Agent-mediated shared conceptualizations in tagging services

Gonzalo A. Aranda-Corral · Joaquín Borrego-Díaz · Jesús Giráldez-Cru

Abstract Some of the most remarkable innovative technologies from the Web 2.0 are the collaborative tagging systems. They allow the use of folksonomies as a useful structure for a number of tasks in the social web, such as navigation and knowledge organization. One of the main deficiencies comes from the tagging behaviour of different users which causes semantic heterogeneity in tagging. As a consequence a user cannot benefit from the adequate tagging of others. In order to solve the problem, an agent-based reconciliation knowledge system, based on Formal Concept Analysis, is applied to facilitate the semantic interoperability between personomies. This article describes experiments that focus on conceptual structures produced by the system when it is applied to a collaborative tagging service, Delicious. Results will show the prevalence of shared tags in the sharing of common resources in the reconciliation process.

Keywords Knowledge reconciliation · Multiagentsystems · Formal concept analysis

G. A. Aranda-Corral (✉)
Department of Information Technology, Universidad de Huelva, Crta. Palos de La Frontera
s/n. 21819, Palos de La Frontera, Huelva, Spain
e-mail: gonzalo.aranda@dti.uhu.es

J. Borrego-Díaz
Department of Computer Science and Artificial Intelligence, Universidad de Sevilla,
Avda. Reina Mercedes s/n. 41012, Sevilla, Spain
e-mail: jborrego@us.es

J. Giráldez-Cru
Artificial Intelligence Research Institute (IIIA-CSIC), Campus Universidad Autónoma
de Barcelona, 08193 Bellaterra, Spain
e-mail: jgiraldez@iiia.csic.es

1 Introduction

The availability of powerful technologies for sharing information among users (social network members) empowers the organization of social resources. Among them, collaborative tagging represents a very useful process for users which aim to add metadata to documents, objects or, even, urls. Particularly impressive is the success of the Delicious bookmarking service (http://www.delicious.com/) which is one of the most popular social tagging platforms for storing, sharing, and discovering bookmarks. The objects which are tagged and stored in the service are urls, thus the content can be text, image, video or any web entity. Other services that provide tagging are more specific (for example Flickr). In both cases (generic and specific tagging systems) the systems are all very similar. It allows the user to add a resource to the system, and to assign it arbitrary tags to it, although it can diverge from professional (standard) vocabularies [11]. The collection of all the user's assignments constitutes his personomy, the collection of all personomies constitutes the folksonomy [16]. Folksonomies are bottom-up semantic structures which represent a social alternative to ontologies. However their semantic, evolution and formal properties are hard to study. In impressive services such as Delicious or Flickr, folksnomies can be considered as a structure that emerges from the complex system that social networks represent.

Tagging is a social method to categorize or to classify documents, whose success in the Web 2.0 lies in the fact that there are not limitations for its personal use. Tagging is a task that different websites consider in many different ways, mainly [25]: for managing personal information (navigating), as social bookmarking to collect and to share digital objects, or for improving the e-commerce experience. The user can explore his personomy, as well as the personomies of the other users, in all dimensions: for a given user one can see all the resources he has uploaded, together with the tags he has assigned to them [17]. However, since tags can be the solution to different users' needs (a fact that humans tend to recognize), automated solutions that utilize social information often tend to neglect the fact that each bookmark may imprint a unique (complex) entity on its own which if considered could contribute to improving those specific tasks. Nevertheless, by increasing the number of users who tag a resource, a core set of tags that outline the object emerges [14] and empowers, for example, recommendation systems [7].

The arbitrary assignment of tags represent a major problem when applied to photos or videos. Description and content are represented in the same tag set. Thus the mix of these kinds of semantically different tags cannot be separated by individual tag frequency information [21] nor topic description (if it is not previously stated). In the case of Delicious (and due to their advanced social features) the problem can be partially solved by the collective intelligence, while in Flickr semantic technologies are introduced to discern those features. In SinNet [4] a mobile Web 2.0 platform for the dissemination of images (and short videos), an intelligent interface guides the user to select tags about the content of the image.

1.1 Motivation: the problem of semantic heterogeneity

As with other social behaviours, tagging shows advantages but also deficiencies, e.g. semantic heterogeneity. Projects like *Faviki* (http://www.faviki.com) or CommonTag (http://commontag.org) attempt to solve these deficiencies. Tagging provides a manner of weak organization of information that, although useful, is mediated by the individual user's behaviour. Within the network, and also based on user preferences, different tagging behaviours exist that actually obstruct automated interoperability. Although solutions exist that assist users' folksonomy (tag clouds, tools based on related tag ideas, collective intelligence methods, data mining, etc.), personal organization of information leads to implicit logical conditions that often differ from the global interpretation of these conditions.

As is argued in [12], tagging is essentially about sensemaking, a process where information is categorized, labeled and, most importantly, through which meaning emerges [26]. Even in a personal tagging structure, concept boundaries and categories are vague, so some items can be doubtfully labeled. Furthermore, users also use tagging task for their own benefit, but nevertheless they contribute usefully to the public good [12]. Therefore, it seems it can be interesting to apply concept mining technologies to facilitate semantic interoperability among different tagging sets. Interest in semantical reconciliation for different tag's sets lies in the fact that, since users' tagging reflects their own set of concepts about documents, the tag-driven navigation among different resources could be insufficient due to semantic heterogeneity.

1.1.1 Navigation between personomies

In order to show semantic heterogeneity, Formal Concept Analysis (FCA) [10] could be a sound tool. FCA is a mathematical theory that, when applied to tagging systems, results in explicit sets of concepts that users manage by tagging, thereby organizing information into structured relationships. FCA can be considered has as a Knowledge Discovering tool, and it can applied to tagging systems (see Section 2 below).

Semantic heterogeneity obstructs the interoperability at the semantic level, thus the solution for this problem is essential to designing intelligent tools for knowledge retrieval and discovering in both, societies and personal knowledge management [18]. Two perspectives can be considered at this point. The first one deals with global interoperability (for example, in companies) and the second one focuses its goal on facilitating the reuse of information not compilled by the user. For example, relating the personal personomy with those of others. The first perspective requires, for a semantic solution, of a global treatment of data, where users, tags and objects are interrelated [16], while the second one focuses on the problem of discovering conceptualizations hidden in user personomy and then to semantically exploiting other personomies (reconciliation tags and concepts) (see Fig. 1 which describes a *conceptual browser* for navigating in Delicious bookmarks, by bridging users' bookmarks sets using reconciled conceptualizations). In order to ensure an efficient use of another user's tag sets, some thought must be given to tags in order to achieve some consensus between both users, which allows us to navigate between different conceptual structures. Relating two personomies is a particular case of knowledge reconciliation, a challenge, which in other fields such as Business Intelligence and Knowledge management is solved by different technologies, including databases and semantic tools (see e.g. [23]). Users' reuse of other personomies by means of knowledge reconciliation is a local technique which avoids the challenge of obtaining a consensual semantic structure from the folksonomy [8, 17], more oriented to client-server structure.

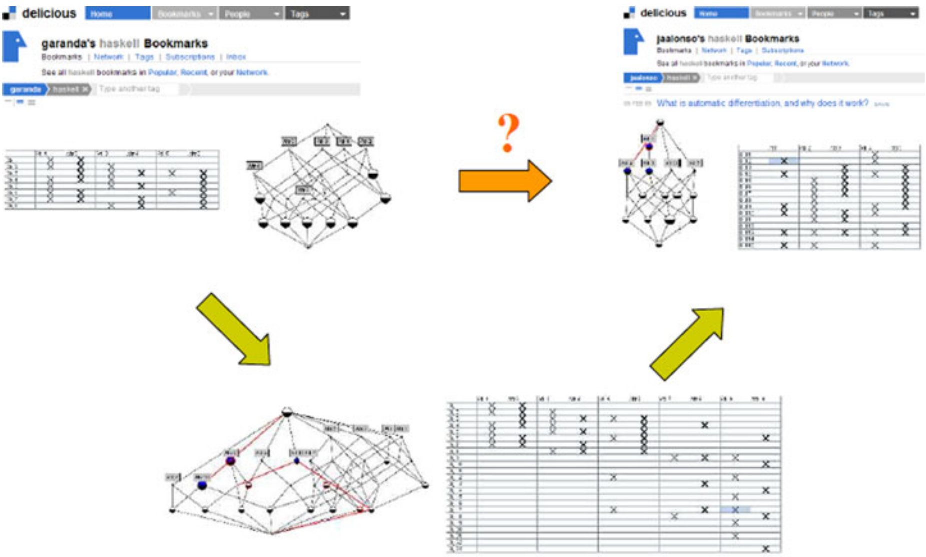

delicious
Home
Bookmarks
People
Tags
garanda's haskell Bookmarks
Bookmarks | Network | Tags | Subscriptions | Inbox
See all haskell bookmarks in Popular, Recent, or your Network.
garanda
haskell
Type another tag
?
jaalonso's haskell Bookmarks
Bookmarks | Network | Tags | Subscriptions
See all haskell bookmarks in Popular, Recent, or your Network.
jaalonso
What is automatic differentiation, and why does it work?

