Advances in Understanding, Mining, and Using People-Tags

Peyman Nasirifard

Submitted in fulfillment of the requirements for the degree of
Doctor of Philosophy

Supervisor:
Dr. Conor Hayes

Co-supervisor:
Dr. Vassilios Peristeras

Internal Examiner:
Prof. Dr. Stefan Decker

External Examiner:
Dr. Robert H.P. Engels

Digital Enterprise Research Institute, National University of Ireland, Galway
May 2012

The studies presented in this thesis were performed at the Digital Enterprise Research Institute at the National University of Ireland, Galway. The research was financially supported by European Union under Grant No. FP6-IST-5-35208 and FP7-257859 and Science Foundation Ireland under Grant No. SFI/02/CE1/I131 and SFI/08/CE/I1380.

Dedicated to my parents.

Abstract

People-tagging involves the process of adding non-hierarchical metadata to users of a system. Such metadata facilitates organising contacts and building user profiles in a collaborative fashion. People-tag-based user profiles can be used in use cases such as finding people with relevant expertise and filtering information. This thesis contributes to the initiative of people-tagging in three areas: a) a better understanding of how people-tags are used; b) automatic extraction, ranking and assigning people-tags to knowledge workers; and c) using tag-based profiles for an information propagation use case.

Due to a lack of sufficient studies on people-tagging behaviour in online social platforms, initially, we studied how users of social media and in particular social blogs tag each other. We extracted people-tags from such websites and classified them into several categories. Our analysis suggests that people-tagging in public online social platforms is highly subjective and this may lead to interoperability drawbacks between systems that operate on top of people-tags. Building domain-specific vocabularies as well as ranking tags are approaches that we considered to eliminate subjectivity of people-tags.

Current practices of tagging knowledge workers are manual processes that offer several disadvantages, such as increasing cognitive overhead for taggers and cold-start problem of people-tag-based systems. To address these issues, we developed approaches to (semi-) automatically extract, rank, and assign people-tags to knowledge workers. To this end, we extract metadata from collaborative platforms used by knowledge workers such as question–answering (Q–A) forums. We rank and assign such metadata to knowledge workers based on their contribution and collaboration history within collaborative platforms (e.g., solving an issue or providing helpful answers).

We use tag-based profiles for an information propagation use case. We developed an access control and in particular an information propagation model which enables end users to define information sharing policies on top of people-tags and a numeric value called distance. The distance value determines the propagation depth of a resource in a network of connected users. As users may need help in drafting appropriate policies for a given resource, we further equipped our model with a policy advisor component to assist users for sharing items such as URLs and community-related announcements. The main goal of the policy advisor is to eliminate information overload and information shortage within a network of connected users. Given an item and tag-based user profiles as input, our policy advisor is capable of analysing the item and recommending topic-sensitive hubs who may propagate information in the network, in order to eliminate information overload and information shortage.

All of our approaches are supported by prototypes that helped us to evaluate them with real-world data such as micro-blog posts and technical Q–A forums. The evaluation showed that our approaches help users to tag each other and to use tag-based profiles for a more user-centric information propagation model in social and collaborative platforms.

Keywords: Tagging, People-Tagging, Expert Finder, Information Filtering, Access Control, Policy Advisor, Recommender System, Social Media.

Acknowledgments

I have to thank a large number of people for their inspiration and support during the research and writing of my Ph.D. work and thesis. First of all, I would like to appreciate my main supervisor, Conor Hayes, for guiding me during my work. I have learned a lot from him. I would also like to thank Vassilios Peristeras who supervised and advised me during initial stages of my Ph.D. A special thanks goes to Stefan Decker for giving me the opportunity to pursue my Ph.D. at the amazing institute called DERI. I also appreciate Robert H.P. Engels for becoming the examiner of my work. Thanks to my fellow students and co-workers in DERI for the fertile discussions, advice, inspiration, proof-reading my work and playing table soccer during coffee breaks. I decided not to name them, as I might unintentionally forget couple of names, however, DERI's current team and alumni pages list all of them. Thanks to DERI SEC, technicians, outreach, and admin people for making DERI such a great place to work. Most importantly, thanks to my friends and family who have always encouraged me and motivated me with their warm words. And thanks to you who always helped me.

List of Publications

Journal Publications

- Peyman Nasirifard, Vassilios Peristeras, Stefan Decker. Annotation-Based Access Control for Collaborative Information Spaces. In Computers in Human Behavior, volume 27, issue 4, pp. 1352-1364, Elsevier, 2011. (5-year ISI Impact Factor: 2.15)

- Vassilios Peristeras, M. Antonia Martínez-Carreras, A. Fernando Gómez-Skarmeta, Wolfgang Prinz, Peyman Nasirifard. Towards a Reference Architecture for Collaborative Work Environments. In International Journal of e-Collaboration, Special Issue on Collaborative Work Environments, volume 6, issue 1, pp. 14-32, IGI Global Publishing, 2010.

Conference and Workshop Publications

- Peyman Nasirifard, Conor Hayes. Tadvise: A Twitter Assistant Based on Twitter Lists. In proceedings of the 3rd International Conference on Social Informatics (SocInfo'11), Springer, Singapore, Singapore, 2011.

- Peyman Nasirifard, Conor Hayes. A Real-Time Tweet Diffusion Advisor for #Twitter. In proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'11), ACM, Hangzhou, China, 2011.

- Peyman Nasirifard, Sheila Kinsella, Krystian Samp, Stefan Decker. Social People-Tagging vs. Social Bookmark-Tagging. In proceedings of the 17th International Conference on Knowledge Engineering and Knowledge Management (EKAW'10), Springer, Lisbon, Portugal, 2010. (Acceptance rate: 28%)

- Peyman Nasirifard, Vassilios Peristeras, Conor Hayes, Stefan Decker. Extracting and Utilizing Social Networks from Log Files of Shared Workspaces. In proceedings of the 10th IFIP Working Conference on Virtual Enterprises (PRO-VE'09), Springer, Thessaloniki, Greece, 2009.

- Peyman Nasirifard, Michael Hausenblas, Stefan Decker. Privacy Concerns of FOAF-Based Linked Data. In proceedings of the Trust and Privacy on the Social and Semantic Web (SPOT), Workshop at the 6th European Semantic Web Conference (ESWC'09), Crete, Greece, 2009.

- Peyman Nasirifard, Vassilios Peristeras. Expertise Extracting Within Online Shared Workspaces. In proceedings of the Web Science: Society On-Line Conference (WebSci'09), Athens, Greece, 2009.

- Peyman Nasirifard, Vassilios Peristeras. Uncle-Share: Annotation-Based Access Control for Cooperative and Social Systems. In proceedings of the 3rd International Symposium on Information Security (IS'08), Springer, Monterrey, Mexico, 2008.

- Peyman Nasirifard, Vassilios Peristeras. Annotation-Based Access Control for e-Professionals. In proceedings of the 14th International Conference on Concurrent Enterprising (ICE'08), Lisbon, Portugal, 2008.
- Peyman Nasirifard. Anatomy of a Semantic Virus. In proceedings of the Nature inspired Reasoning for the Semantic Web (NatuReS), Workshop at the 7th International Semantic Web Conference (ISWC'08), Karlsruhe, Germany, 2008.
- Peyman Nasirifard, Slawomir Grzonkowski, Vassilios Peristeras. OntoPair: Towards a Collaborative Game for Building OWL-Based Ontologies. In proceedings of the Collective Semantics: Collective Intelligence and the Semantic Web (CISWeb), Workshop at the 5th European Semantic Web Conference (ESWC'08), Tenerife, Spain, 2008.
- Peyman Nasirifard, Vassilios Peristeras. Leveraging Access Control in CSCW based on User-Defined and Hidden Semantic Social Networks (position paper). In proceedings of the CSCW and the Web 2.0, Workshop at the 10th European Conference on Computer Supported Co-operative Work (ECSCW'07), Limerick, Ireland, 2007.
- Peyman Nasirifard. Context-Aware Access Control for Collaborative Working Environments Based on Semantic Social Networks. In proceedings of the Doctorial Consortium, Workshop at the 6th International and Interdisciplinary Conference on Modeling and Using Context (Context'07), Roskilde, Denmark, 2007.

Patent

- Peyman Nasirifard, Vassilios Peristeras, Stefan Decker. A System for Annotation-Based Access Control (Pending - Filed on 22 May 2009 with the United States Patent and Trademark Office (USPTO), Docket No. 105437.61622US)

Contents

List of Figures

List of Tables

Part I

Prelude

Chapter 1

Introduction

> Computer science is no more about computers than astronomy is about telescopes.
>
> Edsger Wybe Dijkstra

Tagging is a light-weight approach for adding non-hierarchical metadata to different objects. Such metadata can be used in different use cases such as knowledge discovery and knowledge management. The practice of tagging can be extended to support tagging people. In brief, people-tagging can be defined as adding non-hierarchical metadata to people and is based on the success stories of tagging; and in particular collaborative tagging [Golder and Huberman, 2006]. In this thesis, we focus on the initiative of people-tagging and its capabilities within social and collaborative platforms. People-tagging enables users to add metadata to each other (including themselves). Such metadata may describe a wide range of attributes such as expertise, interests, affiliations, etc. and can be used in different use cases such as improving communication and dynamic team building. This is important as people are primary assets of organisations [Mayo, 2001] and every effort that is made to improve communication and collaboration among people will eventually improve organisational growth.

The properties of people-tag-based metadata may differ from the properties of metadata that is assigned to other objects such as photos, as people are not static and they evolve, e.g., by gaining new expertise, changing affiliation and so on. Thus, it is important to study how people tag each other in order to enhance usability of people-tag-based systems. The people-tagging practice is currently used in platforms like Fringe Contacts [Farrell and Lau, 2006, Farrell et al., 2007] as a means to organise contacts and build user profiles. Grouping people into different categories (e.g., within Facebook[1] or Google+[2]) can be also perceived as a way of tagging them. While grouping people focuses on finding and assigning a common characteristic that all people within a group share (such as friendship), tagging a person can be more fine-grained, as it focuses on detailed characteristics of that particular person.

[1] http://facebook.com/
[2] http://plus.google.com/

In this thesis, we show how users of social media tag each other in terms of general subjective/objective categories [Sen et al., 2006]. While earlier work analysed people-tagging behaviour in corporate environments [Farrell and Lau, 2006, Farrell et al., 2007], our work sheds some light on people-tagging behaviour in public online social platforms.

As people-tagging is a new practice, we aim to help users to find appropriate tags for tagging their contacts or even themselves. To this end, we present our approach for extracting, ranking, and assigning people-tags to knowledge workers who collaborate within online platforms such as question–answering forums or online shared workspaces. Such people-tags may represent expertise elements of the users who collaborate in such platforms.

We then use tag-based user profiles in an information sharing/filtering use case. We develop an information sharing/filtering model in which users can define information sharing policies on top of people-tags. We show that such information sharing models, if equipped with suitable policy recommenders, provide a more effective mechanism for propagating information within a network of connected people by reducing information overload and information shortage. The main task of the policy recommender in our work is to assist users in finding well-connected topic-sensitive hubs who may propagate community-related information in the network and thus reduce information overload as well as information shortage.

All of our approaches are supported by relevant prototypes and this has given us valuable feedback for evaluating our approaches. In particular, we used real-world data such as real-time micro-blog posts and technical question–answering forums for evaluating our approaches. The evaluations suggest that our approaches and prototypes help users to semi-automatically build people-tag-based user profiles and later use such profiles for filtering information as well as finding suitable hubs for effective information propagation.

The remainder of this chapter proceeds as follows: In Section 1.1, we introduce the main research questions that this thesis addresses. In Section 1.2, we briefly present our contributions and approaches to these research questions. In Section 1.3, we present the structure of this thesis and then the impacts made by this thesis in Section 1.4.

1.1 Research Questions

Through this thesis, we address the following major research questions:

- `Are there any differences between tags assigned to a person (i.e., people-tag) versus tags assigned to a bookmark?`

  Collaborative tagging has been well studied in various platforms where there exist a built-in feature for tagging items such as bookmarks and photos [Marlow et al., 2006, Golder and Huberman, 2006, Halpin et al., 2007]. There exist studies on people-tagging behaviour inside corporations [Farrell and Lau, 2006, Farrell et al., 2007, Schmidt and Braun, 2008] which show that users tend to assign more role-based tags to others. Yet, people-tagging behaviour inside public online social networks are not well studied. Are there any differences between tagging a friend and a bookmark? In other words, can we expect that a user tags his/her friends the same way as s/he tags a Wikipedia article that s/he recently read? Addressing this question is important, as this gives us a better understanding on people-tagging behaviour within social platforms in

order to build vocabularies/approaches that can be used for recommending appropriate and relevant people-tags.

- `Can we (semi-) automatically extract, rank and assign people-tags to knowledge workers?`

  Having up-to-date people-tag-based profiles offer various advantages such as forwarding relevant information to relevant people and building dynamic teams composed of people with relevant expertise [Farrell et al., 2007]. However, current people-tagging activities are rather manual tasks in which users give prior information about themselves/others to the system. Despite the fact that there exist vocabularies such as RELATIONSHIP [Davis and Vitiello, 2005] and REL-X [Carminati et al., 2006a] to assist users to tag each other using social relationships, this practice alone is not sufficient and requires complementary approaches due to several reasons. First of all, vocabularies have normally limited scope and thus users may not be able to select a fine-grained tag for a person. In other words, this practice is similar to grouping people which may not be necessarily fine-grained. Next, choosing an appropriate term from a vocabulary could be a time-consuming task for users. If we could (semi-) automate the process of extracting, ranking and assigning people-tags, for example, by extracting metadata from resources that users worked on, this could reduce the overhead for users and moreover address the cold-start problem of people-tag-based (recommender) systems. Ranking tags is an important requirement as a single user may not be a good judge for introducing himself/herself/others in terms of capabilities, expertise, etc. For example, if someone states that s/he is an expert in programming languages, this statement requires some sort of ranking mechanism in order to eliminate subjectivity.

- `Can we utilise people-tag-based user profiles for a more user-centric access control and in particular information propagation model within social platforms?`

  Most current knowledge workers rely on online shared workspaces such as Microsoft SharePoint[3] or OrbiTeam BSCW[4] and/or Web 2.0 services such as micro-blogging to share items, propagate information and generally collaborate together. Current access control mechanisms within online shared workspaces as well as Web 2.0 services suffer from flexibility and fine-granularity [Hart et al., 2007]. For example, users are able to share an online object with all their *friends*, but sharing an item with only *close friends*, is cumbersome, if possible. As an another example, consider a user that finds interesting news related to a topic, e.g., *marketing*, and she would like to share it with relevant colleagues (i.e., those who should or are interested to see that news). Sending the *marketing* news to all contacts in his/her address book as well as sending it to a subset of them offers several drawbacks. The main problem of sending the news to all contacts is the possibility of information overload, as uninterested contacts will also get the news. On the other hand, the main disadvantage of sending the news to a subset of the contacts is the possibility of information shortage, as several relevant users may be unintentionally dropped out from the recipients. The reason for such drawbacks is perhaps due to the fact that access control and in particular information propagation mechanisms

[3] http://sharepoint.microsoft.com
[4] http://www.bscw.de/

that exist in current online collaborative platforms were inspired from approaches such as role-based access control (RBAC) [Ferraiolo and Kuhn, 1992, Sandhu et al., 1996] which were developed decades ago for addressing requirements of mainly desktop environments, in which users are assigned to pre-defined and possibly hierarchical roles (e.g., root, guest) with pre-defined permissions. Such models are too coarse-grained to be used in current Web 2.0 platforms, where users are connected in a social network (i.e., graph) which may not be necessarily hierarchical. Can people-tag-based user profiles be used for building a more user-centric and fine-grained information propagation model among knowledge workers connected in a social network? What characteristics of a social network should be considered when building such models? Addressing these issues improve communication among knowledge workers.

In the following section, we briefly explain our approach to address above research questions.

1.2 Contributions

This thesis presents the following contributions that are both conceptual and also concrete implementations to address the questions that we posed in the previous section. From the conceptual point of view, we provide the following major contributions:

- To address the first question, we provided insight into whether people tag their friends the same way as they tag a bookmark. To this end, we gathered five different sets of tags: one set was composed of the tags that were assigned to friends on several public online social blogs and the other four sets were composed of the tags that were assigned to Wikipedia articles related to famous people (e.g., celebrities), events, cities and countries. We then asked several subjects to categorise the extracted tags based on a scheme inspired from [Sen et al., 2006]. We showed that people use more subjective tags for tagging their friends, whereas more objective tags for tagging a bookmark. This behaviour may lead to interoperability drawbacks between people-tag-based (recommender) systems. Providing domain-specific vocabularies as well as ranking tags are approaches that can be considered to eliminate subjectivity of people-tags.

- To address the second question, we created a novel approach for extracting, ranking and assigning expertise elements (as people-tags) to knowledge workers who collaborate within online platforms. We applied our approach to technical question–answering (Q–A) forums. For the extraction phase, we used several different approaches for topic modelling such as Latent Dirichlet Allocation (LDA) [Blei et al., 2003]. Awarding incentives such as points is a common practice in Q–A forums, in order to encourage and motivate users to participate and solve issues. Our model benefits from such incentives and in particular points for ranking the extracted tags. The expertise elements are assigned to users based on their interaction and contribution history such as solving an issue or providing helpful answers to a question.

- To address the third question, we created an approach for filtering information (and generally access control) based on people-tags that can be used within collaborative and social platforms in order to reduce information overload and information shortage. Our approach is composed of a simple model and an advanced one. In the simple

model, users can tag each other using arbitrary terms and define access control policies to share resources (e.g., bookmarks). Each policy in our basic model is composed of one people-tag and one distance. The people-tag part aims to define those users who may get access to a resource and the distance is a positive integer value which determines the validity scope of a policy. For example, a distance of one from a user means direct contacts of the user, whereas a distance of two indicates contacts of the contacts of said user. Users can get access to a shared item if they meet the requirements specified in the sharing policy of that item (i.e., tag and distance criteria). In the advanced model, a policy may contain more features such as enabling users to define resource tags and use such tags for defining policies.

- Both people-tag (annotation) and distance in our information propagation model may be sources of imprecision due to several reasons such as annotation drift. Moreover, users may forget the tags that they have assigned to others. Thus, our model needs a policy advisor to assist users for sharing items such as URLs or propagating information such as announcements. To this end, we developed a policy advisor for our model which will be described in a separate chapter. The policy advisor takes people-tag-based user profiles and an item (such as a URL) as input and allows end users to explore who in their networks may be related to the main topics of that item. Moreover, it can recommend well-connected topic-sensitive hubs who may propagate information related to a topic to more interested users. In order to convince users that our advice is relevant, we provide explanations to end users.

From the prototyping point of view, we developed several tools as proof-of-concept for this thesis which will be explained in the following:

- We provide a plug-in for OrbiTeam BSCW, an online shared workspace, to (semi-) automatically build expertise profiles for users based on extracted tags from documents stored in OrbiTeam BSCW. We also provide a tool for extracting, ranking and assigning expertise elements to users of Q–A forums. We customised this tool for community network forums of a large corporation (i.e., SAP) which resulted to a prototype called Sapport. The Sapport tool is capable of recommending a ranked list of experts for a given topic and providing text-based as well as graphical explanations to demonstrate what happens behind the scene and why.

- We provide a widget called Uncle-Share that can be easily embedded into any widget platforms such as iGoogle[5] or personal websites such as blogs. This widget enables people to share bookmarks using our people-tag-based access control model. The functionalities of Uncle-Share are wrapped as Web services, so that other applications can benefit from our access control model within their tools.

- We provide a recommender system called Tadvise for the well-known micro-blogging service, Twitter, that helps users to know their communities better and to propagate their community-related tweets more effectively in Twittersphere. Tadvise can be seen as an instance of a policy advisor for our information propagation model by recommending well-connected topic-sensitive users who may act as hubs to propagate community-

[5] http://www.google.com/ig

related information to more interested users. Tadvise provides text-based as well as graphical explanations to convince end users about hubs that it recommends.

There is no global solution for addressing all problems. As our approaches are based on people-tag-based user profiles, a general limitation of our work is the privacy issues of people-tags. While many users find people-tagging practices useful and fun [Bernstein et al., 2009], there exist some users who will not use these techniques due to privacy concerns. We will elaborate on this and other potential limitations in the following chapters of the thesis.

1.3 Structure of This Thesis

Figure 1.1 illustrates an overall view of the thesis structure.

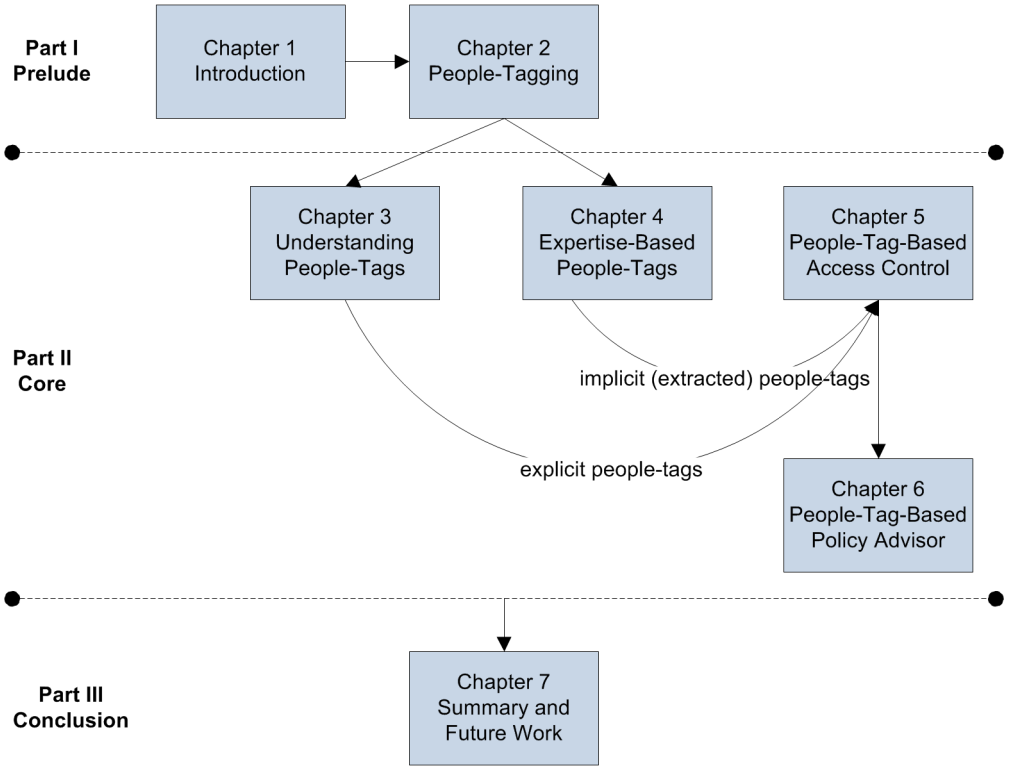

Figure 1.1: A schema of the thesis structure.

In Chapter 2, we present background information and an overview of basic concepts that we use in this thesis. Chapters 3 – 6 are core chapters of this thesis. Due to a lack of sufficient studies on people-tagging, in Chapter 3, we present an empirical study on the properties of people-tags that we gathered from social websites. We show that people-tags are subjective and every person tags others from his/her own cognitive perspective. This may create interoperability issues for applications that are based on people-tags. Ranking people-tags is an approach to eliminate such subjectivity. In Chapter 4, we provide a solution to extract, rank and assign people-tags to knowledge workers based on their document-oriented interaction styles (e.g., read, write, revise) as well as their contribution history within Q–A forums. Chapter 5 introduces an access control (information filtering) model that is built

on top of people-tags. Contrary to current well-adopted yet coarse-grained access control models such as RBAC, our approach aims to provide a more flexible access control model for sharing online resources in enterprise-oriented communities. Yet, due to various factors such as subjectivity of people-tags, this access control model can suffer from broadcasting too much information (information overload) or too little information (information shortage) to relevant people. Thus, it requires an advisor to assist users in drafting policies. In order to realise this policy advisor, we focused on micro-blogging which is currently used in many current enterprise 2.0 organisations for propagating small pieces of information. This gave us relevant real-time data for evaluating our real-time advisor. In Chapter 6, we describe aforementioned policy advisor and its prototype that was designed and built for Twitter. In Chapter 7, we present summary of this thesis and finally we conclude and have an overview of future work.

1.4 Impact

The core chapters of this thesis were published in various international peer-reviewed conferences and journals. The main content of Chapter 3 was published in proceedings of the 17th International Conference on Knowledge Engineering and Knowledge Management (EKAW'10 – 28% acceptance rate) [Nasirifard et al., 2010a]. The main approach of Chapter 4 was published in proceedings of the 10th IFIP Working Conference on Virtual Enterprises (PROVE'09) [Nasirifard et al., 2009] and also in proceedings of the WebSci'09: Society On-Line Conference (WebSci'09) [Nasirifard and Peristeras, 2009]. The main content of Chapter 5 was published in proceedings of the 14th International Conference on Concurrent Enterprising (ICE'08) [Nasirifard and Peristeras, 2008a], in proceedings of the 3rd International Symposium on Information Security (IS'08) [Nasirifard and Peristeras, 2008b], is protected by a pending patent[6] and finally in an Elsevier journal (Computers in Human Behaviour – 5-year ISI Impact Factor: 2.15) [Nasirifard et al., 2010b]. The main content of Chapter 6 was published in proceedings of the 3rd International Conference on Social Informatics (SocInfo'11) [Nasirifard and Hayes, 2011].

We also published and demonstrated earlier thoughts and ideas of this thesis in various symposiums and workshops such as in proceedings of the Doctorial Consortium at the 6th International and Interdisciplinary Conference on Modelling and Using Context (Context'07) [Nasirifard, 2007] and in proceedings of the CSCW and the Web 2.0 Workshop at 10th European Conference on Computer Supported Co-operative Work (ECSCW'07) [Nasirifard and Peristeras, 2007].

Several companies that are active in building collaborative software have shown interest in our work and started to adapt our approaches within their (commercial) products. Among others, OrbiTeam[7] has started to adapt our model within their commercial product (OrbiTeam BSCW shared workspace). To this end, they implemented the people-tagging feature (i.e., the first step of realising our information propagation model) in the latest release of OrbiTeam BSCW[8]. Moreover, our BSCW Expert Finder tool was successfully integrated

[6] Peyman Nasirifard, Vassilios Peristeras, Stefan Decker. A System for Annotation-Based Access Control (Pending – Filed on 22 May 2009 with the United States Patent and Trademark Office (USPTO), Docket No. 105437.61622US).

[7] http://www.orbiteam.de/

[8] http://www.bscw.de/english/bscw44.html

with TXT[9] TeamBuilder tool for importing expertise profiles of users into the TeamBuilder. TeamBuilder is a tool that assists users for building teams composed of people with different desired expertise. The Tadvise system was presented to Cisco Systems[10] and a plan has been agreed to adapt several of the concepts for demonstration use in their virtual presence communication platform.

Our approach for extracting, ranking and assigning people-tags to knowledge workers was developed and presented as part of the EU project ROBUST[11]. The information propagation model was developed and presented as part of the EU project Ecospace[12]. The Tadvise research was developed and presented as part of the Science Foundation Ireland (SFI)-funded LION II project [Decker and Hauswirth, 2008].

[9] http://www.txtgroup.com/
[10] http://www.cisco.com/
[11] http://robust-project.eu/
[12] http://www.ip-ecospace.org/

Chapter 2

People-Tagging

> When the novelty wears off, though, I think that tagging will have altered the information landscape in a fundamental way.
>
> Jon Udell

Tagging has become popular through online services that have allowed ordinary users to contribute to the content of the Web. Social bookmarking services on *delicious.com* and photo-sharing services on websites like *flickr.com* helped to accelerate the use of tagging as a new way of organising information. Users adopted this practice quickly and this has led to collective-tagging systems, so-called folksonomies on the Web. The term folksonomy is derived from combination of the words *folk* and *taxonomy* and defines the practice and method of collaboratively creating and managing tags to annotate and categorise content [Peters, 2009].

Recently, the practice of people-tagging has arisen which enables users to tag each other and build user profiles in a collaborative fashion. People-tagging is currently used in social and collaborative platforms for various purposes such as creating social links and expert-finding – the ability to retrieve a ranked list of experts on a topic in a community such as a group in an organisation.

