

Extended Social Tags: Identity Tags Meet Social Networks

Sonia Lajmi*, Johann Stan^{† ‡}, Hakim Hacid[†], Elöd Egyed-Zsigmond* and Pierre Maret[‡]

*Université de Lyon, F-42023,

CNRS, UMR 5205, LIRIS, F-69621,

Villeurbanne, France,

{sonia.lajmi, elod.egyed-zsigmond}@insa-lyon.fr

[†]Alcatel-Lucent Bell Labs France

Centre de Villarceaux Route de Villejust

91160, Nozay

{johann.stan, hakim.hacid}@alcatel-lucent.com

[‡]Université de Lyon, F-42023

CNRS, UMR 5516, Laboratoire Hubert Curien,

Université de Saint-Etienne, Jean-Monnet

pierre.maret@univ-st-etienne.fr

Abstract—This paper proposes a new approach that uses social networks and common sense deduction rules to adapt the description tags of the photos for the current viewer. We exploit social graphs to enrich the tags associated to the concerned persons in the photo by following the different links between people (i.e. viewer and captured people in the photos). The main contributions of our work are: (i) addition of a more meaningful tagging layer for photos, making tags dynamic and auto-adaptable thanks to the automatic identification of the social context of the visualization. (ii) Due to this dynamics, the search in the social graphs is optimized using a data mining technique. (iii) we propose a new visualization metaphor for the tagging layer to manage users' feedback. We also describe a system architecture and an experimental study that shows significant improvements of the tagging process and execution times on a dataset containing triples in a FOAF graph.

Index Terms—Media tagging, social networks, semantic web, user profile, data mining, optimization.

I. INTRODUCTION

Online communities like Flickr¹, del.icio.us², Facebook³, or Youtube⁴ have established themselves as very popular and powerful services for publishing and searching content. Uploaded data is annotated with information about the content or the context in the form of freely selected keywords, called tags. Among the multimedia data types, images are undoubtedly the most used. Indeed, images are used in various fields like personal usage, medicine, museums, astronomy, etc. A big challenge in this domain is textual semantics association to an image. This can be done by tagging, also called annotation [2][13]. Multimedia data tagging is the task of assigning, for each multimedia document, or part of it, a keyword or a list of keywords describing its meaning. In the case of a personal

photo, such tags generally concern the captured people, the location, and the event. They have three major objectives: (i) to facilitate later retrieval, (ii) to give sense to the photo when sharing it, and (iii) to make it possible to automatically reason about images since this can't be performed directly on the image's low features (e.g. color and texture).

With the widespread of the Web 2.0 and social networks, manual and semi-automatic tagging have gained lot of interests. In Social Networking Sites (SNS) for example, users can tag people on images with their names. However, this gives the image tag a reduced interpretation range, since the tag will only make sense for those who know the involved person(s). Additional information is then needed for the viewer so that she can understand and interpret the potential relations between her and people in the photo. On the other hand, it is important to consider also the adaptive aspect of these tags: *a social relationship only makes sense for a given person*. Thus, these tags must adapt for each viewer.

In this paper we consider the case of photos in social networks and we focus our efforts on the tags performed on people (other tags on the photo may exist, e.g. objects, but we are not dealing with this aspect in this paper). We believe that many constraints need to be considered in this context (social networking). We briefly describe in the following the main requirements for such a tagging system that aims at fixing this kind of problems:

- *Real-time adaptation*: social tags, i.e. tags that are inferred from a social relationship, must adapt in real-time to the viewer. The argument for this is the fact that the path in the social graph between the viewer and actors/witnesses changes with the relationships.
- *Scalability*: social graphs can reach several millions of nodes. The optimization of social path computation is then an important requirement.

¹<http://www.flickr.com/>

²<http://www.del.icio.us/>

³<http://www.facebook.com/>

⁴<http://www.youtube.com/>

- *Relevance feedback management*: generally, many paths exist between two nodes in a social graph. Therefore, users should have the possibility to accept or reject an extended social annotation suggestion. If it is rejected, a new suggestion must be computed.