Fig. 1 Navigation between personomies in Delicious based on FCA

In this scenario, it could be very important to attempt to delegate these tasks to intelligent agents in order to provide to the user with an intelligent tool for reuse of personomies. In [2], an agent-based knowledge reconciliation method is presented, which was implemented in the SinNet social mobile platform [4]. In this case the number of executions of the system is relatively low because it was limited to a relatively small number of friends in the mobile network. Nevertheless, in collaborative tagging systems every user has a potential useful personomy. Thus the solution has to be scalable and distributed, therefore going from the level of agent to a Multiagent System (MAS). We describe in this paper the MAS as well as investigate some insights on its application.

1.2 Contribution and organization of the paper

The aim of the paper is to show how a Multiagent System (MAS) can be applied to shape the complexity of users' conceptual structures into a social bookmarking service by comparing the resource-sharing relationship among users against the tag-sharing relationship between users. The first relationship comprises a complex network where semantic similarities could be weak, whilst it expects that the second allows us some understanding about semantic interoperability based on tags and achieved by reconciliation. The paper aims to shown the prevalence of semantic similarity (knowledge reconciliation) in the tag-sharing relationship.

This paper is organized as follows. Section 2 is devoted to the introduction of FCA. Section 3 reviews original agent-based reconciliation, which is applied in this paper. Section 4 provides a specific implementation of knowledge reconciliation. Section 5 describes the relational structure of tagging in Delicious. Section 6 present the experiments and some results. In Section 7 discusses some related work. Section 8 comments on conclusions and future work.

This article extends our previously published works [2, 3], in the following way: it explains the design rationale that leads to both the reconciliation algorithm and the agent-based system,as well as adding some examples and conclusions about the experiments are added. With respect to [2], it applies the system to realistic datasets extracted from a current social tagging service, where specific features cause some reconsideration of previous statements. It also contains a related work section.

2 Background: formal concept analysis

Convergence between the Social Web and the Semantic Web depends on the specific management of ontologies and similar knowledge organization tools. For example, Ontologies and tags/folksonomies must be reconciled in these kinds of projects. A useful bridge between these two kinds of knowledge representation could be *Formal Concept Analysis* [10], which provides semantic features for folksonomies.

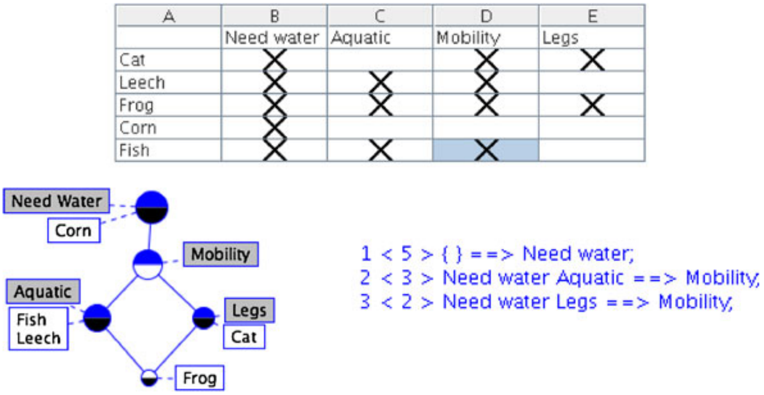
According to R. Wille, FCA [10] mathematizes the philosophical understanding of a concept as a unit of thoughts composed of two parts: the extent and the intent. The extent covers all objects belonging to this concept, while the intent comprises all common attributes valid for all the objects under consideration. It also allows the computation of concept hierarchies from data tables. In this section, we succinctly present basic FCA elements (the main reference is [10]).

The procedure to transform data into structured information, by means of FCA, starts from an entity called *Formal Context*. A formal context $M = (O, A, I)$ consists of two sets, O (objects) and A (attributes) and a relation $I \subseteq O \times A$. Finite contexts can be represented by a 1–0–table (identifying I with a boolean function on $O \times A$). See Fig. 2 for an example of formal context about living beings.

The FCA main goal is the computation of the concept lattice associated with the context. Given $X \subseteq O$ and $Y \subseteq A$ it defines

$$X' := \{a \in A \mid oIa \text{ for all } o \in X\} \text{ and } Y' := \{o \in O \mid oIa \text{ for all } a \in Y\}$$

A (formal) concept is a pair (X, Y) such that $X' = Y$ and $Y' = X$.



A	B	C	D	E
	Need water	Aquatic	Mobility	Legs
Cat	X		X	X
Leech	X	X	X	
Frog	X	X	X	X
Corn	X			
Fish	X	X	X	

1 < 5 > { } ==> Need water;
2 < 3 > Need water Aquatic ==> Mobility;
3 < 2 > Need water Legs ==> Mobility;

Fig. 2 Formal context and associated concept lattice and Stem Basis

If we define the "subconcept relation" between two concepts, $(O_1, A_1) \subseteq (O_2, A_2)$, as $O_1 \subseteq O_2$, a hierarchy among concepts can be obtained and represented (in fact it is a lattice).

For example, concepts from formal context about living beings (Fig. 2, left) are depicted in Fig. 2, right. Actually in Fig. 2, each node is a concept, and its intension (or extension) can be formed by the set of attributes (or objects) included along the path to the top (or bottom). E.g. The node tagged with the attribute Legs represents the concept $(\{Legs, Mobility, NeedWater\}, \{Cat, Frog\})$. Roughly speaking, the Concept Lattice contains every concept which can be extracted from context, and concepts are defined but it is possible there doesn't exist an specific term (word) to denote it.

2.1 Implication and association rules between attributes

Logical expressions in FCA are *implications between attributes*. An implication is a pair of attritute sets, written as $Y_1 \to Y_2$, which is true with respect to $M = (O, A, I)$ according to the following definition. A subset $T \subseteq A$ *respects* $Y_1 \to Y_2$ if $Y_1 \not\subseteq T$ or $Y_2 \subseteq T$. It says that $Y_1 \to Y_2$ holds in M ($M \models Y_1 \to Y_2$) if for all $o \in O$, the set $\{o\}'$ respects $Y_1 \to Y_2$. In that case, it is said that $Y_1 \to Y_2$ is *an implication* of M.

Definition 1 Let $\mathcal{L}$ be a set of implications and L be an implication.

1. L follows from $\mathcal{L}$ ($\mathcal{L} \models L$) if each subset of A respecting $\mathcal{L}$ also respects L.
2. $\mathcal{L}$ is complete if every implication of the context follows from $\mathcal{L}$.
3. $\mathcal{L}$ is non-redundant if for each $L \in \mathcal{L}$, $\mathcal{L} \setminus \{L\} \not\models L$.
4. $\mathcal{L}$ is a (implication) basis for M if $\mathcal{L}$ is complete and non-redundant.

It can obtain a basis from the *pseudo-intents* [13] called *Stem Basis* (SB). A SB for the formal context of living beings is provided in Fig. 2 (down). It is important to remark that SB is only an example of a basis for a formal context. In this paper no specific property of the SB is used, so it can be replaced by any implication basis.

