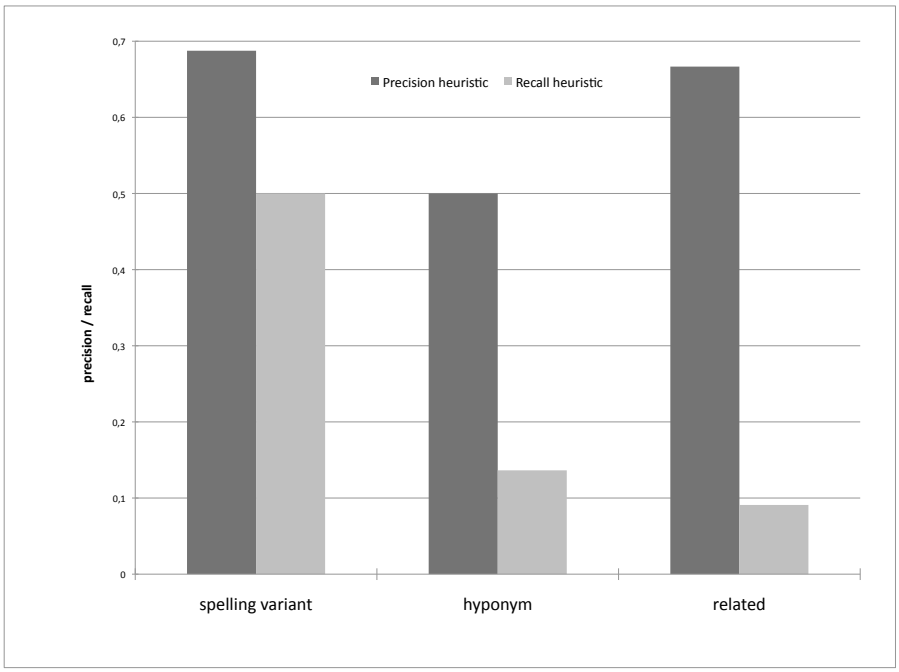


Figure 5.19: Performance of the heuristic string-based metric.

chosen as explained in subsection 5.3.2.5, *i.e.* in our case, 0.9. 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 (*i.e.* that t_1 has for broader notion t_2), or if $\delta \geq \tau_c$ then t_1 is considered broader than t_2 (*i.e.* that t_2 has for broader notion t_1). We saw in subsection 5.3.2.5 that the third threshold can be chosen by picking a value above 0.39, but we also saw in subsection 5.3.2.7 that, by looking at the distributions, this value can be increased up to 0.44 in order to gain more precision by filtering out more false positives while retrieving as many true positives. 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 more chance to be true positives.

5.3.3.2 Performance

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 5.19 the values of the precision and recall for the 3 types of relations. We can clearly see in this figure that

tag t_1	relation	tag t_2
climatechange	related	changement_climatique (climate change)
changements	spelling variant	changement
electricite (electricity)	related	electrodes
electrocatalyse (electrocatalysis)	related	electrodes
conversion d'energie (energy conv.)	has broader	energie
energie	related	energia

Table 5.2: Example results of semantically linked tags thanks to the heuristic string-based method (English translation of french terms in parentheses)

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 recall 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 this cases often correspond to a *hyponym* relation. The *related* case is more difficult (hence the lowest recall) as this relation is the fuzziest and probably the least noticeable in the actual spelling of the tags ("sun" and "energy" *e.g*). Finally, we see that except for the *spelling variant* case, the recall are quite low, and this indicates the need to use other methods to be able to cover other cases where semantically linked tags are not morphologically similar.

5.3.3.3 Example results

In table 5.2 we give some examples of pairs of semantically linked tags retrieved thanks to the heuristic string-based method we have presented in this section. This is a sample of the results we obtained for our dataset, and the details of the dataset and computations are given in chapter 8. We see in these examples that this type of method allows linking different spelling variants of the same notion ("changements" and "changement"), but also to link very similar notions written in different languages ("energie" and "energia") and with different ways of spelling compound words ("climatechange" and "changement_climatique"). The hyponym relations we retrieve are mostly consisting of pairs where the narrower term is a compound word of the broader term, such as in "conversion d'energie" which has for broader notion "energie".

5.3.4 Temporary conclusion

The goal of our benchmark of the string-based similarity metrics is two-fold: (1) we wanted to motivate the choice of the best metric to perform the identification of spelling variant tags (2) we wanted to compare these metrics regarding their ability to detect other relations than spelling variant.

The result of this benchmark is that some metrics are better than others at retrieving a specific semantic relation, and, in the end, we are able to propose a heuristic method that combine efficiently string-based metrics in order to retrieve different types of relation. The MongeElkan_Soundex metric thus revealed to be good at finding the three types of relation; then, the JaroWinkler metric is used to distinguish a *spelling variant* pair from a merely *semantically linked* pair. Finally, we exploit the MongeElkan_QGram metric to tell *hyponym* pairs apart from the remaining pairs, and the pairs of *semantically linked* tags that pass through both of these filters are considered to be *related.*

Lastly, our method based on the combination of string-based metrics is meant to complement and is not opposed to the use of external ontological resources. Indeed, this method allows overcoming the absence of some very specific terms from such ontological resources. As an illustration, only 716 tags out of the 9000 tags contained in our dataset are found in the GEMET thesaurus, which is the reference thesaurus in the field of our target community (see the SPARQL query used to count this number and a sample of the result in Annex A on page 253). But, when the tags to be compared are present on such resources, they will be naturally integrated in our knowledge base since our structuring of tags is based on the semantic relations found in thesauri. Any existing thesaurus can be loaded from the very start of our approach and is in particular useful to avoid the cold-start effect. Furthermore, string-based metrics are specifically suited for very technical terminologies. Indeed, the hierarchical links in such terminologies are typically based on lexical variations, such as in “pollution” and “soil pollution”. And we have shown through our study that using string-based methods to detect such type of semantic relations is a valid approach.

The relatively low recall of the heuristic string-based method shows, if this was needed, that this type of method is not sufficient and needs to be complemented by the analysis of the structure of the folksonomy, which is what is covered in next section.

5.4 Analyzing the structure of folksonomies

In this section we detail our implementation of two methods extracting emergent semantics by analyzing the tri-partite structure of the folksonomy. The first method we present allows retrieving *related* relationships by computing the cosine similarity for the distributional aggregation in the Tag-tag context. The second method allows inferring *hyponym* relations and is based on mining association rules by looking at inclusions of communities of interest defined by the set of users that use a given tag.

5.4.1 Tag-tag context similarity measure to infer *related* relationships

5.4.1.1 Description of the method

As we reviewed it in section 3.3.2 on page 32, one efficient way for extracting emergent semantics from social tagging data consists in computing tag similarity by analyzing the tri-partite structure of folksonomies. Some methods exploit the simple co-occurrence of tags to compute such a similarity, but the method we present here (Cattuto *et al.*, 2008) proposed a more elaborated way of computing tag similarity that is based on two steps:

1. Aggregating the tri-partite structure of folksonomies onto 2-mode view of the tagging data.

2. Applying a similarity measure on this 2-mode view

Regarding the aggregation of tagging data, Cattuto *et al.* showed that exploiting the distributional hypothesis on the folksonomy structure yielded good quality semantics. The distributional hypothesis states that words used in similar contexts tend to be semantically related (Firth, 1957). When applied to folksonomies, this hypothesis means that tags occurring in similar contexts regarding the other elements (namely the users, the resources, and the other tags with which they co-occur) tend to be closely related. Distributional aggregation consists thus in capturing the contextual information and to report it in vector representations of the tags where each entry corresponds to one of the item of the context we consider. The components of the vectors v_t representing each tag t and given for each context by:

- **Tag-Tag Context** : each entry $v_{t,i}$ of the tag vector v_t corresponds to the value of the co-occurrence of the tag t with each tag t_i, except for the tag t with itself (when $t_i = t$) where a weight of 0 is given. This is to avoid to consider two tags related when they merely occur together, but rather when they have similar patterns of co-occurrence, that is, when they co-occur with the same other tags.

- **Tag-Resource Context** . For a tag t, each entry $v_{t,i}$ of the vector v_t is constructed by counting how often the tag t is used to annotate each resource r_i.

- **Tag-User Context .** For a tag t, each entry $v_{t,i}$ of the vector v_t is constructed by counting how often the tag t is used by a each user u_i.

Then the second step of the computation consists in applying a similarity measure between the vector representation of the tags.

To go further in the analysis, and in order to semantically ground these different kinds of similarity measures, Cattuto *et al.* (2008) proposed exploiting the hierarchical structure of WordNet for the tags which can be found within this database.

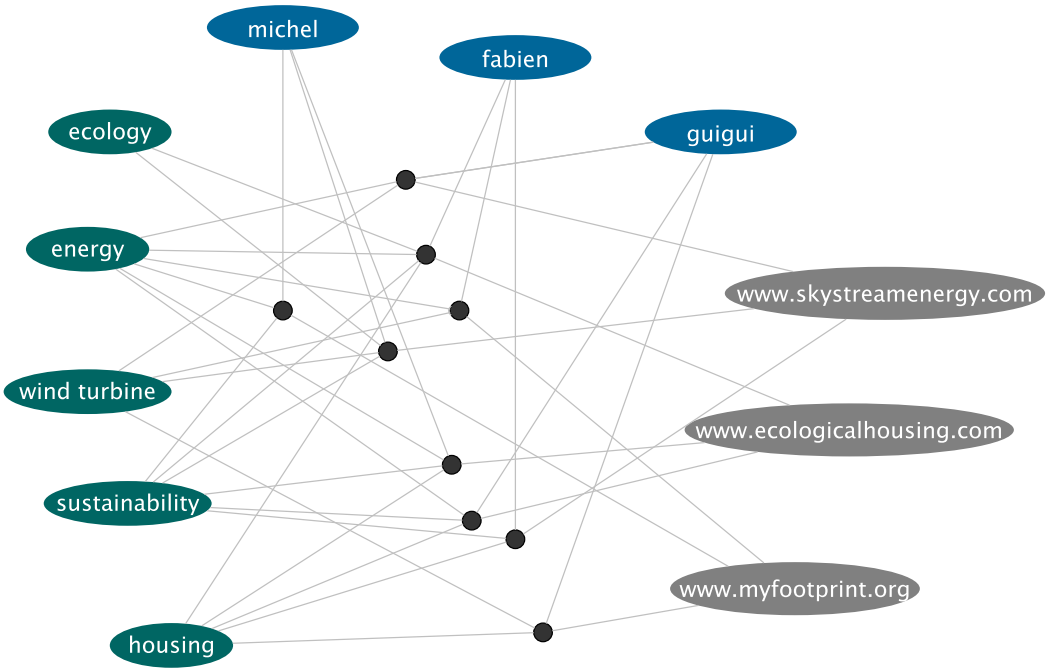


Figure 5.20: Example folksonomy showing the 9 posts of 3 users on 3 different resources using 5 distinct tags. Tags are represented by green nodes, users by blue nodes, and resources by grey nodes. A black dot represent a post, *i.e.* a link between a set of tags, a user, and a resource.

user	resource	tags				
		ecology	energy	wind turbine	sustainability	housing
michel	skystreamenergy.com	x		x	x	
	ecologicalhousing.com		x		x	x
	myfootprint.org	x	x		x	
fabien	skystreamenergy.com				x	x
	ecologicalhousing.com	x	x		x	x
	myfootprint.org		x	x		
guigui	skystreamenergy.com		x	x		
	ecologicalhousing.com		x		x	x
	myfootprint.org			x		x

Table 5.3: Table detailing the contents of the posts of the example folksonomy given in figure 5.20.

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 grandparents. Cattuto *et al.* (2008) 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". In thesauri terms, these notions would be considered *related*.

Thus, 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). To compute this 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 $v_{t_i t_j} = w(t_i, t_j)$ for $t_i \neq t_j$ where $w(t_i, t_j)$ corresponds to the co-occurrence on a same post of the tags (t_i, t_j), and when $t_i = t_j$, $v_{t_i t_i} = 0$. 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.

To illustrate this computation with a concrete example, we consider now the example folksonomy given in figure 5.20, which contains 9 posts of 3 users that associated, globally, 5 distincts tags to 3 different resources. The detail of each post is given in table 5.3. In table 5.4, we give the matrix for the distributional aggregation in the Tag-Tag context for the example folksonomy of figure 5.20. For instance the vector representation of the tag "ecology" is $v_{ecology} = (0, 1, 1, 3, 1)$.

	ecology	energy	wind turbine	sustainability	housing
ecology	0	1	1	3	1
energy	1	0	2	4	3
wind turbine	1	2	0	1	1
sustainability	3	4	1	0	4
housing	1	3	1	4	0

Table 5.4: Example of a distributional aggregation in the Tag-Tag context corresponding to the folksonomy example given in figure 5.20.

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 :

$$\cos(v_i, v_j) = \frac{v_i . v_j}{\|v_i\|_2 . \|v_j\|_2}.$$

	ecology	energy	wind turbine	sustainability	housing
ecology	x	0.90	0.65	0.40	0.89
energy	0.90	x	0.55	0.48	0.67
wind turbine	0.65	0.55	x	0.87	0.80
sustainability	0.40	0.48	0.87	x	0.48
housing	0.89	0.67	0.80	0.48	x

Table 5.5: Similarity values computed in the Tag-Tag context for the example folksonomy of figure 5.20.

In table 5.5 we give the similarity values computed for the example folksonomy of figure 5.20. The detail of the computation for the tags "ecology" and "sustainability", is the following:

$$\begin{aligned} cos(v_{ecology}, v_{sustainability}) &= \frac{0 \times 3 + 1 \times 4 + 1 \times 1 + 3 \times 0 + 1 \times 4}{\sqrt{0^2 + 1^2 + 1^2 + 3^2 + 1^2} \cdot \sqrt{3^2 + 4^2 + 1^2 + 0^2 + 4^2}} \\ &= \frac{9}{\sqrt{12}.\sqrt{42}} \backsimeq 0,4 \end{aligned}$$

And for the tags "ecology" and "housing" :

$$\begin{aligned} cos(v_{ecology}, v_{housing}) &= \frac{0 \times 1 + 1 \times 3 + 1 \times 1 + 3 \times 4 + 1 \times 0}{\sqrt{0^2 + 1^2 + 1^2 + 3^2 + 1^2} \cdot \sqrt{1^2 + 3^2 + 1^2 + 4^2 + 0^2}} \\ &= \frac{16}{\sqrt{12}.\sqrt{27}} \backsimeq 0,88 \end{aligned}$$

We see in these examples that, by setting the co-occurrence for a tag with itself to 0, we do not take into account the fact that the tag "ecology" and the tag "sustainability" co-occurred three times on the same post; on the contrary, this pair of tags gets the smallest similarity value of this sample. Indeed, this method of computing similarity gives higher values for tags that share the same pattern of coccurence, *i.e.* that cooccur with similar tags without necessarily co-occurring together. For instance, the tags "ecology" and "housing" co-occurred only once, but they also co-occurred with the same tags at similar frequencies (for example they both cooccur with the tag "wind turbine" once, and their frequence of co-occurrence with "sustainability" differs only of a unit). As a result, the similarity value is the highest for this pair of tags.

5.4.1.2 SPARQL queries to count cooccurrences

Listing 5.1: Example of tagging assignments with the NiceTag model. The user "michel" has tagged the resource "www.myfootprint.org" with the tags "ecology", "energy", and "sustainability".

```
<!-- tag ecology -->
<rdf:Description rdf:about="http://www.myfootprint.org"
cos:graph="http://ex.org/id/tag-action/01">
   <nicetag:isRelatedTo
      rdf:resource="http://ex.org/id/tag/ecology/>
</rdf:Description>

<nicetag:TagAction rdf:about="http://ex.org/id/tag-action/01">
   <sioc:has_creator rdf:resource="http://ex.org/id/user/michel"/>
</nicetag:TagAction>

<!-- tag energy -->
<rdf:Description rdf:about="http://www.myfootprint.org"
cos:graph="http://ex.org/id/tag-action/02">
   <nicetag:isRelatedTo
      rdf:resource="http://ex.org/id/tag/energy/>
</rdf:Description>

<nicetag:TagAction rdf:about="http://ex.org/id/tag-action/02">
   <sioc:has_creator rdf:resource="http://ex.org/id/user/michel"/>
</nicetag:TagAction>

<!-- tag sustainability -->
<rdf:Description rdf:about="http://www.myfootprint.org"
cos:graph="http://ex.org/id/tag-action/03">
   <nicetag:isRelatedTo
      rdf:resource="http://ex.org/id/tag/sustainability/>
</rdf:Description>

<nicetag:TagAction rdf:about="http://ex.org/id/tag-action/03">
   <sioc:has_creator rdf:resource="http://ex.org/id/user/michel"/>
</nicetag:TagAction>
```

As we have seen it, this method of computing similarity requires the count of co-occurrrence of tags. As the tagging instances in our approach are expressed with semantic metatadata following the NiceTag model, we can make use of a SPARQL engine to count the coocurrence of tags. Listing 5.1 shows an example of tagging assignments from our example folksonomy corresponding to the post of the user "michel" who associated the tags "ecology", "energy", and "sustainability to the resource "www.myfootprint.org". In NiceTag, a tag action links one user with one tag to one resource, this is why we have 3 tag actions to describe this post.

In listing 5.2, we show the SPARQL query that allows counting the co-occurrence of all tag pairs of the dataset. Two tags are said to co-occur when they appear together on the same post. As the notion of post is not directly expressed

in the RDF annotation of a tagging instance with NiceTag, we have to recompose the posts in the query. Lines 2 and 3 look for 2 distinct tag actions posted on the same resource, and lines 4-5 make sure that these two tag actions are posted by the same user. Hence, thanks to these lines, we retrieve the tags that co-occur on the same post. The total count of co-occurrence is done by grouping the results by pair of tags in line 7 and the count of the number of `?tagaction1`–counting `?tagaction2` would be equivalent– in line 1. The result of this query is a list of pairs of tags co-occurring at least once on a post, and the corresponding count of their co-occurrence. This list is then processed by algorithm 5.2 that we present below.

Listing 5.2: SPARQL query to count the co-occurrence for all pair of tags of the dataset

```
1 SELECT ?tag1 ?tag2 count(?tagaction1) as ?coocurence  WHERE {
2    GRAPH ?tagaction1 {?resource nicetag:isRelatedTo ?tag1 }
3    GRAPH ?tagaction2 {?resource nicetag:isRelatedTo ?tag2 }
4    ?tagaction1 sioc:has_creator ?user
5    ?tagaction2 sioc:has_creator ?user
6 }
7 GROUP BY ?tag1 ?tag2
```

5.4.1.3 Algorithm

The different steps of the method to infer *related* relationships between tags by computing the cosine similarity in the distributional aggregation of tagging data in the Tag-Tag context is given in algorithm 5.2. This algorithm requires the list of n tags and the list of cooccuring tags with their count of co-occurrence given by the query of listing 5.2, and an experimentally chosen threshold τ. The first step consists in filling the aggregation matrix A of dimension $n \times n$, for which the values for a tag with itself is set to 0, *i.e* $A[i][i] = 0, \forall i \varepsilon [0, n]$, and for the other pairs of (i, j) values, it is given by the count of co-occurrence of tags (t_i, t_j). Then, the similarity value for each distinct pair of tags (t_i, t_j) is given by the cosine distance computed between the vector representation v_i of each tag given by the corresponding line of the aggregation matrix, *i.e.* $v_i = A[i][]$. When the computed similarity is above a given threshold, the tag t_i is considered to share the semantic relation *related* with the tag t_j.

5.4.1.4 Example results

Table 5.6 shows a series of tags sharing the *related* relation and retrieved thanks to the method presented in this section and summarized by algorithm 5.2. The threshold for the similarity value is 0.83. These results come from the computation we have applied on our dataset, and all the details of these computations are given in chapter 8. The results show relevant associations of related tags regarding the topic of ecology and sustainable development, which is what we could be expecting since our sample folksonomy has been extracted from Ademe's tagging

Algorithm 5.2 Algorithm to infer *related* relationships between tags by computing the tag similarity in the Tag-tag context

Require: List of n tags L_{Tags}
Require: List of p pairs of tags co-occurring on a post $L_{CoocTags}$
Require: Similarity threshold τ
//Filling aggregation matrix
Set aggregation matrix : $A[n][n]$
for all distinct pair of tags (t_i, t_j) from L_{Tags} **do**
    if $t_i = t_j$ **then**
        $A[i][j] = 0$
    else
        look for (t_i, t_j) in $L_{CoocTags}$
        $A[i][j] = cooc(t_i, t_j)$
    end if
end for
//Computing tags similarity
for all distinct pair of tags (t_i, t_j) from L_{Tags} **do**
    $sim(t_i, t_j) = cos(A[i][], A[j][])$ ▷ //Tag vectors given by matrix lines
    if $sim(t_i, t_j) > \tau$ **then**
        t_i related to t_j
    end if
end for

data.

5.4.2 User-based association rules mining to infer hyponym relations

In order to extract hyponym 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.

tag t_1	**relation**	**tag** t_2
voiture (car)	related	automobile
développement (development)	related	durable (sustainable)
construction	related	habitat (housing)
solaire (solar)	related	photovoltaïque (photovoltaic)
réglementation (regulation)	related	thermique (thermal)

Table 5.6: Example results of semantically linked tags thanks to the method based on the cosine similarity computed for the distributional aggregation in the Tag-Tag context (English translation of french terms in parentheses)

5.4.2.1 Description of the method

Mika (2005) was among the first to propose looking at folksonomies as knowledge representations whose emergent semantics can be unveiled by the analysis of the tri-partite structure of folksonomies. Mika proposed a method based on projection aggregation to draw one-mode weighted graphs connecting together related tags. As we have seen it above, the aggregation of tagging data can be done in different contexts, each corresponding to the association of two primary elements of the folksonomy. In particular, Mika studied the results obtained in the Tag-Resource context and in the Tag-User context. He found that the latter context is best suited to extract taxonomic relations between tags. This context of aggregation consists in considering two tags to be related when they are shared by a high number of users, and a similar principle is also found in the Edinburgh Associative Thesaurus (EAT) [5], which was built by asking people to associate the first word they were thinking of when presented a given *stimulus* word (Kiss *et al.*, 1973). As pointed by Mika, the difference in the method based on folksonomy is that the associations are extracted automatically, but similarly to the EAT, the selection of the set of users has a strong influence on the associations that are drawn.

To go further than mere *related* relationships, Mika suggested to look at the inclusions of sets of users of tags to infer *hyponym* relations. Let U_i be the set of users using tag t_i, and U_j be the set of users using tag t_j. If the set U_i is included in the set U_j, *i.e.* if $(U_i \subset U_j)$, we can infer that the tag t_j is broader than the tag t_i. In order to avoid meaningless results we add some constraints and consider that all sets of users should have a minimum of 2 users, *i.e.*, $|U_i| > 1$, and that the number of users in U_j should exceed the number of users in U_i of more than one user, *i.e.*$|U_j| > (|U_i| + 1)$.

5.4.2.2 Using SPARQL to mine inclusions of sets of users of a tag

To detect the inclusions of sets of users in a folksonomy, we made use of the SPARQL engine Corese[6]. The corresponding SPARQL query, given below in listing 5.4, requires the list of associations between each tags and each user. This is easily extracted from the RDF tagging data we collected and wrote using NiceTag model. Listing 5.1 shows an example of a tagging assignment using NiceTag model. From this annotation, we can extract, with a simple rule shown in listing 5.3, the association of users and tags and write them with the property from the SCOT[7] model `scot:usedBy` which links a `scot:Tag` with a `sioc:User`.

Listing 5.3: Rule to transform the use of a tag by a user into an annotation between that tag and the user (the definition of the prefixes is omitted for a better readability)

```
1 <cos:rule>
```

[5] http://www.eat.rl.ac.uk/

[6] http://www-sop.inria.fr/edelweiss/software/corese/

[7] http://scot-project.org/scot/ns#

```
2     <cos:if>
3        {GRAPH ?tagging {?r nicetag:isRelatedTo ?tag}
4        ?tagging sioc:has_creator ?u
5        FILTER (isDistinct(?u))
6        FILTER(isDistinct(?tag))}
7     </cos:if>
8     <cos:then>
9        {?tag scot:usedBy ?u}
10    </cos:then>
11 </cos:rule>
```

In listing 5.4 we report the query used to find the pairs of tags (t_1, t_2) for which the set U_1 of users of the tag t_1 is included in the set U_2 of users of the tag t_2. Lines 3-5 look for users u_1 of both tags t_1 and t_2, and for users u_2 of the tag t_2. Then, lines 6-9 check that no user u_2 uses also tag t_1, and lines 10-18 check that there are no users of t_1 that uses t_1 and only t_1. The results are grouped by pairs of tag, and the number of users from $U_2 - U_1$ is given by nb_2 and users from U_1 is given by nb_1. In order to apply additional constraints mentioned in subsection 5.4.2.1 on the results of the query presented in listing 5.4, further processing is required and is presented in algorithm 5.3.

Listing 5.4: SPARQL query to retrieve pairs of tags `?t1` and `?t2`, so that the set of users of `?t1` is included in the set of users of `?t2`.

```
1  SELECT ?t1 ?t2 count(?u1) as ?nb1 count(?u2) as ?nb2
2  {
3     ?t1 scot:usedBy ?u1
4     ?t2 scot:usedBy ?u1
5     ?t2  scot:usedBy ?u2
6     OPTIONAL {
7        ?t1 scot:usedBy ?u3
8        FILTER(?u2 = ?u3) }
9     FILTER(!bound(?u3))
10    OPTIONAL{
11       ?t1 scot:usedBy ?u4
12       OPTIONAL{
13          ?t2 scot:usedBy ?u5
14          FILTER(?u4 = ?u5)
15       }
16       FILTER(!bound(?u5))
17    }
18    FILTER(!bound(?u4))
19 }
20 GROUP BY ?t1 ?t2
```

5.4.2.3 Algorithm

The algorithm 5.3 exploits the query of listing 5.4, which looks for inclusions of sets U_1 and U_2 containing the users of tags t_1 and t_2, so that $U_1 \subset U_2$, with $|U_1| = nb_1$

and $|U_2| - |U_1| = nb_2$. Each result of this query gives $t_1, t_2, nb_1 = |U_1|, nb_2 = |U_2 - U_1|$. The constraints we want to apply on these first results are that we consider sets of more than 1 user, *i.e.* $|U_1| > 1$, and we want a difference in the sizes of U_2 and U_1 of a minimum of 2 users, *i.e.* $|U_2| > (|U_1| + 1)$, which is equivalent to $|U_2| - |U_1| > 1$. Then we go through all results and if the constraints are fulfilled, then tag t_2 is considered to be broader than tag t_1 (in the algorithm we use the reverse formulation, *i.e.* tag t_1 has broader tag t_2 since this formulation corresponds to the definition used in the SKOS model[8]).

To conclude, we should also remark here that, as discussed by Mika (2005), as the case of perfect inclusion (*i.e.* $U_1 \subset U_2$, with $U_1 \cap U_2 = U_1$) is likely not to be so frequent, the recall of this method can be enhanced by considering near-perfect overlap. This can be translated by setting a threshold for an overlap ratio, *i.e.* $U_1 \subset U_2$, with $n < \frac{U_1 \cap U_2}{U_1} < 1$.

Algorithm 5.3 Algorithm to infer *broader* relationships between tags t_1 and t_2.

Require: List of tag-user associations (t_i `scot:usedBy` u_k)
1: $U_i = \{u_k\} \mid t_i$ `scot:usedBy` u_k
2: ***Process***: query of listing 5.4
3: ***Return:*** List $L_{results}$ of results $r(t_1, t_2, nb_1 = |U_1|, nb_2 = |U_2| - |U_1|)$
4: **for all** r from $L_{results}$ **do**
5: **if** $nb_1 > 1$& $nb_2 > 1$ **then**
6: t_1 has broader t_2
7: **end if**
8: **end for**

5.4.2.4 Example results

In table 5.7 we show some examples of *hyponymy* relations inferred between tags from our dataset (see details on the dataset and this computation in chapter 8). We can see that the inferred relationships reflect on the type of methods which is utilized here as the hyponym relations are based on the usage of tags by the users. Hence some relations might seem arguable, or do not correspond to encyclopedic knowledge as what can be found in WordNet for instance. The benefit of this method is that it may help administrators of the knowledge-based system of a community to build a topic structure that is faithful to the interest of the members of the community.

5.5 Conclusion

In order to bootstrap the process of semantic enrichment of folksonomies, we make use of different types of automatic processing of folksonomies. In this chapter we have presented three types of such automatic processing. Table 5.8 summarizes

[8]http://www.w3.org/TR/skos-reference/skos.html#broader

tag t_1	**relation**	**tag** t_2
supermarket	has broader	shopping
creative	has broader	design
cool	has broader	art
nature	has broader	environnement
outils (tools)	has broader	developpement durable (sustainable dev.)

Table 5.7: Example results of semantically linked tags thanks to user-based association rules mining as proposed by Mika (2005) (English translation of french terms in parentheses)

Automatic processing method		String-based	Tag-Tag context sim.	User-based associations
Rel. type	spelling variant	✓		
	related	✓	✓	
	hyponym	✓		✓
folksonomy structure analysis			✓	✓
tag labels analysis		✓		

Table 5.8: Summary of the main features of the automatic processing methods to infer tag semantics

the main features of these three methods and give the semantic relations that they are able to retrieve from the folksonomies. The first string-based method analyzes the label of tags and is able to propose the three types of semantic relations. The second method is based on the analysis of the structure of folksonomies, and measures the similarity for the distributional aggregation in the Tag-tag context and allows proposing *related* relations between tags. The third method presented in this chapter is based on user-based association rules mining and proposes *hyponym* relations.

The first method is a heuristic combination of string-based metrics that we proposed after having benchmarked a series of such metrics. The aim of this benchmark was 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, ie. 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 which is able to retrieve these three types of semantic relations. This heuristic metric performs best for detecting spelling variants, as expected, but also gives interesting results for hyponym relations in cases such as “pollution” which is broader than ”soil pollution”. The *related* relation is however the most difficult to detect from the morphological features of tags, such as in “energy” and “electricity”, which are related but do not share any common lexical root.

We saw in the introduction that string-based methods to infer tag semantics are independent of the structure of folksonomies and are, thus, (a) incremental, as

computation for newly added tags do not require to recompute all tag similarity values for all the tags, and (b) they can be used to link tags across different folksonomies. However, other approaches analyzing the structure of folksonomies are necessary to retrieve semantic relations when tags sharing semantic relations are not morphologically similar.

The second method we presented in this chapter is the cosine similarity computed for the distributional aggregation of tagging data in the Tag-Tag context, as proposed by Cattuto *et al.* (2008). Indeed, this metric has been chosen for its accuracy in detecting *related* relationships between tags, and for its affordable computational cost. This metric consists in first aggregating the three-mode view of folksonomies into two-mode views, following the distributional hypothesis in the Tag-Tag context, which means that tags having similar patterns of co-occurrence, but not necessarily co-occurring together, should be strongly related. Regarding the aggregation of tagging data, we propose a way to compute the co-occurrence of tags with SPARQL queries that exploits our tagging model NiceTag. Then the cosine similarity measure is applied between the vector representations of the tags for this aggregation method. The result is a similarity metric computed for each pair of tags of the folksonomy. Then, when this similarity value is above a given threshold, we generate a semantic annotation stating that both of the corresponding tags are *related* in the sense of the `skos:related`[9] property.

The third method covered in this chapter is the method proposed by Mika (2005) and that looks for user-based association rules. More precisely, this method looks for inclusions of sets of users of tags in order to infer hyponym relations. Indeed, Mika suggests that when, *e.g.*, the set of users of the tag "biological agriculture" is included in the set of users of the tag "agriculture", then we can infer that the tag "biological agriculture" has for broader tag "agriculture". We proposed a method to mine this association rule that exploits the semantic annotations we generated following our model for tagging NiceTag. A SPARQL query is then used to find the pairs of tags that follow this association rule, and the results of this query are then processed and additional constraints on the size of the sets of users are applied in order to avoid meaningless results. The outcome of this method is a set of annotations stating hyponym relations between tags.

We detail in chapter 8 the results we obtained by applying this computational methods on a real world dataset collected from the Ademe agency. These results, plus the sample we already showed in this chapter, show that these methods allow retrieving meaningfull tags' semantics. However, a number of improvement can be made.

Regarding the string-based method, we can maximize the precision by exploiting external termino-ontological resources to avoid computing the similarity values when the pair of tags or concepts at hand is already present in such resources. However, the low recall of this type of method comes from the fact that semantic relations rarely translates into morphological similarity.

[9] `http://www.w3.org/TR/skos-reference/skos.html#related`

For the second and the third method based on the analysis of the structure of folksonomies, a significant number of research work have proposed alternative way of computing similarity of tags that would be worthwhile to test. Some improvements in terms of complexity of the calculation have been proposed in order to reduce the computation time. For instance, Benz *et al.* (2010) showed that it is possible to reach the same level of quality of the inferred semantics by including in the computation subset of the folksonomy corresponding to a specific profile of users who use a high number of tags per post. Similarly, one could also look at other ways of dividing the set of users, by looking at their main center of interest, or, in the context of an organization, by looking at the group they work with, their social networks, etc. Finally, as no golden standard is available to evaluate the quality of automatically inferred semantics, especially for specific fields of knowledge, a large scale qualitative evaluation conducted among experts would provide for a valuable research result, the challenge lying precisely in the capture of the feedback from the users.

CHAPTER 6

Allowing diverging points of view on the semantic structuring of folksonomies

Abstract. This chapter covers the multi-points of view aspect of our approach. We begin this chapter by recalling the relevant works in the literrature dealing with multi-points of view knowledge representations, and we position motivate our contribution. We then present our model, SRTag, that enables us to represent diverging points of view thanks to a flexible manner to reify the semantic relation between two tags. We detail the different versions of the model we proposed before the current one, and we illustrate it with a series of examples of annotations.

Contents

6.1 Introduction

This chapter covers our approach to place the users in the folksonomy enrichment lifecycle. In the previous chapter we presented our approach to bootstrap the process of semantic enrichment of folksonomies. However, the semantic relations that

are proposed by automatic agents that perform this bootstrap may be inaccurate or simply diverge from some user's point of view. It is indeed difficult to find a golden standard against which it would be possible to compare automatically inferred tags' semantics, and moreover, the goal of enriching folksonomies lies precisely in helping the community to obtain a knowledge representation that fits with its needs and vision of the world. This is why the participation of users in the structuring of the folksonomy is a fundamental aspect of our approach, but we also strive to keep this involvement as little cumbersome as possible, and as respectful of the diversity of the communities as possible.

To be able to get the best out of the users' contributions, we propose a multi-points of view approach to folksonomy enrichment. Indeed, the usage and scenario-based analysis of our target community already showed a variety in the level of expertise of its members, and allowing the plurality of voices to express their intended use of tags accounts for the ambiguous and versatile nature of tags in the first place.

To be able to describe different and possibly diverging points of view regarding the semantic relation between tags, we propose a model that consists in reifying this relation so that it becomes possible to capture the agreement or disagreement of a user with this relation. We have investigated in this respect two different approaches to the reification of semantic relations that we present in detail in this chapter. An important point of this investigation is that we wanted to keep the reification mechanism as flexible as possible, and we also wanted to generate statements compatible with unreified statements in order to be able to import external termino-ontological resources. The current version of our model fulfills both of these requirements.

This chapter is organized as follows. We first present in section 6.2 other research works dealing with multi-points of view representations of ontologies or enriched folksonomies. Then in section 6.3 we motivate in detail our approach before presenting extensively the SRTag model aimed at representing multiple and diverging points of view in section 6.4. Finally we conclude in section 6.5.

6.2 Related works

6.2.1 What is a "point of view" ?

In a general sense, a point of view corresponds to a context or a situation where knowledge about an object, or a concept, or an entity are expressed and considered valid and true according to this point of view. Thus, a point of view can be associated to a person, or a group of persons, but also to a theoretical background that defines a frame through which the world is perceived or conceptualized.

Ribière (1999) distinguishes two types of point of view. (1) A point of view *perspective*, which defines a perspective under which a given object is defined with consensual descriptions. For instance, one can describe wind turbines under the technical aspect, describing the different parts of a wind turbines and how they

work together to produce electricity, or under the financial aspect, detailing the cost of a wind turbine and how they can be financed by a country. These different perspectives on a single entity are meant to be complementary and not to contradict each other. (2) On the other hand, a point of view *opinion* corresponds to a non consensual description of an entity. For instance, one can consider wind turbines according to the point of view of the nuclear industry which will highlight their weak efficiency, or according to a given group of ecologists who will focus in the green and renewable energy wind turbines allow producing. These different *opinion* points of view represent partial representations of the world and are not meant to be compatible with each other.

The notion of context is also close to the notion of point of view according to Bach (2006), and to this regard Benerecetti *et al.* (2001) shows three different aspects of the notion of context in the field of knowledge representation. (1) A context reveals a part of a domain and describes a subset of the knowledge pertaining to a given field. (3) A context can also be seen as a specific approximation of a domain, as it can correspond to different level of granularity or abstraction. For instance, one can consider agriculture in a scholar context, relevant to researches in agronomy, or in the farming context. (3) A context can also refer to a given set of external elements such as the location, the period of time. In this way we can talk about the context of "World War Two", or the context of "the french society". Following Bach, we can state that the notion of context defined in this way can be exploited in the design of multi-point of view systems. Hence, a point of view can be said to be bound to a given context.

6.2.2 Multi-points of view knowledge representations

Several works proposed representing knowledge by taking into account different points of view. Ribière (1999) proposed an approach to the design of knowledge based-systems organized with multiple points of view that are grounded on the conceptual graphs formalisms (Sowa, 1984). The basic elements of the conceptual graphs formalism are concepts, relations, concept types, and relation types. Concepts, as well as relations, can be specified in different types, and typed concepts are organized in graph structures thanks to typed relations. The type of concepts and relations have been extended by Ribière to integrate the notion of point of view. Doing so, it is possible to state that a concept is a sub-concept of another concept according to a given point of view. In this case, the sub-concept is called *v-oriented* concept, for "point-of-view-oriented" concept, and the first concept is called *basic* concept. In this framework, a given entity can be instantiated by multiple concepts, which can be v-oriented or basic concepts, this allows bridging different points of view through the instances.

Going further in the specialization of the points of view, Falquet & Mottaz, 2002 set the integration of the notion of point of view at the core of the definition of the concept. A given concept can thus have different definition, each referring to a different point of view. Likewise, the position of each concept in the hier-

archy will depend on a point of view, and this model allows obtaining multiple representations of concepts as well as multiple hierarchies. This model is meant to reflect faithfully the process of the emergence of a consensual ontology where divergences are likely to arise from the starting point when defining the primitives of an ontology. Regarding the process of ontology construction, we can mention here Bachimont (2000) who states that ontologies are not the *end result* of a consensus, but the *place* in which this consensus is being realized. In this respect, Falquet & Mottaz (2002) proposed a conflict resolution process based on operations of comparison of formal concept to help stake holders solve conflicts between their respective definitions.

Bouquet *et al.* (2004) do 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 ontologies. Each ontology is associated to a context and is considered to belong to a local domain. Two local ontologies may overlap, for instance when they share the same object. C-OWL allows global reasoning across local ontologies, the idea being to be able to distinguish what can be shared between local ontologies, and what should remain at a local level. For example[1], the car manufacturing ontology O_{cm} contains the axiom that "a car has only one engine which is either Diesel or Petrol".. Then, Ferrari wants to enrich its ontology O_F describing its production, and to bootstrap the process, imports the car manufacturing ontology O_{cm}. However, in O_F, there is an axiom stating that the engine of a Ferrari is either "F23" or "F34i". In the global ontology, we want to avoid the inference that the Ferrari engine "F23" is of type Diesel, since Ferrari produces only petrol engines. Thus, thanks to C-OWL, some axioms can be kept on the local level, while some concepts (for instance the concept of "car") can be shared across multiple contextualized and local ontologies. The model C-OWL has been applied by D'Acquin (2005) for the design of a semantic web portal dealing with oncology.