1) *Paper contributions*: This paper has contributions in social network modeling, multimedia document tagging, and information retrieval optimization. Our contributions can be mapped to the different requirements that have been described before. A first contribution is the addition of a more meaningful tagging layer for photos by exploiting the social relationships that can exist between people in a social network. To support this new layer, an extension of the FOAF vocabulary is also performed. This extension allows a meaningful social dimension label, with terms like friend, colleague, neighbor, partner, parent, etc. Since the social tags become highly self-adaptive, i.e. they must change for each viewer, the search in social graphs must be optimized. Our second contribution is the use of data mining to optimize the retrieval of paths. Finally, we propose a new visualization metaphor for the tagging layer. It is clear that multiple paths can exist between two persons in the social graph. Thus, the identity annotations appear in the form of a list of suggestions. Each suggestion can be associated with a person in the photo with a drag-and-drop operation. If the suggestion does not make sense for the user (too complicated, too long), the algorithm computes another one. It is important to note that the algorithm is designed to always suggest the shortest path. To the best of our knowledge, this is the first work that considers the fact that identity tags must have a social dimension in an environment where it is impossible to anticipate who will view the photo.

2) *Paper Organization*: The remainder of this paper is organized as follows: Section 2 discusses the new annotation schema that uses social relations to extend the traditional annotation schema. Section 3 introduces our heuristic algorithm that manages rapid access to annotations thanks to a data mining technique. We discuss a system architecture as well as preliminary experimental results in Section 4. Section 5 is dedicated to the study of some related work. Finally, we conclude and give some future directions in Section 6.

II. ENRICHMENT OF IDENTITY TAGS WITH SOCIAL RELATIONS

A. A new identity annotation schema

In this section we describe the new annotation schema, i.e. strategy, that supports the extended annotation as well as the extension of FOAF. The following terminology is used in the rest of this paper: “Actor” refers to a person present in a photo. A “Witness” is a person who has a social, geographical and temporal proximity with the captured photo, but is not necessarily visible in the photo itself. A person can be both Actor and Witness. An Actor is always a Witness too. The “Tagger” of the photo is the person who currently annotates the photo. The “Viewer” of a photo is the person who currently consults the photo (a Tagger is also a Viewer but the opposite is

not necessarily true). For this paper, we focus only on adapting the social dimension to the viewer. We consider the following definition of a social tag:

Definition 1: (Social Tag) A social tag is a label associated by a social community to a multimedia document, e.g. image, that describes its meaning for later interpretation, processing, or distribution.

To illustrate the proposals of this paper, let’s consider a simplified case of a scientific conference event. Researchers from different countries meet and exchange ideas and collaborate on projects. A main interest is to meet new or known people and create new or boost existing collaborations. Photos are often taken to memorize a meeting between researchers (See Figure 1(a)). David introduced a number of new researchers, Steven and Edward, to Jennifer who took a photo. Later in, Jennifer tagged it with the names of the persons and put it on her social network site. After this, she shared the photo with David, Steven and Edward (since they are in the photo). When Jennifer viewed the photo, after a while in her social network site, she recognized David but did not remember how she was related to the other persons. The reason for this situation is that Jennifer has followed the traditional annotation schema that we can formalize as follows:

$$Tags = [P] \vee [L] \vee [E] \vee [O] \vee [T] \quad (1)$$

Where P stands to persons names, L for location, E for events, O for objects, and T for time. The instantiation of this schema on our example can then produce the following result:

$Tags = [David; Steven; Edward; Joe; Andrew; Alex], [Vancouver], [SocialCom Conference], [Jade; Stanley Park], [29 august 2009].$

One of the most important particularity and weakness related to this annotation schema is that the result is the same for all persons who manipulate this photo, i.e. annotations are static. It becomes clear from this example that there is a missing piece of information, at least from Jennifer’s point of view. In fact, since Jennifer has only a direct social relation with David, the annotations about Steven and Edward become meaningless for her. We propose to overcome this problem by enriching the annotations with specific social relationships and by adapting them to the current viewer. Our proposal improves the traditional annotation schema (Formula 1) with the consideration of the social dimension of the tag about a person. It consists in the inclusion of the potential social relationships between the viewer and the annotated person(s). By supposing that Steven has a family relation with David, say his brother-in-law, and Edward is a research colleague, Jennifer would appreciate to have a more meaningful annotation, like:

$Tags' = [Steven, brother-in-law of my colleague David; David, My colleague; Edward, apprenticeTo my colleague, David; Joe, colleague of my colleague David; Andrew, collaboratesWith my colleague David; Alex, colleagueOf my colleague David], [Vancouver], [SocialCom conference], [Jade; Stanley Park], [29 august 2009]$