In this chapter, we take a closer look at foundations of tagging and in particular people-tagging. We take a deeper look at advantages as well as limitations of the people-tagging practice. Initially, we study the relevant technologies of Web 2.0 which led to widespread adaption of tagging. Semantic Web technologies and standards make a bridge between machines and information by providing unique references to concepts, e.g., by using ontologies. In other words, Semantic Web technologies can be used to disambiguate meaning of ambiguous tags and people-tags. In this chapter, a brief introduction to the Semantic Web will be also presented.

The remainder of this chapter proceeds as follows. In Section 2.1, we take a closer look at Web 2.0 technologies that are relevant to this thesis. Section 2.2 focuses on Semantic Web and its role in making user-generated contents interoperable among applications. Section 2.3 presents a more detailed overview on tagging concepts. In Section 2.4, we present an

introduction to people-tagging, discuss its advantages as well as its limitations in building user profiles. Finally, we close the chapter with a conclusion in Section 2.5.

2.1 Web 2.0: Mass Collaboration and Communication on the Web

The term Web 2.0 refers to a group of loosely related technologies with one thing in common: user-generated content. In typical Web 2.0 systems the users provide content (e.g., videos, blog posts) and/or contribute together towards the same goal (e.g., Wikipedia[1]). [Anderson, 2007] mentions that user-generated content is one of the key ideas behind most Web 2.0 platforms and applications. In the following paragraphs, we have a brief overview of several Web 2.0 concepts and services that are most relevant to the scope of this thesis.

`Tagging and Annotations` – Adding metadata to objects has long being recognised as a way of organising information so that it can be indexed and retrieved later on. Tagging involves adding arbitrary non-hierarchical metadata – so-called tags – to a piece of information such as an image, a document, a song, a video and so on. As illustrated in Figure 2.1, a conceptual model of social tagging composes of three main elements: resource, user and tag [Marlow et al., 2006]. In this model, a user assigns a tag to a resource. As tagging involves adding metadata to a shared item and such practices can be performed by many users for a shared item, such behaviours are so-called collaborative tagging [Golder and Huberman, 2006]. The motivations behind tagging differ [Golder and Huberman, 2006, Ames and Naaman, 2007], however, such metadata helps faster retrieval of information and can be also used as a means to describe an item in more detail from the perspective of a tagger (i.e., who assigns the tags).

As tagging is an important concept of this thesis, in Section 2.3, we have a more detailed overview of tagging and its properties.

`Blog` – Blog or weblog comprises of a simple webpage that enables users to express their diary or opinions about various events in different categories: from technology to personal experiences. Contrary to old and slow models of publishing, blogs enable users to publish their thoughts and make them accessible to a large audience and perhaps receive immediate feedback. This lowers the cost and required time to publish and receive feedback. Moreover, the ability to cross-link various concepts to a blog post eases the process of finding supplementary information. For a better retrieval of blog posts, it is a common practice that author of the post chooses one or more appropriate tags for categorising the post. For more information on blogs, see [Blood, 2002, Hussey, 2010].

`Micro-blogging` – Micro-blogging is a form of blogging with the difference that its posts are typically much shorter than normal blog posts and have usually a limited length (e.g., 140 or 250 characters) in size. Current micro-blogging culture is shaped so that professionals use it to find appropriate information, news and collaboration opportunities [Brooks and Churchill, 2010]. Various studies [Ehrlich and Shami, 2010, Whitlock and Micek, 2008, Ojeda-Zapata, 2008] show the effectiveness of micro-blogging for improving communication within enterprises. It is also evident from the fact that large companies such as IBM are using their own internal micro-blogging platform. Like blogging,

[1] `http://www.wikipedia.org/`

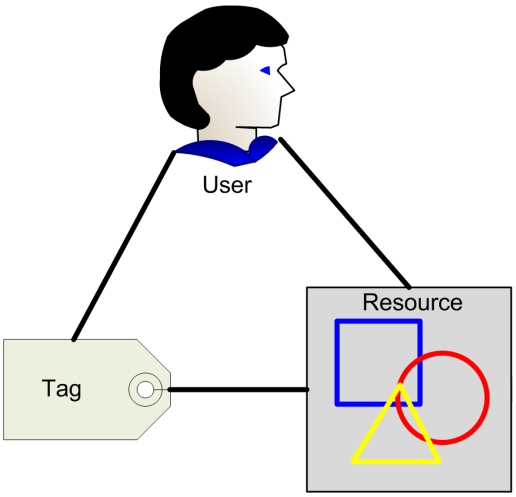

Figure 2.1: A conceptual model of tagging.

tagging is also used in micro-blogging platforms for a better organisation and faster retrieval of the posts. In micro-blogging jargon, such tags are sometimes refer to as *hashtags* and they are recognised by adding a special character such as a "#" to the tag.

`Social Network` – The concept of *social network* has its origins in social science. The small world phenomenon, based on Milgram's idea of six degrees of separation [Milgram, 1967], presents the concept that everyone on this planet is connected to all people in the world by a short chain of social relationships. In brief, "social networking involves the creation of a virtual community where users can share, discuss, collaborate, and even argue about topics of common interest" [Bernal, 2009]. Since its inception, the Internet has been a platform for several different types of virtual social networks such as Usenet groups, mailing lists and online forums. The latest social networking platforms such as *facebook.com* and *twitter.com* have millions of users and have developed services that now blur the distinction behind real and virtual social networks. It is very hard to put a distinguishable line between Web 2.0 and social networking platforms, however, many characteristics, features and technologies of Web 2.0 platforms enable delivery of social networking capabilities [Bernal, 2009]. Social networks can be represented as graphs of nodes/actors and edges between them. The edges can have weights associated with them which denote the strength of the links among the actors.

Sharing in general and multimedia-sharing in particular are key aspects of Web 2.0 platforms. Many current Web 2.0 platforms and social networks aim at providing services for sharing online items such as photos, videos, podcasts, audio blogging, etc. The concept of sharing brings privacy issues though and thus needs considerable attention with regards to access control. Access control is the ability to permit or deny the use of something by someone [Russell and Gangemi, 1991]. We use an extension of tagging practice (i.e., people-tagging) to develop a more user-centric access control model for social platforms. We present an overview of current approaches for access control, their drawbacks and advantages of our tag-based model in Chapter 5.

2.2 Semantic Web

In contrast to the informal methods of assigning metadata, Semantic Web tries to make such metadata interoperable and less ambiguous. For example, a tag like *apple* is ambiguous, as it may refer to an apple as a fruit or apple corporation as a company name. The Semantic Web [Berners-Lee et al., 2001] as an extension to the Web is actually a set of technologies and standards which tries to help machines to understand concepts and derive new information based on existing well-defined information. Using Semantic Web, software engineers are able to build interoperable systems that can benefit from the capabilities of machines to combine data and reason over existing data to infer new information. Ontologies are main building blocks and fundamental elements of Semantic Web and try to define a specific domain in a systematic way. Ontologies can be represented using different standards and languages such as RDFS[2] (Resource Description Framework Schema) and OWL[3] (Web Ontology Language). Both RDFS and OWL are based on RDF[4] (Resource Description Framework) which is a language for representing information about resources. The OWL language comes in three main flavours: OWL Lite, OWL DL and OWL Full which have been sorted according to expressivity and complexity levels. However, these languages may be too complex for putting light-weight semantics into webpages. Recently, two approaches were developed to simplify embedding semantics into websites. These techniques which were successfully adapted by major industry players include Microformats and RDFa.

`Microformats` – Microformats[5] enables Web developers to convey metadata into webpages using existing HTML/XHTML tags. It is a light-weight version of embedding semantics into websites.

`RDFa` – RDFa[6] (Resource Description Framework – in – attributes) is a W3C Recommendation for adding extensions to XHTML via attributes in order to embed rich metadata into webpages. Unlike Microformats, RDFa allows developers to introduce new metadata attributes.

If we consider Web 2.0 as read-write Web, Wolfram [Kobie, 2010] believes that in the next generation of the Web, so-called Web 3.0, machines will generate new content. [Agarwal, 2009] mentions that Semantic Web and personalisation are key aspects of Web 3.0. More information on Semantic Web including ontologies, Microformats and RDFa can be found at [Segaran et al., 2009]. We have used a combination of Semantic Web technologies and social networks, so-called semantic social networks, in our work. In the following section, we have a brief overview on this field of study.

2.2.1 Semantic Social Network

Combining Semantic Web and social networks has attracted many researchers over the past few years as it enables data portability among various social networks. Data portability enables people to use their data such as social connections and contacts across various platforms and applications. [Neumann et al., 2005] and [Downes, 2005] compared different online so-

[2] `http://www.w3.org/TR/rdf-schema/`
[3] `http://www.w3.org/TR/owl-ref/`
[4] `http://www.w3.org/TR/rdf-primer/`
[5] `http://microformats.org/`
[6] `http://www.w3.org/TR/rdfa-syntax/`

cial networks based on different criteria and discussed the importance of combining social networks and Semantic Web portals for a better collaboration in online communities. Several conceptual models were developed to demonstrate how social networks and the Semantic Web technologies can be incorporated. To this end, [Jung and Euzenat, 2007] proposed a three-layer architecture (social layer, ontology layer, and concept layer) for semantic social networks where these three layers are connected together and can influence each other. As an another example, [Mika, 2007] extended the model of ontologies with social dimension and showed how community-based semantics can appear from this new model through a process of graph transformation.

One of the main and earliest initiatives towards building a semantic social network is the FOAF (Friend of a Friend) project[7]. In brief, the FOAF ontology provides a set of terms that are used to describe people, their interests, their friends etc. Most researchers in this domain [Finin et al., 2005, Ding et al., 2005, Mika, 2004] use the FOAF ontology or an extended version of it for further analysis and research. The reason is perhaps due to availability of large FOAF-based corpora which provide sufficient data for further evaluation [Hogan and Harth, 2007]. We also used the FOAF ontology as an upper ontology to develop a tagging helper vocabulary – see Appendix I. Besides FOAF profiles which are commonly stand-alone RDF files, efforts like XFN[8] (XHTML Friends Network) were developed to embed social networks and human relationships using hyperlinks like HTML (i.e., microformats approach).

2.3 Tagging: A More Detailed Overview

Tagging has been popularised by Web 2.0 technologies and platforms and is also used in desktop platforms such as NEPOMUK social semantic desktop[9]. As illustrated in Figure 2.2, tagging falls into intersection of information architecture, social software and personal information management [Smith, 2008].

The collection of tag assignments in a community is commonly called a folksonomy. Thomas Vander Wal who coined the term folksonomy defines it as "*the result of personal free tagging of information and objects (anything with a URL) for one's own retrieval. The tagging is done in a social environment (shared and open to others). The act of tagging is done by the person consuming the information.*" To formalise this definition, we can define a folksonomy as a tuple $F := (U, T, R, Y)$, where U, T and R are finite sets of users, tags and resources, whereas Y is a ternary relation between them [Hotho et al., 2006b].

Folksonomies can be divided into two main groups: broad folksonomies and narrow ones [Peters, 2009]. A Broad folksonomy – like what *BibSonomy*[10] provides – describes the aggregated tags of several users applied to a single item, such as a research paper or a URL. Certain tags will tend to be assigned by users more frequently than others and the tag frequency distribution is generally characterised by a power law. The power law distribution is one of the main characteristics of broad folksonomies and shows that in a folksonomy a few tags will be assigned most of the times (i.e., power tags), while most tags will emerge a few times. Figure

[7] http://www.foaf-project.org/
[8] http://gmpg.org/xfn/
[9] http://nepomuk.semanticdesktop.org
[10] http://www.bibsonomy.org/

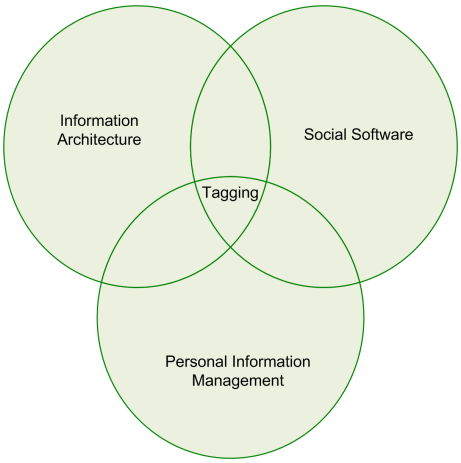

Figure 2.2: Three perspectives on tagging – source: [Smith, 2008].

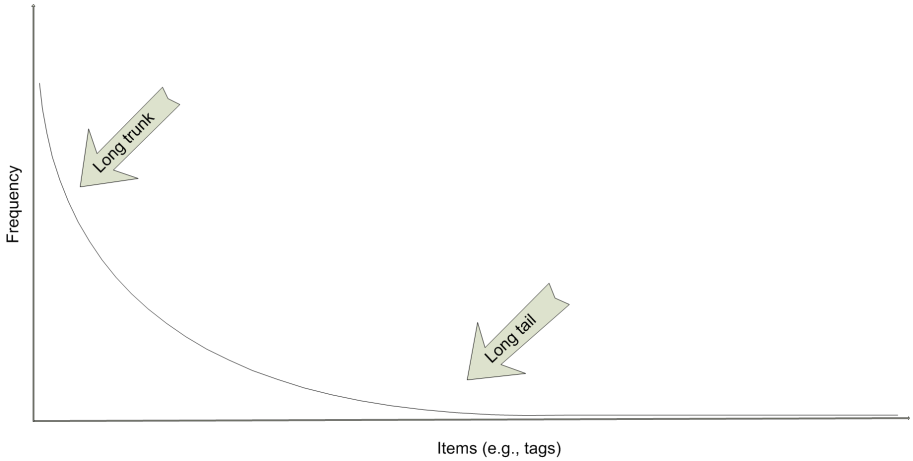

Figure 2.3: A typical power law distribution graph.

2.3 demonstrates a typical illustration of a power law distribution graph in which power tags form the so-called long trunk, whereas the rest of the tags form the so-called long tail. One of the main features of a power law distribution is that tags with higher frequency can be used in tag recommendation systems, as such systems usually recommend tags that are often used by other users and thus have higher frequencies (i.e., power tags). On the other hand, in a narrow folksonomy, a single user typically tags content for personal content management purposes. An example would be the tags assigned by *flickr.com* or *youtube.com* users to their content. [Hotho et al., 2006c] refer to narrow folksonomy as a personomy and describe a folksonomy as an aggregate view of several personomies.

Visualisation

Using tag clouds is a common practice to represent a collection of tags and distinguish between occurrence frequencies of different tags in the collection. The difference between occurrence frequencies of the tags is represented by a distinguishable characteristic such as using a different colour or font-size. Figure 2.4 demonstrates a sample tag cloud which was generated

Figure 2.4: A sample tag cloud (generated by Wordle).

by Wordle[11]. As Figure 2.4 illustrates, occurrence frequency of the tag *Twitter* is more than the tag *Tadvise* in the tag collection and both *Twitter* and *Tadvise* tags occurred more than other tags. Proportional scaling as well as linear scaling are two different approaches that can be used for creating tag clouds. In the linear scaling approach, a logarithmic function is applied to tag counts to flatten out the power-law curve [Smith, 2008] and the generated cloud.

2.4 People-Tagging

People-tagging extends the tagging practice to humans, in which users of a system are able to tag each other by using various terms and thus create people-tag folksonomies[12]. We can define people-tagging as *the process of assigning non-hierarchical metadata to any user of system by a user of that system.* Note that according to this general definition, *self-tagging* is also included. Metadata which is assigned to a user may describe a wide range of user attributes and characteristics such as expertise, interests, social behaviour and so on. Grouping users (e.g., based on their roles) can be also perceived as a way of adding coarse-grained metadata (i.e., group name) to them. Unlike grouping users which commonly involves assigning two or more users to a group, tagging users is more fine-grained and may describe detailed characteristics of said users. Figure 2.5 demonstrates several sample people-tags that were assigned to a user in an organisation. As this figure illustrates, four users tag the user in the middle with various terms such as expertise (java, linked data), social event (FAWM), project (lion2), associated group (URQ), role (student), and so on.

The folksonomy of people-tagging can be defined with a tuple $F' := (U', T', Y')$, where U' and T' are finite sets of users and tags, whereas Y' is a relation between them. The only difference between this definition and the general definition of folksonomies presented in Section 2.3 is that in people-tagging, the resources that are tagged by users are users themselves. Like tagging non-human resources, we can also envision two types of people-tag folksonomies: narrow and broad. In a narrow folksonomy, a user assigns tags to his/her contacts and such tags can be used for personal contact management, whereas in broad people-tagging folksonomies, people-tags assigned to a user will be aggregated. Such aggregated tags can be used in use cases such as expert finding.

[11] http://www.wordle.net/

[12] People-tagging should not be confused with hardware-based tags that are implanted or attached to human bodies mainly for identification purposes.

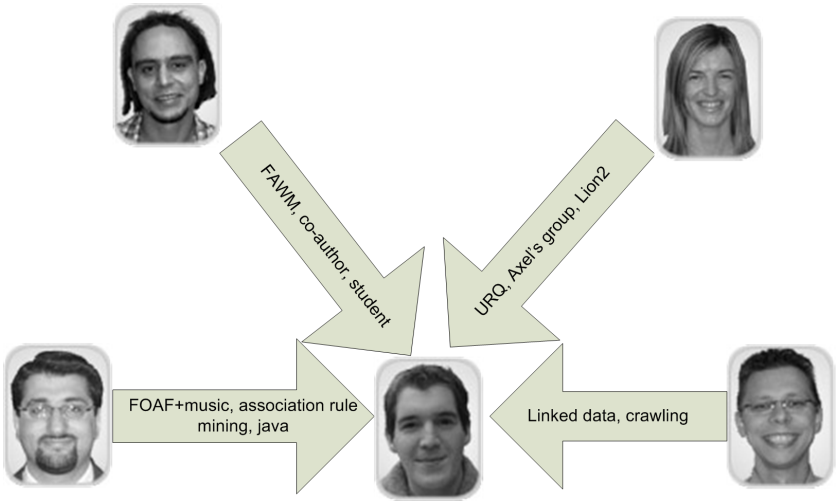

Figure 2.5: Several sample people-tags assigned to a user in an organisation.

From one perspective, people-tagging enables building fine-grained user profiles by making this practice collaborative, in which users attach metadata to a single user. Such a collaborative approach overcomes main limitations of single-user-based approach for updating user profiles in medium-to-large organisations. User profiles in medium-to-large organisations are used in use cases such as building dynamic teams including people with desired expertise [Serdyukov et al., 2011]. Such use cases require profile analysis tools to process personnel expertise and select people with desired expertise. In order to provide input data to these tools, most medium-to-large organisations keep content management systems (CMS) to unify format of the user profiles and thus ease processing of them. Using CMS enables personnel to update their activity streams such as related projects, team partners, publications, and generally their curriculum vitae (CV). Yet, an analysis of user profiles within a big corporation showed that only 40% of professional profiles were updated in a nine months period [Farrell et al., 2007]. Keeping the user profiles up-to-date with all the information may bring additional overhead for users. Moreover, this may be neglected by busy personnel. Thus, it is feasible that some users may update their profiles infrequently. [Peters, 2009] mentions that in people-tagging practice since tags are attached to a profile by one's colleagues, the individual user is spared the annoying task of constantly updating his/her profiles. Moreover, the social part of this practice may incentivise the taggees – users who are tagged by others – to reciprocate and tag the taggers [Bernstein et al., 2009].

In addition to offering less overhead for updating user profiles, in the following paragraphs, we discuss other advantages of using people-tags within organisations.

- Collaborative ranking: When building a skills profile, a user may be required to list and rank his/her capabilities using measures such as *intermediate* or *expert*. Such subjective opinions and endorsements may lead to ranking drawbacks of the profiles. A complimentary approach would involve seeking annotations or endorsements from others based on their experiences with the candidate and knowledge about candidate's skills and expertise. People-tagging as a collaborative approach for building user profiles can be used for ranking user expertise. For example, the more a user is tagged by others with an expertise item such as `java`, we may imply that user activities and perhaps

expertise in `java` is more than others. People-tagging can be used by people who have different hierarchy-like organisational roles (e.g., secretary, boss) and such roles enable application developers to define rules for automate ranking. For example, a people-tagging system may give a higher rank to people-tags that are assigned by a `unit leader` to `unit members` within an organisation.

- Fine-grained topics: In order to facilitate the process of building and maintaining user profiles, most CMS provide simple vocabularies as a list of topics and enable users to choose the topics that they are expert in or familiar with. Giving such hints to users speeds up building personal profiles. While this practice is useful for general topics such as list of ongoing projects within an organisation or list of programming languages, it is not feasible to list all possible fine-grained expertise items. For example, a CMS may list *Java*, *C++*, and *Python* as expertise elements, but fine-grained topics such as *association rule mining* or *query optimisation* are too specific to be listed and suggested to users. As people-tagging involves adding non-hierarchical metadata to users, this practice may generate more fine-grained user profiles.

- Improving collaboration: When organisations grow, employees may not be aware of each others' fine-grained expertise items and this may create obstacles for building dynamic teams with people who have desired expertise. [Serdyukov et al., 2011] mention that members of large organisations frequently look for other colleagues with desired expertise rather than documents. As people-tagging aims to create fine-grained user profiles, this practice may improve collaboration within medium-to-large organisations. Moreover, as such profiles contain fine-gained items of someone's activity, by retrieving someone's ongoing activities, a co-worker may find new collaboration opportunities (see Figure 2.6 for a sample scenario).

- Improving communication: Everyday a huge amount of data is produced on the Web. Various studies [Davenport, 2005, Morris, 2005, Slagter et al., 2007] showed that knowledge workers are quite overloaded by the huge amount of information they receive (i.e., information overload). The information overload issue may hinder relevant information that knowledge workers should or would like to receive which may lead to so-called information shortage drawback, i.e., lack of receiving appropriate information. One of the main issues raised during the Ecospace project[13] user requirement analysis was the lack of support to reduce information overload, for instance by providing clues about the relevancy of material to a specific person. "Knowledge workers feel insufficiently supported by current tools in their efforts to reduce information overload and to provide the right context information with shared material (For instance, "Why is this material relevant for me?")" [Slagter et al., 2007]. The challenge is to filter the irrelevant information and deliver relevant information to each user. People-tagging can be used to develop models for improving communication by reducing information overload as well as information shortage. Through core chapters of this thesis, we demonstrate our approach based on people-tags for a more user-centric information propagation within social and collaborative platforms.

In brief, people-tagging can be seen as a way to harness collective intelligence or wisdom of the crowd – the process of taking into account the collective opinion of a group of individuals

[13]The author was involved in this project for more than three years.

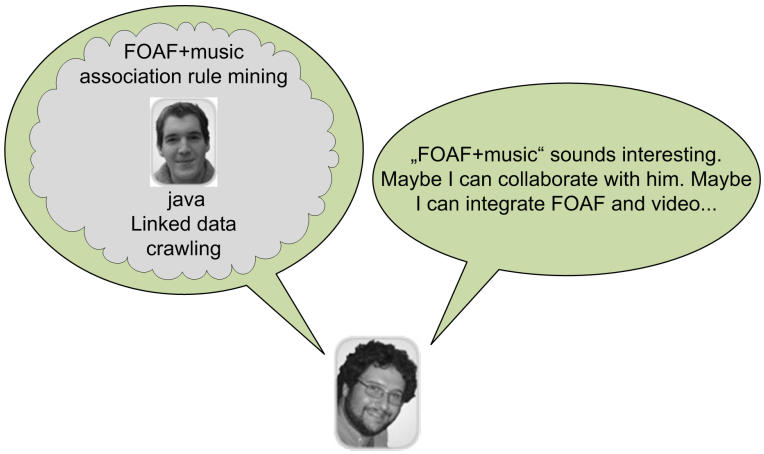

Figure 2.6: Fine-grained user profiles may improve collaboration between knowledge workers.

rather than opinion of a single expert [Surowiecki, 2004]. The people-tagging technique is currently used in platforms like Fringe Contacts [Farrell and Lau, 2006, Farrell et al., 2007] or Tagalag[14]. People-tags within organisations are normally kept private due to security reasons, whereas people-tags in public social platforms may be private or public. For example, the social networking site called XING[15] enables users to tag each other and such people-tags are not disclosed to public, whereas other platforms like Tagalag or blog.co.uk[16] enable public people-tagging.

People-tagging as a way to create social networks and make social links between people has been adopted by an award-winning website called 43 Things[17]. Unlike common social networking sites such as *facebook.com* in which users have to explicitly add contacts or invite them to the system, 43 Things enables users to tag themselves (i.e., self-tagging) and enter goals that they would like to achieve in a period of time. The site uses natural language processing techniques to match user goals and link those users who have similar goals or interests. Figure 2.7 demonstrates a sample tag cloud of users' new year resolutions from 43 Things website. Interestingly, researchers also showed that for a people-tagging system to be useful and bring maximum value for users within organisations, self-tagging is as important as tagging others [Raban et al., 2011].

2.4.1 Limitations and Pitfalls of People-tagging

People-tagging as a complementary approach for building user profiles has its own limitations too. In the following paragraphs, we list several main limitations of this practice and various approaches to overcome such limitations. Note that some of these limitations and pitfalls could be a general problem of many tagging systems and they are not specifically people-tagging issues.

[14] http://tagalag.com/
[15] http://www.xing.com
[16] http://www.blog.co.uk/
[17] http://www.43things.com/

Figure 2.7: A tag cloud from 43 Things: a website that uses self-tagging for connecting people.

- Privacy: Privacy is one of the main concerns of both taggers and taggees. Informal interviews conducted by the author with IT knowledge workers revealed that some of them believe that tagging others could be impolite, unethical or against their privacy. Obviously, tagging others with their physical or behavioural characteristics could be unethical, however, the main goal of people-tagging inside organisations is focused on intra- and inter-organisational use cases, such as expert finding [Serdyukov et al., 2011]. Thus, training users to tag each other with professional domain concepts such as expertise, projects, skills, etc. may address ethical concerns of people-tagging. Nevertheless, [Raban et al., 2011] mention that due to existence of user profiles that map to real knowledge workers and trust among co-workers, misuse of people-tags inside organisations will be limited. Enabling taggers to see who tags them (i.e., non-blind tagging) and with what topics is an approach that can be considered for improving privacy. Moreover, end users may have the permission to monitor the tags and perhaps remove low-ranked and/or inappropriate tags.

- Life span or durability of people-tags: The tags associated with a person may become obsolete as the person changes his/her project, job, role or learns new tasks. In a study that was conducted within a large corporation, 86% of participants indicated that they were conformable with the tags they had received, whereas 14% wanted to remove some tags [Farrell et al., 2007] mainly because some tags were no longer relevant. Understanding how to handle expiration date of people-tags requires further research.

- User motivation: In some cases, users may find adding metadata to their contacts confusing and hard. This may lead to cold-start problem of people-tag-based systems, i.e., lack of sufficient data to be used in the system. Automating the process of extracting, ranking and assigning people-tags to users can address this limitation. An innovative, yet affective approach for encouraging users to tag each other is to develop social games. To this end, [Bernstein et al., 2009] developed Collabio, a *facebook.com* game that encourages users to tag each other. Figure 2.8 demonstrates a snapshot of this game.

- Remembering tags: Unlike assigning a category from a pre-defined taxonomy to a resource, tagging is more an ad-hoc behaviour. Thus, a user may require advice on assigning appropriate tags to a resource [Sigurbjörnsson and van Zwol, 2008], in order to minimise common issues that are associated with tagging practices such as tag synonyms or tag disambiguation. Moreover, the cognitive capacity of human beings is

Figure 2.8: A snapshot of Collabio: a game based on people-tags – source: [Bernstein et al., 2009].

limited. Thus, users may forget the tags that they have assigned to others and generally to other objects such as photos and bookmarks. Developing personalised assistants that can help users to remember tags, propose appropriate tags and generally manage people-tag-based profiles are approaches that can help users to overcome this limitation.

- Messy tags: Messiness of tags (e.g., inappropriate, noisy, too subjective tags) is a potential problem of tagging systems due to non-hierarchical structure of assigned tags as such tags derive from a wide range of domains. Tag suggestions, training users to effectively use people-tags, and finding relationships (e.g., synonyms) between the tags are approaches that can be considered to avoid the mess in tagging folksonomies [Smith, 2008]. Defining a set of super-tags (i.e., metadata for metadata) can be also considered to organise tagging folksonomies. In other words, users may define type of a tag that they assign to someone by explicitly mentioning its category (e.g., affiliation or expertise – in this case, `affiliation` and `expertise` are so-called super-tags [Hotho et al., 2006a]).