In order to work with formal contexts, stem basis, the Conexp[1] software has been selected. It is used as a library to build the module which provides the implications (and association rules) to the reasoning module of our system. The reasoning module is a production system (designed for SinNet platform [2, 4]). Initially it works with SB, and entailment is based on the following result:

Theorem 1 *Let $\mathcal{S}$ be a basis for M and $\{A_1, \ldots, A_n\} \cup Y \subseteq A$. The following conditions are equivalent:*

1. $\mathcal{S} \cup \{A_1, \ldots A_n\} \vdash_p Y$ *($\vdash_p$ is the entailment with the production system).*
2. $M \models \{A_1, \ldots A_n\} \to Y$.

The support of a rule is defined as the number of objects that contain all attributes Y_1 and they hold the implication as well. Based on this property, a variant of

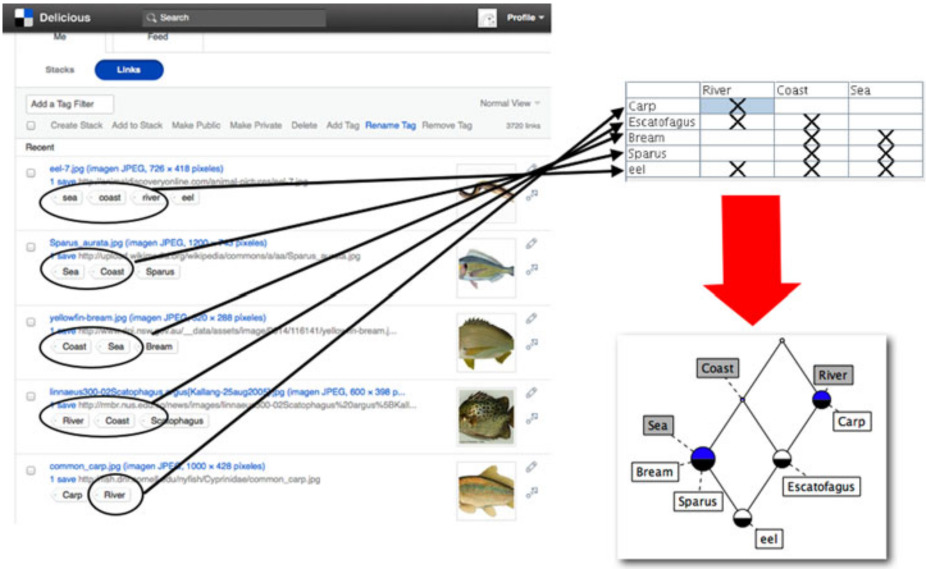
[1] http://sourceforge.net/projects/conexp/

implicational basis is defined, called Stem Kernel basis (SKB), composed by SB's subset where support of each rule is greater than zero.

To illustrate these three entities—formal context, concept lattice, and Stem Basis—an example based on a living being is depicted in Fig. 2, up, left, and right, respectively.

2.2 Tagging, contexts and concepts

The description of FCA (Section 2) should be adapted accordingly to the environment (tags, users, agents). The general approach is to use triadic concept analysis when global shared conceptualization is the goal (see [17]). In the case of personomy analysis, it is sufficient to identify objects (in FCA sense) with resources and tags with attributes. This way FCA directly provides the hidden conceptual structure in the personomy. In Fig. 3 a piece of a Delicious account is interpreted as a formal context by selecting some tags as attributes (in this case, tags that describe the aquatic ecosystem of the fish depicted in the image, that is, attributes are understood as "live in") are depicted in Fig. 2, right. The concept lattice computed from formal context represents the conceptual structure hidden in this personomy. Actually in Fig. 3, each node is a concept, and its intension (or extension) can be formed by the set of attributes (or objects) included along the path to the top (or bottom). For example, the node tagged with the attribute *Sea* represents the concept ({*Bream*, *Sparus*, *eel*}, {*Sea*, *Coast*}). FCA does not provide a term for specifying the concept (in this case, saltwater fishes). For example, bottom concept ({*eel*}, {*Coast*, *Sea*, *River*}) is the concept *euryhaline fish*. Roughly speaking, the concept lattice contains every concept which can be extracted from context, but it is possible there doesn't an specific term (word) to denote it.



□ eel-7.jpg (imagen JPEG, 726 × 418 pixeles)
1 save http://animal.discoveryonline.com/animal-pictures/eel-7.jpg
sea coast river eel

□ Sparus_aurata.jpg (imagen JPEG, 1200 × [illegible] pixeles)
1 save http://upload.wiki[illegible].org/wikipedia/commons/a/aa/Sparus_[illegible].jpg
Sea Coast Sparus

□ yellowfin-bream.jpg (imagen JPEG, [illegible]20 × 288 pixeles)
1 save http://www.[illegible]dpi.nsw.gov.au/__data/assets/image/[illegible]14/116141/yello[illegible]-bream.j...
Coast Sea Bream

□ linnaeus300-02Scatophagus_[illegible]gus[Kallang-25aug2005].jpg (imagen JPEG, 600 × 398 p...
1 save http://rmbr.nus.edu.[illegible]/news/images/linnae[illegible]300-02Scatophagus%20argus%5BKall...
River Coast Scatophagus

□ common_carp.jpg (imagen JPEG, 1000 × 428 pixeles)
1 save http://[illegible]sh.dnr.cornell.edu/nyfish/Cyprinidae/common_carp.jpg
Carp River

	River	Coast	Sea
Carp	X		
Escatofagus	X	X	
Bream		X	X
Sparus		X	X
eel	X	X	X

Fig. 3 Extracting concept lattices from user's tagging

2.3 Context dependent knowledge heterogeneity and FCA

There are several limitations to collaborative tagging in sites such as Delicious. The first is that a tag can be used to refer to different concepts, i.e. there is a context-dependent feature of the tag associated with the user. This dependence-called "Context Dependent Knowledge Heterogeneity" (CDKH)—limits both the effectiveness and soundness of collaborative tagging. The second is the Classical Ambiguity (CA) of terms, inherited from natural language and/or the consideration of different "basic levels" among users [12]. CA would not be critical when users work with urls (content of url induces, in fact, a disambiguation of terms because of its specific topic). In this case, the contextualization of tags in a graph structure (by means of clustering analysis) distinguishes the different terms associated with the same tag [22]. However, CDKH is associated with concept structures that users do not represent in the system, but that FCA can extract. Thus, navigation among concept structures of different users is faced with CDKH. Therefore the use of tagged resources for automatic recommendation is not advisable without some kind of semantic analysis. More interesting is the idea of deciphering the knowledge that is hidden in user tagging in order to understand their tagging behaviour and its implied meaning. In sites such as Delicious, CDKH is the main problem, because tags perform several functions as bookmarks [12].

3 Agent-based reconciliation

Users knowledge reonciliation aims to exploit an important benefit of the Web 2.0, namely information and knowledge sharing. A potential threat is that semantic techniques are adapted to each user. Over time, the user's knowledge can suffer significant changes, and this difference could create knowledge incompatibility issues. In order to navigate through the set of tags and documents from different users, it could be interesting to delegate semantic tasks to agents, in order not only to make these different conceptualizations compatible but to make the process scalable to a great number of users. The agent-based reconciliation process was successfully applied in Mobile Web 2.0 [4] An agent-based reconciliation algorithm was presented in [2]. It is based on the idea that the conceptual structure associated with tags gives more information about users' tagging. The algorithm runs in six steps (see Fig. 4):

1. **Agent creation:** It starts creating two Jade[2] agents, passing through agent names and Delicious data as parameters.
2. **Each agent then builds its own formal contexts and stem basis**.
3. **Initializing dialogue step:** The agent executes tasks related to communications: It sends its own language (attribute set) to the other agent, and also prepares itself to receive the same kind of messages from the other agent.
4. **Restrictions of formal contexts:** After this brief communication, each agent creates a new (reduced) set of common attributes, and with them a new context

