

Proceedings of the International Workshop on

User Profile Data on the Social Semantic Web (UWeb)

co-located with 8th Extended Semantic Web Conference
May 30, 2011, Heraklion, Crete, Greece

Editors: Fabian Abel¹, Qi Gao¹, Eelco Herder²,
Geert-Jan Houben¹, Daniel Olmedilla³, Alexandre
Passant⁴

¹ Web Information Systems, Delft University of Technology
Mekelweg 4, 2628 Delft, the Netherlands
{f.abel,q.gao,g.j.p.m.houben,k.tao}@tudelft.nl

² L3S Research Center, Leibniz University Hannover, Germany
herder@l3s.de

³ Telefonica R&D, Spain
danieloc@tid.es

⁴ Digital Enterprise Research Institute, National University of Ireland, Ireland
alexandre.passant@deri.org

Table of Contents

Preface	i
<i>F. Abel, Q. Gao, E. Herder, G.-J. Houben, D. Olmedilla, A. Passant</i>	
Ambiguity in Tagging and the Community Effect in Searching Relevant Resources in Folksonomies	ii
<i>S. Beldjoudi, H. Seridi, C. Faron-Zucker</i>	
Sensing Presence (PreSense) Ontology User Modelling in the Semantic Sensor Web	iii
<i>A.E. Cano, A.-S. Dadzie, V.S. Uren, F. Ciravegna</i>	
Semantic Social Profile – a Semantic Boost for the Social Information in MediaWiki	iv
<i>Y. Katkov, D. Pokoptsev</i>	
Forecasting Audience Increase on YouTube	v
<i>M. Rowe</i>	
TUMS: Twitter-based User Modeling Service	vi
<i>K. Tao, F. Abel, Q. Gao, G.-J. Houben</i>	

Preface

Social Web sites, such as Facebook, YouTube, Delicious, Flickr and Wikipedia, and numerous other Web applications, such as Google and Amazon, rely on implicitly or explicitly collected data about their users and their activities to provide personalized content and services. As these applications become more and more connected, a major challenge is to allow various applications to exchange, reuse, and integrate user data from different sources. The amount of user data available on the Web is tremendously growing and reasoning on such heterogeneous user data distributed across the Web, i.e. exploiting user data on the Web, is a non-trivial problem that poses several challenges to the Semantic Web community.

While the linking of user data provides advantages for recommendation and personalization, it also raises questions related to provenance, trust and privacy: how does one know that the data gathered from several sources can be trusted, and how can one avoid that sensitive personal data is disclosed to certain services or used to infer and expose sensitive information? Trust and privacy, and associated policies, may therefore impact reasoning on user data.

The UWeb workshop⁵ discusses research on user data in the Social Semantic Web from both angles:

- techniques and applications for linking, reusing and reasoning about user data, as well as
- trust and privacy techniques and their impact on society.

UWeb joins the SPOT⁶ and LUPAS⁷ workshops from ESWC 2010 and provides a forum for academic and industrial researchers and practitioners in the fields of the Social and Semantic Web, trust, privacy, user modeling, and personalization to discuss theoretical and practical knowledge, open research issues, applications, and experiences.

We thank the members of the Program Committee of UWeb 2011 for their support and reviews. Furthermore, we are grateful to all authors who submitted articles to UWeb and contributed with their works to the UWeb workshop.

Fabian Abel	Qi Gao
Eelco Herder	Geert-Jan Houben
Daniel Olmedilla	Alexandre Passant

UWeb Organizing Committee, May 2011

⁵ <http://wis.ewi.tudelft.nl/uweb2011/>

⁶ <http://spot.semanticweb.org/>

⁷ <http://www.personal-reader.de/lupas/>

Ambiguity in Tagging and the Community Effect in
Researching Relevant Resources in Folksonomies

S. Beldjoudi, H. Seridi, C. Faron-Zucker

Ambiguity in Tagging and the Community Effect in Researching Relevant Resources in Folksonomies

Samia Beldjoudi¹, Hassina Seridi¹, Catherine Faron-Zucker²,

¹ Laboratory of Electronic Document Management LabGED, Badji Mokhtar University,
Annaba, Algeria

² I3S, University of Nice Sophia Antipolis - CNRS, France

¹ {seridi,Beldjoudi}@labged.net

² catherine.faron-zucker@unice.fr

Abstract. In this paper we present an original approach to improve the search of web resources tagged with folksonomies. We argue that the semantics in folksonomies can be extracted from the force of the community effect and the social interactions between the members of a community. Our approach called SSF for Semantic Social Folksonomy tackle the problem of ambiguity between tags by considering user profiles and social interactions in folksonomies to reduce tags ambiguity.

Keywords: Folksonomies, Web 2.0, Semantic Web, Social Interactions, Tag Ambiguity.

1 Introduction

Before the revolution of Web 2.0, the creation of keywords or metadata associated to web resources was realized by the professionals of a domain in order to clearly specify its data. Therefore the cost in time and effort was too exaggerated. With the advent of the so called Social Web, users are allowed to create their own keywords which are known as tags and to tag web resources. The term *folksonomy* has appeared on the web to designate a new technology based on the triple (the *User*, the *Tag* and the *Resource*), where the user is an actor in the system, the tag designates the keyword proposed by the user and the resource indicates the object that will be annotated by the user with the tag. According to Wikipedia a folksonomy is a system of classification derived from the practice and method of collaboratively creating and managing tags to annotate and categorize content. For instance, delicious¹ is one of the most famous such websites, based on folksonomies, which allow users to publish, tag and share resources. It represents an effective way to collaboratively manage bookmarks: it enables each user to add sites to his personal collection of links, to categorize those sites with tags, and to share them with other users. Folksonomies are used to tag any kind of multimedia web resources: Flickr² is a photo management and

¹ <http://del.icio.us>

² <http://www.flickr.com>

sharing web application; youtube³ and dailymotion⁴ are dedicated to the sharing of videos; myspace⁵ and Odeo⁶ to the sharing of music files.

These social practices continue every day to gain popularity. However, the use of folksonomies for finding information on the web poses some problems, among which the problem of tags ambiguity: a tag can designate several concepts, i.e. can have several meanings. For example when a user employs the tag "orange" to annotate a resource, the system will not understand whether he is talking about the "color" orange or the "fruit" orange. A related problem comes from the variations in writing the same concept. For instance, "cat" and "chat" both means the same concept (animal) in English and in French, but when a user searches resources annotated with the tag "cat", the system will not offer him those tagged with the word "chat" because it cannot understand that the tag "cat" is equivalent to the tag "chat". In addition, the tags that are freely chosen in these systems are likely to contain spelling errors and therefore make the retrieval of resources more doubtful than the metadata recovering from a lexicon examined by the professionals of information.

In this paper we present an original approach to improve the search of web resources tagged with folksonomies. We argue that the semantics in folksonomies can be extracted from the force of the community effect and the social interactions between the members of a community. Our approach called SSF for Semantic Social Folksonomy tackle the problem of tags ambiguity by considering user profiles and social interactions in folksonomies. More precisely, we intend to propose each user resources relevant to his profile in the folksonomy even when he uses ambiguous tags. The key of our approach is the similarities between the community's members. Our paper is organized as follows: Section 2 presents an overview of the major contributions in this field. The design of our approach is presented in Section 3. In Section 4 the experimental phase is in order to measure the performance of our approach and also the obtained results are discussed. We conclude in Section 5.

2 Related Works

Despite the relative newness of folksonomies, there already are a lot of works in this domain; each one trying to resolve a specific problem. Most of these contributions are balanced between searching semantic relationships among folksonomy's terms, and helping users choosing appropriate terms when annotating their resources for increasing the weights associated to each tag. In this paper, we focus our state on the art on both the works relative to the reduction of the ambiguity of terms in folksonomies and those on the extraction of semantic links between terms in a folksonomy.

Mika [8] proposed to extend the traditional bipartite model of ontologies to a tripartite one: that of folksonomies, where instances are keywords used by the actors of the system in order to annotate web resources. In this article, Mika focuses on

³ <http://www.youtube.com>

⁴ <http://www.dailymotion.com>

⁵ <http://www.myspace.com>

⁶ <http://odeo.com>

social network analysis in order to extract lightweight ontologies and, the exploitation of the strength of these ontologies in order to explicit semantics between the terms used by the users (actors).

In another work, Gruber [4] argued that there is no contrast between ontologies and folksonomies, and therefore recommended to build an "ontology of folksonomy". According to Gruber, the problem of the lack of semantic links between terms in folksonomies can be easily resolved by representing folksonomies by ontologies.

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

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

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

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

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

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

We also investigated the works relative to the recommendation of relevant resources. De Meo et al. [3] proposed to recommend a set of resources to enrich user profiles, a user profile being represented by the list of tags involved in his query. They

expand queries to recommend resources to users performing a keyword-based search; in order to enhance their profiles. A user query is enriched with tags discovered through the exploration of the two graphs TRG (Tag Resource Graph) and TUG (Tag User Graph). According to the authors, this enrichment improves both the strategy of recommender systems and that of collaborative filtering and content-based filtering systems.

To sum up, most of the works relative to folksonomies aim to bring together ontologies and folksonomies as a solution to the tags' ambiguity problem and that of the lack of semantic links between tags. The approaches summerized in this section showed that the social nature of resources sharing is not in contradiction with the possibilities offered by ontology-based systems. But the rigidity which characterizes ontologies and the need of an expert who must control and organize links between terms as in [4] seems a little cumbersome and too much expensive. Even the structures extracted automatically as in [8] still suffer from the ambiguity of concepts. Regarding the work of [11], the use of semantic web ontologies for extracting relationships between terms is not sufficient, because the semantic web does not include enough specific domain ontologies and this will push the problem further. Also the expertise of users which was introduced in [5] is characterized by the complexity of its exploitation when we try as much as possible to avoid a cognitive overload, to limit the necessary effort for the formalization of this expertise. Based on these observations, we propose an original approach which avoids the explicit use of ontologies.

3 The Semantic Social Folksonomy Approach

Our approach is to introduce both semantics and social aspects in folksonomies in order to enable the user to retrieve relevant web resources tagged by the members of his community and close to his preferences. For this purpose, we propose to measure the similarity between the members of a community and to compare their preferences. Our goal is to provide a flexible method to reduce the problem of tags ambiguity and the lack of semantic relationships between terms in folksonomies, based on the similarities between users.

The innovative aspect of our approach is that it enables to surmount the lack of semantic relationships between tags and to suggest the user web resources close to his preferences when he performs a tag-based search. We assign to each suggested resource a degree of recommendation — '*strongly recommended*', '*recommended*' or '*weakly recommended*' — depending on the degree of similarity among users.

Let us note that our approach comes with a new point of view on folksonomies. The 'display's instability' of folksonomies is a corollary to the fact that the results of a search procedure vary depending on the interests of each user. Moreover we answer the problem of ambiguity in tagging by retrieving resources depending on their similarity with user profiles.

Example 1. o better explain our approach, let us consider the following example (Table 1) where three users are each represented by a list of tags:

Table 1: A set of users with their tags

<i>Users</i>	<i>Tags</i>
U_1	computer, java, folksonomies, programming.
U_2	apple, fruit, strawberry, kitchen.
U_3	computer, apple, software.

Let us suppose that user U_1 wants to retrieve resources relative to the word (i.e. the tag) 'apple'. In the current folksonomies, the output result will contain all the resources tagged with the 'apple' i.e. those relative to food and computer, despite the fact that it is clear for a human reading U_1 's tags, that his preferences are relative to computer and not to food. This can be summarized in the following table (Table 2):

Table 2: The key points of our approach: problem, assumption solution

<i>Problem</i>	Lack of semantic links between tags leading to a problem of ambiguity: a tag can refer to several concepts, i.e., a tag can have several meanings.
<i>Hypothesis</i>	Two resources tagged with the same term (tag) are similar, if they are used by users who share similar interests or similar tags ⁷
<i>Solution</i>	Measure the similarity between users, to specify those who have similar preferences.

To make the system flexible, we propose to make it interact with the user to accept or reject the retrieved resources.

To avoid the "cold start" problem which generally results from a lack of data required by the system in order to make good quality recommendation, we propose to measure the similarity between resources when there are no similar users.

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

Table 3: A second set of users with their tags

<i>Users</i>	<i>Tags</i>
U_1	computer, java, folksonomies, programming.
U_2	apple, fruit, strawberry, kitchen.
U_3	computer, apple, software.
U_4	apple.

⁷ Of course the degree of similarity between two resources will differ according to the number of the common tags between them.

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

On the contrary the resources corresponding to the tag 'apple' which is used by user U_2 will be given to U_1 with a percentage 'low' level of recommendation because U_1 and U_3 do not share the same interests.

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

3.1 Formal description of the approach SSF

In this section we explain how the folksonomy is represented and how the community members, resources, and tags are exposed, how the relationships between these three elements are symbolized and how the system can recommend the best resources to each user according to his profile in the folksonomy.

Formally, a folksonomy is a tuple $F = \langle U, T, R, A \rangle$ where U, T, R represent respectively the set of users, tags and resources. A represents the relationship between the three preceding elements (i.e. $A \subseteq U \times T \times R$). In this approach a folksonomy is considered as a tripartite model where the instances are a web resources associated by user to a list of tags. So we can extract a tripartite hypergraph with three types of vertices which are tags, users and resources. However working with this kind of graphs is in general not easy to understand. Thus, we have preferred to extract three bipartite graphs with three social networks 'tag_user', 'tag_ressource' and 'user_ressource' from the folksonomy as in [7]. These three graphs are represented as follow:

-**TU** = **<vertices, edges>** noting that:

- Vertices = $T \cup U$ where $T = \{t_1, t_2, \dots, t_n\}$ is the set of tags and $U = \{u_1, u_2, \dots, u_m\}$ is the set of users.
- Edges = $\{t, u\}$ represents the set of links between the different vertices.

-**TR** = **<vertices, edges>** noting that:

- Vertices = $T \cup R$ where $T = \{t_1, t_2, \dots, t_n\}$ is the set of tags and $R = \{r_1, r_2, \dots, r_k\}$ is the set of resources.
- Edges = $\{t, r\}$ represents the set of links between the different vertices.

-**UR** = **<vertices, edges>** noting that:

- Vertices = $U \cup R$ where $U = \{u_1, u_2, \dots, u_m\}$ is the set of users and $R = \{r_1, r_2, \dots, r_k\}$ is the set of resources.
- Edges = $\{u, r\}$ represents the set of links between the different vertices.

For better representation, the three graphs formalization is represented by a matrix form as follows:

$$\begin{aligned}
-\mathbf{TU} &= [\mathbf{X}_{ij}] \text{ where } X_{ij} = \begin{cases} 1 & \text{if there is an edge between the tag } i \text{ and the user } j \\ 0 & \text{otherwise} \end{cases} \\
-\mathbf{TR} &= [\mathbf{Y}_{ij}] \text{ where } Y_{ij} = \begin{cases} 1 & \text{if there is an edge between the tag } i \text{ and the resource } j \\ 0 & \text{otherwise} \end{cases} \\
-\mathbf{UR} &= [\mathbf{Z}_{ij}] \text{ where } Z_{ij} = \begin{cases} 1 & \text{if there is an edge between the user } i \text{ and the resource } j \\ 0 & \text{otherwise} \end{cases}
\end{aligned}$$

Three binary matrices have gotten and each one indicates possible links or correlations that can exist between every two elements selected from the folksonomy (Tags, Users); (Tags, Resources) or (Users, Resources).

- **Example 3.** The following three tables 4, 5 and 6 shown an example that will illustrate the relations between the different elements in a folksonomy:

Table 4: A matrix ‘tag_user’ (TU)

Tags \ Users	U1	U2	U3	U4	U5
T1= computer	1	1	0	0	0
T2= kitchen	0	0	1	0	1
T3= programming	1	0	0	1	0
T4= apple	0	1	1	0	0

Table 5 : A matrix ‘tag_resource’ (TR)

Tags \ Resources	R1	R2	R3	R4	R5
T1= computer	1	1	0	0	0
T2= kitchen	0	0	0	1	1
T3= programming	1	1	0	0	0
T4= apple	0	0	1	0	1

Table 6: A matrix ‘user_resource’ (UR)

Users \ Resources	R1	R2	R3	R4	R5
U1	1	1	0	0	0
U2	0	1	1	0	0
U3	0	0	0	1	1
U4	1	0	0	0	0
U5	0	0	0	0	1

Noting that; the profile of each user has been represented by the list of his tags or his resources in the folksonomy.

- **Example 4.** From the Example 3 (Table 4), the following profiles are found:

Table 7: A table shows the users' profiles in the form of tags

Users Profiles	Tags
Profile U_1	computer, programming.
Profile U_2	computer, apple.
Profile U_3	cooking, apple.
Profile U_4	programming.
Profile U_5	kitchen.

3.2 The similarities between users

For the similarity between users computing, we suggest to use a measure that allows representing each user by a vector v_i designates a series of binary numbers defining the set of his tags or his resources. Thus, calculating the similarity between two users, for example u_1 and u_2 , is using the cosines of the angle between their associated vectors v_1 and v_2 as shown in the formula (1):

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

The vectors corresponding to each user are computing by using two matrixes: UR x RU and UT x TU must be calculated. Where:

- RU is the transposed matrix of UR .

- UT is the transposed matrix of TU .

3.3 The similarities between resources

When users are not similar, we suggest measuring the similarity degree between resources in order to avoid "cold start" problem. And for that we suggest calculating the matrix **resource** = RT x TR . Where: RT is the transposed matrix of TR .

3.4 The search procedure

In the Figure 1 which is presented below, an activity diagram that will illustrate the search's procedure followed by our approach is introduced:

As it's shown in this Activity Diagram, the Levenshtein distance⁸ is being used when the tag is not found in the folksonomy in order to measure the edit distance between

⁸ By following the example of [10] and [5]

tags in the system, this allows our system to detect spelling variations and so it can offer to user presumably equivalent tags.