Bach (2006) proposed a model that allows building multi-points of view ontologies, MVP-OWL, that is an extension of OWL-DL. Bach adopted the distinction between *perspective* points of view, which are compatible with each other, and *opinion* points of view, which can be contradictory. In his model, the link of subsumption between classes can be defined for a given perspective point of view, so that the global hierarchy across points of view remain logically consistent. Next, instantiation links between entities and classes also belong to points of view. However, the point of view in this case can be of type *opinion*, meaning that the points of view defining instantiations of an entity do not have to be compatible with each other. For example, one point of view can define that the Tricastin's nuclear plant has for type the class "Non-Polluting Power Plant", and another point of view can state that this nuclear plant has for type the (disjoint) class "Polluting Power Plant". Bach also defined additional properties to link classes across the points of view.

[1] http://dit.unitn.it/~fausto/talks/oct.ppt (accessed july, the 28th, 2010)

He defines the link of *equivalence*, which allows stating that two different classes, defined as subclasses according to different points of view, are equivalent. The link of *inclusion* allows stating that a given class, belonging to a given point of view, is included within another class belonging to another point of view. Finally, the link of *exclusion* between two classes belonging to different points of view enables to state that a given instance cannot belong to both classes at the same time. For example, in our nuclear plant illustration, one could define, according to a new point of view, the class "Nuclear Industry" and state that this class is excluded with the class "Non-Polluting Power Plant" (which belongs to another point of view); it would then not be possible for a given entity to be an instance of both of these classes. As a result, the model proposed by Bach allows different experts of a community to define different entities of an ontology in the most relevant way according to their own point of view while guarantying a coherence of the ontology from a global point of view.

In the realm of researches on the socio-semantic Web, several approaches addressed the collaborative structuring or semantic enrichment of tags or concepts used to annotate. Passant and Laublet Passant & Laublet (2008) proposed a model (MOAT) to link tags with their different meanings, which are represented by online resources (URIs) such as Wikipedia articles or concepts available on the Semantic Web. The approach proposed by MOAT allows each user to keep his own point of view since the meaning of a tag is attached to the tagging action of the user. Hypertopic Cahier *et al.* (2005) is an extension of the Topic Maps formalism to take into account multiple points of view. CartoDD (Cahier *et al.*, 2007) uses this formalism to catalogue shared contents with a focus on the collaborative aspect rather than on the formal representation of knowledge. In this approach, each point of view correspond to a specific "perspective" on the field of knowledge, and each point of view has to be logically consistent with the other points of view (for example, a concept cannot belong to several points of view at a time). The approach of Huynh-Kim Bang *et al.* (2008), allowing users to structure tags thanks to a simple syntax, also integrated the possibility for each user to maintain their own point of view. The structuring of the tags in this approach consists in subsumption and synonym relations. In cases of logical inconsistencies between the different points of view, Huynh-Kim Bang *et al.* applied simple rules to prevent the system from displaying meaningless results. For example, when aggregating different user's points of view, cycles in the global hierarchy can occur when tag A subsumes tag B that subsumes tag A. In this case, the system operated iteratively and display first the tag A and then the tag B as a descendant of tag A, but, since tag A is already displayed, the system does not display tag A as a descendant of tag B.

6.2.3 Positioning

In our study, we consider the notion of point of view regarding the semantic structuring of tags. This structuring consists in linking tags with a limited series of semantic relations, namely: *hyponym* relations to states different levels of general-

ity between tags using either *broader* or *narrower*; *spelling variant*, when two tags can be considered as spelling variant of one another; *related*, as when two tags are more loosely related and do not share any hierarchical type of relation.

In the system we propose, a point of view is characterized by the following features:

- Each point of view is associated to a user account, that is, to a single entity. However, these entities can correspond to an abstract entity, like an automatic agent, or a group of persons who share the same point of view, as for instance a "referent point of view" that will correspond to the point of view of the administrators or the group of archivists in the case of the Ademe agency.

- Each point of view consists in a set of statements that we call *semantic actions*, and that express the agreement or disagreement of a user with a statement of a semantic relationship between two tags. For instance, user *A* agrees with the fact that the tag "car" is related to the tag "pollution", but *A* disagrees with the fact that the tag "pollution" is narrower than the tag "car industry", and he thinks instead that the tag "pollution" is related to the tag "car industry". In this respect, our notion of point of view is similar to Bach (2006) or Ribière (1999), in the sense that the tags do not belong to any specific point of view, and a given relation linking two tags is bound to the points of view of the user accounts that approve it.

- Each point of view is logically consistent, *i.e.* a user account cannot agree on two different relations for a the same pair of tags, and a user cannot both approve and reject a given relation between two tags.

- Each point of view is independent from the other points of view, *ie.* all points of view do not need to be compatible with each other. In this respect, we consider *opinion* points of view according to the distinction of Ribière (1999).

Finally, we can discuss our position regarding other approaches anchored within the social and collaborative Web. Our work differs from Passant & Laublet (2008) by specifying the meaning of tags relatively to the other tags of the folksonomy, thanks to a limited set of semantic relations, rather than by linking each tagging assignment to its intended meaning. But our approach does not prevent from linking tags to concepts from termino-ontological external resources when this is relevant to our users. Indeed, we have shown in chapter 4 that our model of tagging, NiceTag, is compatible with MOAT. Furthermore, NiceTag also allows us to directly tag with instances of `skos:Concept` classes from a thesaurus, and the structuring of tags we propose is based on thesauri relations. As a consequence, our system can natively integrate a whole thesaurus and associate it to one point of view by embedding it in a named graph. Cahier *et al.* (2007), who proposed an multi-point of view extension to Topic Maps, have, similarly, integrated the GEMET thesaurus[2]

[2]see http://www.eionet.europa.eu/gemet/index_html

as one of the points of view of the topic map of the CartoDD[3] system. However, our system is meant to support conflicts between points of view, unlike the approach of Cahier *et al.* who consider *perspective* points of view, whereas our approach considers *opinion* points of view that can conflict with each other. Finally, as a complement to the approach of Huynh-Kim Bang *et al.* (2008) that supports diverging points of view, we propose a formal model to represent both the relations and the agreement or disagreement of the users with these relations. As a result, we are able to detect automatically the conflicts from a global point of view and to propose resolutions to these conflicts in order to build a global and logically consistent structuring of the folksonomy.

6.3 Motivation for a multi-points of view approach to folksonomy enrichment

Before going into details about our model for supporting diverging points of view regarding tag semantics, let us briefly recall the main reasons to adopt a multi-points of view approach to folksonomy enrichment.

First, supporting diverging points of view allows each user to organize tags as they wish. This feature is important since empirical studies on the use of tags show that tags are often seen by taggers firstly as a tool to organize their own knowledge base. By allowing each user to maintain his point of view, we account for a core feature of tagging-based systems regarding their usage.

Furthermore, our system goes beyond a user-centric approach by proposing several mechanisms to build a consensual structuring of the tags and to enable each user to benefit from the contributions of the other users. For instance, in the popular social bookmarking service delicious.com, similar functionalities allow users to group tags into bundles. However, delicious.com does not provide any mechanism to share these bundles across users. Our approach is aimed at allowing the sharing of such structured tags.

Taking into account multiple points of view allows us also to take into account the different levels of expertise that are found in our target community. If we take for instance the tags "pollution" and "pollutant", users might not all agree on the semantic relation that should link these two notions. Our approach consists in letting users choose among 4 thesauri-like semantic relations, namely *related*, *broader* or *narrower*, and *spelling variant*. 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, opting for *related*, or some less expert users will even be ready to merge both notions, opting for *spelling variant*, because they are not too concerned about the distinctions that can be made.

In addition, capturing diverging points of view enables to detect tags with mul-

[3] see http://tech-web-n2.utt.fr/dd/?mod=navigation

tiple meanings. Indeed, such tags are likely to be linked to very different tags by different users. For instance the tag "RDF" may be declared to be narrower than the tag "semantic web" for some users, for whom it means Resource Description Framework, or may be placed by some other users as narrower than the tag "African politics", in which case RDF stands for "Rwanda Defence Force". By allowing each user to maintain his point of view, we enable them to focus on their own structuring and understanding of tags, this fostering the emergence of the multiple meanings of a tag.

Then, the experiment we have conducted among 5 users (that we detail in chapter 4) showed that, even among a small set of users, users did not all agree on the semantic relation that should link tags. The global result of this experiment shows that for almost half (46%) of the pairs of tags of the sample dataset used in this experiment, users proposed different semantic relations. Thus, the usefulness of taking into account multiple points of view can be observed even in small groups of users.

Finally, our approach allows a referent user to build a global structuring of the folksonomy that is fed with all these individual contributions. This global structuring can be further exploited in the construction of an in-house thesaurus in which concepts are defined and structured more precisely than tags. For instance, in our "RDF" tag example, this tag will appear to have several broader tags in the global view of the structured folksonomy, and this will help the referent user realize that this tag can have several meanings that can be turned into different concepts of the in-house thesaurus. Thus, our multi-points of view approach to the structuring of folksonomies can help in the construction of more elaborated knowledge representations (as thesauri e.g.) that benefit from the contributions of all members of the community. This is why our approach brings solutions to the classical bottleneck problem in knowledge acquisition, since we give a chance to all users to contribute to the elaboration of a shared knowledge representation.

6.4 SRTag : a model to keep track of diverging points of view

In order to model the semantic structuring of folksonomies while supporting conflicting views, we propose a RDF schema, SRTag[4]. The goal of our model is to describe the semantic relations that may exist between the tags of a folksonomy, and, at the same time, to support conflictual views between the users, or between automatic agents and human users. In this section we present in details the SRTag model, starting with the first version based on standard RDF reification, and then presenting the second version based on named graphs.

[4]: http://ns.inria.fr/srtag/2009/01/09/srtag.html

6.4.1 First version with RDF reification

The first version of our model proposed extending the standard RDF reification of assertions[5] in the case of tags. The idea was to be able to bind users' opinion to statements about the relations between tags, therefore we had to reify these relations. We proposed a RDFS schema (see figure 6.1) in which an assertion on the semantics between two tags of a folksonomy is represented as a RDFS class (`TagSemanticStatement`) that is a subclass of `rdf:Statement`. In standard RDF reification, the object and subject of the reified triple are modeled with the properties `rdf:object` and `rdf:subject`, and these properties find their counterpart in our model with the properties `srtag:tag_object` and `srtag:tag_subject`.

The first version of SRTag also reuses existing ontologies such as SIOC (Bojars *et al.*, 2008) to model the users, or SCOT (Kim *et al.*, 2007) to model the tags or the spelling variant relation. The `scot:Tag` class is subsumed by the `tag:Tag` class from the TagOntology of Newman *et al.* (2005), itself subsumed by `skos:Concept`. This is why we chose the latter class as the range for the properties `srtag:tag_object` and `srtag:tag_subject`, since it consists in the more generic class for tags. Moreover, a user (`sioc:User`[6]), who may also be an automatic agent, may have proposed a semantic assertion (property `srtag:hasProposed`), or approved it (`srtag:hasApproved`), or rejected it (`srtag:hasRejected`). Note also that `srtag:hasProposed` is a subproperty of `srtag:hasApproved`, both properties modeling the agreement of a user with a statement.

The semantic relationships between tags are specified by the subclasses of the class `srtag:TagSemanticStatement` which describes semantic relations between concepts : `srtag:HasNarrower`, `srtag:HasBroader`, `srtag:HasRelated`, and `srtag:HasSpellingVariant`. These semantic relations are those encountered within SKOS or SCOT, except that these relations are now classes instead of properties. However, the semantic relation represented by a reified class can be specified also with the property `rdf:predicate` that link a reified property's class to the corresponding RDF property, such as SKOS subproperties of `skos:semanticRelation` (*e.g.* `skos:broader` for the reified class `srtag:HasBroader`), or such as `scot:spellingVariant` for the reified property's class `srtag:HasSpellingVariant`.

To illustrate the use of this first version of the SRTag model, we show in listing 6.1 the RDF annotation corresponding to the example of statement of figure 6.2. This example states that the tag "environment" has for spelling variant the tag "environmental" (see lines 1-4). This statement has been proposed by an automatic agent who computed the Levenhstein distance between these two tags that is equal to 0.85 (lines 5-6). To illustrate the ability of this model to capture diverging points of view, we see that this statement has been approved by user "John" (lines 9-11) and rejected by user "Paul" (lines 13-15).

As we have already mention, our model considers "opinion" points of view,

[5] see http://www.w3.org/TR/rdf-mt/#Reif

[6] see http://rdfs.org/sioc/spec/

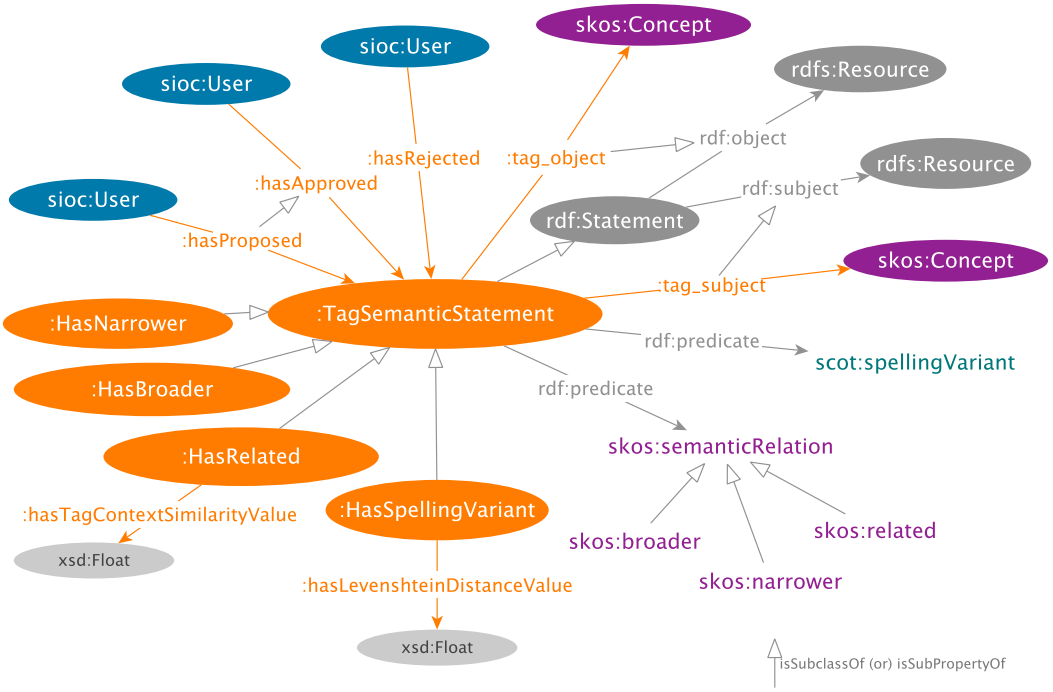


Figure 6.1: First version of the SRTag model based on an extension of the RDF reification class

and this is clearly shown by our choice to model the points of view with a link between the representation of a user and the representation of a statement. The statements describing the relations are thus independent from the points of view, and these statements are generated whenever a user (or automatic agent) proposes a new relation between a pair of tags. Then, each point of view consists in a set of assertions that list the statements he has approved (or has proposed) and the statements has rejected, so that each user's version of the structuring of the folksonomy is kept.

When loading the RDF annotations shown in listing 6.1 in a compliant data store, one can run the SPARQL query shown in listing 6.2 to retrieve the statements made about the tag "environment". Lines 2-8 looks for a statement of type `srtag:TagSemanticStatement` about the tag "environment", and line 9 make sure that the retrieved statements have been approved by the user John.

However, the structuring of the folksonomy is difficult to maintain through reified relations. Indeed, to account for the symmetry of some relations, one has to double each statement to make sure that we retrieve the relation when looking for its symmetric version. For example, if the system contains the relation "environment" is a spelling variant of "environmental", then we want to be also able to retrieve "environmental" has spelling variant "environment", since *spelling variant* is a symmetric relation. The same holds for narrower and broader, except that these relations are inverse of each other. For instance, in the example query shown

Listing 6.1: RDF annotation for the example statement of figure 6.2

```
<srtag:HasSpellingVariant
  rdf:about="http://srtag.ex/statements/spelvar_11">
  <srtag:tag_subject rdf:resource="http://ex.org/tag/environment"/>
  <srtag:tag_object rdf:resource="http://ex.org/tag/environmental"/>
  <srtag:proposedBy rdf:resource="http://ex.org/user/Computer" />
  <srtag:hasLevenshteinDistanceValue>0.85</srtag:hasStringBasedDistanceValue>
</srtag:HasSpellingVariant>

<sioc:User rdf:about="http://ex.org/user/john">
  <srtag:hasApproved rdf:resource="http://srtag.ex/statements/spelvar_11"/>
</sioc:User>

<sioc:User rdf:about="http://ex.org/user/paul">
  <srtag:hasRejected rdf:resource="http://srtag.ex/statements/spelvar_11"/>
</sioc:User>
```

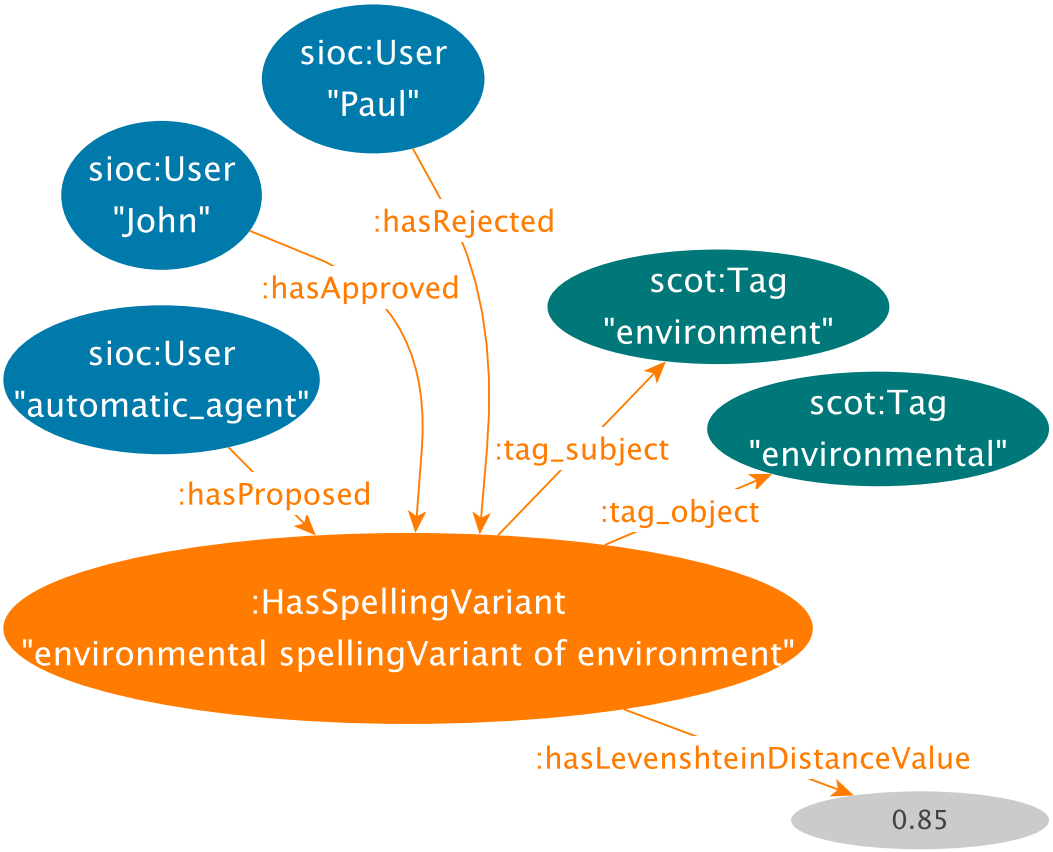


Figure 6.2: Example of a statement with the first version of the SRTag model. The statement 'tag "environment" has spelling variant "environmental"' has been proposed by an automatic agent, approved by user John, and rejected by user Paul. The distance computed by the automatic agent is reported with the property `:hasLevenshteinDistanceValue`.

Listing 6.2: SPARQL query to retrieve semantic statements about the tag "environment" and approved by user "John" with first version of SRTag.

```
1 SELECT * WHERE{
2 ?statement rdf:type srtag:TagSemanticStatement
3 ?statement srtag:tag_subject ?tag1
4 ?statement srtag:tag_object ?tag2
5 ?tag1 a skos:Concept
6 ?tag1 rdfs:label ?l1
7 FILTER (?tag1 =<http://srtag.ex/tag/environment>
8 ?tag2 a skos:Concept
9 ?statement srtag:approvedBy <http://ex.org/user/john>}
```

in listing 6.2, if both tags had been inverted in the RDF annotation (listing 6.1), then the query would have returned no results. The same difficulty is also transferred to the relation between a user and a statement, since we had to make sure that if a user approved a relation, he also approved its symmetric counterpart. This led us to think of a better solution that we present below.

6.4.2 Second version using named graphs

This version of the SRTag model makes use of named graphs mechanisms(Carroll *et al.*, 2005) in conjunction with the mechanism for source declaration proposed by Gandon *et al.* (2007). Named graphs allow for reifying the semantic relationship between two tags without the need to reify it with a corresponding class, as in standard RDF reification. Indeed, the principle of our model is to encapsulate statements about tags' semantics within a named graph (see figure 6.3). Then these named graphs are typed with our class `srtag:TagSemanticStatement` or more precise subclasses (see subsection 6.4.4).

The relationships between tags can be taken from any model, but we chose to limit the number of possible relations to thesauri relations as modeled in SKOS (see figure 5.1 on page 107 in chapter 5). As we no longer reify a property into its corresponding reified class, we can use directly properties from SKOS as they are and benefit instantly from their features, such as their symmetry or their property of being inverse of other properties. We also decided to use the property `skos:closeMatch` to describe the spelling variations of tags instead of `scot:spellingVariant`, and, similarly to the first version of SRTag, to use the class `skos:Concept` as the broadest class for tags. This allows us to use for instance `scot:Tag` for tags freely contributed by users (as in delicious.com for instance), or `skos:Concept` for more controlled tags that would be provided by the archivists of Ademe for example.

Then we modeled a limited series of semantic actions which can be performed on a `srtag:TagSemanticStatement` 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

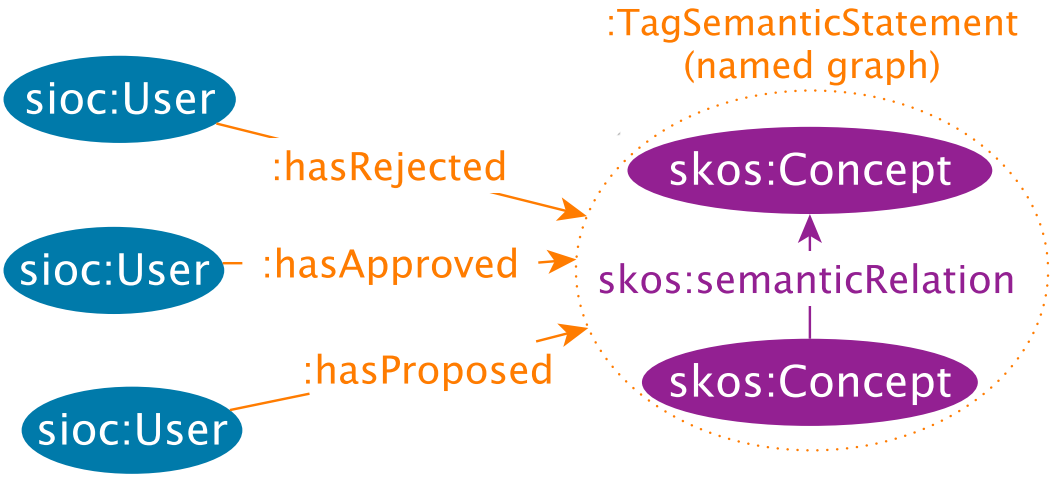


Figure 6.3: Second and current version of SRTag model based on the use of named graphs

asserted relations, which allows us to collect diverging points of view.

6.4.3 Motivation for using named graphs

The choice of using named graphs has several consequences and, we believe, several virtues that we recall briefly here and that we compare with standard RDF reification of other alternatives.

Standard RDF reification method provides reification quads of a statement (the reified property, the subject, the object, and the predicate), but, in RDF, asserting the reification is not the same as asserting the original statement, and neither implies the other. This is particularly problematic in our system, since we want to be able to directly import thesauri so that the statements of an imported thesaurus are homogeneous with the statements of the structured folksonomy. In this case, the whole imported thesaurus will be encapsulated within a named graph and associated to a given point of view. Using named graphs allows us to use the same SPARQL queries to retrieve semantic relations from the structured folksonomy or from an imported thesaurus, since the triples defining the semantic relations follow the same pattern in both cases. Moreover, reification expands the initial triple into a total of five triples (the initial triple plus a reification quad) and the link between the initial triple and its reification quad is not maintained.

As an alternative, the attribute `rdf:ID` can also be used in a property element to (1) produce a reification of the triple that the property element generates and (2) assert it at the same time. However this mechanism remains at the level of triples and there is nothing in the resulting triples that explicitly identifies the original triple and links it to the reification quad. RDF provides no way to associate the subject of the reification triples with an individual triple.

Likewise, statements can be made using the URI of a document as commonly

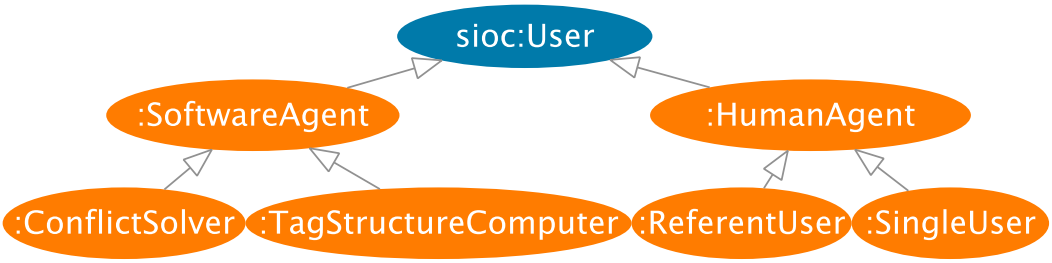


Figure 6.4: Modelization of the types of user in SRTag

done by annotations in OWL. In an ad-hoc application-dependent understanding, those statements could be interpreted as if they were to be distributed over all the statements in the document. But here again we are outside RDF and relying on likening the document to its asserted content does not sound like a good practice. Therefore, nowadays, associating specific URIrefs with specific statements has to be done using mechanisms outside RDF and is one of the motivations behind "identified RDF graphs" in the charter for RDF 2.0.

6.4.4 Modelization of different types of agents and statements

In our approach, different types of agents can be associated to tag semantics statements. We distinguish different types of automatic and human agents according to their role in the life-cycle of the folksonomy (see figure 6.4). We modeled different subclasses of the class `sioc:User` in order to filter tag relations according to the users who approved or proposed it. This includes `srtag:SingleUser` which correspond to regular human 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 on tags, and `srtag:ConflictSolver` corresponding to software agents that propose temporary conflict resolutions for diverging points of view before referent users choose one consensual point of view.

In addition, we also defined a hierarchy of types for the statements that follows the hierarchy of users (see figure 6.5). Each statement can thus be typed according to the type of agent that approved or proposed it. This hierarchy of types of statements also allows defining additional properties that are specific to each statement, as for instance the type and value of the similarity associated to the different type of computation of the tag semantics, as in the case of the `srtag:StringBasedDistanceStatement` with the property `srtag:hasStringBasedDistanceValue`, or the `srtag:TagSimilarityStatement` with the property `srtag:hasTagContextSimilarityValue`.

The connection between the type of user who approved or rejected a given statement, and the type of this statement is realized by setting constraints on the type of value of the properties `srtag:hasApproved` or `srtag:hasProposed` for each type of user. In listing 6.3 we show for instance the definition of the class

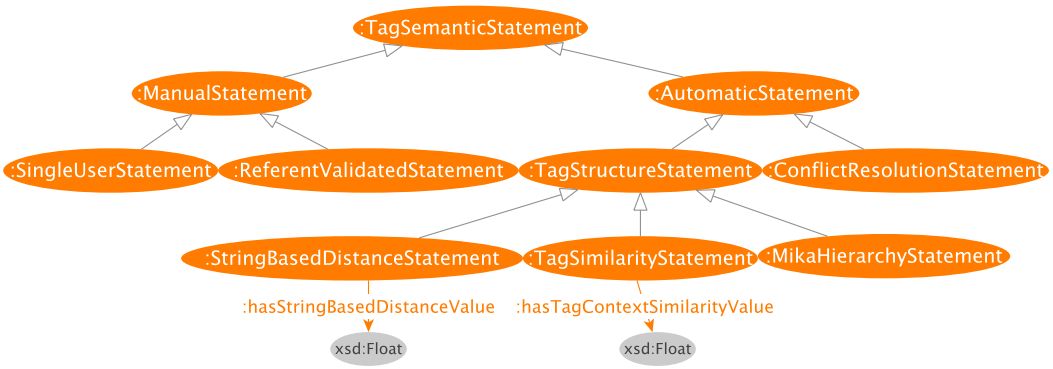


Figure 6.5: Modelization of the types of statements made about tags in SRTag

`srtag:ReferentUser`. In this definition we set a constraint so that all statements that are approved or proposed by a `srtag:ReferentUser` are automatically typed as `srtag:ReferentValidatedStatement`. We will see in the next chapter that this mechanism is useful to recognize with a single lined SPARQL query the type of a statement without the need to check whether or not the corresponding type of user approved it (which would make the query more complex).

Furthermore, as the types of statements are inferred and not explicitly written in the datastore in this case, this mechanism allows us to dynamically update the status of the statements along the life cycle of the enriched folksonomy (see section 4.3 on page 95). For example, when the referent user approves a statement, this statement will be automatically recognized and ignored by the conflict solver in the next passes. And if the referent user happen to change his mind and finally rejects this statement, then the type of the statement will no longer be `srtag:ReferentValidatedStatement` (without requiring to erase or modify the annotation file) and the conflict solver will thus process it.

6.4.5 Example of annotations with second version of SRTag

As we have described the essential aspects of the second version of the model SRTag, let us now illustrate it with a concrete example. In figure 6.6 we show the same example of relation than for the first version of SRTag shown in figure 6.2, that is to say, a statement proposed by the automatic agent and linking the tag "environment" with the tag "environmental" with the property `skos:closeMatch` that model the spelling variant relation. The differences with the first version of the SRTag model is that we are able to utilize directly SKOS properties, and that the original triple between the tag "environment" and the tag "environmental" is also explicitly asserted. This allows the statements we generate to be instantly reusable by other systems unaware of SRTag specific triples[7]. The other noticeable difference is the multi-instantiation of the statements inferred from the rule defined for

[7] in such cases though, there would need to take care of the global coherence of the exported triples, but this aspect will be covered and explained in details in next chapter.

Listing 6.3: Definition of the class ReferentUser in the SRTag model that adds a constraint on the type of statements approved or proposed by the ReferentUser in order to type the corresponding statement merely with an *hasApproved* or *hasProposed* triple on this statement

```
1  <Class rdf:ID="ReferentUser">
2    <label xml:lang="en">Referent User</label>
3    <comment xml:lang="en"></comment>
4    <subClassOf rdf:resource="#HumanAgent"/>
5    <subClassOf>
6      <owl:Restriction>
7          <owl:onProperty rdf:resource="#hasApproved" />
8        <owl:allValuesFrom
9        rdf:resource="http://ns.inria.fr/srtag/2009/01/09/srtag.rdfs#
10               ReferentValidatedStatement" />
11     </owl:Restriction>
12   </subClassOf>
13   <subClassOf>
14     <owl:Restriction>
15       <owl:onProperty rdf:resource="#hasProposed" />
16     <owl:allValuesFrom
17       rdf:resource="http://ns.inria.fr/srtag/2009/01/09/srtag.rdfs#
18               ReferentValidatedStatement" />
19     </owl:Restriction>
20   </subClassOf>
21 </Class>
```

each class of user. Indeed, as the `SingleUser` "John" approved the statement, the system infers (dotted lines in figure 6.6 on the facing page) that this statement is also typed as a `:SingleUserStatement`.

In listing 6.4 we show the RDF annotation corresponding to this example statement. Lines 1-4 corresponds to the initial triple between the tag "environment" and the tag "environmental", plus the declaration of the URI (line 2) of the named graph that encapsulates this triple according to the syntax proposed by Gandon *et al.* (2007) by adding a `cos:graph` attribute value to this triple. Lines 6-11 reuse this URI and type it as being a `srtag:StringBasedDistanceStatement` proposed by the automatic agent "Computer" with the associated similarity value of 0.85. Then the rest of the triples correspond to the diverging opinions of the users. Lines 13-16 correspond to the approval of this statement by the user "John", and lines 18-21 to the rejection by the user "Paul".

By loading this triples in a compliant RDF data store, one can run the query shown in listing 6.5 to retrieve the statements made about the tag "environment" and approved by the user "John". Line 2 look for named graphs that contain a statement between two tags `?tag1`,`?tag2`. Then, lines 3-5 specify the `?tag1`, line 6 specify the type of the second tag, and line 7 makes sure that these statements have been approved by the user "John". Unlike the first version of SRTag, if both tags would have been inverted in the RDF annotation (see listing 6.4), this query would have returned the same results since the property `skos:closeMatch` is symmetric.

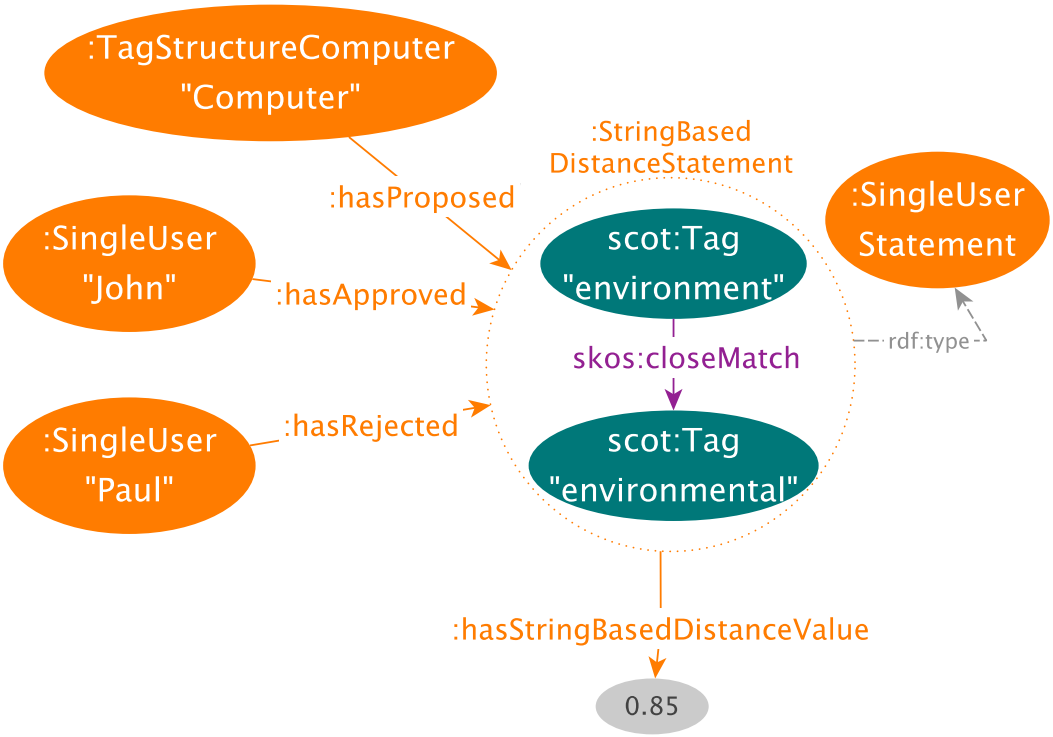


Figure 6.6: Example of a statement with the second version of the SRTag model. The statement "tag 'environment' has spelling variant (modeled with the property `skos:closeMatch`) tag 'environmental"' has been proposed by an automatic agent (typed `srtag:TagStructureComputer`) and has been calculated with a string-based method. Inferred statements are depicted in dotted lines. For instance, in this example, the type `srtag:SingleUserStatement` is inferred from the approval of the statement by user John.

Listing 6.4: RDF annotation for the example statement of figure 6.6

```
<rdf:Description rdf:about="http://ex.org/tag/environment"
cos:graph="http://srtag.ex/statements/spelvar_11">
<skos:closeMatch rdf:resource="http://ex.org/tag/environmental" />
</rdf:Description>

<srtag:StringBasedDistanceStatement
  rdf:about="http://srtag.ex/statements/spelvar_11">
  <srtag:hasStringBasedDistanceValue>0.85
    </srtag:hasStringBasedDistanceValue>
  <srtag:proposedBy rdf:resource="http://ex.org/user/Computer" />
</srtag:StringBasedDistanceStatement>

<sioc:User rdf:about="http://ex.org/user/john">
  <srtag:hasApproved
    rdf:resource="http://srtag.ex/statements/spelvar_11"/>
</sioc:User>

<sioc:User rdf:about="http://ex.org/user/paul">
  <srtag:hasRejected
    rdf:resource="http://srtag.ex/statements/spelvar_11"/>
</sioc:User>
```

Listing 6.5: SPARQL query to retrieve semantic statements about the tag "environment"

```
SELECT * WHERE{
GRAPH ?statement { ?tag1 ?rel ?tag2}
?tag1 a skos:Concept
?tag1 rdfs:label ?l1
FILTER (?tag1 =<http://srtag.ex/tag/environment>
?tag2 a skos:Concept
?statement srtag:approvedBy <http://ex.org/user/john>}
```

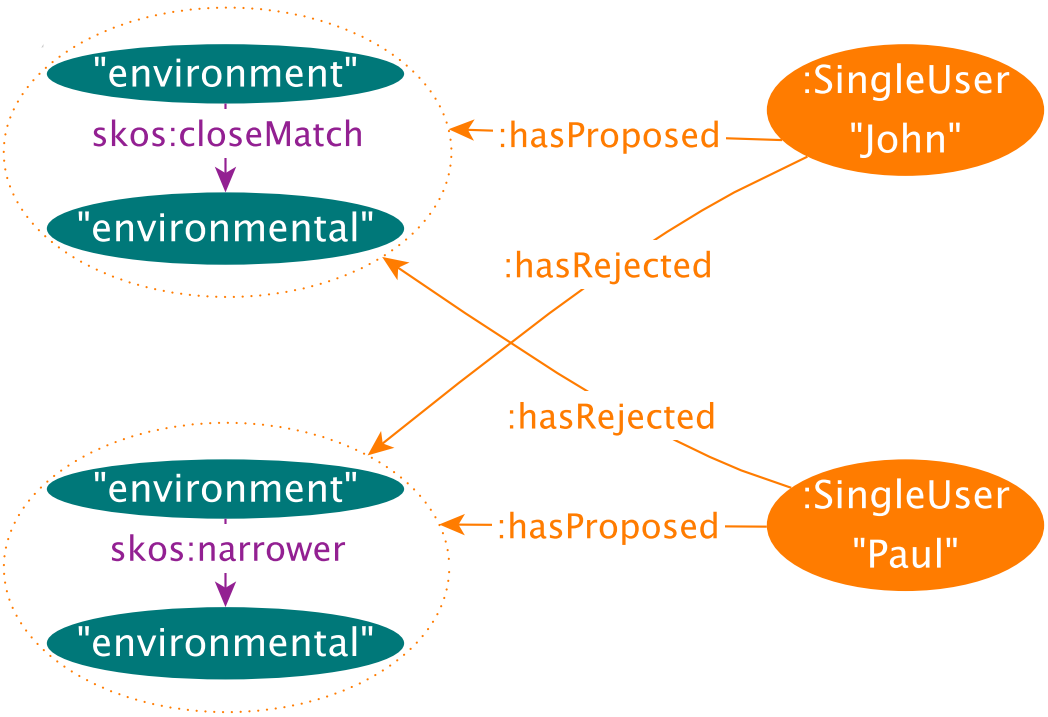


Figure 6.7: Example of diverging points of view captured thanks to SRTag model.