- Vocal minority: There is a so-called vocal minority problem which may emerge in tagging systems [Smith, 2008]. That is when a small set of users dominate the tagging activity in the system by assigning too many tags to others. Obviously, having active taggers bring much value to the system, however, overusing the system by a small set of users may bias the people-tagging folksonomies. Developing algorithms to detect such users and/or eliminate their activities by giving a lower rank to tags that they assign can be considered to overcome such pitfalls.

In core chapters of this thesis, we develop approaches to address some of these limitations. The important contributions are as follows: a) we will provide insight on people-tagging

behaviour outside organisations; b) we provide an approach to extract, rank and assign people-tags to knowledge workers; and c) we use people-tag-based profiles for an access control and information filtering use case.

Our work is a small step towards realising the vision of many researchers [Bush, 1945, Engelbart, 1962] about the future of computation and personal knowledge management [Davies, 2011]. As an instance, our work is aligned with what Bush envisioned in his visionary system called *memex* [Bush, 1945], in which people are able to store and link their knowledge in an environment with hypertext capabilities. People-tags are like flags around which people gather. They can be seen as entities networking people and information along with linking people and knowledge thereby creating networked knowledge.

2.5 Conclusion

In this chapter, we presented a brief overview of various concepts that we use through this thesis. We briefly presented a short list of technologies and services of Web 2.0 or social Web. Such technologies and services enable users to collaborate together more easily and rapidly. We had a brief overview of Semantic Web and how it is used to improve social networking capabilities and tagging practices.

We had a deeper look at online tagging, a practice that was popularised through Web 2.0 services. Collaborative tagging helped to the birth of a new classification scheme called folksonomy. We introduced the initiative of people-tagging as a novel way for building user-profiles, so that expertise and interests of a user can be introduced and/or evaluated by others. The fine-grained user profiles have many capabilities such as in information filtering or building dynamic teams with desired experts. Through core chapters of this thesis, we demonstrate our approach for extracting, ranking and assigning people-tags to users, in order to reduce the overhead of creating fine-grained user profiles. We then use people-tag-based profiles for a more user-centric information propagation model within social platforms.

Part II

Core

Chapter 3

Understanding People-Tags[1]

> The purpose of computing is insight, not numbers.
>
> Richard Hamming

Tagging has been widely used and studied in various domains (as discussed in Chapter 2). Recently, people-tagging has emerged in large organisations such as IBM and also in several social websites such as 43people.com as a means to categorise contacts or discover new ones. In this chapter, we investigate whether there are differences between people-tagging and bookmark-tagging. Such understanding sheds some light on the main characteristics of people-tags, as core concepts of this thesis are based on using people-tags for finding people with relevant expertise as well as information filtering mechanisms. In this chapter, we show that the way we tag online documents about people who we do not know personally, is similar to the way we tag online documents (i.e., bookmarks) about other categories (i.e., city, country, event). However, we show that the tags assigned to a document related to a friend, differ from the tags assigned to someone we do not know personally. This is an important finding, as our approaches are mainly focused on boosting collaboration and communication among people who work together or belong to the same community and thus to some extent know each other. We also analyse whether the age and gender of a taggee – a user who is tagged by others – have influences on people-tags assigned in social Web 2.0 platforms. Overall, our analysis suggests that people-tags may be assigned in different contexts and originated from a wide range of concepts such as from hobbies to daily activities. This creates interoperability issues between applications that use people-tags as a means for building user profiles. Therefore, in order to make people-tag-based applications interoperable, we need end users' consensus on the tags. Ranking people-tags is an approach that can be used to eliminate subjectivity of them.

[1] This chapter is mainly based on [Nasirifard et al., 2010a].

3.1 Introduction and Motivation

Since the widespread adoption of Web 2.0 applications, tagging has been widely used and studied in various platforms, mainly for annotating online resources (e.g., photos, videos, bookmarks). Recently, people-tagging has emerged as a means to organise contacts, build user profiles and manage competencies, especially in large scale organisations [Farrell and Lau, 2006, Farrell et al., 2007, Schmidt and Braun, 2008]. As stated in the previous chapters, people-tagging is simply the (online) tagging of human beings for organisation and retrieval purposes and is a mechanism that is currently used in enterprise platforms such as IBM's Fringe Contacts [Farrell et al., 2007] and websites such as www.blog.ca, www.tagalag.com, and www.43people.com.

Recommendation techniques for tagging online resources have been widely studied and are used in various platforms like delicious.com[2]. Work by [Rattenbury et al., 2007] describes an approach for automatically identifying tags in *flickr.com* which relate to locations and events. There exist also approaches like Tess [Oliveira et al., 2008] or TagAssist [Sood and Hammond, 2007] that recommend tags based on content of a document or a blog post respectively. We are not aware of any study on recommendation techniques for tagging human beings. Our hypothesis is that unlike online resources such as bookmarks, online pages related to people we know are tagged with more subjective tags. The intention of our study on people-tagging behaviour is to give us a starting point to build recommender systems and/or vocabularies that can be used for recommending appropriate and relevant people-tags. In the context of this work, such systems and vocabularies can be used to improve usability of access control mechanisms (e.g., information filtering mechanisms) that are based on annotating people [Razavi and Iverson, 2009, Nasirifard and Peristeras, 2008b, Wang and Jin, 2009]. This will be further discussed and developed in Chapter 5.

In this study, we plan to address two main research questions.

- *Q1*: Do the properties (such as linguistic categories or subjectivity) of tags of articles belonging to various categories (i.e., person, event, country and city) differ? In the first part of our analysis, we compare the tags associated to Wikipedia articles related to persons with the tags that have been assigned to articles of the other categories (i.e., city, country, and event).

- *Q2*: Do the properties (such as linguistic categories or subjectivity) of tags assigned to Wikipedia pages describing persons differ from the tags that are assigned to pages for persons (i.e., friends) in online social network platforms?

Moreover, we also take a look at the roles of gender and age of taggees within social platforms. Figure 3.1 demonstrates an overall view on how we approach to address these questions.

For the remainder of the chapter, we will use the terms Person, Event, City and Country to refer to the subject of a Wikipedia article related to a person, event, city and country respectively; and Friend to refer to a digital representation of a contact on a social network site – for example, this could be their webpages or FOAF profiles.

The remainder of the chapter proceeds as follows: We present related work in Section 3.2. In Section 3.3, we describe our method for extracting people-tags and the statistics of the data

[2] http://www.delicious.com/

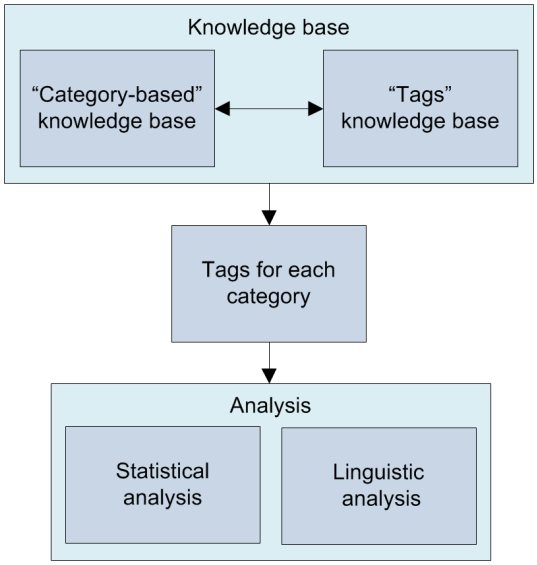

Figure 3.1: An overall view of our analysis approach.

we collected. Next, in Section 3.4, we describe our methodology for analysing the tags and we address the research questions we posed. We present results related to the age and gender of taggees in Section 3.5. Finally, we close the chapter by explaining what we have learned from this study and have an overview of future work in Section 3.6.

3.2 Related Work

Tagging in social media has attracted many researchers over the past few years. Researchers have studied various aspects of tag usage including the behaviour of users of different types of tagging systems [Marlow et al., 2006], motivations behind tagging [Santos-neto et al., 2009], tag distribution, tag dynamics and tag-tag correlations [Halpin et al., 2007], and the changes in user activity in tagging systems over time [Golder and Huberman, 2006], however, none of above studies were performed on online objects representing a person.

Previous research on document classification has shown that people use attributes that are subjective in the organisation of personal documents [Bergman et al., 2003]. This has given us a starting point to investigate whether this hypothesis holds for tagging personal contacts as well.

Previous work comparing tagging for different resource types includes also [Muller, 2007], which studies bookmarks in systems for documents, for people, for blog entries, and for activity records, in an online corporate environment. They found that users' tagging behaviour tended to differ between the systems and as a result there exists a little overlap among tags used in aforementioned systems. Similarly, [Muller et al., 2007] present the results of an experiment, where a service was provided for people to apply tags to one another within an online corporate environment. They classified tag usage and found that users have a preference to apply tags related to expertise to themselves, and to apply tags related to roles to others. Their findings were valuable, however, the data they used for their study is not public and is bounded to one single organisation. Moreover, they did not study people-tagging

behaviour in public social platforms.

To validate and position our classification scheme (i.e., subjective/objective) on tags assigned to people, we further studied relevant works. There exists previous work on classifying tags, both manually and automatically. [Overell et al., 2009] describe a method to automatically classify *flickr.com* tags using a vocabulary constructed from Wikipedia and WordNet. [Bischoff et al., 2008] compare tag characteristics between different types of resources: webpages (*delicious.com*), music (*last.fm*) and images (*flickr.com*). The classification is performed manually. They have shown that the distribution of tag types (e.g., music category) strongly depends on the resources that users annotate. [Sen et al., 2006] classify tags from a movie recommendation system as Factual, Subjective or Personal and study how these classes of tags are used, and how useful these tags are for user tasks. [Xu et al., 2006] present a taxonomy of tags, and use this taxonomy as a means of ensuring diversity in their tag suggestion system. After studying above approaches, we decided to adopt [Sen et al., 2006]'s approach for classifying our tags, as part of our hypothesis is to evaluate subjectivity level of social people-tags.

In brief, our work extends the previous work by comparing the tag classifications derived from [Sen et al., 2006] for different categories of Wikipedia articles (i.e., persons, places, and events). We also compare the tag classifications derived from [Sen et al., 2006] for tags assigned to Wikipedia articles about famous people with those assigned by people to their friends within public social platforms. The objective of these experiments was to study if there exist differences between properties (such as subjectivity) of these tags.

In order to address the research questions that we posed in Section 3.1, we followed an experimental approach. Initially, we collected tags that were assigned to several different categories of Wikipedia articles, including those articles related to persons. Then, we crawled tags from several websites that enable users to tag each other. We used human judgements to evaluate the extracted tags against our classification scheme. Based on feedbacks that we received from our subjects, we applied statistical methods to validate our hypotheses. In the following section, we have a more detailed overview of each step.

3.3 Data Collection

In order to get datasets for our analysis, we identified DBpedia[3] as an instance of "category-based" knowledge base and delicious.com as an instance of "tags" knowledge base, as illustrated in Figure 3.2.

Our first goal was to extract tags that were assigned to a particular type of category namely Person, City, Country, and Event. We decided to narrow down our study to aforementioned categories, as this could give us sufficient data for comparing tags assigned to Persons with tags assigned to non-Persons. As stated, we used DBpedia which is a community effort for extracting structured data from Wikipedia, as an instance of a "category-based" knowledge base. DBpedia transforms Wikipedia pages into structured categorical data. Several end points exist which allow end users to query DBpedia using the SPARQL[4] query language. We used version 3.2 of DBpedia for our analysis. We extracted four different types of categories: person, city, country and event. For extracting the person data (i.e., links to Wikipedia pages

[3] http://dbpedia.org
[4] http://www.w3.org/TR/rdf-sparql-query/

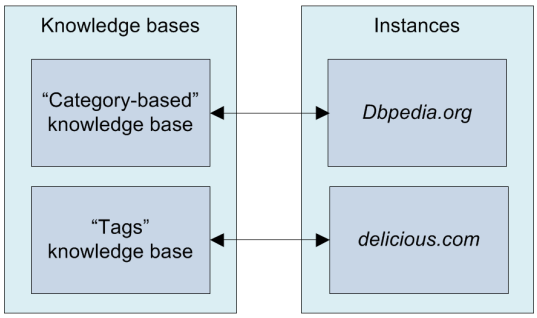

Figure 3.2: Instances of knowledge bases.

Table 3.1: Properties of Wikipedia article collection.

Type	Items	Tagged	Untagged	Total Tags	Unique Tags
Person	20284	4031 (19.9%)	16253 (80.1%)	75548	14346 (19%)
Event	7601	1427 (18.8%)	6174 (81.2%)	8924	2582 (29%)
Country	1734	638 (36.8%)	1096 (63.2%)	13002	3200 (25%)
City	40197	1137 (2.8%)	39060 (97.2%)	4703	1907(40%)

related to persons), we used the complete set of instances in the DBpedia person data dump[5]. As there was no DBpedia data dump for other categories (i.e., city, country and event), we crafted SPARQL queries to extract links to Wikipedia pages. After gathering the Wikipedia links from DBpedia, we crawled delicious.com as an instance of a "tags" knowledge base to retrieve the tags associated to those Wikipedia articles[6]. Figure 3.3 shows a simplified view of our approach to extracting tags associated to a specific category. All tags were rendered into lower-case. We retrieved only the tags associated to English Wikipedia pages. Table 3.1 shows properties of the data retrieved from four different categories of Wikipedia articles.

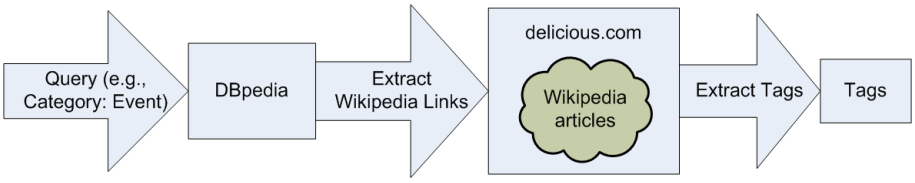

Figure 3.3: Overall approach for extracting category-based tags.

For our second goal, extracting social people-tags, we used four distinct but related social websites[7]. The main purpose of these sites is to blog, but they allow also users to maintain social networks and tag each other. These websites were the only public sites with built-in people-tagging feature that we were aware of and enabled us to retrieve people-tags by developing a simple HTML scraping tool. Two websites are in English, one in German and one in French. Besides using English websites, we also used German as well as French sites for further processing. In order to unify the gathered (crawled) people-tags, we used

[5] http://wiki.dbpedia.org/Downloads32#persondata

[6] We crawled delicious.com in June 2009.

[7] http://www.blog.de, http://www.blog.ca, http://www.blog.co.uk/, and http://www.blog.fr/

Table 3.2: Properties of people-tag collection gathered from social blogs.

Site	Users	Tagged	Untagged	Total Tags	Unique Tags
blog.co.uk	1474	553 (37.5%)	921 (62.5%)	3509	2665 (75.9%)
blog.ca	429	100 (23.3%)	329 (76.7%)	569	492 (86.5%)
blog.de	5836	2035 (34.8%)	3801 (65.2%)	11966	7626 (63.7%)
blog.fr	962	239 (24.8%)	723 (75.2%)	1082	803 (74.2%)
Aggregation	8701	2927 (33.6%)	5774 (66.4%)	17126	10913 (63.7%)

the Google translator API[8] to translate non-English tags, as [Callison-Burch, 2009] showed that French-to-English and German-to-English Google translators are among state-of-the-art automatic machine translation (MT) systems. As the context information (e.g., situations or events in which people-tags were assigned) associated with people-tags was not present in those websites, using the Google API was a good alternative to using a native human translator for top tags which were mostly one-term tags. To validate this hypothesis, we used two German and French natives to evaluate Google translation for top-100 tags. Our two evaluators used Fluency and Adequacy scores to judge the automatic MT results. Using Fluency (i.e., how well the meaning is captured) and Adequacy (i.e., quality with regards to grammar and comprehensibility) [White et al., 1994] is a common practice of using human judgement for validating automatic machine translation. We defined the range of Fluency and Adequacy scores between 0 (i.e., no connection) and 5 (i.e., perfect). On average, for top-100 tags Google German-to-English translator scored 4.81 in Adequacy and 4.92 in Fluency, while French-to-English translator scored 4.71 in Adequacy and 4.87 in Fluency[9]. Note that in non-English websites, some people used English tags as well. Table 3.2 shows properties of the people-tags crawled from those websites. The number of users in Table 3.2 indicates the total number of registered users on the sites[10].

3.4 Experiments

In this section, we address the research questions we posed in Section 3.1, i.e., whether tags of Persons differ from tags of other topics and whether tags of Friends differ from tags of Persons? In order to answer these questions, we followed an approach composed of three steps. First, we compared frequency distribution of the tags. Second, we compared linguistic categories of the tags. And third, we wished to investigate subjectivity level of people-tags. However, the core part of this analysis investigated how subjective people are when assigning tags to each other (i.e., the third step was our core investigation). We took the following measurements and tools for the analysis in each step. For the first step, we checked whether tag distribution follows Zipf's law, meaning that whether the hypothesis of most tags occur a lot, while a small subset of tags occur less, holds for different types of tags. For the second step, we used WordNet [Miller, 1995] to compare the linguistic categories of the tags. For the

[8] https://developers.google.com/translate/

[9] Consulting with one NLP expert in relation to those scores showed that Google automatic translation preformed a pretty acceptable translation.

[10] We crawled the websites in June 2009.

third step (i.e., evaluating the subjectivity levels of people-tags), we used a manual method to categorise the top-100 tags for each topic. Our manual method comprised of asking subjects to categorise tags based on the following scheme derived from [Sen et al., 2006]:

- Objective: Objective tags are those tags that identify the facts about somebody or something. For example, locations, concepts, somebody's role and expertise are categorised as objective tags. Named Entities (NE) fall into this category (e.g., london).
- Subjective: Subjective tags are those tags that reflect the personal opinion and feedback about someone or something. For example, the opinions about physical and behavioural characteristics of somebody are categorised as subjective tags (e.g., jealous).
- Uncategorised: We asked our subjects to assign those tags that could not fit well into one of above categories, or their meaning/usage were ambiguous, to this category (e.g., abcxyz).

We gave aforementioned definitions of subjective, objective and uncategorised tags to 25 subjects. Each of the 25 subjects was assigned the top-100 tags for one category (i.e., Country, City, Event, Person, or Friend) and was asked to categorise them based on our scheme. The subjects were free to search for the meaning of the tags on the Web. We tried to keep the categories as clear and simple as possible, as we did not want to make the categorisation task difficult for our subjects. Our 25 subjects were mainly from computer science and IT backgrounds. In order to analyse subjects' results, we used Analysis of Variance (ANOVA) [Freedman et al., 2007]. The ANOVA statistical method is commonly used in fields such as sociology or human-computer interaction to analyse data obtained from human participants in controlled experiments. This method allows us to determine if differences between results are statistically significant[11].

3.4.1 Research Question 1 (Q1) – Do the properties of tags of articles belonging to various categories of Wikipedia articles differ?

Table 3.3 shows the top-20 tags assigned to Wikipedia articles related to various categories (i.e., Person, City, Country, and Event) plus their frequencies. Note that the collected tags were not related to a particular time period. What we observed from the properties of top tags of Wikipedia articles was the fact that after removing general tags (e.g., wikipedia, people) *Persons* on Wikipedia were mostly tagged with the concepts that they are famous for or expert in (e.g., music, politics, poetry). Most of the top tags associated to *Events* were related to their political-historical context (e.g., ww2, battle, war). Both *Countries* and *Cities* tended to be tagged with their geo-political context. In particular, the *Countries* were likely tagged with their continents, their historic background or the name of the country itself (e.g., europe, empire, japan), whereas the *Cities* were mostly tagged with the countries that they are located in (e.g., spain, germany, uk). There are also interesting common tags across categories. For example, the tag *history* suggests that all categories have historical context for users; or the tag *research* was perhaps used by users for scientific research purposes and/or research for travel or holidays. Both Person and Country are viewed as having an association with the *culture* tag. The distribution of the tags among four categories follows Zipf's law

[11]The author would like to acknowledge Krystian Samp for his guidelines in performing statistical analysis.

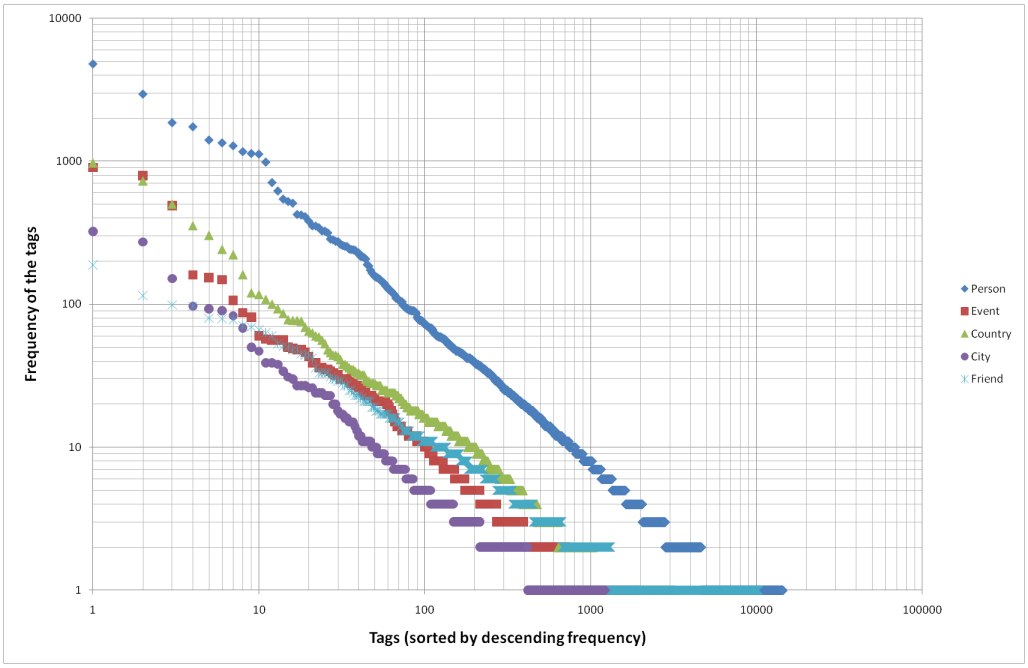

Figure 3.4: Distribution of the tags assigned to various types of Wikipedia articles and also Friends on blog-related websites based on a log-log scale. 64% of the tags assigned to Friends on blog-related websites were unique, whereas only 19% of the tags assigned to Persons on Wikipedia were unique.

[Reed, 2001]. That means most tags occurred rarely, whereas a small subset of tags have been used a lot (see Figure 3.4).

Figure 3.5 demonstrates the linguistic categories of the tags based on WordNet classifications[12]. As illustrated in Figure 3.5, most tags for all four resource types, that were categorised by WordNet were nouns; while verbs, adjectives and adverbs followed respectively. As illustrated in the figure, a large proportion of the tags could not be categorised by WordNet and Person and Friend tags are the most likely tags to be uncategorised.

After getting the categorisation results from our subjects, we ran a Repeated Measures Analysis of Variance (ANOVA) [Freedman et al., 2007] with between-subject factor. As stated, in our study each subject received top-100 tags for one Wikipedia resource type (i.e., Country, City, Event, or Person) and had to categorise them into three groups: objective, subjective or uncategorised. Consequently, the between-subject factor was *wikipedia-resource-type* (i.e., Country, City, Event, and Person) and the within-subject factor was the *tag-category* (i.e., objective, subjective, and uncategorised).

The dependent variable was the number of occurrences of a *tag-category* within top-100 tags for a *wikipedia-resource-type*. There was no significant main effect of *wikipedia-resource-type*, as the mean number of tag occurrences for each resource type was always the same (i.e., 100 tags). There was a significant main effect of *tag-category* ($F(1.48, 23.7) = 270.3$, $p < .001$)[13]

[12] We normalised the values.

[13] The ANOVA's assumption of sphericity for tag-category was violated as indicated by Mauchly's test. In such a situation, it is necessary to use one of the corrections for degrees of freedom. To this end, we used Greenhouse-Geisser correction ($\epsilon = .74$). Note that the other assumptions of ANOVA were not violated.

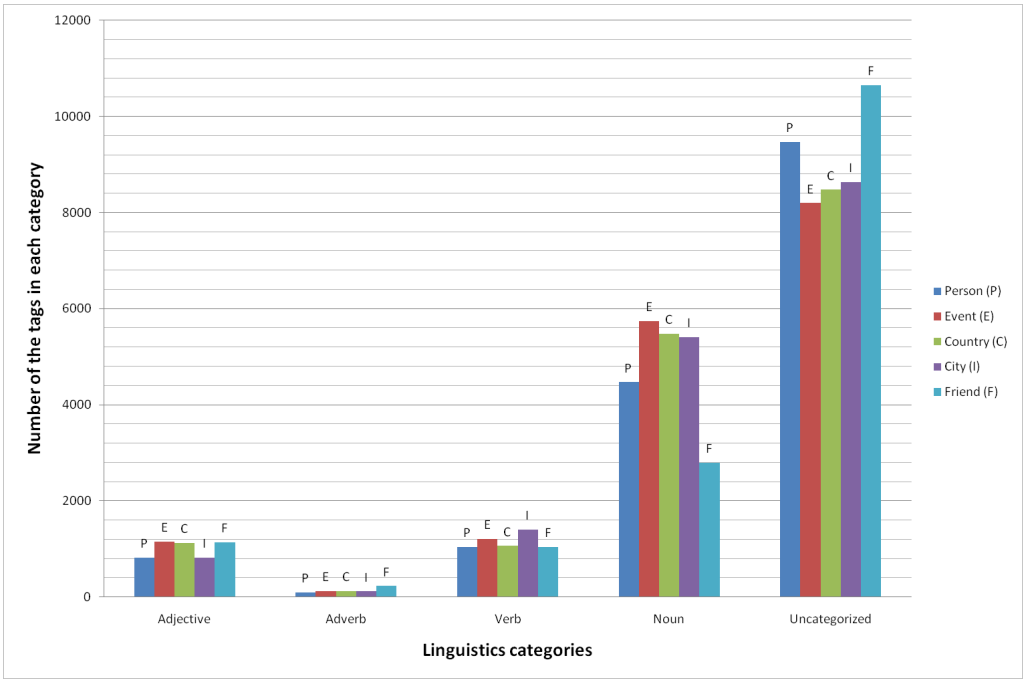

Figure 3.5: Normalised linguistic categories of the tags assigned to various types of Wikipedia articles and also Friends on blog-related websites.

meaning that the numbers of subjective, objective and uncategorised tags differed. Lack of significant interaction effect between *tag-category* and *wikipedia-resource-type* indicated, however, that the ratios of objective, subjective and uncategorised tags were similar across *wikipedia-resource-types*. Pairwise comparisons[14] showed that the number of subjective tags was smaller than objective ones and that the number of subjective and uncategorised tags did not differ from each other.

To answer Q1, these results suggest that people tag Wikipedia resources the same way regardless of the resource type using more objective than subjective tags. Figure 3.6 shows the distribution of subjective, objective and uncategorised tags for different resource types. Note that Figure 3.6 contains tag analysis related to Friend as well, as we refer to it in the following section.

3.4.2 Research Question 2 (Q2) – Do the properties of tags assigned to Wikipedia pages describing persons differ from the tags that are assigned to pages for persons (i.e., friends) in online social network platforms?

To answer Q2, we compared the people-tags assigned in social platforms with the tags that were assigned to Persons on Wikipedia. Table 3.4 shows the top-15 people-tags extracted from previously mentioned blog-related websites, their translation and also the frequencies. We observed that top tags assigned to people in social media were *attributes* and mostly related to physical characteristics and hobbies (e.g., music junkie, pretty, sweet, nice, honest). This

[14]For pairwise comparisons we used Bonferroni adjustment to preserve familywise significance level.

Table 3.3: Top-20 tags associated to various Wikipedia categories and their frequencies.