Example 5. In Table 8, there are some examples for the Levenshtein distance:

Table 8: Levenshtein Distance for some couples of tags

Tag1	Tag2	Distance of Levenshtein
Oracle	Orakle	0.83
User	Users	0.80
Stic2010	Stic-2010	0.88
Cat	Chat	0.75

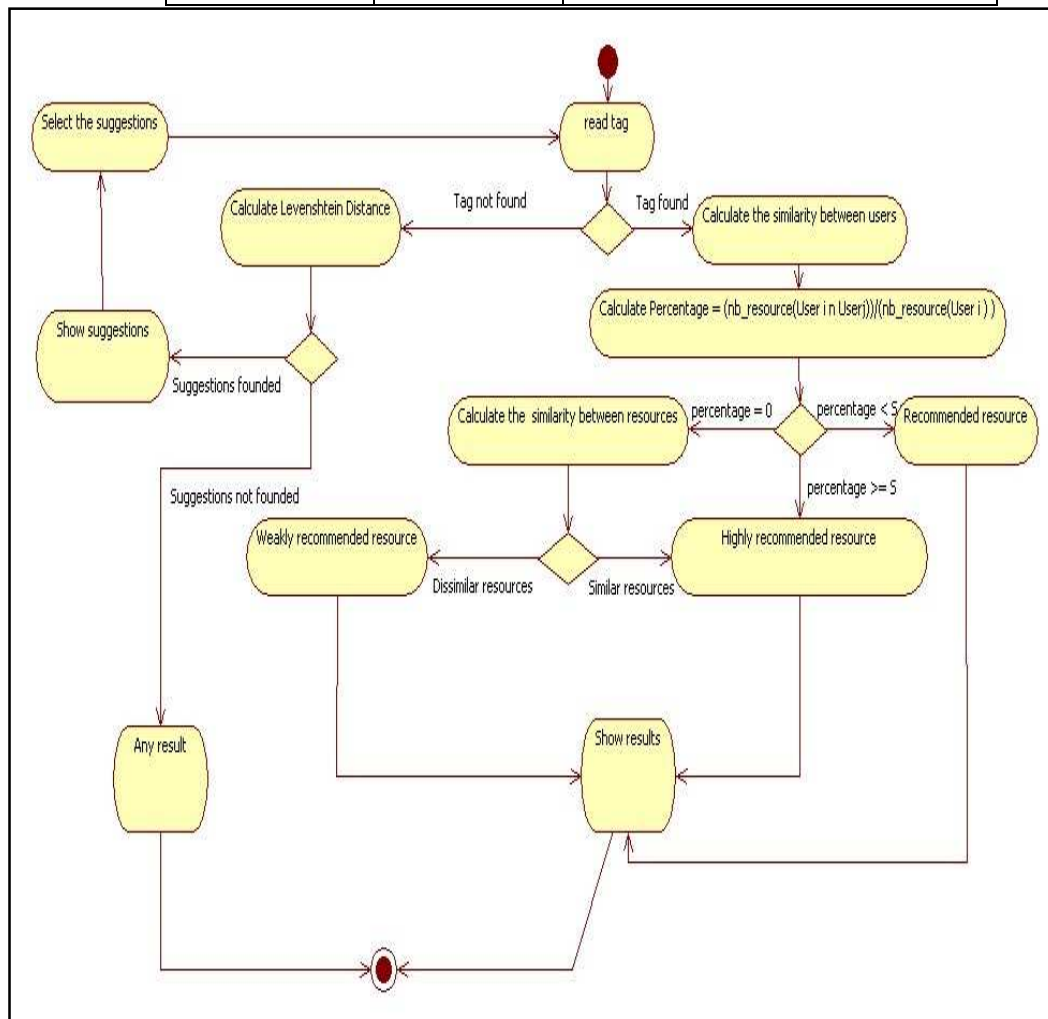


Fig 1: Diagram of Activity for the search procedure

For each resource recommended by the system a factor is proposed to indicate the percentage of its recommendation, i.e. the resource is 'highly recommended', 'recommended' or 'weakly recommended'. To achieve this classification, we have proposed to calculate the ratio between the number of resources used by the user himself (i.e. the one who does the research) and the number of shared resources between him and the others users. It must first select a threshold $S \in [0, 1]$ to schedule the results. Where for each recommended resource, the following steps are performed:

$$1. \text{ Calculate } \frac{nb_resource(User\ i \cap User\ j)}{nb_resource(User\ i)} \quad (2)$$

$$2. \text{ If } \frac{nb_resource(User\ i \cap User\ j)}{nb_resource(User\ i)} \begin{cases} \geq S & \text{then the resource is highly recommended} \\ < S & \text{then the resource is recommended} \\ = 0 & \text{then calculate the similarity between resource} \end{cases}$$

- **Example 6.** In table 9 the symmetric matrix $User = UR \times RU$ is represented:

Table 9: A symmetric matrix $User = UR \times RU$

User User	U1	U2	U3	U4	U5
U1	2	1	0	0	0
U2	1	2	0	1	0
U3	0	0	2	0	1
U4	0	1	0	1	0
U5	0	0	1	0	1

In this matrix; for each user we have a vector as shown in Table 10:

Table 10: The correspondence between each user and his vector

User	Vector
U1	$v_{U1} = (2,1,0,0,0)$
U2	$v_{U2} = (1,2,0,1,0)$
U3	$v_{U3} = (0,0,2,0,1)$
U4	$v_{U4} = (0,1,0,1,0)$
U5	$v_{U5} = (0,0,1,0,1)$

For a threshold $S = 0.5$ for example. Supposing that after the similarity between users computing, the system provides the resource R_3 used by the user U_2 and which is annotated by the tag 'apple' to user U_1 . Noting in Table 8 that the resource number used by user U_1 is equal to 2, and the number of shared resources between the user U_1 and the user U_2 is equal to 1, so:

$$\frac{nb_{resource}(u1 \cap u2)}{nb_{resource}(u1)} = \frac{1}{2} = 0.5$$

Having $\frac{nb_{resource}(u1 \cap u2)}{nb_{resource}(u1)} = S$, and the resource is highly recommended.

4 Experiment

This section describes the used Dataset in the experiment and its treatment's method in order to fully understanding the given results. But first it should be noted that the experiment aim is to satisfy users when a research by keywords is done within folksonomies by offering them pertinent resources even in the presence of ambiguous tags. For this, we followed the steps described below:

4.1 Data Set

The dataset used in the test phase is described in this section followed by some analysis and discussion of the obtained results. The database of the website del.icio.us has been employed in this experiment representing the most used dataset for overall conducted experiments in folksonomies. Noting that we have selected a random set of data (i.e. a random sets of users, tags and resources) to well demonstrate the validity of our proposal. To fully shown the validity of SSF, two classes of users are randomly chosen: the first one contains the users who employed ambiguous tags and the second contains those who don't use these words but who can get them in the future. The users' number is equal to 21. For the set of tags used in this experiment we selected 97 different ones, among them there are some tags which are ambiguous. The used resources number is equal to 92; every resource can have multiple tags, and even multiple users. And in total we have 207 tag assignments. The table 11 summarizes the information about the data sample used in this experiment:

Table 11: The correspondent value for every element in the data sample

	Nb tags	Nb users	Nb resources	Nb tag assignments
Delicious dataset	97	21	92	207

4.2 Data treatment

The aim of this contribution is to generate a flexible technique, in other words, a technique which can be adapted to any situation. For this reason, we tried to automate it by using tools that can considerably reducing the site's administrator effort. Thus, after collecting all used data in the test phase, a tool of social network analysis called

resources that will help a user to satisfy his need for information, basing on its long-term profile. For this, a set of metrics are used in evaluating the proposal:

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

Recall: measures the system's ability to retrieve all relevant resources inherent to a query. It's given by the relevant retrieved resources ratio and all relevant resources in the database.

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

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

Where R : is the number of relevant resources in the entire collection, M : is the number of resources selected by the system and $R+$: is the number of relevant resources selected by the system.

It should be noted that elements of each set (i.e. those of R), M and $R+$ are selected from our data sample as follow:

-The R number for each user is calculated according to its profile in the folksonomy. For example we took the case of user who is identified by this list of tags {java, computers, mac}, in the evaluation we supposed that preferences of this latter are similar to computing sciences field and not to the food when he did a search by the keyword *apple*. All resources that are close to the first domain are considered relevant to this user. Now in order to avoid the case when this user changes his opinion and likes to search resources related to second field (i.e. food) we give him in our proposal this resources at the end of list in order to avoid this kind of problems. In other words the approach SSF takes the probable changes in tastes of each user.

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

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

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

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

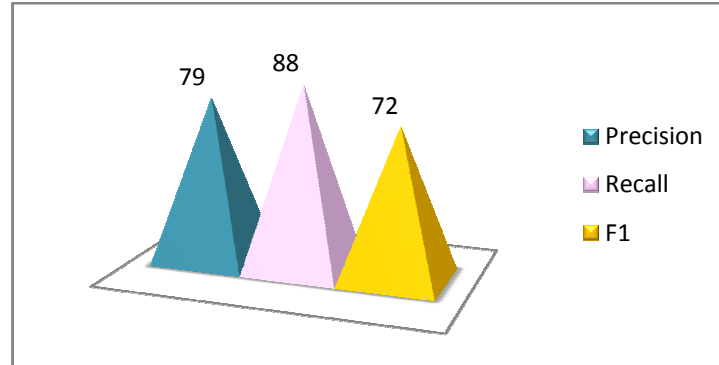


Fig 5: The average of the three metric: precision, recall and F1

4.5 Discussion

In SSF approach, we propose finding ways to equip in a "dynamic" manner the development of the sharing systems which trying to extract semantic in order to allow users to capture in the course of their daily tasks the social dimension of their use of some terms. It appears from this study highlights that can be summarized as follows: The consensus among users who have similar interests for using the same tags or the same resources plays an important role in eliminating the ambiguity problem. Also the increase in the weights of these terms or these resources has an influence on the emergence of semantic even when there are tags that can have several meanings. In fact, in the view of the obtained promising results in the experimental phase, it can be said that the use of analytical tools of social networks and also the collaborative filtering methods, even in the absence of ontologies, can help considerably to automate the nature of applications seeking to extract the semantic in the folksonomies. Indeed the obtained results the used data sample show that the technique SSF is succeeded in distinguishing between ambiguous tags.

Comparing SSF approach with others trying to discuss the problem of tags' ambiguity; for example the Pan's and al work [8], we can conclude that our results are very optimistic especially when we know that the proposed approach is flexible i.e. the result of the search's procedure will be changed according to interests and the profile of each user in contrary to other approach. Also concerning the works that aimed to recommending a set of resources for each user like in the approach cited in [3]. We find that the technique that has been designed doesn't take into account the semantic between terms, in particular it can't distinguish between the ambiguous tags and therefore it can provide to a user resources that can rejected by him because they are not close to his preferences.

5 Conclusion and Future Works

Our work presented in this paper contributes to answer the problem of tags ambiguity when searching for tagged resources. Our approach is based on taking into account social interactions between the members of a community to implicitly extract the semantics of the tags in a folksonomy. We therefore call it a Semantic Social Folksonomy. Our first experiments have shown the efficiency of our approach. However it is too early to conclude – the test phase should first cover the whole del.icio.us database. A short term issue will then be the validation of our approach on other large databases.

In the continuation of our work, we intend to enrich our approach of a Semantic Social Folksonomy by improving semantic network analysis (SNA) metrics.

References

1. Angeletou, S., Sabou, M., Specia, L., Motta, E.: Bridging the Gap Between Folksonomies and the Semantic Web: An Experience Report, Proc. Of ESWC workshop on Bridging the Gap between Semantic Web and Web (2007)
2. Buffa, M., Gandon, F., Ereteo, G., Sander, P. and Fabon, C.: SweetWiki: A semantic Wiki. *J. Web Sem.*, 6(1), 84–97 (2008)
3. De Meo, P., Quattrone, G., Ursino, D.: A query expansion and user profile enrichment approach to improve the performance of recommender systems operating on a folksonomy. *User Model. User-Adapt. Interact.* 20(1): 41-86 (2010)
4. Gruber, T.: Tag Ontology- a way to agree on the semantics of tagging data, <http://tomgruber.org/> (2005)
5. Limpens, F., Gandon, F., Buffa, M.: Sémantique des folksonomies: structuration collaborative et assistée, *IC* (2009)
6. Markines, B., Cattuto, C., Menczer, F., Benz, D., Hotho, A., Stumme, G.: Evaluating similarity measures for emergent semantics of social tagging. *WWW 2009*: 641-650. (2009)
7. Mika, P.: Ontologies are us: A unified model of social networks and semantics, In *ISWC*, volume 3729 of LNCS, p. 522–536: Springer (2005)
8. Pan, J., Taylor, S., Thomas, E.: Reducing Ambiguity in Tagging Systems with Folksonomy Search Expansion, In *Proc. of the 17th International World Wide Web Conference* (2009)
9. Specia, L. Motta, E.: Integrating Folksonomies with the Semantic Web, 4th European Semantic Web Conference (2007)
10. Scott A. Golder, Bernardo A. Huberman: Usage patterns of collaborative tagging systems. *J.Information Science* 32(2): 198-208 (2006)
11. Specia, L. Motta, E.: Integrating Folksonomies with the Semantic Web, 4th European Semantic Web Conference (2007)

Sensing Presence (PreSense) Ontology – User Modelling
in the Semantic Sensor Web

A.E. Cano, A.-S. Dadzie, V.S. Uren, F. Ciravegna

Sensing Presence (PreSense) Ontology – User Modelling in the Semantic Sensor Web

A.E. Cano, A.-S. Dadzie, V.S. Uren, F. Ciravegna

Department of Computer Science, The University of Sheffield,
Sheffield, United Kingdom
`{firstinitial.surname}@dcs.shef.ac.uk`

Abstract. Increasingly, people’s digital identities are attached to, and expressed through, their mobile devices. At the same time digital sensors pervade smart environments in which people are immersed. This paper explores different perspectives in which users’ modelling features can be expressed through the information obtained by their attached personal sensors. We introduce the *PreSense* Ontology, which is designed to assign meaning to sensors’ observations in terms of user modelling features. We believe that the *Sensing Presence (PreSense)* Ontology is a first step toward the integration of user modelling and “smart environments”. In order to motivate our work we present a scenario and demonstrate how the ontology could be applied in order to enable context-sensitive services.

Keywords: linked data streams, semantic sensor web, user modelling, smart objects

1 Introduction

Digital sensors have pervaded the modern world, and increasingly make up the majority of connected devices, in, e.g., intelligent buildings, traffic lights and in particular in mobile devices. These advances have resulted in “smart” environments, marking an evolution in the generation of information, and the interaction between humans, smart and ordinary devices and sensors. Human-computer interaction now extends to everyday objects attached to the end user or located in their changing environment [8]. Users produce data streams through their mobile devices, wearable and implantable micro-sensors (e.g., GPS tracklogs, heart rate monitors). These devices now frequently act as the gateway to cyberspace, which is increasingly becoming an extension of the lives of humans in the real, physical world. Therefore, these can provide information regarding a user’s physical context (e.g. location, physiological state), in addition to their digital environment (e.g. adding new friends to an online social network, tweeting on an evolving event). This leads to a bond between the user and their mobile devices and sensors, in which the latter act as an extension of the user’s identity, providing real-time information that can reveal important user and environmental characteristics.

This provides motivation to explore new techniques for combining current user modelling methods, that depict the digital identity of a given person, with sensor information distributed across the online and physical worlds.

The contributions of this paper are as follows: we explore different perspectives in which the attachment of sensor data to user models can impact the derivation of tailored services that feed into users' interaction with smart objects and environments, by providing real-time contextualisation. We propose the *Sensing Presence* (PreSense) ontology as an approach to modelling the attachment of sensor data streams to a user profile, allowing rich, semantic, real-time change in a user's representation. This enables also the integration of observable user features to the linked data cloud [3].

The paper is structured as follows: in section 2 we present the motivation for our work by discussing different perspectives on the use of sensor data for user modelling; in section 2.3 we introduce a scenario that highlights challenges and benefits that the attachment of sensor data to user profiling presents; section 3 discusses existing ontologies that consider sensor data in user modelling; in section 4 we introduce a set of requirements for modelling the attachment of sensor data streams to a user profile; in section 5 we introduce the Sensing Presence Ontology (PreSense). Section 6 describes the application of the ontology in relation to our scenario; finally, section 7 discusses the potential of this work, plans for evaluation, and concludes the paper.

2 Motivation

The relevance of users to the Sensor Web has been explored from the perspective of users acting as collective sources of information. Goodchild [9] highlights the relevance of the Social Web in Volunteered Geographic Information, where users have created a mesh of global information. Projects like SensorBase¹ and SensorPedia² provide a platform for sharing online sensor information within user communities. However, little attention has been paid to the importance of users' sensors as gateways for personal feature information. This section motivates our work by introducing different perspectives from which users engage with the physical and online worlds through sensor data.

2.1 Mobility in the Digital Society

In the past few years, users' online activities, including web browsing, online shopping, and social web media use [15,20], have served as information sources for user modelling. Further, the emergence of compelling social web platforms (e.g. Facebook³, Twitter⁴) have encouraged users to proactively participate, shaping their online personae and influencing their perception about how they are viewed by others (a.o., [23]).

Social studies on the adaptation of users to online technologies highlight that users appropriate telecommunications technologies in ways that fit their social groups, life stages, sociability and activities [10]. Since mobility has become a central aspect of the digital society, the introduction of location-aware services in social web platforms for mobile devices has received considerable attention from researchers in recent years. Research in this area includes scenarios for emergency response, tracking, navigation,

¹ <http://sensorbase.org>

² <http://www.sensorpedia.com>

³ <http://www.facebook.com>

⁴ <http://twitter.com>

billing and social networking [16,21,27]. Part of the success of these applications is the user's increased dependency on mobile devices; which have become, for some, an indispensable tool. While the use of sensors for registering users' features (e.g. location) has proved to be fundamental in these applications, transient, sensor-based information has, to date, not been considered as an inherent component in user modelling.

2.2 Sensors and Users' Context

Sensors refer not only to physical sensor devices but also to values computed as a result of the composition of indirect or abstract measurements derived from multiple, distributed, often heterogeneous data streams [5,19,24]. Such sensors are usually referred to as *virtual sensors*; they allow the abstraction of data collection away from a fixed set of physical objects. A virtual sensor may define a number of valid sources of information, allowing it to poll for and retrieve information from different sources and at varying levels of granularity.

Following this definition, we consider a *web-based sensor* as an extension of the concept of virtual sensors, in which the measuring computation involves data streams generated from web resources. A web-based actuator may be regarded as a reactive computation that produces a response to a specified event. E.g. NASA Hurricane⁵ on Twitter is a data stream of instantly updated information generated from the continuous monitoring of different devices sensing meteorological conditions for predicting hurricanes and tropical cyclones all over the world.

In the same way, personal data streams may be regarded as gateways reporting relevant information for user modelling. The information embedded in these streams involves different users' context. User context is built on static, stable and dynamic contexts. A user's static and stable contexts represent information about or related to a user that does not, or rarely, changes in time, e.g. the relation between a user and their hometown or work place. A user's dynamic context, in contrast, reflects highly changing information, which is often influenced by the environment in which a user is immersed; this includes, e.g., changes in position, anxiety levels while in a traffic jam.

Advances in intelligent, context-aware systems promote a vision of increasingly autonomous and ubiquitous applications that act on proactive knowledge to provide tailored services to individuals. These smart systems must not only support users in static, pre-defined environments, but also adapt to users' changing context and evolving goals. However, the integration of user context and the user's immersed environmental context, taking into account tempo-spatial restrictions, still requires research. We present next a scenario highlighting the role of sensor data streams in a user profile.

2.3 Scenario

Imagine Alice, a *Doctor* working at a public hospital, and Bob, a *Patient* suffering from Type II diabetes and obesity. Bob's treatment combines regular insulin injections with a diet plan. His nutritionist works with Alice to monitor how well he follows the plan, his physical activity and the impact both have on his overall health. Periodic reviews will

⁵ NASAHurricane: <http://twitter.com/NASAHurricane>

take this information into account in updating his treatment. Alice must also monitor Bob's blood glucose levels, to determine the suitability of both his diet and medication.

This scenario requires Bob to wear multiple sensors that communicate with Alice over a network. More precisely, it requires the attachment of sensor information to Bob's personal attributes. Given the emergence of sensor-enabled mobile devices, we can imagine that Bob owns a device that connects to the Internet and monitors his location and health [4,7]. In this context, Alice accesses, in real-time, the data generated by Bob's sensors. Should Bob's blood sugar reach a dangerous level, Alice must be able to dispatch emergency assistance to Bob in the most efficient way. Information on his diet is not critical, so is only uploaded periodically.

Let us consider a weekday when Bob is returning to his office after inspecting a construction site with a client. His sensors have recorded higher than usual physical activity and that he missed his usual mid-morning snack. His smartphone warns him of the danger of his blood glucose levels dropping too low. Since it is close to lunchtime his nutrition monitor (NutrApp) polls for suitable eateries between his current location and his office (see Fig. 1A). It also checks Bob's online social network for recommendations by friends he often eats out with. The NutrApp polls for the ingredients of meals and portion sizes from virtual sensors, and determines suitability by matching with the requirements of his diet plan. Time to cook is also important – his calendar has posted a reminder about an early afternoon meeting he must prepare for (Fig. 1B). By merging online information with GPS the NutrApp will try to locate members of Bob's social network in the neighbourhood, whose calendars or status information show they are available – if any are found Bob will receive a suggestion to invite them to join him.