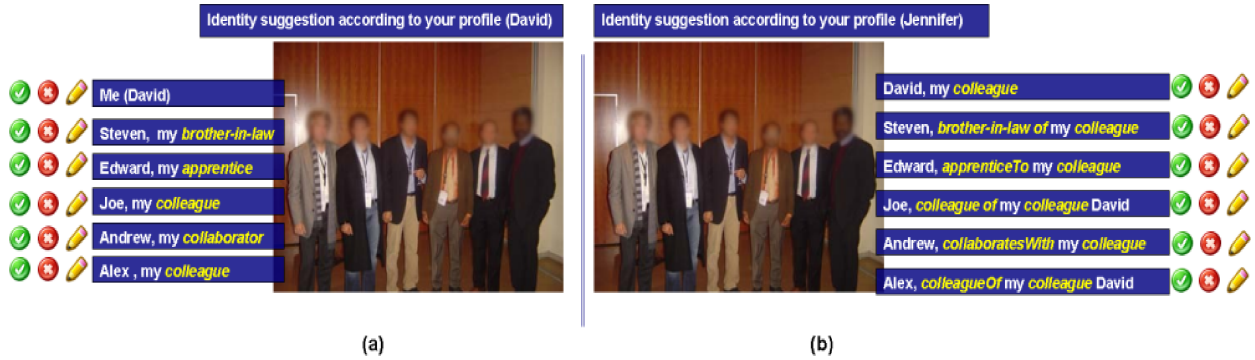


Fig. 1. Use Case Example. (a) identity suggestion for David, (b) identity suggestion for Jennifer

The general form of the new annotation schema becomes then:

$$Tags = [P \wedge S] \vee [L] \vee [E] \vee [O] \vee [T] \quad (2)$$

Where S is the social dimension associated to people's names. We define then a new concept "extended social tag" as follows:

Definition 2: (Extended social tag) An extended social tag is a social tag enriched with a social dimension describing the social relations between the concerned entities.

It results from Definition 2 a dynamic property that is associated to a social tag. In fact, since the social dimension is exploited and since social relations change from a person to another, the annotations need to be dynamic.

These two annotation schemes can be easily explained from a social network perspective. Consider a graph representation of a social network where nodes represent people and arcs relations between these people. The traditional annotation schema can be viewed as the task of assigning to each node in the graph a semantic annotation. This makes the nodes separate, preventing other nodes, which are not directly concerned by the photo, to interpret it. Our model proposes to build the missing part of the tags, i.e. the connections between the nodes, to improve the meaning of the annotated objects. This makes the annotations more meaningful for all the nodes that have a potential path to people annotated in the photo (See Figure 1).

To make this solution efficient and useful, rich relation representation description and formalisms need to be used. Currently, one of the most used formalism to represent relations is the Friend-Of-A-Friend (FOAF) vocabulary [3]. FOAF is an XML based vocabulary offering the possibility to describe relations between people. Using FOAF in our context is certainly necessary since we are considering relations between people. However, in its current definition, FOAF seems to be very basic and can not be exploited to describe a set of relations rich enough that can exist between people and which we want to exploit for annotations enrichment. We propose in the following to extend this vocabulary for a better satisfaction of our constraints.

B. Extension of the FOAF vocabulary

We focus here on describing the extensions that we have made on FOAF in order to consider additional types of relations between people. The extension of FOAF, and social ontologies in general, is not particular to our work but is investigated in other research studies [8]. The proposed extension is an attempt to address our specific issues. The difference between Relationship Ontology⁵ and our proposal called "SocialSphere Ontology" is that (i) we define a more exhaustive set of relationships, (ii) we categorize those relationships on different Social Network Categories, and (iii) we add common sense rules to perform reasoning about those relationships.

Inspired by the SAUPO model [15], we define the "Social Network Category" (SNC) concept as an extension of FOAF and a specialization of the <foaf:Group> concept. From the user's perspective, each member of her social network belongs to a category and therefore the <foaf:Person> concept is connected to the SNC concept by the "belongTo" property. Five different SNCs that specialize the SNC concept are defined: (i) Professional, (ii) Family, (iii) Neighborhoodship, (iv) Friendship, and (v) Intime. For each category, we provide the list of the corresponding relationships hereafter:

- *Professional*: worksWith, colleagueOf, collaboratesWith, employedBy, mentorOf, apprenticeTo, supervisorOf
- *Family*: parentOf, childOf, grandChildOf, grandParentOf, ancestorOf, descendantOf, siblingOf, uncleOf, cousinOf, nephewOf, nieceOf
- *Neighbourship*: livesWith, neighbourOf.
- *Friendship*: friendOf, lostContactWith, closeFriendOf, hasMet, acquaintanceOf.
- *Intime*: wifeOf, husbandOf, lifePartnerOf, engagedTo, girlfriendOf, boyfriendOf, husbandOf, wifeOf and each relation with the "ex" prefix (for expressing the unstable aspect of intime relations).