[2] http://jade.tilab.com

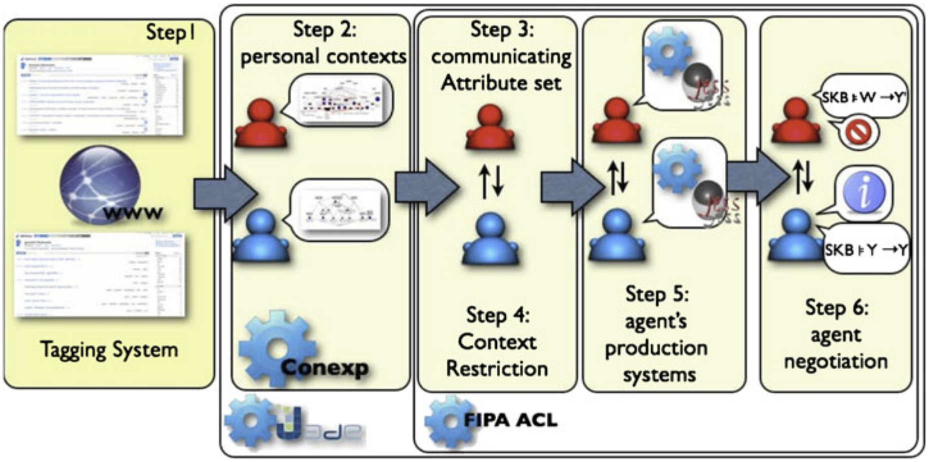
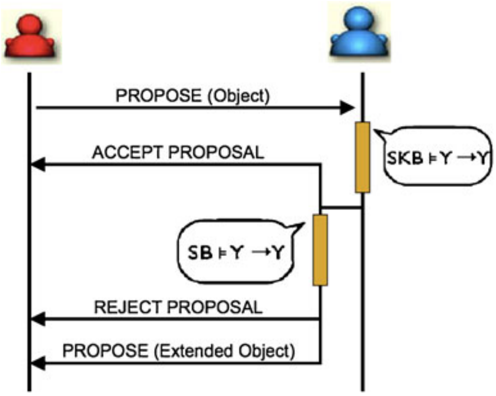
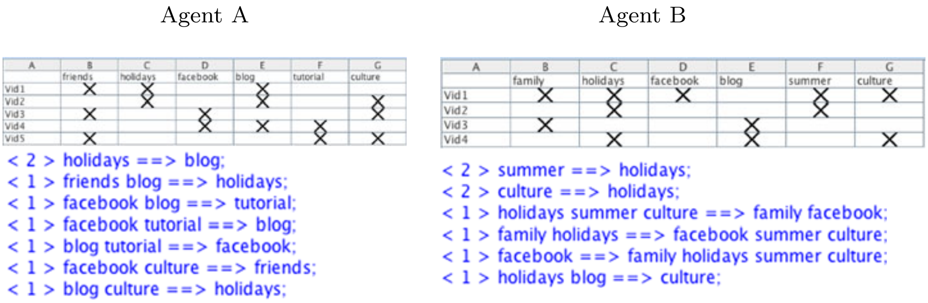


Fig. 4 Basic knowledge reconciliation algorithm

Fig. 5 Agent negotiation





Agent A

A	B	C	D	E	F	G
	friends	holidays	facebook	blog	tutorial	culture
Vid 1	X	X		X		
Vid 2		X		X		X
Vid 3	X		X			X
Vid 4			X	X	X	
Vid 5	X				X	X

< 2 > holidays ==> blog;
< 1 > friends blog ==> holidays;
< 1 > facebook blog ==> tutorial;
< 1 > facebook tutorial ==> blog;
< 1 > blog tutorial ==> facebook;
< 1 > facebook culture ==> friends;
< 1 > blog culture ==> holidays;

Agent B

A	B	C	D	E	F	G
	family	holidays	facebook	blog	summer	culture
Vid 1	X	X	X		X	X
Vid 2		X			X	
Vid 3	X			X		
Vid 4		X		X		X

< 2 > summer ==> holidays;
< 2 > culture ==> holidays;
< 1 > holidays summer culture ==> family facebook;
< 1 > family holidays ==> facebook summer culture;
< 1 > facebook ==> family holidays summer culture;
< 1 > holidays blog ==> culture;

Fig. 6 Contexts and implication basis for agents A and B

Table 1 Common language for agents A and B

Holidays	Facebook	Blog	Culture

to which are added all of the objects from the original context are added, along with the values and attributes of the common language.

5. **Extraction of the production system** (Stem Basis) for the new contexts.
6. **Knowledge negotiation between agents** (Fig. 5): Agents establish a conversation based on objects, accepting them (or not) according to their tag set and their own Stem Kernel Basis: if the object matches the rules, it is accepted. If not, the production system is applied, considering the object's tags as facts, getting the answer (new facts which should be added in order to be accepted as a valid object) that is added to the object and re-sent to the other agent to be accepted.

Once this process is completed, the agents will achieve a common context. Therefore they can extract new concepts and suggestions from a common context, and thus a shared conceptualization.

To clarify all processes described above, we have designed an experimental example where two agents are involved and they get a common conceptualization on a multimedia database.

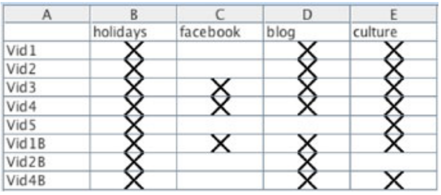
In early stages, we create two agents, called A and B, which represent to 2 human users (step 1), and recover all personal information from the database (step 2) for building its own contexts and Stem basis (Fig. 6).

At the third step agents send their own vocabulary (set of tags) to each other and create a new empty context where they will build the common context (knowledge), based only on the common set of tags (Table 1).

Then, agents reduce their previous contexts to common language (step 4), removing uncommon tags and avoiding empty objects (videos which are not held by any of common tags). These reduced contexts will be used for negotiation and building the common context. Some extracted implications (step 5) are:

```
holidays -> blog                        | culture -> holidays
blog, culture  -> holidays              | facebook -> holidays, culture
holidays, facebook, blog  -> culture    | holidays, blog  -> culture
```

At the last stage agents send objects (with associated tags) belonging to these reduced contexts to the other agent requesting if this object can be accepted for them. In order to decide if objects would be accepted (or not), each agent sets up two production systems (step 5): one with SB implications and other with SKB implications.



A	B	C	D	E
	holidays	facebook	blog	culture
Vid1	X		X	X
Vid2	X		X	X
Vid3	X	X	X	X
Vid4	X	X	X	X
Vid5	X			X
Vid1B	X	X	X	X
Vid2B	X		X	
Vid4B	X		X	X

```
{} -> holidays
holidays, facebook -> culture, blog
```

Fig. 7 Common context (after reconciliation process) and implication basis

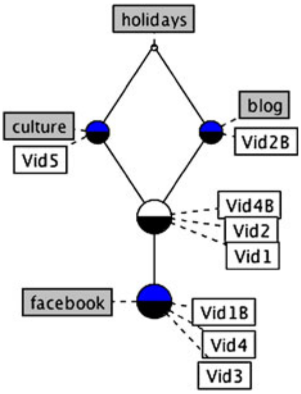


Fig. 8 Concept lattice associated to the reconciled context

Finally, agents send the objects to each other to be analysed (step 6). If they are accepted, both agents add the objects to common context, previously created. If not, the receiving agent, based on SKB, tries to add some suggestions (more tags) and send back the object to the other agent, restarting the process. In case there is no suggestion, the object will be rejected, and will not belong to the common context. The common context is in Fig. 7. See in Fig. 8 the concept lattice associated to new formal context.

4 Multiagent system

Our aim is to find a good strategy in order to apply the reconciliation algorithm presented above in Delicious. This algorithm allows us to compute the reconciled knowledge. However, it requires high computational resources. Hence, choosing the right pairs of users to execute the algorithm, among the whole community, remains a problematic issue. In order to execute a solution that computes reconciled knowledge for the whole tagging system, a negotiation based on MAS is proposed, in which agents represent tagging system users. They interact with each other in order to generate new common knowledge using the above-mentioned algorithm. In the following section, the results obtained are presented for different parameters used in the negotiation process. The MAS has also been implemented in Jade, where the implementation of the previous algorithm can be easily integrated. The execution of MAS can be described by means of the following steps (see Fig. 9. Section below is devoted to describe the Data Cleaning process which appears in the figure).