Person	F.	Event	F.	Country	F.	City	F.
wikipedia	4776	history	904	wikipedia	972	travel	322
people	2941	war	791	history	727	wikipedia	273
philosophy	1856	wikipedia	488	travel	500	italy	151
history	1737	ww2	160	geography	354	germany	97
wiki	1404	politics	153	africa	303	history	93
music	1341	wiki	148	culture	242	london	90
politics	1279	military	106	wiki	222	uk	83
art	1164	battle	87	reference	161	wiki	68
books	1130	wwii	81	europe	120	places	50
literature	1119	iraq	60	country	117	england	47
science	984	reference	57	countries	108	geography	39
biography	708	ajalugu	56	world	100	scotland	39
reference	618	wars	56	politics	93	europe	38
authors	543	olympics	56	research	86	brazil	34
author	522	civilwar	50	empire	78	slow_italy	31
research	508	usa	49	islands	77	city	30
film	424	wwi	48	information	77	information	27
toread	420	vietnam	48	india	76	japan	27
artist	409	russia	46	info	69	barcelona	27
psychology	380	iraqwar	43	japan	65	spain	26
poetry	353	china	39	island	63	berlin	26
religion	352	music	39	china	60	cities	24
culture	342	300	36	asia	59	photography	24
design	325	research	36	australia	56	reference	24
writing	324	ww1	35	germany	53	switzerland	23

was also proved by mapping the tags to WordNet. The mapping showed that among those people-tags that could be categorised by WordNet, the frequency of adjectives and adverbs in social media were higher than for Wikipedia articles related to persons (i.e., Person), whereas the frequency of nouns was lower (see Figure 3.5 for a normalised comparison). The distribution of Friend people-tags follows Zipf's law as well, but with a longer tail, meaning that 64% of the tags assigned to Friends on blog-related websites were unique, whereas only 19% of the tags assigned to Persons on Wikipedia were unique (see Figure 3.4).

After getting the results from our subjects, we ran the same type of ANOVA with the difference that the *resource-type* factor had 2 levels: Person and Friend. Again, there was no significant main effect of *resource-type*. There was a significant main effect for *tag-category* ($F(1.33, 13.3) = 69.6$, $p < .001$)[15] meaning that the numbers of subjective, objective and uncategorised tags differed. This time, there was a significant interaction effect ($F(2, 20) = 67$, $p < .001$) indicating that the ratios of objective, subjective and uncategorised tags differed for both *resource-types*. For pairwise comparisons we used dependant and independent t-tests.

[15]Again, as the data for *tag-category* was non-spherical, we used Greenhouse-Geisser correction ($\epsilon = .67$).

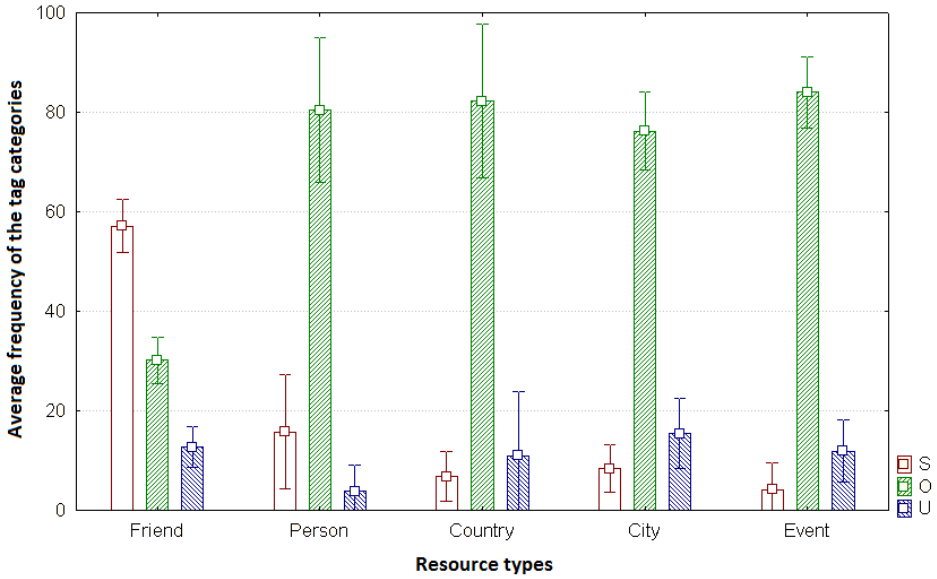

Figure 3.6: Average frequencies (+/- standard deviations) of subjective (S), objective (O) and uncategorised (U) tags as a function of resource type. For Person, Country, City and Event resource types tags are mostly objective (Q1), while for Friend tags are mostly subjective (Q2).

Table 3.4: Top-15 people-tags in different languages from blog-related websites, their translation, and their frequencies.

blog.de	F.	blog.fr	F.	blog.ca & .co.uk	F.
musikjunkie (music junkie)	188	art (art)	19	funny	34
nett (nice)	81	politique (politics)	14	music	32
leben (live)	77	musique (music)	14	life	31
lustig (funny)	73	gentil (kind)	9	kk friend	29
lieb (dear)	69	adorable (adorable)	8	funky	25
intelligent (intelligent)	66	amour (love)	7	friendly	23
huebsch (pretty)	61	sympa (sympa)	7	lovely	22
sexy (sexy)	59	dessin (drawing)	7	cool	22
liebe (love)	58	amitié (friendship)	7	sexy	19
ehrlich (honest)	56	digne de confiance (trustworthy)	7	love	18
interessant (interesting)	48	bon (good)	6	art	16
musik (music)	45	histoire (history)	6	poetry	16
kreativ (creative)	45	vie (life)	6	nice	14
humorvoll (humorous)	42	humour (humor)	6	photography	13
freundlich (kind)	41	sensible (sensitive)	5	barking	13

The number of subjective tags for Friend was higher than for Person (independent samples t-test, $t(10) = 8.5$, $p < .00001$). The number of objective tags for Friend was lower than for Person (independent samples t-test, $t(10) = -8.7$, $p < .00001$). Also, the number of uncategorised tags was higher for Friend than for Person (independent samples t-test, $t(10) = 3.29$, $p = .008$). For Person, the number of objective tags was higher than subjective ones (dependent samples t-test, $t(4) = -5.6$, $p = .005$); there were more objective tags than uncategorised ones (dependent samples t-test, $t(4) = 9.2$, $p < .001$), but the number of subjective and uncategorised tags did not differ from each other. For Friend, the number of subjective tags was higher than objective ones (dependent samples t-test, $t(6) = 7.9$, $p < .001$); there were more subjective tags than uncategorised ones (dependent samples t-test, $t(6) = 14.2$, $p < .0001$) and more objective than uncategorised (dependent samples t-test, $t(6) = 6.5$, $p < .001$).

To answer Q2, these results suggest that people use different tags for their friends compared to resources describing other persons. Friends are mostly assigned subjective tags while other persons objective ones. This subjectivity may create interoperability issues for approaches that are based on people-tags. Developing domain-specific vocabularies as well as ranking tags are approaches that can be considered to eliminate such subjectivity. In Appendix I, we present a domain vocabulary called CoVoc that we developed to help users to tag each other in IT-related research environments.

We envision that our results could be generalised, as both datasets that we used as instances of "category-based" and "tags" knowledge bases (i.e., Wikipedia and delicious.com) are widely adapted and are comprehensive enough to be considered as general pieces of information.

3.4.3 Random Tags

We also wished to investigate whether the properties of a random set of tags were similar to the properties of the most popular tags that we used in our experiments. Many of the tags in delicious.com and people-tags within social platforms are part of the long tail (i.e., tags with lower frequencies) and it is possible that these tags have different usage patterns to the top tags. Therefore, we also created one set of random-100 people-tags (i.e., associated with Friends) and one set of random-100 Wikipedia tags for each category (i.e., Person, Event, City and Country) and asked each of the 25 subjects in our experiments to categorise them based on our scheme. For the Wikipedia articles, the result showed that even for random tags on average more objective tags are assigned than subjective. But for Friend, the amount of subjective and objective tags were statistically equal, taking into account that more than 40% of the tags could not be categorised by our subjects. The subjects mentioned that some tags were strange, as they could not understand the meaning of them. Thus, they assigned them to the uncategorised group. We speculate that our subjects were not aware of the context that the people-tags were assigned. In other words, among friends, there are usually lots of events and issues that only those friends are aware of and can be used as tags (i.e., subjective), but from a subject's point of view, they could not be categorised.

Finally, we calculated the inter-annotator agreement of the random-100 sets and the top-100 sets. The average inter-annotator agreement for random-100 tags was 76%, whereas for top-100 tags it was 86%. The long-tail tags do not have as clear a meaning as the top tags, and therefore are probably less useful in applications such as information filtering.

Table 3.5: Top-15 tags for varying age ranges and gender of taggees.

≤ 25	> 25 and ≤ 50	> 50	Male	Female
music junkie	funny	kk friend	music junkie	music junkie
pretty	music junkie	funky	funny	love
nice	nice	funny	music	pretty
love	music	politics	nice	nice
sweet	live	love	politics	funny
funny	love	kunst	intelligent	sexy
music	intelligent	live	sexy	live
sexy	sexy	kind	love	dear
crazy	dear	music	live	sweet
dear	cool	friendly	kind	honest
honest	humorous	helpful	reliable	intelligent
intelligent	interesting	art	dear	crazy
creative	honest	humor	sports	interesting
interesting	kind	sensitive	sweet	music
thoughtful	creative	honest	art	cool

3.5 Age and Gender

The data on social blog sites contains age and gender of taggees as well. Although this is not the main focus of this chapter, the data gave us useful input for further analysis. Towards this direction, we conducted an additional experiment. We prepared three sets of top tags ($A1$,$A2$, and $A3$) assigned to various age ranges (age ≤ 25, $25 <$ age ≤ 50, and age > 50) respectively; and for the second part of the experiment, we prepared two sets of top tags ($G1$ and $G2$) assigned to the male and female genders respectively. Note that we removed tags that refer to a specific gender for this experiment (such as *girly*). Table 3.5 shows the top-15 tags of $A1$, $A2$, $A3$, $G1$, and $G2$. We asked 10 subjects to a) take a look at the first three sets and let us know, if they could predict the possible age range of taggees for each set; and b) take a look at the second two sets and let us know if they could justify whether one set is more suitable for a specific gender. We asked these questions in order to determine whether there were perceptible differences between the tag sets. All subjects agreed that $A1$ is used for younger people due to existence of some tags that are mostly used for teens and young people (e.g., naive, music junkie, sexy, freak, dreamy). These tags were categorised as subjective tags. Most subjects did not see any major differences between $A2$ and $A3$, and they mentioned that it was extremely difficult to distinguish between them, but they claimed that both sets are more likely used for older people due to existence of some tags like politics, art, and poetry (i.e., objective tags). For the second part of the experiment, there was a consensus indicating that no major differences exist between $G1$ and $G2$. Most subjects claimed that both sets suit both genders. The results suggest that age of a taggee can be considered as a design implication for recommending appropriate people-tags in social platforms.

3.6 Conclusion and Future Work

In this chapter, we presented the result of experiments related to people-tagging and bookmark-tagging. We showed that the pages related to persons on Wikipedia are tagged the same as other types (i.e., events, cities, and countries) – in terms of subjective/objective/uncategorised categories. People use more objective tags for tagging Wikipedia articles related to aforementioned article types. However, the tags assigned to a webpage related to a person on Wikipedia differs from the way we tag a friend on a social website. Friends on social websites are mostly tagged with subjective tags. These results could be generalised, as both Wikipedia and delicious.com are considered to be comprehensive dataset instances of knowledge bases.

In addition, we found that in social media, younger taggees are primarily assigned with more subjective tags, whereas older ones tend to be assigned with objective tags.

As taggers tend to use more subjective tags for their contacts within social platforms, this brings new challenges for recommending appropriate people-tags or making interoperable applications. Thus, it will be necessary to find some ways to handle or eliminate the subjectivity of people-tags (e.g., by defining and using controlled or semi-controlled vocabularies based on the top used people-tags), in order to increase the precision of people-tag-based recommenders and thus, usability of people-tag-based information filtering frameworks that we refer to them in future chapters.

It would be beneficial if we help users to tag each other in a semi-automated manner. In the next chapter, we present our approach for extracting, ranking and assigning expertise elements to users. Such expertise elements can be used as a way to semi-automatically tag users with their expertise.

Chapter 4

Expertise-Based People-Tags[1]

> If HP knew what HP knows, we would be three times as profitable.
>
> Lew Platt – Former CEO of Hewlett-Packard

In the previous chapter, we studied how people tag each other in social media, specifically in social blogs. Ranking tags and providing domain specific vocabularies are approaches that can be considered to eliminate the subjectivity of people-tags and increase usability of people-tag-based systems. However, manually assigning tags to users could offer additional overhead and may decrease user motivation to tag each other. Thus, besides using people-tags that have been manually assigned, in this chapter we provide semi-automated approaches that assist users to build their or others' profiles. To do such, we use online shared workspace platforms as well as question–answering (Q–A) forums that are used in organisations to ease and boost collaboration among users. Our approach helps users to semi-automatically build their expertise profiles which compose of a (ranked) list of tags (i.e., can be seen as people-tags).

4.1 Introduction and Motivation

Online shared workspaces (e.g., OrbiTeam BSCW, UNIT4 BC[2], Microsoft SharePoint) provide tools and technologies for users to collaborate via shared objects and processes. When people collaborate within online shared workspaces, they leave behind traces of their interactions and activities. These activity traces include events that happen while collaboratively producing documents (e.g., read, revise, delete) to events such as inviting a new member to the workspace. Most online shared workspaces log the activity traces in log files, which can be exported in different formats such as XML, JSON, and CSV. The log files contain valuable information about the collaboration behaviour of users.

[1] This chapter is mainly based on [Nasirifard et al., 2009, Nasirifard and Peristeras, 2009].
[2] http://www.unit4collaboration.com/

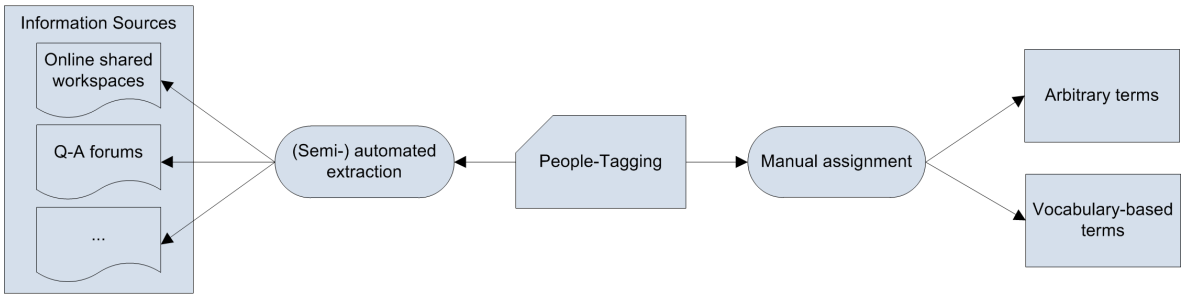

Figure 4.1: Different approaches for tagging people.

We use log files of online shared workspaces to extract latent information in order to enrich user profiles. This can be seen as a general approach for extracting tags (i.e., people-tags), as illustrated in Figure 4.1. Initially, we identify the objects, people and activities within the online shared workspace. For example, we identify the documents and people associated with them such as (co-)authors or reviewers, and the document-focused actions carried out by these people. We build a network of connected entities composed of users, actions (events), and objects such as documents. In this context, we have focused on the documents as the shared objects of collaboration. Stored documents in online shared workspaces contain rich information about collaboration activity that can be further analysed and utilised. We demonstrate a prototype called BSCW Expert Finder that extracts and assigns expertise elements to users of the OrbiTeam BSCW shared workspace.

Due to a lack of sufficient information in log files of online shared workspaces, we could not systemically rank expertise items to evaluate our expert finder framework. Therefore, we decided to adapt our approach to a different yet larger corpus which will be described as the second use case in this chapter. In particular, we focused on Q–A forums, where users ask/discuss questions and others try to address issues that are relevant to their expertise. Such active participation may be awarded by incentives (e.g., points) and we use such incentives to rank expertise. We formally define how our generalised expert finder framework work and how we evaluated it. Overall, our approach is composed of several general steps. The first step is to extract key phrases from a corpus. After extracting key phrases, they need to be ranked and assigned to users. It is generally understood that the most frequently occurring phrases in technical and mature (finished) documents such as deliverables reflect important terms of a document. In Information Retrieval (IR) and text mining domains, this is measured by Term Frequency (TF). However, if the same terms occur regularly throughout the corpus of documents, then such terms are considered not to be particularly useful. To measure this, along with TF, Inverse Document Frequency (IDF) is typically used and shows how frequent a word appears in the whole corpus. The final weight which is calculated by multiplying TF and IDF shows how important a word is (regarding the whole corpus). For example, in a corpus of Semantic Web documents, the term "Semantic" may occur in every document. Thus, it is not much good at discriminating between documents or between user expertise in this case. Such technical terms with higher TF-IDF may reflect the *expertise* of a user who has (co-)authored the document. Note that we use the term *expertise* to refer to important key phrases of a finished document (e.g., deliverable) or content of a thread in Q–A forums. Based on extracted key phrases (i.e., expertise) and document-based interaction events, as well as awarded points in Q–A forums, we build expertise profiles for users. For extracting key

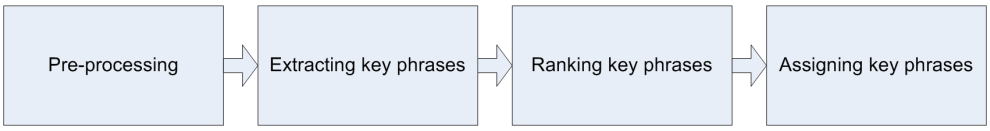

Figure 4.2: General overview of our approach for (semi-) automated extraction of people-tags.

phrases, we used various methods including machine learning as well as statistical approaches. For evaluating our approach, we used a corpus containing technical Q–A forums. We divided the corpus into two different sets: a training set and a test set. We used the training set for building expertise profiles and the test set for recommending experts, who are able to provide very helpful answer to a question. We evaluated our approach for both title as well as body of a new question posted to a forum. Our evaluation showed that the statistical approach (i.e., LDA) performed slightly better than the machine learning approach on our corpus. We also present a prototype called Sapport that was designed for building expertise profiles for users of Q–A forums. Figure 4.2 illustrates the overall view of our approach for (semi-) automated extraction of people-tags (i.e., expertise items).

The remainder of this chapter proceeds as follows: In Section 4.2, we present related work that positions our work within relevant activities in the expert finding domain. Overall, in this chapter, we discuss two use cases. The first use case (Section 4.3) is based on extracting and assigning expertise elements to users of a OrbiTeam BSCW online shared workspace. Due to a lack of sufficient data, we could not elaborate more on ranking expertise items for evaluating our approach. Thus, we decided to adapt our approach to a different use case. This use case is based on extracting and assigning expertise elements to users of Q–A forums and is presented in Section 4.4 onwards. In particular, in Section 4.4, we have an overview of this use case and details of the corpus that we used for building expertise profiles. In Section 4.5, we present various methods that we used for extracting key phrases from our corpus. In Section 4.6, we present our algorithms for ranking and assigning expertise elements to users of Q–A forums. Our simple prototype implementation is presented in Section 4.7. Section 4.8 is focused on how we evaluated our approach. Finally, Section 4.9 presents conclusions and our approaches for future work.

4.2 Related Work

In brief, expert finding deals with returning a ranked list of experts for a given topic and has attracted many researchers of Information Retrieval community over the past few years [Campbell et al., 2003, Dom et al., 2003, Balog and de Rijke, 2006, Demartini, 2007, Chen et al., 2006, Zhang et al., 2007, Jung et al., 2007, Balog et al., 2006]. Unlike early attempts towards extracting expertise of users which were mainly focused on unifying heterogeneous databases [Maron et al., 1986, Davenport and Prusak, 1998], recent attempts are mainly based on (semi-)automatically extracting expertise items from heterogeneous document collections. [Adamic et al., 2008]'s analysis on Yahoo! Answers (YA) knowledge sharing activity showed that there exist users who prefer to focus on specific topics, rather than participating in different discussions. This implies feasibility of finding experts in Q–A forums. Similar results were presented by [Zhang et al., 2007] on a technical forum which was focused

on the Java programming language.

Users who participate in Q–A forums to answer questions, have various motivations such as altruism, reputation and active learning [Constant et al., 1996, Yu et al., 2007]. Moreover, in order to encourage responders to provide faster and more accurate replies, Q–A forum moderators may provide incentives for answerers. Monetary incentives as well as social awards such as points have been used in various Q–A forums. Besides the effectiveness of incentives on quality of answers [Yang et al., 2008, Harper et al., 2008], awarding such incentives could help to identify users with higher relevant expertise. In other words, incentives such as points can be used for ranking expertise items and thus experts.

Our approaches for extracting expertise compose of analysing stored documents within online shared workspaces as well as forum contents within Q–A forums. [Seid and Kobsa, 2003] present a domain model of expert finder systems. Their general model can explain different approaches that exist for extracting expertise from different corpora. According to their model, we use document authorship and name-topic co-occurrence metrics as expertise deduction operations. Researchers use extracted expertise profiles for various purposes, from dynamic team building to helping Ph.D. candidates to find out potential supervisors [Liu et al., 2002].

We map expertise items extracted from online shared workspaces and Q–A forums to user IDs which later can be mapped to real names. Thus, we do not face one of the main problems of expert finding systems, i.e., name disambiguation [Tang et al., 2008]. We used Natural Language Processing (NLP) techniques [Schutz, 2008, Bordea, 2010, QasemiZadeh et al., 2012] as well as Latent Dirichlet Allocation (LDA) [Blei et al., 2003] for topic modelling purposes. The LDA approach is similar to probabilistic latent semantic indexing (pLSI) [Hofmann, 1999] with the difference that in the LDA approach, a Dirichlet distribution is used that remains the same for the whole corpus (for drawing topic mixture).

4.3 First Use Case: BSCW Expert Finder

From one perspective, online social networks can be divided into two main groups: object-centric and user-centric (i.e., ego-centric) [Stutzman, 2007]. In object-centric social networks, an object (e.g., document, video, music) connects people together, whereas in user-centric social networks, users are directly connected to each other.

For extracting expertise from log files of online shared workspaces, we use object-centric social networks for identifying actions that users performed on objects. Extracted user-centric social networks from log files of online shared workspaces are out of scope of this chapter, however, we provide a brief overview of a use case from this type of social network in Appendix II.

Recently, several online shared workspaces including OrbiTeam BSCW and UNIT4 BC, started to export RDF data to ease interoperability among various online workspaces. We also use the RDF data model in our work so as to make our data accessible to various platforms. For more information regarding the RDF data model, see Chapter 2.

4.3.1 Object-Centric Social Networks

A log file comprises of several log records and in each record, we assume that user ID, event name and object ID exist as a minimum. User ID is the unique identifier of a person who performs an event on an object that is also uniquely identifiable by an object ID. For example,

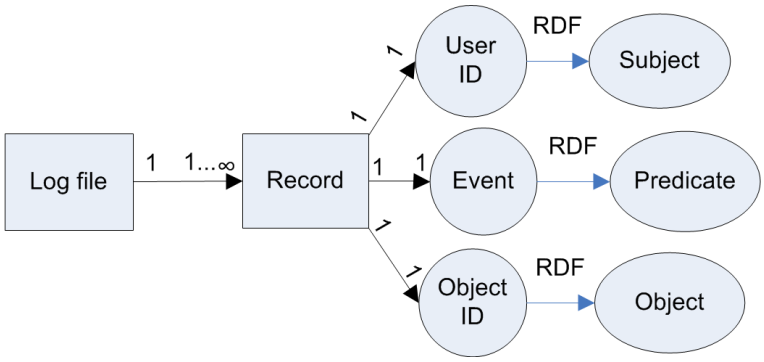

Figure 4.3: Mapping between log records and RDF concepts.

a log record in natural language can be: *the person with ID 123 revised the document with ID 456*. In addition, the log records can contain more information, such as description of the log records, temporal aspects (e.g., time-stamps) of the log records, etc.

Building an object-centric social network from such a log file is pretty straightforward. We translate the log records into RDF triples and store them in an RDF store. In order to do this, we map the main elements of the log records to RDF triples (i.e., subject, predicate and object). As Figure 4.3 demonstrates, the User ID of a record is mapped to RDF subject; the event is mapped to RDF predicate and the object ID is mapped to RDF object. As an example, in order to represent the notion *the person with ID 123 revised the document with ID 456* in RDF, there will be a subject denoting *123*, a predicate denoting *revise event* and an object denoting *456*. Obviously, a namespace will be added to the subject, the predicate and the object.

We approach the log file or extracted RDF triples in a document-centric perspective, which results in virtual clouds (i.e., containers) containing a document in the middle and several users around the document who have performed various events on that document, as illustrated in Figure 4.4.

The RDF repository is a silo of graph-like (linked) data. Thus, we need a query language to query the graph-like data. There exist several query languages for RDF. The most well-known one is SPARQL which was released as a W3C Recommendation. We use dynamic SPARQL queries for building such clouds from the RDF repository. A document-centric perspective of a log file does not make sense, unless needed for a specific use case. In the next section, we describe the use case.

4.3.2 Using Object-Centric Social Networks

We can use document-centric clouds as a means of extracting *expertise*. We define *expertise* as a piece of knowledge that has been acquired by a person in the past or the person is recognised as having that piece of knowledge. This piece of knowledge can be used as a recommender for annotating contacts and building user profiles. We extract and assign expertise in three steps which are presented in the following.

The first step is to analyse log files in order to extract document-centric clouds. The result of this phase will give us relevant information on who did what (see the previous section).

The second step is key phrase extraction. We use documents and in particular scientific

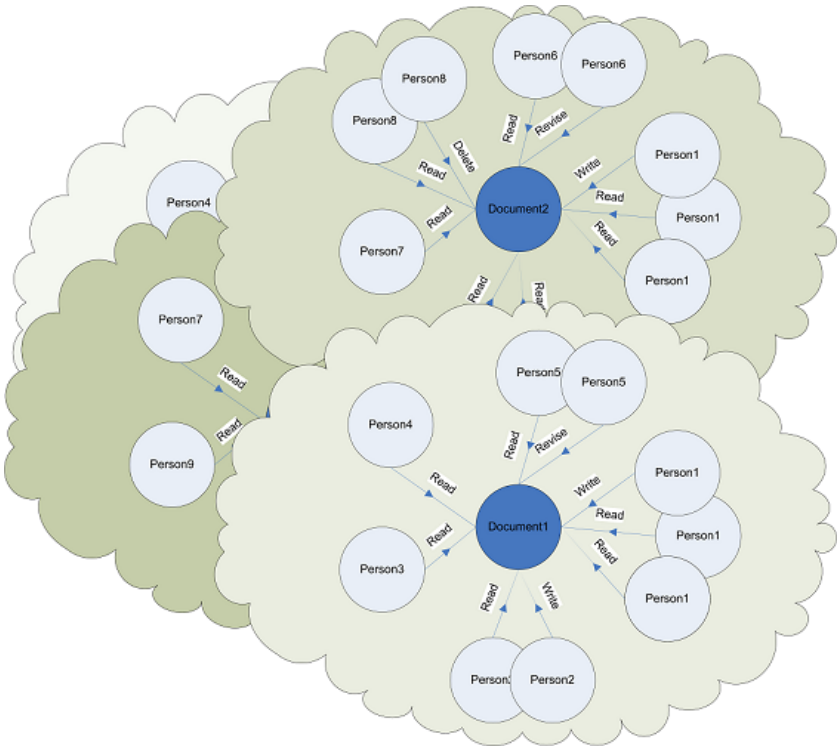

Figure 4.4: A document-centric perspective of a log file.

deliverables as input for key phrase extraction. The main reason behind this policy is the fact that deliverables are mature enough to be used for further processing, as they contain the final (research) results of project members.

Normally, in large research-oriented projects with participation of many partners, some templates are defined and used by the consortium as a way to standardise the structure of the documents that they exchange. In several pan-European research projects (i.e., Ecospace and inContext) that the author was involved in, the partners were encouraged to follow similar templates for preparing reports and deliverables. As an example, each document (deliverable) has a *version history* that contains some metadata and contribution statistics of partners (who wrote/changed what parts). Such information has potential to be used for further processing and perhaps for finding expertise. However, processing such unstructured data (e.g., in the form of tables) offers several challenges, related to text processing and name ambiguity resolution. Meanwhile, some users may unintentionally forget to update the tables or provide sufficient useful information. In addition to this, such information is not rich enough and does not include fine-grained material and concepts that could be potentially extracted using content analysis techniques.