6.4.6 Temporary conclusion: allowing diverging points of view

As a temporary conclusion we can recall that the model SRTag allows capturing each "opinion" point of view about the semantic relations existing between the tags of the folksonomy. We saw in this section that this feature requires the reification of the relation between two tags, and we propose using named graphs for this purpose. Using named graphs allows reifying relations while using directly the initial reified triple. This has two main consequences. The first is that the relations between tags are explicitly stated using SKOS properties, and this eases significantly the compatibility of the structuring of the folksonomy with external thesauri. Second, this mechanism allows keeping the original features of the properties we use, such as their symmetry for instance.

The SRTag model enables us to capture diverging points of view regarding the relations between tags. For instance, as it is shown in figure 6.7, we are able to capture the fact that for the user "John" the tag "environment" is a spelling variant of the tag "environmental", while the user "Paul" believes that the tag "environment" has for narrower tag "environmental". These two statements contradict each other as they state two different relations for the same pair of tags. Nevertheless, our model allows both of these points of view to coexist. We will see in chapter 7 the methods we propose to detect and propose solutions to these cases of conflicts in order to foster the emergence of a global and consensual point of view.

6.5 Conclusion

In this section we have presented a model to capture and represent the points of view of users regarding the semantic relations between tags. Compared to other relevant works proposing multi-points of view representations, our approach consists in capturing *opinion* points of view that are not necessarily compatible with each other from a global perspective. Each point of view consists in a series of formal annotations representing the agreement or disagreement of each each user with a series of semantic relations between tags. Each user's point of view represents the specific structuring of tags according to this user.

The benefits of supporting multiple and possibly diverging points of view on the structuring of tags are the following: (1) This gives users more incentive to organize tags, which are often seen from an individual perspective as a means to organize one's own resources, and to do so as they wish without fearing to "destroy" others' contributions. (2) This also fosters the emergence of richer knowledge representations by giving a chance to each possible meaning or particular understanding of the tags to be expressed, and finally (3) this approach also allows the community to benefit from all the collected individual contributions.

The support of diverging points of view is based on the reification of the semantic relation between tags. In this chapter we have presented the evolution of SRTag, the model we propose to achieve this goal. A first version was based on standard RDF reification, but this option entails an heavy management of approval or disapproval of the different possible relations between tags. To overcome the burden of standard reification, we proposed using named graphs to capture the asserted relations. Each triple representing a semantic relation between two tags is thus encapsulated within a named graph, which is then linked to a user that can approve or reject it. In addition, SRTag allows typing each tag semantics statement according to the type of user who approves it. This feature enables us to track the status of each statement along the lifecycle of the enriched folksonomy, as for example, statement *X* has been first proposed by string-based computational method, then approved by a human user, but not yet by the referent user.

As a results, our model SRTag allows each user to maintain his own point of view. However, from a global point of view, some logical inconsistencies may arise, due to conflicts between some user's points of view. We will see in chapter 7 how this mechanism and the whole SRTag framework is exploited to detect and solve conflicts between diverging points of view, and also to enrich each user's point of view with the contributions of other human or automatic agents.

CHAPTER 7

Combining and exploiting individual points of view

Abstract. In this chapter we are going to go into the details of the module of the folksonomy enrichment in which the different individual points of view regarding the semantics of tags are sorted out and combined together. The goal is to obtain a global and consensual point of view, but also to allow each user to benefit from the other contributions while preserving the logical consistency of their own point of view. We present first in detail the mechanism we propose to detect and solve conflicts that may arise when several relations are stated for the same pair of tags. Then, we explain how this step is exploited to help the elaboration of a referent point of view. The statements approved by the referent user, or the conflict resolutions when the latter is absent, are then utilized in a series of rules that allow enriching each user's point of view with the others' points of view. As a result, we obtain a structured folksonomy in which individual and possibly diverging points of view feed each other and contribute to the emergence of a global and consensual structuring of the tags.

Contents

7.1 Introduction

An important aspect of our approach is to allow each user to maintain his own point of view on the semantic enrichment of the folksonomy. In the first place, automatic agents detect semantic relationships that are used to bootstrap the semantic enrichment of folksonomy by suggesting relations between tags to the users. Then, users of the system can contribute with their own point of view regarding semantic relations between tags. Each user maintain his own point of view by validating or correcting the relations suggested by the system, or by proposing new relations thanks to the interface presented in the previous chapter.

However, these points of view are independent, which means that they can contradict with each other regarding the relations chosen for a given pair of tag. A contradiction arises, for example, when user *A* has approved that the tag "environment" is broader than the tag "pollution" while user *B* disapproved this statement and agreed instead with the fact that "environment" is related to "pollution". The purpose of overcoming the contradictions between individual points of view is twofold:

1. Building a consensual point of view that is logically consistent and that benefits from the contributions of all members of the community. This consensual point of view can then be exploited to help administrators or archivists build a centrally maintained thesaurus. This consensual point of view can also be exploited for the second point.

2. Allowing each user to benefit from the contributions of the other users. Indeed, when two different relations for a given pair of tags have been stated by different users, we need to find a way to choose one of these relations that will be suggested to a user who did not express his opinion for this pair of tag. In this case, the consensual point of view can be exploited to pick one relation among the conflicting ones. Furthermore, when a user has already chosen a relation for a given pair of tags, we also need a set of rules that avoid bringing noise to this user because of other relations that may be chosen by other users for this pair of tags. In such cases, which will be detailed in the remaining of this chapter, a series of rules is applied to maintain a coherent experience for each user.

In our approach to folksonomy enrichment, a specific type of automatic agent first detects the conflicts arising between individual points of view and then proposes a solution to these conflicts. These solutions are then exploited to help a referent user maintaining a global and logically consistent structuring of the folksonomy. Furthermore, the solutions proposed by the conflict solver can be utilized for the relations that the referent user has not already treated. Then, a series of rules are applied when it comes to suggesting related tags to the users. These rules exploit the consensual point of view of the referent user, or the conflict solver when the latter is absent, to allow each user to benefit from the others' contributions when

they contradict with each other. These rules also enable the system to propose coherent and noise-free suggestions of related tags to each user.

This chapter is devoted to going into details on the mechanisms we propose for detecting and solving conflicts, and for combining diverging points of view. In section 7.2 we are going to detail the principle of the detection of conflicts and the proposal of temporary solutions. To evaluate this method we have conducted an experiment on a sample folksonomy with users from the Ademe agency and from the public that illustrate the different situations of agreement or disagreement that are likely to occur in a collective regarding the semantic relations between tags. The results of this experiment are quantified and qualitatively analyzed. Then, we cover in section 7.3 the construction of a consensual and global point of view. In this respect we illustrate this approach with graphic visualizations of the structured folksonomy that include the conflicts and the solutions proposed by the conflict solver. Section 7.4 then gives details about our strategy to exploit the consensual point of view and to efficiently combine the individual points of view in order to allow each user to benefit from other's contributions while preserving a coherence with his own point of view.

7.2 Detecting and solving conflicts

In this section we detail the principles of the conflict solver and report the results of an experiment we conducted on a sample of 94 pairs of tags for which we asked users of Ademe and the public to choose a semantic relation. These results are statistically and qualitatively analyzed. In particular, this experiment shows that users are likely to not necesssarily agree on the semantic relations between tags, and that some types of tags are more likely to be a source of conflicts.

7.2.1 ConflictSolver mechanism

In the SRTag model, we introduced another type of automatic agent, which is 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 human users on the same pair of tags (if a user changes his mind, we simply update his point of view so that it remains logically consistent). 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". In addition, other users can say that "pollution" is related to "co2". Hence, by allowing each user to maintain his own point of view, several relations can be stated for the same pair of tags. The conflict solver's task is to detect these cases and to suggest a solution.

The conflict solver mechanism works in two steps. First a SPARQL query, shown in listing 7.1, looks for pairs of tags linked with more than one relation. In

Listing 7.1: SPARQL query to detect contradictions, *i.e.* when two tags are linked with different relations

```
1  PREFIX srtag:<http://ns.inria.fr/srtag/2009/01/09/srtag.rdfs#>
2  PREFIX skos:<http://www.w3.org/2004/02/skos/core#>
3
4  SELECT * count(?uS1) as ?nbS1
5  WHERE{
6  GRAPH ?s1 { ?tag1 ?rel1 ?tag2}
7  GRAPH ?s2 { ?tag1 ?rel2 ?tag2}
8  FILTER (?s1 != ?s2)
9  {
10  {?s1 a srtag:TagStructureStatement }
11  UNION   {?s1 a srtag:SingleUserStatement}}
12 {
13  {?s2 a srtag:TagStructureStatement }
14  UNION   {?s2 a srtag:SingleUserStatement}}
15 FILTER(?tag1 < ?tag2)
16 ?rel1 srtag:incompatibleWith ?rel2
17 ?tag1 a skos:Concept
18 ?tag2 a skos:Concept
19 OPTIONAL{
20    ?uS1 srtag:hasApproved ?s1
21    ?uS1 a srtag:SingleUser}
22 OPTIONAL{
23    ?uS2 srtag:hasApproved ?s2
24    ?uS2 a srtag:SingleUser}
25 OPTIONAL{
26    GRAPH ?sRelated {?tag1 skos:related ?tag2}}
27 OPTIONAL{
28    GRAPH ?sReferent { ?tag1 skos:semanticRelation ?tag2}
29    ?sReferent rdf:type srtag:ReferentValidatedStatement}
30 FILTER(!bound(?sReferent))
31 }
32 GROUP BY ?s1
```

this query lines 6-18 looks for a distinct (line 15) pair of tags `?tag1`, `?tag2` on which more than one statement (lines 6-8) has been proposed. This implies that both tags (whose possible classes are subclasses of `skos:Concept`, see lines 17-18) are linked with at least two incompatible relations (line 16). Then this query (lines 19-24) allows counting the number of human users, modeled as `srtag:SingleUser`, who have approved or proposed each relation. Lines 25-26 retrieves, if it exists, the statement corresponding to the *related* semantic relation, since this relation will be used as a compromise if no clear consensus is met for another type of relation (see algorithm 7.1). Then lines 27-30 make sure that this pair of tag has not already been treated by the referent user. If this is the case, the conflict solver ignores this pair of tags. Finally, line 32 allows counting the number of approval for each conflicting statement.

To model the fact that two relations linking the same pair of tags conflict with each other, we have added a property to the SRTag model, `srtag:incompatibleWith`. This custom property is then used to state that the four properties we use to describe the semantic relations (namely `skos:related`, `skos:broader`, `skos:narrower`, and `skos:closeMatch` –used for the spelling variant relation–) are incompatible with each other, so that `skos:related` is incompatible with `skos:narrower` for example. Indeed, we avoided the use of a simple SPARQL filter rule (as `FILTER(?rel1 != ?rel2)`) since some distinct properties in SKOS can be compatible in our system, such as `skos:narrower` and `skos:narrowerTransitive`.

It is important to note here the role played by the modelisation of the different types of agent in our system. In the SRTag model, each statement can be typed according to the type of agent who approved or proposed it. For instance if a statement is approved by a `SingleUser`, then the corresponding statement is typed `SingleUserStatement`. Likewise, if a statement has been proposed by automatic agents in charge of the computation of the semantic relation, it is typed as a `TagStructureStatement`. In the SPARQL query of listing 7.1, we look for pairs where different relations have been proposed or approved by human users or automatic agents (lines 9-14). However, in the resolution of the conflict, we count only the number of human users who approved a relation (see lines 19-24). Hence, we give priority to the opinion of human users over the proposal of automatic agents since if at least one human user has proposed a different relation than an automatic agent, then the conflict solver will propose the human user's relation as a solution.

Algorithm 7.1 processes the list of conflicting pairs given by the SPARQL query of listing 7.1. In cases of conflict on a given pair of tag, the solver first counts the number of approval $nbApp_i$ for each conflicting statement $s_i \varepsilon \{s_i\}_n$, n being the total number of statements made on a given pair of tags. Then, it retrieves the maximum $max\{nbApp_i\}_{i\varepsilon[1,n]} = nbApp_{max}$, and compares the ratio $r = \frac{nbApp_{max}}{\sum_n nbApp_i}$ 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.

Algorithm 7.1 Conflict solver algorithm

Require: threshold ratio for consensus τ_{cs}
Require: List L_t of pairs of tags (t_1, t_2) with a set of conflicting statements $\{s_i\}_{t_1,t_2}$
1: $|\{s_i\}_{t_1,t_2}| = n_{t_1,t_2}$
2: **for all** distinct pairs of tags (t_1, t_2) of L_t **do**
3: **for all** conflicting statements s_i of $\{s_i\}_{t_1,t_2}$ **do**
4: count number $nbApp_i$ of approval of s_i
5: retrieve max value $nbApp_{max} = max\{nbApp_i\}_{n_{t_1,t_2}}$
6: **end for**
7: **if** $\frac{nbApp_{max}}{\sum_{n_{t_1,t_2}} nbApp_i} > \tau_{cs}$ **then**
8: `ConflictSolver` approves s_i
9: **else**
10: `ConflictSolver` approves t_1 is related to t_2
11: **end if**
12: **end for**

7.2.2 Protocol of the experiment

We have conducted an experiment with 5 users among which 3 were members of the Ademe agency, and 2 were persons working in ecology-related areas that would typically fall in the category of users consulting the documents of Ademe made available to the public. 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 (this questionnaire is shown in annex B on page 257). In addition to these 5 points of view, we have integrated in this experiment the relations proposed by the automatic agents for the pairs of tags of this experiment (these statements correspond to the results of the computation detailed in chapter 7 and chapter 8). This set of relations proposed by automatic agents cover 32 pairs of tags out of the 94 pairs of the dataset.

When a user has chosen one of the first four possibilities, *i.e.* spelling variant, or broader, or narrower, or related, we say that this user has approved the corresponding statement. When a user chose the fifth possibility, *i.e.* that t_1 is not related to t_2, we have applied a 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 enables 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. For example, if a user has chosen that two tags are not related at all, and that a second user has chosen that these two tags are spelling variants, then the rule will allow us to infer

that the first user has rejected the spelling variant relation approved by the second user.

The outcome of the responses of the users is a series of semantic annotations about the opinions of the users on a series of statements. These statements correspond each to a specific semantic relation between the tags of the 94 pairs of the dataset. We have then applied the conflict solver on this set of relations and points of view. 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 and never rejected by any user. Indeed, as the questionnaire allowed stating that two tags of a pair are unrelated, and since we have translated this choice as being a rejection of all the other possible relations, some relations can be approved or rejected by some users.

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 has been stated about a tag, but has been both approved by some users and rejected by some other users.

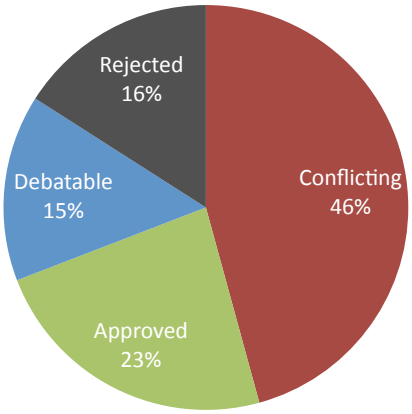
4. Rejected statements : when a relation has only been rejected. This case corresponds to pairs of tags for which all users (that expressed themselves, as some users have not picked any choice for some pairs of tags) have picked the unrelated choice.

In this experiment, we chose a threshold value for the resolution of the conflicts equal to $2/3$.

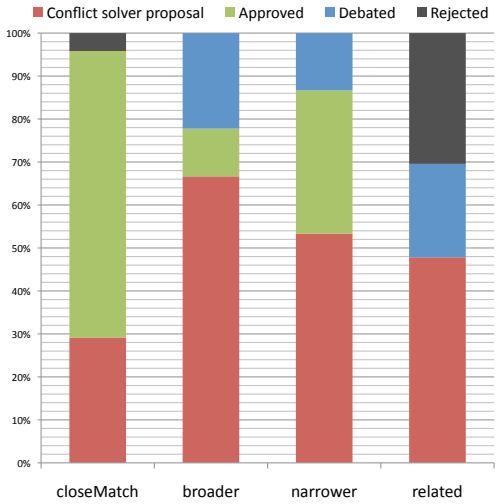
7.2.3 Statistical results analysis

In figure 7.1 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 thing we shall notice is that we are far from having a consensus on all the relations chosen by the users. This comes as evidence of the usefulness of taking into account the multiple points of view that may arise among the members of a community. Indeed, by doing so we are able to see the emergence of conflicting points of view, but also to distinguish different situations of agreement or disagreement as accounted by the different cases of conflict solving that we detail in the following.

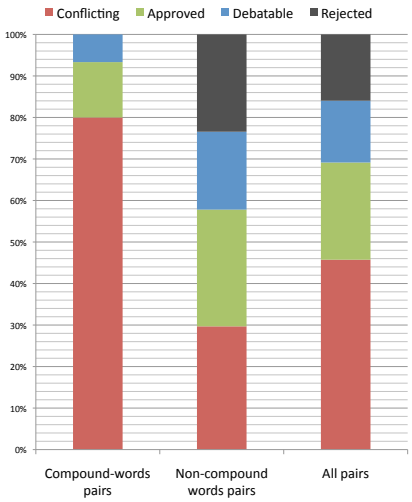
Global distribution. The first chart (a) shows the distribution of the different cases of conflict solving over the 94 pairs of tags. We see that almost half of the



(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.

Figure 7.1: Result of conflict solving

pairs (46%) are counted as conflicting since several relations have been proposed for these pairs. 23% of the pairs are linked with a single relation that has only been approved, while 15% of the pairs share a single relation that has both been approved and rejected by users. Finally 16% of the pairs of tags have a relation that has only been rejected.

Influence of the types of semantic relations. 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 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 it is either involved in conflicts (48% of the statements) or debatable (52% of the statements). We should note here 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 *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* in 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. Indeed, 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, opting for *related*, or will even be ready to merge both notions, opting for *spelling variant*, because they are not too concerned about the distinctions that can be made.

Influence of the form of tags. 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 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.

7.2.4 Conclusions on the conflict detection

This section has covered our method to first detect pairs with several conflicting relations, and then to propose a resolution based on evaluation of the degree of consensus reached by the most frequent relation. If no consensus has been reached regarding one specific relation among the conflicting ones, then the conflict solver proposes the *related* relation as a compromise.

The specific strategy to propose a solution for the pairs with conflicting relation can be parameterized by the administrators of the system. It is indeed possible to set a different threshold for the ratio above which a relation is considered to be consensual, so that it is possible to opt for a strict majority-based policy with a threshold value of 0.5 for instance. It is also possible to replace this criterion by some other one not necessarily based on such voting approach. We should also remark here that the core of our contribution in this regard lies in the fact that we capture each users point of view with a formal model that allows us to type each point of view according to the type of user who approved or rejected it. As a result we are able to detect the conflicts between the different points of view with a single SPARQL query as shown in this section.

The experiment we conducted with a set of real users showed that even with a small set of users, we already observed a significant amount of direct conflicts (several relations for a single pair of tags) and also several debatable statements, *i.e.* statements both approved and rejected by users. This result shows the usefulness of taking into account multiple points of view in the semantic enrichment of folksonomy, since a consensus is rarely met (roughly less than a fourth of the cases) regarding the semantic relation between two tags. Next, we found that *hyponym* and *related* relations were more often conflicting with other relations than the *spelling variant* (close match) relation. Pairs of tags involving a compound word of one of the other tag of the pair seems also more likely to be at the origin of debatable or conflicting semantic relations. Now we are going to see how the outcome of the conflict solver can be exploited to build a consensual and global point of view.

7.3 Creating a consensual point of view

This section covers the creation of a consensual point of view for which we exploit the results of the conflict solver. We first see how we can construct visualizations of the structured folksonomy that include conflict resolutions and that can then be utilized by the referent user to build a global and coherent point of view.

Listing 7.2: SPARQL query used to retrieve all the relations of the structured folksonomy

```
1  SELECT  * count(?uApprove) as ?nbApprove count(?uReject) as ?nbReject
2  WHERE{
3  GRAPH ?s {?tag1 ?rel1 ?tag2}
4  ?s rdf:type ?s1type
5  ?s rdf:type srtag:TagSemanticStatement
6  {{?tag1 rdf:type scot:Tag} UNION {?tag1 rdf:type svic:MC}}
7  {{?tag2 rdf:type scot:Tag} UNION {?tag2 rdf:type svic:MC}}
8  ?tag1 rdfs:label ?tag1l
9  ?tag2 rdfs:label ?tag2l
10 FILTER(?tag1l <= ?tag2l)
11 OPTIONAL{
12   ?s srtag:hasRelationWeight ?weight}
13 OPTIONAL{
14   ?uApprove srtag:hasApproved ?s
15   ?uApprove rdf:type srtag:SingleUser}
16 OPTIONAL{
17   ?uR1 srtag:hasRejected ?s1
18   ?uR1 rdf:type srtag:SingleUser}
19 OPTIONAL{
20   graph ?sConflictResolution {?tag1 ?rel2 ?tag2}
21   ?sConflictResolution a srtag:ConflictResolutionStatement}
22 }
```

7.3.1 Visualization of the structured folksonomy

The construction of a referent and global points of view suggests the use of a global visualization that will allow the referent user to browse the whole structured folksonomy. This global visualization exploits the outcome of the Conflict Solver and is generated thanks to the results of the SPARQL query shown in listing 7.2. In this query, lines 3-5 look for statements `?s` of type `srtag:TagSemanticStatement` and lines 6-7 make sure that these statements are made on tags of type `scot:Tag` or `svic:MC`. Then we retrieve the labels of the tags in lines 8-9, and we avoid getting twice the same pair of tags in line 10. Line 12 allows retrieving the weight of the relation when available, *i.e.* when the relations of the retrieved statement has been proposed by an automatic agent in the first place. Lines 13-15 allows counting the number of approval by human users of this relation, and lines 16-19 the number of rejection. Finally, lines 19-21 allow us to detect whether the current relation is conflicting with another one since, in this case, we can find another statement made on the same pair of tags but with a different relation than the current one. Furthermore, the type of the statement retrieved in line 4 indicates us if the current statement happens to be the proposal of the Conflict `Solver`. The numbers of approvals and rejections are calculated in line 1 as `?nbApprove` and `?nbReject`.

The global map is then constructed by drawing a graph whose nodes are the tag linked with at least a semantic relation, and the edges are the relations that are retrieved thanks to the query of listing 7.2. A color code is attributed to each edge

corresponding to the situation of the relation regarding its type or the number of users who approve or proposed it:

- if a relation has only been approved, ie if `?nbApprove`> 0 and `?nbReject`$= 0$, then it is drawn in green
- if a relation has both been approved and rejected, *i.e.* if `?nbApprove`> 0 and `?nbReject`> 0, then it is drawn in blue
- if there exists another conflicting relation for the same pair of tags, *i.e.* if `?sConflictResolution` is not an empty string, then it is drawn in red
- if the type of the relation is equal to `srtag:ConflictResolutionStatement`, then it is drawn in orange
- finally, when a relation has only been rejected, if `?nbApprove` $= 0$ and `?nbReject` > 0, then it is drawn in black

In figures 7.3 and 7.2 we show some samples of the graph that we are able to build thanks to the conflict solver's output. In these graphs, all the tags that share a relation are linked. For each pair of linked tags, we display the different relations that exist. As we have seen it above, each relation can have different status that is represented with a color. Tags in our experiment were of two types: in blue we represent controlled tags, *i.e. tags* that have been proposed by Ademe's archivists, and in green we represent free tags, *i.e.* tags that have been proposed by regular members of Ademe or by external users. The relation *rel* indicated on each arrow should be read "has for *rel* tag", so that *e.g.,* the tag "energie" has for *narrower* tag "energie renouvelable" (this conforms to the convention adopted in SKOS[1] where `skos:narrower` is labelled "has narrower").

For each sample of the structured folksonomy, we also present the corresponding table (tables 7.1 and 7.2) that shows the detail of the relations for each pair of tags and, for each relation, the number of users who approved or rejected it. If we look, in figure 7.2, at the pair of tags "energie" (energy) and "energie renouvelable" (sustainable energy), we see that a clear majority of the users (4) approved that "energie" has narrower tag "energie renouvelable" while only 1 user approved the *related* relation. Consequently, the conflict solver chose the relation *narrower* as a conflict resolution. However, if we look at another pair with conflicts in figure 7.3, *e.g.* "agriculture durable" (sustainable agriculture) and "agriculture raisonnee" (responsible farming[2]), we see that the situation is not as clear as in the previous example. In this case, 1 user approved the *spelling variant* relation, 2 users approved the *narrower* relation, and 2 users approved the *related* relation. Since no clear majority can be drawn for this pair of tag, the conflict solver proposed the *related* relation as a compromise.

[1] http://www.w3.org/TR/skos-reference/skos.html#narrower

[2] However the translation in English is not trivial since it is a much debated term, as this discussion illustrates it : http://forum.wordreference.com/showthread.php?t=665386

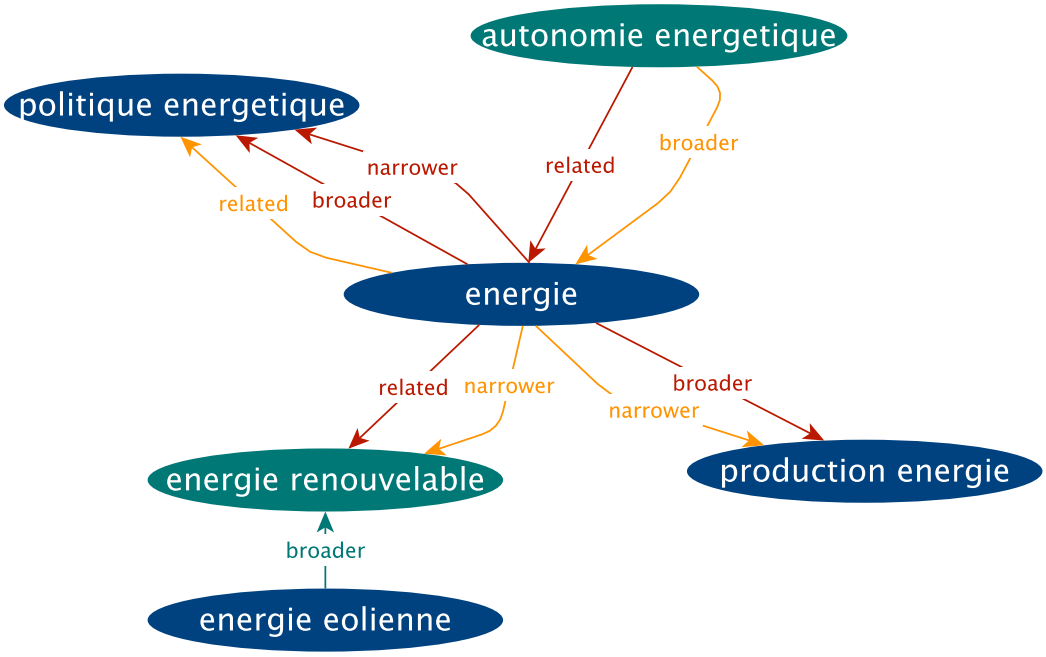


Figure 7.2: Sample of the structured folksonomy graph for the controlled tag "energie"

Source	Target	Relation	Nb Approve	Nb Reject	Status
energie	politique energetique	broader	1	0	conflicting
		narrower	3	0	conflicting
		related	**1**	**0**	**conflicting**
energie	production energie	broader	1	0	conflicting
		narrower	**4**	**0**	**conflicting**
autonomie energetique	energie	**broader**	**3**	**1**	**conflicting**
		related	1	1	conflicting
energie	energie renouvelable	**narrower**	**4**	**0**	**conflicting**
		related	1	0	conflicting
energie eolienne	energie renouvelable	broader	5	0	approved

Table 7.1: Table reporting the number of approval and rejection for the relation of the example graph of figure 7.2 (relation proposed as a solution by the conflict solver in bold characters)

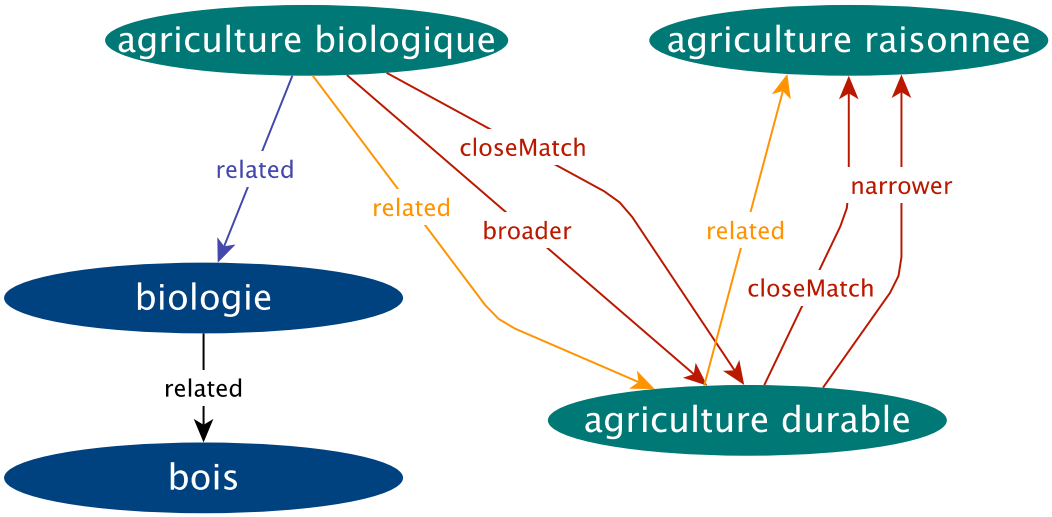


Figure 7.3: Sample of the structured folksonomy graph for the tag "agriculture biologique"

Source	Target	Relation	Nb Approve	Nb Reject	status
agriculture durable	agriculture raisonnee	spel. var.	1	0	conflicting
		narrower	2	0	conflicting
		related	**2**	**0**	**conflicting**
agriculture biologique	agriculture durable	broader	3	0	conflicting
		spel. var.	1	0	conflicting
		related	**1**	**0**	**conflicting**
agriculture biologique	biologie	related	1	3	debatable
biologie	bois	related	0	5	rejected

Table 7.2: Table reporting the number of approval and rejection for the relation of the example graph of figure 7.3 (relation proposed as a solution by the conflict solver in bold characters)

7.3.2 Constructing the referent point of view

In the SRTag model, we introduced another type of human agent modeled with the class `srtag: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 his own and consensual point of view. The conflict solver mechanism will assist the referent user in his task by pointing out the conflicts already existing in the structured folksonomy. As we have seen it above, it is possible with the visualization of the structured folksonomy to highlight the conflicting pairs, or the relation that are only rejected and are, thus, candidates to be removed. Then, all the statements that the referent user has already treated will be ignored in further passes of the `ConflictSolver`. In figure 7.4 we show similar graphs as in figures 7.3 and 7.2 but showing the choices of our referent user, with purple arrows for

relations chosen by the referent user, and grey doted lined arrows for the relations he rejected.

7.3.3 Visualization of the points of view as layers

It is also possible to look at the structured folksonomy and the points of view that coexist with the metaphor of the layers, as suggested by figure 7.5 on page 183. At the bottom of this figure lies the set of tags with no semantic relations. Then each user is associated with a layer that contains all the relations that he has approved. In this figure, the user Paul has approved the following relations: "environment" has narrower "pollution", and "pollution" has narrower "pollutants". Then, John has rejected both relations that Paul had approved, and instead he proposed the relation "pollution" has narrower "environment". This is why we do not see in John's layer the second relation approved by Paul. Then, the conflict solver proposed, as a solution to the conflict for the pair of tags "pollution" and "environment", the *related* relation. Then the referent user, whose layer is represented at the top, has approved the conflict solver's solution and the relation, proposed by Paul, "pollution" has narrower "pollutants". Now we are going to see how these diverging points of view and the referent point of view are exploited and filtered to offer a coherent experience to all users.

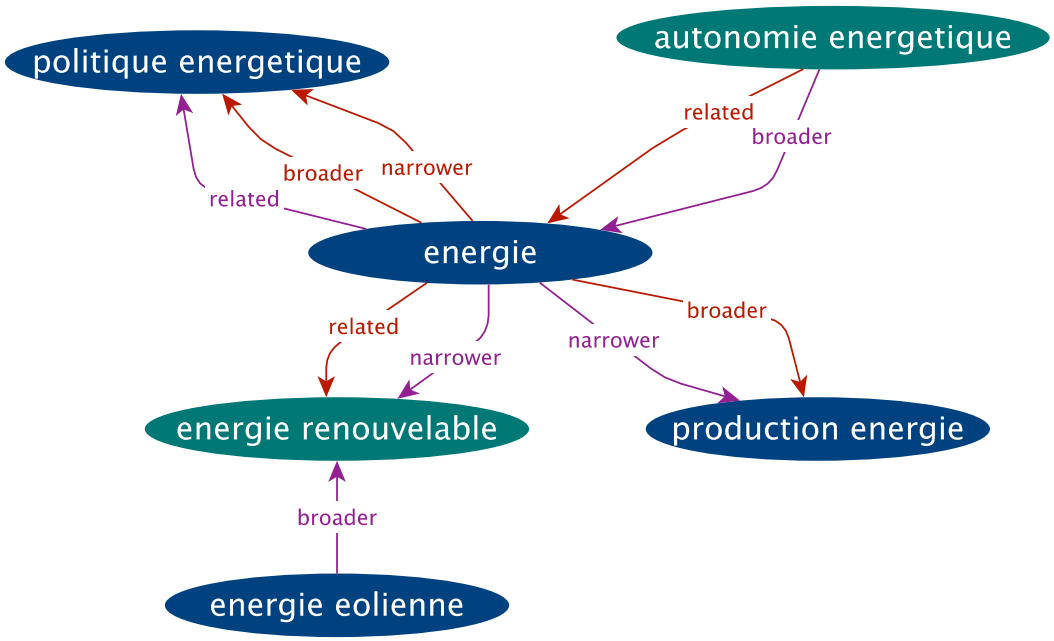
7.4 Exploiting and filtering points of view

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 allow each user to benefit from the points of view of the other users while preserving a local coherence for each user.

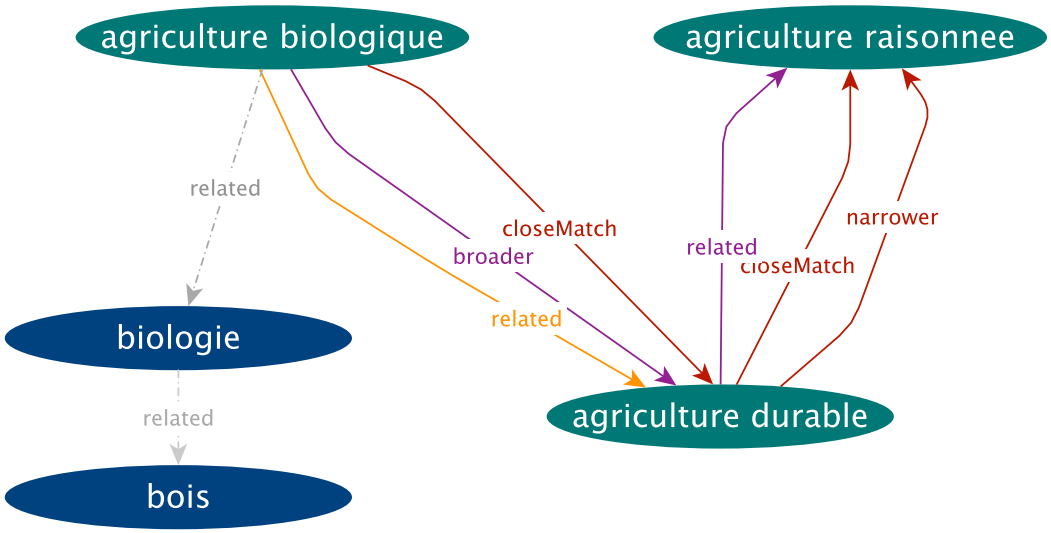
7.4.1 Principle

The idea of this module of our system is to enrich the points of view of each user with the other points of view. Indeed, each point of view is made of the statements that each user has proposed or approved, and this set of statements is likely to be limited. Our purpose now is to set up a strategy to include statements from other agents (the referent user, other human agents and automatic agents) that are not contradictory with the point of view of the user we consider. The type of contradiction here is the same type that we saw above for the conflict solver mechanism, *i.e.* a statement is in conflict with another statement when it asserts, for the same pair of tags, a different relation than the second statement.

Thanks to the SRTag model, we are able to keep track of the type of agents associated to each statement. Thus, we are able to give a priority to the statements that are integrated within a user's point of view according to the type of agent to



(a) Referent choices for the tag “energie”



(b) Referent choices for the tag “agriculture biologique”

Figure 7.4: Sample of the structured folksonomy showing the **choices of our referent user** (**purple** arrows for approved relations, and **grey doted lined** arrows for rejected relations).

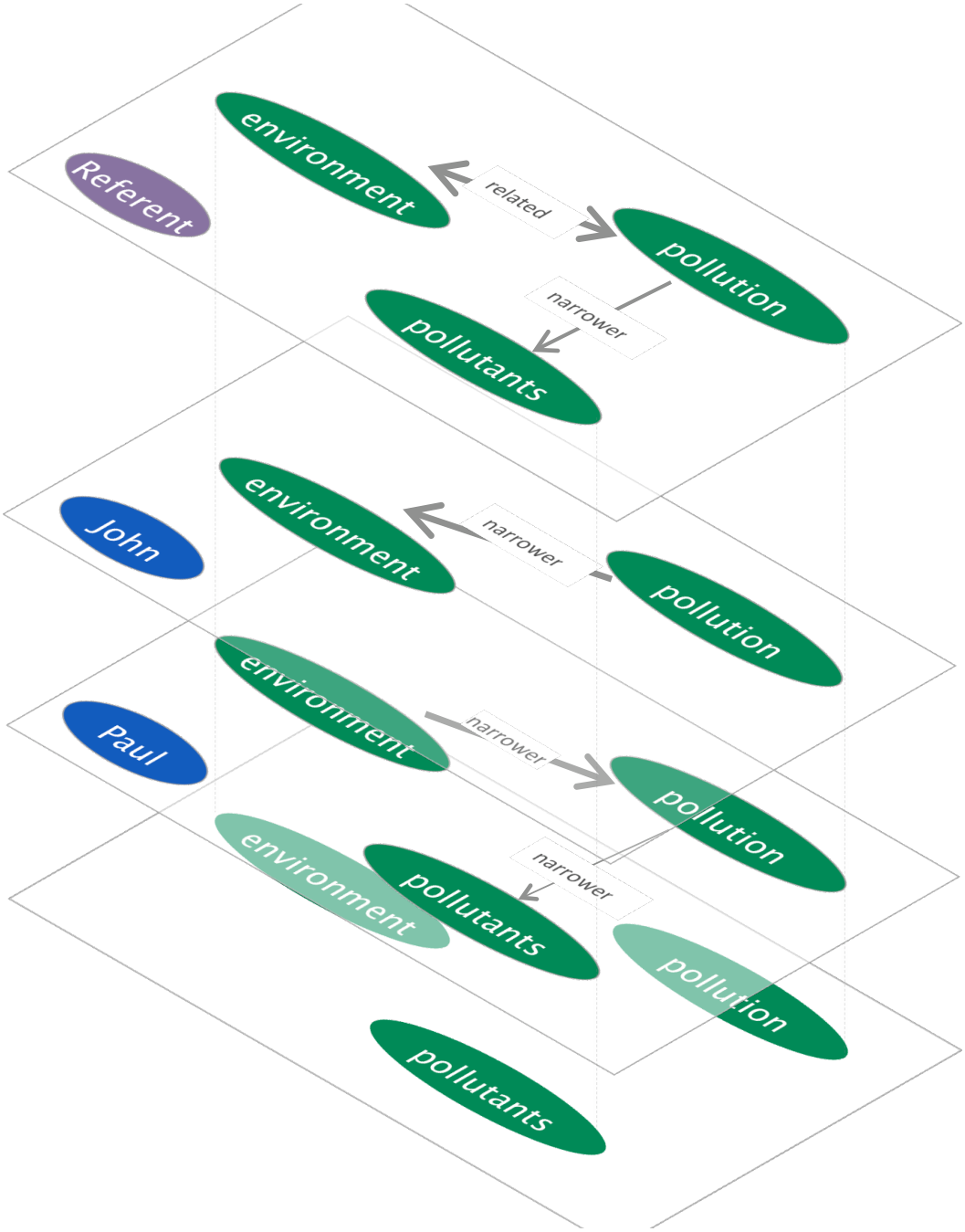


Figure 7.5: Representation of the structured folksonomy with layers: each layer correspond to a user's point of view, and the referent user's point of view is made after all individual contributions

which they are associated. The priority order we follow, when integrating others' point of view to the point of view of user u, is given below:

1. First, we integrate the set of statements S_u approved by the user u.

2. Then, we integrate the set of statements S_{ru} approved by the `ReferentUser`, except if they conflict with one statement from S_u.