4.1 Initialization

In this step, as many user agents as needed are created. Only users sharing a minimum number of tags (threshold) participate in the MAS. Execution starts by creating an agent, called *control*, which passes this threshold as a parameter. This agent searches the DB for all pairs of users satisfying the threshold condition, and creates them within the MAS. Therefore, the agents present in the system are known by the *control* agent. The control agent may be useful in managing the MAS when integrated in more complex systems. Every *User*$_i$ must know their personal information

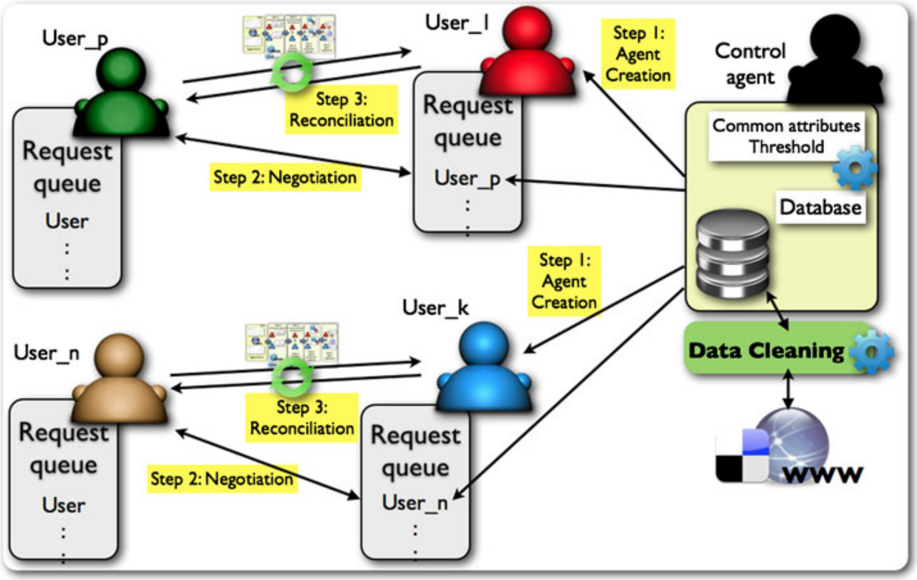


Fig. 9 Execution of the multiagent system

(username, links and tags), and initialize itself by creating its own request queue. This queue contains references of all users having an equal or greater number of common attributes. It is sorted by the number of common attributes, in descending order. However, these queues are simplified by removing all the references to users with an identifier lower than their own. As a result, a pair of users satisfying the threshold, will be referenced only once in one of their request queues, i.e, in the queue of the user with lower ID. Further experiments use the number of common objects of a pair of users as the threshold in order to compare results from both executions.

4.2 Negotiation

User agents must execute a dual behaviour in order to perform the negotiation process: sending and receiving requests. The mechanisms to perform the negotiation are based on communication protocols implemented in Jade as messages between agents. This negotiation establishes a very simple method to decide when a pair of users starts the reconciliation process. Each user is only allowed to perform one reconciliation process at a time. Furthermore, received requests have priority over the sent ones. Two possible states for each user are defined: *free*, if it is not performing any reconciliation at the moment, or otherwise, *busy*. As such, only free users may send or receive requests. On one hand, every user sends proposals to the user having the highest priority in its request queue. If it receives a response, the reconciliation process with the addressee starts. Should this not be the case, it reiterates with the user having the next highest priority. Nonetheless, if it is the case that none of the proposals sent has been accepted, this user continues sending proposals from the beginning of its request queue, i.e., it sends a proposal to the user having highest priority in this queue, and so forth. On the other hand, every free user accepts any incoming proposals, even if it has already sent another proposal, which will be cancelled by timeout. The following conditions ensure that all of the conciliations will be processed: their number is finite, and there are always free users ready to accept new conciliations, reducing the number of unsolved processes. When starting a reconciliation, the user's state switches from free to busy.

4.3 Reconciliation

The algorithm presented in Section 3 is used to compute the common knowledge between two users. The steps 1 and 2 (user's concept lattice and SB) are executed only once, when the user runs it for the first time. The rest of the steps (3–6), are executed each time the user runs the algorithm. The obtained common knowledge, a formal context with objects and common attributes, is stored in the DB. Both users switch from a busy to a free state.

4.4 Finalization

When a user's request queue becomes empty, its behaviour is limited to receiving incoming proposals. However, if all the users' request queues are empty, no proposal is received by any of them. Therefore, this situation requires that the execution stop. The *control* agent is used to manage it. It is informed by every user when their request queue becomes empty. When all the users have completed this action, the control agent stops the MAS execution.

5 Cleaning data from delicious

We have chosen the bookmarking service Delicious due to its large volume of data. In Delicious, objects are web links (urls), and attributes are tags. Users save their personal web links tagged with their personal tags. However several users may share common objects (with different attributes for each one), or common attributes (tagged in different links). The structure and dynamics of tagging with Delicious have been extensively analyzed [12]. Because of limited computing capacity, certain reduction operations must be performed in order to ensure the normal functioning of the solution presented in this paper. Therefore, a subset of public Delicious data has been extracted, in which all the links are tagged with the tag *haskell*, and saved in a private database (DB) used to drive experiments. We have selected this tag (*haskell*) in order to reduce the possible semantic heterogeneity generated by other more general tags. In this case, this tag can be almost always found related to the functional programming language with the same name. Hence, this heterogeneity is considerably reduced.

The process of obtaining this data is achieved through a query by tag (*haskell*) in Delicious. The result of this query is a list of links and its public information: the user who tagged it and the tags used in that link by that user. All this content is saved in our private DB, generating a huge amount of information: lists of links, users, tags, and tuples *{user, link, tag}*. This DB initially has 4,327 users, 3,163 links, 2,715 tags and 57,497 tuples. Data extraction was performed on March 1st, 2011. This data set has a volume large enough to expect significant results. However, this set of data does not encompass all the links related to the *haskell* tag, instead only the results of the first query.

The Delicious extraction process above described, generates a large amount of information which is saved into our private DB. However, some optimizations can be performed in order to obtain a better organisation of data. Some of these tasks remove irrelevant or duplicated data contained in the DB. Others improve the relational structure of tags.

5.1 Removing irrelevant tags

On the one hand, *haskell* is an irrelevant tag because all links are tagged with it. Therefore it does not provide any useful information. On the other hand, there are several tags that appear only once after removing the tag *haskell*. Based on this, tags can be considered marginal because they have no relation[3] with any other.

In order to ensure database consistency, besides removing such tags, all tuples containing these tags must also be removed. Moreover, links with no tags (after tuple elimination) make no sense in our experiments. Therefore, they are also removed. This task removes 45 tags, 101 links and 1,550 tuples.

5.2 Removing equal links

It is important to understand how Delicious stores urls to appreciate that some links can be repeated in different registers of the database. For instance, if several users save the same url with different descriptions, Delicious creates different descriptions as registers in its DB, although the url is always the same.

For our purpose, this means an important problem of inconsistency in the set of objects (links). To solve it, repeated links are simplified in one of them, and tuples containing these links are updated. Additionally, if some tuples become repeated after being updated, they are also removed. The result of this task is the elimination of 14 links and 3 tuples, and the updating of 552 tuples.

5.3 Removing equivalent links

In the same manner of the previous case, several links can be duplicated in the Delicious DB due to the users' behaviours when saving it, e.g., links with and without a final slash. As mentioned above, equivalent links are simplified, and tuples containing these links are updated. The result of this task is the elimination of 19 links and 40 repeated tuples, and the updating of 4,086 tuples.

5.4 Joining singular and plural tags

As is explained in Section 2.3, there exists a limitation of extracting knowledge from the tags due to the own limitations of the users' natural language. In the next section, it is shown that tags present a clear relational structure. However, this structure can

[3] A pair of tags are related when both of them tag the same link by the same user.

be strongly improved with few changes to the context. One of the more common examples is the usage of tags, either in singular or in plural, depending on the user. This intersection brings about a separation that FCA tools can rarely correct. Nevertheless, some simple rules can be performed to drop these limitations and get a better running of the experiments.

Motivated by this argument, singular and plural tags are simplified in a unique tag. The process of obtaining these pairs of tags is quite simple: the same tag with and without a final '*s*'. But this rule must be considered with caution, because it is neither correct, nor complete. Consequently, not all singular-plural pairs are found, and some incorrect pairs can be also found (in fact, 6 of the 197 results are incorrect and must be considered as exceptions to avoid the corruption of the data). However, the result of applying this rule is amazing: the elimination of 191 tags and 1,667 repeated tuples, and the updating of 3,765 tuples.