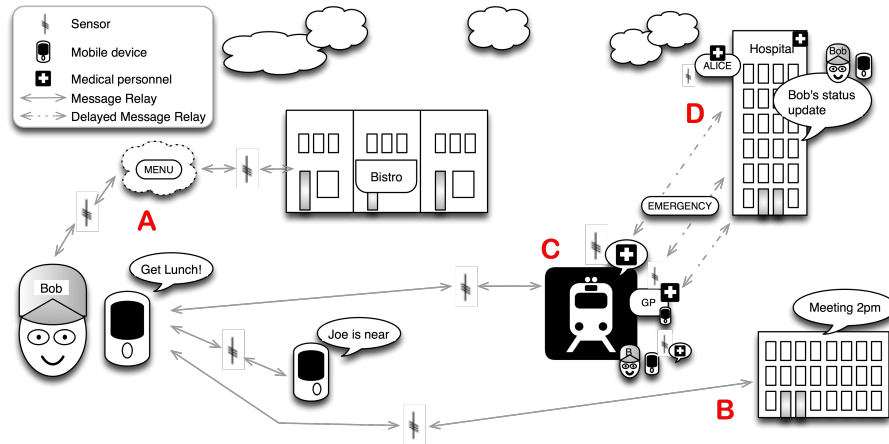


Fig. 1. Schematic for the PreSense scenario, illustrating the exchange of data streams between sensors, and the interaction between human actors and other entities as a result.

To illustrate an emergency in which situational context is communicated to external actors via sensors, let us consider what happens if Bob ignores the alerts he receives to

stop for lunch, because he forgot to carry his medication. He decides to return directly to his office, 45 minutes away via the subway. However, due to a signal failure Bob's train is held stationary just outside a station 1 hour later (Fig. 1C). The stressful situation, combined with more exercise than normal and the time since his last meal, result in a large drop in his blood sugar. Without the sensors recording Bob's blood sugar he may not recognise symptoms of hypoglycaemia till they become severe. In our scenario his smartphone warns him to consume food or drink with high sugar content urgently. His sensors attempt to warn Alice when his blood sugar reaches a critical threshold; however with the train stuck in a tunnel Bob's sensors are unable to connect to Alice's.

Bob's sensors also attempt to locate nearby resources that can help to alleviate his symptoms. A *General Practitioner* (GP) in the next carriage receives the emergency alert. Virtual sensors apply a context-sensitive filter to Bob's medical information (some of which is held on his personal devices). Another virtual sensor calculates Bob's location using GPS and a schematic of the train and transmits this to the GP's mobile device over a (local) wireless network. The GP locates Bob, and armed with the information needed to attend to the semi-conscious patient, successfully handles the emergency.

When the train exits the tunnel the delayed emergency alert is relayed to Alice (Fig. 1D), with a timestamp that indicates that it has now expired. An update with more current, valid information on Bob's status is also relayed to Alice over the Internet.

To be effective, this scenario implies the need to connect different information streaming sources in time- and location-constrained situations, via (context-sensitive) virtual sensors. Particularly, it illustrates the demands of attaching streaming information to real world entities such as people – the GP must be able to identify the patient via sensor stream ownership. Wireless networks also play a role in information exchange; in the emergency situation this is how the virtual and physical sensors communicate.

3 Related Work

Ontologies for user modelling follow two paradigms: standardisation- and mediation-based modelling [26]. The first is based on a top-down approach in which ontologies for user modelling are designed to be domain-independent (top-level ontologies), or still high-level but domain-specific (upper ontologies) in order to be reusable by multiple systems. The second is a bottom-up approach which proposes an integrated user model for a specific goal within a specific context [1].

The Friend of a Friend (FOAF) ontology⁶ is a top level ontology that models generic information about a user, including their name, social graph, interests and location. However, current FOAF profiling is based on the *static* representation of, in some cases, highly changing data, such as the temporal location of a user, or their current position in the world (à la *foursquare*⁷). Since many of these highly changing user properties can be observed through sensors, different ontologies for considering sensor data in user modelling have emerged. The Service-Oriented Context-Aware Middleware (SOCAM) ontology [11] is an upper ontology which introduces concepts like *Activity*, *Location*, *ComputationalEntity* and *Time* under the umbrella

⁶ FOAF ontology: <http://xmlns.com/foaf/spec>

⁷ <http://www.foursquare.com>

concept of `ContextEntity`. Although it models sensors using the `Device` concept, it does not provide a link between a sensor and its owner, nor a relation between the sensor's observations and user properties.

The General User Model Ontology (GUMO) is a top-level ontology introduced in 2005 [14]. GUMO is based on the User Mark-up Language (UserML) [13] and considers dimensions including personality, demographics, emotional and physiological state. The use of sensors in GUMO is considered particularly for users' physiological state; Heckman et al. [14] suggest the use of wearable bio-sensors to register users' body conditions such as pupil dilation and blood pressure. Although they consider the use of UbiWorld⁸ for integrating users in ubiquitous environments, there is no clear definition of the way in which a sensor's relationship with a user's properties could be addressed.

The Ontonym ontologies⁹ [25] are a collection of seven upper ontologies for pervasive computing, including the *Sensor*, *Device* and *Person* ontologies. They are designed to allow the definition of ownership between a `Sensor` and a `Person` through the `Device` class's `owns` and `ownedBy` properties. However, Ontonym requires the definition of a new ontology to map each sensor observation to user properties. For example, to add a relationship between a user's mobile device GPS's location observations and the user's location, the *Location* ontology is defined to declare the `LocatableEntity` and `LocatableFeature` classes (and associated properties). Ontonym is, to the best of our knowledge the only user modelling ontology available online.

Work done in sensor data integration into ontology-based user modelling following a bottom-up approach includes the Mobile Ontology-based Reasoning and Feedback System (MORF) [2], which defines a set of domain-specific ontologies which include classes such as `Patient`, `Doctor` and `HeartRateSensorData`. Their model allows monitoring and transmitting a patient's data through a mobile device. However, restrictions due to domain-specific design prevent MORF and other such bottom-up ontologies from being extensible to generic user modelling.

Relevant components of standard ontologies are discussed in the requirements identified in section 4 and revisited in 5.4 where we assess the extent to which these are met.

4 Requirements

The scenario presented in section 2.3 highlights not only the relevance of the identification of sensors and their observations as meaningful web resources, but also the importance of addressing the generated data streams as users' feature properties. In this section, we identify requirements for associating sensor data to user modelling.

Identification and Addressability: To uniquely identify and dereference sensor resources. In our scenario, Alice should be able to identify Bob's sensors, as well as the potential relations among these sensors. For example, by exposing Bob's physical activity and sugar levels, through the definition of his pedometer, as well as his glucose sensor as web resources, health care services could react in a contingency situation, in which external entities such as nearby emergency medical services could respond according to Bob's physical location (see section 6.2).

⁸ UbiWorld can be tested at: <http://www.ubisworld.org>

⁹ Ontonym ontologies: <http://ontonym.org>

Sensor Ownership and Provenance: To establish the sources of information, including entities and processes involved in the generation of measurements from observed stimuli. Provenance in sensor data is crucial for assessing trust judgements on information. An *Entity*, in particular a *Person*, should be able to address a sensor as its own – Bob, for instance, should be able to associate sensors with himself. Given a sensor data stream, it should be possible to access the sensor publishing the given stream and identify the sensor’s owner. In the scenario, Alice and the GP should be able to identify the streams they are consuming as Bob’s.

Association of Sensor Data and Profile Information: To map explicitly, a user’s property characterised by a stimulus with the sensor that observes this stimulus. In the scenario, Alice must be able to associate Bob’s (continuously changing) location with, e.g., Bob’s `current_location` property, observed by his GPS.

Privacy in Data Streams: To consider how identity information should be exposed and to whom: (1) The consumer of a data stream should be guaranteed that no other service has impersonated the sender; (2) The owner of a data stream should be able to establish authentication methods so only authorised consumers have access to it. E.g., besides Bob, Alice should be authorised to access Bob’s health information, as well as the closest emergency doctor who treats Bob at the scene (the latter will have access to a filtered view).

Sensor Data Expiration: To enable a data stream to declare an estimation of the period of time in which its data should be considered valuable. In our scenario, Alice must be able to tell if the received information is still valid, e.g., Alice must know the latest (valid) position of Bob and the time beyond which it is no longer valid.

Interaction with Smart Entities: To allow the representation of collective stimuli in which different entities, including the user, are involved. With collective stimuli we refer to the aggregation of common detectable changes in observable properties. E.g., Bob’s location-based proximity social graph is a property derived from the collective stimulus of being located in the vicinity of Bob, within a radius of 5km.

Integrate Physical and Virtual Presence Stimuli: To identify and incorporate virtual and physical stimuli as part of a user’s presence. This integration would bridge the user’s physical and online personae. In the scenario, the NutrApp would make use of Bob’s online social network to obtain the references of those entities to be monitored for physical presence proximity.

5 The Sensing Presence Ontology

In this section we introduce concepts related to users’ presence and present the Sensing Presence (PreSense) Ontology¹⁰. It defines key concepts and properties required to describe users’ features in terms of virtual sensor observables. PreSense models users as entities whose presence is the aggregation of online and physical properties. It represents sensors’ observations for deriving presence properties and particular features of interest, following the Stimulus-Sensor-Observation (SSO) ontology design pattern [18]. Fig. 2 illustrates the structure of the PreSense ontology, focusing on the relationships between its core components.

¹⁰ PreSense ontology available at: <http://purl.org/net/preSense/ns>

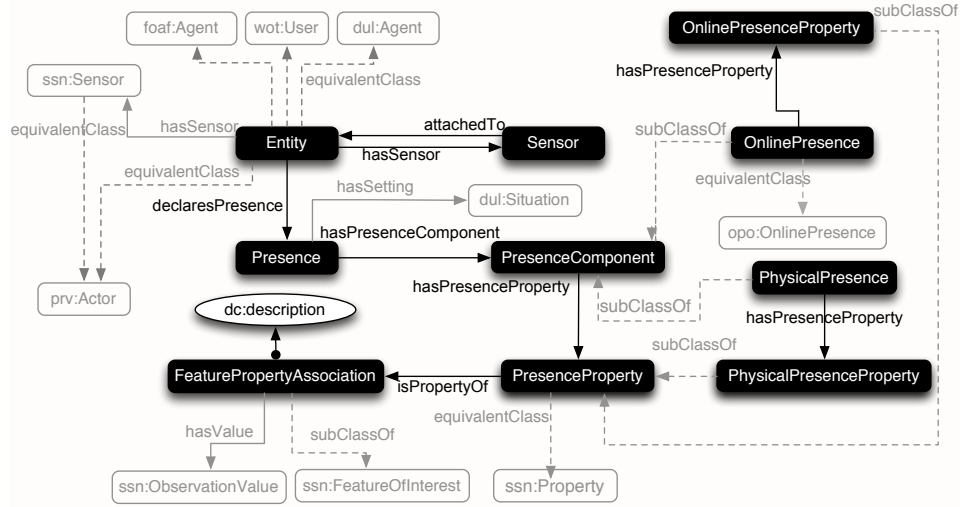


Fig. 2. Sensing Presence Ontology (PreSense) Overview

5.1 User Modelling Based on Personal Sensors

Based on the definition of virtual sensors (in section 2.2) and the SSO design pattern [18], we introduce the concept of “*personal sensors*” to refer to both physical sensor devices and compositions of computations or procedures that measure a user’s properties. The information embedded in the data produced by personal sensors includes the users’ *online* and *physical presence contexts*. By *online presence context*, we refer to the information provided by the aggregation of personal data streams (e.g. microblog posts, emails, text messages) generated by a user within a window of time. We consider the *physical presence context* as the abstraction of physical features, measured by sensor devices, regarding a user’s state of existence or being present in a place or a thing (e.g. the user’s location, body temperature). Both online and physical presence interweave dynamically with a user’s surrounding environmental context, which can include other entities like people, places and things (e.g. members of the user’s social graph who are close by, or local points of interest – POIs).

5.2 Imported Ontologies

Specifications on how to exchange sensor data and their observations have been defined by the Open Geospatial Consortium (OGC). In particular the OGC’s Sensor Web Enablement¹¹ (SWE) suite is a broad standardisation initiative which comprises models such as the Sensor Model Language¹² (SensorML) and the Observation & Measurement¹³ (O&M) standards, and services such as the Sensor Observation Service (SOS)

¹¹ <http://www.opengeospatial.org/projects/groups/sensorweb>

¹² <http://www.opengeospatial.org/standards/sensorml>

¹³ <http://www.opengeospatial.org/standards/om>

[18]. However, sensor data sharing and discovery expose different challenges involving semantic heterogeneity and integration. The Semantic Sensor Web (SSW) [24] approaches these challenges by providing an ontological platform that defines a machine-readable specification of the conceptualisations that underlie this sensor data.

There are over twelve sensor ontologies [5] for declaring a specification of sensing devices; some include sensors’ domain definitions and their relation to observations and measurements. The need for a domain independent and end-to-end model for sensing applications led to the creation of the W3C’s Semantic Sensor Network Incubator Group¹⁴ (SSN-XG), who developed the Sensor and Sensor Network (SSN) ontology¹⁵ [6,22]. Taking into account available standards such as the OGC’s SWE, the SSN ontology merges sensor-, observation- and system-focused views. The ontology describes sensors following the SSO ontology design pattern [18] and considers spatial provenance properties through the SSN’s Deployment module.

Following ongoing research and standardisation efforts, we use the SSN ontology to represent sensors in PreSense. Further, we use the Provenance Vocabulary¹⁶ (PRV) [12] to extend provenance-related metadata regarding both sensors and their owners through `prv:Actor`. For modelling an `Entity` asserting the ownership of a sensor, we use `foaf:Agent`. According to the FOAF specification, a `foaf:Agent` can refer to a person, a group, software or a physical artifact. The Web of Trust¹⁷ (WOT) ontology is used to ensure that the ownership of a sensor cannot be falsified by a third party, thus providing a solid base for valid sensor attachment. PreSense models a user to be equivalent to a `wot:User`. This equivalence allows a user to assert a digital signature to a web resource, which ensures that: (1) The provenance of the resource cannot be falsified easily; (2) The resource cannot be modified without revoking the provenance of the information.

From the Online Presence Ontology¹⁸ (OPO), we reuse `opo:OnlinePresence` to model users’ online presence properties. Finally, from the Dolce Ultralight Ontology (DUL)¹⁹ we reuse `dul:Agent` to align existing properties of SSN with a `preSense:Entity` (abbreviated prefix `ps:` used hereafter), and `dul:Situation`, in defining the contextual setting of an entity’s `ps:Presence`.

5.3 Core Components

Table 1 summarises the requirements fulfilled by each of the core components of the PreSense ontology, which we discuss next:

Entity An entity is modelled to be equivalent to `foaf:Agent`, `wot:User`, `dul:-Agent` and `prv:Actor`. The function of the `Entity` class is twofold: (1) to describe the identity of an individual (not only persons but entities in general) to whom the sensor data should be attached; and (2) to avoid provenance falsification

¹⁴ SSN Incubator Group: <http://www.w3.org/2005/Incubator/ssn/wiki>

¹⁵ SSN Ontology: <http://purl.oclc.org/NET/ssnx/ssn>

¹⁶ Provenance Vocabulary: <http://purl.org/net/provenance>

¹⁷ WOT Ontology: <http://xmlns.com/wot/0.1>

¹⁸ Online Presence Ontology: <http://online-presence.net/opo/spec>

¹⁹ Dolce Ultralight Ontology: <http://www.loa-cnr.it/ontologies/DUL.owl>

Table 1. Match of core PreSense ontology components to requirements

	Ident. & Address-ability of Sensors	Sensor's Ownership & Provenance	Sensor & User Profile Assoc.	Privacy	Data Expiration	Interaction with Smart Objects	Integration of Phys. & Virt. Pres. Stimuli
Entity	•	•	—	◦	—	—	—
Sensor	—	—	•	—	•	•	—
Presence	—	—	—	—	—	•	•
PhysicalPresence	—	—	—	—	—	•	•
OnlinePresence	—	—	—	—	—	—	•
FeaturePropAssoc.	—	—	•	—	—	—	—

Legend: • Yes. ◦ Limited. — No.

through the use of digital signatures in `wot:User`. The `Entity` class considers the property `hasSensor` for attaching a sensor to an entity (its inverse property is `attachedTo`). `Entity` is `skos:closeMatch` with `ssn:Platform`, which is considered to be an `Entity` to which a System of sensors is attached. However, SSN considers a `Platform` to be a `dul:PhysicalObject` which is disjoint with `dul:SocialObject`.

Sensor A sensor is defined by the `ssn:Sensor` class and refers to a physical object that detects, observes and measures a stimulus. The `ps:attachedTo` property is used to assert that a `ps:Entity` owns this sensor (its inverse property is `ps:hasSensor`). In order to extend provenance metadata of a sensor and its observations, we model the `ssn:Sensor` to be equivalent to a `prv:Actor`.

Presence A `Presence` refers to the state or fact of existing or being manifest in a place or a thing. We consider that a `Presence` is an aggregation of an `Entity`'s online and physical manifestations, that occur within a situation or setting. Following DUL, a situation is defined as a “*relational context*” created by an observer on the basis of a description frame.

Physical Presence This is the abstraction of the aggregation of physical properties featuring a quality of an entity. These properties are derived by sensors observing physical stimuli. The `ps:PhysicalPresence` class manifests an entity to be in a state of existing or being present in a place or a thing. These physical presence properties can be broken down into different modules regarding different dimensions in which users' properties can be linked to sensor data.

Online Presence This is equivalent to `opo:OnlinePresence`; it refers to the abstraction of the aggregation of online properties featuring a quality of an `Entity`, e.g., a user. These properties are derived by virtual sensors observing stimuli involving this `Entity`, e.g., the detection of a user's change of status on a social network site through the `ps:OnlineStatusStream`.

Feature Property Association Following the SSO ontology design pattern we introduce this class to bridge a sensor's observed stimulus and the feature that this stimulus characterises in the user model. It is a subclass of `ssn:FeatureOfInterest`; which being an abstraction of real world phenomena, proxies a stimulus through a

quality that can be observed by (an `ssn:Property` of) a sensor, and the PreSense property describing this quality (i.e., `ps:PresenceProperty`). The `ps:FeaturePropertyAssociation` class establishes a relation with the `ps:Presence` through the property `ps:hasPresenceProperty`, and by declaring `ps:PresenceProperty` to be `owl:sameAs ssn:Property`.

5.4 Fulfilment of the Requirements

The PreSense ontology addresses all of the requirements identified in section 4. Table 2 summarises the differences between the PreSense Ontology and the existing upper ontologies introduced in section 3.

Table 2. The PreSense Ontology, compared to existing, standard models

	Ident. & Addressability of Sensors	Sensor's Ownership	Sensor's Provenance	Sensor & User Profile Assoc.	Privacy	Data Expiration	Interaction with Smart Objects	Integration of Phys. & Virt. Pres. Stimuli
FOAF	—	—	—	—	—	○	—	○
SOCAM	○	—	○	—	—	○	●	○
GUMO	—	●	—	—	—	○	○	○
Ontonym	—	●	●	—	—	●	—	—
PreSense	●	●	●	●	○	●	●	●

Legend: ● Yes. ○ Limited. — No.