3. Then, we integrate the set of statements S_{cs} approved by the `ConflictSolver`, except if they conflict with one statement from S_u or S_{ru}.

4. Then, we integrate the set of statements S_{ou} approved by other users (`SingleUser`), except if they conflict with one statement from S_u, S_{ru}, or S_{cs}

5. Finally, we include the set of statements S_{tc} approved by automatic agents (`TagStructureComputer`), except if they conflict with one statement from S_u, S_{ru}, S_{cs}, or S_{ou}.

Hence, we see that the referent point of view (or the conflict solver point of view when the latter is absent) is crucial in this module when integrating statements from other users about a pair of tags on which the current user has not approved any statement. Indeed, it allows choosing one `SingleUserStatement` when several such statements have been proposed on this pair of tags. Regarding statements from the TagStructureComputer, this situation cannot occur since only one statement is proposed for a given pair of tags by the automatic agent.

A typical scenario of application of this strategy can be found when suggesting to a user tags semantically linked to a searched-for tag. This scenario is detailed below.

7.4.2 Application to the suggestion of semantically linked tags

We are now going to illustrate the principle of the integration of points of view when suggesting to a given user tags semantically linked to a searched-for tag. In this case, 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 linked to the searched-for tag.

We detail this process by going trough a concrete example. In this example the user "claire" is looking for tags linked to the tag "environnement". In table 7.3 we show the tags linked to the tag "environnement" with the relations approved by other users, by the referent user, by the conflict solver, or by the automatic agent. This table is meant to help the reader understand the results of each query. To summarize all the results, we report in table 7.4 on page 189 all the tags and the corresponding relations given by each query.

tag linked to "environnement"	relation	approved by
grenelle de l'environnement	narrower	**claire**, delphine, monique, alex
	related	referent
compentence environnementale	narrower	**claire**
	related	referent, alex
preoccupations environnementales	narrower	**claire**, delphine
	related	referent, alex
domaines environnementaux	related	referent
	spel. var	conflict solver, monique, alex
environmental	spel. var.	auto. agent
environment	spel. var.	auto. agent

Table 7.3: Tags linked to the tag "environnement". We indicate the tags linked according to the point of view of user "claire" and other users, and according to the referent user, the conflict solver, and the other automatic agent that performs calculations for bootstrapping.

7.4.2.1 First step

If the user "claire" is searching for the tag "environnement", the system will first suggest tags coming from assertions she has approved thanks to the query shown in listing 7.3. Lines 3-4 look for statements made on the tag "environnement", and lines 5-6 make sure that these statements have been approved by user "claire". In this case, this first query will return the three tags and corresponding relations according to the statements approved by user "claire", namely "grenelle de l'environnement" as a narrower tag, "competence environnemental" as a narrower tag, and "preoccupations environnementales" as a narrower tag.

Listing 7.3: SPARQL query used to retrieve statements (`?g`) `about` the tag "environnement" and approved by the `SingleUser` "claire"

```
1 SELECT *
2 WHERE{
3 GRAPH ?g {?search-tag ?p ?suggested-tag}
4 FILTER (?search-tag = <http://ns.inria.fr/isicil/id/tag/environnement>)
5 ?g rdf:type srtag:SingleUserStatement
6 ?g srtag:approvedBy <http://ns.inria.fr/isicil/id/useraccount/claire>
7 }
```

7.4.2.2 Second step

We give in listing 7.4 the second query that is issued and that looks for statements approved by the `ReferentUser` (lines 3-5) and that (*i*) are not directly rejected by the current user (lines 6-9) and (*ii*) that do not conflict with the ones approved by the current user (lines 10-14). For instance the `ReferentUser` have approved three statements that conflict with statements approved by user "claire", namely

that the tags "grenelle de l'environnement", "competence environnementale", and "preoccupations environnementales" where *related* to the tag "environnement". These statements conflict with those approved by user "claire" since, in the SRTag ontology, the property `skos:related` is declared to be `srtag:incompatibleWith` the property `skos:narrower`. However, the referent user has approved a fourth statement that does not conflict with any statements from user "claire", and this is the only statement that this query return in this example, *i.e.* that the tag "environnement" is *related* to the tag "domaines environnementaux".

Listing 7.4: SPARQL query used to retrieve statements about the tag "environnement" and approved by the `ReferentUser`, and that are not directly rejected by user "claire" or contradictory with statements she has approved.

```
1  SELECT *
2  WHERE{
3  GRAPH ?g {?search-tag ?p ?suggested-tag}
4  FILTER(?search-tag = <http://ns.inria.fr/isicil/id/tag/environnement>)
5  ?g rdf:type srtag:ReferentValidatedStatement
6  OPTIONAL {
7   ?u srtag:hasRejected ?g
8   FILTER(?u  = <http://ns.inria.fr/isicil/id/useraccount/claire>)}
9  FILTER(!bound(?u))
10 OPTIONAL{
11  GRAPH ?g2 {?search-tag ?p2 ?suggested-tag}
12  ?g2 srtag:approvedBy <http://ns.inria.fr/isicil/id/useraccount/claire>
13  ?p srtag:incompatibleWith ?p2   }
14 FILTER (!bound(?g2)) }
```

7.4.2.3 Third step

The system proceeds with the next query shown in listing 7.5. This query looks for statements made on the tag "environnement" that are proposed by the conflict solver (lines 3-5). Lines 6-11 make sure that the referent user or the current user "claire" has not rejected these statements. Then we make sure that the returned statements are not incompatible with statements approved by the current user (lines 12-16), or with statements approved by the referent user (lines 17-21). For instance the conflict solver has approved the relation *spelling variant* between the tag "environnement" and the tag "domaines environnementaux", but this statements contradict the referent user that approved instead the relation *related* for the same pair of tags. Hence, this third query does not return any statement.

Listing 7.5: SPARQL query used to retrieve statements about the tag "environnement" that are approved by the `ConflictSolver` but are not directly rejected by user "claire" or the referent user, and that are not contradictory with statements approved by any of them.

```
1  SELECT *
```

```
2  WHERE{
3  GRAPH ?g {?search-tag ?p ?suggested-tag}
4  FILTER(?search-tag = <http://ns.inria.fr/isicil/id/tag/environnement>)
5  ?g rdf:type srtag:ConflictResolutionStatement
6  OPTIONAL {
7   ?u srtag:hasRejected ?g
8   ?u rdf:type ?userType
9   FILTER(?u  = <http://ns.inria.fr/isicil/id/useraccount/claire> ||
10        (?userType = srtag:ReferentUser))}
11 FILTER(!bound(?u))
12 OPTIONAL{
13  GRAPH ?g2 {?search-tag ?p2 ?suggested-tag}
14  ?g2 srtag:approvedBy <http://ns.inria.fr/isicil/id/useraccount/claire>
15  ?p srtag:incompatibleWith ?p2   }
16 FILTER (!bound(?g2)) }
17 OPTIONAL{
18  GRAPH ?g3 {?search-tag ?p3 ?suggested-tag}
19  ?g3 rdf:type srtag:ReferentValidatedStatement
20  ?p srtag:incompatibleWith ?p3   }
21 FILTER (!bound(?g3)) }
```

7.4.2.4 Fourth step

The fourth query shown in listing 7.6 looks for statements made on the tag "environnement" and approved by other "human" users (modelled with the class `srtag:SingleUser`, see lines 3-5). We then make sure that these statements have not been directly rejected by the referent user, or the current user "claire" (lines 6-11). Then the remaining lines (lines 12-26) make sure that the returned statements are not incompatible with statements approved by the current user "claire", or by the referent user, or the conflict solver. In our case, some users have approved the same statements that the current user "claire" has already approved, namely those involving the tags "grenelle de l'environnement" and "preoccupations environnementales" with the relation *narrower*, and these statements will thus be returned. The other statements approved by other human users are either incompatible with statements approved by user "claire" (such as "competence environnementale" and "preoccupation environnementales" with the relation *related*) or with statements approved by the referent user (such as "domaines environnementaux" with the relation *spelling variant*).

Listing 7.6: SPARQL query used to retrieve statements about the tag "environnement" and approved by other `SingleUser` but are not directly rejected nor contradictory with statements approved by user "claire", or the referent user, or the conflict solver.

```
1  SELECT *
2  WHERE{
3  GRAPH ?g {?search-tag ?p ?suggested-tag}
4  FILTER(?search-tag = <http://ns.inria.fr/isicil/id/tag/environnement>)
5  ?g rdf:type srtag:SingleUserStatement
```

```
6  OPTIONAL {
7   ?u srtag:hasRejected ?g
8   ?u rdf:type ?userType
9   FILTER(?u  = <http://ns.inria.fr/isicil/id/useraccount/claire> ||
10        (?userType = srtag:ReferentUser))}
11 FILTER(!bound(?u))
12 OPTIONAL{
13  GRAPH ?g2 {?search-tag ?p2 ?suggested-tag}
14  ?g2 srtag:approvedBy <http://ns.inria.fr/isicil/id/useraccount/claire>
15  ?p srtag:incompatibleWith ?p2   }
16 FILTER (!bound(?g2)) }
17 OPTIONAL{
18  GRAPH ?g3 {?search-tag ?p3 ?suggested-tag}
19  ?g3 rdf:type srtag:ReferentValidatedStatement
20  ?p srtag:incompatibleWith ?p3   }
21 FILTER (!bound(?g3)) }
22 OPTIONAL{
23  GRAPH ?g4 {?search-tag ?p4 ?suggested-tag}
24  ?g4 rdf:type srtag:ConflictResolutionStatement
25  ?p srtag:incompatibleWith ?p4   }
26 FILTER (!bound(?g4)) }
```

7.4.2.5 Fifth step

The fifth and last query shown in listing 7.7 looks for statements made on the tag "environnement" proposed by automatic agents that compute semantic relationships and are modelled with the class `srtag:TagStructureComputer` (and their corresponding statements with the class `srtag:TagStructureStatement`, see lines 3-5). Then it makes sure in lines 6-11 that these statements have not been rejected by the current user "claire" nor the referent user. Then, the remaining lines make sure that returned statements are not incompatible with statements approved by any other human user (and their corresponding statements `srtag:SingleUSerStatement`, see lines 12-16), or approved by the referent user (lines 17-21), or proposed by the conflict solver as resolutions to conflicts (lines 22-26). In our case the statements "environnement" has *spelling variant* "environmental" and "environnement" has *spelling variant* "environment" proposed by the `TagStructureComputer` fulfills all these conditions and will be returned by this query.

7.4.2.6 Summary

To summarize the results of each query, we have reported in table 7.4 all the tags and semantic relations returned by each query detailed above. We see that the query shown in listing 7.6 returns similar statements to those returned by query of listing 7.3, but this is not problematic since the system merges the results before returning them. In figure 7.6, we show the result of the merging of the suggested tags for user "claire". The relations approved by this user are depicted in blue,

Listing 7.7: SPARQL query used to retrieve statements about the tag "environnement" and approved (or proposed) by `TagStructureComputer` agents but are not directly rejected nor contradictory with statements approved by user "claire", or the referent user, or the conflict solver, or any other `SingleUser`.

```
1  SELECT *
2  WHERE{
3  GRAPH ?g {?search-tag ?p ?suggested-tag}
4  FILTER(?search-tag = <http://ns.inria.fr/isicil/id/tag/environnement>)
5  ?g rdf:type srtag:TagStructureStatement
6  OPTIONAL {
7   ?u srtag:hasRejected ?g
8   ?u rdf:type ?userType
9   FILTER(?u  = <http://ns.inria.fr/isicil/id/useraccount/claire> ||
10        (?userType = srtag:ReferentUser))}
11 FILTER(!bound(?u))
12 OPTIONAL{
13  GRAPH ?g2 {?search-tag ?p2 ?suggested-tag}
14  ?g2 rdf:type srtag:SingleUserStatement
15  ?p srtag:incompatibleWith ?p2   }
16 FILTER (!bound(?g2)) }
17 OPTIONAL{
18  GRAPH ?g3 {?search-tag ?p3 ?suggested-tag}
19  ?g3 rdf:type srtag:ReferentValidatedStatement
20  ?p srtag:incompatibleWith ?p3   }
21 FILTER (!bound(?g3)) }
22 OPTIONAL{
23  GRAPH ?g4 {?search-tag ?p4 ?suggested-tag}
24  ?g4 rdf:type srtag:ConflictResolutionStatement
25  ?p srtag:incompatibleWith ?p4   }
26 FILTER (!bound(?g4)) }
```

query	tag linked to "environnement"	relation
List. 7.3	grenelle de l'environnement	narrower
	compentence environnementale	narrower
	preoccupations environnementales	narrower
List. 7.4	domaines environnementaux	related
List. 7.5	(no results)	-
List. 7.6	grenelle de l'environnement	narrower
	preoccupations environnementales	narrower
List. 7.7	environmental	spel. var.
	environment	spel. var.

Table 7.4: Summary of the tags and semantic relations returned by each of the 5 queries issued to apply the priority order and to present the user "claire" with coherent results when searching tags related to the tag "environnement".

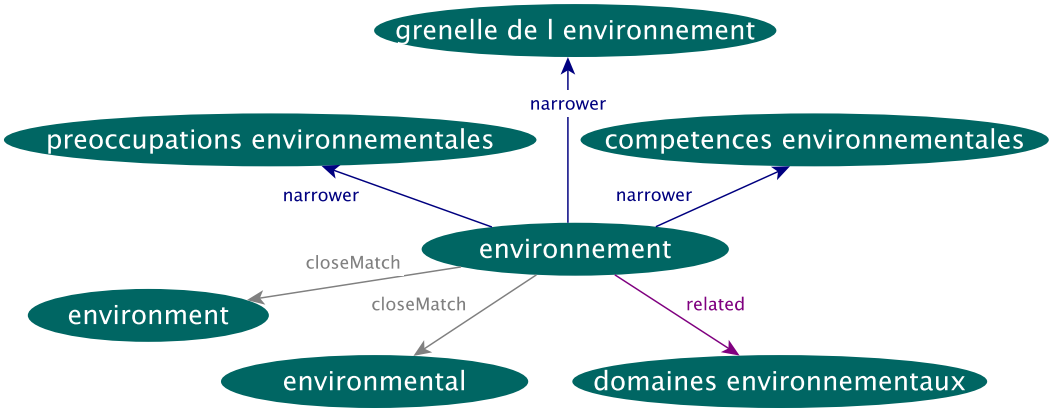


Figure 7.6: Example of the integration to a user's point of view of relations proposed by the other human agents or by other types of agents. In this example, we show the relation approved by the user"claire" in blue, the relations approved by the referent user in purple, and the relation proposed by the automatic agent (`TagStructureComputer`) in grey. The relations approved by other users overlap with the one already approved by user "claire".

those approved by the referent user in purple, and those proposed by the automatic agent in grey. This figure shows the tags that would be suggested by the system to the user "claire" when searching for the tag "environnement".

As a consequence, the combination of these queries allows us to enrich each user's point of view with 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.

7.5 Conclusion

In this chapter we have presented our method to build a consensual point of view out of the diverging individual contributions from users and other automatic agents. We have also proposed a method to allow each user to benefit from the others' contributions, while still preserving the coherence of their point of view.

In the preceding chapters we saw how automatic agents bootstrap the process of folksonomy enrichment and how we enable each user to validate or to correct these automatically generated statements about the semantics of tags. However, as each user can maintain his own point of view independently of the others' points of view, some logical inconsistencies may arise from a global point of view. The conflict solver detects these inconsistencies. This conflict solver first issues a SPARQL query to detect the pairs of tags that are linked with more than one semantic relation. Then these conflicting pairs are further processed and the conflict solver checks whether one of these conflicting relations gained a clear consensus, and if it is not the case, it proposes the *related* relation which can be considered as

a compromise.

We also presented in this chapter the experiment we conducted to evaluate the conflict solver mechanism among 5 users who were asked to pick up one semantic relation for a set of 94 pairs of tags. This experiment showed the usefulness of our multi-points of view approach to folksonomy enrichment as we observed that users proposed for almost half of the pairs of tags several conflicting relation. Moreover, for 15% of the pairs of tags, some users rejected the single relation that was approved by other users. This shows that a consensus is not necessarily met regarding the semantics of tags, even among a relatively small set of users. This experiment also showed that the *hyponym* and *related* relations were more often conflicting with other relations than the *spelling variant* (close match) relation. Furthermore, pairs of tags involving a compound word of one of the other tag of the pair seems also more likely to be at the origin of debatable or conflicting semantic relations.

The results of the conflict solver are then exploited to build a global visualization of the structured folksonomy that allows us to spot the pairs of tags with conflicting relations, but also the pairs of tags with a debatable or only rejected relation. This is meant to help the referent user maintain a global and consistent point of view that will be used to enrich each user's point of view with others' contributions.

Finally, we presented in detail the set of rules that we apply to allow each user to benefit from the other individual contributions and from the automatically generated relations. The idea of this module of our approach is to integrate progressively the statements by following a priority order, starting from the statements approved by the considered user, and then adding statements from the referent user, then from the conflict solver, then from other human users, and finally from automatic agents. In this process, we make sure that each integrated statement does not contradict the statements previously integrated. As a result, each user's point of view is enriched with statements approved by other agents while preserving its logical consistency.

At this stage of the folksonomy enrichment we obtain a structured folksonomy in which diverging points of view of the users coexist with a transversal and logically consistent point of view maintained by the referent user. The cycle of the folksonomy enrichment can then restart as soon as new tags are added or new relations are proposed or modified by the users.

CHAPTER 8

Implementation of a semantic tagging-based system fostering multi-points of view enrichment of the folksonomy

Abstract. This chapter covers the implementation of a tagging-based system grounded on our approach to folksonomy enrichment. This system is the tagging-related part of ISICIL. This project aims at providing tools combining Semantic Web and Web 2.0 paradigms to leverage technological watch within organizations. We detail the specifications of the Tag and Computing Servers that take care of basic tagging functions. We also describe the semantic structuring of the folksonomy that is based on the models and our approach to folksonomy enrichment we presented in the previous chapters. We also give detailed results obtained on Ademe's dataset by the computational methods implemented in the Computing Server to automatically infer tags' semantics. Next, we give a full presentation of the conception of the interface for capturing user's points of view via a tool that integrates micro-editing of the folksonomy within tag-based search. Lastly we introduce the interface for helping a referent user build a global point of view.

Contents

8.1 Introduction

During this thesis, we took an active part in the ISICIL project that is devoted to applying the principles of the Web 2.0 with the help of Semantic Web technologies to leverage technological watch within organizations. The assumption behind this project is that combining the key aspects of the success of the Social Web with the Semantic Web can greatly benefit to knowledge exchange within organizations. In this regard, we were involved in the development of the tagging-related part of the ISICIL solution. The implementation of the tagging based system is grounded on the models we have presented in this thesis, and this system is aimed at fostering the semantic enrichment approach we have defended along the previous chapters. The goal of this chapter is to give the reader a precise idea of how our approach has been implemented with concrete programs and interfaces.

The principle of the ISICIL solution is to propose an approach to the design of knowledge management tools used by organizations. The ISICIL solution is made of a set of services that are exploited by front end clients in order to better index and organize both Web content and intraweb content. The solution in ISICIL consists in a Tag server that provides core-tagging as well as semantic structuring functions, and a Computing Server processing tagging data. Front end clients include tools to tag resources, navigate and structure the folksonomy.

The software parts of the Tagging-based system of the ISICIL solution exploits a series of models used to describe the different elements linked to tags. Our contribution consists in adapting existing models to the needs of the project, but also to propose new models especially to describe tags in a flexible way (NiceTag), and to reify semantic relations between tags (SRTag) in order to be able to capture users' opinions on these relations.

Next, we give the detailed specifications of the Tag and Computing Servers. These two servers can be seen as the basis of the implementation of our approach since they provide means to tag resources using NiceTag, and to express tags' semantics thanks to SRTag. The Computing Server allows bootstrapping the semantic enrichment process thanks to a series of computational methods. We detail in this chapter the results given by these methods on a concrete dataset collected

at the Ademe agency. The Computing Server is also in charge of solving conflicts possibly arising between users' points of view. These points of view are stored and queried thanks to the Tag Server, but, in the first place, they are captured using a dedicated interface, SRTAgEditor.

Our solution to capture users' contributions as seamlessly as possible with SRTagEditor comes as a continuation of a research conducted in the Edelweiss team to design user-friendly tools for the collaborative editing of ontologies and structured folksonomies. The interface we propose can be seen as a micro-editor of folksonomies that allows users to modify the structuring of tags that is proposed by automatic agents in the first place. Furthermore, we propose to integrate this structuring tasks within everyday tasks such as folksonomy navigation. Our goal is to make the individual contributions a secondary effect of the normal use of our system, and this is achieved by grounding the structuring functions on simple and optional drag and drop manipulations.

Lastly, we present the interface envisioned to help the Referent User build a global and coherent structuring of the folksonomy out of all the individual contributions. The results of the Conflict Solver are exploited in this end to produce a global visualization of the structured folksonomy that includes all the possibly conflicting points of view.

This chapter is organized as follows. In section 8.2 we give an overview of the infrastructure of the ISICIL solution and the main elements that compose it. Then, section 8.3 gives a summary of the models utilized in our approach, and section 8.4 and section 8.5 detail the specifications of, respectively, the Tag Server and the Computing Server. Next, we provide a detailed presentation of the results yielded by the Computing Server for the Ademe's dataset in section 8.6. We address after that the design of SRTAgEditor, the front end client for capturing individual contributions in section 8.7, and the interface for helping the Referent user build a global point of view in section 8.8, before we conclude in section 8.10.

8.2 Infrastructure of the ISICIL solution

In this thesis, we took an active part in the development of the components of the ISICIL solution. The goal of the ISICIL project is to propose novel knowledge management tools within the context of companies or organizations that integrate the principle of collaboration of the Social Web, enhanced with semantic technologies. In particular, these tools are aimed at helping users to share and take the best of the pieces of knowledge they collect in their activity of technological or scientific monitoring. In this project, we were involved in the design of the Tag Server and the Computing Server that take care of the tagging of the shared resources and the semantic structuring of tags.

The Tag and Computing Server are integrated within the ISICIL solution, which operates at three distinct levels, as shown in figure 8.1:

1. Legacy : this level corresponds to the content already existing in an organi-

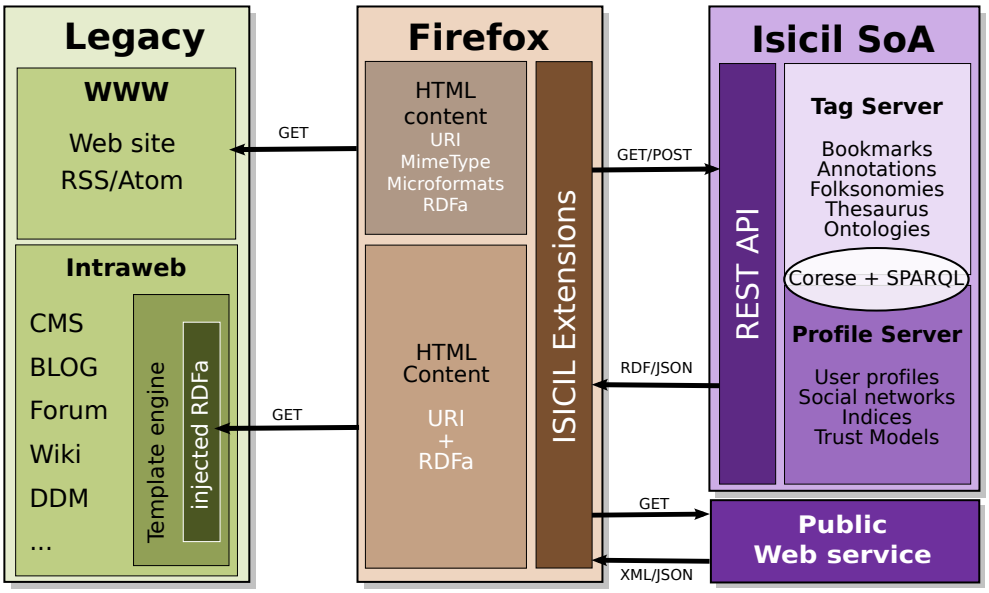


Figure 8.1: Global architecture of the ISICIL solution presenting the three main layers of the solution adopted. In this thesis, we contributed to the *SoA* layer by developing the Tag Server and to the *Firefox* layer by developing an extension to search and structure the folksonomy (SRTagEditor)

zation. Indeed, most of the organizations maintain an intraweb composed of a variety of tools (blogs, forum, wikis), and it is often difficult to ask these organizations to change their intraweb's infrastructure or to migrate towards alternative platforms. Therefore, ISICIL made the choice to seamlessly integrate existing systems by injecting semantic metadata within the already existing content management systems (CMS). One strategy to achieve this goal is to modify the template engine of these systems in order to inject RDFa data within the web pages they generate. Tagging existing content is also an alternative, the main idea being to be able to describe already existing content with semantic metadata.

2. Firefox: this level corresponds to the front end used to access and share knowledge. The idea here is to propose a set of Firefox extensions that interoperate with the ISICIL's services to provide knowledge organization or navigation functions. The objective is also to collect the most relevant metadata according to the context of navigation. For example, when a user is browsing a web page, the system is able to indicate that some other members of the community have already tagged this page with a set of tags suggested to the user. The choice of Firefox is justified by the openness of its framework for developing extensions. An example of such interfaces is the SRTagEditor dedicated to the capture of user's contributions regarding the semantic structuring of tags, and that is presented in details in section 8.7 on page 224.

3. ISICIL SoA (Service-oriented Architecture): this level corresponds to the core ISICIL infrastructure that publishes RESTful web services. These web services are composed of a Tag Server that manages the indexing of resources with tags, but also the lifecycle of the folksonomy and the termino-ontological resources (Ontologies, thesauri). The second part is the Profile Server that deals with users' profile, social networks, and the access rights. Both of this part of the ISICIL server have a dedicated part in the Computing Server that is in charge of the heavy computations, for instance to suggest automatically semantic relations between tags. The semantic engine used by all these components to query and make inferences on the RDF data is Corese[1], an engine developed by the Edelweiss team.

In figure 8.2 we detail the ISICIL server and client part. The ISICIL framework consist in the skeleton for all the application of the ISICIL solution. The three main servers thus depend on it as it provides the core functionalities and implementations. This framework also ensure an efficient interoperability between the three servers that have the following main features:

- Tag Server provides a set of services to manage the annotations of the resources based on tags, the action of semantic structuring performed by the user or by automatic agents, and the different termino-ontological resources

[1] COnceptual REsource Search Engine, http://www-sop.inria.fr/edelweiss/software/corese/

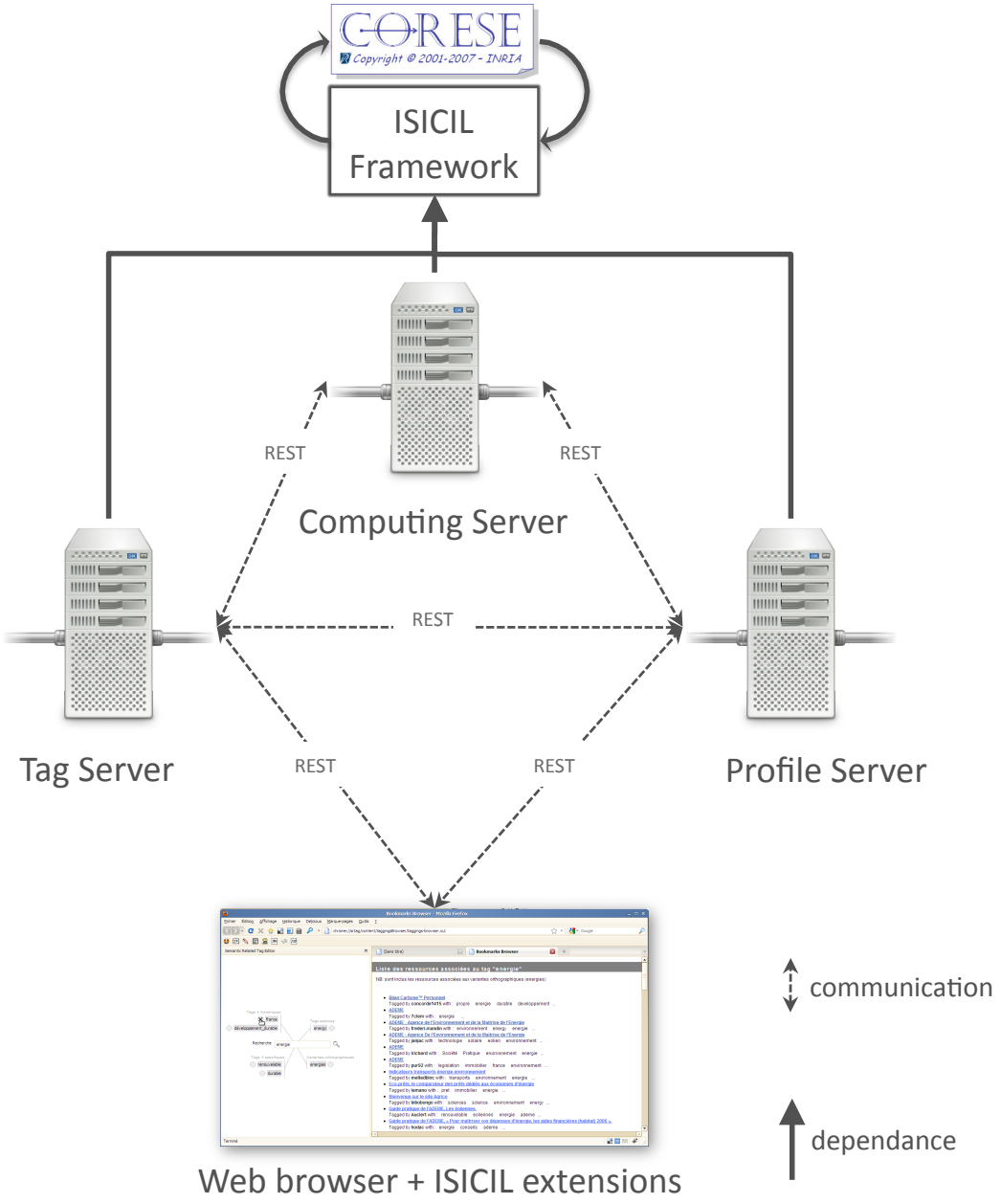


Figure 8.2: Detail of the organization of the ISICIL servers and clients

used in our target communities. This server allows for instance dedicated clients to create and navigate tagging data, but it also allows suggesting semantically linked tags and manages the edition of semantic relations.

- The Profile Server provides a set of services to help users maintain their personal profiles with their subject of interests, their proficiencies, the relations they share with other members of the community or outside the community. It can also be used to find experts on a given topic, or find persons who were involved in a given project, etc.

- The computing server is aimed at performing heavy computations needed by both the Profile and the Tag Servers, as for instance the computation of indicators on the structure of the social network, or semantic relations between tags, or conflicts existing between the different points of view of the users.

The Tag Server and the Profile Server are the only two servers to communicate directly with the clients that consist in ISICIL Firefox extensions such as SRTagEditor for example. They can also communicate with each other when for instance one looks for the tags of a given user, and they both communicate with the Computing Server that, for instance, calls them to load the current semantic annotations that they each manage. The communication between all these components and the clients follow the protocol REST that has several advantages, among which : XML-based exchange of data and information, flexible integration of heterogeneous platforms via HTTP protocol, many programming languages available, no proprietary clients, simple publishing procedure, etc. The implementation of the REST web services has been done with JAX-RS Jersey[2], and the creation and transformation of XML data from and to Java objects is based on Sun Java API JAXB[3] (Java Architecture for XML Binding). The persistence of the data that cannot be managed by Corese is done using a relational database based on an H2 engine [4] that is manipulated through the Hibernate[5] framework.

In the remaining of this chapter, we will focus on the 3 parts on which we actively participated:

1. the specification and the development of the Tag Server

2. the specification of the computation modules related to tags in the Computing Server

3. the conception and development of SRTagEditor, a Firefox extension aimed at capturing user's contributions regarding the semantics of tags.

[2]Jersey, open source JAX-RS (JSR 311) Reference Implementation for building RESTful Web services, https://jersey.dev.java.net/

[3]https://jaxb.dev.java.net/

[4]H2, SQL relational database engine written in Java, http://www.h2database.com/

[5]Hibernate, open source Java persistence framework project, https://www.hibernate.org/

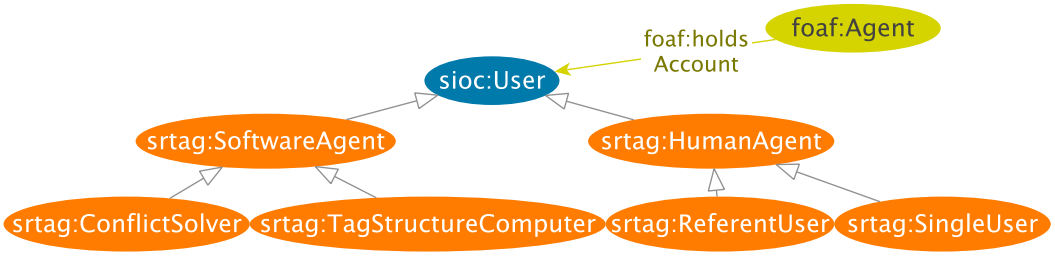


Figure 8.3: Model for the users including SRTag, SIOC and FOAF

8.3 Different elements to model

Before going into details in the implementation of the tagging-based system of ISICIL, let us recall the models we chose to represent the main elements that we manipulate in this regard.

Resources

Resources are the documents indexed at Ademe for instance, or the web pages that are bookmarked by users. By default resources are modeled with a custom class from NiceTag `nicetag:TaggedResource`. This class is the equivalent of the class `irw:Resource` from the Identity of Resources on the Web (IRW) ontology proposed by Halpin & Presutti (2009) as a way to overcome the fuzziness around the link between a resource and its URI, a problem coined as the "Identity crisis" of the Web.

Users

Users (see figure 8.3) are modeled with the class `sioc:User`, but each user account is linked with an instance of `foaf:Agent` that corresponds to the person (in our case) that holds the user account represented with `sioc:User`. Hence, one `foaf:Agent` can have several `sioc:User` instances. Furthermore, as we have seen in chapter 6, we have extended the class `sioc:User` in order to represent the different types of agents that take part in the folksonomy enrichment.

Tags

Tags (see figure 8.4) will be modeled with different ontologies according to their provenance and the way they are chosen:

1. For the tags freely provided by users, we make use of the class `scot:Tag` that is itself a subclass of the tag class from Newman *et al.* (2005) ontology `tag:Tag`.

2. For the controlled tags provided by Ademe's archivists, we proposed a custom class `svic:MC` (for Mot-Clé, the french for key-word) that is a subclass

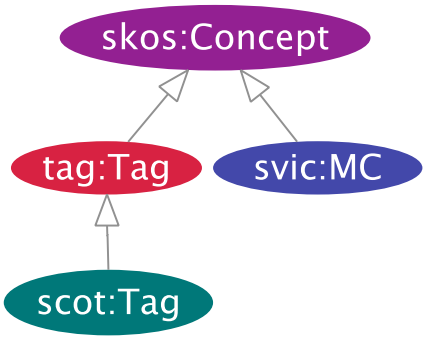


Figure 8.4: Hierarchy between the different models to represent tags including Newman's TagOntology (tag), SCOT, SKOS, and a custom schema to represented tags from a controlled vocabulary `svic:MC`.

of `skos:Concept`. We used a custom class instead of `skos:Concept` because we wanted to account for the fact that these tags come from a controlled vocabulary whose structure remains nevertheless flat for the moment, unlike the structure of a thesaurus.

3. For the tags that would actually come from a genuine thesaurus, we make use of the class `skos:Concept`. This would be the case for instance when tagging with concepts from the GEMET thesaurus at Ademe for instance.

Since all the tag and key word class inherits from the class `skos:Concept`, all the queries on tags can be made with this class.

Representing tagging with TagOntology

In the first version of our implementation we used the model of tagging proposed first by Newman *et al.* (2005) and later integrated in the SCOT model. This modeling of tagging (see figure 8.5) includes thus SCOT on the side of the tags, and Newman's TagOntology to represent a tagging that links a series of tags, a tagged resource (modeled as an `rdfs:Resource`) and a tagger (modeled with `sioc:User`).

We give an example of a tagging instance with the TagOntology in listing 8.1. Line 1 corresponds to the instanciation of the class `tag:Tagging`, and line 2 indicates the tagged resource, line 3 indicates the associated tag, lines 4-5 gives the user who performed the tagging, and line 6 gives the date.