We rely on available third-party solutions [Schutz, 2008] to extract key phrases from the documents. In brief, the result of this phase (key phrase extraction) is a list of the key concepts of a document that are later assigned to users as expertise elements based on their interactions (log file analysis). For better performance, we store the results, as we need to query them later. We store each document name plus its extracted key phrases and their confidence values in the RDF repository. The confidence value of a key phrase is a value

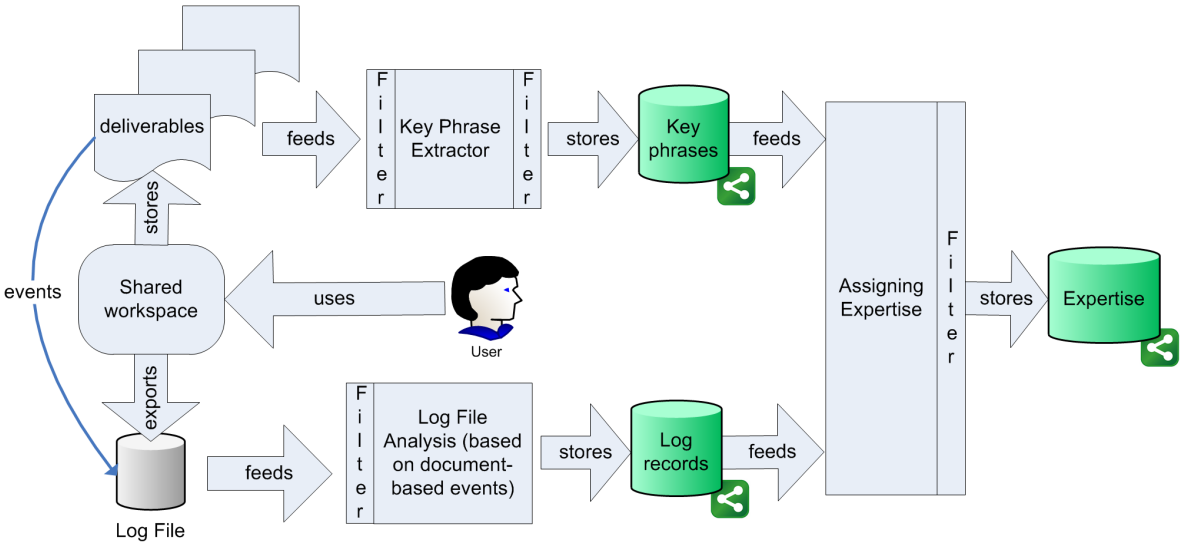

Figure 4.5: Overall view of mining and assigning expertise items to users from log files of online shared workspaces.

that is generated by key phrase extractor and can be used for ranking the phrases. In other words, it is a value that determines whether an extracted phrase can be potentially a *domain-relevant* key phrase or not (based on Natural Language Processing (NLP) techniques). We have a filtering phase after extracting key phrases to remove repetitive terms and several stop words that our key phrase extractor [Schutz, 2008] failed to identify and remove.

For the final step, our main assumption is that if somebody creates or revises a document with a topic X, then s/he has more expertise on topic X than a person that only reads that document. Towards this direction, we assign two different expertise granularities to a user: *expert in* and *familiar with*. A user is *expert in* a topic X, if s/he created or revised a document that contains the X topic. A user is *familiar with* a topic Y, if s/he just read a document that contains the Y topic. This step uses SPARQL queries that are built dynamically (using user IDs of online shared workspaces) as a means for matching expertise based on the already extracted information. We filter the result of this step too in order to restrict repetitive terms, as various documents may contain similar key phrases. Figure 4.5 demonstrates the overall view of our approach.

4.3.3 Prototype

As a proof-of-concept, we developed a prototype called BSCW Expert Finder. BSCW Expert Finder is a prototype for extracting and assigning expertise elements to users of an OrbiTeam BSCW shared workspace.

For demo purposes, we used the OrbiTeam BSCW log data of the Ecospace project. Ecospace is a European Integrated Project (IP) in the area of Collaborative Working Environment (CWE). According to its website[3], Ecospace pursues the vision that by 2012 every professional in Europe is empowered by seamless, dynamic and creative collaboration across teams, organisations and communities through a personalised collaborative working environment.

[3] http://www.ip-ecospace.org/

Event	Frequency	Max (user)	Min (user)	Mean (user)
Create	6320 (20.8%)	559	0	34.5
Revise	473 (1.5%)	21	0	2.6
Read	15201 (50.1%)	856	0	83.0
Delete	391 (1.3%)	18	0	2.1

Table 4.1: Event statistics of Ecospace log files from March 2005 to December 2008.

One of the main objectives of Ecospace is to create new tools and techniques to ease collaboration and communication among people. Ecopsace partners use the OrbiTeam BSCW shared workspace as a platform to synchronise their activities and project-related achievements. OrbiTeam BSCW logs all transactions and events. The events can be exported as a CSV file or can be accessed via API.

As input for extracting expertise, we used around fifty documents (mostly deliverables) of the Ecospace project in PDF format, which were submitted to four review sessions of the project. On average, our key phrase extractor extracted forty key phrases with various confidence values (between 0.3 and 0.8) for each deliverable. We used log files of the OrbiTeam BSCW shared workspace of the Ecospace project activities from March 2005 to December 2008. This log file contains 30,341 records. 183 users (active and inactive) were extracted from the log file. Table 4.1 shows the distribution of four main document-based events in comparison with other events (and whole log records).

Figure 4.6 demonstrates several snapshots of our BSCW Expert Finder. The prototype enables authorised users to search for users with appropriate expertise as well as browsing detailed expertise of users. The result will be shown in two orders: users who are expert in a topic, and users who are familiar with a topic. The prototype is accessible online[4].

The BSCW Expert Finder provides two Web services that wrap its functionalities. The *Extract Expertise* service receives as input the BSCW username of a user and returns his/her expertise in RDF or XHTML formats which can be simply transformed into any other XML formats using XSLT. The other service is called *Handle Data* and enables authenticated users to lodge data into RDF repository. This service accepts as input an XML file which contains raw data. The authentication is based on username/password and is done via an XML input file. After a successful transaction, the input log data will be stored in the repository. This service is currently not accessible for public. The *TeamBuilder* tool from TXT Group[5], a tool for creating dynamic teams, uses BSCW Expert Finder Web services for finding people with required expertise.

4.3.4 Discussion

Unfortunately, OrbiTeam BSCW log files can contain noisy records. Noisy log records are those ones that do not include user ID, object ID or event name. In order to remove noisy log records, we carried out a filtering phase before using them. To do this, we defined some patterns and those records that did not follow the patterns were automatically removed from

[4] http://purl.oclc.org/projects/expertui
[5] http://www.txtgroup.com/

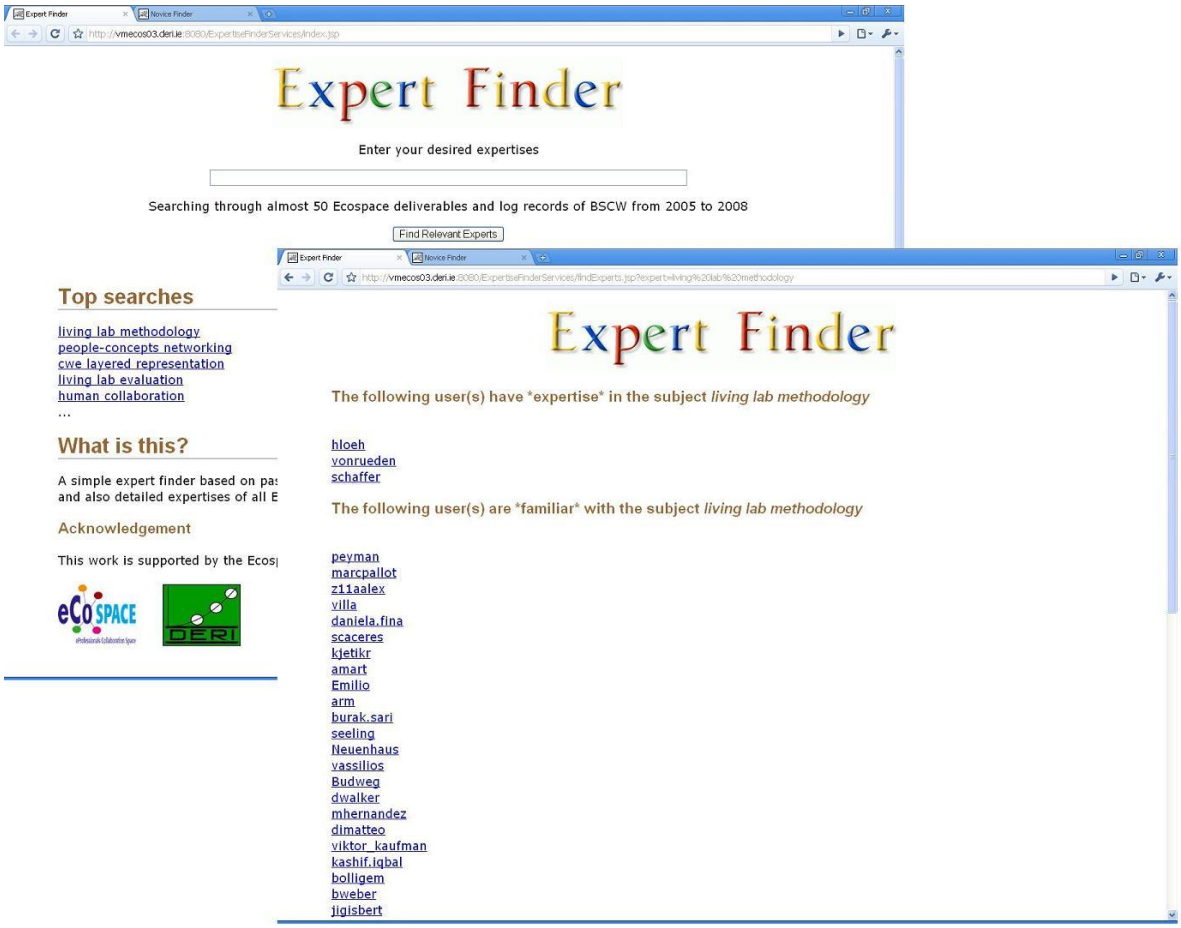

Figure 4.6: Two snapshots of BSCW Expert Finder: a tool for extracting and ranking expertise items to users of an OrbiTeam BSCW.

the process. The patterns include:

- The date and time of a log record should be a valid date and time, i.e., all elements of date and time should be valid.
- User ID, username, object ID, object name, event ID and event name should not be `null`, i.e., a string with zero length.
- All IDs should be positive numerical values.
- All events should end with the term `Event`.

We did an experimental evaluation for BSCW Expert Finder. We asked 12 members of the Ecospace project to take a look at their extracted expertise. All of them confirmed that the extracted phrases are relevant to the concepts that they are familiar with or expert in. However, such subjective feedback could not give us enough data to measure how accurate our BSCW Expert Finder operates. We could not rank expertise items and evaluate our framework systematically. Moreover, our dataset (i.e., Ecospace deliverables) was not comprehensive enough for building fine-grained expertise profiles. Therefore, we decided to adapt

our approach to a larger corpus which offered us new challenges. To this end, we focused on Q–A forums which will be discussed in the following sections.

4.4 Second Use Case: Q–A Forums

Due to a lack of sufficient data in our first use case, we adapted our approach to Q–A forums and in particular, we focused on SAP Community Network (SCN) forums[6] due to several reasons. First of all, SCN forums were established more than 8 years ago and thus the corpus that we could get from the forums were comprehensive enough for building expertise profiles and further evaluation. Moreover, SCN forums are composed of different categories, each contains various forums. Dividing threads into different categories eases finding, classification and assigning expertise elements (see Section 4.5.4 for more details). Last but not least, SCN forums have a mechanism for awarding points to helpful replies. The points help to rank expertise items and thus experts. In the following section, we give an overview of the SCN forums corpus.

4.4.1 Corpus Details

In each SCN forum, there exists various threads, each containing one or more posts. Table 4.2 shows statistics of data that we used for building our expert finder framework and evaluating it. We used 33 different SCN forums. In total, 32,724 unique users initiated or contributed to 95,017 threads containing 415,121 posts dated between 15-Dec-2003 and 27-Jan-2011. The data was provided by SAP in the context of the ROBUST project[7].

4.4.2 User Points

In SCN forums, there exists a mechanism for awarding points to useful replies. According to the SCN website[8], points are assigned to posts replying to a thread marked as "question" as follows:

- Two points: helpful reply. This may happen unlimited times in a thread.
- Six points: very helpful reply. This may happen maximum twice in a thread.
- Ten points: solved the problem. This may happen maximum once in a thread.

Moreover, they mention that points can be awarded by the initiator of a thread or by a forum moderator. To encourage users to assign points, the initiator receives one point when s/he awards points to a helpful answer and this may happen maximum once in a thread. Note that according to the SCN policies points are not restricted to only forum posts and they may be awarded by contributing in other areas like blog posts too. In our corpus of 415,121 posts, points were assigned to 78,106 posts (18.8% of the total posts).

[6] http://scn.sap.com
[7] http://robust-project.eu/
[8] http://www.sdn.sap.com/irj/scn/crphelp

Index	Forum Title (Forum ID)	#Threads	#Posts
1	Enterprise Social Systems (486)	3	7
2	Green IT (468)	8	39
3	GS1 Standards and SAP (485)	14	44
4	ASAP Methodology and Project Management (482)	36	116
5	Sustainability (281)	42	190
6	Organizational Change Management (200)	47	227
7	International Financial Reporting Standard (IFRS) (400)	78	289
8	Operational Performance Management - General (Spend Performance, SCPM, Other) (411)	89	394
9	SAP Business One Training (420)	119	474
10	Standards (201)	163	367
11	Analytics (210)	170	488
12	SAP Business One Partner Solutions (Add-ons) (354)	184	664
13	Best Practice and Benchmarking (319)	214	483
14	Manufacturing Execution (ME) (470)	301	1,403
15	Business Process Modeling Methodologies (198)	305	950
16	SAP Strategy Management (414)	399	1,905
17	SAP Discovery System for Enterprise SOA (226)	408	1,115
18	SAP Business One Integration Technology (161)	812	3,132
19	SAP Business One Product Development Collaboration (265)	1,127	2,570
20	SAP Business One E-Commerce and Web CRM (252)	1,389	4,482
21	Financial Performance Management - General (PCM, FC, Other) (270)	2,483	8,871
22	Service-Oriented Architecture (101)	2,570	9,516
23	Business Process Expert General Discussion (197)	2,609	7,430
24	SAP Business One System Administration (419)	3,219	16,429
25	Business Planning and Consolidations, version for SAP NetWeaver (412)	3,463	14,315
26	SAP Business One - SAP Add-ons (418)	3,987	19,333
27	Business Planning and Consolidations, version for the Microsoft Platform (413)	4,245	18,689
28	Governance, Risk and Compliance (256)	4,279	18,914
29	Process Integration (44)	4,907	27,102
30	SAP Business One Reporting and Printing (353)	7,744	37,943
31	ABAP, General (50)	13,262	54,118
32	SAP Business One Core (264)	17,838	84,316
33	SAP Business One SDK (56)	18,503	78,806
	Total: 33 forums	95,017	415,121

Table 4.2: The SCN forums data statistics. The table is sorted based on the total number of threads in each forum. The first column is simply an index for each forum. The second column shows title of the forum plus its ID. The third column demonstrates the total number of threads in each forum and the last column shows the total number of posts in each forum.

4.5 Pre-Processing and Key Phrase Extraction

Overall, our expert finder framework is composed of two main elements. The first component extracts key phrases (i.e., expertise items) from a corpus and the second component assigns key phrases (i.e., expertise items) to users based on their contribution history.

In order to extract key phrases from our corpus, we decided to consider a *thread* as the smallest element for processing, meaning that instead of analysing single posts, we analyse threads. The reason behind this policy is the fact that threads are composed of posts and posts of a thread typically address/discuss one or more issue(s) and this/these issue(s) can be seen as key concepts of that thread. Moreover, key topics of a thread may happen more than once in different posts of a thread. This helps identifying key topics of a thread, as most key phrase extractors use statistical approaches for extracting key phrases.

For extracting key phrases, we decided to analyse both the title and the body of a thread. It is common in Q–A forums that users who ask questions or seek information, try to choose a relevant title for initiating a thread. In other words, the title of a thread could be a short, yet very relevant summarisation of the thread. Thus, besides looking at thread bodies, we also evaluated how accurately we can recommend relevant experts by only looking at titles.

4.5.1 Pre-Processing

The SCN forums data was gathered by crawling HTML pages of the forums website. Thus, it contains HTML elements such as *
* tag. These tags can be seen as a word by key phrase extractors and thus may generate noise. Using regular expressions, we removed HTML tags and then merged all posts of a thread in order to pass it to the key phrase extractor.

We also removed threads that had a length less than 10 characters, because our analysis on such threads showed that they do not contain useful information. For example, a thread contained just the "thank you" phrase and thus was removed from further processing.

Unlike extracting expertise from well-structured deliverables (like we explained in the previous section), here we have forum data, where users tend to use conversational (chat-like) texts. As an example, on a forum users may use social greetings such as *hi*, *thanks*, *regards*, etc. which may be identified as a topic by key phrase extractors. We identified such phrases by looking at highly frequent terms and removed them before processing any further.

In the following sections, we present different approaches that we considered for extracting key phrases.

4.5.2 Linear Classification

Initially, for extracting key phrases we followed the same approach as we performed for extracting key phrases from structured deliverables (see Section 4.3.2). We used Saffron [Bordea, 2010], an extended work of [Schutz, 2008] for extracting key phrases. Saffron gets a corpus as input, indexes all tokens and returns a list of potential key phrases in a text file. We were advised that Saffron is more suitable for structured and well-formed documents and not conversational documents such as Q–A forum contents. In order to briefly evaluate Saffron results, we calculated the frequency of extracted key phrases and sorted the results based on frequency. Looking at the top-1000 tags revealed that results are too noisy, as top

tags were composed of words that are often used in chats such as "let me know", meaningless terms, or usernames who contributed a lot to the forums. The reason is perhaps due to the conversational behaviour of forums, where users ask questions and others try to solve those issues, sometimes by explicitly mentioning their usernames to address a post to them.

To overcome the drawbacks of Saffron, we decided to replace Saffron with a linear classification approach (i.e., machine learning) [QasemiZadeh et al., 2012] which was mainly designed to extract technology-related terms from an unannotated corpus. [QasemiZadeh et al., 2012] mention that it is possible to build a language model for automatic extraction of technology-related terms (TRT) from an unannotated corpus by limited use of user feedback. This approach uses user-defined seed terms (e.g., technology) to build a dataset of positive and negative records which are later used for training purposes. This approach uses linear classification, as it tends to be faster than similar methods such as support vector machine (SVM). The benefit of such an approach is that it analyses each sentence and thus does not require the whole corpus. This is important, as it can quickly extract key phrases from new posts – for use cases such as building real-time alerting systems. In order to use this approach, we were advised not to remove any words (including stop words) in pre-processing steps, as this may affect identifying tokens and thus sentences. We followed this advice, however, we just removed HTML tags. We used an in-house implementation of linear classification for extracting technology-related terms which was provided as a Web service [QasemiZadeh et al., 2012]. The result is returned with a confidence value for each extracted term, indicating to what extent an extracted key phrase can potentially be a relevant technology-related key phrase. However, we noticed that some key phrases with low confidence values were actually relevant technology-related key phrases in our corpus. Thus, we omitted confidence values and returned all technology-related key phrases as potential key phrases in a thread.

The only post-processing step that we applied on the result was to remove common conversational terms (like *regards*) as we did not remove them in the pre-processing step.

We extracted key phrases for titles as well as bodies of the threads and stored the results (keywords plus confidence values) into our MySQL database. Unlike in the previous case, where we used an RDF store for storing our data, here we used databases since SCN forums data was provided to us as an SQL dump and we decided to keep its underlying schema. Obviously, this data can be easily transformed into RDF, if desired. In our corpus, thirteen threads were removed from further processing, as the key phrase extractor failed to identify sentences in them. Taking a closer look at these threads showed that they were mainly a mixture of programming language source codes and software exception traces.

Besides using linear classification for extracting technology-related terms, we decided to use other approaches, in order to compare the extracted key phrases of different approaches. As a candidate, we decided to use Luxid.

4.5.3 Luxid

Luxid is a natural langauge processing (NLP) engine/product from Temis[9] for analysing documents and extracting information/knowledge. Luxid can be equipped/plugged with various components called *Skill Cartridges*. Each skill cartridge is capable of analysing a specific aspect of input text and returning the results in various formats such as XML, RDF or HTML.

[9] http://www.temis.com

Such plug-and-play architecture increases the extensibility of Luxid. In the following, we take a closer look at skill cartridges that we used for extracting information. The cartridges were provided to us in the context of the ROBUST project.

- Nominal-Verbal-Adjectival phrases: This cartridge returns names, verbs and adjectives of a sentence.
- Sentiment analysis: This cartridge returns positive/negative phrases in a text. For example "like" is perceived as a positive phrase, whereas "do not like" is perceived as a negative sentiment.
- Named-Entity extraction: This cartridge returns named entities, like locations, people names, companies, organisations, products, and so on.
- General Relevant Phrases: This cartridge extracts key topics relevant to a domain. This cartridge is more focused on nominal elements and tries to pick those items that seem to be "more informative".

Luxid skill cartridges are great for identifying named entities like locations and person names. But the drawback is that not all expertise elements (such as ABAP, web service, user management, etc.) can be classified as named entities. Thus, none of Luxid cartridges were suitable for our purpose. Moreover, the cartridges were designed to analyse well-formed documents (e.g., documents with proper case-sensitivity). As an example, Luxid named-entity extraction cartridge can identify *Microsoft* as the name of a company, but it fails to recognise the lower case (i.e., *microsoft*). Therefore, in a conversational and chatty environment like Q–A forums, where users may not put much effort on proper punctuation, Luxid is not suitable.

Due to such drawbacks, we tried to identify other approaches to replace Luxid. We chose to try Latent Dirichlet Allocation (LDA).

4.5.4 LDA

The Latent Dirichlet Allocation (LDA) is a generative probabilistic topic model for discrete data such as text corpora. Initially, this approach finds co-occurring words and then it groups them together to form a topic. Then, the topics will be assigned to each piece of text in the corpus. For example, the terms such as *RDF*, *OWL*, *Semantic Web*, *Web*, and *linked data* can form a topic together, if they co-occur frequently in a corpus. For more information on LDA, see [Blei et al., 2003]. For applying the LDA analysis to our corpus, we used MALLET[10] (MAchine Learning for LanguagE Toolkit), a free open source library for statistical natural language processing such as LDA. MALLET does its job in two steps. First, it gets a corpus and the n-gram settings (i.e., length of key phrases) as input and generates a so-called mallet file. The mallet file contains tokens of the corpus. In the next step, which is composed of extracting m topics and assigning them to each piece of text, MALLET does not require the corpus anymore. The output of this step is composed of two files: one XML file and one text file. The XML fie contains key topics of the corpus. A set of keywords (e.g., 20 keywords in our case) with different n-grams (e.g., one, two and three in our case) will be identified and assigned to each topic. In other words, this file contains (topic ID, key phrase) pairs.

[10] http://mallet.cs.umass.edu/

Topics	Key phrases
Topic 1	system, web, language, version, string, windows, page, synch, webtools, file, object, netpoint, assembly, de, english, exception, forms, dll, message
Topic 2	service, web, sap, services, web_service, wsdl, webservice, http, proxy, call, create, soap, url, system, ws, abap, client, web_services, ecc
Topic 3	tax, price, amount, total, code, discount, list, price_list, vat, excise, tax_code, freight, prices, unit, decimal, item, document, special, unit_price
Topic 4	java, sap, java_sap, engine, sap_engine, java_sap_engine, services, engine_services, sap_engine_services, server, virsa, impl, thread, service, ae, virsa_ae, core, java_virsa, thread_impl
Topic 5	server, client, sql, install, installed, sql_server, windows, sap, installation, machine, folder, connection, microsoft, connect, run, local, manager, running, files

Table 4.3: Assigned key phrases to five sample topics after applying the LDA to our corpus (n-gram=1,2,3).

Table 4.3 shows assigned key phrases to five sample topics after applying LDA to our corpus. The other output file of the LDA analysis (i.e., the text file) shows relevancy level of each topic to each corpus item. This file contains (corpus item, topic ID, confidence value) triples, where confidence values represent relevancy scores of topic IDs to corpus items. The higher the confidence value, the more likely that the extracted key topic can represent the corpus item more accurately.

As LDA requires the number of topics to be specified as a parameter, we wished to investigate behaviour of a varying number of topics in our corpus. We evaluated this hypothesis in the context of expert finding by measuring how accurately we can recommend experts based on various number of topics. In particular, we focused on six different numbers of topics: 50, 100, 200, 300, 500, and 1000. For more information on this, see Section 4.8.

We also had the hypothesis that if we break down the corpus into forums, we may get more relevant results by applying the LDA to each forum in our corpus. To evaluate this hypothesis, we ran MALLET on 1) the corpus as a whole (both for titles and thread bodies) and 2) divided corpus into forums (both for titles and thread bodies). All instances of MALLET ran on a server composed of 24 cores (2.40 GHz each) and 96 GB of RAM. We also set the MALLET to remove stop-words and be case-insensitive. We set the LDA to generate maximum 20 key phrases for each topic. Despite the fact that we were interested in key phrases with the length (i.e., n-gram) of 1, 2 or 3, we evaluated our hypothesis against different n-grams. The LDA analysis generates log-likelihood values for each (key phrase, topic) pair (see Equation 4.1) and the goal is to maximise this value. In other words, the bigger the log-likelihood value, the more likely that a key phrase belongs to a specific topic.

$$\log(P(\text{key phrase} \mid \text{topic})) \tag{4.1}$$

We used log-likelihood (LL) values of the output after the 1000th iteration to evaluate which combinations work better. Figure 4.7 shows the LL-values of applying the LDA to the corpus as a whole as well as divided into forums for both titles and thread bodies (n-gram=1,2,3). For other comparisons, see Appendix III. Note that the LL-value of the forum-based corpus

(both titles and thread bodies) is the average value for all 33 forums after the 1000th iteration. Figures 4.8 and 4.9 show the LL-value for each forum (i.e., not average) after the 1000th iteration for n-gram=1,2,3. The results show that dividing the corpus into forums give us better results for various number of topics for both titles and thread bodies, as the LL-value is bigger than the non-forum-based corpus. Thus, we used the forum-based corpus for our expert finder framework.

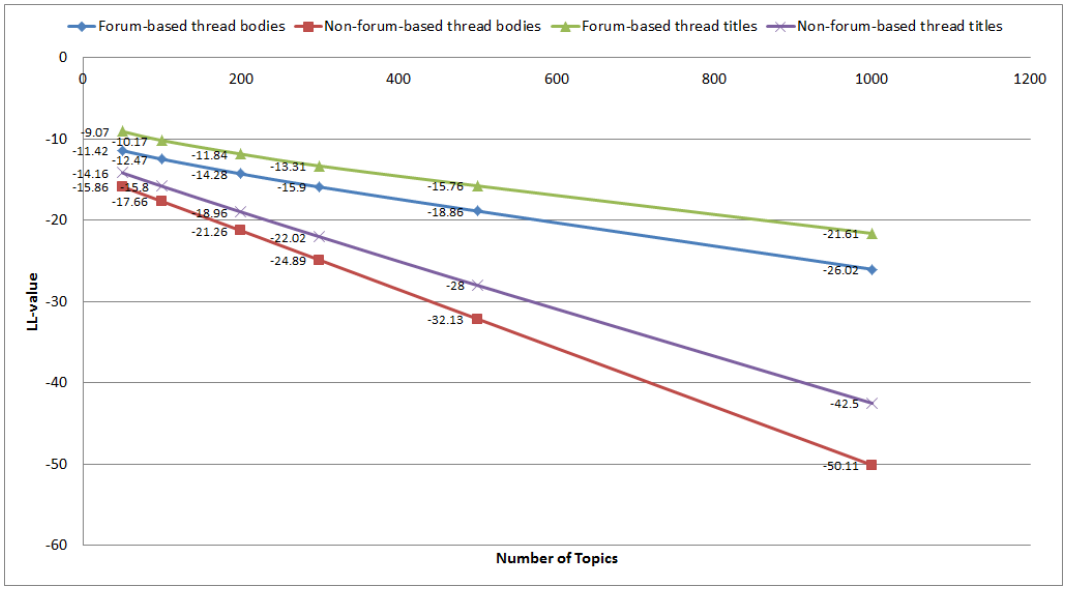

Figure 4.7: The LL-values for different number of topics after applying the LDA to our corpus. For the LDA analysis we considered both the corpus as a whole and also the divided corpus into forums. Thus, our analysis constituted four different settings as follows: forum-based thread bodies: divided thread bodies based on their forums; non-forum-based thread bodies: thread bodies as a whole; forum-based thread titles: divided thread titles based on their forums; non-forum-based thread titles: thread titles as a whole. The X-axis shows number of the topics. The Y-axis shows the LL-value after the 1000th iteration (#n-gram=1,2,3, #topics=50,100,200,300,500,1000, #keywords in each topic=20).