PreSense uses the `SSN:Sensor` ontology to model sensors and sensor data. `Entity` acts as the bridge through which sensor data and profile information can be associated. By reusing the FOAF, WOT and PRV ontologies, entities and sensor ownership can be uniquely identified. The use of WOT partially covers privacy issues. However questions still remain about the correct structure for introducing privacy settings within data streams; we aim to tackle this in future work. Sensor data expiration can be handled using `ssn:observationSamplingTime`. PreSense allows the representation of physical and online presence and their corresponding properties by enabling a bridge between a user's properties and the sensors observing these properties.

6 Applying PreSense

This section revisits the scenarios presented in section 2.3 and provides an overview on how to represent different information with the PreSense core ontology.

6.1 Extending PreSense Core Ontology with Modules

PreSense modules are extensions to the PreSense core vocabulary that provide additional information regarding a specific type of property. Currently PreSense has

two modules, the *spatial properties* module and the *health properties* module. The spatial property module includes *Location*, which is a spatial quality of an entity; this property is linked to a sensor by the *ps:FeaturePropertyAssociation* whose value is the observation of, in this case, a GPS sensor. The health properties module considers the *PhysiologicalState* class and its subclasses (Fig. 3).

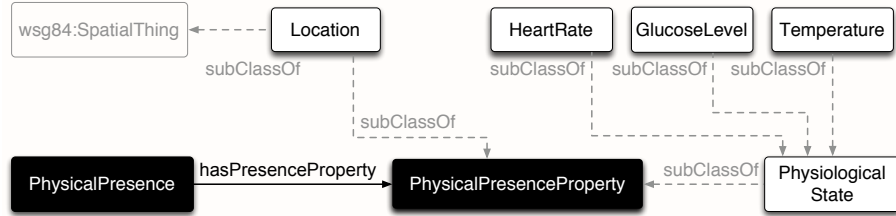


Fig.3. PreSense modules for handling features related to Location and PhysiologicalState.

6.2 Scenario with PreSense

In this scenario Bob's levels of glucose can be monitored as part of his profile. This could be modelled with the PreSense ontology as:

```
@prefix ps: <http://purl.org/net/preSense/ns#> .
@prefix physioState: <http://purl.org/net/preSense/physioState/ns#> .
@prefix prvTypes: <http://purl.org/net/provenance/types#> .
@prefix prv: <http://purl.org/net/provenance/ns#> .
@prefix ssn: <http://purl.oclc.org/NET/ssnx/ssn#> .
<http://my.identity.org/Bob> a ps:Entity, a foaf:Person;
ps:hasSensor <http://my.identity.org/Bob/sensors/glSen1/>.
ps:declaresPresence _:p1.

_:p1 a ps:Presence;
ps:hasPresenceComponent _:phyPr.

_:phyPr a ps:PhysicalPresence;
ps:hasPresenceProperty _:prop1.

_:prop1 a physioState:GlucoseLevel;
ps:hasPresenceProperty _:glucoseLevel.
ps:isPropertyOf _:bloodGlucose .

<http://my.identity.org/Bob/sensors/glSen1/>
a ssn:Sensor, prv:Actor, prvTypes:Sensor;
prv:operatedBy <http://my.identity.org/Bob> .
prv:observedBy <http://my.identity.org/Bob/sos/observations/glSen1/>.
<http://my.identity.org/Bob/sos/observations/glSen1/> a ssn:Observation;
ssn:observedProperty _:glucoseLevel.
_:glucoseLevel a ssn:Property, ps:PresenceProperty;
ssn:isPropertyOf _:bloodGlucose.
_:bloodGlucose a ps:FeaturePropertyAssociation;
```

In this example, Bob registers his glucose level measuring sensor (*glSen1*) and his physical presence (*phyPr*). His physical presence considers in this example the health

properties module; in particular the glucose level property (`glucoseLevel`). This property corresponds to the property observed by his `glSen1` sensor. This sensor observes changes in his blood sugar levels, (`bloodGlucose`), which is the feature of interest. This association enables Alice to monitor Bob’s sugar levels. Following the URI scheme for linked sensor data proposed by Janowicz et al. [17], Alice could refer to, e.g., `http://my.identity.org/Bob/sos/observations/glSen1/mgPerdL`, which is a reference to all observations gathered by `glSen1` corresponding to the feature of interest `bloodGlucose` for the observed property, milligrams per decilitre, `mgPerdL`. In a similar way the PreSense ontology could be applied for registering Bob’s heart rate micro-sensor.

The scenario also considers the attachment of virtual sensors to the user’s profile. Bob could allow other systems to consume his online status stream (e.g., tweet streams) as:

```
<http://my.identity.org/Bob> a ps:Entity, a foaf:Person;
ps:hasSensor <http://my.identity.org/Bob/sensors/stSen1/>.
ps:declaresPresence _:p1.

_:p1 a ps:Presence;
ps:hasPresenceComponent _:onlPr.

_:onlPr a ps:OnlinePresence;
ps:hasPresenceProperty _:prop2.

_:prop2 a ps:OnlineStatusStream;
ps:hasPresenceProperty _:personalStatusStream.
ps:isPropertyOf _:twitterStatusStream .

<http://my.identity.org/Bob/sensors/stSen1/>
a ssn:Sensor, prv:Actor, prvTypes:Sensor;
prv:operatedBy <http://my.identity.org/Bob> .
prv:observedBy <http://my.identity.org/Bob/sos/observations/stSen1/>.
<http://my.identity.org/Bob/sos/observations/stSen1/> a ssn:Observation;
ssn:observedProperty :personalStatusStream.
_:personalStatusStream a ssn:Property, ps:PresenceProperty;
ssn:isPropertyOf _:twitterStatusStream.
_:twitterStatusStream a ps:FeaturePropertyAssociation
```

In this case, Bob declares his personal status stream as a property of his online presence `onlPr`. This property is a proxy for generated contingency tweets on behalf of Bob, and is observed by the virtual sensor `stSen1`. In this case all observations regarding generated tweets could be obtained through `http://my.identity.org/Bob/sos/observations/stSen1/status`. Data derived from his health monitoring devices could trigger an alert when Bob is facing a health contingency situation. This alert could be proxied through Bob’s `stSen1` sensor; which could alert, e.g., a particular list of Bob’s followers in his physical environment about his need for medical attention. They in turn could, on validating the information and its provenance, notify health services about the impending emergency.

7 Conclusions

The PreSense ontology is designed to extend people’s digital identities through the information obtained by their attached personal sensors. It provides a first step toward

the integration of user modelling and “smart environments”. PreSense distinguishes between the notions of physical presence, e.g., location data obtained from digital sensors, and virtual presence, provided, for instance, by the aggregation of personal data streams, but affords equal status to both. Moreover, PreSense allows the assignment of meaning to sensors’ observations in terms of user modelling features.

Future work includes the development of PreSense modules addressing interaction with smart entities and environments, by mapping a user’s location to that of other nearby entities (*NearByPOI* and *NearByFriends* modules). We are also testing the application of the PreSense ontology in real world scenarios, starting with the exploration of new environments and ongoing events. We plan to carry out an evaluation of PreSense during the Tramlines Festival²⁰ in Sheffield in July 2011. Bearing in mind privacy restrictions, only information about participants’ interests in music and festivals, and preferences when exploring a new location will be broadcast. Our architecture relies on SparqlPush²¹, which handles real-time notifications by associating feeds with SPARQL/Update²² triggers. Collective information on events and POIs, gathered from Twitter and public Facebook feeds, will be filtered based on users’ profiles and their current geo-location (`ps:PhysicalPresence`), obtained from GPS on their mobile devices. Where public, users’ online status information (`ps:OnlinePresence`) will be used as an additional filter, e.g., to suggest festival events and *NearbyPOIs* that other users with similar interests have visited, and notify them about *NearByFriends*.

Acknowledgements A.E. Cano is funded by CONACyT, grant 175203. A.-S. Dadzie and V. Uren are funded by the European Commission (EC) 7th Framework Programme (FP) project SmartProducts (EC grant ICT-231204). The work reported in this paper was also supported by the EC 7th FP project WeKnowIt (EC grant ICT-215453). The authors thank Gregoire Burel and Matthew Rowe for their valuable comments.

References

1. N. Baumgartner and W. Retschitzegger. A survey of upper ontologies for situation awareness. In *Proc., 4th IASTED International Conference on Knowledge Sharing and Collaborative Engineering*, pages 1–9, 2006.
2. R. Benlamri and L. Dockstader. MORF: A mobile health-monitoring platform. *IT Professional*, 12(3):18–25, 2010.
3. C. Bizer, T. Heath, and T. Berners-Lee. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, 2009.
4. P. Bonato. Wearable sensors and systems. *IEEE Engineering in Medicine and Biology Magazine*, 29(3):25–36, 2010.
5. M. Compton, C. Henson, H. Neuhaus, L. Lefort, and A. Sheth. A survey of the semantic specification of sensors. In *SSN09: Proc., 2nd International Workshop on Semantic Sensor Networks at ISWC 2009*, pages 17–32, 2009.
6. M. Compton, H. Neuhaus, and K. N. Tran. Reasoning about sensors and compositions. In *2nd International Semantic Sensor Networks Workshop*, 2009.

²⁰ Tramlines Festival 2011: <http://www.tramlines.org.uk>

²¹ SparqlPush: <http://code.google.com/p/sparqlpush>

²² SPARQL Update Language: <http://www.w3.org/Submission/SPARQL-Update>

7. K. D. An overview of wearable and implantable medical sensors. *Yearbook of medical informatics*, 66(9), 2007.
8. D. Estrin, D. Culler, K. Pister, and G. Sukhatme. Connecting the physical world with pervasive networks. *IEEE Pervasive Computing*, 1(1):59–69, 2002.
9. M. F. Goodchild. Citizens as sensors: the world of volunteered geography. *GeoJournal*, 69:211–221, 2007.
10. N. Green, R. H. R. Harper, G. Murtagh, and G. Cooper. Configuring the mobile user: Sociological and industry views. *Personal and Ubiquitous Computing*, 5:146–156, 2001.
11. T. Gu, X. H. Wang, H. K. Pung, and D. Q. Zhang. An ontology-based context model in intelligent environments. In *Proc., Communication Networks and Distributed Systems Modeling and Simulation Conference*, pages 270–275, 2004.
12. O. Hartig. Provenance information in the web of data. In *LDOW 2009: Proc., Linked Data on the Web Workshop at WWW'09*, 2009.
13. D. Heckmann and A. Krueger. A user modeling markup language (UserML) for ubiquitous computing. In *Proc., 9th international conference on user modeling*, pages 393–397, 2003.
14. D. Heckmann, T. Schwartz, B. Brandherm, M. Schmitz, and M. von Wilamowitz-Moellendorff. GUMO – the General User Model Ontology. In L. Ardissono, P. Brna, and A. Mitrovic, editors, *User Modeling 2005*, pages 428–432. Springer, 2005.
15. T. Hogg. Inferring preference correlations from social networks. *Electronic Commerce Research and Applications – Special Issue: Social Networks and Web 2.0*, 9(1):29–37, 2010.
16. D. Hong, D. Chiu, V. Shen, S. Cheung, and E. Kafeza. Ubiquitous enterprise service adaptations based on contextual user behavior. *Information Systems Frontiers*, 9:343–358, 2007.
17. K. Janowicz, A. Broering, C. Stasch, and T. Everding. Towards meaningful URIs for linked sensor data. In *FIS Workshop: Towards Digital Earth: Search, Discover and Share Geospatial Data*, 2010.
18. K. Janowicz and M. Compton. The Stimulus-Sensor-Observation ontology design pattern and its integration into the Semantic Sensor Network ontology. In *SSN2010: Proc., 3rd International Workshop on Semantic Sensor Networks at ISWC 2010*, 2010.
19. S. Kabadayi, A. Pridgen, and C. Julien. Virtual sensors: Abstracting data from physical sensors. In *WoWMoM 2006: Proc., International Symposium on a World of Wireless, Mobile and Multimedia Networks*, pages 587–592, 2006.
20. A. Kamis, T. Stern, and D. Ladik. A flow-based model of web site intentions when users customize products in business-to-consumer electronic commerce. *Information Systems Frontiers*, 12:157–168, 2010.
21. A. Küpper. *Location-Based Services: Fundamentals and Operation*. Wiley, 2005.
22. H. Neuhaus and M. Compton. The semantic sensor network ontology: A generic language to describe sensor assets. In *AGILE Workshop: Challenges in Geospatial Data Harmonisation*, 2009.
23. M. Rowe. The credibility of digital identity information on the social web: a user study. In *Proc., 4th ACM Workshop on Information Credibility on the Web*, pages 35–42, 2010.
24. A. P. Sheth, C. A. Henson, and S. S. Sahoo. Semantic sensor web. *IEEE Internet Computing*, 12(4):78–83, 2008.
25. G. Stevenson, S. Knox, S. Dobson, and P. Nixon. Ontonym: a collection of upper ontologies for developing pervasive systems. In *CIAO '09: Proc., 1st Workshop on Context, Information and Ontologies*, pages 1–8, 2009.
26. M. Viviani, N. Bennani, E. Egyed-zsigmond, and I. Liris. A Survey on User Modeling in Multi-Application Environments. In *Third International Conference on Advances in Human-Oriented and Personalized Mechanisms, Technologies and Services*, 2010.
27. T. Yamada, M. Kaneko, and K. Katou. A mobile communication simulation system for urban space with user behavior scenarios. In L. T. Yang et al., editors, *High Performance Computing and Communications*, pages 979–990. Springer, 2005.

Semantic Social Profile – a Semantic Boost for the
Social Information in MediaWiki

Y. Katkov, D. Pokoptsev

Semantic Social Profile – a Semantic Boost for the Social Information in MediaWiki

Yury Katkov¹ and Dmitry Pokoptsev²

¹ National Research University of Information Technologies, Mechanics and Optics
Saint Petersburg, Russia

`katkov.juriy@gmail.com`

² National University of “Kyiv-Mohyla Academy”
Kiev, Ukraine

`d.pokoptsev@gmail.com`

Abstract. Semantic MediaWiki proved its effectiveness as a platform for creating semantically connected hypertexts. Unfortunately, in most wiki engines, information about contributors (wiki authors) remains in shadow for human observers and is not available for program agents. Semantic Social Profile proof of concept introduces a flexible way of integration the social features of MediaWiki and Semantic MediaWiki annotations. In the paper we describe the extension development process and some of the use cases that will become possible with it.

1 Introduction

Wiki systems are getting more and more popular: now Wikipedia is one of the most visited websites on the planet³, many corporations use wiki in their enterprise systems, almost every project tracker includes a wiki engine; wikis are widely used in education. Despite their usefulness, with constant growth of information wiki systems require new ways of data managing. Most of wiki engines lack tools that provide multi-attribute search and automatic checks for data consistency. Moreover, wikis may benefit from social features, since most users are accustomed to them and communication is essential for better articles quality. In this work, by the example of semanticizing social profiles, we illustrate the fact that the data converted to a suitable format for automatic processing, make the implementation of various practical use cases possible much easier and faster.

The paper is organized as follows: Section 2 briefly introduces the advantages of semantic wikis, in Section 3 we show the importance of social data in wikis and describe the Social Profile extension which allows wiki users to manage their social profiles and provide them with a set of social networking tools. In Section 4 we show some promising use cases that will be available with Semantic Social Profile extension; namely the improvements in user statistics and search, providing information for better tracking of the revision history, and ability to reuse the profile outside the wiki. Section 5 includes the main aspects of

³ <http://www.alexa.com/topsites>

implementation as well as description of the current mapping from Social Profile fields to existing properties in various ontologies.

2 Semantic MediaWiki

One of the main problems of regular wikis is that they are in significant dependence on authors efforts in content management [9]. This disadvantage becomes especially apparent when users create list articles, the type of articles that contain a list of links to other wiki pages that relate to a particular topic. Whenever a new article is created, it needs to be manually added to the corresponding lists. In addition to this, when the information in some of the items is changed, one has to do routine manual work to edit information in the list article. Hence, authors are responsible for data consistency.

Moreover, wikis offer only fulltext search and page categorization as tools for finding information. So, neither can wiki information be reused by machines nor can it be fully accessed by those who need it [9].

The problems of traditional wiki systems mentioned above were already known in the mid 2000s when the first semantic wiki engines were created [8]. Semantic wiki allows querying within the wiki, aggregate data in convenient views (tables, maps, calendars), export data to semantic formats (RDF, OWL) and allow for reasoning with it [10].

One of the most popular semantic wiki engines is Semantic MediaWiki (SMW) created in Karlsruhe Institute of Technology (KIT) and described in [7], [2], [9]. It is an extension for MediaWiki⁴ that offers a simple yet powerful mechanism of semantic annotations [2]. These annotations are used to make named links between the articles and to assign datatype properties to them. By combining the traditional MediaWiki features and the tools provided by SMW, it is possible to solve the above-mentioned problems of traditional wiki systems as described in [9].

Semantic MediaWiki provides various datatypes⁵ (Page, String, Number, Data, URL, etc) to represent the values of different properties. It also has an expressive query language that can be used both to process non-trivial search and to create list articles automatically. For example, a list of annual events announced on semanticweb.org⁶ has been created automatically from already existing data. Of course, in this case, the term “list” should be understood in much broader sense than just one-dimensional table, since SMW coupled with other extensions provides various forms of data retrieval. Another advantage that SMW provides is the ability to export data to RDF and import external vocabularies so that property names can be mapped to OWL ontologies such as FOAF, DC, SKOS.

⁴ <http://mediawiki.org>

⁵ http://semantic-mediawiki.org/wiki/Help:Properties_and_types#Data_types_for_properties

⁶ http://semanticweb.org/wiki/Concept:Semantic_Web_events_2011

3 Social Information in wiki systems. Semantic Social Profile extension

Social networks play an important role in everyday life of Internet users and significantly influence the structure of the Web. Managing social information is an important part of the Semantic Web vision [3], [5] and the Web 2.0 era we live in now. We believe that wiki systems can also leverage from the social data in a variety of ways.

First of all communication between authors leads to quality improvement of articles created by them. System of ranks, points and encouragement of the best authors stimulates efficient work. Furthermore, social information can be used for recommendation algorithms, offering users information they may be interested in. It also can be used as additional input data for contextual advertising algorithms. Finally, administrators and analysts get a possibility to retrieve useful information about the groups of wiki editors to manage the development of wiki content.

Social Profile⁷ is an extension that represents social information of wiki users and adds interfaces to manage their social data. It provides such features as storing profile information (personal information, interests, avatar), user-specific statistics, rank system, relationships between wiki users (friends or foes), and a wall to leave messages.

Although Social Profile extension provides many features, user data is still not so easy to access, since it is stored in a database and the only way to use it is to implement new MediaWiki extension. We propose an approach of semanticizing social data that will allow the profile information to be used more efficiently. Semantic Social Profile⁸ is an extension that is designed to utilize Semantic MediaWiki and Social Profile extensions to automatically benefit from both of them. Social Profile provides us with forms for data input, interfaces for communication and relationship management and other useful social features, whereas Semantic MediaWiki offers a ready-to-use system of querying the data and displaying results in various formats. SMW can also be used to make profiles portable between different social networks.

This extension deals with social data: it reacts on profile changes, e.g., avatar uploads, adding/removing friends, personal information editing and adds semantic annotations to the user page.

To sum up, Semantic Social Profile has two main goals:

1. To represent the information in the social profile in a format that is good for reuse inside the wiki. In other words we want to represent users profile in a form suitable for Semantic MediaWiki queries.
2. To implement profile portability by applying import vocabulary feature of SMW to the profile information.

Achieving the goals listed above will enable the use cases that were hard to implement before. We discuss these use cases in the next section.