This extension, as discussed before, enables the definition of additional relationships that can appear in a social network. This is important in our context to semantically characterize the links of a social network. It should be noted that the

⁵<http://vocab.org/relationship/html>

relations as defined before are not always symmetric, e.g. if A is the wife of B then B is the husband of A. To prevent the definition of all the alternatives of the relations between people, we make use of some common sense rules for reasoning on the social graph resulting in a higher exploitation of the relationships. Additional implicit rules are defined using four OWL properties: (i) the inverse property $\langle owl:inverseProperty \rangle$, (ii) the transitive property $\langle owl:transitiveProperty \rangle$, (iii) the symmetric property $\langle owl:symmetricProperty \rangle$, and (iv) the similarity property $\langle owl:sameAs \rangle$. Since this is not the main topic of this paper and due to space limitation, this aspect is no longer detailed here.

III. OPTIMIZED SEARCH IN THE SOCIAL GRAPH

In real life, graphs representing social networks can reach a very large size. On Facebook, for example, the graph connecting all users has so far reached over 120 million nodes. To optimize the identity tag suggestion process in large social networks, we propose to optimize the search in the FOAF graph. A query traverses FOAF profiles and returns $\langle foaf:name \rangle$, $\langle SocialSphere:Relationship \rangle$ and $\langle foaf:seeAlso \rangle$ of the potential actors. Before presenting the optimized algorithm, we consider the following formal definitions:

Let $G(V, E)$ be a directed graph representing the social network. V is the set of n vertices/nodes ($|V| = n$) and E is the set of all arcs connecting the vertices in V . We consider that G is a connected graph and that it doesn't contain isolated vertices. If $v_i, v_j \in V$ then we denote by $(v_i, v_j) \in E$ the arc connecting the node v_i to the node v_j . It should be noted that since the graph is directed, the relations are not symmetric and thus $(v_i, v_j) \neq (v_j, v_i)$. The following functions on E are defined:

- $N(v_i) \rightarrow V$: defines the direct neighbors of a particular node.
- $R(v_i, v_j) \rightarrow SocialRelations$: defines the relations that link v_i to v_j . For example $R(v_i, v_j) = SiblingOf$.
- $C(v_i, v_j) \rightarrow RelationCategories$: defines the categories of the relation between v_i and v_j . For example $C(v_i, v_j) = Family$

To improve the search, we apply a heuristic optimization of the Breath First Search algorithm [6]. This heuristic is based on the observation that the spatio-temporal context⁶ plays an important role to determine social categories of potential actors. For example, if the viewer is located at work and the time is day, it is more likely that she is with her "Professional relationship" than with people belonging to her "Family relationship". In other words, it is more likely that people that are related to a photo taken in this context (actors or witnesses) are members of the social category "Professional relationship". Another heuristic optimization consists in stopping the search when reaching a path of length 2. Indeed longer social paths aren't meaningful for the viewer. This means that all the

⁶The context capture is out of the main scope of this paper so we don't detail it.

further relations are presented *relatively* to the second path of the viewer.

This case represents clearly an association between the user location and her social relations. Thus, we exploit association rule mining [1] to handle this part of the problem⁷. The rules are built on data (user context data) gathered with a mobile device for a well defined period (2 months in our testing scenarios). These rules associate a physical context of the user (Location and Time) with a social context (present people). After the extraction of rules, the right-hand-side of the rule is reprocessed in order to have a high-level description of the social context (e.g. a list of names are grouped into a category, like Friends or Family). This is achieved using the extended FOAF profile.

In real life, these rules do not apply for all individuals. This is mainly because there are specific locations where the user can be with people belonging to several categories. For this reason, we have implemented a data mining approach to extract rules that associate a situation (a spatio-temporal context) to a social network category. By using this data mining approach, the following constraints are considered:

- 1) Each node v_i has a list of corresponding set of FOAF attributes, like name, address, etc.
- 2) The functions $R(v_i, v_j)$ and $C(v_i, v_j)$ can be associated to each arc $(v_i, v_j) \in E$ and each arc annotated with R must have a corresponding annotation with C .
- 3) There exists a dependence between R and C . For example if R represents *SiblingOf*, then C could represent *Family relationship*. This dependence is shown in Section II-A.