Note that although we explicitly set the number of topics in the LDA, if it is unable to produce exactly that number, it will automatically reduce the total number of the topics.

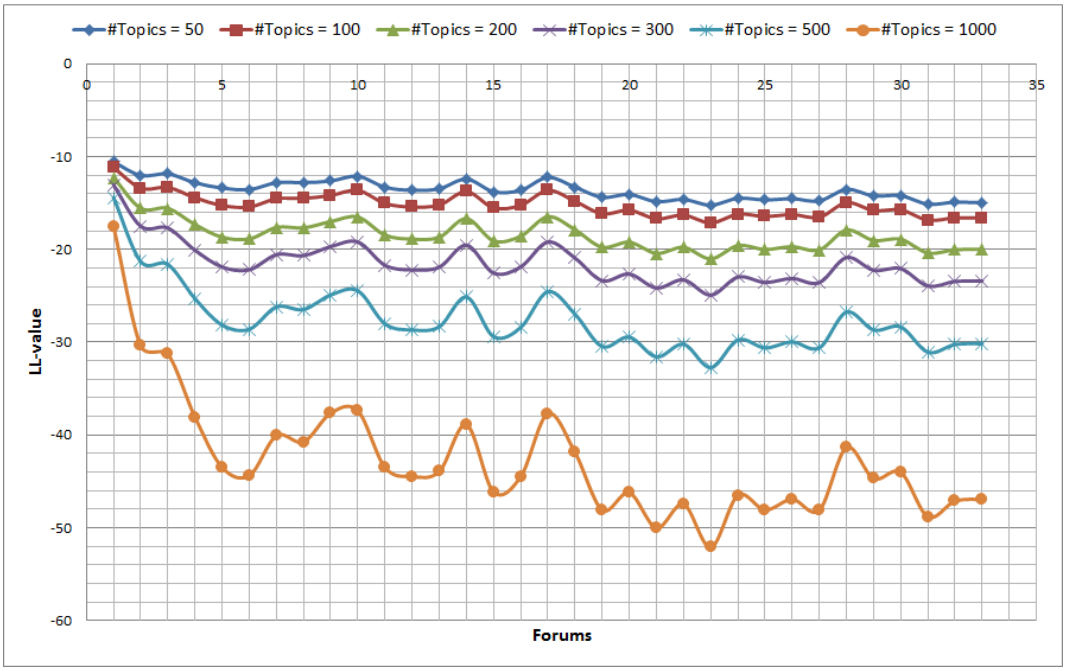

Figure 4.8: The forum-based LL-values of the thread bodies corpus after applying the LDA. The X-axis shows the index of the forums (i.e., sorted based on the total number of threads in each forum – see Table 4.2). The Y-axis shows the LL-value after the 1000th iteration (#n-gram=1,2,3, #topics=50,100,200,300,500,1000, #keywords in each topic=20).

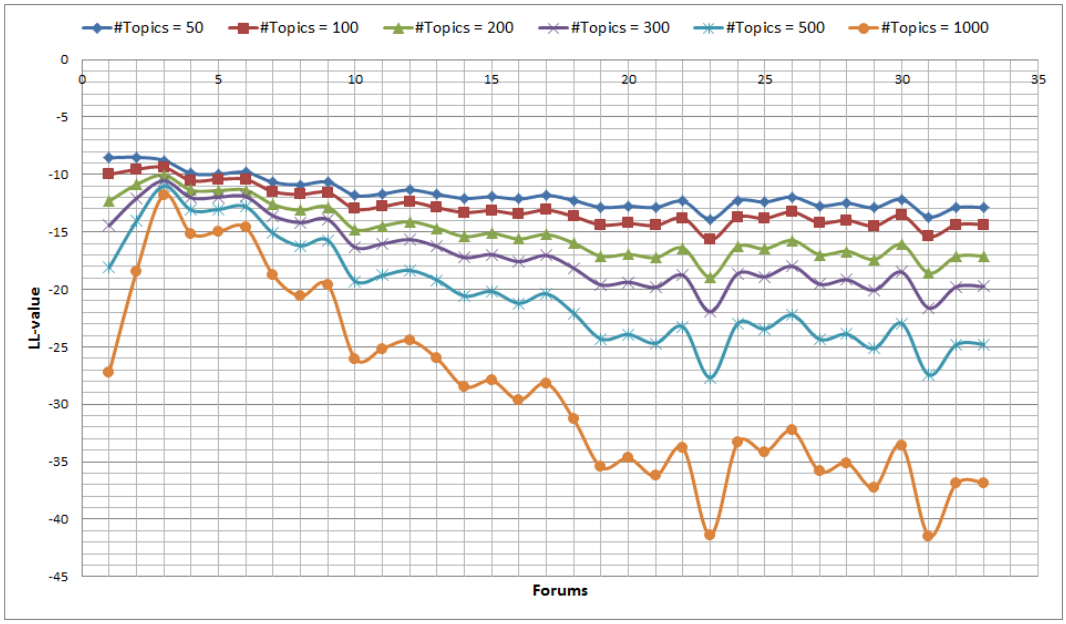

Figure 4.9: The forum-based LL-values of the thread titles corpus after applying the LDA. The X-axis shows the index of the forums (i.e., sorted based on the total number of threads in each forum – see Table 4.2). The Y-axis shows the LL-value after the 1000th iteration (#n-gram=1,2,3, #topics=50,100,200,300,500,1000, #keywords in each topic=20).

4.6 Ranking and Assigning Expertise

Unlike the previous use case (see Section 4.3), where we analysed *read-revise-write* events of online shared workspaces for building either expertise or familiarity profiles (i.e., a binary case), here we build our profiles using points that were awarded to users based on their contribution history. These points can give us more fine-grained possibilities for ranking expertise items. On the other hand, it is very difficult/subjective to put a distinctive border between the points (i.e., make separate expertise and familiarity profiles), however, we could assume that if someone is awarded with 6 or 10 points in relation to a topic, then said person is expert in that topic and gaining less than 6 points (i.e., 1 or 2 points) can be perceived as familiarity. Obviously, this classification could be highly subjective. For example, someone may argue that only replies that solved an issue (i.e., got 10 points) can be considered as expertise elements. Therefore and due to subjectivity of the points, here we build a single profile for all users, which composes of a ranked list of expertise items. The lower-ranked items could be perceived as items that the user is not expert in, but familiar with.

Besides points, we also consider the number of replies by each person in each thread (except for the thread initiator), meaning that each user gets one additional point for posting a reply in a thread. Our assumption is that each reply in a thread is an attempt to solve an issue which the first user initiated. Thus, assigning one point to each reply in a thread is a complementary approach to identify top contributors in a thread. We do not consider multiple replies by the thread initiator, as s/he may post multiple replies to a thread to clarify his/her problems/issues.

To make it more clear how we build a ranked list of expertise items for each user as well as a ranked list of experts for an item, in the following, we present formal definitions.

We assume that there exists a set of users U and n is the number of users:

$$U = \{u_1, u_2, ..., u_n\}, |U| = n$$

Let P be the set of posts:

$$P = \{p_1, p_2, ..., p_m\}, |P| = m$$

One or more posts make a thread. We define the set of threads as follows:

$$T = \{t_1, t_2, ..., t_k\}, |T| = k$$

where $t_i = (p_i \in P, p_j \in P, ...)$ such that for $i < j$: $(postedDate(p_i) < postedDate(p_j))$.

One or more threads make a forum:

$$F = \{f_1, f_2, ..., f_l\}, |F| = l$$

where $f_i = (t_i \in T, t_j \in T, ...)$.

The users initiate or contribute (i.e., posting replies) to threads. We define a set PU that is composed of user and post pairs, meaning the posts that were posted by users.

$$PU = \{(u_i, p_j) \mid u_i \in U \wedge p_j \in P\}$$

We need several helper functions as follows. The $getThreadPosts(t_k)$ function returns (chronologically sorted) posts of thread t_k. The $getUsers(t_k)$ function returns list of users

who initiated or contributed to thread t_k. It is defined as follows:

$$getUsers(t_k) = \{u_i \in U \mid (u_i, p_j) \in PU \wedge p_j \in getThreadPosts(t_k)\}$$

The $getThread(p_j)$ function returns the thread that the post p_j belongs to. The $getThreads(u_i)$ function returns all threads that a user either initiated or contributed to. It is defined as follows:

$$getThreads(u_i) = \{t_k \in T \mid (u_i, p_j) \in PU \wedge t_k = getThread(p_j)\}$$

The $getUserPosts(u_i, t_k)$ returns posts that u_i posted to t_k. It is defined as follows:

$$getUserPosts(u_i, t_k) = \{p_j \in P \mid (u_i, p_j) \in PU \wedge getThread(p_j) = t_k\}$$

As we process our corpus at a *thread* level, we define a function $getKeywords(t_k)$, where it returns pairs of key phrases and their confidence values (calculated by the NLP or the LDA analysis) of a thread: (w_i, c_j) – w_i is the key phrase and c_j is its confidence value.

The $getPoint(u_i, t_k)$ function returns the point value that was awarded to u_i in t_k. We define $getScore(u_i, t_k)$ that returns the score of u_i in t_k. As we explained, for calculating score we also consider the number of posts in a thread as a contribution history.

$$getScore(u_i, t_k) = \begin{cases} 1: \text{ if } u_i \text{ is a thread initiator} \\ getPoints(u_i, t_k) + |getUserPosts(u_i, t_k)|: \text{ otherwise} \end{cases}$$

After defining our helper functions, in Algorithm 1, we present pseudocode for building expertise profiles of users. The expertise profile of a user is composed of a weighted list of keywords, where keywords with larger weights represent topics that the user has more expertise in. Algorithm 1 does not depend on the method that is used for extracting key phrases or key topics as it only requires (key phrase, confidence value) pairs and both NLP and LDA approaches will generate such pairs. The input to Algorithm 1 is the user ID of u_i and the output will be the expertise profile of u_i. Initially, the algorithm fetches all threads that u_i initiated or contributed to. For each thread that u_i either initiated or contributed to, the *score* of u_i will be calculated. After that, by using NLP or LDA analysis, the algorithm will generate (key phrase, confidence value) pairs. After these steps, (key phrase, score) pairs will be added to the expertise profile of u_i. The variable α in Algorithm 1 can be used for tuning the score. For example, it can be replaced with or influenced by confidence value (i.e., c_q). In other words, the confidence value of an extracted key phrase (i.e., c_q) may have influence on the score that someone may acquire in relation to that key phrase. For example, a developer may decide to multiply the *score* value by confidence value to imply that keywords with higher confidence values have more influence on the *score*.

The other scenario for our expert finder framework is to query over a topic to get a ranked list of experts in relation to that topic. The pseudocode in Algorithm 2 demonstrates this piece of functionality.

The input to Algorithm 2 is a topic t and the output is a ranked list of experts. Initially, the algorithm finds all threads that are related to the t topic (e.g., containing this topic). This task is performed by the $findThreads(t)$ function. Then, for each thread t_k that is related to the t topic (i.e., the output of the $findThreads(t)$ function), we fetch all users who posted

input : $u_i \in U$
output : $profile(u_i)$
$profile(u_i) \leftarrow \emptyset$;
foreach $t_k \in getThreads(u_i)$ **do**
  $score \leftarrow getScore(u_i, t_k)$;
  foreach $(w_p, c_q) \in getKeywords(t_k)$ **do**
    $profile(u_i) \leftarrow profile(u_i) \cup (w_p, score * \alpha)$ // α is a parameter for tuning score. It can be replaced with or influenced by c_q;
  end
end
return $profile(u_i)$;

Algorithm 1: Building Ranked Expertise Profile.

at least one post to the t_k thread. After that, for each user u_i that posted at least one post to the t_k thread, we calculate the *score* of u_i in t_k. Then (u_i, *score*) pair will be added to the result set. If the result set already contains u_i, then u_i's score will be updated accordingly.

The variable α in Algorithm 2 can be used for tuning the score. For example, it can be influenced by importance of the forum that the t_k thread belongs to, number of views of the t_k thread, creation date of the t_k thread (i.e., expertise may have life cycle) and so on. Above, the two algorithms were simplified to represent the main flow and thus pre-processing steps such as stemming of the t topic were not presented in the body of the algorithms. A threshold can be passed to both algorithms in order to get only expertise/experts who acquired a score higher than the threshold. Such thresholds can be used for dividing an expertise profile into two sub-profiles (i.e., expertise and familiarity), if desired.

input : topic t
output : $result = \{(u_1, rank_1), (u_2, rank_2), ..., (u_i, rank_i)\}$ such that if $i > j$ then $rank_i > rank_j$
$result \leftarrow \emptyset$;
foreach $t_k \in findThreads(t)$ **do**
  foreach $u_i \in getUsers(t_k)$ **do**
    $score \leftarrow getScore(u_i, t_k)$;
    if $(result.contains(u_i)$ **then**
      $result \leftarrow result \cup (u_i, updateScore(u_i, score * \alpha))$;
    end
    else
      $result \leftarrow result \cup (u_i, score * \alpha)$;
    end
  end
end
return $result$;

Algorithm 2: Querying for Ranked List of Experts.

4.7 Prototype

We implemented a prototype called Sapport that enables end users to perform two kind of searches: 1) search for experts with relevant expertise, and 2) fetch all expertise elements of

a user. Figure 4.10 demonstrates a snapshot of a Sapport result pane. Sapport is able to return a ranked list of users who are expert in a topic. In order to convince end users that the rankings are relevant, Sapport provides text-based and visual explanations in the form of pie charts. The explanation is composed of the history of threads that a user has contributed to. For returning the expertise elements of a user, Sapport enables an end user to choose between the NLP and the LDA approaches, as well as searching over thread bodies or titles. As search phrases consist of arbitrary phrases, we use the Porter stemming algorithm [Porter, 1980] to stem the searching phrases. The Sapport prototype is accessible online[11].

4.8 Evaluation

In the following section, we present how we evaluated our expert finder approach and also the detailed evaluation results.

4.8.1 Evaluation Methodology

We required two sets of threads namely training set and test set for evaluating our expert finder. We used the training set for finding users with relevant expertise and the test set to evaluate our approach. For each thread in the test set, we tried to identify key phrases by looking at the first post of the thread (i.e., asked question) as well as the title of the thread, as we wanted to evaluate how accurately we can recommend experts when only looking at the title of a new question. Our experiment is composed of two parts. For the first part of our experiment, we used all threads that were published before 2011-January-01 as our training set (i.e., 93364 threads) and those threads that were published on or after 2011-January-01 as our potential test set (i.e., 1653 threads). Among our potential test set candidates, we narrowed down and picked those ones for which at least one user provided a very helpful answer or solved the issue (i.e., got 6 or 10 points) in them. In total, 492 threads fulfilled this requirement and were used for evaluating our approach for the first experiment.

For both title and first post of a thread, we extracted key phrases by using two methods: LDA and NLP (as explained in the previous section). Note that the NLP method we used does not require looking at whole corpus, whereas the LDA approach needs the whole corpus. We ran the LDA with n-gram=1,2,3 for forum-based corpus (i.e., divided the corpus into 33 forums) with a varying number of topics: 50, 100, 200, 300, 500, and 1000. This could show us the behaviour of our expert finder framework based on a varying number of topics.

We extracted all key phrases (using NLP or LDA analysis) for each thread in our test set (body or title). Then for each extracted key phrase, we used Algorithm 2 to create a ranked list of experts in relation to said key phrase. We then merged the ranked lists of experts for each key phrase to build a single ranked list of experts for the thread. If a user u_i popped up as an expert in the result list of various key phrases, we summed up u_i's score for building our single ranked list of experts for the thread. The single ranked list of experts can be perceived as what we present to end users as potential candidates for answering a question. We then identified the ranking of the user who actually provided a very helpful answer to a test set thread in our result set to check if we can provide a relevant expert at a higher rank.

[11] http://purl.oclc.org/projects/sapport

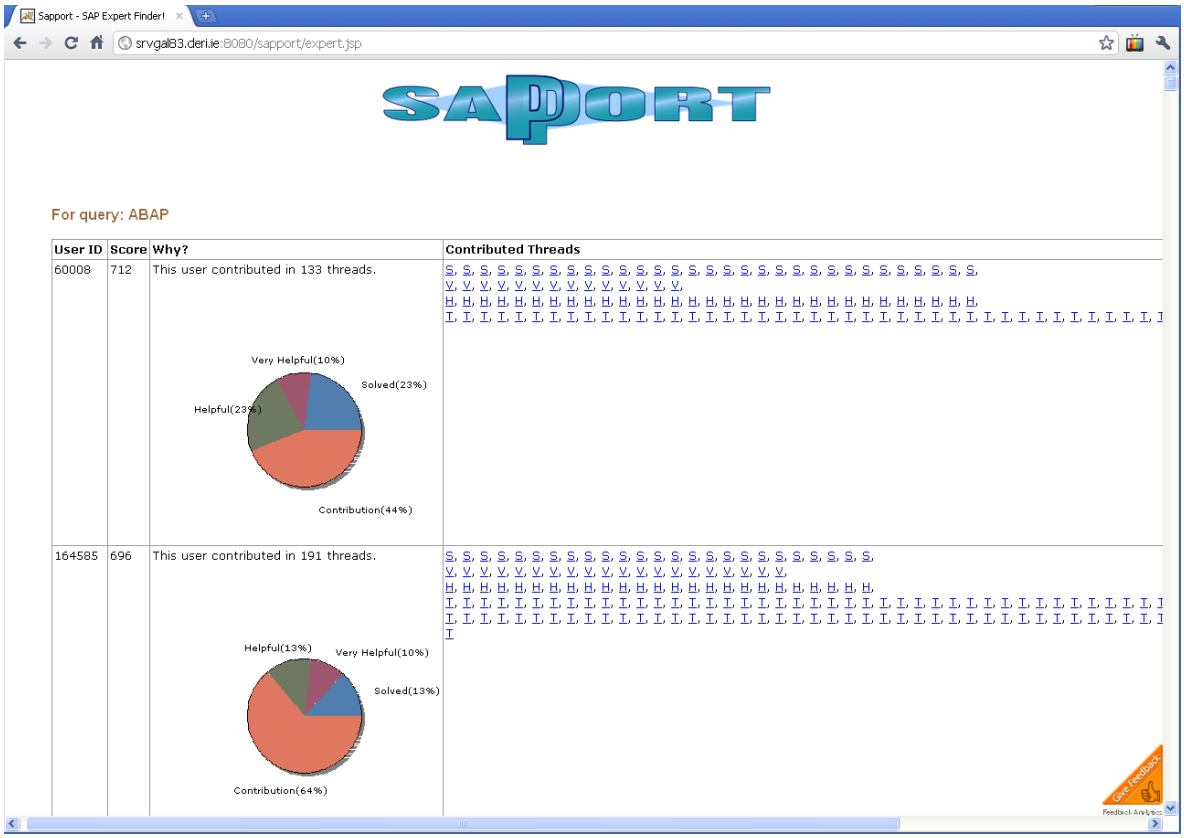

Figure 4.10: Sapport (one of the result panes): The first column lists user IDs. The second column lists their ranking in relation to the query (in this case *ABAP*). The third column explains what happened behind the scene by demonstrating a pie chart indicating various percentages: solving an issue in a thread, providing a very helpful or helpful answer, or just contributing to that thread. The last column gives explanation by providing a click-able list of thread IDs that the users contributed to. The "S" letter indicates threads where the user solved the issues in them. The "V" and "H" letters indicate threads that the user provided very helpful or helpful answers in, respectively. The "T" letter indicates other threads that the user contributed to. Sapport is also able to show a ranked list of expertise for each user, however, this result pane is not demonstrated in the above figure.

We ran the first experiment using two different methods (including fourteen different approaches) as follows:

- NLP (i.e., linear classification): The evaluation was conducted on both titles as well as first posts of the test set threads which constituted 2 approaches.
- LDA: The evaluation was conducted on both titles as well as first posts of the test set threads using various number of topics (#topics = 50, 100, 200, 300, 500 and 1000) which constituted 12 approaches.

Figure 4.11 demonstrates an overall view of the aforementioned evaluation methods.

For the second experiment, we wished to analyse more threads by expanding our test set to all threads that were published on or after 2010-October-01 (i.e., 8994 threads) and picked

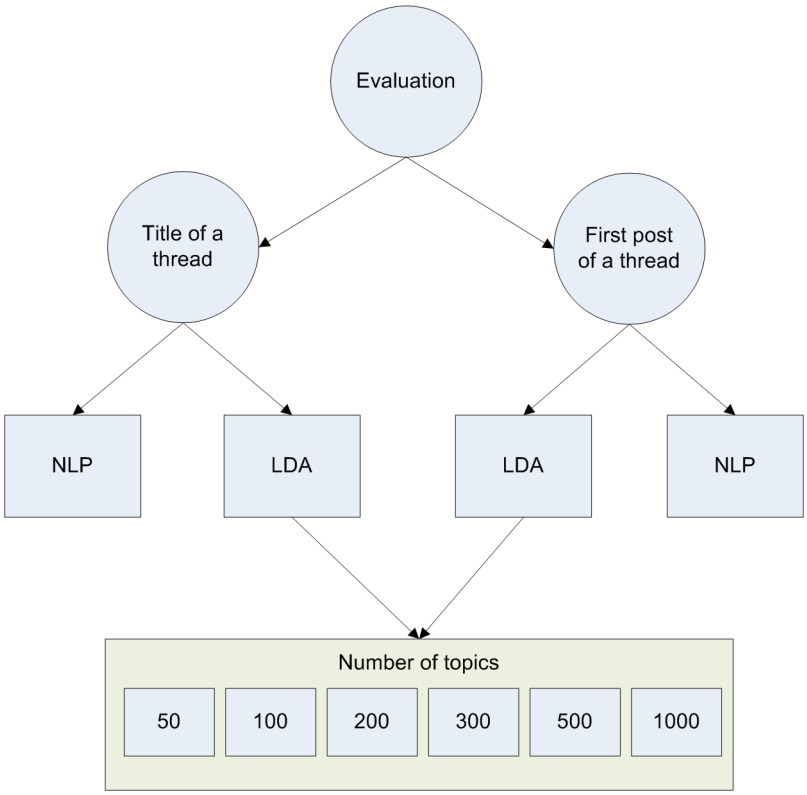

Figure 4.11: Evaluating our approach using two different methods which constituted fourteen different approaches for key phrase extraction.

those that someone provided a very helpful answer or solved the issue (i.e., 2828 threads). The main reason that we conducted the second experiment was to see how our expert finder approach would behave in two different time intervals. In other words, our goal was to check if our approach becomes more accurate (i.e., presenting relevant experts at a higher rank) with time, as more threads will be used for building expertise profiles. We used four different methods (2 methods based on NLP and 2 methods based on LDA) for the second experiment. The reason we ran this experiment for only 2 LDA methods is due to the results that we got from our first experiment, i.e., number of topics does not make a significant change in our expert finding use case. We elaborate more on this in Section 4.8.2. In the following section, we present the results of our evaluation.

4.8.2 Evaluation Result

Using a Precision-Recall curve is a standard method for evaluating the results of ranked retrieval systems. In most threads of our test set, there was only one user who provided a very helpful answer or solved the issue. This is due to the natural life cycle of normal threads in a forum. It is common that if someone solves an issue in a Q–A thread, other experts in the field do not provide similar replies and the thread will be archived. Moreover, according to SCN forums policy, awarding points (i.e., 6 or 10) to users are limited in a thread (see Section 4.4.2). Due to such behaviour, we expected to get a recall value of 1 for most threads that we can recommend the exact appropriate user. Therefore, we decided to focus on precision to see

how accurately we can provide relevant results at high ranks. To this end, we used normalised discounted cumulative gain (nDCG) [Järvelin and Kekäläinen, 2002, Croft et al., 2009].

Discounted cumulative gain (DCG) is used in Information Retrieval (IR) systems in order to measure usefulness or gain of ranked items based on their positions in the result set. The idea behind DCG is that given the same query, the result set that its relevant items appear at the top of the list is better and more useful than the result set that its relevant items appear at the bottom. Given a result set, the cumulative gain (CG) at position p is calculated as follows (Equation 4.2).

$$CG_p = \sum_{i=1}^{p} rel_i \tag{4.2}$$

The rel_i in Equation 4.2 is the relevance score of the result item at position i. Obviously, the cumulative gain does not consider the ordering of the relevant result. Thus, any changes in the ordering does not affect it. Discounted cumulative gain (DCG) is used to address this drawback. For a position p in the result set, DCG_p is calculated as follows (Equation 4.3).

$$DCG_p = rel_i + \sum_{i=1}^{p} \frac{rel_i}{\log_2 i} \tag{4.3}$$

The $\log_2 i$ in Equation 4.3 guarantees a smooth decrease in the gain for lower ranked relevant items in the result set. In order to compare various values of DCG_p, we need to normalise the values by identifying an ideal DCG at position P ($IDCG_P$) as follows (Equation 4.4).

$$nDCG_p = \frac{DCG_p}{IDCG_P} \tag{4.4}$$

In our case, an ideal position will be when the user who acquired the highest point appears at top-1 position of the result set. We compared nDCG values of the LDA and the NLP approaches for both title as well as body of the test set threads. Figures 4.12–4.15 show the result of this comparison for all test set threads (both bodies and titles respectively). The closer the nDCG values to 1, more useful the result is to the user, as it will be closer to the ideal case. For most threads of the test set, the exact relevant candidate in the result set that was suggested by the LDA approach appeared in a higher position in the list. A pairwise t-test showed that the LDA approach for both titles and bodies is better than the NLP approach. The main result is that for the titles, the LDA approach performed 16% better (p-value = 0.00), whereas for the bodies, the LDA approach performed 2% better (p-value = 0.00).

The nDCG values are good representatives for measuring the usefulness or gain of various methods. However, we also measured the positions at which users who provided very helpful replies or solved various issues (i.e., got 6 or 10 points) appeared. These detailed results are presented using top-k% notion in Appendix IV. Table 4.4 shows an overview of the results.

We also measured to what extent the candidate who provided a very helpful answer or solved an issue appeared at top-k (instead of top-k%) position in our ranked recommended results. This is important, as this does not depend on the total number of recommended experts. Consider an alerting use case scenario, in which an alert is sent to users who can potentially answer a question. In such case, it is common that with the arrival of a new question, an alert

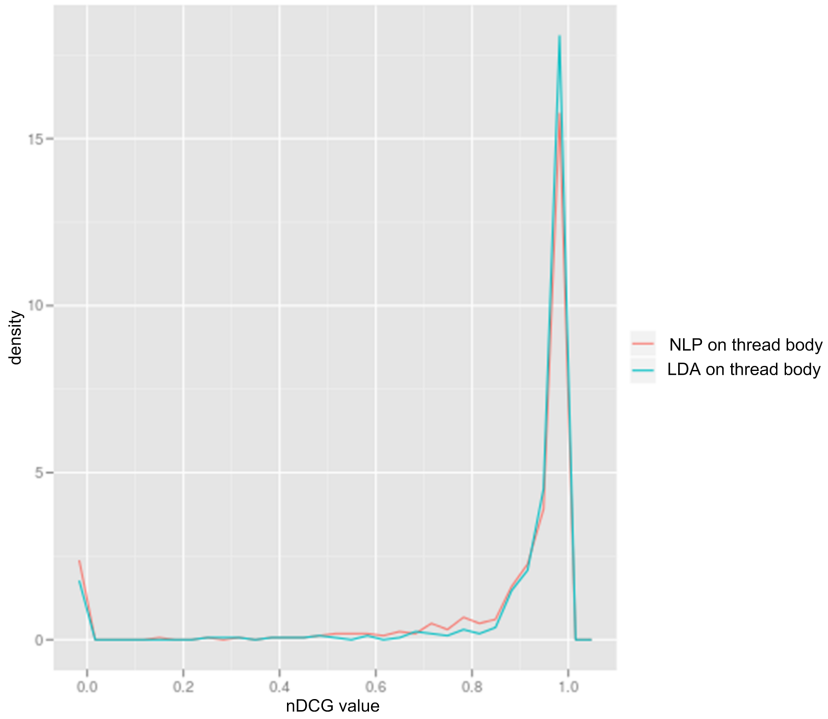

Figure 4.12: Comparison between nDCG values of test set thread bodies based on NLP and LDA. A pairwise t-test showed that the LDA approach performed 2% better than the NLP (p-value = 0.00).

will be sent to a fixed set of users such as top 10 or top 50 users and not top 10% or top 50%, as the list of recommended experts may be long and sending an alert to all the experts is nearly a flooding approach. The detailed results of top-k relevant experts are also presented in Appendix IV. An overall view of these results is presented in Table 4.5.