5.5 Joining equivalent tags

Likewise, many tags can be grouped because they are very similar, but they are written with different words, e.g., *math, maths, mathematic, ...*. In the next section, the process of obtaining the main tags and the structure between them is illustrated. In this case, no automatic process has been implemented for this task. However, the analysis of the structure of tags lets us to know the most important groups of tags that must be classified. They are *math, programming, functional programming, programming language, language, tutorial* and *to read*.

This task consists of simplifying these groups of tags, and updating the corresponding tuples. The result is the elimination of 52 tags and 158 repeated tuples, and the updating of 611 tuples.

5.6 Removing empty users and empty links

Finally, a maintenance task must be performed in order to assure the coherence of the DB, i.e., removing users with no links tagged, and links with no tags. This task removes 68 users and 1 link.

The resulting DB is composed by 4,259, 3,028, 2,427 tags and 45,079 tuples.

5.7 The relational structure of tags

In order to estimate the complexity of the relationships among of data source tags, a graph was generated. In this graph, nodes represent tags, and a pair of different tags are connected by an edge when both tags are used in the same link by a Delicious user. Thus, these edges can be weighted depending on the number of different tuples links-users in which these tags are tagged: The weight function $w : E \to \mathbb{R}^+$ is defined by

$$w((a,b)) = \sum_{\substack{M_i = (O_i, A_i, I_i) \\ a, b \in A_i}} \sum_{o \in O_i} (I_i(o,a) \wedge I_i(o,b))$$

where for each i, M_i is the formal context associated with a Delicious user $User_i$.

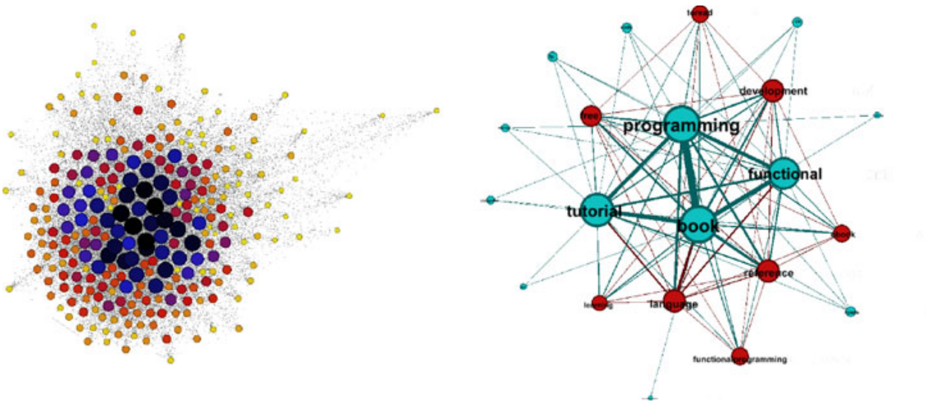


Fig. 10 Analysis of tag communities induced by the *haskell* tag in Delicious. Both graphs are simplified versions of the original one. Nodes are scaled according to their degree, and edge width is scaled according to its weight. The *colours* of the nodes represent the different communities obtained

The size of this graph depends on the volume of data, which is very large in our experimentation (as well as in real applications such as Delicious). However, in order to understand the structure of the graph and the number of relevant tags, some simplifications have to be made.

Figure 10 shows data resulting from the computing of *semantic communities* (using the method [6]), which are simplified graphs. In these graphs, each node is characterized by its color (determining the community it belongs to), its size (scaled according to its degree), and by the width of its edges (scaled according to the weight of the edge). On the left, we present the structure of communities of the original graph, i.e., the different colours of the nodes. In order to make its visualization possible, only the most relevant nodes (270) and edges (16,059) are shown—accordingly measured by their importance in terms of degree and weight, respectively. On the right, a very reduced version of the previous graph is shown in order to understand the structure of the main tags. This graph shows 2 different communities, demonstrating that tags of a same community are very interconnected, unlike tags of different communities, which display little connectivity. As above, only the most relevant nodes (21) and edges (86) are shown.

6 Experiments

Different experiments have been conducted with data described in Section 5 using several criteria. The first criterion is setting a threshold of common attributes (tags) between users. The second criterion is setting a different threshold of common objects (urls). In both cases, the threshold is a necessary condition of a minimum number of attributes or objects that two users must have in common in order to execute the reconciliation algorithm. For each executed reconciliation process, a common knowledge is obtained. This knowledge is a formal context where the attributes are common to both users, and objects belong either to one of them, or both. In this way, the global result is a set of reconciled contexts.

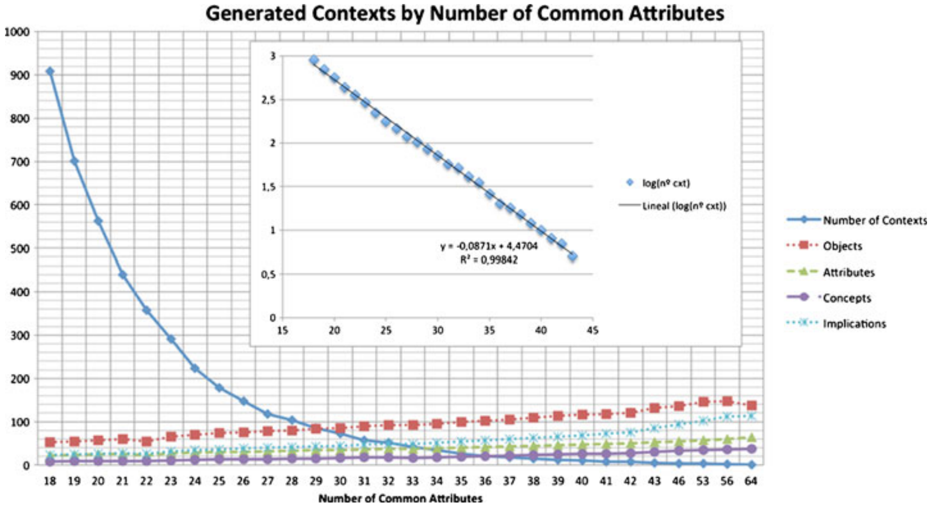


Fig. 11 Results of reconciliation attending to the number of common attributes (number of contexts in Log-Log scale in the *top–right* screen

The results obtained for both experiments using graphic representations are presented below. In order to do so, the results have been measured with five parameters for a fixed value of the threshold. They are the number of contexts obtained (there are as many contexts as number of executed reconciliation processes), and the average values of objects, attributes, concepts and implications per context. Finally, both experiments are compared.

6.1 Reconciliation from common attributes

In this experiment, the threshold value is set to 18. That is, that two users having a common vocabulary[4] size greater or equal to 18 reconcile their knowledge. It is assumed that users having a very small number of common attributes do not share a relevant amount of information. Therefore, for the purpose of this study, we have arbitrarily selected this value (18), that generates a huge amount of results and avoids excessive time calculation of lower thresholds.

A total of 908 contexts were obtained, with an average value of 52.3 objects, 22 attributes, 8.1 concepts, and 22.8 implications per context. As the threshold value increases, the number of generated contexts decreases exponentially, as we can see in the log plot in Fig. 11. However, the four average values tend to increase. Although the number of contexts is smaller, they are semantically better, since the two users generating these contexts share more information. In this DB, the maximum number of common attributes is 64.

[4] The set of common tags that both of them use, independent of wether or not these tags have been used in different urls or not.

6.2 Reconciliation from common objects

In the second experiment, the threshold value is set to 3. The implication is that two users having a set of common objects with size greater than or equal to 3 reconcile their knowledge. As previously mentioned, it is assumed that sharing less than 3 objects is not relevant for the purpose of this study. In Fig. 12, the results are presented. This case shows a total of 663 contexts, with an average value of 33.55 objects, 9.41 attributes, 6.09 concepts, and 9.75 implications per context. As the threshold value increases, the number of obtained contexts also has a large decrease. The maximum number of common objects is 11, which is very small. In Fig. 12, we show the numerical result for this criterion depending on the threshold value.