Algorithm 1 shows the pseudo-code of the Social Query Optimization (SQO) algorithm. Given the graph $G(V, E)$ and two nodes v_i and v_j (source and destination), SQO returns a set of potential paths that connect the two nodes (i.e. people). The SocialDimension variable (line 2) holds the different solutions. Our algorithm retrieves paths with a maximum length of 2. We believe that beyond this length, paths are meaningless for viewers (line 6). Thus, if an actor is not a direct acquaintance of a direct acquaintance of the viewer, the SocialDimension becomes too long and difficult to follow. This principle is followed by many social network sites such as Facebook which stops at this level also for friends recommendations. We agree that in this case some actors may remain with empty SocialDimension, but this heuristic keeps the algorithm fast. If we try to retrieve longer paths the complexity of the algorithm will increase and it may become useless for a real-time adaptation scenario.

We enqueue the first and second level descendants of the state whose category belongs to the category set returned by the heuristic function GetConcernedCategory (line 8). This function returns the potentially concerned relationship categories according to time, space or situation constraints. For example, if the situation is "working", the concerned

⁷The objective in this paper is not to detail the usage of association rules. Thus, we only discuss briefly this aspect of the solution here.

Algorithm 1 Social Query Optimization (SQO) design

```

1: Input:  $G(V, E)$ , source node  $v_i$ , destination node  $v_D$ 
2: Output: SocialDimension List of List
3: NodesCovered List, ParentState List,  $q$  queue,
   ConcernedCategory List
4:  $q \leftarrow \emptyset$ ,  $state \leftarrow v_i$ ,  $j \leftarrow 0$ , ParentsState  $\leftarrow \emptyset$ , NodesCovered
    $\leftarrow \emptyset$ 
5: repeat
6:   if ( $v_i \in N(state)$  OR  $state = v_i$ ) then
7:     ConcernedCategory  $\leftarrow$  GetConcernedCategory(state)
8:     for all  $Cat \in$  ConcernedCategory do
9:       Enqueue  $v$  in  $q$  such that  $v \notin q$  and  $\exists C(v, state) = Cat$ 
10:    end for
11:    end if
12:    Add state to NodesCovered
13:     $state \leftarrow q.front()$ 
14:     $q.dequeue()$ 
15:  until  $state = v_j$  OR  $q.empty()$ 
16:  if  $state \neq v_j$  then
17:    return  $v_j$ 
18:  end if
19:  ParentState  $\leftarrow (v, v \in N(state)$  AND  $v \in NodesCovered)$ 
20:  for all  $v \in ParentState$  do
21:    repeat
22:      Add state to SocialDimension[ $j$ ]
23:      if  $v_i \in N(state)$  then
24:        Add 'My' to SocialDimension[ $j$ ]
25:        Add  $R(state, v_i)$  to SocialDimension[ $j$ ]
26:         $state = v_i$ 
27:      else
28:        Add  $R(state, v)$  to SocialDimension[ $j$ ]
29:         $state = v$ 
30:      end if
31:    until  $state = v_i$ 
32:    Increment  $j$ 
33:     $state = v_j$ 
34:  end for
35: return SocialDimension

```

category is “Professional relationship”. Considering only paths of length ≤ 2 and filtering the social categories enables us to reduce the complexity of the BFS algorithm. The complexity is detailed with the experimental results in a further section. If the viewer doesn’t know any actor, the algorithm will only return the destination node v_j (line 17). After retrieving the path between v_i and v_j , the SQO algorithm constructs the *SocialDimension* of v_j for v_i . This *SocialDimension* can contain several paths. For example, Steven may be the *brother-in-law of my colleague David*, as he can be the *mentor of my friend Peter*. An example of the different levels in the social graph are represented in Figure 2.