⁷ <http://www.mediawiki.org/wiki/Extension:SocialProfile>

⁸ <http://code.google.com/p/semantic-social-profile>

4 How to use semantic social information

4.1 User statistics improvement

Semantic Social Profile allows one to query non-trivial statistics about wiki users. For example, our extension makes it possible to ask “Get the users who are over 22 years old, located in Greece and interested in chemistry”. Much easier examples will work as well: “Get all avatars of users and show in a map where those users live”.

In order to retrieve such statistics, the user ought to have some knowledge of how to work with Semantic MediaWiki query language. Usually, administrators and experienced wiki users are those who are able to get this task done. Wiki administrators are always aware of what kind of people are using wiki as it allows them to tune the engine, to run the projects on content creation, and to find out what kind of topics need more attention.

This feature is also useful for the automated services, bots and extensions that use data mining methods to provide recommendations of friends, and recommendation of articles they may be interested in or context ads. We suppose that with lots of structured data its much easier to divide a community into groups with similar interests. For example, one can adjust the weighting coefficients in such recommendation algorithms as collaborative filtering [4] based on users social profiles.

4.2 Turning wiki into FOAF editor

Since MediaWiki makes it possible to export the semantic properties of the given page and since we align those properties with commonly used ontologies, the wiki becomes a full-fledged FOAF editor, like FOAF-a-MATIC⁹. FOAF profiles created on the wiki can be used in other websites that support FOAF.

4.3 Providing information for the semantic history

In [1] Jie Bao et. al presented a tool called Semantic History (SH) that enables one to track the semantic history in SMW. Implemented as SMW extension, Semantic History facilitates the construction of the complex queries related to the revision history. The questions like “How many Ph.D students participated in creation of this page during the last 10 months?” is just one of the possible examples of such queries.

Note that in order to address such kind of questions as the one presented above, semantic history needs formally described information about the users. Of course, this can be done directly from the user page but Semantic Social Profile make this process much easier.

⁹ <http://www.ldodds.com/foaf/foaf-a-matic>

4.4 Search improvement

Another important thing is that semantic social data can be used to improve search abilities of the wiki. For instance, it is possible to use the user page properties as facets in faceted search. Such kind of search is implemented in popular social networks and it was proved to be efficient when dealing with fixed structure of the data. To implement the faceted search, it may be reasonable to use the SMW extension called Semantic Drilldown¹⁰ which groups wiki pages by their properties.

5 Implementation

Semantic Social Profile has simple design. Basically, it is a group of callback functions related to the hooks in Social Profile extension. When profile information changes, callback functions update the semantic properties of the user page. Based on this information one could easily implement statistics features described in the first use case.

Since social information needs to be saved in a standard way for each user, Semantic Social Profile keeps properties in the template, instead of setting them directly. The mechanism of MediaWiki templates is explained in [6]: “the template defines the logic and the appearance of a part of a page. It keeps placeholder variables, which are filled by the instantiating page. Inserting annotations in the template entails the annotation of all pages using the template with the same annotations.” Semantic Social Profile uses a template which stores all social properties on one page, and transcludes it automatically inside each user page. Using templates allows the extension to add properties in a flexible way and thus prevents from manual maintenance of the text consistency.

To implement the FOAF editor use case, we have used the Import vocabulary feature of SMW. With this feature it is possible to reuse vocabulary that belongs to existing Semantic Web documents in a way of associating the vocabulary’s elements with wiki terms. After making the FOAF vocabulary available for import, the RDF exported from the profile page will directly use elements of the FOAF specification, thus allowing users to edit FOAF profiles through wiki.

The installation script of the Semantic Social Profile creates the properties necessary for a persons description and aligns these properties with the elements of imported vocabularies.

By now we have the following mapping between the Social Profile fields and properties of the ontologies used to describe people in RDF. This is a very basic variant, though it is being constantly developed.

The mapping refers to the following vocabularies: Friend of a friend (FOAF)¹¹, Address Schema¹², ResumeRDF¹³, and Relationship¹⁴.

¹⁰ http://www.mediawiki.org/wiki/Extension:Semantic_Drilldown

¹¹ <http://xmlns.com/foaf/spec/>

¹² <http://schemas.talis.com/2005/address/schema>

¹³ <http://rdfs.org/resume-rdf/>

¹⁴ <http://vocab.org/relationship/>

Table 1. Social Profile to FOAF mapping

Social Profile field	Ontology Element
Avatar	foaf:img
Name	foaf:name
Login	foaf:nick
E-mail	foaf:mbox
Birthday	foaf:dateOfBirth
Interests	foaf:interest
Websites	foaf:homepage
City (Location)	ad:localityName
State (Location)	ad:regionName
Country (Location)	ad:countryName
City (Hometown)	cv:birthPlace
State (Hometown)	cv:birthPlace
Country (Hometown)	cv:birthPlace
About Me	cv:aboutPerson
Friends	rel:friendOf
Foes	rel:enemyOf

6 Conclusion and future work

One can say that wikis are very popular tools but they are still not nearly as popular as the social networks – in order to check that fact it is enough to compare the number of users on Wikipedia and Facebook. Social relationships and actions get people involved in what they do. Our hypothesis is that properly adjusted social features can make a big improvement to the process of collective knowledge creation.

However, social features should be more tightened with the rest of the wiki providing new opportunities for the end-users. In this paper we have described our first step – to convert social information in a machine readable form.

In the current implementation of SSP we have very simple representation of the geodata. In the next releases we plan to add geographical coordinates support and to use URIs of cities, countries etc instead of just names of the geographic objects.

Also, custom fields and interests are going to be linked with URIs of the Linked Data objects. For example, if a user inputs “soccer”, the system will offer to chose Soccer term from DBPedia. Future work also includes integration of the ratings (so called karma) in the wiki and development of the tools for emphasizing the information about the authors of the wiki articles. Both of these tools can motivate the user to participate in the content creation process.

Wikis are designed specially for storing the common knowledge of communities. By developing and promoting social wiki tools we hope to turn many people from chaotic activity to supporting semantic evolution of data.

7 Acknowledgements

Thanks to Li Ding, Jie Bao, creators of Semantic History extension for inspiring us to do our work. Many thanks to Jack Phoenix and the whole Social Profile team for their support and advice.

References

1. Bao, J., Ding, L., McGuinness, D.: Semantic history: Towards modeling and publishing changes of onlinesemantic data. In: The 2nd Social Data on the Web workshop (SDoW2009). Citeseer (2009)
2. Bao, J., Ding, L., Hendler, J.: Knowledge representation and query in semantic mediawiki: A formal study. In: Tetherless World Constellation (RPI) Technical Report. pp. TW-2008-42 (2008), http://www.cs.rpi.edu/~baojie/pub/2009-06-02_iswc-bao_tr.pdf
3. Berners-Lee, T.: Giant global graph. Decentralized Information Group (DIG): timbl's blog pp. 15-12 (2007), <http://dig.csail.mit.edu/breadcrumbs/node/215>
4. Breese, J., Heckerman, D., Kadie, C., et al.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the 14th conference on Uncertainty in Artificial Intelligence. pp. 43-52. Madison: Morgan Kaufmann (1998)
5. Hendler, J., Berners-Lee, T.: From the Semantic Web to social machines: A research challenge for AI on the World Wide Web. *Artificial Intelligence* 174(2), 156-161 (2010)
6. Herzig, D., Ell, B.: Semantic mediawiki in operation: experiences with building a semantic portal. *The Semantic Web-ISWC 2010* pp. 114-128 (2010)
7. Krötzsch, M., Vrandečić, D., Völkel, M.: Semantic mediawiki. *The Semantic Web-ISWC 2006* pp. 935-942 (2006)
8. Renaud, E.O., Delbru, R., Möller, K., Völkel, M.: Annotation and navigation in semantic wikis. In: In Völkel et. p. 29 (2006)
9. Völkel, M., Krötzsch, M., Vrandečić, D., Haller, H., Studer, R.: Semantic wikipedia. In: Proceedings of the 15th international conference on World Wide Web. pp. 585-594. ACM (2006)
10. Ziesche, S.: Semantic wikis and disaster relief operations (2006), <http://www.xml.com/pub/a/2006/12/13/semantic-wikis-and-disaster-relief-operations.html>

Forecasting Audience Increase on YouTube

M. Rowe

Forecasting Audience Increase on YouTube

Matthew Rowe

Knowledge Media Institute, The Open University,
Milton Keynes, United Kingdom
`m.c.rowe@open.ac.uk`

Abstract. User profiles constructed on Social Web platforms are often motivated by the need to maximise user reputation within a community. Subscriber, or follower, counts are an indicator of the influence and standing that the user has, where greater values indicate a greater perception or regard for what the user has to say or share. However, at present there lacks an understanding of the factors that lead to an increase in such audience levels, and how a user's behaviour can affect their reputation. In this paper we attempt to fill this gap, by examining data collected from YouTube over regular time intervals. We explore the correlation between the subscriber counts and several behaviour features - extracted from both the user's profile and the content they have shared. Through the use of a Multiple Linear Regression model we are able to forecast the audience levels that users will yield based on observed behaviour. Combining such a model with an exhaustive feature selection process, we yield statistically significant performance over a baseline model containing all features.

Keywords: User modelling, Forecasting, Social Web, Data Mining, Behaviour

1 Introduction

A fundamental quality of Social Web platforms is the provision of profile building functionality, allowing users to construct an identity within such systems that conveys their interests and persona in a bespoke form. The motivation for constructing user profiles is often born of the need to share content and develop a reputation within a given community, or site. The Oxford English dictionary defines reputation as:

*...the beliefs or opinions that are generally held about someone or something.*¹

In developing a reputation within a Social Web platform, one is able to accrue a larger audience comprised of other users who are keen to listen to, read or watch the content that is being shared. Therefore the greater the reputation and standing of a user within the platform, the more subscribers and followers the user will gain. Reputation is also synonymous with influence, providing a

¹ http://oxforddictionaries.com/view/entry/m_en_gb0702720

mutually beneficial relationship where the greater the reputation of an individual the greater their influence, and vice versa.

Quantifying the standing and perception of a user can be achieved by measuring his/her in-degree on a given Social Web platform - i.e. the number of subscribers they have, or rather, the size of their audience. Therefore a key motivation behind the participation of users on Social Web platforms, and the construction of their profiles and development of reputation, is to increase this in-degree level. Exploring the relation between subscriber counts and behaviour features is particularly important on Social Web platforms that revolve around the sharing of *social objects* - i.e. videos, photos, etc. At present there is no understanding of what factors contribute to an increase in subscriber counts, and audience levels, nor is there an analysis of the affects of user behaviour on such levels. Exploring such a relationship would identify patterns of usage and isolate the correlation between certain behaviours and the advance of a user's reputation.

In this paper we seek to provide the link between reputation - quantified through audience levels - and the behaviour characteristics of both users and the content they share. For our analysis we use the video-sharing, social networking platform YouTube,² exploring the question: *What factors influence audience levels?* For behavioural features we extract user profile attributes, such as the number of channels the user has subscribed to and the number of videos they have watched, and attributes of the content they have shared, such as the number of views and favourites their content has had.

To perform our analysis we employ Multiple Linear Regression (MLR) Models. First, we assess the correlation between subscriber counts and our collected behavioural features - identifying key feature correlations. Second we use observations from our analysis to forecast audience levels of individual users at various time-steps, comparing a MLR model using all features against another MLR model post-feature selection. Experiments are performed over 10 days worth of data collected from YouTube over 4 hour time intervals.

This paper is structured as follows: section 2 explains our data collection approach, the features we model with an overview of the collected datasets, and the role that semantics plays in our work. Section 3 presents our approach to forecasting subscribers using Multiple Linear Regression modelling over time-series data, detailing the correlation between the in-degree of users and other behaviour features, and our experiments with forecasting in-degree changes. Section 4 describes related work to our approach and section 5 concludes the paper.

2 Data Collection and Overview

In order to predict the number of followers - denoted as subscribers on YouTube - that a given person will develop over time we required data against which our forecasting methods could be empirically compared. To gather such a collection

² <http://www.youtube.com>

we utilised the YouTube Data API,³ using the following process: we queried the API for the most recent 100 uploaded videos to the UK version of YouTube - in order to increase the likelihood that we gathered English-language videos, as our future work will look at the language features of the videos and comments. For each video collected, we wanted to analyse the relationship between audience levels of the person who shared the content and both the behaviour of the user and the reception of the content they have shared. Therefore, every 4 hours, we logged 6 individual statistical features associated with both the video uploader and the video itself. Below we describe these features and our reason for their selection:

– *User Statistics:*

- *In-degree:* Measures the number of users who are currently subscribed to the user - i.e. the video uploader. This feature provides the value that we wish to predict for individual users: audience levels.
- *Out-degree:* Measures the number of YouTube channels, and therefore other users, that the given user is currently following. This captures the openness of the user and the degree to which they are interested in following the content of other users.
- *User View Count:* Measures the number of unique views of videos on YouTube. Provides a measure of the extent to which the user watches content and the time spent on the platform, thereby gauging the activity behaviour of the user.
- *Post count:* Measures the number of unique videos uploaded by the user onto the platform. Allows the extent to which the user shares original content with the community to be gauged.

– *Post Statistics:*

- *Post View Count:* Measures the number of times a given video has been watched. This provides a notion of popularity for the user’s uploaded content. As we are only logging the statistics for individual videos, this measure is for a user’s single video, not for all videos uploaded by the user.
- *Favourite count:* Measures the number of times a given video has been ‘favourited’ by users. Like the Post View Count, this feature assesses the popularity of the uploaded content, but in a more extreme sense - given that the action of ‘favouriting’ a video requires the viewer to watch the video and be impressed by what he/she sees.

At 4 hour time intervals we collected the most recent 100 uploads to UK YouTube, thereby progressively building up a collection of videos, stopping once we had collected 2000. This process was continued for 10 days, logging the details of the 2000 collected videos. This method of collection is similar to the one used in [8], which collected a dataset for inspection of view counts as an indicator of popularity analysed over time.

³ <http://code.google.com/apis/YouTube/2.0/reference.html>

For our analysis however we wanted to use a scaled down version of the dataset, that would allow graphical inspection of the data. Therefore we randomly chose 10% of this dataset for use in our analysis and experiments - producing a collection of 200 videos. For each of the videos we analyse the first 10 time steps from each video's upload. In the following section we detail our approach for forecasting subscriber counts for individual users, utilising Multiple Linear Regression Models for this process. In order to induce these models we require analysis to be performed over the data, thereby assessing the fit of the models with respect to the data. To do this *model selection* phase we divided the dataset of 200 videos, and their logged information over time, into a training/testing split using an 80/20 random split. The former set provides the data over which we perform analysis of features and their change over time.

2.1 The Role of Semantics

At present our analysis only covers a small facet of the Social Web: YouTube. The models that we explore and learn in this paper will form the basis for comparisons against other Social Web platforms - e.g. Twitter⁴ and Flickr⁵ - allowing patterns in behaviour to be associated with audience levels and the effects of applying such patterns observed. The heterogeneity of the Social Web hinders such explorations without a common understanding of statistical features across domains, and the annotation of such features using a common schema or ontology. To this end the information that we have logged is converted to RDF and labelled with concepts from a Behaviour Ontology⁶ specifically designed for this purpose.

The ontology contains an abstract concept called *Impact* that captures the impact of either a post or a user at a given point in time. This concept is specialised for a post using the class *PostImpact*, allowing the number of views and favourites that a piece of content has incurred to be recorded as data type properties at a given point in time. Likewise *UserImpact* records statistical features of the user at a given point in time from the above list. We omit further details of the ontology in order to focus on feature analysis and forecasting, however it is suffice to say that the role of semantics at present is not fully exploited, given that we are only using information collected from a single Social Web platform. Our future work, that we describe within the conclusions of this paper, seeks to extend this work to cover forecasting using other social data.

3 Forecasting Audience Increase

Figure 1 shows the change in in-degree level of YouTube users over time - shown as a *logged* value in Figure 1(a) - and the increase in view counts of the uploaded videos over time - shown in Figure 1(b). These graphs demonstrate that, in each

⁴ <http://www.twitter.com>

⁵ <http://www.flickr.com>

⁶ http://people.kmi.open.ac.uk/rowe/ontologies/UserOnto_0.23.rdf

case, we observe that as time increases, so too does the in-degree of the user and the view counts of the video. This prompts the question as to which features of the user, and the content that they have uploaded, are correlated with the increase in their subscriber count, and thus the reputation and influence that they may have within the community.

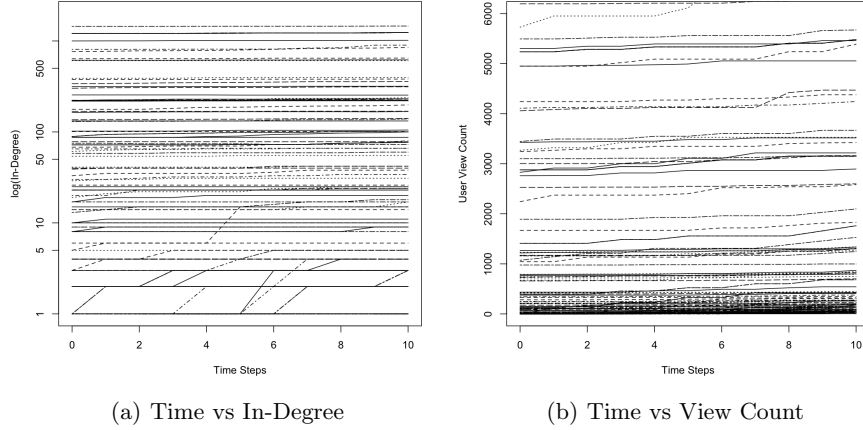


Fig. 1. Analysis of 80% training split, showing an increase in the in-degree of the user over time, and the increase in view counts over time also.

To analyse this correlation we employ a Multiple Linear Regression (MLR) model, allowing multivariate data to be employed as the empirical basis for model induction. The general form MLR, the form of which is shown in Eq. (1), is to learn some coefficients - contained within β - that fit the observed data, allowing the model to be assessed for *goodness-of-fit*. Within this model α denotes the intercept of the induced plane within an n -dimensional space and ϵ denotes a normally-distributed random error vector.

$$Y = \alpha + \sum_{i=1}^n \beta_i X_i + \epsilon \quad (1)$$

As we described previously, our data contains various variables that we have measured over incremental time periods. We wish to incorporate these variables into a MLR model, and then assess the correlation that each feature has with the dependent variable - in our case the in-degree of the user. Table 1 details the features used, together with their vector notation labels. Our data is however time-series, in that we have collected it over successive 4-hourly increments, therefore X_i is actually a vector of length n corresponding to n time steps.

Table 1. Features used within the model and their corresponding independent variable labels within the models

Label	Time-series Feature
X1	User In Degree
X2	User Out Degree
X3	User View Count
X4	User Post Count
X5	Post View Count
X6	Post Favourite Count

3.1 Model Selection

The first analysis that we perform is to assess the fit of a model using all possible features. Given that our dependent variable is the in-degree of a user - denoted by $X1$ - we can rewrite Eq. (1) as the form shown in Eq. (2):

$$X1 = \alpha + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \beta_5 X_5 + \beta_6 X_6 + \epsilon \quad (2)$$

Using this model we can assess the correlation of the model's variables with the in-degree of the user, thereby asking: *What is the correlation between in-degree and other features?* To do this we took the training split of our dataset and derived average measures for each of the features from Table 1 from all of the users within that split. Our intuition is that we can observe a general pattern in the data that can be later used for forecasting over individual samples (users) - we denote this model, including all the features, as Ψ_{all} .