Representing tagging with NiceTag

The detail of the presentation of the NiceTag model aimed at describing tagging instances is given in chapter 4, and we briefly recall here the core of the model. A tagging instance is modeled in NiceTag as a link between a tagged resource (`nicetag:TaggedResource`) and a tag. The type of the tag is not constraint in NiceTag, so that it is possible to use any of the models we chose to represent tags,

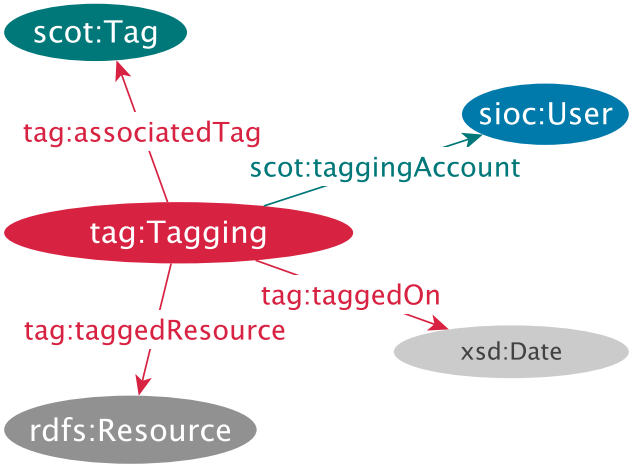


Figure 8.5: Tagging model based on SIOC, SCOT and Newman's TagOntology

Listing 8.1: Example of an RDF annotation of a tagging instance with Newman's TagOntology

```
<tag:Tagging rdf:about="http://mysocialsi.te/tagging#7182904""
  <tag:taggedResource rdf:resource="http://www.yesand.com/"/>
  <tag:associatedTag>improvisation</tag:associatedTag>
  <scot:taggingAccount
    rdf:resource="http://mysocialsi.te/user/fabien.gandon"/>
  <tag:taggedOn>2009-10-07T19:20:30.45+01:00</tag:taggedOn>
</tag:Tagging>
```

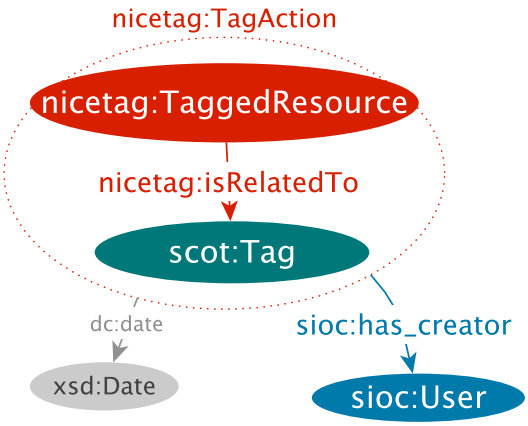


Figure 8.6: NiceTag model for tagging which allows using SKOS for the tag, and SIOC for the tagger

namely, `scot:Tag`, `skos:Concept`, and `svic:MC`. The link between a tagged resource and a tag is then encapsulated within a named graph, and the tagger can be linked with the tag action thanks to the property `sioc:has_creator` (see figure 8.6) . The main difference with Newman's TagOntology is that, in NiceTag, each tag action is strictly defined as a link between 1 resource, 1 user, and 1 tag, whereas in the TagOntology a `tag:Tagging` can link several tags to 1 user and 1 resource and hence represents a post. However, a post can also be easily recomposed in NiceTag when one queries all the tag actions associated to a given resource by a given user. An example of a RDF annotation of a tagging instance using NiceTag, as well as the way to query this type of data is given in section 4.2.6.

Representing semantic relations between tags and user's points of view with SRTag

As the SRTag model has been presented in detail in chapter 6, we briefly recall here the main parts of the model used to describe and reify semantic relations between tags, and to capture user's opinions on these relations. An illustration of this model can be found in figure 6.3 on page 159 where we present the core of SRTag. A semantic relation between two tags is encapsulated within a named graph that is further typed according to the type of user (see the hierarchy of user's type in figure 8.3). Then, the point of view of users is captured by representing the agreement or disagreement with the reified relation with the properties `srtag:hasApproved`, `srtag:hasRejected`, or `srtag:hasProposed`. An example of a RDF annotation of a reified relation approved and rejected is given in section 6.4.5 on page 161.

8.3.1 Combining namespaces

To summarize all the models that we used in our implementation, we give the list of their namespaces in table 8.1. The DublinCore (`dc`) schema is used for generic

purposes as, for instance, giving the title, the date, an abstract, or the author of a resource when no custom property are given by the model we use. XML Schema Definition (`xsd`) is used in our system to specify the type of a value, for instance an integer or a double. COrese Schema (`cos`) contains the property `cos:graph` that allows declaring the source of a graph, and this is used in NiceTag and in SRTag to assign an URI to a triple encapsulated within a named graph. Friend Of A Friend (`foaf`) allows describing the information pertaining to a person, an organization, etc. such as its name, address, topic of interest, the documents it/she/he has produced, etc. Simple Knowledge Organisation Schema (`skos`) gives a framework to describe thesauri in RDF by providing classes for a concept, a collection of concepts, etc. and also properties to describe semantic relations between concepts. Semantically Interlinked Online Communities (`sioc`) is used in our system to describe the notion of a user account (`sioc:User`)[6]. Semantic Cloud Of Tag (`scot`) is used in our implementation to describe freely contributed tags, in contrast with tags provided by professional archivists that are described with the SVIC schema (whose name come from the name of the archivists' service at Ademe). Newman's TagOntology (`tag`) was used in the first version of the Tag Server to describe tagging instances, but it is now replaced by the NiceTag model that is described in chapter 4. Finally, the Semantically Related Tag model (`srtag`) is used to described reified semantic relations about tags and the opinion of users regarding these relations (see chapter 6 for a detailed presentation). Lastly, we indicate the `base` we used as the root of the URI of the resources we created in our system; then a custom suffix is added to this root depending the type of resources:

- `tag` for freely contributed tags,
- `mc` for tags provided by archivists,
- `tagging` for tagging instances,
- `srtag-rel` for the reified semantic relations between tags,
- `user` for the URI of the user accounts.

The last part of the URIs we generated is given by a unique identifier constructed automatically by encoding the information related to the current resource, such as the label of the tag for example.

8.4 Specification of a semantic tagging server

In this section we are going to detail the specification of the functionalities that have been implemented on the ISICIL Tag Server in order to allow users to tag resources and to search for tags and tagged resources, and also to enable users to

[6] which has been renamed sioc:UserAccount recently, http://sioc-project.org/ontology#term_UserAccount

short	namespace
dc	http://purl.org/dc/elements/1.1/
xsd	http://www.w3.org/2001/XMLSchema#
cos	http://www.inria.fr/acacia/corese#
foaf	http://xmlns.com/foaf/0.1/
skos	http://www.w3.org/2004/02/skos/core#
sioc	http://rdfs.org/sioc/ns#
scot	http://scot-project.org/scot/ns#
tag	http://www.holygoat.co.uk/owl/redwood/0.1/tags/
nicetag	http://ns.inria.fr/nicetag/2009/09/25/voc#
srtag	http://ns.inria.fr/srtag/2009/01/09/srtag.rdfs#
svic	http://www.ademe.fr/2009/svic-schema.rdfs#
base	http://ns.inria.fr/isicil/id

Table 8.1: Namespaces of the models used in our system, plus the namespace corresponding to the base of the URI of the instances we create

reject or propose semantic relations between tags. These elements have been implemented as RESTful webservices that can be called by the other servers (Profile Server and Computing Server) and by the clients.

8.4.1 Core tagging functions

8.4.1.1 Creation of a new tag

This web service (see table 8.2) is called whenever a tag is submitted by a user while tagging or managing the folksonomy, or by an automatic agent when, for example, importing an external folksonomy. This web service takes as input parameter the string of the tag *S* and the type of the tag. If the tag is submitted by a regular user, we use the type `scot:Tag`, and when the tag is submitted by a referent user, such as an archivist from Ademe, the model for the tag node is `svic:MC` to account for the controlled nature of the tag. We assume that `skos:Concept` will be created in a tailored thesaurus editor, but as we already pointed, concepts from thesauri are natively integrated in our system, since the `skos:Concept` class is the "mother" class of all the tag's classes we used.

8.4.1.2 Suggestion of tags for autocompletion

To help users choose a tag when they search tags or when they post a tagging, this web service (see table 8.3) will send the list of existing tags whose labels start with the letter the user has issued in the dedicated interface (search bar or text input for tags while tagging).

Input	• Character string *S* • Type of node < `freeTag` || `controlledTag` >
Behavior	**IF:** tag with label *S* does not already exist **THEN (A):** create new tag with label *S* and typed as: 1. `scot:Tag` if type = `freeTag` 2. `svic:MC` if type = `controlledTag` **ELSE (B):** do nothing
Output	**(A)** Annotation of new tag **(B)** message "tag already exists"

Table 8.2: Web service to create a new tag node

Input	• Character string *S*
Behavior	Look for the tags already existing in the data base whose labels start with *S*
Output	List of tags starting with *S*, and for each tag: • tag URI • tag label • tag type

Table 8.3: Web service to suggest tags whose labels start with string *S* for autocompletion purposes

8.4.1.3 Tagging a resource

This web service is called when a user is tagging a resource (see table 8.4). This situation occurs when a user posts a bookmark for instance and submit a list of tags associated to the tagged resource. It can also correspond to when a user tags a wiki page. As several tag with the same label, but with different meanings can exist, the tagging interface should make use first of the tag suggestion and the tag creation service to make sure that the user choose the right tag's URI. Indeed, we

propose the following strategy regarding the choice of the different models of tags, and the choice of already existing tags:

- Free tags modeled with `scot:Tag` should be created when a user merely provides a character string as a tag and does not choose a controlled tag or a concept from a thesaurus for example.

- Then, if a controlled tag modeled with `svic:MC` has the same label that the user wanted to use, then this tag should be chosen by default by the tagging client.

- In the case when we also have several choices of tags or concepts with the same label as the user wanted to use, the tagging interface should allow the user to choose the concept that suits his intended meaning (with a popup describing briefly its definition *e.g.*)

As a result, after the user has entered the list of desired tags, the tagging client should provide a list of tag's URIs each corresponding to the following cases:

- URI of a newly created tag when no other tag already existed in the database

- URI of an already existing free tag, or controlled tag, or thesaurus' concept.

In cases when the user is updating a tagging of resource he has already tagged before, then the system updates the corresponding annotation entry in the database.

Input	• user's URI *user* • list of tags' URI $L_{tag} = \{tag_1, tag_2, ..., tag_n\}$ • tagged resource's URI *tr*
Behavior	**FOR EACH** tag of L_{tag} : **IF** *user* has not already tagged *tr* **THEN** create a tagging annotation **ELSE** update already existing tagging annotation
Output	1 tagging annotation following NiceTag model (see an example in listing 5.1) for each tag submitted

Table 8.4: Web service to create tagging instances

8.4.1.4 Tag-based search of tagged resources

This web service is aimed at providing for a list of tagged resources associated to the tag submitted. In order to also include the spelling variants of the searched-for tag or narrower concepts, we proposed adding the possibility to submit a list of tags to be searched for. The web service that gives the list of tags sharing a semantic relation with the searched-for tags is presented below.

Input	• list of tags' URI $L_{tag} = \{tag_1, tag_2, ..., tag_n\}$ to be searched for
Behavior	Look for the tagging and the associated resources associated to ONE of the tags from L_{tag}
Output	List of tagged resources, with the following details for each: • URI • user who tagged it • title of the resource

Table 8.5: Web service to search for tagging of resources

8.4.2 Semantic relations: search and rejection/proposal

Now we present the web service that are linked with the semantic enrichment of the folksonomy. These web services are used to get the list of semantically linked tags to a searched-for tag, but also to reject or propose a relation. These web services are for instance exploited by the SRTagEditor that we present in section 8.7 on page 224.

8.4.2.1 Searching for semantically linked tags

This web service (see table 8.6) provides a list of tags semantically linked to the searched for tag. As we have seen in chapter 7, we utilize in this case the strategy we propose to enrich a user's point of view with the contributions of the other users and the automatic agents, giving the priority to the Referent User's point of view when several relations conflict for a given pair of tags. As the detail of this strategy is explained in this chapter, we briefly recall here the basic principle. When searching for semantically linked tags to the searched-for tag *for* a given user u, we progressively look for statements about semantic relations that have been approved by different types of user, following the priority order given below:

1. First, we integrate the set of statements S_u approved by the user u.

2. Then, we integrate the set of statements S_{ru} approved by the `ReferentUser`, except if they conflict with one statement from S_u.
3. Then, we integrate the set of statements S_{cs} approved by the `Conflict-Solver`, except if they conflict with one statement from S_u or S_{ru}.
4. Then, we integrate the set of statements S_{ou} approved by other users (`SingleUser`), except if they conflict with one statement from S_u, S_{ru}, or S_{cs}
5. Finally, we include the set of statements S_{tc} approved by automatic agents (`TagStructureComputer`), except if they conflict with one statement from S_u, S_{ru}, S_{cs}, or S_{ou}.

As a result, we obtain a list of semantically linked tags to the searched-for tag that is compatible with the point of view of the current user.

Input	• URI of the searched-for tag t • URI of the currently logged in user u
Behavior	Look for the tags sharing a semantic relation with searched-for tag t following a priority order regarding the type of statements
Output	List of semantically linked tags, and for each tag: • URI • label • type of the relation shared with t • URI of the statement reifying the relation

Table 8.6: Web service to search for semantically related tags

8.4.2.2 Rejecting a semantic relation

Once relations are suggested to the user, they have the opportunity to reject the relations with which they disagree. This web service (see table 8.7) is aimed at generating the corresponding annotations that follows the SRTag model. An example of an annotation of rejection is given in listing 6.4. Furthermore, in order to maintain a local coherence of the currently logged in user's point of view, this web service makes sure that the user has not approved before the relation he is rejecting. If this is the case, then the relation of approval is deleted.

8.4.2.3 Proposing a semantic relation

This web service (see table 8.8) allows user to propose a relation that :

Input	• URI of relation *rel* to be rejected • URI of the currently logged user *u*
Behavior	**DO:** create annotation $< u >$ `srtag:hasRejected` $< rel >$ **IF:** $\exists$ annotation $A_{app} :=< u >$ `srtag:hasApproved` $< rel >$ **THEN:** delete annotation A_{app}
Output	Error or success message

Table 8.7: Web service to reject a relation

1. has not been proposed yet

2. or that was hidden to the user due to the strategy followed when searching for related tags (see section 8.4.2.1).

In both cases, user are able to propose another relation than the one that has been suggested to them. The parameters to submit a relation are the URI of the tag-subject, the URI of the tag-object, the relation type and the URI of the user who proposes this relation. In cases the relation already exists (which corresponds to the second option mentioned above), the system generates an annotation stating that the user *approves* this relation. If this is not the case, then it creates the new relation plus an annotation stating the user *has proposed* it. Similarly to the web service to reject relation, the system checks if the user had not previously rejected a relation, and in this case it simply erase the rejection annotation in the database.

8.4.3 Temporary conclusion

We have detailed in this section the specification of the Tag Server that allows creating tags, suggesting tags for autocompletion purposes, tagging resources, and search for tagged resources. The Tag Server also implements the functionalities related to the semantic enrichment of the folksonomy and the actions that user can perform in this respect. This includes the search for semantically linked tags following the strategy to enrich, in a coherent way, each user's point of view with other contributions from human and automatic agents. This Tag Server also includes the possibility for users to reject a relation or propose another relation than the one that have been suggested to them. Now we are going to focus on the implementation of the Computing Server that is in charge of calculating semantic relations between tags.

Input	• URI of the tag-subject tag_{sub} of the relation • URI of the tag-object tag_{ob} of the relation • URI of relation *rel* proposed • URI of the currently logged user u
Behavior	**IF:** $\exists$ statement $s :=< tag_{sub} >< rel >< tag_{ob} >$ **THEN:** create annotation$< u >$ `srtag:hasApproved` $< s >$ **IF** $\exists$ annotation $A_{reject}:=< u >$ `srtag:hasRejected` $< s >$ **THEN** delete A_{reject} **ELSE:** 1. create statement $s :=< tag_{sub} >< rel >< tag_{ob} >$ 2. create annotation $< u >$ `srtag:hasProposed` $< s >$
Output	Error or success message

Table 8.8: Web service to propose a relation

8.5 Design of the computing server

The purpose of the Computing server is to bootstrap the process of semantic enrichment of the folksonomy by submitting to the Tag Server a set of statements of semantic relations between tags. This computation is performed by different types of automatic agents corresponding each to a method presented in chapter 5:

- **String-based heuristic method** that combines different types of similarity metrics to detect *related*, *broader/narrower,* and *spelling variant* relations.
- **Tag-Tag context similarity method** that computes a similarity between tags as the cosine distance between the vector representations of the tags according to the distributional aggregation in the Tag-Tag context. When this similarity is above a threshold, the system proposes a *related* relation between the corresponding tags
- **User-based association rules mining** that looks for inclusions of sets of users associated to a tag to infer *broader/narrower* relations.

Then these statements can be rejected or approved by some users, and new ones can also be proposed by users thanks to the services of the Tag Server presented above.

At this point, the **Conflict Solver**, which is part of the Computing Server, detects and proposes solutions to the conflicts that may have emerged between the users. A conflict emerges when several users proposed different relations for a given pair of tags. The method used to solve conflicts is detailed in chapter 7, but we recall briefly here that the Conflict Solver: (1) looks for pairs of tags for which several relations have been approved by different users, (2) checks whether one of these conflicting relations reaches a clear consensus, and approves this relation if it finds it, or (3) proposes the *related* relation as a compromise when no consensus has been reached.

The Computing Server will be used during low activity periods of time due to the time it takes for all these computations to complete. These computations are performed by automatic agents, and each statement they make is linked to the corresponding type of agent, each modeled as a subclass of `srtag:Automatic-Agent`. This allows tracking back the source of each statement in further processing. Moreover, the Tag-Tag Similarity and User-based association methods are not incremental since when new tags are added, the structure of the whole folksonomy is modified. This is not the case for the string based method that analyzes only the labels of tags, and for this method we will only compare the labels of newly added tags with all the other tag labels.

In the case of a computation with any of these four modules, the outcome of the Computing Server can be the following:

1. a set of reified relations
2. a set of approval or proposal of reified relations

The annotations of the reified relations (1) are merely added to the existing ones, and they are never erased, even if one relation is only rejected by users. Indeed, if a newcomer in the community approves it, we want to be able to reuse the same statements, but also we want to keep track of the memory of the process of the folksonomy enrichment. Moreover, some methods to compute tags' semantics rely on the structure of the folksonomy. And when new tags and tagging instances are added, the structure of the folksonomy can change to the point that a relation previously inferred is no longer inferred. **This is why the set of relations computed at day D_i will be merged with the set of relations computed at D_{i-1}**. This allows avoiding to erase a relation that is no longer inferred but nevertheless already approved by some users. Of course, when the mass of annotations of reified relations reaches an upper bound, our approach allows erasing all or parts of the reified relations that are *not approved* by anyone (a single query is needed in such a case).

Regarding the annotations of approval or proposal of relations by the automatic agents of the Computing Server (2), the situation depends on the type of computation. For the methods computing tags' semantics, the corresponding set of annotations can also be kept as long as storage capacities are not overtaken. However, in the case of the Conflict Solver, the set of approvals will be regenerated at each pass. Indeed, as explained in section 7.2, if one of the conflicting

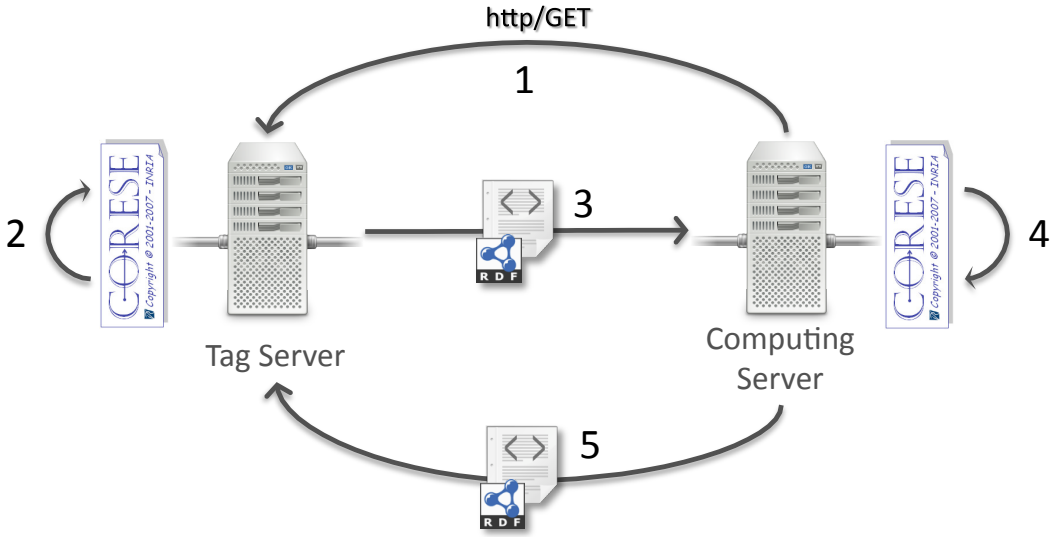


Figure 8.7: Principle of the computing server

relation on a given pair of tags is approved by the Referent User after a pass of the Conflict Solver, then the Conflict Solver will ignore this pair of tags in its next pass. Thus, the set of approvals or proposals of the Conflict Solver computed at day D_i will replace the corresponding set computed at day D_{i-1}.

In figure 8.7 we show the principles of the Computing Server regarding the communication with the Tag Server. Indeed, the tagging data to be processed by the Computing Server in our case is stored and managed by the Tag Server, thus both servers exchange data. For all types of computation the Computing Server first (1) sends a HTTP/GET request to the Tag Server in order to retrieve the data it needs. The Tag Server processes this query (2) in order to retrieve the RDF data that depends on the type of computation:

- **String-based heuristic method** : retrieves the tags and their labels,
- **Tag-Tag context similarity method** and **User-based association rules mining:** retrieves the tagging instances since they contain all the information of the structure of the folksonomy.
- **Conflict Solver :** retrieves the statements about semantic relations between tags and the points of views of all users of the system.

Then (3) the Tag server sends RDF data to the computing server that performs the desired computation. As we saw above, when computing tags' semantics, the Computing Server merges the data generated with the data it received from the Tag Server. In the Conflict Solver case, generated data replaces received data. In both cases (5) the output data is sent to the Tag Server.

In table 8.9 we give a summary of the features of the four modules of the Computing Server. For the three methods to compute semantic relations between tags

and the Conflict Solver we give the type of input data, the type of computation, the values of the thresholds we used, and the section in this thesis where further details can be found. Regarding the String-based heuristic (see section 5.3.3 on page 130), we indicate the values of the threshold for: (a) the MongeElkan_Soundex metric used to retrieve *semantically linked* tags, (b) the JaroWinkler similarity used to distinguish *spelling variant* tags from merely *semantically linked* tags, and (c) the difference between the MongeElkan_QGram similarity for a pair of tag and its symmetric that allows distinguishing *hyponym* tags from merely *semantically linked* tags. Regarding the other methods, the User-based association method does not require a threshold value because it does not compute a similarity but rather looks for associations rules between tags. Then we indicate the threshold similarity value for the Tag-Tag context method, and, for the Conflict Solver, the consensus ratio above which a relation has gained enough approval to be considered as consensual. Finally, we also give the type of output data, and the type of strategy regarding the merging or update with the results of previous computations.

8.6 Application of the automatic computation of tags semantics

In this section we detail the results we obtained when applying the automatic computation of semantic relations between tags.

8.6.1 Description of the dataset

We have performed the three types of calculation described above on a real-world dataset made of the following parts:

- *delicious* : this dataset comes from the social bookmarking service delicious.com and is made of the tagging instances of users who tagged at least one of their bookmarks with the tag "ademe" as of the 1st of October, 2009.

- *thesenet* : this dataset comes from a database of Ademe which lists all the PhD projects funded by the agency. Each project has been tagged with a list of keywords by the PhD student. We have translated these data into tagging instances by considering each keyword as a tag, each identified project as a tagged resource, and each PhD student as the tagger.

- *caddic* : this is the name of the documents' indexing base of the Ademe's archivists, and this sample is made of all entries of the past five years. In tagging terms, each tagged resource corresponds to a document, and each tag to one keyword from the list of keywords associated to each document. Since no traces of the person who validated each entry is kept, there is only one tagger for all these taggings, namely the archive service.

	String-based heuristic	**Tag-Tag context similarity**	**User-based association rules**	**Conflict Solver**
Input data	Tags + labels	Tagging	Tagging	Semantic relation + user's points of view
Type of computation	Combine standard string-based metrics to infer tags' semantics from morphological similarity	Compute tag similarity based on the distributional aggregation method in the Tag-Tag context	Mine user-base association rules to discover inclusions of sets of users associated to a tag	Detect and propose resolutions between conflicting relations for a given pair of tags
Name and values of thresholds utilized	related = 0.9 spelling variant = 0.9 Hyponym = 0.44	similarity = 0.83	-	consensus ratio = 0.6
Section for detailed explanation	5.3.3 on page 130	5.4.1 on page 134	5.4.2 on page 140	7.2 on page 169
Output data	statements of *related*, *hyponym*, *spelling variant* relations.	statements of *related* relations	statements of *hyponym* relations	approval of existing statements OR proposal of statements of *related* relations for compromise
Merging/Update of previous output	merging	merging	merging	update

Table 8.9: Summary of the features of the four modules of the Computing Server

	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. posts	1013	1425	4765	**7203**
Nb. distinct Users	812	1425	1	**2238**
Nb. distinct Tags / User	1.3	4.6	1439	**4.0**
Nb. distinct Tags / Resource	5.2	4.6	0.3	**1.4**

Table 8.10: Description of the dataset

In table 8.10 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 distinct tagged resources; the number of users; the number of distinct tags per user; and the number of distinct tags per resource.

We can notice here that in the case of thesenet and caddic dataset, there are as much resources as posts since each resource has been tagged only once. We can also remark that the average number of distinct tags per user is equal to $\backsimeq$ 1.3 in the delicious case, whereas the same ratio is equal to $\simeq$ 4.6 in the thesenet case, and 1439 in caddic as there is only one user. This suggests that tags in delicious are more often shared among users than in thesenet, and this is explained by the higher level of specificity of the terms used in the latter case. Last, we can also notice that the average number of distinct tags per resource is significantly lower in the caddic case ($\backsimeq$ 0.3), than in the delicious case ($\backsimeq$ 5.2) and in the thesenet case ($\backsimeq$ 4.6). Indeed, the archivists control very carefully the list of terms they use to index, and eliminate all words that are too closely related to each other, keeping only a limited set of the most precise and unambiguous terms.

8.6.2 Global results of the automatic processing of tags

The different methods of computation are referenced in the following way:

String-based refers to the method described in section 5.3.3 and which looks at the label of tags to provide related, hyponym, and spelling variants relations between tags. This method has been applied on the whole dataset at a time and allows inferring statements between tags across different datasets, unlike the two other methods that depend on the graph structure of each of the folksonomies which, in our dataset, are independent from each other for they don't have common pieces of data (common users, or common tags, or common tagged resources).

User-based association refers to the method described in section 5.4.2.1. Since there were only one user in caddic dataset, this method has only been applied on the two other datasets, each separately.

Tag-Tag Similarity refers to the method described in section 5.4.1 that computes the similarity of tags in the tag-tag context to infer related relations. This method has been applied on each dataset separately.

In table 8.11 we give some details on the results we obtained for each of these methods of computation.

The first thing to notice is that the String-Based method yields far more results (71034 statements) than the other methods, and this is true for all three types of relations. The second type of computation in terms of number of results is the Tag-Tag Similarity which brings a total of 8377 statements of related relations. For this type of method however, most of the relations (97%) comes from the delicious

	String-based	User-based assoc.		Tag-Tag Similarity			Total
	Full dataset	delicious	thesenet	delicious	thesenet	caddic	
Nb. related	59889	-	-	8141	206	30	68633
Nb. Broader/Narrower	10952	106	196	-	-	-	11254
Nb. Spelling variants	3193	-	-	-	-	-	3193
Computation time (s)	20952	5	10	4200	180	300	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

Table 8.11: Description of the results of automatic processing.

dataset, and this 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 (i.e. co-occurring with the same tags, even if they do not necessarily co-occurr together). Indeed, we saw above that tags are more often shared among users in delicious, and we can therefore state that, 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 with the other tags, and thus to be linked with a related relation via the Tag Similarity method.

For the User-based association method, we obtained comparable numbers of relations in the delicious and in the thesenet dataset. This can be partly explained by the fact that the thesenet dataset has around 75% more users than the delicious dataset, and even more distinct tags (around 6 times as many), hence a greater probability of having embedded sets of users of common tags, and consequently more chances to have broader statements between tags.

In the bottom part of table 8.11 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 on distinct pairs.

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. This value does not take into account the time we would save in further computation for the string based method by considering only newly added tags.

8.6.3 Example of automatically computed semantic relations

Let us now look more closely at examples of semantic relations computed with the different methods of the Computing Server.

String-based heuristic

The String Based method has been applied on the full dataset as it allows inferring relations across distinct folksonomies. We give an example of the results obtained with this method for the tag "transports" in figure 8.8, for the tag "energie" in figure 8.9, and for the tag "electrodes" in figure 8.10. For all figures the size of the nodes indicates the number of entering edges (the *in degree*). The green nodes correspond to tags from thesenet and delicious dataset, and blue nodes correspond to tags from caddic dataset. A color code helps distinguish the types of semantic relation: red for broader, blue for narrower, green for spelling variant, and yellow for related.

In these figures we remark that we have sometimes two free tags with the same label and that come each from the delicious or the thesenet dataset: the tag "transport" and the tag "transports" for example in figure 8.8. In these cases, the sibling tags are linked with the spelling variant relations, which allows enriching the results when searching for resources associated to these tags. In these three figures, we remark that the spelling variant relation link tags differing by at most a single letter. Hyponym relation are inferred mostly between a tag and another tag made of the first tag, such as "transport modal" *has broader* "transports" in figure 8.8, or another variant of the first tag, such as "production autonome d energie electrique" *has broader* "energie," in figure 8.9. The *related* relation is obviously not the most accurate since the fact for two tags to be *merely related* is arguable. Nevertheless, we observe relevant relations inferred by the String-Based method such as "electricité" *has related* "electrodes" in figure 8.10, or "transfert" *has related* "transports" in figure 8.8.

Tag-Tag context similarity methods

This method relies on the structure of the folksonomy, so we have applied it to each dataset separately. We show an example of semantic relations computed with this method for the *caddic* dataset in figure 8.11, for the *delicious* dataset in figure 8.12, and for the *thesenet* dataset in figure 8.13. For each figure controlled tags are in blue (and are found only in the caddic dataset) and free tags in green. The relation that is computed for each case is *related.*

As these three dataset are closely connected to the Ademe agency, the type of association computed with this similarity method is biased towards the field that is the most represented in the folksonomies. For instance we found in caddic a link between "eau grasse" (greasy waters) and "déchet de restauration" (restaurants' waste), or a link between "maitrise-énergie" (energie efficiency) and "écocitoyen"

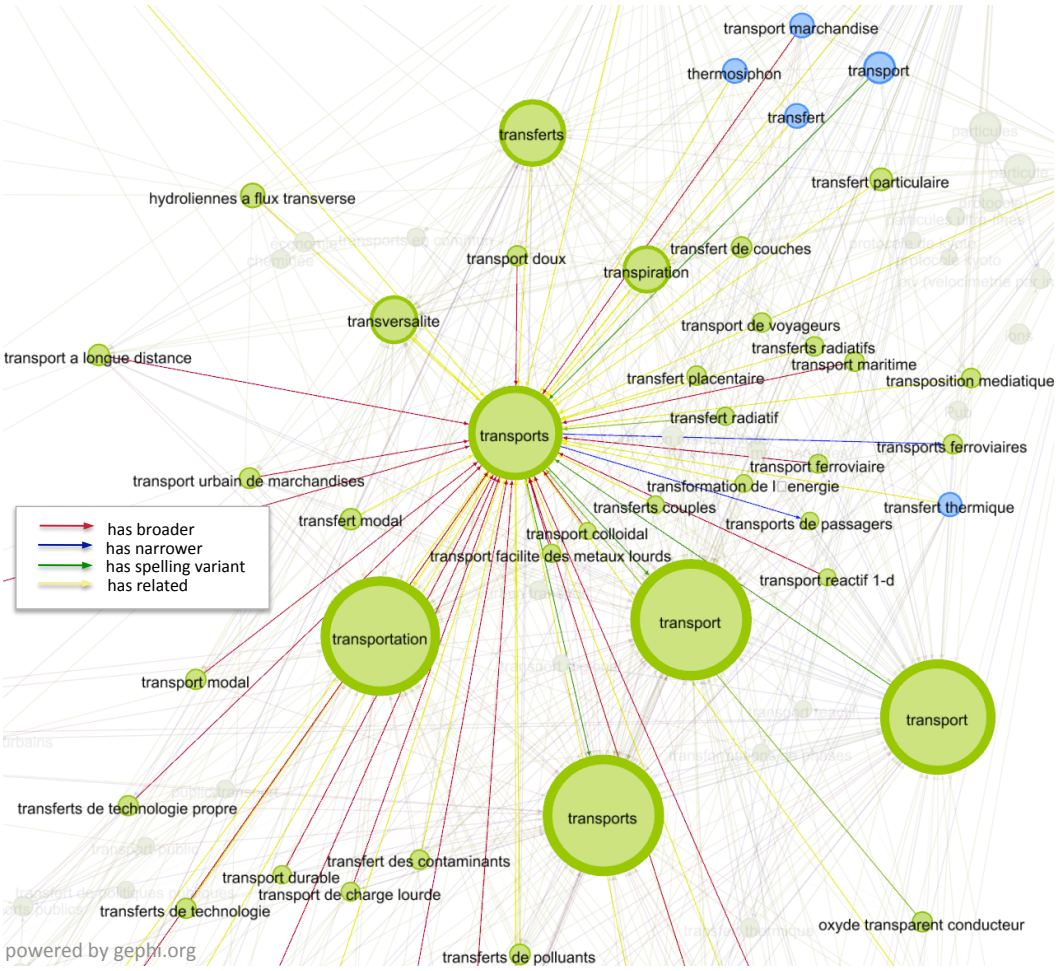


Figure 8.8: Example of the results of automatic processing with the String Based method showing tags linked with the tag “transports”.

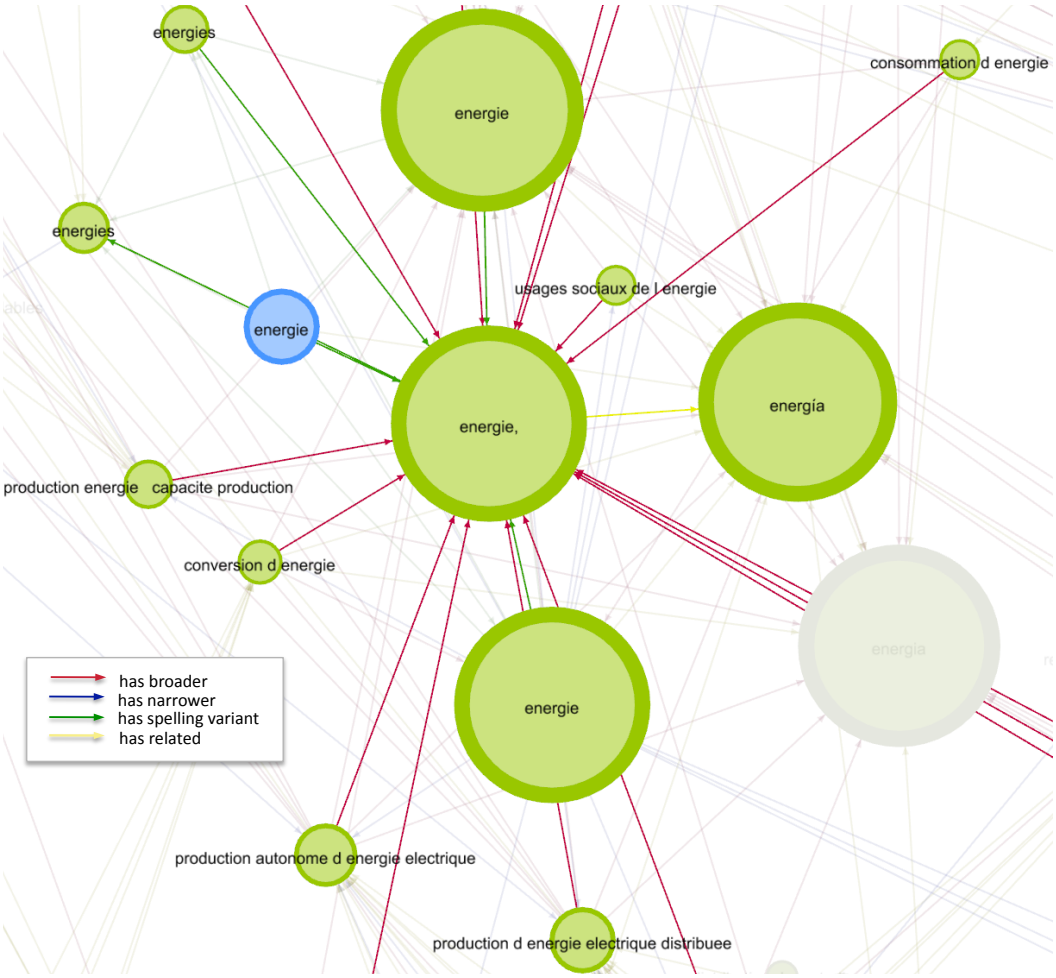


Figure 8.9: Example of the results of automatic processing with the String Based method showing tags linked with the tag “energie”.

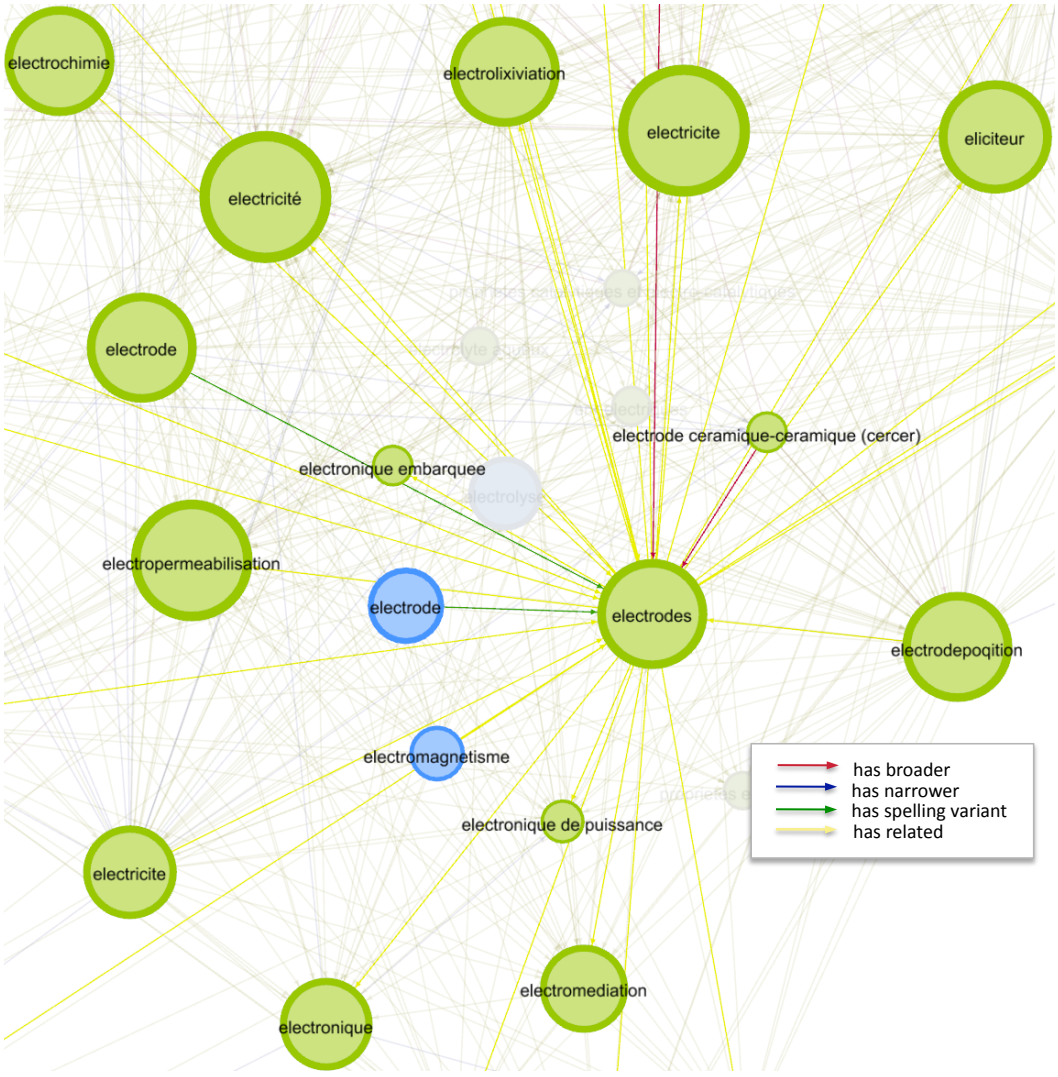


Figure 8.10: Example of the results of automatic processing with the String Based method showing tags linked with the tag “electrodes”.