IV. IMPLEMENTATION AND EXPERIMENTAL RESULTS

In this section we will discuss two main aspects: (i) the system architecture which supports our proposal, and (ii) an experimental study which describes results of the evaluation of the optimized search. It is important to stress here that since the annotation task is directly related to the user appreciation, it is actually very hard to provide concrete results about it at this stage. In fact, we are confronted to the common difficulties in the evaluation of such processes: users’ subjectivity, statistical significance of the obtained results (since the participant set is very small), etc. This part is under consideration and

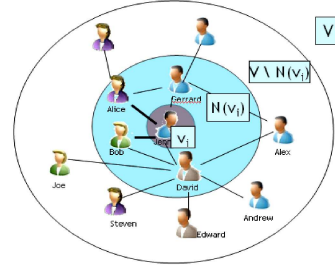


Fig. 2. Different levels in the social graph. V_i represents Jennifer in the graph. $N(V_i)$ represents the set of direct neighbors of V_i . $N \setminus V_i$ represents the set of vertices that are neighbors of $N(V_i)$. Finally, V is the set of vertices in the social graph.

will not be described in this paper. Instead, we discuss more concrete and quantifiable results, related to execution times.

A. System Architecture

Figure 3 shows the general architecture of the system. The system assumes three main layers: (i) back-end social layer, (ii) advanced social annotation engine, and (iii) front-end interactions layer. The first layer (from the bottom to the top in Figure 3) is responsible of managing the interactions between our annotation system and the existing social network sites. This layer is important in the overall architecture of our system since this system is expected to complement the existing social network sites and not to replace them. The front-end layer as for it is dedicated to manage the interactions of the user by the system. This includes the introduction of queries by the user (in the form of a selection of a photo), and the results showing (in the form of extended and adapted annotations to the specific user). This layer is composed of three main components:

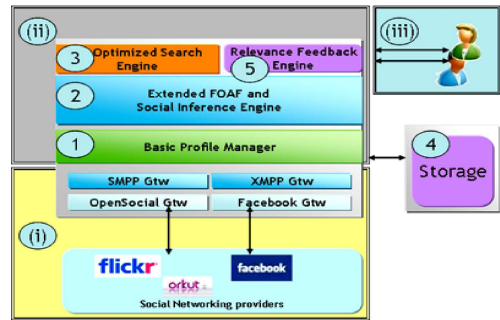


Fig. 3. Overview of the system architecture

1) *Basic Profile Manager*: This component is responsible for maintaining and managing the profiles as they are defined in the different social network sites. This component is composed of different gateways that enable our system to communicate with the different social network sites.

2) *Extended FOAF and Social inference engine*: Relations between basic profiles are extended as we have discussed it before. This engine helps to maintain a meaningful link

between the basic profiles by capturing the semantic relations that can exist between people. The extension of FOAF, as described in the previous sections, is applied at this stage. The correspondences between the basic profiles and the FOAF extended definitions are operated off-line. This component is also responsible of extracting (i.e. inferring) a meaningful semantic description according to the viewer’s profile and role. It exploits both the basic profiles as well as the extended FOAF definitions of the relationships. This task operates on-line.

3) *Optimized search engine*: This engine implements the described algorithm in Section III to improve the search.

4) *Data Storage Engine*: This engine supports a relational database that stores the captured context of a photo. The context data associated to a photo, containing Location, Time, as well as the identifiers of Actors and Witnesses is captured by a client application and uploaded to the database⁸.

5) *Relevance Feedback Engine*: This engine supports the tag suggestion improvement, using widgets that appear next to each Social Dimension tag. These widgets are: Accept (when the suggestion is meaningful), Reject (when the suggestion is not meaningful and the viewer wants a re-computation) or Edit (when no suggestion has been found and thus the user can manually describe the social dimension). The use of the system is not conditioned by the use of these widgets. In fact, the user may not give her feedback. In this case, the system doesn’t integrate this information in the next queries.

B. Preliminary Results

The objective of our experiments is to compare the proposed algorithm (SQO) with the classical Breadth First Search (BFS) algorithm [6]. The two algorithms are implemented using Java on a machine with 2.10 GHz CPU and 2 GB RAM. We have measured the response time with respect to the data size (i.e., the number of RDF triples). We evaluate the performance of the proposed algorithm with respect to the size of the data (i.e. the number of RDF triples). We considered a dataset with ≈ 1 million of RDF triples. To compare the performances of our algorithm, we considered the classical Breadth First Search (BFS) as a reference. The experimental protocol is as follows: we vary the size of the FOAF graph from 153 to 1,060,683 million triples and we recover the execution times of the two algorithms (BFS and SQO). The obtained results are reported on Figure 4 where the execution time is expressed in milliseconds.