6.3 Results

The main conclusion that can be drawn from results is that a common attributes criterion is better than a common objects criterion for the following reasons. On one hand, the decrease in generated contexts is higher when using common objects rather than common attributes (Log-Log scale shows better than behavior). On the other hand, the number of objects, concepts and implications obtained and the number of objects and attributes of the context obtained in the reconciliation, measured along with their average values (i.e., objects, attributes, concepts and implications), is higher using attributes rather than objects. In the first case, average values increase in a linear fashion (see Fig. 13) while the second case does not (Fig. 14, $R^2 < 0.9$). It

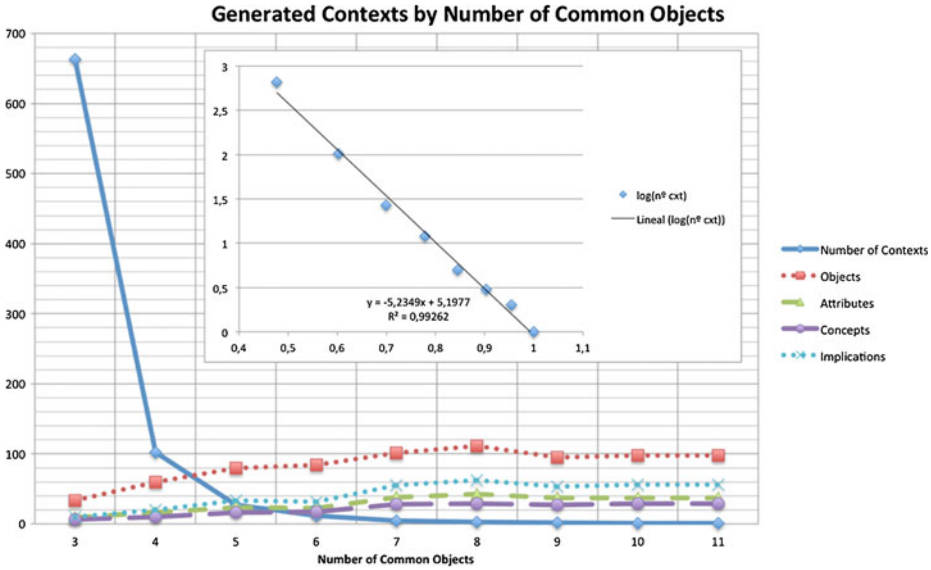


Fig. 12 Results of reconciliation attending to the number of common objects (number of contexts in Log-Log scale in the *top–right* screen)

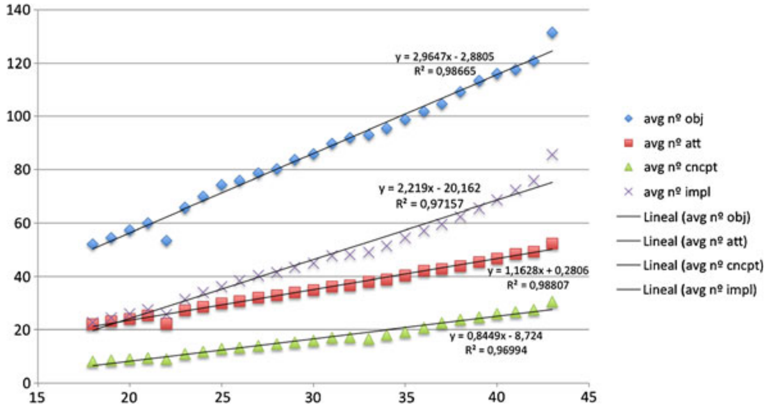


Fig. 13 Growth of average values in the first experiment (common attributes criteria)

is then thought that the higher the number of common attributes, the more quality in the reconciled context obtained. It seems to suggest that the quality of the generated contexts does not depend on the number of common objects. It is worthwhile to remark on the lineal growth of average size of implication basis (Stem Basis) in the first case. We can conclude that the reconciliation process based on the number of common attributes does not generate a considerable amount of new relations, hence the reconciliation algorithm produces a manageable implication basis on the relations among tags. Therefore the use of the implication basis for recommending tags is feasible (as in [4] in Mobile Web 2.0).

In conclusion, previous results lead us to think that the common attributes criterion more effectively separates the sample of generated contexts. Indeed, despite the fact that it returns a smaller amount of contexts, increasing the threshold value leads to *semantically* better results. Therefore, it is a good measurement of the semantic similarity of two users.

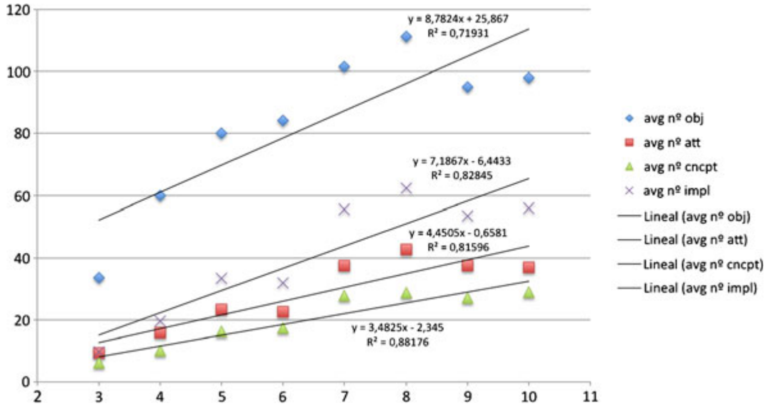


Fig. 14 Growth of average values in the second experiment (common objects criteria)

7 Related work

It is worthwhile to comment that, throughout several works related to the topic of this paper, the FCA-based approach for knowledge reconciliation can be considered a broad approach to the problem, although specific features of FCA, limits the application of the method in complex scenarios. The most natural option could be the use of ontologies, but, in a way, ontologies can suppose a barrier while folksnomies are more versatile Knowledge organization methods. As it is remarked in [17], unlike ontologies, folksonomies do not suffer from the knowledge acquisition bottleneck, as shown by the significant provision of content by many people. On the other hand, folksonomies unlike ontologies, do not explicitly state shared conceptualizations, nor do they force users to use the same tag sets. Our method does not only force shared conceptualizations, in fact the Knowledge reconciliation is a post-tagging process that the user needs in a concrete case. However, the usage of tags from users with similar interests tends to converge to a shared vocabulary (at least a core set of tags [14]). In the cited [17] the intention is to discover shared conceptualisations that are hidden in a folksonomy, so this convergence is a feature of collaborative tagging systems that are more relevant than in the approach presented in the paper. The discovery of shared conceptualizations uses *triadic concepts* (a generalization of formal concepts where authors of tags are explicitly considered), but triadic concepts are harder to compute.

The rationale behind our approach is more similar to that of [18], in which the author aims to uncover hidden relationships between people whose contexts are most similar without to establishing global structures both to organize and to retrieve knowledge.

An important problem in the framework considered in this paper is the heterogeneity, both in semantic tagging and user behavior. A intermediate solution between shared conceptualizations and personomies can be the use of well-established vocabularies used by professionals on the subject. For example, the work [11] is devoted to investigating the vocabulary users employ when describing videos and to compare it with the vocabularies used by professionals, as well as to establish which aspects of the video are typically described and what type of tags are used for this. The results presented in the paper suggest that tagging is a nice complement to metadata and that tags are mainly used to describe objects in the video. From the point of view of Knowledge Reconciliation, the presence of two complementary vocabularies imply two different tasks: at metadata level, ontological alignment methods should be selected, while at tagging level, our approach can be considered. Tools such as SHIATSU [5] allow to reconciliate both kinds of vocabularies in a unique framework by means of the use of term hierarchies, as well as prediction other tags. A similar complex scenario appears when textual notes also appears as tags in images or video. In *OpenAnnotation Model* terms [15], those are annotations that have a textual body and an image resource as a target. However, in many scenarios the prevailing view that annotation bodies are textual is insufficient. Our approach cannot work with triples (Annotation, Body, target) as these are considered in [15], although it may be feasible to use triadic concepts to consider them.

A relatively similar approach to our method is described in [20]. The aim of the authors is to discover knowledge from folksonomies, by isolating a set of users that share some formal concepts, and reapplying FCA-methods on this set of users and their tagging. That is to say, the selection of users to conciliate the knowledge is based on the sharing of some concept in the global shared conceptualization. Our approach is an *user-centric knowledge retrieval tool*, thus it considers the Knowledge reconciliation between users with no common concepts (possibly due to semantic heterogeneity.