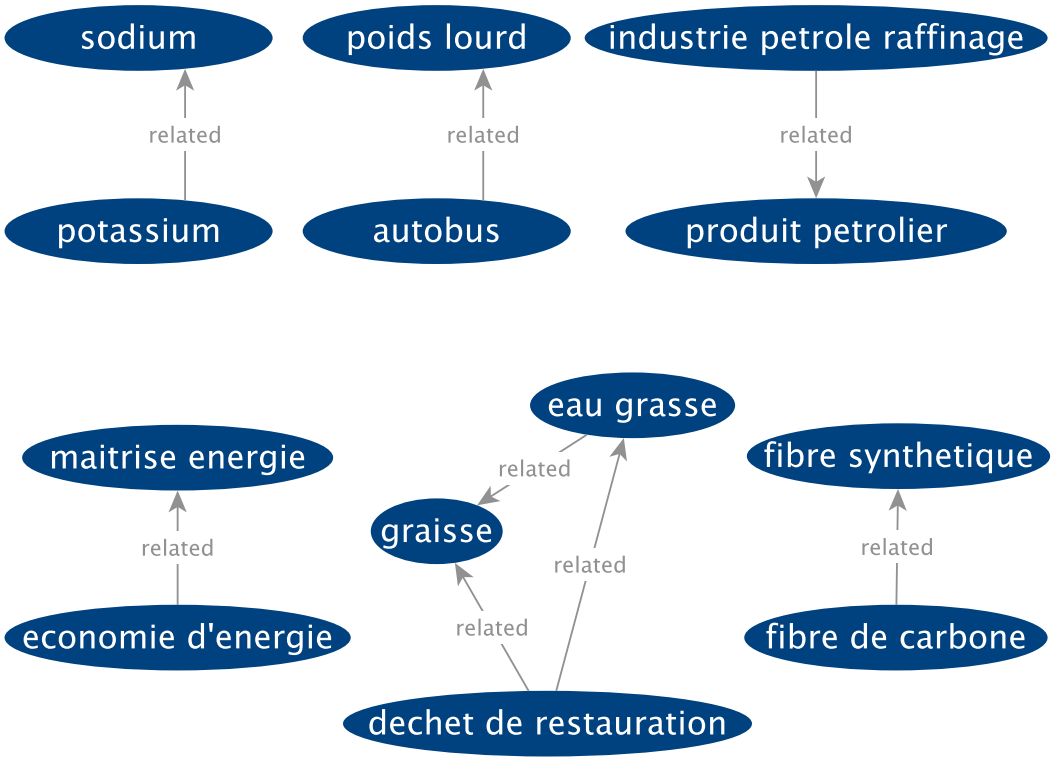


Figure 8.11: Example of semantic relations computed with the Tag-Tag context similarity for the **caddic dataset**

(eco-citizen) in delicious, and finally a link between "architecture" and "projet urbain" in thesenet. All these terms are *related* regarding the field of environment, but may not necessarily be *related* in another domain.

User-based association rules method

This method also depends on the structure of the folksonomy, and therefore cannot provide semantic relations across different folksonomies. As it is based on associations of tags via users, it cannot be applied on caddic since this dataset contains only one single user. The semantic relation that is inferred by this method is broader or narrower, but for simplicity's sake we represented the relation always with *broader* on our graphical visualizations. We show some examples of these relations for the delicious dataset in figure 8.14, and for the thesenet dataset in figure 8.15.

Similarly to the Tag-Tag context similarity method, the relation are relevant regarding the prominent field of knowledge of our target community, which is the environmental issues agency in our case. Thus, the subsumption links are sometimes questionable from an ontological perspective, such as "ministère" (ministry) *has broader* "logement" (housing) for example. But these links are not supposed to follow an ontological approach, and consist rather in a classification of the topics

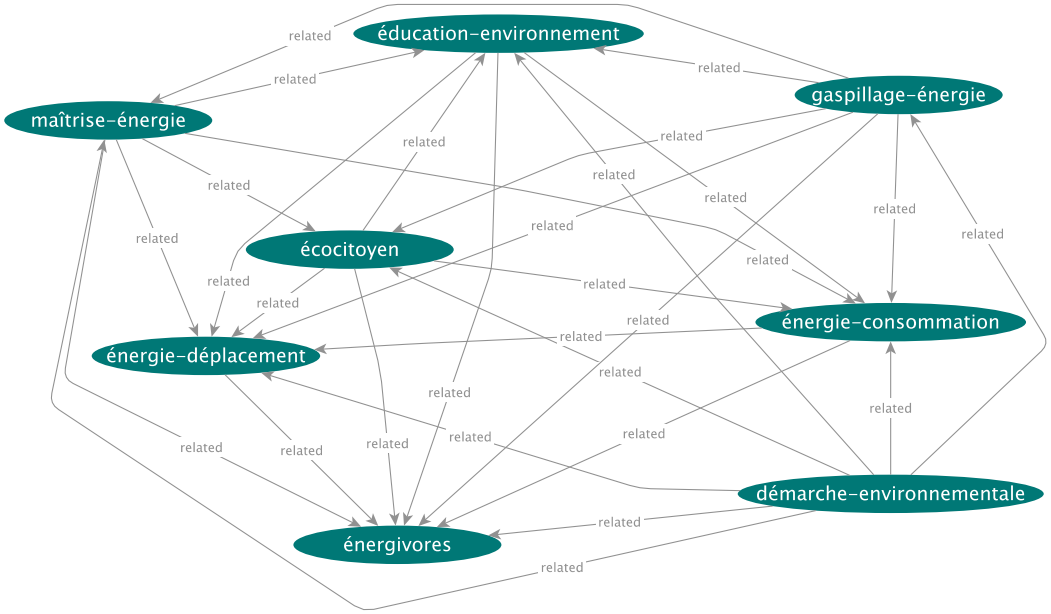


Figure 8.12: Example of semantic relations computed with the Tag-Tag context similarity for the **delicious dataset**

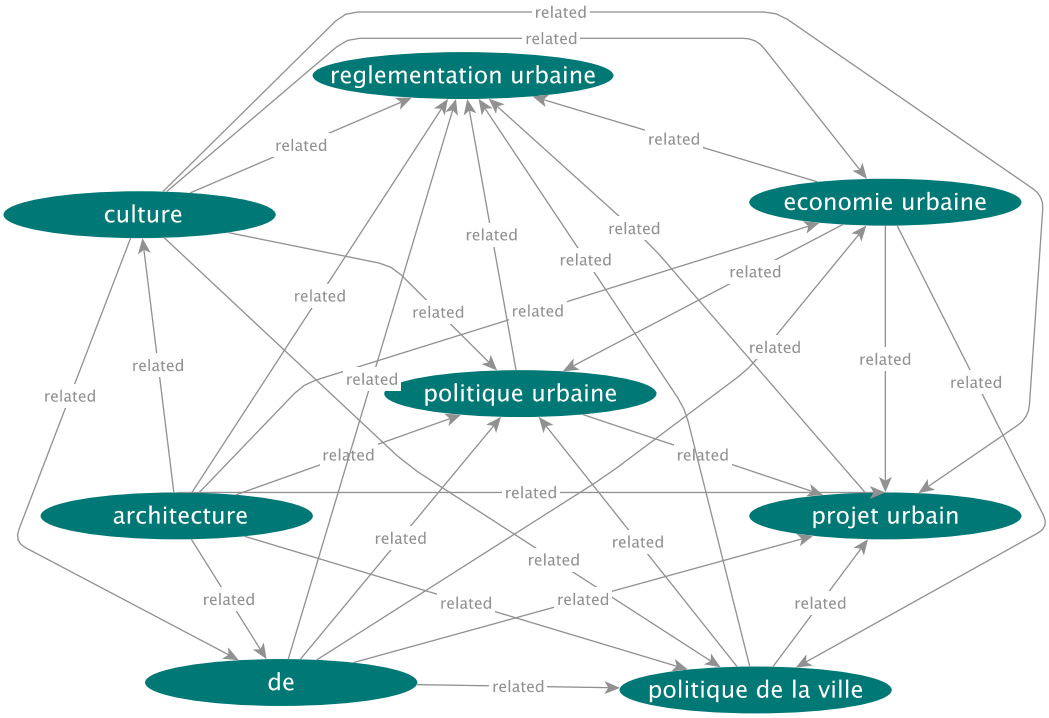


Figure 8.13: Example of semantic relations computed with the Tag-Tag context similarity for the **thesenet dataset**

of interest of our target community. If we take our example, the fact of having the topic "logement" as a broader topic than "ministère" refers certainly to the fact that there exists a ministry in charge of housing questions, but, in the Ademe agency, the topic of "housing" is a broad topic that has been used by more users that "ministère", hence the hyponym relation between these two tags. Furthermore, this computational method provides also for more common sense levels of knowledge such as, in delicious, the tag "electricité" *has broader* "énergie", or the tag "nature" *has broader* "environnement" in the thesenet dataset.

8.6.4 Temporary conclusion

We have covered in previous section and in this section the design of the Computing Server and the results it provides on the Ademe dataset regarding the computation of semantic relations between tags. The three methods we implemented in this server are complementary because they allow computing relations both across and within folksonomies, and they provide the two main types of relations that can be found in a thesaurus, namely *related* and *broader/narrower,* and, in addition, relations linking *spelling variant* tags. The three datasets that compose the Ademe's full dataset also gives a broad view of different types of folksonomies with specific uses of tags. The results of the automatic extraction of tags' semantics with the methods we implemented shows that the inferred relations reflect the field of knowledge of the communities since they are biased by the focus in the tagging instances that are, themselves, representative of the community's interest; in short, folksonomies provide a blur picture of a community's knowledge, and automatically inferred relations reveal *some* of the parts of the knowledge structure of the community.

In order to achieve a clearer and more precise picture, we propose involving users in the validation or correction of the relations automatically inferred thanks to an interface that we present in next section.

8.7 SRTag Editor: Capturing user contributions

This section covers our approach to capture users' contributions on the semantics of tags. We first present the previous studies on collaborative editors of structured folksonomy that motivated our design choices for the interface we propose to allow users to structure tags. Then we detail with concrete examples the functions of the Firefox extension we developed, named SRTagEditor.

8.7.1 Background studies on ontology and structured folksonomy editor

Before introducing in details the interface we propose to capture user's contributions regarding the semantics of tags, we give some insights on the previous

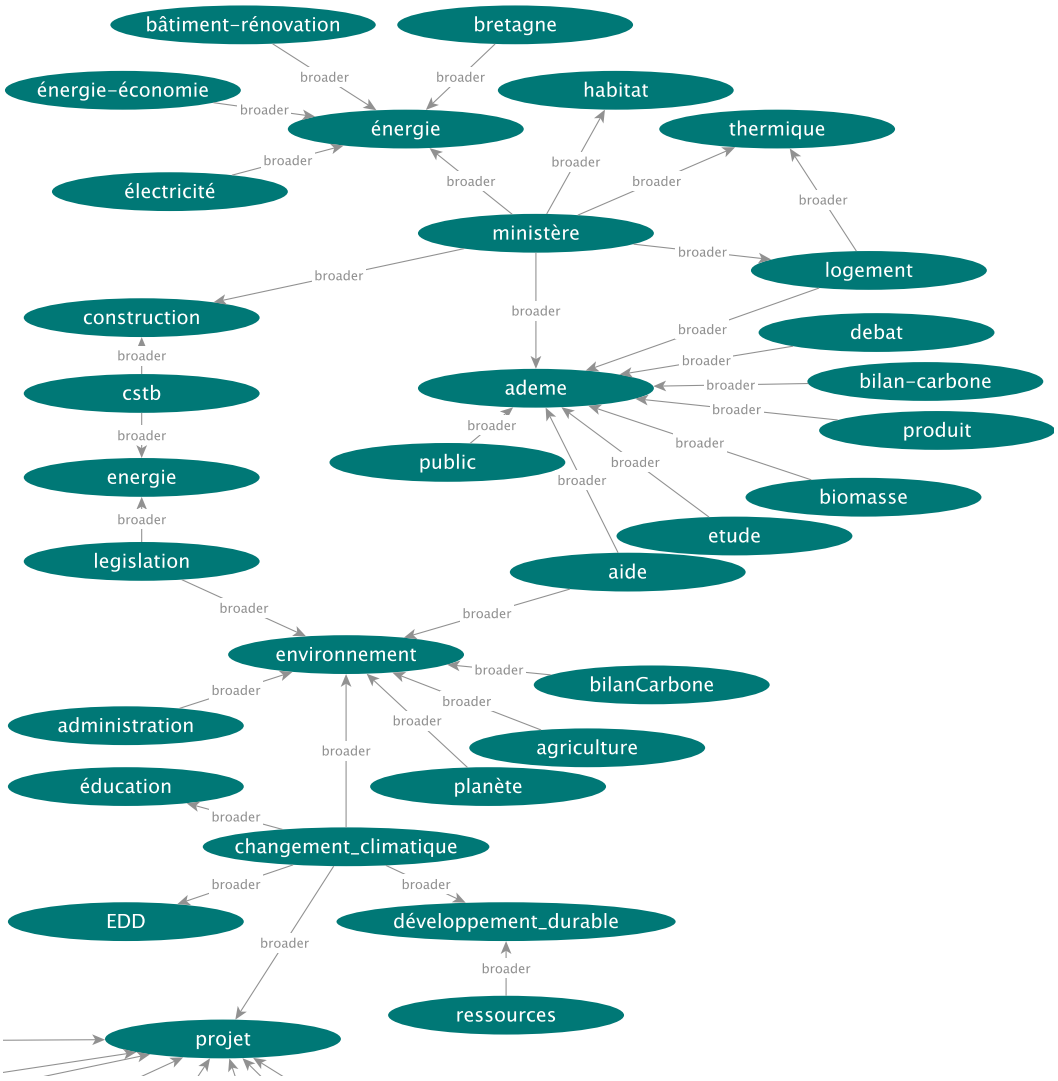


Figure 8.14: Extract of the semantic relations inferred thanks to user-based association rule method and for the **delicious dataset**

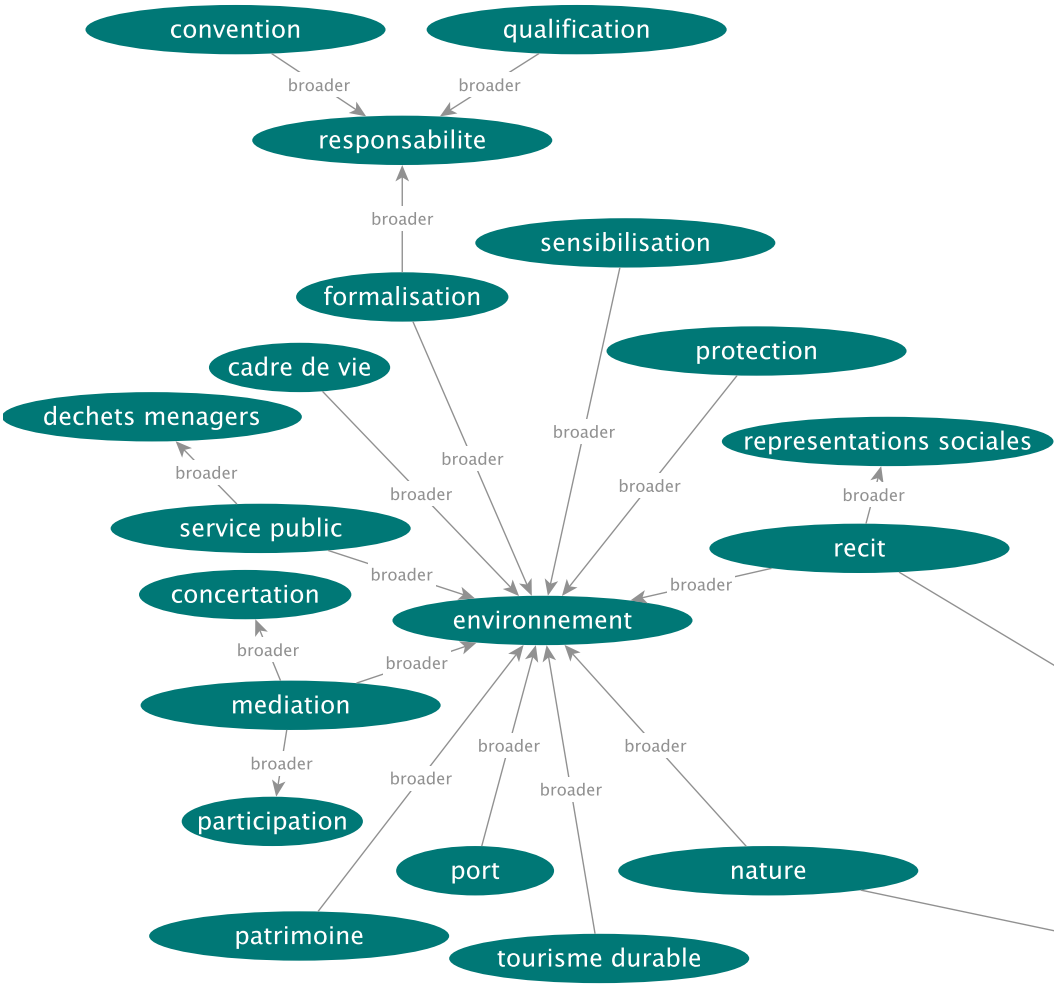


Figure 8.15: Extract of the semantic relations inferred thanks to user-based association rule method and for the **thesenet dataset**

studies and development of collaborative ontology editors conducted in our research team. 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 (ECCO[7]) or a structured folksonomy (SweetWiki by Buffa *et al.*, 2008).

8.7.1.1 Collaborative ontology editor ECCO

The proposal of an interface allowing users to contribute to the edition of the structured folksonomy is a continuation of the efforts initiated in Edelweiss since the ontology editor ECCO. The conception of the second version of ECCO was part of the project e-WOK[8] whose objective was to provide collaborative knowledge engineering tools that take into account the workflow of the organizations working together and sharing ontologies. The goal of ECCO is to involve a heterogeneous group of users, who are not all equally proficient regarding ontology engineering, in the elaboration of domain ontologies. It is aimed at providing a strong collaborative component, in contrast with popular editors such as Protégé[9] for instance. The specificity of ECCO is to span over the different steps of the process of the construction of an ontology, from the extraction of terms (with the help of NLP tools) to the specification of the semantic relations between extracted terms and the formalization of the ontology in OWL-Lite (see section 3.4.5 on page 65 for the details of the methodology of ontology construction).

The second version of ECCO integrated the suggestions of the users of the first version, who expressed the need to edit the ontology with graphical tools, and also to be able to follow the evolution of the ontology. Hence, ECCO2 proposed some features answering to these requests. First, the different steps of the process of ontology construction are explicitly represented in the interface, so that users can go back and forth between different steps and thus have a panoramic view of the progress of the construction of the ontology. The process usually starts with an extraction of terms with the dedicated module, but a hierarchy of concepts can also be imported from graphical concept maps editors such as CmapTool[10]. Then the relations between the extracted terms or concepts from concept maps can be more precisely defined via web-based forms in which users can detail the range and domain of some properties for instance. The evolution of the ontology is recorded along the process so that a proper versioning of the ontology is kept. Furthermore, the validation of the ontology is also lead collaboratively by allowing each user to validate each part of the ontology. These validations, or non-validations, are bound to a user profile so that it is possible to track back each user's point of view. However, the visualization of the ontology was still not graphical, a feature that

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

[8]http://www-sop.inria.fr/edelweiss/projects/ewok/

[9]http://protege.stanford.edu/

[10]http://cmap.ihmc.us/

has been integrated in the collaborative editor of structured folksonomy of the semantic wiki SweetWiki.

8.7.1.2 SweetWiki's folksonomy editor

SweetWiki (Buffa *et al.*, 2008) is a wiki based on semantic technologies and aimed at involving users in the organization and indexing of the content. Pages and documents can be tagged, and these tags can then be semantically structured, similarly to an ontology. SweetWiki thus integrates a folksonomy editor that allows users to organize tags in a hierarchical structure, but also defining tags as ontology concepts and defining formal properties between these tag-concepts. The idea is that, since the edition is collaborative, each single contribution instantly benefit to the whole community. Indeed, semantic relations defined between tags are further exploited to suggest narrower tags when searching wiki pages associated to the searched-for tag. The semantic structuring of tags can also be exploited when monitoring resources about a tag by integrating resources associated to synonym or narrower tags for example. This folksonomy editor also features a graphical navigation of the hierarchy of tag-concepts, as shown in figure 8.16.

A study of the user-friendliness of the folksonomy editor of SweetWiki has been conducted (Peron, 2009). This study was composed of a first analysis done by a professional ergonomist to evaluate the global usability according to several criteria: the guidance offered by the tool to the user, the cognitive load undertaken by the user, the level of control that evaluates the mapping between actions and expected results, the adaptability of the tool to the user's experience, the management of errors, the global coherence of the behavior of the tool, and the way in which the elements of the user interface are set up. The results of this first analysis revealed that the folksonomy editor is cognitively costly to its users. Indeed, navigating the ontology is not easy since the edit operations that can be performed on it (definition of a concept, of some properties linking concepts, etc.) are separated from the visualization of it, which implies to often go back and forth. Moreover the system offers little feedback mechanisms to confirm for instance that the modifications made on a concept have been taken into account, or to prevent accidental manipulations.

A second ergonometric evaluation of the folksonomy editor of SweetWiki has been conducted with 6 users with different level of expertise regarding ontology engineering. This second evaluation consisted in analyzing the activity of the users while they were invited to structure a series of tags and to place new tags into the structured folksonomy. An interview with each user concluded the experiment. Peron (2009) observed that some users were hesitating to modify the existing tags or to propose different semantic structuring. Furthermore, this experiment resulted in a series of divergences between the 6 versions of the structured folksonomy. This second evaluation gave the chance to collect needs expressed by the users. Several users mentioned the need for a better articulation between visualization and editing of the folksonomy. Half of the users did also mention drag'n

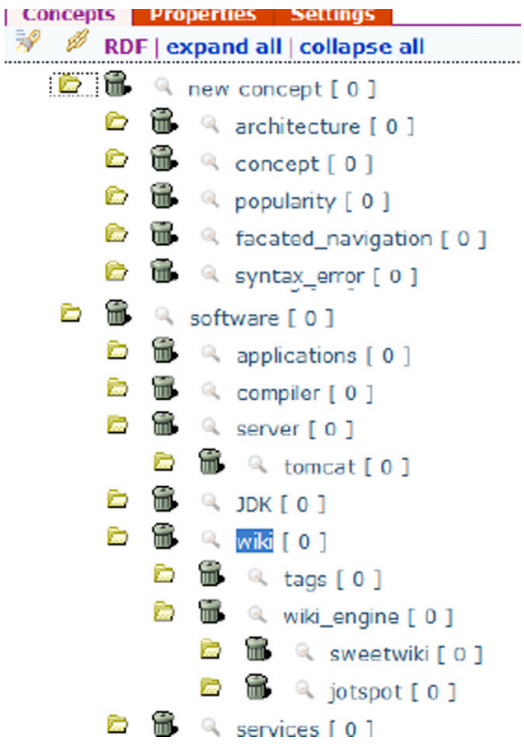


Figure 8.16: Folksonomy editor of SweetWiki

drop functionalities in order to minimize the manipulations required to structure tags. Some users proposed also integrating suggestions from automatic agents to help and guide them while structuring tags. Regarding the collaborative aspect, several users pointed the lack of means to directly interact with the other users, as a dedicated chat room, or a way to properly annotate and explain their actions. Finally, all the users mentioned the need to be able to track back all the editing actions.

8.7.2 Micro-editing of the folksonomy embedded in everyday tasks

The ergonometric 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.

Then, since our model to capture the structuring of tags supports diverging points of view, we wanted to allow users to contribute to the semantic structuring of the folksonomy while keeping as low as possible the cognitive overload that this task may involve. To achieve this goal we propose integrating simple and non-obtrusive structuring functionalities within everyday tasks of users. For instance,

in our target community at Ademe, this can consist in capturing the expertise of the expert-engineers when they browse the corpus of Ademe resources.

In the interface we implemented, we have integrated semantic structuring functions within search tasks. Our hypothesis is that users may be keener on providing for a little amount of efforts several times in a day rather than dedicating a longer time slot to structure tags. In order for this hypothesis to be efficient, the tag structuring functions should be absolutely optional so that users can decide to keep focusing on the navigation task without being annoyed by a demand from the system to validate or correct a semantic relation. This interface has been implemented as a Firefox extension, and this choice reinforce the natural integration within search tasks since web browsers are often the dedicated tool to search for information. The development has been done using the XML User Interface Language (XUL) from the Mozilla foundation[11].

Our proposal consists in an interface for navigating the folksonomy in which tags are suggested and ordered according to their semantic relations with the current searched-for tag (see figure 8.17). 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.

Hence our interface can be seen as a micro-editor which is focused on the editing of the semantic relationships of the tags around the searched-for tag, in contrast with the editor of SweetWiki (Buffa *et al.*, 2008), in which users can edit the whole structured folksonomy at a time, making it a kind of macro-editor of the folksonomy. Moreover, our interface allows a complete synchronization between the visualization and the editing operation thanks to the drag and drop feature.

Then, each user of the system, modeled with the `srtag:SingleUser` class, can structure the relations around the searched-for tag as they wish in order to maintain their own point of view that is captured thanks to the SRTag model. In chapter 7, we will see how it is possible also to integrate related tags coming from the points of view of the other users while preserving the logical coherence of their own point of view.

8.7.3 Examples of micro-editing actions

Let us now present in details 2 examples of editing actions that are allowed by SRTagEditor. One to simply reject a relation between two tags, and the other one to correct a relation by rejecting it and proposing a new one thanks to drag and drop manipulation.

In figure 8.18 we show in details the interaction we propose to allow users rejecting a semantic relation with the current searched-for tag “energie”. In this case,

[11] https://developer.mozilla.org/en/xul

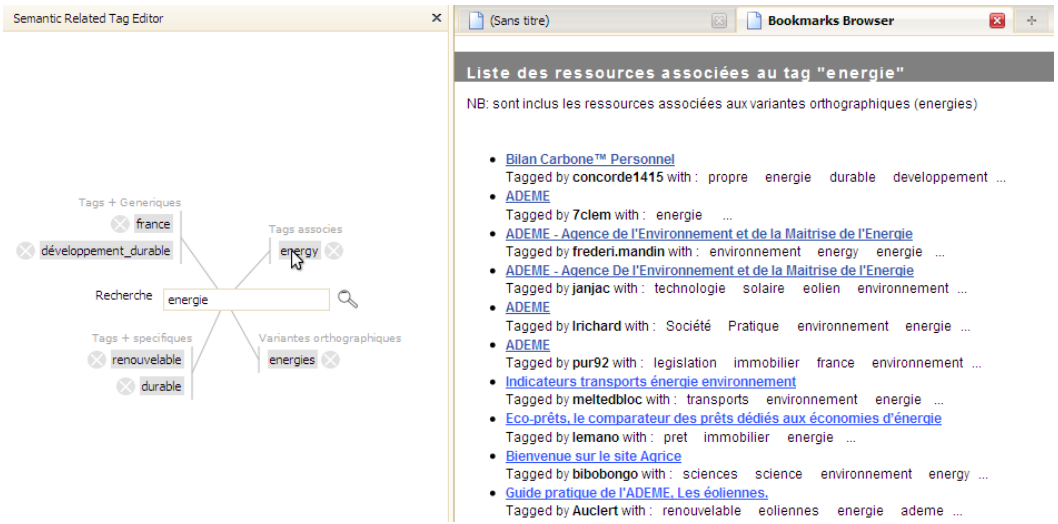


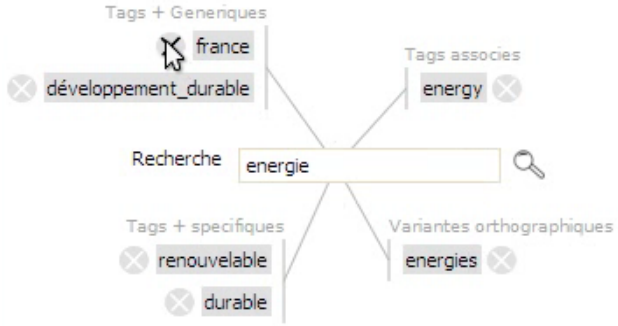
Figure 8.17: Screenshot of SRTagEditor, a Firefox extension seamlessly integrating tag structuring capabilities within an interface for navigating the folksonomy. On the right side are displayed the resources associated to the current searched-for tag "energie". On the left side tags semantically linked to the searched-for tag are displayed and arranged according to their semantic relation: *broader* (top left), *related* (top right), *spelling variant* (bottom right), or *narrower* (bottom left).

users simply have to click on the cross besides the tag they want to reject, and this action is interpreted as the rejection of the relation (whose type is given by the position of the tag) between the searched-for tag and the rejected tag. The annotations generated by this action are shown in listing 8.2. Lines 1-4 correspond to the statements for the relation that is rejected and that states that the tag "energie" has a *broader* tag "france". Lines 6-8 correspond to the rejection by user "anonym" of this statement.

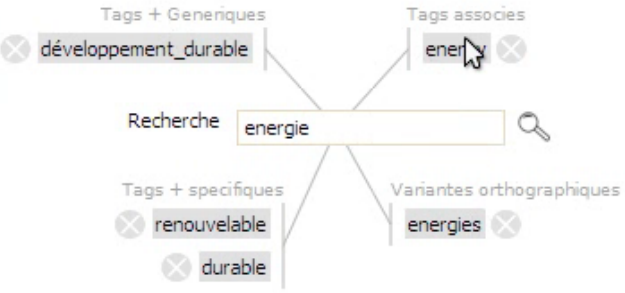
The second example details the action of correcting a semantic relation by proposing another one. In figure 8.19 we show the action of dragging and dropping the tag "energy" from the *related* area towards the *spelling variant* area of the SRTagEditor. This manipulation corresponds to two distinct actions and the cor-

Listing 8.2: RDF annotation corresponding to the action of a user "anonym" who rejected the relation {"energie" *has broader* "france"}, as shown in figure 8.18.

```
1 <scot:Tag rdf:about="http://srtag.org/examples#tag_energie"
2   cos:graph="http://srtag.org/examples#broader_01">
3   <skos:broader  rdf:resource="http://srtag.org/examples#tag_france"/>
4 </scot:Tag>
5
6 <sioc:User rdf:ID="user_anonym">
7   <srtag:hasRejected rdf:resource="http://srtag.org/examples#broader_01"/>
8 </sioc:User>
```



(a)



(b)

Figure 8.18: User rejecting the semantic relation {"france" *is broader than* "energie"} in SRTagEditor by clicking on the cross besides the tag "france".

Listing 8.3: RDF annotation corresponding to the action of a user "anonym" who rejected the relation {"energie" *has related* "energy"} and proposed instead the relation {"energie" has *spelling variant* "energy"}. These annotations correspond to the drag and drop action shown in figure 8.19.

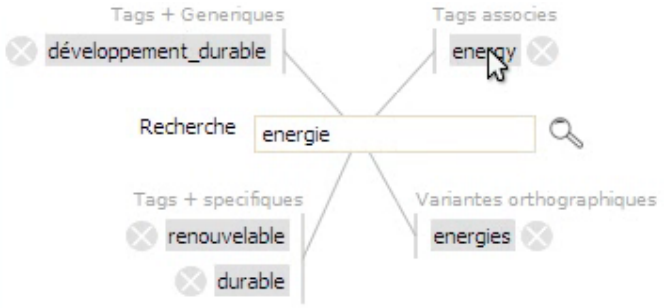
```
1  <!-- energie has related energy -->
2  <scot:Tag rdf:about="http://srtag.org/examples#tag_energie"
3    cos:graph="http://srtag.org/examples#related_01">
4    <skos:related  rdf:resource="http://srtag.org/examples#tag_energy"/>
5  </scot:Tag>
6
7  <sioc:User rdf:ID="user_anonym">
8    <srtag:hasRejected rdf:resource="http://srtag.org/examples#related_01"/>
9  </sioc:User>
10
11 <!-- energie has spelling variant energy -->
12 <scot:Tag rdf:about="http://srtag.org/examples#tag_energie"
13   cos:graph="http://srtag.org/examples#spelvar_01">
14   <skos:closeMatch  rdf:resource="http://srtag.org/examples#tag_energy"/>
15 </scot:Tag>
16
17 <sioc:User rdf:ID="user_anonym">
18   <srtag:hasProposed rdf:resource="http://srtag.org/examples#spelvar_01"/>
19 </sioc:User>
```

responding annotation are shown in listing 8.3. Lines 2-5 correspond to the statement that links the tag "energie" with the relation *related* to the tag "energy", and the first consequence of the drag'n drop is the rejection of this relation shown in lines 7-9. The second consequence is the creation of a new statement that links the tag "energie" to the tag "energy" with the relation *spelling variant* (lines 12-15), which is modeled in our system with the property `skos:closeMatch`. Then lines 17-19 account for the fact that user "anonym", who performed the drag and drop here, has proposed the latter statement.

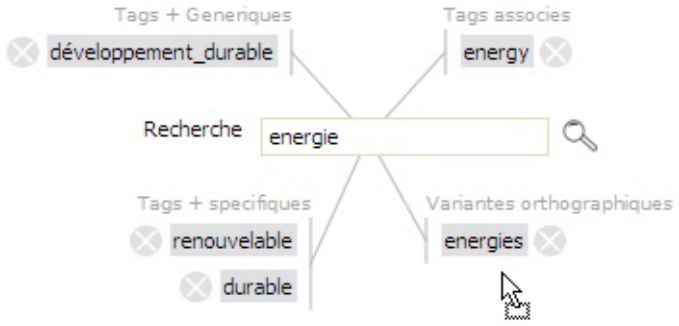
In figure 8.20 we show the status of the SRTagEditor after both editing actions presented above have been performed. We see on the right side of the interface the list of the resources associated to the searched-for tag. We can observe that, since the tag "energy" is now a spelling variant of the searched-for tag "energie", the resources associated to the tag "energy" are now included in addition to the resources associated to the tags "energie" (the searched-for tag) and "energies" (another spelling variant). This feature can also serve as a feedback mechanism and instant incentive since users benefit directly from the structuring of tags by getting richer and more precise results.

8.8 Reporting of the conflicts to the Referent User

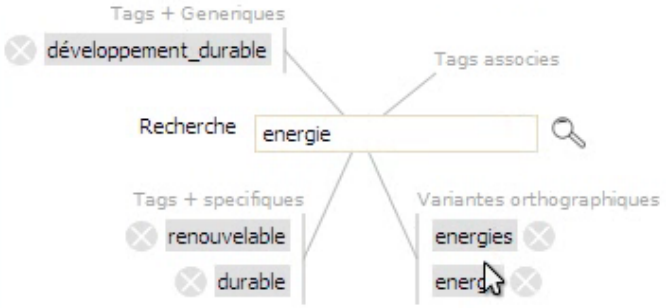
After capturing individual points of view, the conflicts are detected by the Conflict Solver. The Conflict Solver proposes solutions by approving the most consensual



(a)



(b)



(c)

Figure 8.19: User dragging and dropping the tag "energy" from the *related* area towards the *spelling variant* area of SRTagEditor.

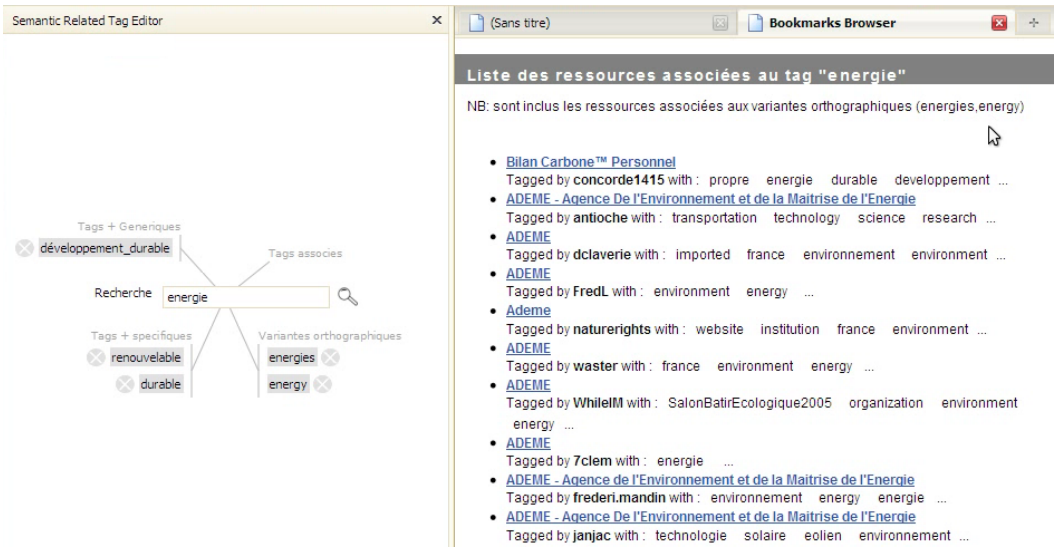


Figure 8.20: Screenshot of the SRTagEditor interface after both editing action presented in figures 8.18 and 8.19. By comparing with the same screenshot but before these editing actions (see figure 8.17), we see that on the right side the resources associated to the tag "energy" are now included.

relation from the conflicting one if a consensus is reached, otherwise it proposes the *related* relation as a compromise.

The outcome of the Conflict Solver can be exploited by a client that helps the Referent User maintain a global and conflict-free structuring of the folksonomy. In chapter 7 (see section 7.3) we detailed the principles of the visualization of the structured folksonomy including the conflicting status of some relations, as well as the consensual or debated status of some other relations. We showed how this can help the Referent User choosing the relations he/she wants to approve and the relations that he/she wants to reject.

The global structuring can be achieved thanks to an interface that allows the referent user to view all relations, with the conflicts highlighted so that he/she can gives his/her own opinion. The client dedicated to the referent user is currently being implemented at the time of the writing of this thesis, but we give here the first elements that consist in a global map of the structured folksonomy (shown in figure 8.21) and a report that gives some global statistics and invite the Referent User to gives his choice on the pairs of tags with conflicting relations (see figure 8.22).

The map, of which we show a sample in figure 8.21, gives a global view of the structured folksonomy with all the relations that have been proposed by users and automatic agents. Furthermore, it allows the Referent User pointing out the relations that are conflicting, but also the relations that are good candidates to be added to the thesaurus when only approved, or, on the contrary, relations that can be erased because they have only been rejected and are thus more likely to be

meaningless.

The second type of interface (see figure 8.22) provides first global statistical information :

- the total number of tags linked with a relation to another tag, and the number, among them, that are involved in a pair with conflicts.
- the total number of pairs of tags linked with a relation, and among them, the number of pairs with : conflicts, or with a single relation only approved, or a single relation only rejected, or a single relation proposed only by an automatic agent, or a single relation debated.

Then, for each case of pairs of tags with conflicts, it details the number of approvals and rejections. This part is synchronized to the global map so that when the user clicks on a conflicting relation on the global map, then the details are shown for this pair of tags. Finally, the Referent User is invited to submit his/her choice.