Figure 4 clearly shows: (i) the important gain that can be obtained using the proposed optimization compared to the classical algorithm and (ii) the difference in the behavior between the two approaches. More specifically, execution times of the BFS algorithm increased from 1594 s with 154 triples to 16429 minutes with 1 millions triples. More importantly, the curve in the Figure 4 shows an exponential behavior of the Breadth First Search algorithm.

The time and space complexity of the BFS is $O(b^m)$ (where b stands to the branching factor and m to the maximum path

⁸The different techniques of capturing these data with a mobile device are not detailed in this paper.

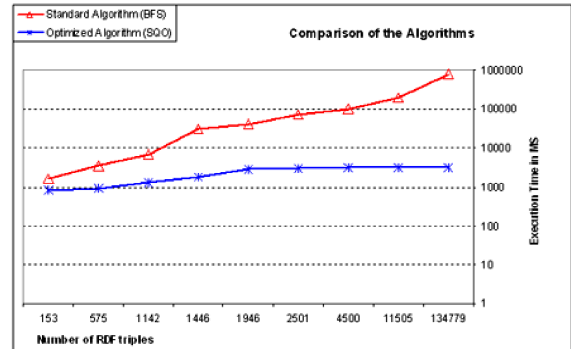


Fig. 4. Comparison of the two algorithms

length) because every node in the graph $G(V, E)$ is examined and the whole frontier must be stored in memory. From the other hand, for the SQO algorithm the execution time does not vary much in a large scale (e.g., for ≈ 1 millions triples it still as 4 seconds).

This is visible even if the complexity is $O(b^2)$ with $b \ll n - 1$. If $b \rightarrow \lim_{n \rightarrow \infty} b$, then the complexity of $SQO \rightarrow O(n)$. The curve in Figure 4 shows that the behavior of the SQO algorithm does not vary much, meaning that our algorithm is faster. This is very encouraging since we can tackle a very large datasets. As an example, if we consider a dataset with $n = 10^6$ nodes, the algorithm can theoretically process it in about 4 seconds.

V. RELATED WORK

Tagging has emerged as a popular means to annotate on-line objects such as bookmarks, photos, and videos. Tags vary in semantic meaning and can describe different aspects of a media object. Tags describe the content of a media as well as locations, dates, peoples and other associated meta-data. Tagging content is now a natural capability of social media services, like Flickr, Youtube, Del.icio.us or social networking sites (SNS), like Facebook. Tagging can take different forms: automatic, when the system automatically assigns keywords to the image; semi-automatic, when the system assigns tags in the form of suggestions and the user accepts/rejects the result (relevance feedback) or manual, when the user enters a freely selected keyword. There are a number of research prototypes that address tagging in photos in automatic or semi-automatic manner. To achieve this perspective, they use data in the captured context of the user (location, environment, time, users around). CONFOTO [12] is a system with the capability of semantic tagging and navigation in photos that are related to conferences. PhotoCompas [14] uses the moment of the photo shot and location for the high-level interpretation of the context. The system suggests the identity of persons by comparing the context of the photo with the previously taken in a similar context. Zonetag [9], a prototype system for Motorola and Nokia smart phones, allows uploading photos to Flickr with semantic tags. The system uses contextual

information such as the location and the time of the shot to suggest tags. The suggestion is based on the principle that in a similar context there is a higher chance to be with the same people and to make the same activity. This work captures the social context of a photo with Bluetooth (actors and witnesses) and annotates the photo with this information.

The difference between our approach and these approaches is that we enrich the identity tags of photos using social relationships which are derived from the profiles of persons. In addition, the derived annotations are adapted to the current user profile making the annotation more dynamic and auto-adaptable. To the best of our knowledge, this is the first work that considers the dynamic and auto-adaptability of the tags.

Image tagging is known in other research areas as image annotation. Image annotation is investigated in different fields like data mining, databases, and statistics. Several approaches have been proposed to tackle this problem. Most of the work is based on a coupling of image processing techniques, e.g. color and texture, and data analysis techniques to provide the most suitable annotation to an image [4]. Clustering is heavily used to attach a text to images [11]. With these methods, it is possible to predict the label of a new image by calculating some probabilities. Minka and Picard [13] proposed a semi-automatic image annotation system which allows users to choose the area to be annotated in the image. On the other hand, probabilistic models such as Cross Media Relevance model [5] and Latent Semantic Analysis [10] were also proposed. Jia and Wang [7] use a two-dimensional hidden markov chains to annotate images. All these approaches rely on some intelligent way to associate a tag to an image after a heavy processing. The result of the process is a static list of keywords that are associated to each image. Our work doesn't focus on low-level image processing for annotation. Instead, it relies mainly on the users, i.e. manual annotation (also called tagging), as it's the case currently on the web.