The results in Tables 4.4 and 4.5 also suggest that the number of topics in the LDA approach does not play a significant role in our use case. The reason is perhaps the community structure of forums, as people discuss a limited number of topics in each forum. High values for the results (i.e., top-k and top-k%) suggest that there exist several users who are extraordinarily active in the forums. As an example, we observed that a user acquired more than 50,000 points in a short time by being extraordinarily active in the forums, posting approximately 25,000 posts in about one year. His/her activities caused other users to address him/her in various threads including some threads dedicated to him/her. As stated, such behaviour also affected our statistical key phrase extractors by identifying usernames of main contributors of a thread or a forum as a potential key phrase since such usernames were often used in the corpus to address a question or comment to them.

For the second part, we repeated the experiment with a larger test set and thus a smaller training set. We used 2828 threads as test set. We chose to run LDA with n-gram=1,2,3 and 200 topics, as the result of first experiment showed that the number of topics does not play a significant role in applying LDA for our use case. We chose 200 topics as a balance between required processing time by LDA and a sufficient number of topics for building expertise profiles. Tables 4.6 and 4.7 show an overall view of the second experiment. For detailed

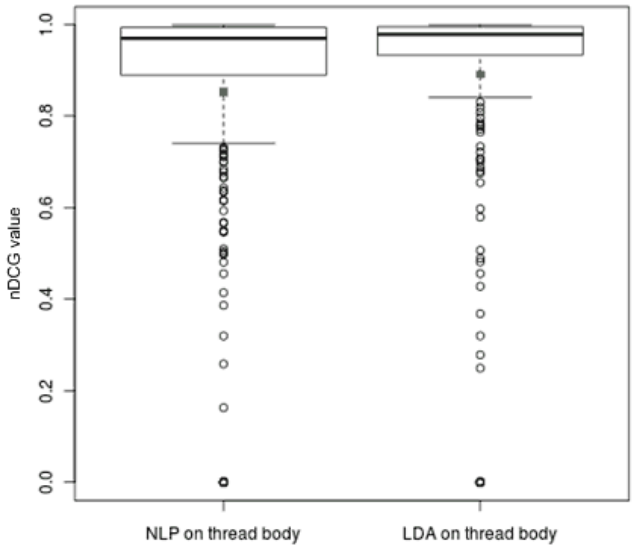

Figure 4.13: Boxplot of nDCG values of test set thread bodies based on NLP and LDA. For NLP: Interquartile range (IQR) = 0.103, for LDA: Interquartile range (IQR) = 0.061.

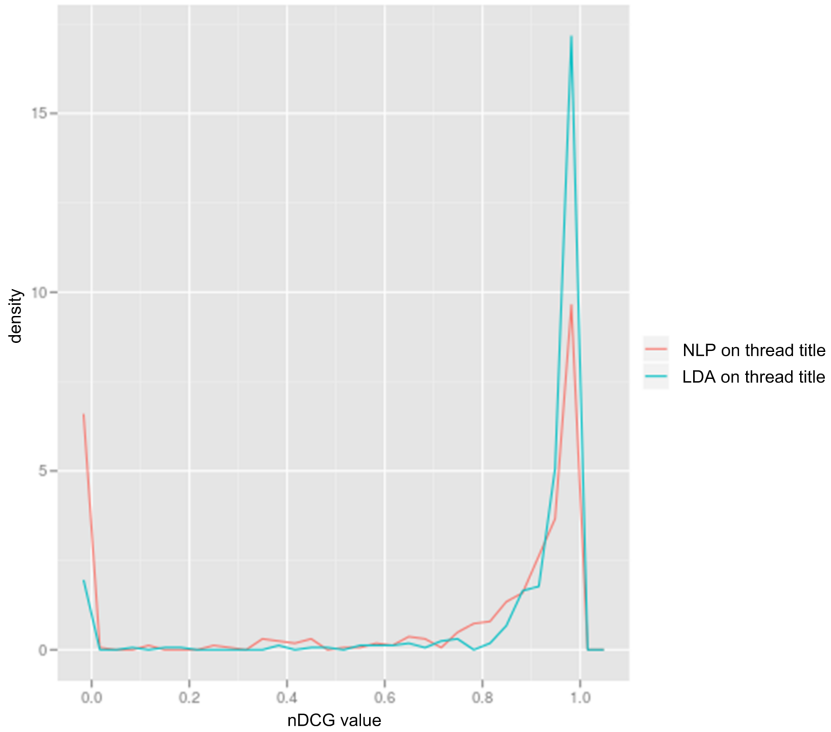

Figure 4.14: Comparison between nDCG values of test set thread titles based on NLP and LDA. A pairwise t-test showed that the LDA approach performed 16% better than the NLP (p-value = 0.00).

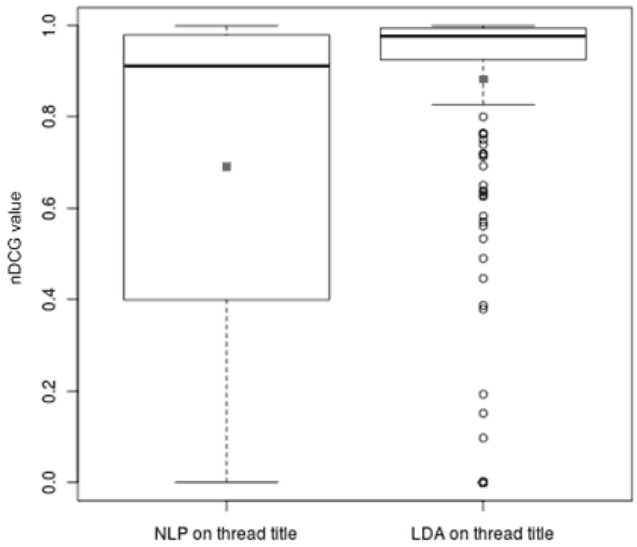

Figure 4.15: Boxplot of nDCG values of test set thread titles based on NLP and LDA. For NLP: Interquartile range (IQR) = 0.599, for LDA: Interquartile range (IQR) = 0.07.

	NP	NT	LP1	LP2	LP3	LP4	LP5	LP6	LT1	LT2	LT3	LT4	LT5	LT6
Top 1%	274	190	301	297	300	295	298	299	293	293	287	286	280	280
Top 10%	397	320	436	433	433	434	428	437	437	430	428	427	416	414
Top 100%	453	384	467	465	463	464	463	467	469	465	460	464	453	449

Table 4.4: Evaluation result of various approaches for 492 test set threads. The numbers in the table show the total number of threads that we could recommend a relevant expert (i.e., an expert who can provide a very helpful answer or solve an issue) at top-k% position. The abbreviations are as follows: NP: NLP applied to first post of a thread; NT: NLP applied to title of a thread; LP1: LDA applied to first post of a thread (#topics=50); LP2: LDA applied to first post of a thread (#topics=100); LP3: LDA applied to first post of a thread (#topics=200); LP4: LDA applied to first post of a thread (#topics=300); LP5: LDA applied to first post of a thread (#topics=500); LP6: LDA applied to first post of a thread (#topics=1000); LT1: LDA applied to title of a thread (#topics=50); LT2: LDA applied to title of a thread (#topics=100); LT3: LDA applied to title of a thread (#topics=200); LT4: LDA applied to title of a thread (#topics=300); LT5: LDA applied to title of a thread (#topics=500); LT6: LDA applied to title of a thread (#topics=1000). Other LDA parameters: n-gram=1,2,3, #keywords in each topic=20. The results show that in 95% of the cases, we could recommend the expert who solved the issue or provided a very helpful answer.

	NP	NT	LP1	LP2	LP3	LP4	LP5	LP6	LT1	LT2	LT3	LT4	LT5	LT6
Top 1	97	89	98	98	99	99	97	98	98	99	98	98	95	96
Top 5	144	133	148	147	146	148	148	149	143	143	140	142	142	139
Top 10	163	168	160	162	160	161	163	158	160	157	158	157	160	152
Top 50	272	243	265	262	264	260	263	266	258	258	256	258	259	256
Top 100	320	288	308	305	309	314	310	308	297	299	297	298	299	304

Table 4.5: Evaluation result of various approaches for 492 test set threads. The numbers in the table show the total number of threads that we could recommend a relevant expert (i.e., an expert who can provide a very helpful answer or solve an issue) at top-k position. The abbreviations are as follows: NP: NLP applied to first post of a thread; NT: NLP applied to title of a thread; LP1: LDA applied to first post of a thread (#topics=50); LP2: LDA applied to first post of a thread (#topics=100); LP3: LDA applied to first post of a thread (#topics=200); LP4: LDA applied to first post of a thread (#topics=300); LP5: LDA applied to first post of a thread (#topics=500); LP6: LDA applied to first post of a thread (#topics=1000); LT1: LDA applied to title of a thread (#topics=50); LT2: LDA applied to title of a thread (#topics=100); LT3: LDA applied to title of a thread (#topics=200); LT4: LDA applied to title of a thread (#topics=300); LT5: LDA applied to title of a thread (#topics=500); LT6: LDA applied to title of a thread (#topics=1000). Other LDA parameters: n-gram=1,2,3, #keywords in each topic=20. The results show that in 65% of the cases, we could recommend the expert who solved the issue or provided a very helpful answer at top-100 position.

results, see Appendix V.

Overall, high precision results suggest that there exist several users in enterprise Q–A forums who actively participate in replying questions. Perhaps getting more points or other incentives from forum moderators is the main motivation of such active participation. In both experiments, there existed several threads for which we could not recommend any experts (see Appendices IV and V for more details). The reason is perhaps due to the fact that both NLP and LDA approaches failed to extract suitable key phrases or there was actually no history of a relevant expert in our training data set. It is also possible that users chose non-relevant titles for their threads (in evaluating test set threads based on their titles).

Figure 4.16 shows a comparison between the two experiments. As illustrated, the results of our second experiment is slightly lower than the result of our first experiment, as the number of threads in our training set decreased.

	NP	NT	LP	LT
Top 1%	1596	981	1708	1660
Top 10%	2269	1526	2372	2322
Top 100%	2549	1886	2587	2585

Table 4.6: Evaluation result of various approaches for 2828 test set threads. The numbers in the table show the total number of threads that we could recommend a relevant expert (i.e., an expert who can provide a very helpful answer or solve an issue) at top-k% position. The abbreviations are as follows: NP: NLP applied to first post of a thread; NT: NLP applied to title of a thread; LP: LDA applied to first post of a thread; LT: LDA applied to title of a thread. LDA parameters: n-gram=1,2,3, #topics=200, #keywords in each topic=20. The results show that in 91% of the cases, we could recommend the expert who solved the issue or provided a very helpful answer.

	NP	NT	LP	LT
Top 1	562	483	555	556
Top 5	869	754	872	887
Top 10	1037	893	1019	1009
Top 50	1556	1324	1543	1532
Top 100	1763	1483	1746	1746

Table 4.7: Evaluation result of various approaches for 2828 test set threads. The numbers in the table show the total number of threads that we could recommend a relevant expert (i.e., an expert who can provide a very helpful answer or solve an issue) at top-k position. The abbreviations are as follows: NP: NLP applied to first post of a thread; NT: NLP applied to title of a thread; LP: LDA applied to first post of a thread; LT: LDA applied to title of a thread. LDA parameters: n-gram=1,2,3, #topics=200, #keywords in each topic=20. The results show that in 62% of the cases, we could recommend the expert who solved the issue or provided a very helpful answer at top-100 position.

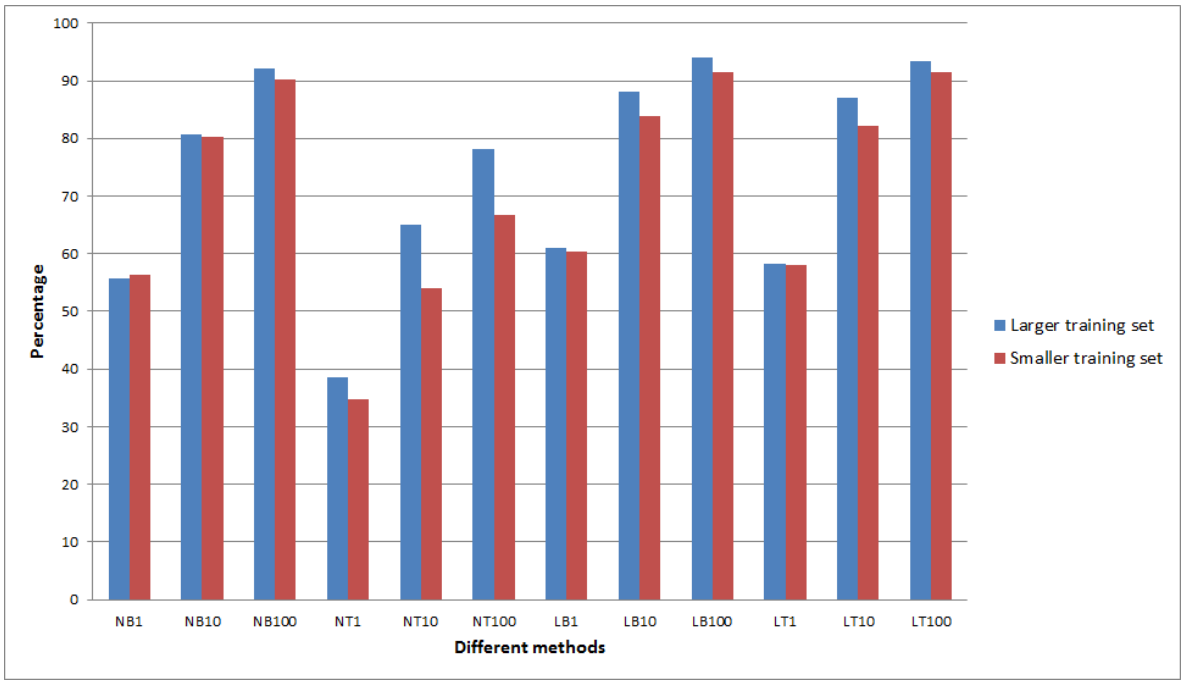

Figure 4.16: Comparison between the two experiments (i.e., smaller and larger training sets). The abbreviations are as follows: NB1, NB10 and NB100: top 1%, top 10% and top 100% result based on NLP for thread bodies; NT1, NT10 and NT100: top 1%, top 10% and top 100% result based on NLP for thread titles; LB1, LB10 and LB100: top 1%, top 10% and top 100% result based on LDA for thread bodies; LT1, LT10 and LT100: top 1%, top 10% and top 100% result based on LDA for thread titles. LDA parameters: n-gram=1,2,3, #topics=200, #keywords in each topic=20.

4.9 Conclusion and Future Work

In this chapter, we presented our approach for extracting and utilising object-centric social networks from log files of online shared workspaces. We used the extracted social network for building expertise profiles. Due to a lack of sufficient information in log files of online shared workspaces, we adapted our approach to Q–A forums, where forum points were used for ranking expertise items and thus experts.

For evaluation purposes, we divided our corpus into two different sets: a training set that we used for building expertise profiles and a test set for measuring how accurately we can recommend ranked relevant experts based on expertise profiles that we built using training set threads. In order to see how our approach behaves in two different time intervals, we ran two experiments. The first experiment used a larger training set and a smaller test set, whereas the second experiment used a smaller training set and a larger test set. The result showed that our approach becomes more accurate with time, as expertise profiles of users become more comprehensive. Various use cases can be developed on top of such expertise profile. For example, an alerting system can be developed to forward incoming questions to top-k relevant experts.

In Q–A forums, it is common that users choose a very relevant title for their questions to make it more visible to relevant experts. In other words, the title of a question is a short, yet

very relevant summary of a question. Thus, we evaluated our approach for both title as well as body of a new post (i.e., question). The result showed that yet only for the title of a new question, we could recommend relevant experts at a high rank. This implies the importance of choosing a right and accurate title for posting a question to a Q–A forum.

We tried various methods for topic modelling. In particular, we focused on the following two main methods: the NLP method which uses machine-learning algorithms to extract key phrases and the LDA method which is based on statistical analysis. The evaluation results showed that the LDA method for topic modelling performed slightly better than the NLP method that we used. In particular, the LDA method performed 16% better than NLP for the body of a new question and 2% better for the title of a new question. Despite the fact that the LDA method performs slightly better than NLP in our corpus, the feasibility of applying the LDA for real-time use cases (such as an alerting system) is questionable due to its required processing time.

There exist several improvement possibilities for finding and assigning expertise-based people-tags. We may enable end users to change the threshold of the confidence values in order to increase/decrease the scope of expertise. Introducing temporal aspects (e.g., validity period of expertise-based people-tags) is one of the possible future improvements. Clustering users based on their expertise-based people-tags is one motivating use case. In this case, we may identify groups of people that can possibly collaborate together (e.g., writing future proposals).

Chapter 5

People-Tag-Based Access Control[1]

> Everything should be made as simple as possible, but no simpler.
>
> Albert Einstein

In Collaborative Working Environments (CWE) or social media sites, where people collaborate and share online resources (e.g., profiles information, documents, events, etc.), there should exist some kind of access control mechanisms to restrict unauthorised accesses.

Despite the fact that approaches like Role-Based Access Control (RBAC) are well-adopted in many platforms, different studies such as [Hart et al., 2007, Gates, 2007] show that current access control mechanisms within Web 2.0 platforms and shared workspaces suffer from flexibility and fine-granularity. For example, users are able to share an online object with colleagues who have specific roles or belong to specific groups, but excluding several colleagues from those who are eligible to access said object can not be expressed. In other words, access control approaches that were mainly developed for desktop environments in an age when Internet and online platforms were absent, are coarse grained to be used for Web 2.0 platforms where users generate enormous amount of data everyday.

Through previous core chapters, we presented our approaches for building fine-grained user profiles using people-tags. In this chapter, we present a user-centric approach for controlling access to online personal resources that is based on people-tags. Our approach is a trade-off between usability and security in online platforms for sharing personal data.

5.1 Introduction and Motivation

In real-life, we share the resources we own based on social acquaintances or (trust) credits we grant to people, with whom we communicate. As an example, we may share the keys of our apartments with our parents, but not with our friends as we normally give more (trust) credits to our family members rather than friends. Access control emerges almost

[1] This chapter is mainly based on [Nasirifard and Peristeras, 2008a, Nasirifard and Peristeras, 2008b, Nasirifard et al., 2010b].

Website	Purpose	Relationships	Protection Options
Bebo	general	friend	public, private, 1st-degree contacts, selected contacts
Facebook	general	friend	public, private, 1st-2nd-degree contacts, selected contacts
Friendster	general	friend	members from selected continents, private, 1st-2nd-degree contacts
MySpace	general	friend	public, members > 18 years old, private, 1st-degree contacts
Multiply	general	various	public, private, 1st- and nth- degree contacts, 1st-degree but not online contacts, selected contacts
Orkut	general	friend	public, private, 1st-2nd-degree contacts
Flickr	photos	friend/family	public, private, 1st-degree contacts (friends or family)
Last.fm	music	friend	public, private, 1st-degree contacts (and profile neighbours)
Xing	business	generic	public, private, 1st-4th-degree contacts
LinkedIn	business	various	public, private, 1st- and nth- degree contacts

Table 5.1: Access control options in different online Web-based social networks – source: [Carminati et al., 2009].

together with the concept of "sharing". In brief, access control defines who can access what [Russell and Gangemi, 1991].

"Sharing" is a key concept for collaborative information spaces like Web 2.0 platforms (e.g., *flickr.com*, *youtube.com*, *delicious.com*) and CWE (e.g., Microsoft SharePoint, OrbiTeam BSCW). These platforms and applications provide the infrastructure and services for different types of users to collaborate together and share resources that may vary from songs and photos to documents and calendars. In these Web-based environments of massive-scale sharing, access control takes interesting characteristics due to additional requirements such as having a more fine-grained control over personal data. Such shortcomings undermine the utility of online shared workspaces and bring privacy-related issues in Web 2.0 platforms. As a result, users sometimes use email and instant messaging to share resources such as documents with each other that usually brings overhead as well as version controlling problems for users.

Table 5.1 (source: [Carminati et al., 2009]) summarises access control mechanisms of different online social networking sites. Most current online social networking sites and CWEs provide so-called *friends* model [Hart et al., 2007]. This model enables users to make a list of friends and restrict their resources to be visible only to them. Such models are easy-to-use [Hart et al., 2007], however, they are not flexible enough to express fine-grained information sharing policies. In the following section, we present a scenario in which its requirements can not be expressed with current access control models within social networking sites and online shared workspaces.

5.1.1 Scenario

We present a simple scenario to challenge functionalities of current access control models which are used in social and collaborative platforms.

Bob is the name of the main actor. He is currently working on a European project in a collaborative distributed infrastructure with other team members from different organisations. Partners are geographically distributed in different countries with various time zones. This project has different Work Packages (WP) and Bob is the leader of WP two. The project has a website for public visitors. The website includes project news, newsletters, public events, public deliverables and information about the scope and the mission of the project. The project has also a private collaborative working environment (shared workspace). The private side includes a wiki, a forum, a calendar to document events, some folders for uploading documents to be accessed by team members, a bunch of documents, presentations, photos from meetings, contracts, time sheets etc. In this private workspace, Bob has uploaded several documents, photos, and presentations. The issue is that all project members should not have access to Bob's resources. In our case, Bob wants to set the following access control rules based on the roles defined by the project.

- Bob wants to give access of work-in-progress deliverables to all WP leaders including project coordinator. In case some of them are not available (e.g., on vacation), this access should be given to their proxies who have the authority to access such documents.
- Bob wants to give access of a confidential contract only *once* and only to a specific person.
- Bob wants to give access of a particular presentation only during the meeting (temporal restrictions) and only to specific meeting participants.
- Bob wants to give access of a particular background document only to members that are currently working on a particular deliverable.
- Bob wants to share a photo only to his close friends.
- Bob wants to give access of his presentation, only after finishing it and only to particular members.
- Bob does not want to give access to a document to the friends that were not present in a particular meeting and whose trust levels are less than a threshold.
- Bob wants to share a technical report with responsible people from other projects who are related to his project (i.e., same domain)

The above items can be seen as user-centric requirements for setting access control policies. In general, with current *friends*-based and/or role-based access control mechanisms within most social and collaborative platforms, it seems to be very difficult or even impossible to apply above rules. The lack of a fine-grained access control mechanism for online shared workspaces within collaborative working environments is one of the main use cases that we want to address in this chapter. The term *fine-grained* can be perceived as a flexible, parametric, context-aware, open and extensible access control mechanism.

In this chapter, we propose a people-tag-based model[2] to address access control requirements in social and collaborative platforms and implement our approach using Semantic Web technologies. Our model is conceptually based on how we grant access to the resources we own in real-life. Through previous core chapters, we presented approaches to assist users to tag each other in order to build a comprehensive and fine-grained profile for themselves and also for their contacts. In this chapter, we show how we use such profiles for controlling information flow within enterprises.

In order to make the semantics of our model machine-understandable, we express them in a formal way using semantic technologies. We are using the Resource Description Framework (RDF) data model to express relationships amongst actors of a social network. For more information on RDF data model, see Chapter 2.

The remainder of this chapter is structured as follows: In Section 5.2, we discuss related work. In Section 5.3, we present detailed specification of Annotation-Based Access Control (AnBAC) model which consists of necessary definitions and rules of the model. Note that implementation and privacy guidelines are presented in Appendix VI. We present formal description of our model in Section 5.4. In Section 5.5, we present a use case scenario that consists of a comprehensive example on how our model operates based on people-tags. As AnBAC model is a very comprehensive model composing of many components and features, in Section 5.6, we present a simplified version of the AnBAC model for users who have minor requirements. In Section 5.7, we present our prototype called Uncle-Share that realises the AnBAC model. Our experimental evaluation together with comparing our approach with state-of-the-art will be presented in Section 5.8 and finally in Section 5.9, we conclude and have an overview of future work.

5.2 Related Work

There exist plenty of approaches and mechanisms for controlling access to resources, such as access control lists, role-based access control, attribute-based access control, ontology-based access control and so on. Each approach has its own advantages, disadvantages and feasibility scope. In this section, we briefly introduce current relevant approaches and after explaining our model in Section 5.3, we compare these approaches with ours.

Many researchers try to combine different mechanisms of access control in order to build a more powerful mechanism and decrease the disadvantages of each mechanism. [Kern and Walhorn, 2005] present an architecture for role-based access control to use different rules to extract dynamic roles. [Alotaiby and Chen, 2004] present a team-based access control that is built on top of role-based access control. [Periorellis and Parastatidis, 2005] introduce another extension to role-based access control that is called task-based access control. They discuss task-based access control as a mechanism for dynamic virtual organisation scenarios. Collaborative role-based access control (C-RBAC) for distributed systems is discussed by [Kim et al., 2005]. The C-RBAC model tries to address the conflicts from cross-domain role-to-role translation. Our model can be also seen as an extension to RBAC, where roles are defined in a user-centric manner.

[2]Through this thesis, we use *Annotation-Based Access Control (AnBAC)* to refer to our access control model that is based on people-tags.

Efforts are also being made to enrich access control mechanisms by means of Semantic Web technologies. To this end, a rule-based access control which is based on OWL and SWRL[3] is presented in [Li et al., 2005]. They propose an OWL ontology to describe terms and SWRL to express access policy rules. [Priebe et al., 2006] discuss that attribute-based access control (ABAC) is a bit complex and error-prone and they propose a solution by pushing Semantic Web technologies and ontologies into ABAC. We also learned valuable lessons from above approaches and decided to use semantic data models in our work.

As our model suits mainly in collaborative and social environments, in the following paragraphs, we take a look at current access control models within such environments. The study of access control mechanisms in cooperative systems is not new and was in existence along with work on e-Collaboration. Basic requirements for role-based access control within collaborative systems are presented in [Jaeger and Prakash, 1996]. [Shen and Dewan, 1992] studied access control mechanisms in a simple collaborative environment, i.e., a simple collaborative text editing environment. [Tolone et al., 2005] provide a comprehensive study on access control mechanisms in collaborative systems and they compare different mechanisms based on various criteria, e.g., complexity, understandability, ease of use, etc. [Demchenko et al., 2006] propose an access control model and mechanism for grid-based collaborative applications.

As social networks provide the necessary infrastructure for sharing various items, the need for a more flexible access control in such platforms is more evident. Recent research and news proved the importance of privacy and security in online social networks [Fogel and Nehmad, 2009, Irvine, 2008, Young, 2008]. Some researchers like [Gates, 2007, Hart et al., 2007] argue that current mechanisms of access control within Web 2.0 and social sites are not fine-grained enough. Most of the literature related to access control within social platforms focuses on relationships that people may acquire in a social network. [Kruk et al., 2006] suggest a role-based and policy-based access control for social networks, where the access rights will be determined based on social links and trust levels between people. [Carminati et al., 2006b] present a similar approach and in [Carminati et al., 2007, Carminati et al., 2009], they extend and provide a detailed explanation of their model. [Gates, 2007] proposes Relationship-Based Access Control (ReBAC) to tackle privacy concerns within social platforms. The technical details of ReBAC is not published, however, [Giunchiglia et al., 2008] present and define a model, called RelBAC which is based on relationships between people. Prior to [Gates, 2007], [Barkley et al., 1999] proposed to utilise relationships in Role-Based Access Control. [Shehab et al., 2008] present an access control framework for social networks API to restrict unauthorised accesses to profile information. This is mainly considered to be utilised for developers. Lockr, an access control framework for Web 2.0 applications using social access control lists, is presented in [Tootoonchian et al., 2008]. [Hong and Shen, 2008] benefit from transitive relationships to define access control permissions in Web-based social networks. [Razavi and Iverson, 2007, Razavi and Iverson, 2009] studied possible user-oriented privacy solutions and candidate access control models for social software and they discuss tagging people as a new paradigm for access control. They also developed a platform called OpnTag to enable users to use this access control model. Using social network analysis metrics (e.g., betweenness centrality) in access control is proposed by [Mori et al., 2005]. Our approach has certain differences and offers some advantages compared to these approaches, which will be discussed in Section 5.8.

[3] http://www.w3.org/Submission/SWRL/

5.3 Annotation-Based Access Control

Annotation is a common mechanism that is nowadays used by social platforms for annotating shared informational resources and is based on mechanisms that allow users to describe resources with "tags". In this way, users attach metadata to commonly shared resources (social tagging). As explained in Chapter 2, these tags later facilitate browsing and discovery of relevant resources. Annotations and tags are important mechanisms of what has been called Web 2.0 or Social Web.