8.9 Discussion of the scalability of the system

As mentioned in the introduction, our main target are online communities of interest, that is, groups of people who share, publish, coauthor, comment, or tag resources via dedicated online platforms. The ISICIL solution, to which we contributed here with the tagging module, is an example of such a platform. We address in this section the scalability of our approach for enriching folksonomies when the folksonomy dataset or the number of relationships between tags grow.

The first step in the folksonomy enrichment lifecycle concerned with scalability is the automatic computing of semantic relations between tags. We saw in table 8.11 on page 217 that the string-based heuristic is the most complex and the most costly in computation time. This is due to the fact that we apply a combination of 3 string-based metrics (from which the MongeElkan_Soundex metric has quadratic complexity according to the length of the tags) to all the possible pairs of tags. However, the string-based heuristic is the only incremental method since it does not depend on the structure of the folksonomies and therefore the relative similarity between tags has to be computed only for the newly added tags, saving a significant amount of time[12]. Then, we see in table 8.11 on page 217 that the 2 other methods completed the computation in less than a hour and a half for the full Ademe's dataset. The second method, the user-based association, corresponds to the graph projection aggregation presented in section 3.3.2.3 on page 36,which has a logarithmic complexity. Therefore, this method should not be a problem when the size of the dataset increases. The third method, the tag-tag context similarity, has a high complexity due to the count of co occurrence of tags. However,

[12]Indeed, if we had checked the similarity for n^2 pairs of tags, n being the number of tags of the folksonomy at the first iteration, then if k tags have been added afterwards, $k \lll n$, then similarity according to the heuristic string-based method has to be computed only for the $k.n$ new pairs of tags, considering that $k.n \lll n^2$.

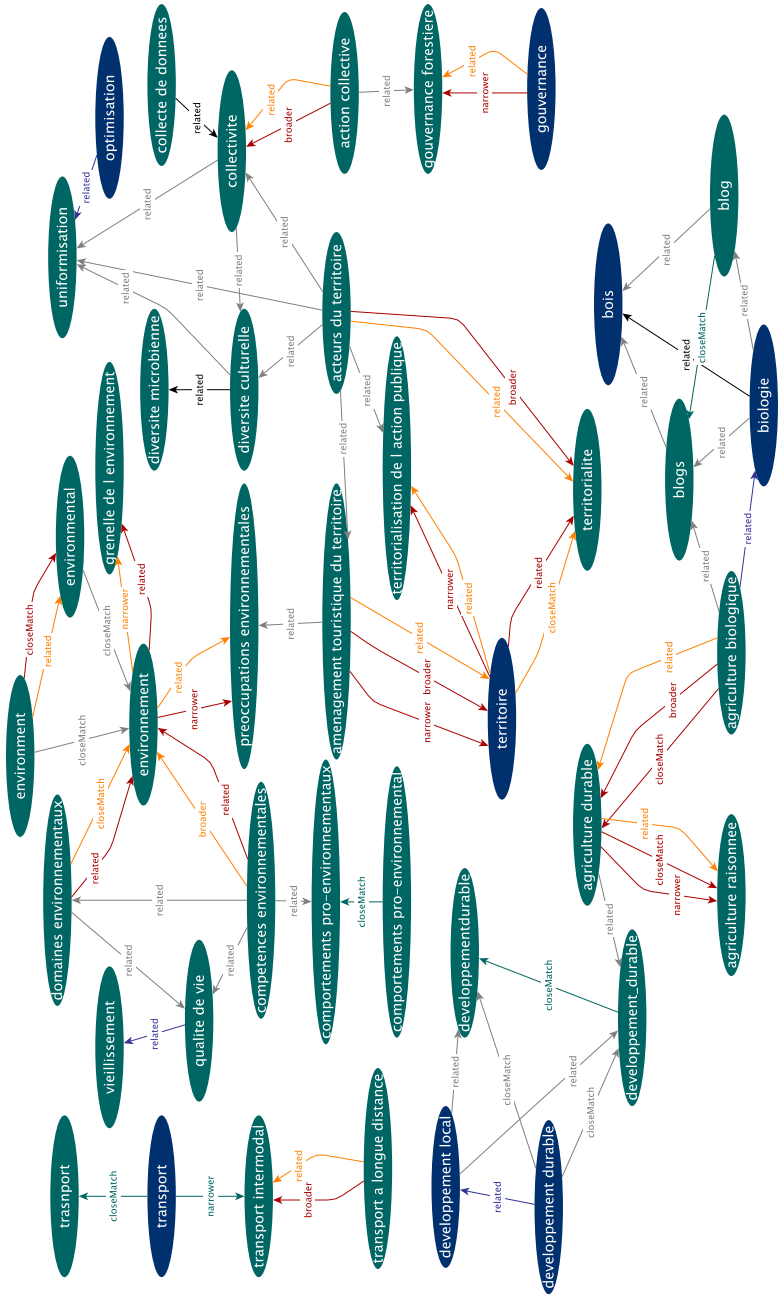


Figure 8.21: Global map of the structured folksonomy showing the different relations between tags. The conflicting relations are in red, and the proposal of the conflict solver in orange. Relations that do not conflict with any other are (1) in green when they have only been approved by users, (2) in blue if they have been both approved and rejected, (3) in black if they have only been rejected, and (4) in grey relations proposed by automatic agents but not yet reviewed by human agents

Figure 8.22: Example of a report made for the Referent User thanks to the results of the Conflict Solver

it is possible to optimize the computation by selecting most representative sub-folksonomies, as shown by Benz *et al.* (2010) (see conclusion in chapter 5).

The second concern regarding scalability comes from the complexity of the SPARQL queries that are used in the Tagging Server to retrieve related tags (see section 6.4.5 on page 161), or in the conflict solver to retrieve the pairs of tags for which more than one relation has been approved (see section 7.2 on page 169). Regarding this aspect, the complexity of SPARQL queries has been addressed by Schmidt *et al.* (2010). Schmidt *et al.* gives the fundamental elements for the evaluation of the complexity of SPARQL queries, evaluating the cost linked to SPARQL operators (`AND`, `OPTIONAL`, `FILTER`, and `UNION`) used in addition to regular triple patterns. They show for instance that the `OPTIONAL` operator alone increases greatly the complexity towards the PSPACE class. And this operator is used in our approach to detect conflicts (see section 7.2 on page 169) and to enrich each individual point of view with the contributions of the other agents (see section 7.4 on page 181). However, the `OPTIONAL` operator in these queries are used to check the absence of certain triples, especially triples linking the approval or rejection of a relation by a given user or a given type of agent. The method to check the absence of certain triples using the `OPTIONAL` operator is called "negation by failure"[13] and the use of this operator should no longer be needed to achieve this task in SPARQL 1.1 with the introduction of the NOT EXIST operator[14]. We can therefore believe that the complexity of the queries used in our approach can be greatly reduced when translated in SPARQL 1.1. Finally, we decided to limit the number of semantic relations that can be stated between two tags to four SKOS relations (`related`, `broader`, `narrower`, and `closeMatch`). This design choice allows us to limit the time to execute the SPARQL queries involved in the management of the multiple points of view regarding the structuring of folksonomies. We have shown that the current implementation of our approach to folksonomy enrichment is able to handle the dataset of an organisation as big as the Ademe agency, and the growth of the data regarding the multiple points of views can also be handled thanks to the limited number of possible relations.

8.10 Conclusion

In this chapter we have presented our implementation of a semantic tagging-based system that fosters the enrichment of a folksonomy initially flat and shared among a community of users. The different modules of this system implement the lifecycle of the enriched folksonomy we presented in chapter 4. This is illustrated by figure 8.23 where we reproduce the lifecycle schema with the different modules corresponding to each step. The development of a semantic tagging system is one of the goal of the ISICIL project in which we took part. The modules of this system are based on a common framework and, for the tagging-related part, are made of

[13] see http://www.w3.org/TR/sparql-features/#Negation
[14] see http://www.w3.org/TR/sparql11-query/#func-filter-exists

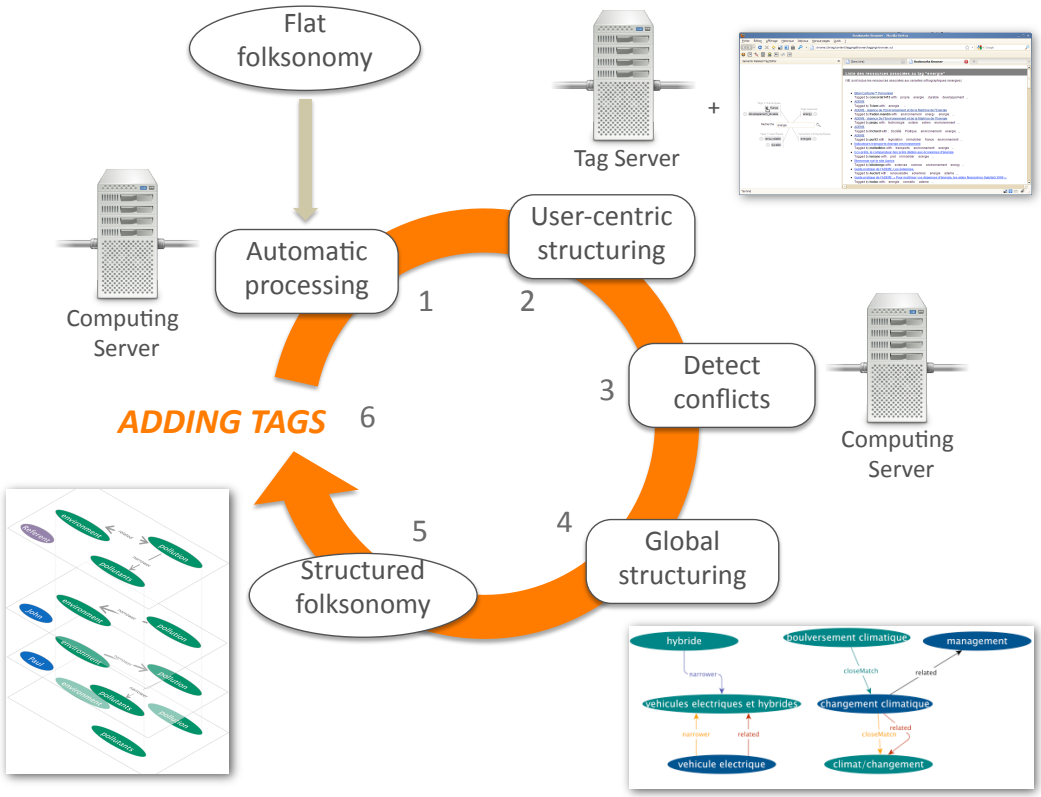


Figure 8.23: Enriched folksonomy lifecycle and the corresponding elements of the tagging-based system implemented.

two servers and a series of clients communicating with them.

The enrichment cycle starts with automatic processing of the folksonomy that is performed by the Computing Server. The Computing Server is in charge of bootstrapping the process by computing semantic relations according to three different methods: String-based heuristic, the Tag-Tag context similarity method, and the User-based association rules mining method. To illustrate this step of the process, we have applied these three types of computation to a real-world dataset collected from the Ademe agency and made of three subparts: taggings taken from delicious.com and related to Ademe, the indexing of Ademe's corpus with controlled tags, and the tagging of research projects by the PhD students who conducted them. The results showed the complementarity of the computational methods that can either provide cross-folksonomies relations, but also for three different types of relation: *related*, *hyponyms*, and *spelling variant*. In terms of algorithmic complexity, the last two types of computation are relatively costly and, overall, not incremental since we have to analyze the whole folksonomy to compute the similarity of newly added tags. The result of this first step helps avoid the cold start effect by providing for a set of relations reified, thanks to the SRTag model, in order to be able to capture users' points of view.

The second step of the lifecycle consists in letting users contribute by validating, rejecting, or proposing semantic relations between tags. This is made possible by SRTagEditor, a dedicated interface implemented as a Firefox extension and that integrates lightweight editing functions within a tool to navigate the folksonomy. For achieving this purpose, it uses the functions implemented in the Tag Server. The Tag Server is in charge of the core-tagging functions and thus allows users to create tags and tagging instances, but also to search for tagged resources. The Tag Server also provides on demand the semantic relations between tags, and also enables users to contribute to the structuring of the folksonomy by rejecting or proposing relations. The design of the interface allowing users to contribute to the structuring of the folksonomy has integrated previous experiences gained in our team on the conception of collaborative ontology and structured folksonomy editors (ECCO and the folksonomy editor of SweetWiki). Our hypothesis for this scenario is that users may be keener on providing for a little amount of efforts several times in a day while they search for tags rather than dedicating a longer time slot to structure tags. The interface we propose can be seen as a micro-editor of structured folksonomies as it allows users to modify the semantic environment of the searched-for tag. This semantic environment consists in tags suggested thanks to the Tag Server, and placed in four different areas of the interface according to their semantic relation with the searched-for tag. Users can then simply, and *optionally*, reject a relation by clicking on a cross besides a suggested tag, or modify an existing relation by dragging an dropping a tag from one area to another.

The third step of the cycle consists in detecting and solving conflicts that may arise between users' points of view. This is achieved by a fourth module of the Computing Server, named the Conflict Solver. The Conflict Solver detects the pairs of tags with several conflicting relations and proposes a relation as a solution. The outcome is a set of approval of relations or *related* relations proposed as a compromise when none of the conflicting relations reached a parametrized level of consensus.

The output of the Conflict Solver is exploited for the fourth step where a Referent User proceeds to a global structuring of the folksonomy. To help the Referent in this task, we construct a global map of the structured folksonomy. This global map integrates the conflictual situation of some pairs of tags with several relations, following a color code to highlight the solutions proposed for the conflicts, but also the relations that gained an absolute consensus, or on the contrary, relations that can be omitted because only rejected by users. The result of this step is a global and coherent point of view regarding the structuring of the folksonomy.

The fifth step is actually the result of our approach to the semantic enrichment of folksonomies. Indeed, at this point the structured folksonomy contains several points of view that benefit from the contributions of others thanks to a set of rules that allow enriching one point of view while keeping its local coherence. This feature is implemented in the Tag Server and is used for instance by the SRTagEditor when looking for the tags semantically linked to the searched-for tag. Hence, we can look at the structured folksonomy as a layered structure where each layer cor-

responds to the point of view of a user containing the relations he has approved or rejected. Then, the relations of each layer contributes to the global point of view of the Referent User. This global and coherent point of view is then utilized as a reference when enriching each individual point of view with others' points of view. The sixth step corresponds to the cycle restarting to take into account the newly added tags or semantic relations.

As a result, we are able to offer the users of our targeted communities a tagging-based system that helps them navigating the folksonomy with suggestions of relevant tags to extend or enrich the search of information. This system also fosters, as a natural second effect of its use, the emergence of a structuring of the folksonomy, bootstrapped by automatic processing, and regulated by the community. Our approach to folksonomy enrichment thus provides synergetic ways of combining automatic processing with human input, by supporting multiple and diverging points of view along the whole cycle.

CHAPTER 9

Conclusion and perspectives

9.1 Summary of the contributions of this thesis

This thesis is an attempt at bridging social web and semantic web approaches in the context of knowledge-based systems used by online communities of interest. Such communities are found on the Web, but organizations are willing today to encourage similar practices and uses by importing the tools of typical Web 2.0 platforms, and in particular tagging-based systems on their intranets. This research took place in the context of the ISICIL project that is devoted to providing science and technology monitoring tools for collecting, organizing, and sharing information in the context of organizations. Our contributions thus concern the improvement of tagging-based systems with the help of semantic technologies in order to help their users better exploit tagging data to browse and organize their shared corpus, but also to help administrators and referent users in their task of building a consensual and global structured knowledge representation. Before addressing the perspectives and future works, let us now review our contributions.

Modeling heterogeneous tagging data We proposed in this thesis the NiceTag model that consists in representing tag actions primarily as a link between a tagged resource and a sign used to tag. This link can be typed with one of the properties accounting for the diversity of use of tags. The link between a single tag and a tagged resource is then encapsulated within a named graph, a technical choice that allows reifying and typing the tag action (without the burden of standard RDF reification) to account for other dimensions regarding the nature or other factual traits of tags. Its flexibility makes it eligible to serve as a pivot between current tagging models, and therefore contribute to improving the interoperability between tagging data repositories, while allowing a full expressivity to represent tags from a multiplicity of facets. NiceTag is a first step towards enriching tagging data by allowing us to filter out tags by their use and avoid some cases of ambiguity such as when using the tag "blogs" to say that a resource *is* a blog, or to say that a resource *is about* blogs. Thus, the semantic enrichment brought by NiceTag is complementary to the enrichment that consists in structuring tags relatively to each other, and that we call folksonomy enrichment.

Folksonomy enrichment lifecycle We proposed a complete life-cycle of the process of semantically enriching folksonomies. This life-cycle is grounded on a

scenario-based analysis that accounts for the current activity of Ademe's members, which can be turned into opportunities to contribute. Indeed, we found a convergence between our research objectives and the archivists' goal of turning their controlled but flat vocabulary into a thesaurus, including in the process the contributions of both experts and external members. This enrichment consists in semantically structuring the folksonomy with thesaurus-based relations stated between tags. The specificity of our approach lies in a synergistic combination of users' contributions and automatic processing, while supporting multiple points of view. The different steps of the folksonomy enrichment life-cycle are decomposed as follows:

- **Bootstrapping with automatic processing**. The first step of the process consists in automatically computing semantic relations between tags with a combination of different types of methods. We contributed in this respect by proposing a novel method that combines standard string-based similarity metrics, which were selected thanks to a benchmark we conducted in order to evaluate the ability of such metrics to detect different types of semantic relations. The result is a heuristic string-based method that is able to retrieve three types of semantic relations. This method performs best for detecting *spelling variants*, as expected, but is also capable of detecting *hyponym* relations in cases such as "pollution" which is broader than "soil pollution", and *related* relations such as between "energy" and "energetic". Another benefit of this method is that it is independent from the folksonomy structure and therefore (a) it is incremental, as computations for newly added tags do not require recomputing the similarity values for all the tags, and (b) it can be used to link tags across different folksonomies or to link tags to concepts of termino-ontological resources. For the second part of our contribution regarding automatic processing, we have adapted two state of the art methods, based on the analysis of the structure of the folksonomy, by including SPARQL queries as part of the computation. The first one computes the cosine similarity for the distributional aggregation of tagging data in the Tag-Tag context, as proposed by Cattuto *et al.* (2008) who showed that this similarity brings associative relations, which are called *related* in SKOS. To compute this similarity, we first count the co-occurrence of tags with a SPARQL query applied on our tagging data modeled with NiceTag. Then, the principle of this method is that tags having similar patterns of co-occurrence, but not necessarily co-occurring together, are linked with the *related* relation. The second state-of-the-art method, proposed by Mika (2005), looks for inclusions of sets of users of tags in order to infer *hyponym* relations. The principle of this method is that if the set of users of the tag "biological agriculture" is included in the set of users of the tag "sustainability", then we can infer that the tag "biological agriculture" is narrower than the tag "sustainability". In our implementation, we proposed a SPARQL query that allows us to retrieve pairs of tags following this association rule. The outcome of this method is

a set of annotations stating *hyponym* relations between tags. All these methods for automatically computing tag relations have been implemented in the Computing Server, which is a part of the ISICIL solution that we contributed to develop.

- **User centric structuring.** The next step in the process consists in letting each user contribute by correcting automatically computed relations or proposing new ones, while maintaining his own point of view independently of the other users. This step is realized thanks to a model, SRTag, and an interface that extends a tool for navigating the folksonomy with tags structuring capabilities. SRTag allows representing diverging points of view regarding the semantic relation between tags. This model encapsulates the triple asserting a semantic relation within a named graph, reifying the relations in a more flexible manner than with standard RDF reification. The reified relation can then be bound to the users with properties marking their approval or disapproval. As a result we are able to capture more than one relation for a given pair of tags, allowing each user to maintain his own structuring of the tags. In order to capture the contributions of users, we proposed integrating functions of micro-editing of the structured folksonomy within a navigation tool. The aim of this interface is to benefit from everyday tasks of users without overloading them. Indeed, each user is able to propose or correct the semantic relations around the tag he is searching for thanks to optional and simple drag and drop manipulations. This step of the process is based on the SRTag model to capture diverging points of view and is implemented by SRTagEditor, a front end client developed as a Firefox extension that makes use of web services that we contributed to develop.

- **Detection of conflicts**. The next step of the process consists in detecting the conflicts that may arise between all users' points of view. The Conflict Solver is part of the Computing Server and looks for pairs of tags with several relations. Then, it checks whether one of the conflicting relations reaches a parametrized level of consensus, and if this is the case, it approves this relation, otherwise, it proposes the *related* relation as a compromise. We have conducted an experiment among several members of Ademe who where asked to pick one relation for a sample set of pairs of tags. The results showed that conflicts are likely to arise, even among a small set of users, and that some pairs of tags are more likely to cause conflicts, as well as some types of relation that are more debatable than others.

- **Global structuring.** The outcome of the Conflict Solver is then exploited to build global representations of the structured folksonomy where conflicting pairs are highlighted, but also pairs linked with a relation that found a consensus among users who expressed themselves, or, on the contrary, relations disapproved by all users who expressed themselves. This global representation can then be used to help a referent user, like Ademe's archivists, to

choose his solution for the conflicts and, more generally, to maintain his own point of view. This global point of view is used when enriching the points of view of each user as a reference that helps choose a relation for the conflicting pairs of tags.

In this thesis, we have shown the validity of our approach to folksonomy enrichment by completing a first lap of the life-cycle we proposed. The outcome of this process is a semantically structured folksonomy where possibly diverging points of view coexist and can also mutually enrich one another thanks to a global and coherent point of view. This semantic enrichment helps users while navigating the folksonomy as it allows them to either enrich their search results by including *narrower* or *spelling variant* tags, but also to enlarge their search by suggesting *broader* or *related* tags. Ambiguous tags, which have several possible meanings, can also be discovered thanks to this way of structuring, as we support multiple inheritance in the hierarchy of tags, and ambiguous tags are likely to have several broader tags, each one accounting for a specific meaning of the tag. Finally, the coherent and global structuring of the folksonomy that emerges from the contributions of all users consists already in a *draft* thesaurus than can be further refined. Hence, our approach also brings a solution to the bottleneck effect when acquiring knowledge by allowing all members of the community to contribute to the final knowledge representation, while avoiding overloading them thanks to a tailored automated assistance.

9.2 Publications

During this thesis, our results have been presented in a series of publications that we introduce below.

Our first contribution consisted in investigating the current research work that tries to bridge ontology-based and folksonomy-based systems (published in french (Limpens *et al.*, 2008b), and in English (Limpens *et al.*, 2008a)). These surveys have allowed us to cover as exhaustively as possible the current approaches to automatically extracting tag semantics that have been later included in our approach to folksonomy enrichment. We also participated in collaborative position papers exploring the possibilities brought by approaches bridging Social Web and Semantic Web for e-learning applications (Henri *et al.*, 2008, 2009), or more generally for the future of social networking (Ereteo *et al.*, 2009b).

After this first phase of understanding and discussion of the state of the art approaches, we developed two models aimed at the semantic enrichment of tagging and folksonomies. The NiceTag model is thus an attempt to fully embrace the richness of tags' usages and forms that we can find on the Web. A first version (Limpens *et al.*, 2009c) laid down the basic principles of NiceTag and an extended version (Monnin *et al.*, 2010) explored the nature of speech acts of tags and proposed an augmented modelisation of tag actions in this respect. This model has

been integrated in the solution developed in the ISICIL project as the model to describe tagging data. SRTag (Semantically Related Tag) is the second model we proposed and is aimed at describing the semantic structuring of tags with thesauri-like relations, while allowing diverging points of view. It is the foundation of our multi-point of view approach to folksonomy enrichment first presented in French (Limpens *et al.*, 2009b), and latter in English (Limpens *et al.*, 2009a).

In the last phase of this thesis, we introduced in (Limpens *et al.*, 2010) the complete lifecycle for the semantically enriched folksonomy that includes several contributions. This publication presents the customized versions of state-of-the-art automatic processing in addition to the string-based method that we developed. This publication also details the results of the experiment we conducted among a set of Ademe's users that showed that conflicts regarding the semantic structuring of tags are likely to arise even among a few users. This experiment also showed the benefits such an approach could bring by allowing administrators of a knowledge-based system to build a structured representations fed by all the members of the community.

9.3 Discussion

Let us now address some still-incomplete points of our contributions before covering the perspectives it opens up for future works.

Regarding the string-based heuristic, in order to evaluate the accuracy of automatically inferred semantic relations, we have asked an expert from Ademe to validate a list of 88 pairs of tags linked with a specific relation. The thresholds for each case of semantic relation have then been chosen according this referent sample dataset. The validity of these thresholds for other datasets can thus be discussed. It would have indeed been interesting to be able to evaluate our inferred relations with a larger dataset and also in another domain. One could also imagine applying the string based methods on the pairs of tags of a thesaurus, and compare the result of the computation with the relations from the thesaurus. However, the relations existing in a thesaurus are no less arbitrary, and cannot be considered as an absolute golden standard. Furthermore, very specific terms are often absent from thesauri, and this is a type of vocabulary for which, we believe, string-based methods have the most potential regarding the fact that terms linked in such specific domains often share common tokens, such as in "offshore wind turbines" and "wind turbines" that we can find in the Ademe vocabulary for instance.

Another arguable point is the lack of a full scale evaluation of our method. In particular, we did not have the chance to evaluate the user-friendliness of SRTagEditor, the interface aimed at capturing individual contributions, on a large set of users. The evaluation of the multi-points of view approach has also been done with a small set of users, and we did not have the possibility to have a longer term experiment to get feedback on how the tasks of structuring tags help or disturb the users in their everyday tasks. However, the design of SRTagEditor has

benefited from the experience of our in-house ergonomist who worked on several collaborative ontology editors that have been evaluated among partners of several research projects[1]. Next, the experiment we did regarding the multiple points of view allowed us to validate this approach as we observed a significant ratio of conflicts, even among a small group of users.

Lastly, one can question the validity of our approach at the scale of the web since we claim to have based our approach on a usage analysis of our targeted end users. If we stated that current tagging based systems have a lot to gain from such analysis, we also believe that our approach is still relevant for other types of end users. Indeed, the usage analysis helps optimize the process, and for instance, in the system we propose, we took into account the current activity of the archivists and included it as an opportunity to monitor the process and to provide a reference in cases of conflicts. In the absence of referent users of the type of the Ademe archivists, an automatic agent, the Conflict Solver, replaces it by detecting and proposing solutions to the conflicts. Thus, we believe that our approach can also be applied to social bookmarking platforms and bring substantial improvements.

9.4 Improvements and perspectives

Deployment in ISICIL partners

The first and short term perspective is the deployment of the method presented in this thesis within the ISICIL solution presented in chapter 8. The ISICIL solution will be tested at Ademe and Orange Labs as a knowledge management tool to assist monitoring activities. In particular, beyond the developpment of a tool that will be used by the Ademe's archivists to help them structure their controlled folksonomy (a scenario already envisionned at the beginning of this thesis) the close collaboration with the Ademe agency has led to the development of another tool aimed at retrieving people according to their field of expertise. In this context, our method for structuring folksonomies will be exploited to enrich the profile of members of the agency. Indeed, members of Ademe provide for a list of keywords describing their fields of expertise, but these lists are often scarce, and the navigation within this database may be greatly enhanced if these keywords are semantically linked together, and also linked to the tags used for other resources. This application of the semantic enrichment of folksonomies opens up promissing perspectives in a world where the topics of interest of members of organizations rapidly evolve, making static catalogues of expertise harder to maintain.

User interfaces

In this thesis, we focused on setting up simple but robust models to ground the process of folksonomy semantic enrichment, and we have developed one inter-

[1] the e-Wok Hub project, http://www-sop.inria.fr/edelweiss/projects/ewok/ and the Palette project http://palette.ercim.org/

face to navigate the folksonomy and capture individual contributions. Some other interfaces could also exploit the possibility opened by our models. For instance, a tagging interface could make use of NiceTag and allow users to specify the type of relation they wish between the tag and the tagged resource. However, in order to keep the simplicity that made the success of tagging tools, this choice should remain as unobtrusive as possible and the designer of the interface should also carefully select a subset of relevant properties for their targeted end users. For instance, some properties oriented towards personal use of tags may be more relevant for a social bookmarking tool than for tools used by archivists to tag documents of a library corpus.

Regarding the semantic structuring of tags, Huynh-Kim Bang *et al.* (2008) proposed a simple syntax to allow users to structure tags at tagging time. Similarly, we envision integrating micro-editing functionalities in the tagging interface by proposing that users specify the semantic relations that exist between the tags they have chosen for the tagged resource. In addition, semantically linked tags can also be suggested to the user in order to help him improve his tagging with meaningful tags he may not have thought of.

Regarding the navigation within the folksonomy, the interface we developed, SRTagEditor, already makes use of the semantic relations to suggest semantically linked tags in order to refine or broaden the search. In this interface, the resources tagged with spelling variant tags are automatically included in the results. However, one could imagine also including narrower tags, but the risk, then, is to increase noise. In this case, the interface we envision could list the tags automatically included and would allow the user to refine the results by selecting or deselecting the tags he wants to include. In this manner, the results are enriched thanks to the semantically linked tags, and the user still keeps control on the filtering. Another improvement regarding the enhancement of the navigation could also consist in a ranking algorithm that would take into account the data concerning the points of view. For instance, we could weight the relations according to the level of agreement they reached, and use this data to rank the semantically linked tags to be included.

Another type of interface that could exploit our model is a thesaurus or lightweight ontology editor that would include the multi-points of view structured folksonomy resulting from our approach. Such an interface could first draft a thesaurus from the global point of view. Then, all along the lifecycle, it could suggest new relations when they reach a parametrized level of agreement, or suggest the addition of a new concept from a tag that newly appeared and gained a certain level of usage.

Additional automatic processing

Another improvement would consist in integrating additional automatic processing methods for the detection of semantic relations. In chapter 3, we covered a number of methods that we did not have the opportunity to include in our system.

In particular, methods for inferring subsumption relations can reinforce the limited number of such relations in the results we obtained with our set of automatic methods. Furthermore, in order to evaluate automatic methods, it would also be possible to exploit the capture of users' agreement or disagreement with certain semantic relations in order to measure, for each type of computational method, the level of approval or rejection it gains.

Application to semantic social network analysis

Administrators of information systems in organizations or online communities of interest may be willing to detect subgroups of users centered on given specific topics, or bound by a specific type of link, for instance by the documents they all tag, or by the fact they used the same tags for the same resources. These questions are investigated by Ereteo *et al.* (2009a) who proposed adding a semantic layer to the social networks analysis. To this end, our method can bring new opportunities to type users and the links between users. For instance, when a user has expressed himself about several relations made on a given tag, we can infer that his interest is stronger than someone who merely used this tag. The act of structuring tags entails a greater degree of involvement and is more likely to reveal an expertise on the corresponding fields of knowledge. Furthermore, if two users agree on a number of relations between tags, we can infer a potentially stronger link between these users.

Deeper usage analysis

We also believe that it is possible to further improve our approach with a closer analysis of the activity of users. This analysis can help identify other kinds of tasks which could be turned into opportunities for the semantic enrichment of shared knowledge. Furthermore, an accurate knowledge of user activity is crucial for the effective adoption of a technological solution. What is at stake here is the identification of the critical points that make possible the emergence of a synergy between the goals of the users and the assistance provided by the system. In this respect, we believe that the success of attempts to bring Semantic Web technologies into platforms of knowledge exchange lies in their ability to set up virtuous circles where users are willing to provide slightly more inputs because they *perceive* the benefit of doing so, and in the end get more in return than they gave in the first place. For example, social tagging platforms, and especially social bookmarking systems, managed to set up such a virtuous circle when users realized that the time they had spent tagging was compensated by the time they earned when they wanted to retrieve the resources they had tagged. As James Hendler put it, "a little semantics goes a long way"[2], and we strongly believe that knowledge sharing platforms of the Web can turn this motto into a tangible enhancement from the users' perspective.

[2]http://www.cs.rpi.edu/~hendler/LittleSemanticsWeb.html

9.5 Towards an open Web

We would like to conclude this dissertation with a few thoughts on our vision of the future of the Web. The Web, and its evolution, with the Social Web, towards a space of social interaction and exchange, consists in a revolution in the history of media that brought, maybe for the first time, true means for users to contribute actively to the published resources, but also to the elaboration of a genuine space of expression and debate. In this regard, the Web has become a public space and, as such, has taken a political dimension in the sense that citizens should have the right to take part in the way the Web is shaped. Social tagging already made an advancement (yet limited in its current implementation) by allowing users to influence the indexing of their favorite resources by *voting* with their tags (Gruber, 2007). However, the Social Web has also seen the emergence of centralized platforms that control vast amount of information and data, and that wish to turn this data into powerful levers to become unavoidable in the paths towards knowledge.

In this regard, we see the Semantic Web as an opportunity for users to further contribute to the technical means through which they access knowledge. The Linking Open Data initiative, grounded on the Semantic Web principles of exposing and interconnecting data, presents a novel model that enables to publish data in open standard formats and in decentralized repositories in contrast to the emerging monopolies of information access. Moreover, by contributing to the construction of structured representations of their knowledge, for instance thanks to semantically augmented tagging platforms, users can influence in a conscious way and enhance greatly the manifold possible paths between relevant resources. The Linking Open Data initiative, complemented by a multidisciplinary approach to the developement and reflexion on the Web, as proposed by the Web Science Trust[3] for example, will contribute, we believe, to shaping the Web as an open space of exchange and collaborative construction of knowledge. It is to such a vision of the Web that we sincerely wish to contribute in our future work.

Freddy Limpens

[3] http://webscience.org/home.html

APPENDIX A

SPARQL query and results to count the number of tags present in the GEMET thesaurus.

This annex contains the SPARQL query and the corresponding results that count the number of tags from the Ademe dataset that are present in the GEMET thesaurus[1], a reference thesaurus in the fiels of environment and ecology. We first show the results for the controlled tags of the Ademe's archivists, and then for the other freely contributed tags from other members of Ademe and delicious.com users.

Listing A.1: SPARQL query and results to find the freely contributed tags (scot:Tag) that are present in the GEMET thesaurus (french version)

```
<?xml version='1.0' encoding='UTF-8'?>
<cos:result xmlns:cos='http://www.inria.fr/acacia/corese#'>
<cos:tquery>
<![CDATA[prefix bookmark: <http://www.polytech.unice.fr/bookmark.rdfs#>
prefix srtag: <http://ns.inria.fr/srtag/2009/01/09/srtag.rdfs#>
prefix foaf: <http://xmlns.com/foaf/0.1/>
prefix sioc: <http://rdfs.org/sioc/ns#>
prefix scot: <http://scot-project.org/scot/ns#>
prefix xsd: <http://www.w3.org/2001/XMLSchema#>
prefix gemet: <http://www.eionet.europa.eu/gemet/2004/06/gemet-schema.rdf#>
prefix skos: <http://www.w3.org/2004/02/skos/core#>
prefix tag: <http://www.holygoat.co.uk/owl/redwood/0.1/tags/>
prefix nicetag: <http://ns.inria.fr/nicetag/2009/09/25/voc#>
prefix cos: <http://www.inria.fr/acacia/corese#>
prefix rdfg: <http://www.w3.org/2004/03/trix/rdfg-1/>
prefix dc: <http://purl.org/dc/elements/1.1/>
prefix irw: <http://www.ontologydesignpatterns.org/ont/web/irw.owl#>
prefix ctag: <http://commontag.org/ns#>
prefix svic: <http://www.ademe.fr/2009/svic-schema.rdfs#>
select *
where
{?t rdf:type scot:Tag .
?g rdf:type skos:Concept .
?t rdfs:label ?l1 .
?g skos:prefLabel ?l2 .
filter (?t != ?g)
filter (str(?l1) = str(?l2)) }
group by ?l1 group by ?l2
]]></cos:tquery>
<cos:info><![CDATA[
0.03 s for 376 projections
```

[1] http://www.eionet.europa.eu/gemet

```
]]></cos:info>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='t'/>
<variable name='g'/>
<variable name='l1'/>
<variable name='l2'/>
</head>
<results>
<result>
<binding name='t'><uri>http://ns.inria.fr/isicil/id/tag/banlieue</uri></binding>
<binding name='g'><uri>http://www.eionet.europa.eu/gemet/concept/8182</uri></binding>
<binding name='l1'><literal >banlieue</literal></binding>
<binding name='l2'><literal xml:lang='fr'>banlieue</literal></binding>
</result>
<result>
<binding name='t'><uri>http://ns.inria.fr/isicil/id/tag/hydrobiologie</uri></binding>
<binding name='g'><uri>http://www.eionet.europa.eu/gemet/concept/4088</uri></binding>
<binding name='l1'><literal >hydrobiologie</literal></binding>
<binding name='l2'><literal xml:lang='fr'>hydrobiologie</literal></binding>
</result>
<result>
<binding name='t'><uri>http://ns.inria.fr/isicil/id/tag/strontium</uri></binding>
<binding name='g'><uri>http://www.eionet.europa.eu/gemet/concept/8144</uri></binding>
<binding name='l1'><literal >strontium</literal></binding>
<binding name='l2'><literal xml:lang='fr'>strontium</literal></binding>
</result>
<result>
<binding name='t'><uri>http://ns.inria.fr/isicil/id/tag/hydraulique</uri></binding>
<binding name='g'><uri>http://www.eionet.europa.eu/gemet/concept/4085</uri></binding>
<binding name='l1'><literal >hydraulique</literal></binding>
<binding name='l2'><literal xml:lang='fr'>hydraulique</literal></binding>
</result>
```

Listing A.2: SPARQL query and results to find the controlled tags (svic:MC) from the vocabulary of Ademe's archivists that are present in the GEMET thesaurus (french version)

```
<?xml version='1.0' encoding='UTF-8'?>
<cos:result xmlns:cos='http://www.inria.fr/acacia/corese#'>
<cos:tquery>
<![CDATA[prefix bookmark: <http://www.polytech.unice.fr/bookmark.rdfs#>
prefix srtag: <http://ns.inria.fr/srtag/2009/01/09/srtag.rdfs#>
prefix foaf: <http://xmlns.com/foaf/0.1/>
prefix sioc: <http://rdfs.org/sioc/ns#>
prefix scot: <http://scot-project.org/scot/ns#>
prefix xsd: <http://www.w3.org/2001/XMLSchema#>
prefix gemet: <http://www.eionet.europa.eu/gemet/2004/06/gemet-schema.rdf#>
prefix skos: <http://www.w3.org/2004/02/skos/core#>
prefix tag: <http://www.holygoat.co.uk/owl/redwood/0.1/tags/>
prefix nicetag: <http://ns.inria.fr/nicetag/2009/09/25/voc#>
prefix cos: <http://www.inria.fr/acacia/corese#>
prefix rdfg: <http://www.w3.org/2004/03/trix/rdfg-1/>
prefix dc: <http://purl.org/dc/elements/1.1/>
prefix irw: <http://www.ontologydesignpatterns.org/ont/web/irw.owl#>
prefix ctag: <http://commontag.org/ns#>
prefix svic: <http://www.ademe.fr/2009/svic-schema.rdfs#>
select *
where
{?t rdf:type svic:MC .
?g rdf:type skos:Concept .
```

```
?t rdfs:label ?l1 .
?g skos:prefLabel ?l2 .
filter (?t != ?g)
filter (str(?l1) = str(?l2)) }
group by ?l1 group by ?l2
]]></cos:tquery>
<cos:info><![CDATA[
0.02 s for 410 projections
]]></cos:info>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
<variable name='t'/>
<variable name='g'/>
<variable name='l1'/>
<variable name='l2'/>
</head>
<results>
<result>
<binding name='t'><uri>http://ns.inria.fr/isicil/id/mc/sucre</uri></binding>
<binding name='g'><uri>http://www.eionet.europa.eu/gemet/concept/8185</uri></binding>
<binding name='l1'><literal >sucre</literal></binding>
<binding name='l2'><literal xml:lang='fr'>sucre</literal></binding>
</result>
<result>
<binding name='t'><uri>http://ns.inria.fr/isicil/id/mc/banlieue</uri></binding>
<binding name='g'><uri>http://www.eionet.europa.eu/gemet/concept/8182</uri></binding>
<binding name='l1'><literal >banlieue</literal></binding>
<binding name='l2'><literal xml:lang='fr'>banlieue</literal></binding>
</result>
<result>
<binding name='t'><uri>http://ns.inria.fr/isicil/id/mc/subvention</uri></binding>
<binding name='g'><uri>http://www.eionet.europa.eu/gemet/concept/8165</uri></binding>
<binding name='l1'><literal >subvention</literal></binding>
<binding name='l2'><literal xml:lang='fr'>subvention</literal></binding>
</result>
<result>
<binding name='t'><uri>http://ns.inria.fr/isicil/id/mc/hydraulique</uri></binding>
<binding name='g'><uri>http://www.eionet.europa.eu/gemet/concept/4085</uri></binding>
<binding name='l1'><literal >hydraulique</literal></binding>
<binding name='l2'><literal xml:lang='fr'>hydraulique</literal></binding>
</result>
```

APPENDIX B

Questionnaire for the experiment conducted in chapter 7 aimed at capturing user's points of view for a sample of pairs of tags

Nom Prénom :	
Poste :	
Profil en quelques mots-clés :	

Indiquer par un "X" la relation que vous jugez la plus exacte entre les deux tags.
Choisissez une seule relation pour chaque tag. Les deux premières lignes sont des exemples fictifs.