Table 2. Model results for Ψ_{all}

Feature	Est' Coefficient	Standard Error	t-Value	P($x > t$)
X2	-4.565e+00	3.025e+00	-1.509	0.205719
X3	2.239e-04	7.193e-04	0.311	0.771099
X4	-1.165e+00	1.408e+00	-0.827	0.454550
X5	3.160e-02	7.518e-03	4.203	0.013664*
X6	2.220e+00	1.594e+00	1.393	0.236016
Summary Res. St Err: 0.2447 Adj R^2 : 0.9872 $F_{5,4}$: 140.3 p-value: 0.0001399				
Signif. codes: p-value < 0.001 *** 0.01 ** 0.05 * 0.1 . 1				

The results shown in Table 2 are from evaluating the null hypothesis that no significant relationship exists between the dependent variable - the in-degree of the user - and the individual independent variables, using the t-test and the given t-Value for specific features. This hypothesis is rejected if a statistically significant relationship exists between the two variables and the p-value is below a given significance level. The results indicate that only X5 - post view count

the subset size increases, the performance of the subsets models does too. This figure also demonstrates the strong correlation between the number of views that a video receives and the in-degree of the user - given the high adjusted R^2 value yielded for the use of solely X5 within a model. At the same subset size of 1, the number of posts that the user has made on the site - i.e. videos uploaded - is found also have high predictive value, followed by the number of the videos that the user has watched. As one would expect, as a user interacts more on the platform and watches more videos and uploads more content, other members of the community's awareness is raised, and the user is noticed - resulting in an increase in subscribers and therefore audience members.

Table 3. Results for Ψ_{best}

Feature	Est' Coefficient	Standard Error	t-Value	P($x > t$)
X2	-4.594196	2.736807	-1.679	0.15406
X4	-1.301767	1.211048	-1.075	0.33153
X5	0.032853	0.005746	5.718	0.00229**
X6	2.528960	1.128538	2.241	0.07513.
Summary Res. St Err: 0.2215 Adj R^2 : 0.9895 $F_{4,5}$: 213.9 p-value: 8.963e-06				
Signif. codes: p-value < 0.001 *** 0.01 ** 0.05 * 0.1 . 1				

Feature selection identifies an optimal combination of features as being the inclusion of all original features from Ψ_{all} without X3, we therefore construct another model using this combination of features and denote this as Ψ_{best} . The fit of this new model may alter the correlation of the independent variables with the in-degree of the user, given the omission of a previous feature, therefore we explore the following question: *What is the correlation between in-degree and best features following feature selection?* Table 3 shows the results from a MLR model fitted with the best performing features. We note that unlike the previous model, the number of favourites that a video has yielded is now found to have a significant correlation with the in-degree of the user - at a significance level of $\alpha = 0.1$. The induced coefficients show an interesting pattern in the relationship of the model, given that an increase in the number of videos viewed by the user actually has a negative effect on in-degree increase, however this feature is not statistically significant within the model, and therefore conclusions drawn from such a correlation are not well supported. We formally define this new model as:

$$X1 = \alpha + \beta_2 X_2 + \beta_4 X_4 + \beta_5 X_5 + \beta_6 X_6 + \epsilon \quad (3)$$

3.2 Forecasting

Model selection has identified two models that we can use for prediction: 1) Ψ_{all} - using all 5 features; and 2) Ψ_{best} - using 4 features identified as optimising

adjusted R^2 values. Our analysis was performed over the 80% training split, allowing general patterns in the data to be observed by averaging the features. In this stage we now wish to test the predictive quality of these two models and their ability to forecast the in-degree of individual users. For our experiments we test two scenarios: the first predicts the in-degree at time t using the previous k time steps as training data for inducing the model's coefficients. The second experiment predicts the in-degree at the final time step - i.e. $t = 10$ - when trained on the previous k steps.

Unlike the previous model selection stage, at this point we perform predictions at the *micro-level* for each user in our held-out test split of 20% - therefore predicting the in-degree for 40 users. To test the performance of each of our models we train both models for a single person, using their respective features collected over time, rather than building a general model as in the previous section using averages from which the coefficients are then induced. Our predicted in-degree should match, as close as possible, to the actual observed in-degree in the data, therefore to measure the error in predictions we measure the Root Mean Square Error of the prediction as follows: let $\hat{Y}$ denote our predicted value and Y denote our actual observed value, we define the Root Mean Square Error (RMSE) as:

$$RMSE(\hat{Y}, Y) = \sqrt{\frac{\sum_{i=1}^n (\hat{Y}_i - Y_i)^2}{n}} \quad (4)$$

One-Step Forecast Our first experiment, as described above, tests the predictive quality of the two models when forecasting the in-degree of each user one-step ahead. That is to say that at time $t = k + 1$, the model is trained using the previous k time steps. Table 4 presents the RMSE achieved by both Ψ_{all} and Ψ_{best} as k is iteratively increased. Training each model using only 1 step achieves equivalent performance, however as k is increased we see differences in the RMSE produced by each model, culminating in an average performance by the latter - Ψ_{best} - which outperforms the use of all features.

Table 4. Prediction Error Rates for Ψ_{all} and Ψ_{best} predicting the in-degree at the next time-step when trained using the past k steps.

Model	1	2	3	4	5	6	7	8	9	Average
Ψ_{all}	0.707	0.720	1.101	0.602	1.282	0.345	3.794	0.464	0.478	1.055
Ψ_{best}	0.707	0.622	1.027	0.542	0.575	0.248	0.868.	0.795	0.605	0.665

Signif. codes: p-value < 0.001 *** 0.01 ** 0.05 * 0.1 .

In order to assess the significance of the error values achieved at differing values of k for the tested models, we utilised the Sign Test. We tested the null hypothesis that there was no difference in performance between the models, rejecting this hypothesis should a significant improvement be found when using

Ψ_{best} . For each value of k the test was performed for all 40 test samples - denoting the 40 users in the test set whose in-degree was to be predicted. From our analysis we note that where $k = 7$, the model derived following feature selection outperforms the use of all features at a liberal significance level of $\alpha = 0.1$. The difference in performance between Ψ_{all} and Ψ_{best} over all values of k was not found to be statistically significant, indicating that the feature selection improves predictive quality but not in a significant manner.

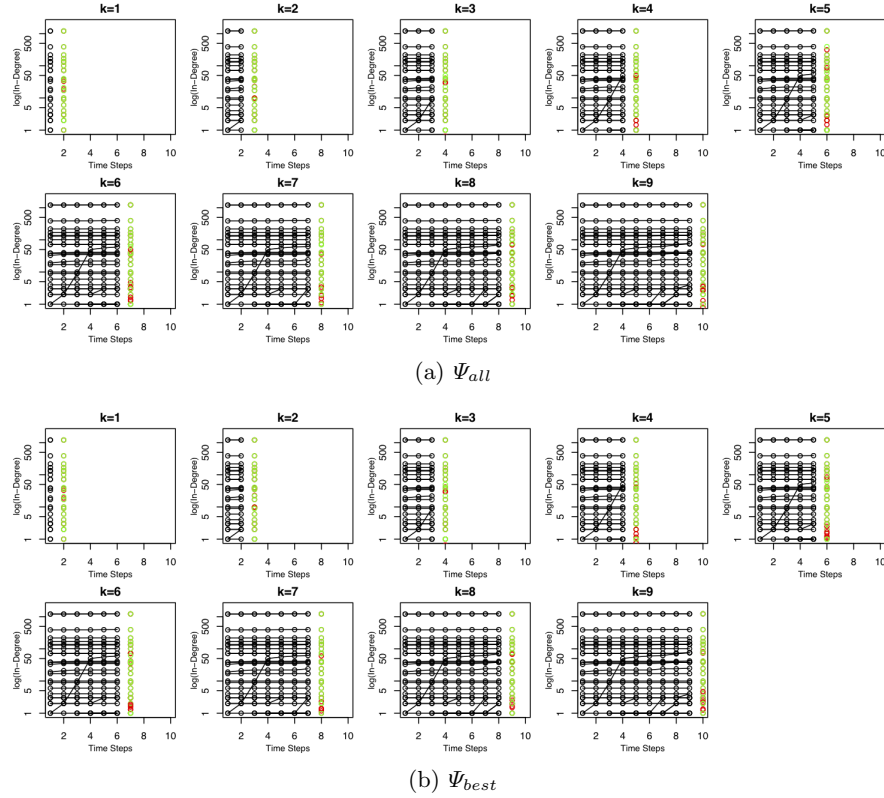


Fig. 3. Predictions (green) and observations (red) for user in-degree at next $k + 1$ time-step, when trained using past k steps.

Figure 3 presents the overlay plots for predicted in-degree values against the observed values for each of our 40 tested users. Predictions are denoted using green circles and observations using red circles, while training time-steps are denoted by black circles and path lines. Our goal is to minimise the number of red circles that can be seen by overlaying the green circles - denoting the predicted values, thereby providing a qualitative assessment of prediction accuracy.

Differences in performance between the two models can be observed at various values of k , for instance at $k = 5$ where Ψ_{best} has a notably less red circles, and therefore errors, for larger audience levels than Ψ_{all} .

Final-Step Forecast Rewrite after re-running the experiments again with the correct error measures For our second experiment we explored the prediction of a user’s in-degree at the final time step in the collected data - i.e. $t = 10$. Therefore each of the MLR models was trained using the previous k steps, and then applied to the final time step. As previous, we measure the RMSE achieved as k is iteratively increased and report on statistically significant results. As Table 5 indicates Ψ_{all} outperforms Ψ_{best} on average, where the difference in performance post-feature selection was found to be statistically significant at a significance level of $\alpha = 0.05$. However when comparing the results at individual values of k no significant improvements in performance were found. Qualitative analysis of the predictions using both models is shown in Figure 4, where we observe at $k = 6$ for larger in-degree levels the reduction in prediction errors when using Ψ_{best} over Ψ_{all} .

Table 5. Prediction Error Rates for Ψ_{all} and Ψ_{best} predicting the in-degree at the final step ($t = 10$) when trained using the past k steps.

Model	1	2	3	4	5	6	7	8	9	Average
Ψ_{all}	0.806	0.739	0.641	0.438	0.261	0.329	0.761	0.495	0.478	0.550
Ψ_{best}	0.806	0.546	0.352	0.161	0.149	0.259	0.571	0.490	0.605	0.438*
Signif. codes: p-value < 0.001 *** 0.01 ** 0.05 * 0.1 .										

4 Related Work

The measurement of subscriber counts, and therefore in-degree values, as a gauge of influence has been studied in various pieces of recent work. For instance, work in [9] explored influence in citation networks, to assess the flow of topics amongst authors. Using Topical Affinity Propagation, the authors model the topic distribution of authors - the intension being to identify experts, or heavily influential nodes in citation networks. Their experiments found, naturally, that those nodes with a higher in-degree - derived from incoming citations - for a given topic, represented an influential member of that community. Citation networks were also studied by [10] - although they are defined as ‘*social networks*’ - also, like our work, using Multiple Linear Regression models over time-series data. The motivation behind their work was to note the influence that content network measures had upon social network measures and vice-versa - explained through analysing the coefficients of the induced models. Experiments were performed over 4 yearly time steps of the collected citation networks - formed using SPARQL queries against bibliographic endpoints.

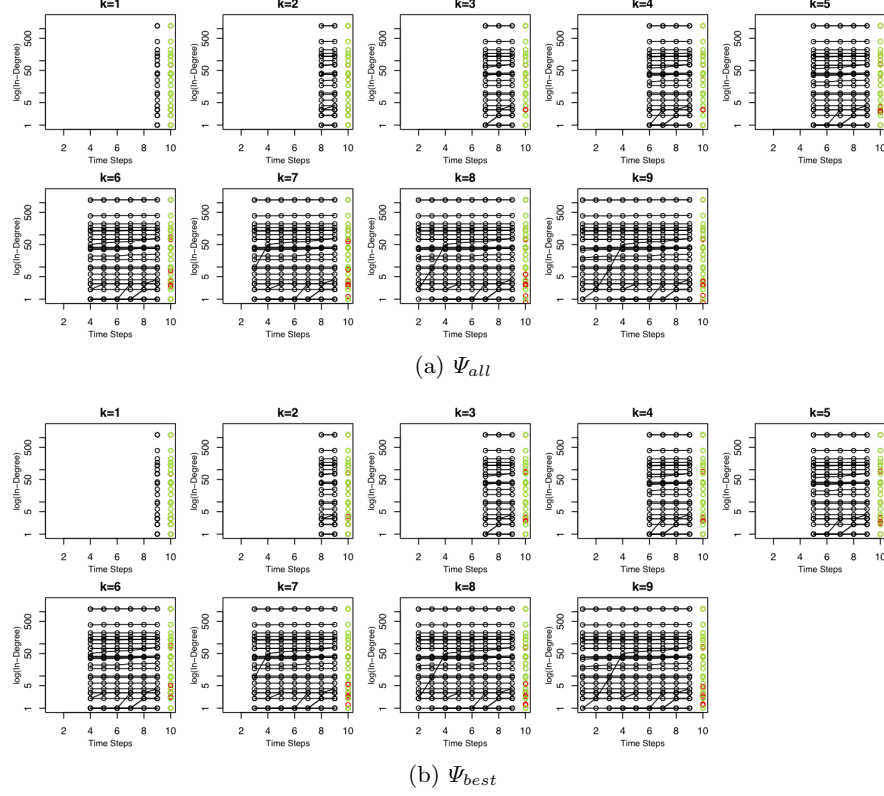


Fig. 4. Predictions (green) and observations (red) for user in-degree at final time step, when trained using previous k steps.

Assessing influence on Twitter is described in recent work by [4] where three distinct notions of influence are defined: in-degree influence, retweet influence and mention influence. The authors found that the in-degree of a user is not correlated with audience engagement - unlike retweets and mentions. Influence among social network members is also studied in [1], this time over data collected from the photo-sharing platform Flickr.⁷ To test the affects of influence the authors examine the propagation of tags from nodes in the network, analysing whether a connected node, after a certain point, begin using a tag that their network members had used. The authors found no significant influence affect, however this could be explained by the nature of the data source used - given that Flickr is used to share photos of external resources, not necessarily connecting users by common topics.

⁷ <http://www.flickr.com>

Prediction and forecasting using social data has been explored in [3] where the notion of user authority is compared against user affinity in the context of Yahoo! Answers. The described approach models the objectivity of the question poster: where high-objectivity is associated with the poster looking for an expert. Their results show that a combination of affinity and authority - the latter denoting influence - provides the best approach to predicting the best answer selection. The work described in [5] predicts whether a user will retweet a given URL or not. The authors claim that so-called ‘*powerusers*’ are highly-influential users with many followers who contribute to information spread via retweets. They model a user’s influence over time-series data collected from Twitter, inducing the model’s parameter up to a point, and then using this model to predict propagation after the point in time. The model uses the notion of neighbourhood influence as one parameter, where the propagation probability is dependent on past influence in the surrounding network. Twitter is also used for prediction in [2] which predicts stock market levels from Twitter mood. First analysis is performed of the correlation between independent mood variables and the Dowjones Index - denoting the dependent variable. A fuzzy neural network is then used to predict the index based on the mood variables.

The analysis of in-degree distributions within social networks on Social Web platforms has been presented in work by [7] and [6]. The former, [7], assesses the statistical properties of social networks collected from several platforms, including YouTube, assessing the clustering coefficients of the collected networks, and the in-degree/out-degree distributions. Their findings showed a high level of local clustering, indicating the possibility of topical cliques forming around niche subjects, and a power-law distribution for the in-degree. In [6] the in-degree distribution on the popular technology-news web site Slashdot⁸ is analysed. On Slashdot moderators post stories on to the site, functionality which is not available to commenters. However after sorting users by their in-degree, the authors identified the top-ranked individual not as a main content contributor, but as a commenter, indicating that the increase in subscribers/followers/friends is based on community interaction and discussion, and not on the creation of initial content.

Of the work studied, although several pieces deal with influence and the measurement of in-degree levels, no known work attempts to predict in-degree levels and the gain in reputation that a user could yield based on behavioural characteristics. The approach presented within this paper attempts to fill this gap by studying the correlation between in-degree levels and various features - similar to the modelling approach described in [2] - in order to identify predictive features for forecasting the evolution of a user’s standing on a Social Web platform.

5 Conclusions

In this paper we have examined the relationship between behavioural features, of both users and the content they share, and user subscriber levels - attempting

⁸ <http://slashdot.org/>

to quantify the reputation that a user has on a given Social Web platform. In essence we sought to explore *what factors influence audience levels?* We found that the reception of content by the platform had a strong link to an increase in audience levels. In particular we observed two key aspects from our analysis:

1. *The greater the number of views of uploaded content, the greater the audience levels of the user*
2. *The more content is ‘favourited’ by users, the greater the audience levels of the user*

Interestingly we observed a negative relationship between audience levels and the behaviour of the user, suggesting that as the user participates more in the community by watching more videos and uploading more content, that such behaviour can have a negative effect on the person’s audience levels, and therefore his/her reputation. However it is worth noting that such relationships, although present in our analysis, are not statistically significant.

Combining the behavioural features into a single Multiple Linear Regression Model (Ψ_{all}) sought to explore the predictive power of feature combinations and the fit of the model to the empirical data. Post-feature selection, using exhaustive subset tests, found an alternative model (Ψ_{best}) using all possible behavioural features without the view count of the user. Comparison of the two models was then performed via two forecasting experiments over a held-out sample from our previous model selection stage. We found that for predicting a user’s in-degree at a final time-step when trained on the previous k steps, Ψ_{best} showed statistically significant performance over Ψ_{all} .

Our future work will explore two key avenues of work: the first is to extend this study over a larger dataset, one that we are currently collecting, and expanding the feature scope to include reciprocal features to capture the interaction behaviour of the user - i.e. via commenting activity. The second avenue is to expand this approach to other Social Web platform - e.g. Flickr and Twitter - to mine patterns that explain the link between reputation on such platforms and the behaviour of its users. For this second area of work, we require semantics to encode information in a common, machine-readable form, from which our methods can then function. In section 2 of this paper we briefly discussed the behaviour ontology that is used to represent the statistical features employed without our approach in a common form. This ontology will form the basis for our future work, allowing information from disparate platforms and provided using heterogeneous data schemas, to be represented in a common, machine-readable form.

Acknowledgements

This work is funded by the EC-FP7 project Robust (grant number 257859).

References

1. A. Anagnostopoulos, R. Kumar, and M. Mahdian. Influence and correlation in social networks. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 7–15, New York, NY, USA, 2008. ACM.
2. J. Bollen, H. Mao, and X.-J. Zeng. Twitter mood predicts the stock market. *CoRR*, abs/1010.3003, 2010.
3. S. Budalakoti and K. S. Barber. Authority vs affinity: Modeling user intent in expert finding. *Social Computing / IEEE International Conference on Privacy, Security, Risk and Trust, 2010 IEEE International Conference on*, 0:371–378, 2010.
4. M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi. Measuring User Influence in Twitter: The Million Follower Fallacy. In *Fourth International AAAI Conference on Weblogs and Social Media*, May 2010.
5. W. Galuba, D. Chakraborty, K. Aberer, Z. Despotovic, and W. Kellerer. Out-tweeting the Twitterers - Predicting Information Cascades in Microblogs. In *3rd Workshop on Online Social Networks (WOSN 2010)*, 2010.
6. V. Gómez, A. Kaltenbrunner, and V. López. Statistical analysis of the social network and discussion threads in slashdot. In *Proceeding of the 17th international conference on World Wide Web*, WWW '08, pages 645–654, New York, NY, USA, 2008. ACM.
7. A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *In Proceedings of the 5th ACM/USENIX Internet Measurement Conference (IMC'07)*, 2007.
8. G. Szabo and B. A. Huberman. Predicting the popularity of online content. *Commun. ACM*, 53(8):80–88, 2010.
9. J. Tang, J. Sun, C. Wang, and Z. Yang. Social influence analysis in large-scale networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 807–816, New York, NY, USA, 2009. ACM.
10. S. Wang and P. Groth. Measuring the dynamic bi-directional influence between content and social networks. In P. Patel-Schneider, Y. Pan, P. Hitzler, P. Mika, L. Zhang, J. Pan, I. Horrocks, and B. Glimm, editors, *International Semantic Web Conference*, volume 6496 of *Lecture Notes in Computer Science*, pages 814–829. Springer Berlin / Heidelberg, 2010.