VI. CONCLUSION AND FUTURE WORK

Image annotation is currently performed in a static way, i.e. we associate static tags to objects or people on the photo. These tags are thus presented in the same form to any person regardless if they can be meaningful for her or not. We have then tackled in this paper the problem of using social relations to provide a user with more meaningful annotations that help her to situate unknown persons according to potentially known ones in a photo, i.e. directly connected in the social network. First we have proposed a new annotation model that considers the social interactions in a social network. This helps in modeling additional information to characterize the relation that can exist between people. Then, the FOAF vocabulary has been extended to consider more relationships. This was necessary in order to offer the user with more meaningful annotation. The last contribution is an optimized search algorithm that improves the search performances in a semantic, i.e. FOAF, database. Furthermore, we have described a system architecture that supports the proposals of this paper. Finally, preliminary results have been discussed showing the

interest of the proposed search algorithm and the important gains compared to the classical one.

As a future work, we first plan to perform an intensive evaluation of the proposed approach to consider possible problems or improvements. The idea is to invite real users to test our application and perform a qualitative evaluation. A second important future work is to extend the proposed approach in the context of mobility. The idea is to capture the current user context (i.e. location and time) according to his communication means, and use this information to better exploit the annotations. We are currently investigating a data mining approach to handle this part and a preliminary discussion has been already started in this paper. Another possible research direction is to improve the optimization algorithm by considering the social proximity. At the present, we don't define any metric to measure the proximity between people considering the number of times that a person appears as actor in social media. Finally, in this paper we have considered only the case of a viewer. It would be interesting to investigate how to consider the different people that are included in the whole photo sharing process (e.g. Actor, Witness, etc.).

REFERENCES

- [1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *VLDB'94*, pages 487–499. Morgan Kaufmann, 1994.
- [2] K. Barnard and D. Forsyth. Learning the semantics of words and pictures. In *International Conference on Computer Vision*, pages II:408–415, 2001.
- [3] D. Brickley and L. Miller. The Friend Of A Friend (FOAF) vocabulary specification - <http://xmlns.com/foaf/spec/>, November 2007.
- [4] H. Hacid. Neighborhood graphs for semi-automatic annotation of large image databases. In *MMM (1)*, volume 4351 of *Lecture Notes in Computer Science*, pages 586–595. Springer, 2007.
- [5] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. *SIGIR*, 2003.
- [6] D. E. Knuth. *The art of computer programming, volume 1 (3rd ed.): fundamental algorithms*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 1997.
- [7] J. Li and J. Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1075–1088, 2003.
- [8] P. Mika. Social networks and the semantic web. In *Web Intelligence*, pages 285–291, 2004.
- [9] F. Monaghan and D. O'Sullivan. Automating photo annotation using services and ontologies. In *MDM '06*, page 79, Washington, DC, USA, 2006. IEEE Computer Society.
- [10] F. Monay and D. Gatica-Perez. On image auto-annotation with latent space models. In *ACM Multimedia*, pages 275–278. ACM, 2003.
- [11] M. Naaman, Y. J. Song, A. Paepcke, and H. Garcia-Molina. Automatic organization for digital photographs with geographic coordinates. In *Proceedings of the Fourth ACM/IEEE Joint Conference on Digital Libraries*, pages 53–62, Tucson, Arizona, June 2004. ACM Press.
- [12] B. Nowack. Confoto: A semantic browsing and annotation service for conference photos. In *International Semantic Web Conference*, pages 1067–1070, 2005.
- [13] R. Picard and T. P. Minka. Vision texture for annotation. Technical Report 302, MIT Media Laboratory Perceptual Computing Section, 1995.
- [14] B. Sigurbjörnsson and R. van Zwol. Flickr tag recommendation based on collective knowledge. In *WWW '08*, pages 327–336, New York, NY, USA, 2008. ACM.
- [15] J. Stan, E. Egyed-Zsigmond, A. Joly, and P. Maret. A User Profile Ontology For Situation-Aware Social Networking. In *3rd Workshop on Artificial Intelligence Techniques for Ambient Intelligence (AITAmI2008)*, July 2008.