Tag1	Tag2	Si je cherche des informations, je dois pouvoir utiliser indifféremment le Tag1 ou le Tag2 (Tag1 et Tag2 sont équivalents)	Si je cherche des informations liées à Tag1, les informations liées à Tag2 sont pertinentes, mais pas le contraire (Tag1 est plus général que Tag2)	Si je cherche des informations liées à Tag2, les informations liées à Tag1 sont pertinentes, mais pas le contraire (Tag2 est plus général que Tag1)	Si je cherche des informations sur l'un des tags, il est pertinent de suggérer des informations sur l'autre tag (Tag1 et Tag2 liés)	Ces 2 tags ne sont pas spécialement liés
agriculture durable	agriculture raisonnee					
biologie	agriculture biologique					
changements sociaux	changement social					
chimie verte	chanvre					
Climat/changement	changement climatique					
collectivite	action collective					
collectivite	collecte de donnees					
commande	communication entre acteurs					
comportements pro-environnementaux	comportements pro-environnemental					
compost	composant					
conception	ecoconception					
conception	travail collaboratif vis a vis de la conception					
cycle de rankine	cycle organique de rankine					
developpement durable	developpement local					
accumulateurs li-ion	tours d'habitation					
acteurs du territoire	territorialite					
agglomeration	cooperation					
agriculture durable	agriculture biologique					
diversite culturelle	diversite microbienne					
ecologie	ecology					
elements finis	methode des elements finis					
energie	politique energetique					
energie	production energie					
energie	energie renouvelable					
energie	autonomie energetique					
energy	energies					
environmental	environment					
environnement	domaines environnementaux					
environnement	grenelle de l'environnement					
environnement	competences environnementales					
environnement	socialisation aux preoccupations environnementales					
ester	gasteropodes					
experimentation	electromediation					
extraction	phytoextraction					
finance	financement					
gestion	gestion stock					
gestion stock	gestion des ressources naturelles					
gouvernance	gouvernance forestiere					
hybride	vehicules electriques et hybrides					

List of Figures

List of Tables

Bibliography

ABBATTISTA F., GENDARMI D. & LANUBILE F. (2007). Fostering knowledge evolution through community-based participation. In *Workshop on Social and Collaborative Construction of Structured Knowledge(CKC 2007) at WWW 2007*, Banff, Canada.

ABEL F., FRANK M., HENZE N., KRAUSE D., PLAPPERT D. & SIEHNDEL P. (2007). Groupme! - Where Semantic Web meets Web 2.0. In *ISWC/ASWC*, volume 4825 of *LNCS*, p. 871–878: Springer.

AGRAWAL, R. I. T. & SWAMI A. (1993). Mining association rules between sets of items in large databases. In *SIGMOD1993, ACM Press*.

AL-KHALIFA H. S. & DAVIS H. C. (2007). Towards better understanding of folksonomic patterns. In *HT '07: Proceedings of the eighteenth conference on Hypertext and hypermedia*, p. 163–166, New York, NY, USA: ACM.

ALEMAN-MEZA B., BOJARS U., BOLEY H., BRESLIN J. G., MOCHOL M., NIXON L. J. B., POLLERES A. & ZHDANOVA A. V. (2007). Combining rdf vocabularies for expert finding. In E. FRANCONI, M. KIFER & W. MAY, Eds., *ESWC*, volume 4519 of *Lecture Notes in Computer Science*, p. 235–250: Springer.

ANGELETOU S., SABOU M. & MOTTA E. (2008). Semantically Enriching Folksonomies with FLOR. In *CISWeb Workshop at European Semantic Web Conference ESWC.*

AUER S., BIZER C., KOBILAROV G., LEHMANN J., CYGANIAK R. & IVES Z. G. (2007). Dbpedia: A nucleus for a web of open data. In K. ABERER, K.-S. CHOI, N. F. NOY, D. ALLEMANG, K.-I. LEE, L. J. B. NIXON, J. GOLBECK, P. MIKA, D. MAYNARD, R. MIZOGUCHI, G. SCHREIBER & P. CUDRÉ-MAUROUX, Eds., *ISWC/ASWC*, volume 4825 of *Lecture Notes in Computer Science*, p. 722–735: Springer.

AUSSENAC-GILLES N., BIÉBOW B. & SZULMAN S. (2000a). Corpus analysis for conceptual modelling. In *Workshop on Ontologies and Texts at Knowledge Acquisition, Modeling and Management, 12th International Conference, EKAW 2000*.

AUSSENAC-GILLES N., BIEBOW B. & SZULMAN S. (2000b). Revisiting ontology design: A methodology based on corpus analysis. In *Proceedings of Knowledge Acquisition, Modeling and Management, 12th International Conference, EKAW 2000*, p. 172–188, London, UK: Springer-Verlag.

BACH T. L. (2006). *Construction d'un Web Sémantique multi-points de vue*. PhD thesis, Ecole de Mines de Paris à Sophia Antipolis.

BACHIMONT B. (2000). *Ingénierie des connaissances: Evolutions récentes et nouveaux défis*, chapter Engagement sémantique et engagement ontologique: conception et réalisation d'ontologies en ingénierie des connaissances. Eyrolles.

BACHIMONT B. (2005). *Arts et sciences du numérique : Ingénierie des connaissances et critique de la raison computationnelle*. Habilitation à Diriger des Recherches, Université de Technologie de Compiègne.

BEGELMAN G., KELLER P. & SMADJA F. (2006). Automated tag clustering: Improving search and exploration in the tag space. In *Proceedings of the WWW 2006 Workshop on Collaborative Web Tagging*, Edinburgh.

BENERECETTI M., BOUQUET P. & GHIDINI C. (2001). On the dimensions of context dependence: partiality, approximation, and perspective. In V. AKMAN, P. BOUQUET, R. THOMASON & R. YOUNG, Eds., *Proceedings of the Third International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT'01)*, volume 2116 of *LNCS*, p. 59–72: Springer.

BENZ D., HOTHO A. & STUMME G. (2010). Semantics made by you and me: Self-emerging ontologies can capture the diversity of shared knowledge. In *Proceedings of the 2nd Web Science Conference (WebSci10)*, Raleigh, NC, USA.

BERNERS-LEE T., HENDLER J. & LASSILA O. (2001). The Semantic Web. *Scientific American*, **284**(5), 34–44.

BOJARS U., PASSANT A., CYGANIAK R. & BRESLIN J. (2008). Weaving SIOC into the Web of Linked Data. In *Proceedings of the WWW 2008 Workshop Linked Data on the Web (LDOW2008), Beijing, China*.

BOLDI P., SANTINI M. & VIGNA S. (2004). Do your worst to make the best: Paradoxical effects in pagerank incremental computations. In *In Proceedings of the third Workshop on Web Graphs (WAW), volume 3243 of Lecture Notes in Computer Science*, p. 168–180: Springer.

BOTHOREL C. & BOUKLIT M. (2008). An algorithm for detecting communities in folksonomy hypergraphs. In *Proc. of Innovative Internet Computing Systems international conference, June 16-18, Schoelcher, Martinique Sponsored by IEEE*.

J.-F. BOUJUT, Ed. (2005). *Proceedings of the International Workshop on Annotation for Collaboration - Methods, Tools and Practices, La Sorbonne, Paris, France, 2005, November 23-24*. CNRS - Programme société de l'information.

BOUQUET P., GIUNCHIGLIA F., VAN HARMELEN F., SERAFINI L. & STUCKENSCHMIDT H. (2004). Contextualizing ontologies. *Web Semantics: Science, Services and Agents on the World Wide Web*, **1**(4), 325 – 343. International Semantic Web Conference 2003.

BOURIGAULT D. & JACQUEMIN C. (1999). Term extraction + term clustering: an integrated platform for computer-aided terminology. In *Proceedings of the 9th conference on European chapter of the Association for Computational Linguistics*, p. 15–22, Morristown, NJ, USA: Association for Computational Linguistics.

BRAUN S., SCHMIDT A., WALTER A., NAGYPÁL G. & ZACHARIAS V. (2007). Ontology maturing: a collaborative web 2.0 approach to ontology engineering. In *CKC*, volume 273 of *CEUR Workshop Proceedings*: CEUR-WS.org.

BRESLIN J. G., HARTH A., BOJARS U. & DECKER S. (2005). Towards semantically-interlinked online communities. *Lecture Notes in Computer Science : The Semantic Web: Research and Applications*, p. 500–514.

BRICKLEY D. & MILLER L. (2004). *FOAF Vocabulary Specification*. Namespace Document 2 Sept 2004, FOAF Project. http://xmlns.com/foaf/0.1/.

BRIN S. & PAGE L. (1998). The anatomy of a large–scale hypertextual web search engine. *Computer Networks and ISDN Systems*, **30**(1–7), 107–117.

BUFFA M., GANDON F., ERETEO G., SANDER P. & FARON C. (2008). SweetWiki: A semantic Wiki. *J. Web Sem., special issue on Web 2.0 and the Semantic Web*, **6**(1), 84–97.

CAHIER J.-P., ZAHER L., PÉTARD X., LEBOEUF J.-P. & GUITTARD C. (2005). Experimentation of a Socially Constructed "Topic Map" by the OSS Community. *workshop on Knowledge Management and Organizational Memories, IJCAI-05*.

CAHIER J.-P., ZAHER L. & ZACKLAD M. (2007). Information seeking in a "socio-semantic web" application. In *ICPW07: Proceedings of the 2nd international conference on Pragmatic web*, p. 91–95, New York, NY, USA: ACM.

CARROLL J. J., BIZER C., HAYES P. & STICKLER P. (2005). Named graphs, provenance and trust. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, p. 613–622, New York, NY, USA: ACM.

CATTUTO C., BENZ D., HOTHO A. & STUMME G. (2008). Semantic grounding of tag relatedness in social bookmarking systems. In *ISWC '08: Proceedings of the 7th International Conference on The Semantic Web*, p. 615–631, Berlin, Heidelberg: Springer-Verlag.

CILIBRASI R. & VITANYI P. (2006). Similarity of objects and the meaning of words. In *In Proc. 3rd Annual Conferene on Theory and Applications of Models of Computation (TAMC06), volume 3959 of LNCS*, p. 21–45: Springer.

CIMIANO P. (2006). *Ontology Learning and Population from Text: Algorithms, Evaluation and Applications*. Secaucus, NJ, USA: Springer-Verlag New York, Inc.

COHEN W., RAVIKUMAR P. & FIENBERG S. (2003). A Comparison of String Distance Metrics for Name-Matching Tasks. In *Proc. of IJCAI-03 Workshop on Information Integration on the Web (IIWeb-03), August 9-10, 2003, Acapulco, Mexico*.

D'ACQUIN M. (2005). *Un portail sémantique pour la gestion des connaissances en cancérologie*. PhD thesis, Université Henri Poincaré, Nancy.

DELALONDE C. & SOULIER E. (2007). DemonD: Leveraging social participation for Collaborative Information Retrieval. In *1st Workshop on Adaptation and Personalisation in Social Systems: Groups, Teams, Communities, Corfu, Greece*.

DEWEY M. (1876). *A Classification and Subject Index for Cataloguing and Arranging the Books and Pamphlets of a Library*. Amherst, USA.

DIENG R., CORBY O., GANDON F., GIBOIN A., GOLEBIOWSKA J., MATTA N. & RIBIÈRE M. (2005). *Knowledge Management: Méthodes et outils pour la gestion des connaissances*. Dunod.

DOLEZAL F. (2005). *A History of Roget's Thesaurus by Werner Hullen*, volume 32. John Benjamins Publishing Company.

ECHARTE F., ASTRAIN J. J., CORDOBA A. & VILLADANGOS J. E. (2007). Ontology of folksonomy: A new modelling method. In S. HANDSCHUH, N. COLLIER, T. GROZA, R. DIENG, M. SINTEK & A. DE WAARD, Eds., *SAAKM*, volume 289 of *CEUR Workshop Proceedings*: CEUR-WS.org.

ERETEO G., BUFFA M., GANDON F., & CORBY O. (2009a). Analysis of a real online social network using semantic web frameworks. In *Proc. International Semantic Web Conference, ISWC'09, Washington, USA*.

ERETEO G., BUFFA M., GANDON F., LEITZELMAN M. & LIMPENS F. (2009b). Leveraging social data with semantics. In *W3C Workshop on the Future of Social Networking, Barcelona.*

EUZENAT J. & SHVAIKO P. (2007). *Ontology Matching*. Berlin, Heidelberg: Springer.

FALQUET G. & MOTTAZ C.-L. (2002). A model for the collaborative design of multi point-of-view terminological knowledge bases. In *Knowledge Management and Organizational Memories, Kluwer*.

FELLBAUM C. (1998). *WordNet An Electronic Lexical Database*. Cambridge, MA ; London: The MIT Press.

FERNANDEZ-LOPEZ M., GOMEZ-PEREZ A. & JURISTO N. (1997). Methontology: from ontological art towards ontological engineering. In *Proceedings of the AAAI97 Spring Symposium*, p. 33–40, Stanford, USA.

FIRTH J. (1957). A synopsis of linguistic theory 1930-1955. *Studies in linguistic analysis*, p. 1–32.

GANDON F. (2008). *Graphes RDF et leur Manipulation pour la Gestion de Connaissances*. Habilitation à Diriger des Recherches, University of Nice - Sophia Antipolis.

GANDON F., BOTTOLIER V., CORBY O. & DURVILLE P. (2007). Rdf/xml source declaration, w3c member submission. http://www.w3.org/Submission/rdfsource/.

GAVED M., HEATH T. & EISENSTADT M. (2006). Wikis of locality: insights from the open guides. In D. RIEHLE & J. NOBLE, Eds., *Int. Sym. Wikis*, p. 119–126: ACM.

GIANNAKIDOU E., KOUTSONIKOLA V., VAKALI A. & KOMPATSIARIS Y. (2008). Co-clustering tags and social data sources. *Web-Age Information Management, 2008. WAIM '08. The Ninth International Conference on*, p. 317–324.

GIBOIN A., GANDON F., CORBY O. & DIENG R. (2002). Assessment of ontology-based tools: Systemizing the scenario approach,. In J. ANGELE, Ed., *Proceedings of EON2002: Evaluation of Ontology-based Tools Workshop at the 13th International Conference on Knowledge Engineering and Knowledge Management EKAW, Siguenza (Spain)*, p. 63–73: York Sure.

GIBOIN A., LEITZELMAN M., SOULIER E., BUGEAUD F., LEMEUR V., HERLEDAN F. & MERLE A. (2009). *Analyse des pratiques et des exigences pour les utilisateurs d'ISICIL*. Rapport interne, Deliverable ISICIL - ANR-08-CORD-011-05.

GLIGOROV R., VAN OSSENBRUGGEN J., AROYO L., VAN EES A., OOMEN J., BALTHUSSEN L. B. & BRINKERINK M. (2010). Towards integration of end-user tags with professional annotations. In *WebSci10: Extending the Frontiers of Society On-Line*.

GOLDER S. A. & HUBERMAN B. A. (2006). Usage patterns of collaborative tagging systems. *J. Inf. Sci.*, **32**(2), 198–208.

GOLEBIOWSKA J. (2002). *Exploitation des ontologies pour la memoire d'un projet-vehicule - Methode et outil SAMOVAR.* PhD thesis, Universite de Nice-Sophia Antipolis.

GOOD B. M., KAWAS E. & WILKINSON M. (2007). Bridging the gap between social tagging and semantic annotation: E.d. the entity describer. *Nature Precedings*, **945**(2).

GOTOH O. (1981). An improved algorithm for matching biological sequences. *Journal of Molecular Biology*, **162**, 705–708.

GRUBER T. (1993). A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, **5**(2), 199–220.

GRUBER T. (2007). Ontology of folksonomy: A mash-up of apples and oranges. *International Journal on Semantic Web & Information Systems*, **3**(2), 1–11.

GRUBER T. (2008). Collective knowledge systems: Where the Social Web meets the Semantic Web. *J. Web Sem.*, **6**(1), 4–13.

HAGEN L. W. & KAHNG A. B. (1992). New spectral methods for ratio cut partitioning and clustering. *IEEE Trans. on CAD of Integrated Circuits and Systems*, **11**(9), 1074–1085.

HALPIN H. & PRESUTTI V. (2009). An ontology of resources: Solving the identity crisis. In L. AROYO, P. TRAVERSO, F. CIRAVEGNA, P. CIMIANO, T. HEATH, E. HYVONEN, R. MIZOGUCHI, E. OREN, M. SABOU & E. P. B. SIMPERL, Eds., *ESWC*, volume 5554 of *Lecture Notes in Computer Science*, p. 521–534: Springer.

HALPIN H., ROBU V. & SHEPHERD H. (2007). The complex dynamics of collaborative tagging. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, p. 211–220, New York, NY, USA: ACM.

HAUSENBLAS M. & REHATSCHEK H. (2007). mle: Enhancing the Exploration of Mailing List Archives Through Making Semantics Explicit. In *Semantic Web Challenge, ISWC*.

HAYES C., AVESANI P. & BOJARS U. (2007). An analysis of bloggers, topics and tags for a blog recommender system. *From Web to Social Web: Discovering and Deploying User and Content Profiles*, p. 1–20.

HEATH T. & MOTTA E. (2007). Revyu.com: a Reviewing and Rating Site for the Web of Data. In *ISWC/ASWC*, volume 4825 of *LNCS*, p. 895–902: Springer.

HENRI F., CHARLIER B. & LIMPENS F. (2008). Understanding ple as an essential component of the learning process. In *World Conf. on Educational Multimedia, Hypermedia & Telecommunications*.

HENRI F., CHARLIER B. & LIMPENS F. (2009). Understanding and supporting the creation of more effective PLE. In *Int. Conf. on Information Resources Management, Dubai*, Dubai, UAE.

HERTZUM M. & PEJTERSEN A. M. (2000). The information-seeking practices of engineers: Searching for documents as well as for people. *Information Processing and Management*, **36(5)**, 761–778.

HEYMANN P. & GARCIA-MOLINA H. (2006). *Collaborative Creation of Communal Hierarchical Taxonomies in Social Tagging Systems*. Rapport interne, Stanford InfoLab.

HJELMSLEV L. (1963.). *Prolegomena to a theory of language.* University of Wisconsin Press, Madis.

HOTHO A., JÄSCHKE R., SCHMITZ C. & STUMME G. (2006). Information retrieval in folksonomies: Search and ranking. In *The Semantic Web: Research and Applications*, volume 4011 of *Lecture Notes in Computer Science*, p. 411–426, Heidelberg: Springer.

HUYNH-KIM BANG B., DANÉ E. & GRANDBASTIEN M. (2008). Merging semantic and participative approaches for organising teachers' documents. In *Proceedings of ED-Media 08 ED-MEDIA 08 - World Conference on Educational Multimedia, Hypermedia & Telecommunications*, p. p. 4959–4966, Vienna France.

JACCARD P. (1912). The distribution of the flora in the alpine zone. *New Phytologist*, **11**(2), 37–50.

JARO M. A. (1989). Advances in record linking methodology as applied to the 1985 census of tampa florida. *Journal of the American Statistical Society*, **64**, 1183–1210.

JÄSCHKE R., HOTHO A., SCHMITZ C., GANTER B. & STUMME G. (2008). Discovering Shared Conceptualizations in Folksonomies. *J. Web Sem.*, **6**(1), 38–53.

JIANG J. & CONRATH D. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In *Proc. of the Int'l. Conf. on Research in Computational Linguistics*, p. 19–33.

KAHAN J., KOIVUNEN, PRUD'HOMMEAUX E. & SWICK R. R. (2002). Annotea: an open rdf infrastructure for shared web annotations. *Computer Networks*, **39**(5), 589–608.

KIM H. L., PASSANT A., BRESLIN J. G., SCERRI S. & DECKER S. (2008a). Review and alignment of tag ontologies for semantically-linked data in collaborative tagging spaces. In *ICSC '08: Proceedings of the 2008 IEEE International Conference on Semantic Computing*, p. 315–322, Washington, DC, USA: IEEE Computer Society.

KIM H.-L., SCERRI S., BRESLIN J., DECKER S. & KIM H.-G. (2008b). The state of the art in tag ontologies: A semantic model for tagging and folksonomies. In *International Conference on Dublin Core and Metadata Applications*, Berlin, Germany.

KIM H.-L., YANG S.-K., SONG S.-J., BRESLIN J. G. & KIM H.-G. (2007). Tag Mediated Society with SCOT Ontology. In *Semantic Web Challenge, ISWC*.

KIPP M. E. (2008). @toread and cool : Subjective, affective and associative factors in tagging. In *Proceedings Canadian Association for Information Science/L'Association canadienne des sciences de l'information (CAIS/ACSI)*.

KISS G., ARMSTRONG C., MILROY R. & PIPER J. (1973). *The Computer and Literary Studies*, chapter An associative thesaurus of English and its computer analysis., p. 1–1. University Press, Edinburgh.

KOERNER C., BENZ D., STROHMAIER M., HOTHO A. & STUMME G. (2010). Stop thinking, start tagging - tag semantics emerge from collaborative verbosity. In *Proceedings of the 19th International World Wide Web Conference (WWW 2010)*, Raleigh, NC, USA: ACM. (to appear).

KOIVUNEN M. R. (2006). Annotea and semantic web supported collaboration. http://www.annotea.org/eswc2005/01_koivunen_final.pdf.

KOZAKI K., KITAMURA Y., IKEDA M. & MIZOGUCHI R. (2002). Hozo: An environment for building/using ontologies based on a fundamental consideration of role and relationship. In A. GOMEZ-PEREZ & V. BENJAMINS, Eds., *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, volume 2473 of *Lecture Notes in Computer Science*, p. 155–163. Springer Berlin / Heidelberg.

LANIADO D., EYNARD D. & COLOMBETTI M. (2007). Using wordnet to turn a folksonomy into a hierarchy of concepts. In *Semantic Web Application and Perspectives - Fourth Italian Semantic Web Workshop*, p. 192–201.

LAVE J. & WENGER E. (1991). *Situated Learning: Legitimate Peripheral Participation*. Cambridge, UK: Cambridge University Press.

LEVENSHTEIN V. I. (1966). Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady.*, **10**(8), 707–710.

LIMPENS F., GANDON F. & BUFFA M. (2008a). Bridging Ontologies and Folksonomies to Leverage Knowledge Sharing on the Social Web: a Brief Survey. In *Proc. 1st International Workshop on Social Software Engineering and Applications (SoSEA)*, L'Aquila, Italy.

LIMPENS F., GANDON F. & BUFFA M. (2008b). Rapprocher les ontologies et les folksonomies pour la gestion des connaissances partagées : un Etat de l'art. In *Proc. 19èmes journées francophones d'Ingénierie des Connaissances, Nancy*, Loria, Nancy, France.

LIMPENS F., GANDON F. & BUFFA M. (2009a). Collaborative semantic structuring of folksonomies (short article). In *IEEE/WIC/ACM Int. Conf. on Web Intelligence*, Milano, Italia.

LIMPENS F., GANDON F. & BUFFA M. (2009b). Sémantique des folksonomies: structuration collaborative et assistée. In *Proc. Ingénierie des Connaissances IC'09*, Hammamet, Tunisia.

LIMPENS F., GANDON F. & BUFFA M. (2010). Helping online communities to semantically enrich folksonomies. In *Proceedings of the WebSci10: Extending the Frontiers of Society On-Line*, Raleigh, NC, USA: http://webscience.org.

LIMPENS F., MONNIN A., LANIADO D. & GANDON F. (2009c). Nicetag ontology: tags as named graphs. In *International Workshop in Social Networks Interoperability, Asian Semantic Web Conference 2009*.

LIN H. & DAVIS J. (2010). Computational and crowdsourcing methods for extracting ontological structure from folksonomy. In L. AROYO, G. ANTONIOU, E. HYVONEN, A. TEN TEIJE, H. STUCKENSCHMIDT, L. CABRAL & T. TUDORACHE, Eds., *ESWC (2)*, volume 6089 of *Lecture Notes in Computer Science*, p. 472–477: Springer.

LIN H., DAVIS J. & ZHOU Y. (2009). An integrated approach to extracting ontological structures from folksonomies. In *6th Annual European Semantic Web Conference (ESWC2009)*, p. 654–668.

MARKINES B., CATTUTO C., MENCZER F., BENZ D., HOTHO A. & STUMME G. (2009). Evaluating similarity measures for emergent semantics of social tagging. In *18th International World Wide Web Conference*, p. 641–641.

MATHES A. (2004). *Folksonomies - Cooperative Classification and Communication Through Shared Metadata*. Rapport interne, GSLIS, Univ. Illinois Urbana-Champaign.

MAZHOUD O., PASCUAL E. & VIRBEL J. (1995). Représentation et gestion d'annotations. In *3ème Conférence Hypertextes et Hypermédias, -, -*, p. 127–138, Paris: Hermes.

MAZHOUD O., PASCUAL E. & VIRBEL J. (1996). Annoting as a Document Management Tool. In *ALLC ACH'96, Bergen, -*, p. 199–201: Norwegian Computing Centre for the Humanities.

MIKA P. (2005). Ontologies are Us: a Unified Model of Social Networks and Semantics. In *ISWC*, volume 3729 of *LNCS*, p. 522–536: Springer.

MILLER G. A., BECKWITH R., FELLBAUM C., GROSS D. & MILLER K. J. (1990). Introduction to wordnet: An on-line lexical database. *Journal of Lexicography*, **3**(4), 235–244.

MONGE A. E. & ELKAN C. P. (1996). The field matching problem: Algorithms and applications. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, p. 267–270.

MONNIN A. (2009). Tags and folksonomies as artifacts of meaning. *Philosophy of Engineering and Artifact in the Digital Age. A First Eastern European (Romanian) Perspective, Cambridge Scholar Publishing*.

MONNIN A., LIMPENS F., GANDON F. & LANIADO D. (2010). Speech acts meet tagging: Nicetag ontology. In *I-SEMANTICS '10: Proceedings of the 6th International Conference on Semantic Systems*, p. 1–10, New York, NY, USA: ACM.

NEWMAN M. E. J. & GIRVAN M. (2004). Finding and evaluating community structure in networks. *Physical Review*, **E 69**(026113).

NEWMAN R., AYERS D. & RUSSELL S. (2005). Tag Ontology Design. http://www.holygoat.co.uk/owl/redwood/0.1/tags/.

PARK J. & HUNTING S. (2002). *XML Topic Maps: Creating and Using Topic Maps for the Web*. Addison-Wesley Professional.

PASSANT A. (2007). Using Ontologies to Strengthen Folksonomies and Enrich Information Retrieval in Weblogs. In *International Conference on Weblogs and Social Media*.

PASSANT A. (2009). *Technologies du Web Sémantique pour l'Entreprise 2.0*. PhD thesis, Université Paris IV - Sorbonne.

PASSANT A. & LAUBLET P. (2008). 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 (LDOW2008), Beijing, China*.

PERON S. (2009). *Etude ergonomique de Folkon*. Rapport interne, UNSA, INRIA.

POTHEN A., SIMON H. D. & LIOU K.-P. (1990). Partitioning sparse matrices with eigenvectors of graphs. *SIAM J. Matrix Anal. Appl.*, **11**(3), 430–452.

RASTIER F. (1994). *Sémantique pour l'analyse*, chapter Interprétation et compréhension, p. 1–22. Masson, Paris.

REINACH A. (1983). *The A Priori Foundations of Civil Law (trans. John Crosby)*. Aletheia.

RIBIÈRE M. (1999). *Représentation et gestion de multiples points de vue dans le formalisme des graphes conceptuels*. PhD thesis, Université Nice-Sophia Antipolis.

RONZANO F., MARCHETTI A. & TESCONI M. (2008). Tagpedia: a semantic reference to describe and search for web resources. In *WWW 2008 Workshop on Social Web and Knowledge Management, Beijing, China*.

ROSE K. (1998). Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. In *Proceedings of the IEEE*, p. 2210–2239.

SANDERSON M. & CROFT B. (1999). Deriving concept hierarchies from text. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, p. 206–213, New York, NY, USA: ACM.

SAUSSURE F. D. (1916). *Cours de linguistique générale*. Paris: Bayot.

SCHMIDT M., MEIER M. & LAUSEN G. (2010). Foundations of sparql query optimization. In *ICDT '10: Proceedings of the 13th International Conference on Database Theory*, p. 4–33, New York, NY, USA: ACM.

SCHMITZ C., HOTHO A., JÄSCHKE R. & STUMME G. (2006). Mining Association Rules in Folksonomies. *Data Science and Classification*, p. 261–270.

SCHMITZ P. (2006). Inducing ontology from flickr tags. In *Proc. of the Collaborative Web Tagging Workshop (WWW06)*.

SCHWARZKOPF E., HECKMANN D., DENGLER D. & KRONER A. (2007). Mining the structure of tag spaces for user modeling. *Proceedings of the Workshop on Data Mining for User Modeling at the 11th International Conference on User Modeling*, p. 63–75.

SEGUELA P. & AUSSENAC-GILLES N. (1999). Extraction de relations sémantiques entre termes et enrichissement de modèles du domaine. In *Ingénierie des Connaissances*, p. pp. 79–98.

SEN S., LAM S. K., RASHID A. M., COSLEY D., FRANKOWSKI D., OSTERHOUSE J., HARPER F. M. & RIEDL J. (2006). tagging, communities, vocabulary, evolution. In *CSCW '06: Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*, p. 181–190, New York, NY, USA: ACM.

SERVANT F.-P. (2006). Semanlink. In *Jena User Conference (JUC)*.

SINHA R. (2005). A cognitive analysis of tagging. http://www.rashmisinha.com/archives/05_09/tagging-cognitive.html.

SINHA R. (2006). Tagging from Personnal to Social : Observations and Design Principles. In *Tagging Workshop, at International Conference on World Wide Web*.

SIORPAES K. & HEPP M. (2008). Ontogame: Weaving the semantic web by online gaming. In M. HAUSWIRTH, M. KOUBARAKIS & S. BECHHOFER, Eds., *Proceedings of the 5th European Semantic Web Conference*, LNCS, Berlin, Heidelberg: Springer Verlag.

SMITH T. F. & WATERMAN M. S. (1981). Identification of common molecular subsequences. *J Mol Biol*, **147**(1), 195–197.

SOWA J. (1984). *Conceptual Structures: Information Processing in Mind and Machine*. The Systems Programming Series. Addison-Wesley.

SPECIA L. & MOTTA E. (2007). Integrating folksonomies with the semantic web. In *Proc. of the European Semantic Web Conference (ESWC2007)*, volume 4519 of *LNCS*, p. 624–639, Berlin Heidelberg, Germany: Springer-Verlag.

SUNAGAWA E., KOZAKI K., KITAMURA Y. & MIZOGUCHI R. (2003). An environment for distributed ontology development based on dependency management. In D. FENSEL, K. P. SYCARA & J. MYLOPOULOS, Eds., *International Semantic Web Conference*, volume 2870 of *Lecture Notes in Computer Science*, p. 453–468: Springer.

TANASESCU V. & STREIBEL O. (2007). Extreme tagging: Emergent semantics through the tagging of tags. In P. HAASE, A. HOTHO, L. CHEN, E. ONG & P. C. MAUROUX, Eds., *Proceedings of the International Workshop on Emergent Semantics and Ontology Evolution (ESOE2007) at ISWC/ASWC2007, Busan, South Korea*.

TARDINI S. & CANTONI L. (2005). A semiotic approach to online communities: Belonging, interest and identity in websites' and videogames' communities. In *IADIS Intl. Conf. e-Society*, p. 371–378: IADIS.

TESCONI M., RONZANO F., MARCHETTI A. & MINUTOLI S. (2008). Semantify del.icio.us: Automatically turn your tags into senses. In *Proceedings of the First Social Data on the Web Workshop (SDoW2008)*.

TORNIAI C., JOVANOVIC J., BATEMAN S., GASEVIC D. & HATALA M. (2008). Leveraging folksonomies for ontology evolution in e-learning environments. In *ICSC '08: Proceedings of the 2008 IEEE International Conference on Semantic Computing*, p. 206–213, Washington, DC, USA: IEEE Computer Society.

TUMMARELLO G., MORBIDONI C. & NUCCI M. (2006). Enabling semantic web communities with dbin: An overview. In I. F. CRUZ, S. DECKER, D. ALLEMANG, C. PREIST, D. SCHWABE, P. MIKA, M. USCHOLD & L. AROYO, Eds., *International Semantic Web Conference*, volume 4273 of *Lecture Notes in Computer Science*, p. 943–950: Springer.

VAAST E., BOLAND R., DAVIDSON E., PAWLOWSKI S. & SCHULTZE U. (2006). Investigating the "knowledge" in knowledge management: A social representations perspective. *Communications of the Association for Information Systems*, **vol 17**, 314–340.

VAN DAMME C., HEPP M. & SIORPAES K. (2007). Folksontology: An integrated approach for turning folksonomies into ontologies. In *Bridging the Gep between Semantic Web and Web 2.0 (SemNet 2007)*, p. 57–70.

VANDERWAL T. (2004). Folksonomy Coinage and Definition. http://www.vanderwal.net/folksonomy.html.

VERES C. (2006). The language of folksonomies: What tags reveal about user classification. In *Natural Language Processing and Information Systems*, volume 3999/2006 of *Lecture Notes in Computer Science*, p. 58–69, Berlin / Heidelberg: Springer.

VON AHN L. & DABBISH L. (2008). Designing games with a purpose. *Communications of the ACM*, **51**(8), 58–67.

WELLER K. & PETERS I. (2008). Seeding, weeding, fertilizing. different tag gardening activities for folksonomy maintenance and enrichment. In S. AUER, S. SCHAFFERT & T. PELLEGRINI, Eds., *Proceedings of I-Semantics08, International Conference on Semantic Systems. Graz, Austria, September 3-5*, p. 10–117.

WENGER E., MCDERMOTT R. & SNYDER W. M. (2002). *Cultivating Communities of Practice - A guide to managing knowledge*. Boston, MA: Harvard Business School Press.

WILLE R. (1982). *Ordered Sets*, chapter Restructuring lattices theory : An approach based on hierarchies of concepts, p. 445–470. Reidel, Dordrecht-Boston.

WINKLER W. E. (1999). *The State of Record Linkage and Current Research Problems*. Rapport interne, Statistical Research Division, U.S. Census Bureau.

WOLFF C., HECKNER M. & MUHLBACHER S. (2008). Tagging tagging. analysing user keywords in scientific bibliography management systems. *Journal of Digital Informaton*, **9**(27).

ZACKLAD M., BÉNEL A., CAHIER J., ZAHER L., LEJEUNE C. & ZHOU C. (2007). Hypertopic : une Métasémiotique et un Protocole pour le Web Socio-Sémantique. In *IC*, p. 217–228: Cépaduès. ISBN 978-2-85428-790-9.

ZHOU M., BAO S., WU X. & YU Y. (2007). An unsupervised model for exploring hierarchical semantics from social annotations. In K. ABERER, K.-S. CHOI, N. NOY, D. ALLEMANG, K.-I. LEE, L. J. B. NIXON, J. GOLBECK, P. MIKA, D. MAYNARD, G. SCHREIBER & P. CUDRÉ-MAUROUX, Eds., *Proceedings of the 6th International Semantic Web Conference and 2nd Asian Semantic Web Conference (ISWC/ASWC2007), Busan, South Korea*, volume 4825 of *LNCS*, p. 673–686, Berlin, Heidelberg: Springer Verlag.

Multi-points of view semantic enrichment of folksonomies

This thesis, set at the crossroads of Social Web and Semantic Web, is an attempt to bridge Social tagging-based systems with structured representations such as thesauri or ontologies (in the informatics sense). Folksonomies resulting from the use of social tagging systems suffer from a lack of precision that hinders their potentials to retrieve or exchange information. This thesis proposes supporting the use of folksonomies with formal languages and ontologies from the Semantic Web. Automatic processing of tags allows bootstraping the process by using a combination of a custom method analyzing tags' labels and adapted methods analyzing the structure of folksonomies. The contributions of users are described thanks to our model SRTag, which allows supporting diverging points of view, and captured thanks to our user friendly interface allowing the users to structure tags while searching the folksonomy. Conflicts between individual points of view are detected, solved, and then exploited to help a referent user maintain a global and coherent structuring of the folksonomy, which is in return used to garanty the coherence while enriching individual contributions with the others' contributions. The result of our method allows enhancing the navigation within tag-based knowledge systems, but can also serve as a basis for building thesauri fed by a truly bottom up process.

Keywords

Social tagging, Folksonomies, Ontologies, Thesauri, Social Web, Semantic Web

Enrichissement sémantique multi-points de vue de folksonomies

Cette thèse, au croisement du Web Social et du Web Sémantique, vise à rapprocher folksonomies et représentations structurées de connaissances telles que les thesauri ou les ontologies informatiques. Les folksonomies, résultant de l'usage de plateformes de *social tagging,* souffrent d'un manque de précision qui les rend difficile à exploiter pour la navigation. Cette thèse présente notre approche multi-points de vue de l'enrichissement sémantique des folksonomies. L'amorçage est assuré par des traitements automatiques qui permettent d'extraire des relations sémantiques entre tags grâce à la combinaison d'une méthode que nous avons mise au point et analysant les labels de tags, et de méthodes que nous avons adaptées et analysant la structure de folksonomies. Les contributions des utilisateurs sont décrites par notre modèle SRTag supportant les points de vue divergents, et capturées par une interface intégrant à la navigation des fonctionnalités de micro-édition de folksonomie. Les conflits entre points de vue sont détectés et solutionnés par un agent automatique dont les résultats sont ensuite exploités pour aider un utilisateur référent à maintenir une structuration globale et cohérente de la folksonomie, servant en retour pour enrichir chaque point de vue individuel avec les autres contributions tout en garantissant une cohérence locale. Notre méthode permet d'améliorer la navigation dans les systèmes de connaissances à base de tags, mais fournit aussi une base à des thesauri nourris par un processus *bottom-up* d'acquisition de connaissances.

Mot-clés

Social tagging, Folksonomies, Ontologies, Thesauri, Web Social, Web Sémantique