TUMS: Twitter-based User Modeling Service

K. Tao, F. Abel, Q. Gao, G.-J. Houben

TUMS: Twitter-based User Modeling Service

Ke Tao, Fabian Abel, Qi Gao, Geert-Jan Houben

Web Information Systems, Delft University of Technology
{k.tao,f.abel,q.gao,g.j.p.m.houben}@tudelft.nl

Abstract. Twitter is today’s most popular micro-blogging service on the Social Web. As people discuss various fresh topics, Twitter messages (tweets) can tell much about the current interests and concerns of a user. In this paper, we introduce TUMS, a Twitter-based User Modeling Service, that infers semantic user profiles from the messages people post on Twitter. It features topic detection and entity extraction for tweets and allows for further enrichment by linking tweets to news articles that describe the context of the tweets. TUMS is made publicly available as a Web application. It allows end-users to overview Twitter-based profiles in a structured way and allows them to see in which topics or entities a user was interested in a specific point in time. Furthermore, it provides Twitter-based user profiles in RDF format and allows applications to incorporate these profiles in order to adapt their functionality to the current interests of a user. TUMS is available via: <http://wis.ewi.tudelft.nl/tums/>

Key words: user modeling, twitter, semantic enrichment, service

1 Introduction

Applications that aim for personalization and would like to adapt their functionality to the current interests and demands of a user require information about their users [1]. User-adaptive systems suffer from cold-start and sparsity problems [2]. For example, when systems have to deal with new users or fresh content, those systems require user profile information that allows for estimating the interests of the users. In this paper, we present a Twitter-based user modeling service (TUMS) that exploits users’s Twitter activities to infer semantically meaningful profiles.

Since Twitter was launched in 2007, it became the most popular micro-blogging service, with 190 million users who post more than 65 million posts every day¹. As people are not limited to a certain domain, they can discuss various topics of which most of the topics are related to news which could also be found in mainstream news articles [3]. There is also an active research community on Twitter studying topics including social network analysis [3, 4], community and user influence mining [5–7], recommendations of URLs [8] or sentiment analysis [9, 10]. Inferring user profiles from individual Twitter activities is a hard

¹ <http://techcrunch.com/2010/06/08/twitter-190-million-users/>

problem as tweets are limited to 140 characters which makes the deduction of semantics difficult. Rowe et al. [11] propose to exploit the context in which tweets have been published. In particular, they propose to map tweets to events and exploit semantic descriptions of these events to clarify the semantic meaning of tweets. In previous work, we proposed strategies to automatically map Twitter messages to related news articles which allows us to exploit the news articles to enrich the semantics of individual tweets [12]. This enrichment builds the basis for the user modeling strategies [13] that are made available via the TUMS service which we present in this paper.

The TUMS service allows end-users to inspect Twitter-based profiles and enables other applications to re-use these profiles. People can overview their personal Twitter activities or profiles of other users to explore the topics those users were concerned with in the past. Entity-based, topic-based and hashtag-based tag clouds allow to quickly grasp the content of the profiles and charts further visualize the profiles. Visualizing the profiles is important to make people aware of what can be inferred from their Twitter activities so that they can reconsider how they publish tweets on Twitter. TUMS is also of interest for other services on the Social Web as it enables them to consume Twitter-based profiles in RDF format. Profiles can be used for personalization and are particularly interesting for other applications that suffer from sparsity problems (e.g. services that cannot collect sufficient data about their users) and services that are interested in “realtime” or very fresh profile information.

In the following section we summarize related work before we introduce TUMS in detail in Section 3. We outline the architecture of our service, the user modeling strategies featured by TUMS and present the graphical user interface as well as the service API. In Section 4 we conclude and give an outlook on our plans for future work.

2 Related Work

With the advent of Semantic Web technologies and appropriate vocabularies such as FOAF [14], SIOC [15] or the Weighted Interest vocabulary [16], re-use of user profiles is becoming easier nowadays. Research on generic user modeling services [17], mediating user models [18], identifying users across system boundaries [19] and cross-system user modeling and personalization [20, 21] further supports the re-use of user profiles in different application contexts. In this paper, we introduce a service that generates user profiles by exploiting Twitter and allows for applying these profiles in other applications.

Research on Twitter often focuses on analyzing large fractions of the Twitter network to study information propagation patterns [3, 22, 23] or to identify influential users [6, 7]. Dong et al. [24] exploit Twitter to detect and rank fresh URLs that have possibly not been indexed by Web search engines yet. Lately, Chen et al. conducted a study on recommending URLs posted in Twitter messages and compare strategies for selecting and ranking URLs by exploiting the social network of a user as well as the general popularity of the URLs in Twitter [8].

Yet, there exists little research on analyzing the semantics of individual tweets and exploiting Twitter as a source for modeling user interests. Rowe et al. [11] proposed to exploit contextual information to enrich the semantics of Twitter messages. In previous work, we followed this suggestion and linked tweets to related Web resources [12]. Given this semantic enrichment, we proposed user modeling strategies that allow for recommending news articles. In this paper, we make those enrichment and user modeling strategies available to the public and present TUMS, a Twitter-based User Modeling Service.

3 TUMS: Twitter-based User Modeling Service

TUMS is a user modeling service that exploits the Twitter messages of individual users to infer and provide user interest profiles. TUMS targets two types of consumers: (i) end-users who would like to overview and inspect their profiles (or profiles of friends) in a structured way and (ii) applications that would like to incorporate the RDF-based user profiles in order to adapt their functionality to the current interests of a user. In this section, we first present the architecture of TUMS before we summarize the user modeling capabilities of TUMS (see Section 3.2

3.1 Architecture

TUMS aims to deliver both a human and a machine readable representation of the user characteristics that can be deduced from the posts a user published on Twitter. The only input required by TUMS is the Twitter username of the person that should be profiled. Figure. 1 depicts the architecture of the TUMS service. TUMS is composed of three modules: (i) data crawling and storage, (ii) Twitter-based user modeling module, (iii) a Web service that enables applications to query for profiles and (iv) a graphical user interface that allows applications to consume profiles.

Crawler and back-end From a user’s perspective the TUMS functionality is very easy to use as TUMS just requires a Twitter username as input. Given a username, TUMS initiates a pipeline for crawling, processing, storing and analyzing the Twitter data that is publicly available for the corresponding user. TUMS features a tweets and news crawler: the tweets crawler aggregates Twitter posts of the given user and stores the tweets in the TUMS data repository. The news crawler continuously monitors traditional news media. At the moment, TUMS monitors 62 RSS feeds published by mainstream news publishers such as CNN, New York Times, and BBC and aggregates the full content of each news article by making use of Boilerpipe² Once a

² <http://code.google.com/p/boilerpipe/>, a Java library that detects supplemental text (e.g. advertisements or menus) in a website and allows for the extraction of the main content.

TUMS Display your profile in a visualized way!

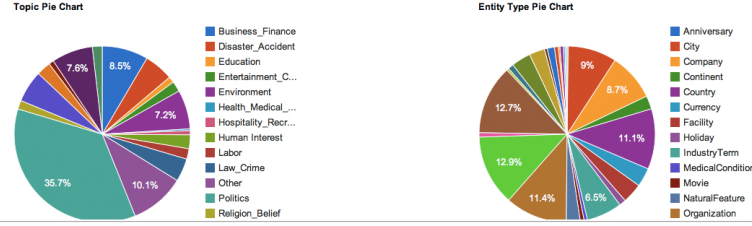
Hello, USAGodG20! Welcome back!

[hashtag cloud]

[peta](#) [car](#) [conspiracy](#) [teaparty](#) [no](#) [Globalization](#) [BUSH](#) [London](#) [patriot](#) [NWO](#) [MerryChristmas](#) [PayAt](#) [CFL](#) [nokill](#) [MerryChristmasEve](#) [NAU](#) [FF](#) [financial](#) [noki](#) [n](#) [24](#) [resistnet](#) [infowars](#) [ireport](#) [Obama](#) [2](#) [tcoot](#) [aspc](#) [2](#) [tpp](#) [dog](#) [NAFTA](#)

[Topic Pie Chart]

[Entity Type Pie Chart]



[Top 200 entities cloud]

[\[Prime Minister\]](#) [\[President\]](#) [\[CNN\]](#) [\[United States\]](#) [\[USD\]](#) [\[China\]](#) [\[European Union\]](#) [\[Russia\]](#) [\[United Kingdom\]](#) [\[Washington\]](#) [\[BBC\]](#) [\[Japan\]](#) [\[White House\]](#) [\[New York\]](#) [\[Congress\]](#) [\[Iraq\]](#) [\[GBP\]](#) [\[Senate\]](#) [\[North Korea\]](#) [\[South Korea\]](#) [\[the New York\]](#) [\[JavaScript\]](#) [\[Beijing\]](#) [\[official\]](#) [\[leader\]](#) [\[spokesman\]](#) [\[Islamic Republic of Iran\]](#) [\[David Cameron\]](#) [\[Barack Obama\]](#) [\[chairman\]](#) [\[London\]](#) [\[Italy\]](#) [\[Ireland\]](#) [\[Department of State\]](#) [\[Secretary of State\]](#) [\[correspondent\]](#) [\[Christmas\]](#) [\[Australia\]](#) [\[Pakistan\]](#) [\[America\]](#) [\[India\]](#) [\[California\]](#) [\[Obama administration\]](#) [\[food\]](#) [\[Twitter\]](#) [\[judge\]](#) [\[Middle East\]](#) [\[spokeswoman\]](#) [\[Republican Party\]](#) [\[New Year's Day\]](#) [\[Florida\]](#) [\[lawyer\]](#) [\[football\]](#) [\[Captain\]](#) [\[Texas\]](#) [\[The Times\]](#) [\[The New York Times\]](#) [\[Facebook\]](#) [\[Google\]](#) [\[iPhone\]](#) [\[Chicago\]](#) [\[injury\]](#) [\[New York City\]](#) [\[Qatar\]](#) [\[Thanksgiving\]](#) [\[Boston\]](#) [\[http\]](#)

Fig. 1. The architecture of TUMS

new username is provided through the graphical interface, TUMS will continuously start monitoring the tweets of the corresponding user. Whenever a new tweet is observed, the tweet will get processed to enrich the semantics of the tweet (see below). Tweets, news articles and semantic metadata, which is extracted from tweets and news, are stored in the data repository which builds the basis for user modeling.

User modeling Given the raw data crawled from Twitter, TUMS aims to infer user interest profiles that adhere to the Friend-Of-A-Friend (FOAF) [14] vocabulary and the Weighted Interest vocabulary [16]. Hence, given the raw text of tweets, TUMS will output profiles which specify how much a user is interested into a certain topic. For example:

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix wi: <http://purl.org/ontology/wi/core#> .
@prefix wo: <http://purl.org/ontology/wo/core#> .
@prefix ex: <http://example.org/> .
@prefix dbpedia: <http://dbpedia.org/resource/>
```

```
<http://twitter.com/taubau>
  a foaf:Person ;
  foaf:name "Ke Tao";
  wi:preference [
    a wi:WeightedInterest ;
    wi:topic dbpedia:Jazz ;
    wo:weight [
      a wo:Weight ;
```

```

    wo:weight_value 10.0 ;
    wo:scale ex:AScale
  ] .

```

The above profile snippet expresses that a user (<http://twitter.com/taubau>), whose name is “Ke Tao” is interested into jazz (*dbpedia:Jazz*). This interest is further classified via a weight (*wo:weight_value*) on a certain scale (*wo:scale*). In order to infer interests of a user from the tweets this user posted on Twitter, TUMS performs four main steps: (1) it notifies the tweet crawler to start collecting tweets, (2) it enriches the semantics of the user’s tweets by categorizing the topic of a tweet and extracting hashtags and other entities (e.g. persons, locations) from the tweet, (3) if possible, it links the tweet to related news articles to further enrich the semantics of a tweet with semantics extracted from the news article and (4) it applies different user modeling strategies (see Section 3.2) to generate user profile information.

The first three steps will be executed whenever a new Twitter message is observed while the last step, the actual user modeling, is executed on query time, i.e. when a user (via the graphical interface) or client application (via the Web service) requests profile information.

Graphical Interface and RDF Endpoint Given the data aggregated, processed and stored by the above modules, the Twitter-based user profiles can be retrieved by visiting the graphical user interface or by invoking the TUMS Web service API. The graphical interface enables end-users to overview their profiles by means of tag clouds (topic-based, hashtag-based and entity-based clouds) and diagrams (see Section 3.3). The REST-like Web service interface outputs user profiles in RDF using FOAF [14] and Weighted Interest vocabulary [16] as depicted in the above example.

3.2 User Modeling Strategies

The goal of TUMS and the user modeling strategies in particular is to generate user interest profiles that conform to the following model.

Definition 1 (User Profile). *The profile of a user $u \in U$ is a set of weighted concepts where with respect to the given user u for a concept $c \in C$ its weight $w(u, c)$ is computed by a certain function w .*

$$P(u) = \{(c, w(u, c)) | c \in C, u \in U\}$$

Here, C and U denote the set of concepts and users respectively.

To generate such profiles, we developed a set of user modeling strategies [13] that vary in four design dimensions: (i) the type of profiles created by the strategies, (ii) the weighting scheme, (iii) the data sources exploited to further enrich the Twitter-based profiles and (iv) temporal constraints that are considered when constructing the profiles (see Table 1). The generic model for profiles representing users is specified in Definition 1.

As listed in Table 1, TUMS allows for three *types* of profiles that differ with respect to the type of concepts $c \in C$ for which we specify an interest weight (see

design dimension	design alternatives
profile type	(i) hashtag-based, (ii) topic-based or (iii) entity-based
weighting schemes	(i) term frequency (TF) or (ii) $TF \times IDF$
enrichment	(i) tweet-only-based enrichment or (ii) linkage and exploitation of external news articles (propagating entities/topics)
temporal constraints	(i) specific time period(s), (ii) temporal patterns (<i>weekend</i> , <i>night</i> , etc.) or (iii) no constraints

Table 1. Design space of Twitter-based user modeling strategies.

Definition 1): entity-based, topic-based and hashtag-based profiles. For entity-based profiles, we differentiate between 39 types of entities such as persons, organizations, cities, countries, events, or medical conditions. The topic-based profiles are rather broad and abstract from the concrete content as we map the tweets to 19 static topics such as sports, politics or music.

TUMS allows for different methods as weighting scheme $w(u, c)$. For example, using term frequency (TF), the weight of a concept is determined by the number of Twitter activities in which user u refers to concept c . In a hashtag-based profile $w(u, \#technology) = 5$ means that u published five Twitter messages that mention “#technology”. Other weighting methods such as $TF \times IDF$ are possible as well and a detailed comparison of different weighting schemes is planned for future work. The resulting user profiles will be normalized so that the sum of all weights in a profile is equal to 1: $\sum_{c_i \in C} w(u, c_i) = 1$. With $\mathbf{p}(u)$ we refer to $P(u)$ in its vector space model representation, where the value of the i -th dimension refers to $w(u, c_i)$.

Twitter messages posted by a user u may refer to external resources. A user can explicitly link to other Web resources in her Twitter message or she could discuss topics and events that are, for example, discussed in mainstream news articles as well. TUMS aims to also take these external resources into account when constructing entity-based and topic-based user profiles (see semantic enrichment in Table 1). In particular, profiles are enriched with entities and topics extracted from news articles that are linked with Twitter messages. In previous work, we have done the study on selecting appropriate news articles for enriching tweets [12]. We revealed that for Twitter messages, which do not explicitly link to external Web resources we can find related news articles that report about the same event as the Twitter message with a high precision of more than 70%.

Temporal constraints are also considered as the fourth dimension of the profiles (see Table 1). By specifying temporal constraints, client applications can, for example, retrieve the latest profile of a user or a profile that is only based on Twitter activities which a user performed on weekends. In previous work [13], we revealed, for example, that there are significant differences between user profiles created on the weekends with those created during the week.

By selecting and combining the different design dimensions and alternatives, TUMS can generate different profiles in the form of visualized charts and RDF triples.



Fig. 2. Graphical user interface on the start page of TUMS

3.3 Graphical User Interface of TUMS

End-users can easily access TUMS via its graphical user interface which requires only a Twitter username as input and is thus easy to use. Figure 2 shows the start screen that is displayed when users are accessing TUMS. In the given example, the profile of a user named “USAGodG20” should be returned. By clicking on the “Get Tweet Profile” button the user will be directed to the next page that allows the user to overview and inspect the profile of USAGodG20 (see Fig. 3).

TUMS can represent user profiles by means of different visualizations, including hashtag-based clouds, pie charts visualizing topic-based profiles and entity types referenced by a user, and entity-based clouds (see Fig. 3). In the following, we examine the implemented visualization features of TUMS and describe them by example results that we obtain for requesting the profile of the user “USAGodG20”.

Hashtag-based clouds The hashtag-based clouds represent the hashtag-based profile of a user and thus visualize the hashtags which the corresponding user mentioned in her Twitter messages. Different hashtags will be displayed in different font size depending on their weight within the hashtag-based profile (cf. $w(u, c)$ in Definition 1). The higher the weight $w(u, c)$ within the profile $P(u)$ the bigger the font size. Hashtags displayed within the tag cloud are hyperlinks and direct the user to the Twitter search which will display the latest tweets that mention the hashtag.

For example, at the top of Fig. 3 we see the hashtag cloud of “USAGodG20”. The visualization of the hashtag-based profile tells us that “USAGodG20” is particularly interested in the news about globalization, NAFTA³, and political leaders in the United States, e.g. Bush, Obama.

Pie charts for topic-based profiles One of the reasons for the popularity of Twitter lies in the idea that anyone can contribute and discuss her/his opinion about any topic. TUMS generates topic-based profiles and the TUMS user interface provides a pie chart that displays the relative importance of a topic

³ NAFTA stands for North American Free Trade Agreement. It is an agreement signed by the governments of Canada, Mexico, and the United States, creating a trilateral trade bloc in North America.

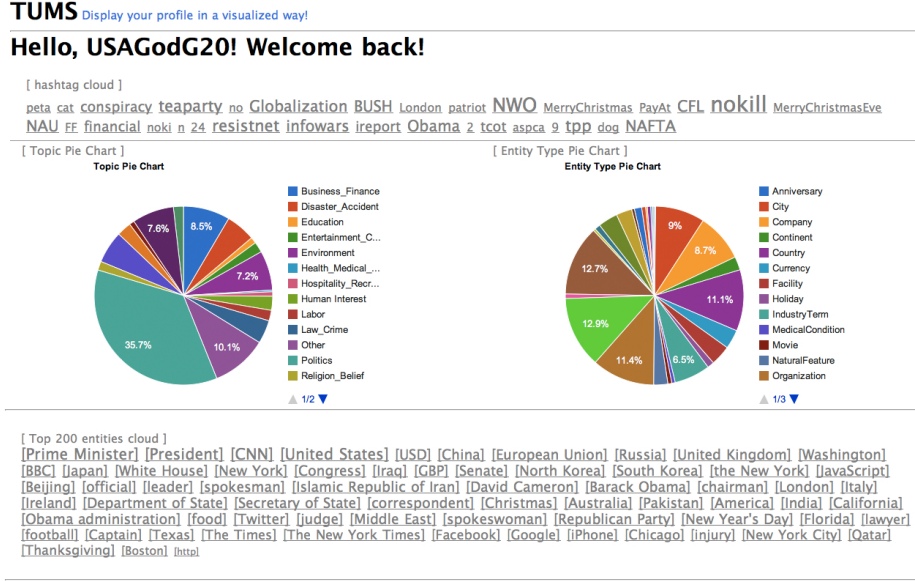


Fig. 3. Overview of user profile page of TUMS

within a user profile. Topic-based profiles feature 19 different dimensions that correspond to broad topics like politics or education. Given the pie chart visualization of a topic-based profiles, we can see about which topics a user has been published most.

Fig. 4 zooms into the pie chart of USAGodG20's profile and shows that 35.7% of his tweets are concerned with politics. This observation is consistent with the results we got from the hashtag-based cloud. Via the pie chart we can moreover compare this interest with interests for other topics. For example, we see that the user is also interested in "Business & Finance" (8.5%) but rather not interested in education or entertainment and culture.

Pie charts of referenced entity types During the user modeling process, entities are extracted from tweets and external sources for enriching the semantic of Twitter messages. TUMS extracts 39 types of the entities. The pie chart of entity types shows the importance of the entity types for a given user and can be inferred from the entity-based profiles.

The example pie chart generated based on the entity-based profile of US-AGodG20 shows that USAGodG20 is often referring to persons (12.9%), organizations (11.4%), countries (11.1%) and cities (9%). Natural features (e.g. beach, ocean) or movies are less frequently mentioned.

Entity-based clouds With entity-based clouds, TUMS visualizes the entity-based profile of a user and thus lists the top entities mentioned by a user. It refines the entity type based pie chart overview and shows particular persons, locations, etc. in which a user is interested. As for the hashtag-based cloud,

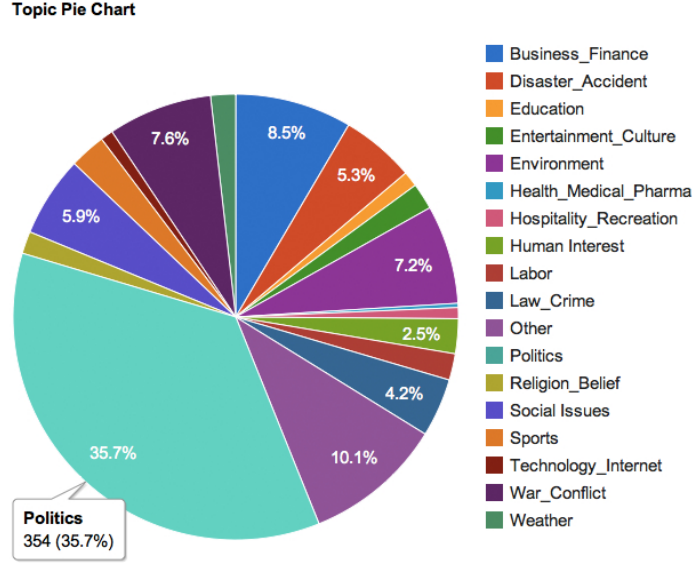


Fig. 4. The screenshot of the pie chart of topics

entities are represented in different font size according to their weight $w(u, c)$ in the entity-based profile (cf. Definition 1).

At the bottom of Fig. 3 we see the most important entities within the entity-based profile of USAGodG20. Entities with larger font size in the cloud, such as “Prime Minister”, “President”, “United States”, “China”, and “European Union” are words that are quite common within the political domain which confirms again the findings from the above profile visualizations.

3.4 RDF Endpoint of TUMS

User profiles are furthermore made available via Web service in RDF format. Via HTTP content negotiation⁴ or by specifying the format in the HTTP query string (*format=rdf*), client applications are enabled to request the different types of profiles generated by TUMS in RDF. The URL pattern that represents a query for profile information is defined as follows:

```
.../profile/[username]/[profile type]/[weighting]/[enrichment]/?[temporal constraints]
```

Hence, in addition to the *username*, which refers to the Twitter username of the person whose profile should be returned, there are four types of parameters which can be customized in the pattern. These four parameters correspond to the four dimensions in Table 1. The format and possible values of these parameters are explained below.

⁴ <http://tools.ietf.org/html/rfc2616#section-12>

profile type The profile type parameter enables applications to specify which type of profile they would like to retrieve: hashtag-based, topic-based or entity-based. Given a user profile $P(u) = \{(c_1, w(u, c_1)), (c_2, w(u, c_2)), \dots\}$ where $c_i \in C$, the profile type thus specifies the type of the concepts c_i . While topics and entities are represented via a URI by nature and therefore explicitly describe the semantic meaning of a topic and entity respectively, we try to clarify the semantic meaning of hashtags by referring to *#tagdef*⁵. Given the three types of profiles, the corresponding parameter values that can be used in the HTTP request are the following: *hashtag*, *topic*, and *entity*. For example, the hashtag-based profile can be retrieved as follows:

```
.../profile/USAGodG20/hashtag/
```

The RDF-formatted response to this HTTP request might be the following:

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix wi: <http://purl.org/ontology/wi/core#> .
@prefix wo: <http://purl.org/ontology/wo/core#> .
@prefix tums: <http://wis.ewi.tudelft.nl/tums/rdf/> .

<http://twitter.com/USAGodG20>
  a foaf:Person;
  foaf:name "John of God";
  wi:preference [
    a wi:WeightedInterest ;
    wi:topic <http://tagdef.com/conspiracy> ;
    wo:weight [
      a wo:Weight ;
      wo:weight_value 0.0757 ;
      wo:scale tums:Scale
    ] ;
  wi:preference [
    a wi:WeightedInterest ;
    wi:topic <http://tagdef.com/Globalization> ;
    wo:weight [
      a wo:Weight ;
      wo:weight_value 0.1576 ;
      wo:scale tums:Scale
    ] .
```

Hence, the hashtag-based profile of USAGodG20, which is in the graphical user interface represented by means of a tag cloud (see above), is represented using FOAF [14] and the Weighted Interest vocabulary [16]. For each concept (*wi:topic*) in the profile, the weight is clearly specified using *wo:weight_value*. TUMS normalizes the weights in a profile vector so that the sum of all weights in the profile is equal to 1. Correspondingly, the scale (*tums:Scale*) on which the weights are specified ranges from 0 to 1 (*wo:min_weight* = 0, *wo:max_weight* = 1, cf. Weighting Ontology⁶).

⁵ <http://tagdef.com>

⁶ <http://purl.org/ontology/wo/core>

weighting TUMS supports different weighting schemes for computing the weights in a user profile (cf. $w(u, c)$ in Definition 1). At the moment, TUMS supports (i) term frequency (TF) and (ii) TF multiplied by inverse document frequency ($TF \times IDF$). The corresponding parameters are *tf* and *tfidf* so that a request may have the following format:

```
.../profile/USAGodG20/hashtag/tf/
```

enrichment Currently, the possible options for this parameter are: (i) *tweet* or (ii) *tweetnews*. By default, TUMS extracts semantics from Twitter messages (*tweet*). However, developers can specify whether external sources should also be exploited to further enrich the semantics of tweets when constructing the profile (*tweetnews*). If no parameter is specified then TUMS aims to provide the best possible profile and thus will return the profile that is further enriched with semantics extracted from related news articles (*tweetnews*). In future versions of TUMS we plan further options for adjusting the semantic enrichment such as means for selecting the target ontology of concepts that are referenced as *wi:topic* (e.g. developers should be enabled to decide whether these concepts should be mapped to DBpedia concepts). Furthermore, we plan to parametrize the linkage strategies that are used for the news-based enrichment (*tweetnews*).

temporal constraints Temporal constraints of profiles can be specified in the URL-based query as well. For specifying a certain time period, developers can use *start* and *end* to specify the start and end of the period that should be considered when generating the profiles. For example, the profile from December 1st 2010 to February 28th 2011 can be retrieved via:

```
.../profile/USAGodG20/hashtag/?start=2010-12-01&end=2011-02-28
```

Moreover, it is possible to specify pattern-like time constraints like weekend or night. They can be specified by adding further parameters to the query string. At the moment, TUMS allows for a *wk* parameter which can be *weekend* (Twitter activities on Saturdays or Sundays) or *weekday* (Monday till Friday) and a *dk* parameter which can be *night* (from 6pm till 6am) or *day* (from 6am till 6pm). For example:

```
.../hashtag/?start=2010-12-01&end=2011-02-28&wk=weekend&dn=night
.../hashtag/?wk=weekday
```

If no temporal constraints are specified then TUMS will create the profiles based on all available Twitter activities performed by the user.

3.5 Preliminary Evaluation of TUMS

Given tweets of 1619 users for which TUMS created user profiles, we randomly selected 50 entity and 50 topic assignments that were inferred by TUMS as part of the semantic enrichment. Ratings were done on a three-dimensional scale. For entity extraction, 3 (*perfect match*) means extracted entity has the name and

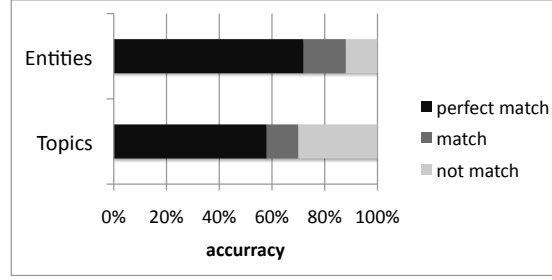


Fig. 5. Accuracy of entity extraction and topic detection.

type which exactly match with its meaning in the tweet; 2 (*match*) stands for that there might be some noisy words in the name of the entity, but the part other than the noise still matches with its real meaning, and the type the entity is correct; 1 (*not match*) denotes that the entity doesn't match the meaning in the tweet. For topic detection, the rating of 3 means the detected topic perfectly fits with the content of the tweet; the tweet rated as 2 has the content partly related to with the assigned topic; the wrong topic detection will be rated as 1.

Figure 5 lists the results and shows that, nearly 90% of the entity assignments and 70% percent topic detection are rated as 3 (*perfect match*) or 2 (*match*). Among these meaningful semantic enrichment, 82% of the entity assignments and 80% of the topic detection exactly match the meaning in the tweets.

3.6 Application Scenarios

There exist several applications that will benefit from using TUMS. In particular, systems that aims for personalization and are interested in fresh profile information such as news recommender systems might be interested in the Twitter-based profiles delivered by TUMS. Moreover, the visualization of the profiles can help adaptive systems to explain their users why certain adaptation decisions have been done, e.g. why an item was recommended.

Recommending Fresh Content Liu et al.[25] analyzed content-based recommender for Google News. They showed that interests in news topics such as technologies, politics, etc. change over time. Hence, the latest interests of a user are beneficial for building a personalization system. Given that most tweets on Twitter are related to news [3], Twitter seems to be an interesting source of information and TUMS the corresponding service that can support systems that are interested in current interests of a user. In [13] we report on first results in recommending news articles based on TUMS profiles. The TUMS profile can also be used for generating personalized email, recommending webpages, etc.

Explaining Recommendations The semantic user profiles and the visualizations of the profiles provided by TUMS can be applied to explain why

recommendations have been generated. Such explanations may foster acceptance of recommendations. For example, for some e-shopping website like Amazon⁷ it is common to give an explanation why a product was recommended. Using TUMS, recommender systems could link to the profile of a user that they used in order to compute the recommendation.

Authoring Support Services like Zemanta⁸ or SMOB [26] enable authors of (micro)blogs to attach semantics to their posts. Since TUMS user profiles such as entity-based profiles can be used to support the user when annotating their blog and microblog posts with semantically meaningful URIs.

Privacy Awareness Twitter users may not know what can be derived from their Twitter activities. Hence, the graphical components of TUMS that allow for exploring the Twitter-based profiles in a more structured way can foster user awareness.

4 Conclusion and Future Work

In this paper, we present TUMS, a Twitter-based User Modeling Service, that generates semantic user profiles by exploiting Twitter messages. Based on functionality for enriching the semantics of tweets, TUMS features a variety of user modeling strategies that produce entity-based, topic-based or hashtag-based profiles. TUMS makes these profiles available to other applications via an RDF endpoint and allows end-users to explore profiles visually. Given a Twitter username, TUMS aggregates tweets from Twitter and starts monitoring the Twitter user. All tweets published by the user will be processed by the semantic enrichment module of TUMS which, for example, extracts entities from tweets and links tweets to related news articles to further enrich the semantics of a tweet. Given the semantically enriched tweets, TUMS can be queried to return semantic profiles for specific periods in time.

In our future work, we plan to support platforms that aim for personalization with TUMS user modeling capabilities. First experiments show that entity-based user profiles generated by TUMS, allow for high precision when recommending news articles [13]. Our ambition is to make this news recommender available as Web application using TUMS for generating user profiles.

Acknowledgements. This work is partially sponsored by the EU FP7 projects ImREAL (<http://imreal-project.eu>) and GRAPPLE (<http://grapple-project.org>).

References

1. Jameson, A.: Adaptive interfaces and agents. The HCI handbook: fundamentals, evolving technologies and emerging applications (2003) 305–330

⁷ <http://www.amazon.com>

⁸ <http://www.zemanta.com>

2. Brusilovsky, P., Kobsa, A., Nejdl, W., eds.: The Adaptive Web, Methods and Strategies of Web Personalization. In Brusilovsky, P., Kobsa, A., Nejdl, W., eds.: The Adaptive Web. Volume 4321 of Lecture Notes in Computer Science., Springer (2007)
3. Kwak, H., Lee, C., Park, H., Moon, S.: What is twitter, a social network or a news media? In: Proceedings of the 19th International Conference on World Wide Web (WWW '10), New York, NY, USA, ACM (2010) 591–600
4. Krishnamurthy, B., Gill, P., Arlitt, M.: A few chirps about twitter. In: Proceedings of the first workshop on Online social networks. WOSP '08, New York, NY, USA, ACM (2008) 19–24
5. Java, A., Song, X., Finin, T., Tseng, B.: Why we twitter: understanding microblogging usage and communities. In: Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis. WebKDD/SNA-KDD '07, New York, NY, USA, ACM (2007) 56–65
6. Weng, J., Lim, E.P., Jiang, J., He, Q.: Twittrrank: finding topic-sensitive influential twitterers. In Davison, B.D., Suel, T., Craswell, N., Liu, B., eds.: Proceedings of the Third International Conference on Web Search and Web Data Mining (WSDM '10), New York, NY, USA, ACM (2010) 261–270
7. Cha, M., Haddadi, H., Benevenuto, F., Gummadi, P.K.: Measuring User Influence in Twitter: The Million Follower Fallacy. In Cohen, W.W., Gosling, S., eds.: Proceedings of the Fourth International Conference on Weblogs and Social Media (ICWSM '10), Washington, DC, USA, The AAAI Press (2010)
8. Chen, J., Nairn, R., Nelson, L., Bernstein, M., Chi, E.: Short and tweet: experiments on recommending content from information streams. In: Proceedings of the 28th international conference on Human factors in computing systems (CHI '10), New York, NY, USA, ACM (2010) 1185–1194
9. Gaffney, D.: #iranElection: quantifying online activism. In: Proceedings of the WebSci10: Extending the Frontiers of Society On-Line. (2010)
10. Diakopoulos, N.A., Shamma, D.A.: Characterizing debate performance via aggregated twitter sentiment. In: Proceedings of the 28th international conference on Human factors in computing systems. CHI '10, New York, NY, USA, ACM (2010) 1195–1198
11. Matthew Rowe and Milan Stankovic and Philippe Laublet: Mapping Tweets to Conference Talks: A Goldmine for Semantics. In Passant, A., , Breslin, J., Fernandez, S., Bojars, U., eds.: Workshop on Social Data on the Web (SDoW '10), colocated with ISWC '10, Shanghai, China. Volume 664., CEUR-WS.org (2010)
12. Abel, F., Gao, Q., Houben, G.J., Tao, K.: Semantic Enrichment of Twitter Posts for User Profile Construction on the Social Web. In: Extended Semantic Web Conference (ESWC '11), Heraklion, Greece, Springer (May 2011)
13. Abel, F., Gao, Q., Houben, G.J., Tao, K.: Analyzing User Modeling on Twitter for Personalized News Recommendations. In: International Conference on User Modeling, Adaptation and Personalization (UMAP), Girona, Spain, Springer (July 2011)
14. Brickley, D., Miller, L.: FOAF Vocabulary Specification 0.91. Namespace document, FOAF Project, <http://xmlns.com/foaf/0.1/> (November 2007)
15. Bojars, U., Breslin, J.G.: SIOC Core Ontology Specification. Namespace document, DERI, NUI Galway, <http://rdfs.org/sioc/spec/> (January 2009)
16. Brickley, D., Miller, L., Inkster, T., Zeng, Y., Wang, Y., Damjanovic, D., Huang, Z., Kinsella, S., Breslin, J., Ferris, B.: The Weighted Interests Vocabulary 0.5. Namespace document, Sourceforge (September 2010)

17. Kobsa, A.: Generic user modeling systems. *User Modeling and User-Adapted Interaction* **11**(1-2) (2001) 49–63
18. Berkovsky, S., Kuflik, T., Ricci, F.: Mediation of user models for enhanced personalization in recommender systems. *User Modeling and User-Adapted Interaction (UMUAI)* **18**(3) (2008) 245–286
19. Carmagnola, F., Cena, F.: User identification for cross-system personalisation. *Information Sciences: an International Journal* **179**(1-2) (2009) 16–32
20. Abel, F., Henze, N., Herder, E., Krause, D.: Interweaving public user profiles on the web. In Bra, P.D., Kobsa, A., Chin, D.N., eds.: *User Modeling, Adaptation, and Personalization*, 18th International Conference, UMAP 2010, Big Island, HI, USA, June 20-24, 2010. Proceedings. Volume 6075 of *Lecture Notes in Computer Science.*, Springer (2010) 16–27
21. Mehta, B.: Learning from what others know: Privacy preserving cross system personalization. In Conati, C., McCoy, K.F., Paliouras, G., eds.: *User Modeling*. Volume 4511 of *Lecture Notes in Computer Science.*, Springer (2007) 57–66
22. Sakaki, T., Okazaki, M., Matsuo, Y.: Earthquake shakes twitter users: real-time event detection by social sensors. In: *Proceedings of the 19th international conference on World wide web. WWW '10*, New York, NY, USA, ACM (2010) 851–860
23. Lerman, K., Ghosh, R.: Information contagion: an empirical study of spread of news on digg and twitter social networks. In: *Proceedings of 4th International Conference on Weblogs and Social Media (ICWSM)*. (May 2010)
24. Dong, A., Zhang, R., Kolari, P., Bai, J., Diaz, F., Chang, Y., Zheng, Z., Zha, H.: Time is of the essence: improving recency ranking using twitter data. In: *WWW '10: Proceedings of the 19th international conference on World wide web*, New York, NY, USA, ACM (2010) 331–340
25. Liu, J., Dolan, P., Pedersen, E.R.: Personalized news recommendation based on click behavior. In Rich, C., Yang, Q., Cavazza, M., Zhou, M.X., eds.: *Proceeding of the 14th international conference on Intelligent user interfaces (IUI '10)*, New York, NY, USA, ACM (2010) 31–40
26. Passant, A., Hastrup, T., Bojars, U., Breslin, J.: Microblogging: A Semantic Web and Distributed Approach. In Bizer, C., Auer, S., Grimnes, G.A., Heath, T., eds.: *Proceedings of the the 4th Workshop Scripting For the Semantic Web (SFSW '08) co-located with ESWC '08*, Tenerife, Spain. Volume 368., CEUR-WS.org (2008)