Semantic heterogeneity in tagging is a well-known problem that is harder in multimedia tagging. In [24] it aims to solve the problem of identifying descriptive tags, i.e. tags that relate to visible objects in a picture. The combination of the solution with Knowledge Reconciliation, applied to descriptive tags, allows the isolation the navigation between personomies using only features of the video/image contents.

In [21] an approach to measure the topical relevance by using the tags attached to web objects is introduced. It is based on the preference of the set which covers as many related subtopics as possible. In this way it can avoid noisy tags. In that paper the solution is applied to photo set search at flickr.com, where individual photos are annotated with texts such as titles and tags. In our case, it is interesting to exploit the technique to reduce the number of tags by eliminating non-relevant tags as a prior task. In this way, it hopes that the number of implications among non-relevant tags can be reduced.

Knowledge reconciliation can be a useful tool in multilingual folksnomies. The method described in [19] allows the discovery of a set of tag correspondence in multilingual communities of practice. Once the tag set is discovered, it suffices to semantically identify tags (represent the same attribute) and then to apply the Knowledge reconciliation method. In this way we can empower multilingual communities of practice by reconciling social tagging.

Lastly, the knowledge reconciliation method can be considered as a dialogue between two agents which is based on the recommendation of new tags for resources. Although the method is not designed to be a P2P recommender system (a centralized system to make this was implemented in SinNet [4].

8 Conclusions and future work

The experiments described in this paper show the prevalence of semantic techniques (tags) in resource sharing, when users aim to exploit knowledge organization from other users in Delicious as a recommendation source. Although this result seems evident, Web 2.0 shows several examples where url sharing by social networks represents a powerful method for information diffusion (e.g. Twitter).

Therefore, we have empirical evidence that semantic similarity between users is better supported by using the method of reconciling the knowledge among users with a large set of common attributes, rather than the sharing of tagged resources. One of our lines of research is the intensive application of definability methods based on completion [1] in order to enrich the bookmarking system and to facilitate the reconciliation.

In a more general framework, and due to the popularity of tagging as weak knowledge organization method, it is worthwhile to emphasize how useful the Knowledge reconciliation method presented in the paper is for the automatization of integration of distributed information sources. The ambiguity in concept interpretation, also known as semantic heterogeneity, has become one of the main obstacles to this process [9]. Since our method is based on FCA, reconciliation is a disambiguation of concept interpretation between two users. Therefore it is interesting to consider its application in P2P-like environments for knowledge integration.

Acknowledgements This work has been supported by TIN2009-09492 and TIN2010-20967-C04-01 projects of Spanish Ministry of Science and Innovation and *Excellence project* TIC-6064 of *Junta de Andalucía* cofinanced with FEDER founds.

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper.

References

1. Alonso-Jiménez J-A, Aranda-Corral G-A, Borrego-Díaz J, Ferníndez-Lebrón M-M, Hidalgo-Doblado M-J (2008) Extending attribute exploration by means of boolean derivatives. In: Proc. 6th int. conf. on concept lattices and their applications, CLA-2008, Olomouc, Czech Republic. http://ceur-ws.org/Vol-433/paper10.pdf. Accessed 17 May 2012
2. Aranda-Corral G-A, Borrego-Díaz J (2010) Reconciling knowledge in social tagging web services (2010). In: Proc. 5th int. conf. hybrid AI systems, HAIS 2010. Springer, Berlin, Heidelberg, pp 383–390
3. Aranda-Corral G-A, Borrego-Díaz J, Giráldez-Cru J (2012) On the complexity of shared conceptualizations. In: Proc. 11th international conference on artificial intelligence and soft computing, ICAISC 2012. Springer, Berlin, Heidelberg, pp 629–638
4. Aranda-Corral G-A, Borrego-Díaz J, Giráldez-Cru J (2012) Conceptual-based reasoning in mobile web 2.0 by means of multiagent systems. In: Knowledge engineering notes, proc. 4th int. conf. agents and artificial intelligence ICAART 2012, pp 176–183
5. Bartolini I, Patella M, Romani C (2011) SHIATSU: tagging and retrieving videos without worries. Multimed Tools Appl pp 1–29. doi:10.1007/s11042-011-0948-1
6. Blondel V-D, Guillaume J-L, Lambiotte R, Lefebvre E (2008) Fast unfolding of communities in large networks. J Stat Mech Theory Exp 10:1–12. doi:10.1088/1742-5468/2008/10/P10008
7. Carmel D, Roitman H, Yom-Tov E (2010) Social bookmark weighting for search and recommendation. VLDB J 19(6):761–775
8. Eklund P-W, Wray T (2010) Social tagging for digital libraries using formal concept analysis. In: Proc. 7th int. conf. on concept lattices and their applications, CLA 2010, Seville, Spain, pp 139–150. http://ceur-ws.org/Vol-672/paper13.pdf. Accessed 17 May 2012
9. Gal A, Shvaiko P (2008) Advances in ontology matching. In: Advances in web semantics I. Springer, Berlin, Heidelberg, pp 176–198. doi:10.1007/978-3-540-89784-2_6
10. Ganter B, Wille R (1999) Formal concept analysis - mathematical foundations. Springer, Berling, Heidelberg
11. Gligorov R, Hildebrand M, van Ossenbruggen J, Schreiber G, Aroyo L (2011) On the role of user-generated metadata in audio visual collections. In: Proc. 6th international conference on knowledge capture, K-CAP '11, Banff, Alberta, Canada, pp 145–152. doi:10.1145/1999676.1999702
12. Golder S, Huberman B-A (2006) The structure of collaborative tagging systems. J Inf Sci 32(2):98–208
13. Guigues J-L, Duquenne V (1986) Familles minimales d' implications informatives resultant d'un tableau de donnees binaires. Math Sci Hum 95:5–18

14. Halpin H, Robu V, Shepherd H (2007) The complex dynamics of collaborative tagging. In: Proc. 16th int. conf. World Wide Web, WWW '07, Banff, Alberta, Canada, pp 211–220. doi:10.1145/1242572.1242602
15. Haslhofer B, Sanderson R, Simon R, van de Sompel H (2012) Open annotations on multimedia Web resources. Multimed Tools Appl pp 1–21. doi:10.1007/s11042-012-1098-9
16. Jäschke R (2011) Formal concept analysis and tag recommendation in collaborative tagging systems. IOS Press, Heidelberg
17. Jäschke R, Hotho A, Schmitz C, Ganter B, Stumme G (2008) Discovering shared conceptualizations in folksonomies. J Web Semantics 6(1):38–53
18. Jung J-J (2009) Knowledge distribution via shared context between blog-based knowledge management systems: a case study of collaborative tagging. Expert Syst Appl 36(7):10627–10633
19. Jung J-J (2012) Discovering community of lingual practice for matching multilingual tags from folksonomies. Comput J 55(3):337–346
20. Kang Y-K, Hwang S-H, Yang K-M (2009) FCA-based conceptual knowledge discovery in Folksonomy, world academy of science. Eng Technol 53:842–846
21. Lee S, Park J (2011) Topic based photo set retrieval using user annotated tags. Multimed Tools Appl pp 1–20. doi:10.1007/s11042-011-0850-x
22. Man C, Yeung A, Gibbins N, Shadbolt N (2009) Contextualising tags in collaborative tagging systems. In: Proc. 20th ACM conference on hypertext and hypermedia, Torino, Italy. doi:10.1145/1557914.1557958
23. Onifade O-F-W, Thiéry O, Osofisan A-O, Duffing G (2010) Fuzzontology: resolving information mining ambiguity in economic intelligent process. Commun Comput Inf Sci 54:232–243
24. Pammer V, Kump B, Lindstaedt S (2011) Tag-based algorithms can predict human ratings of which objects a picture shows. Multimed Tools Appl 59(2):441–462. doi:10.1007/s11042-011-0761-x
25. Smith G (2007) Tagging: people-powered metadata for the social web. First New Riders Publishing, Berkeley, CA
26. Weick K-E, Sutcliffe K-M, Obstfeld, D (2005) Organizing and the process of sensemaking. Organ Sci 16(4):409–421