Our access control model is based on annotations (and in particular people-tags) too. In our approach, end users are able to annotate their contacts (social network) and resources (e.g., bookmarks, photos, documents) and then define policies for granting access to their resources based on these annotations. Only contacts that fulfill the required policies get access to specific resources. A simple example follows: User A owns several resources (e.g., documents) which have been tagged as *ResearchPaper*. The term *owns* can be defined as having a resource (such as a document) in a home folder or bookmarking a resource (such as a URL) into a Web browser. User A annotates user B which is part of his/her social network as *CollaborateWith*. User A defines an access control policy to share the resources that have been tagged as *ResearchPaper* (by himself/herself) with his/her contacts that have been tagged as *CollaborateWith* (by himself/herself). In this case, all appropriate resources will be automatically accessible to user B, as user B meets the policy requirements. Annotation-based access control is very close to how we share resources in real-life. We may share our credit card details with our parents, but not with our friends. Based on this simple scenario, in annotation-based access control, both our parents and friends are parts of our social network, but our parents have been tagged as *parent* and our friends have been tagged as *friend* and our credit card details are resources with a policy to be shared with only *parent*.

Figure 5.1 demonstrates the elements and relationships of Annotation-Based Access Control (AnBAC) model. The simplified version of AnBAC is presented in Section 5.6. Here we present an advanced model. The advanced AnBAC model comprises three main entities: *Person*, *Resource*, and *Policy*; and four main concepts: *Annotation*, *Distance*, *Context*, and *Action*.

A Person is an entity with the RDF[4] type Person. A Person is connected unidirectionally to zero or more other Person(s) and the source Person can tag the target Person with zero or more Annotation(s). A Person owns zero or more Resource(s). A Person defines zero or more Policy(ies). A Person may, i.e., it is conditional, be assigned (hasBeenAssigned) one or more Policy(ies) by other Person(s). A Person has zero or more Context(s) that aim to describe the context information of that Person.

A Resource is an entity with the RDF type Resource and is owned by (isOwnedBy) one Person. Note that a single Resource can be conceptually owned by more than one Person. A Resource is online material, for example, a photo or bookmark, although, the model can be applied to an offline (e.g., key) Resource as well. A Resource owner can assign zero or more Annotation(s) to a Resource. A Resource may be assigned (hasBeenAssigned) one or more Policy(ies) by Resource owner. A Resource can conceptually be either private or public. A private Resource has either zero Policy(ies) or the Policy(ies) that are never matched with

[4]RDF is simply the data model that we used in this work. RDF can be simply replaced with other data models.

any specified Annotation(s), Distance(s), and/or Context(s); whereas a public Resource has one or more Policy(ies) which will be met (matched) at some point. A Resource has zero or more Context(s) that aim to describe the context information of that Resource.

A Policy is an entity with the RDF type Policy. A Policy is defined by (isDefinedBy) one Person. A Policy may be assigned to (isAssignedTo) one or more Resource(s). A Policy may be assigned to (isAssignedTo) one or more Person(s). A Policy has at least one and at most two Annotation(s). A Policy has one Distance. A Policy has at least one Action. A Policy has zero or more Context(s) that aim to describe the context information of a Person, Resource or that Policy.

An Annotation is a term or set of terms that are connected together and aims to describe the Person(s) or the Resource(s) that the Resource(s) should be shared with. An Annotation can be (hasType) either Explicit or Implicit. An Explicit Annotation is a fixed term or set of terms that are assigned by a Person to a Person, Resource or for defining a Policy. An Implicit Annotation may be a term or set of terms. An Implicit Annotation is calculated and/or assigned during run-time. As an example, an Implicit Annotation may be the output of a Web service which is invoked at run-time. An Annotation can have zero or more Attribute(s). An Attribute has one Value and is a criterion to "tune" the Annotation (e.g., friend is an Annotation which can be assigned with an Attribute like reliability and a Value like very high). The Value of an Attribute can be assigned statically (Explicit) or dynamically (Implicit). For example, an Implicit Value may be the result of a Web service which is invoked at run-time.

A Distance is a numerical value which determines the depth or extent across a network of connected entities for which the Policy is valid. The depth is length of the shortest path among two Person(s) with consideration of Annotation(s) between Person(s) connected in a graph-like shape.

A Context represents the context information of an entity. It is difficult to give one valid global definition to the context. The main reason is that there is no absolute context and context gets its meaning in relation to something [Bazire and Brézillon, 2005]. Defining context information is domain/implementation dependant. Theoretically, context information of an entity can contain "anything" regarding that entity (e.g., status, profile). In our model, Context is either Explicit or Implicit. An Explicit Context refers to that context information which is assigned to or manually specified at any time. An Implicit Context is calculated and/or assigned during run-time. As an example, an Implicit Context may be the output of a Web service which is invoked at run-time. A Person can assign an Explicit Context to himself/herself (e.g., travelling). A Resource can have context information (e.g., size, modification date, source). A Policy can also have context information (e.g., Context of a Policy may determine the validity period of that Policy; or a Policy may be enabled/disabled by setting an Explicit Context of that Policy to True/False). Note that Context of a Person is set by that Person (or can be fetched considering the status of that Person); Context of a Resource is set by the Resource owner (or can be fetched considering the status of that Resource); and Context of a Policy is set by the Person that defines the Policy.

An Action is an event (function) that can happen on a Resource (e.g., Read, Revise, Copy, Delete, Synchronise, Archive, etc.)

For the purposes of the present specification, there are several definitions:

- Definition 1: A policy is called an Explicit Policy, if all of its Annotation(s) and Context(s) are Explicit.

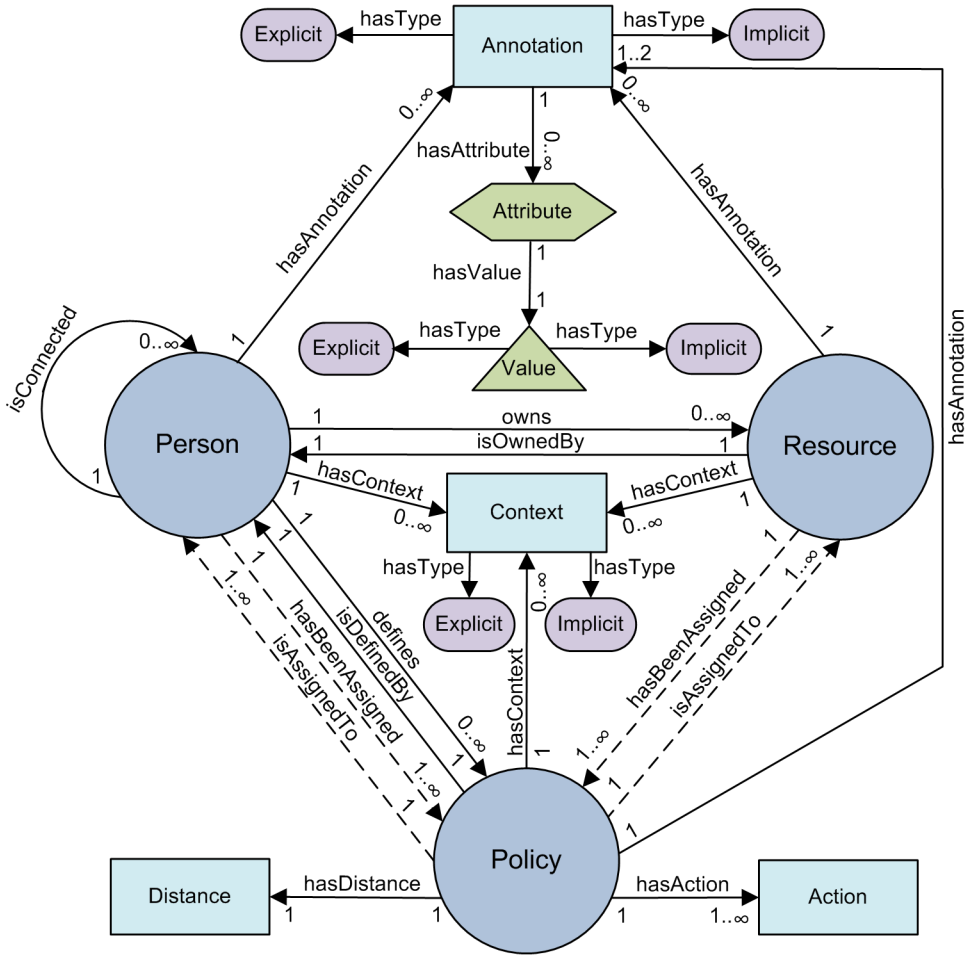

Figure 5.1: Main elements of the Annotation-Based Access Control (AnBAC) model and their relationships. Dashed arrows demonstrate conditional links.

- Definition 2: A policy is called an Implicit Policy, if all of its Annotation(s) and Context(s) are Implicit.
- Definition 3: The Annotation used in a Policy is called a Person Annotation, if it refers to Person(s).
- Definition 4: The Annotation used in a Policy is called a Resource Annotation, if it refers to Resource(s).
- Definition 5: A Context is called a Person Context, if it describes the context information of a Person.
- Definition 6: A Context is called a Resource Context, if it describes the context information of a Resource.
- Definition 7: A Context is called a Policy Context, if it describes the context information of a Policy.

There exist several rules (meta-policies) which are useful for understanding this model:

- Rule 1: A Person acquires access to a Resource, if and only if (*iff*) s/he meets *all* or *part* of the Policy(ies) that have been defined by the Resource owner for that Resource. Note that logical connectives can be used to specify *all* or *part* of a Policy that should be validated.

- Rule 2: Only the Resource owner is eligible to define Policy(ies) for the Resource(s) that s/he owns.

- Rule 3: If a Policy has one Annotation and if that Annotation is a Person Annotation, then the Policy must be assigned to (isAssignedTo) one or more Resource(s). In this case, those Resource(s) are assigned (hasBeenAssigned) to that specified Policy.

- Rule 4: If a Policy has one Annotation and if that Annotation is a Resource Annotation, then the Policy must be assigned to (isAssignedTo) one or more Person(s). In this case, those Person(s) are assigned (hasBeenAssigned) to that specified Policy.

- Rule 5: Distance belongs to Person Annotation.

- Rule 6: Distance is calculated taking Annotation(s) into account. Annotation(s) are used to build a graph among people which may contain several paths between two Person(s) and preferably all paths are considered when determining how to reach a target Person from a source Person. For example, if Person A is connected to Person B and has annotated Person B with *student*, the Distance from Person A to B (unidirectional) with the consideration of *student* is one. The Distance from Person A to B (unidirectional) with the consideration of any other Annotation (e.g., friend) is infinity. The Distance from Person B to A (unidirectional) is also infinity, if Person B has not defined an outgoing link to Person A.

- Rule 7: Distance should be considered/calculated/enabled for only commonly-agreed Annotation(s) with commonly-agreed meanings.

- Rule 8: Each Person, Resource, and Policy should be uniquely identifiable.

- Rule 9: A Person may assign Annotation(s) to other Person(s), but s/he may define Context for himself/herself (i.e., self-Annotation).

- Rule 10: The Value of an Attribute needs to be based on a pre-defined scale (e.g., "1" to "10" or "very low" to "very high").

- Rule 11: A Resource/Policy/Person may be removed or edited by the owner. Removing/editing an entity affects also all belonging components.

- Rule 12: Policy(ies) may complement each other, but may not be conflicting.

Implementation and privacy guidelines are presented in Appendix VI.

The model becomes clearer with a use case scenario, which is presented in Section 5.5. In the following section, we formally define the annotation-based access control model.

5.4 Formal Representation

The AnBAC model can be represented with $< S(subject), O(object), A(action) >: -condition$, where a subject can acquire access (of type action) to an object, if the pre-defined conditions are satisfied. As stated in Chapter 2, tagging objects by users create folksonomies. A folksonomy F can be defined as a tuple $F := (U, T, R, Y)$, where U, T and R are finite sets of users ($U = \{u_1, u_2, ..., u_i\}$), tags ($T = \{t_1, t_2, ..., t_j\}$) and resources ($R = \{r_1, r_2, ..., r_k\}$), whereas Y is a ternary relation between them ($Y \subseteq U \times T \times R$) [Hotho et al., 2006b]. The folksonomy of people-tagging can be defined using a tuple $F' := (U, T', Y')$, where U and T' are finite set of users and tags; whereas Y' is a relation between them, such that $Y' \subseteq U \times U \times T'$.

In the AnBAC model, a user u_i can be referred to using a Universal Resource Identifier (URI). A resource r_k can be any online digital material. A resource can be also referred to using an URI. Besides URI, a resource r_k has a set of objective values $\{r_{k1}, r_{k2}, ..., r_{km}\}$ which are associated to a set of objective attributes $\{o_1, o_2, ..., o_m\}$ and a set of subjective tags defined by user u_i. Objective attributes are those ones that describe the characteristics of a resource. For example MIME-Type, size, and language of a document are objective attributes of that document. Subjective tags are those ones that are related to personal view of a user on that resource.

In the AnBAC model, a user u_i can define an access control policy p_j for sharing resources. We denote this using $\Psi(u_i, p_j)$, meaning that u_i defines p_j. The policy p_j can be represented with (t, t', d, a), where $t \in T$, $t' \in T'$, $d = \Delta(u_i, u_j, t') \in Positive\ Integer$ and $a \in A = \{a_1, a_2, ..., a_m\}$. In natural language, $\Psi(u_i, (t, t', d, a))$ defines that action a to resources that have been tagged with t is allowed to people that have been tagged with t' and have a maximum distance of d to u_i. Note that in the policy p_j, each of t, t', d, a can be replaced with the $*$ wildcard. We define $\Delta(u_i, u_j, t')$ as the distance between u_i and u_j, where the shortest path is considered; Distance is calculated taking into account the annotation values. In other words, $\Delta(u_i, u_j, t')$ is n, if and only if (iff) the shortest sequence of $\underbrace{\{(u_i, u_x, t') \in Y' \wedge (u_x, u_y, t') \in Y' \wedge ... \wedge (u_z, u_j, t') \in Y'\}}_{n}$ has n members. For example, $\Delta(u_i, u_j, t') = 1$, iff $\{(u_i, u_j, t') \in Y'\}$. Note that $\Delta(u_i, u_j, t')$ is ∞, if there is no such sequences. The set A is a finite set of actions that can happen on a resource (e.g., Read, Write).

Note that $\Psi(u_i, p_j)$ can be conceptually associated to a single resource or a single user. The associated policy to a single resource r_k is represented as $p_j = (r_k, t', d, a)$ and the associated policy to a single user u_j is represented as $p'_j = (t, u_j, a)$. Obviously, in the latter case the concept of distance is undefined as the policy is conceptually attached to a single person. We define $\Psi'(u_i, p'_j)$ for the latter case. A policy that links a single user to a single resource is trivial: $p'_j = (r_k, u_j, a)$.

As stated, a policy can be also represented with state-of-the-art policy languages, like PROTUNE policy framework [Coi et al., 2008]. For example, a policy like Ψ(me, ('thesis-topic', 'student', 2, READ)) can be represented with the following PROTUNE syntax: allow(access(Resource, Requester, READ)) $\longleftarrow$ linked(Resource, tag, 'thesis-topic'), linked(Requester, tag, 'student'), distance(me, Requester, 2).

In order to realise this model, we propose the following helper functions to be implemented:

- $Tags(u_i, r_k)$ returns the tags that u_i has assigned to r_k and is defined as $\{t \mid (u_i, r_k, t) \in$

$Y\}$. Respectively, $Tags(*, r_k)$ returns the tags that all users have assigned to r_k.

- $Tags(r_k)$ returns the associated metadata to resource r_k (e.g., MIME-type)
- $Tags(u_i, u_j)$ returns the tags that u_i has assigned to u_j and is defined as $\{t' \mid (u_i, u_j, t') \in Y'\}$. Respectively, $Tags(*, u_j)$ returns the tags that all users have assigned to u_j. Note that $Tags(u_i, u_i)$ is trivial.
- $Network(u_i)$ returns the (annotated) social network of u_i and is defined as $\{(u_j, t') \mid (u_i, u_j, t') \in Y' \wedge t' \in T' \cup \emptyset\}$. Respectively, $Network(u_i, t')$ returns the social network of u_i that has been tagged with t'. The definition of the latter is trivial.
- $Network'(u_j)$ returns the users who have (annotated) connections to u_j and is defined as $\{(u_i, t') \mid (u_i, u_j, t') \in Y' \wedge t' \in T' \cup \emptyset\}$. Respectively, $Network'(u_j, t')$ returns the users who have annotated u_j with t'. The definition of the latter is trivial.
- $Resources(u_i)$ returns the (annotated) resources of u_i and is defined as $\{(r_k, t) \mid (u_i, r_k, t) \in Y \wedge t \in T \cup \emptyset\}$. Respectively, $Resources(u_i, t)$ returns the associated resources to u_i that were tagged with t (by u_i). The definition of the latter is trivial.
- $Users(\Psi(u_i, p_j)) = Users(\Psi(u_i, (t, t', d, a)))$ returns eligible users that have an access of type a to associated resources of u_i, based on p_j and is defined as $\{(u_j, r_k, a) \mid Tags(u_j) \subseteq t' \wedge Tags(r_k) \subseteq t \wedge \Delta(u_i, u_j, t') \leq d)\}$. Note that $Users(\Psi'(u_i, p'_j))$ is trivial.

In the next section, we study a use case scenario.

5.5 Use Case Scenario

In order to clarify the concepts and make our model more understandable, we present a scenario. For a simplified scenario based on the basic AnBAC, see [Nasirifard and Peristeras, 2008a].

Referring now to Figure 5.2, where return arrows are not shown in the figure for simplicity, several users are connected together. In our scenario, the default distance is `1` and the default action is `Read`. We use the policy representation presented as an implementation guideline in Appendix VI, e.g., we use a semicolon (;) for separating annotations and actions in the policies, and a colon (:) for separating person annotation and distance.

Users perform the following actions. Alice adds the following people to her social network:

- She adds Bob and annotates him with two explicit annotations: *collaborateWith* and *doResearchWith*. We suppose these two explicit annotations originate from a commonly-agreed vocabulary with commonly-agreed meaning.
- She adds Mary to her contacts and annotates her with one explicit annotation: *boardOfDirectors*.
- She adds John to her contacts and annotates him with one explicit annotation: *boardOfDirectors*.

- She adds Paul to her contacts and annotates him with one explicit annotation: *friend*. (Note 1: For tuning purposes, Alice also defines an attribute called "experts in" for this annotation. From the technical point of view, she points out to a Web service that returns expertise of a user, as an implicit value for this attribute.)
- She adds Ed to her contacts and annotates him with one explicit annotation: *friend*. (Note 2: In order to tune this annotation, Alice defines an attribute called "experts in" for this annotation as well. She defines an implicit value for this attribute which points out to a Web service.)

Alice owns the following resources:

- A bookmark: *www.resource1.com* with one explicit annotation: *mustSee*.
- A bookmark: *www.resource2.com* with two explicit annotations: *mustSee* and *interesting*.
- A short message: *I_need_to_talk_to_you_please*.

Alice defines the following policies:

policy1: (mustSee); (collaborateWith:2); Read;. This policy gives Read access of all resources that have been annotated as *mustSee*, to people that have a maximum distance of two (i.e., contacts' contacts) to Alice, and have been annotated as *collaborateWith* (i.e., a term which originated from a vocabulary).

policy2: (mustSee); (doResearchWith:2); Read;. This policy gives Read access of all resources that have been annotated as *mustSee*, to people that have a maximum distance of two to Alice, and have been annotated as *doResearchWith*.

policy3: ;(boardOfDirectors:1):online; Read; for *I_need_to_talk_to_you_please* resource. This policy has no "Resource Annotation". Thus, it needs to be attached to a single or multiple resources. This policy means that those direct contacts of Alice that have been annotated as *boardOfDirectors* and their context information denoting that they are *online*, have Read access to *I_need_to_talk_to_you_please*.

Policy4: (interesting); (friend:1:[experts in: social networks]); Read;. This policy gives Read access to all resources that have been annotated as *interesting*, to direct *friend*(s) of Alice who are "expert(s) in" "social networks".

Alice also defines that, in order to access a resource, a person should meet all policies that have been conceptually assigned to that resource.

Bob adds the following people to his social network:

- He adds Alice and annotates her with one explicit annotation: *student*. We suppose this explicit annotation comes from a commonly-agreed vocabulary with commonly-agreed meaning.
- He adds Tom and annotates him with two explicit annotations: *collaborateWith* and *doResearchWith*. We assume that these two explicit annotations originate also from a commonly-agreed vocabulary with commonly-agreed meaning.

- He adds Ben to his contacts and annotates him with one explicit annotation: *brother*. We assume that this explicit annotation comes from a commonly-agreed vocabulary with commonly-agreed meaning.
- He adds Anna to his contacts and annotates her with one explicit annotation: *mother*. We assume that this explicit annotation comes from a commonly-agreed vocabulary with commonly-agreed meaning.
- He adds Phil to his contacts and annotates him with one explicit annotation: *student*. As stated, this annotation comes from a commonly-agreed vocabulary with commonly-agreed meaning.
- He adds Paul to his contacts and annotates him with one explicit annotation: *colleague*. (Note 3: In order to tune this annotation, Bob defines an attribute called "collaboration level" for that. He also sets an explicit value for this attribute: "high".)
- He adds Lisa to his contacts and annotates her with one explicit annotation: *colleague*. (Note 4: Bob also defines an attribute called "collaboration level" for this annotation. He also sets an explicit value for this attribute: "low".)
- He adds Adam to his contacts and does not annotate him.

Bob owns the following resources:

- A document: *file1.doc* with one explicit annotation: *research*.
- A document: *file2.ppt* with one explicit annotation: *student*.
- A document: *file3.doc* with one explicit annotation: *proposal*.
- A document: *file4.pdf* with one explicit annotation: *proposal*.
- An image: *image.jpg* with one explicit annotation: *private*.

Bob defines the following policies for his resources and sets that in order to access a resource a person should meet all policies conceptually belonging to the resource.

Policy5: (research); (doResearchWith:1); Revise; date:10.01.2009-15.01.2009. This policy gives Revise access to all resources that have been annotated as *research*, to people that have a maximum distance of one to Bob (i.e., direct connection), and have been annotated as *doResearchWith*. This policy has a validity period (i.e., Policy Context). The validity period is from 10.01.2009 to 15.01.2009.

Policy6: (research); (collaborateWith:1); Revise; date:10.01.2009-15.01.2009. This policy gives Revise access to all resources that have been annotated as *research*, to people that have a maximum distance of one to Bob (i.e., direct connection), and have been annotated as *collaborateWith*. This policy has a validity period (i.e., Policy Context). The validity period is from 10.01.2009 to 15.01.2009.

Policy7: (student); (student); Revise;. This policy gives Revise access to all resources that have been annotated as *student*, to people that have a maximum distance of one (i.e., default distance) to Bob (i.e., direct connection), and have been annotated as *student*.

Policy8: (student); (student:2); Read;. This policy gives Read access to all resources that have been annotated as *student* to people that have a maximum distance of two (i.e., contacts' contacts) to Bob, and have been annotated as *student*.

Policy9: (proposal): fileType: doc; (colleague:1:[collaboration level:high]); Revise;. This policy gives Revise access to all resources that have been annotated as *proposal* and have "doc" fileType (i.e., Resource Context), to people that have a maximum distance of one to Bob (i.e., direct connection) and have been annotated as *colleague*, and their "collaboration level" (i.e., attribute) is also "high" (i.e., value).

Policy10: (private); (FAMILY); Copy;. This policy gives Copy access to all resources that have been annotated as *private*, to people that have a maximum distance of one to Bob (i.e., direct connection), and have been annotated with an implicit annotation: *FAMILY*. We assume that this annotation comes from a commonly-agreed vocabulary with commonly-agreed meaning that models the members of a "family". In other words, the members of a family (e.g., brother, mother) will be considered at run time. The reason that Bob capitalised this annotation is to emphasise the implicitness of the annotation.

Policy11: (private); (FAMILY); Synchronise;. This policy gives Synchronise access to all resources that have been annotated as *private*, to people that have a maximum distance of one to Bob (i.e., direct connection), and have been annotated implicitly with *FAMILY* which originates from a commonly-agreed vocabulary with commonly-agreed meaning and should be matched at run-time.

Policy12: (student);; Read;. This policy gives Read access of the resources that have been annotated as *student*, to specified person(s). This policy has no "Person Annotation" and is conceptually attached (isAssignedTo) to one person (i.e., Adam).

Tom adds the following people to his contacts:

- He adds Phil to his contacts and annotates him with one implicit annotation. (Note 5: In order to set an implicit annotation, Tom defines the following rule: If Phil has updated his blog in the past one month, then he is annotated as *activeBlogger*, otherwise *inactiveBlogger*.)
- He adds Tim to his contacts and annotates him with one explicit annotation which comes from a vocabulary with commonly-agreed meaning: *proxy*.

Tom also owns one resource (i.e., short message): *Can_I_aggregate_your_posts?*. He also defines the following access control policy for this resource:

Policy13: ;(activeBlogger:1); Read;. This policy means Read access of the specified resource is allowed to direct contacts of Tom (i.e., a distance of one) that have been annotated as *activeBlogger*. This policy has no "Resource Annotation" and thus, it is conceptually attached to only one resource (isAssignedTo).

Note 6: Tom defines some implicit context elements for himself using some rules: If I am not online in my IM client, then I am "travelling", otherwise I am "notTravelling" (i.e., in the office). Note 7: Tom also defines that if I am "travelling", then my contacts that have been annotated as *proxy* are eligible to access the same non-private resources. This concept (i.e., *private*) originates from a commonly-agreed vocabulary with commonly-agreed meaning.

Phil adds Jim to his contacts and annotates him with one explicit annotation: student. As we said, this explicit annotation comes from a commonly-agreed vocabulary with commonly-agreed meaning. Phil does not own any resources. He does not define any context elements for himself.

The other users (i.e., Mary, John, Paul, Ed, Lisa, Anna, Ben, Tim, Jim, and Adam) do not add any specific contacts or resources. However, Mary and John who have been annotated as *boardOfDirectors* by Alice, define context information for themselves. Mary sets a context element for herself: "online", and John sets a context element for himself: "offline".

Considering above connections and policies, we have granted access to the followings persons/resources. Clearly, each resource owner has full access to his/her resources.

Alice has Read and Revise access to *file2.ppt* via Bob, because *file2.ppt* is accessible to Bob's contacts that have been annotated as *student* and have a maximum distance of one or two to Bob. Alice fulfills these requirements.

Bob has access to two of Alice's resources: *www.resource1.com* and *www.resource2.com*, as he fulfils the required policies.

Mary has Read access to the short message of Alice, *I_need_to_talk_to_you_please*, as she is online.

John will not have Read access to the short message of Alice, as he is offline and does not meet the policy requirements.

Paul may gain Read access to *www.resource2.com* via Alice, as he has been annotated with an annotation (i.e., *friend*) which has been tuned by an attribute and an implicit value. In other words, in the case of a request, Paul's expertise will be extracted and if he has sufficient expertise (i.e., in social networks), he will gain access to the specified resource. Paul also has Revise access to *file3.doc* via Bob, as his collaboration level with Bob is "high" (as explicitly mentioned by Bob).

Ed may gain Read access to *www.resource2.com* via Alice, as he has been annotated with an annotation (i.e., *friend*) which has been tuned by an attribute and an implicit value. Like Paul, Ed's expertise is calculated at run-time.

Lisa does not have any access to Bob's resources, as her collaboration level with Bob is "low" (as explicitly mentioned by Bob).

Anna has Copy and Synchronise access to *image.jpg* via Bob, as she has been annotated as *mother* and based on the predefined vocabulary, "mother" is part of the "FAMILY" and she fulfills the requirements.

Ben has Copy and Synchronise access to *image.jpg* via Bob, as he has been annotated as *brother* and based on the predefined vocabulary, "brother" is part of the "FAMILY" and he fulfills the requirements.

Tom has Revise access to *file1.doc* which is shared via Bob to him and also Read access to *www.resource1.com* and *www.resource2.com* which are shared via Alice to him.

Phil has Read and Revise access to *file2.ppt* via Bob. Phil may also gain access to this message: *Can_I_aggregate_your_posts?* via Tom, if he has updated his blog in the past month.

Jim has Read access to *file2.ppt* via Bob, but he does not have Revise access to that file, as his distance to Bob is 2.

Tim may gain access to whatever that Tom has access, in case Tom is "travelling" (i.e., not

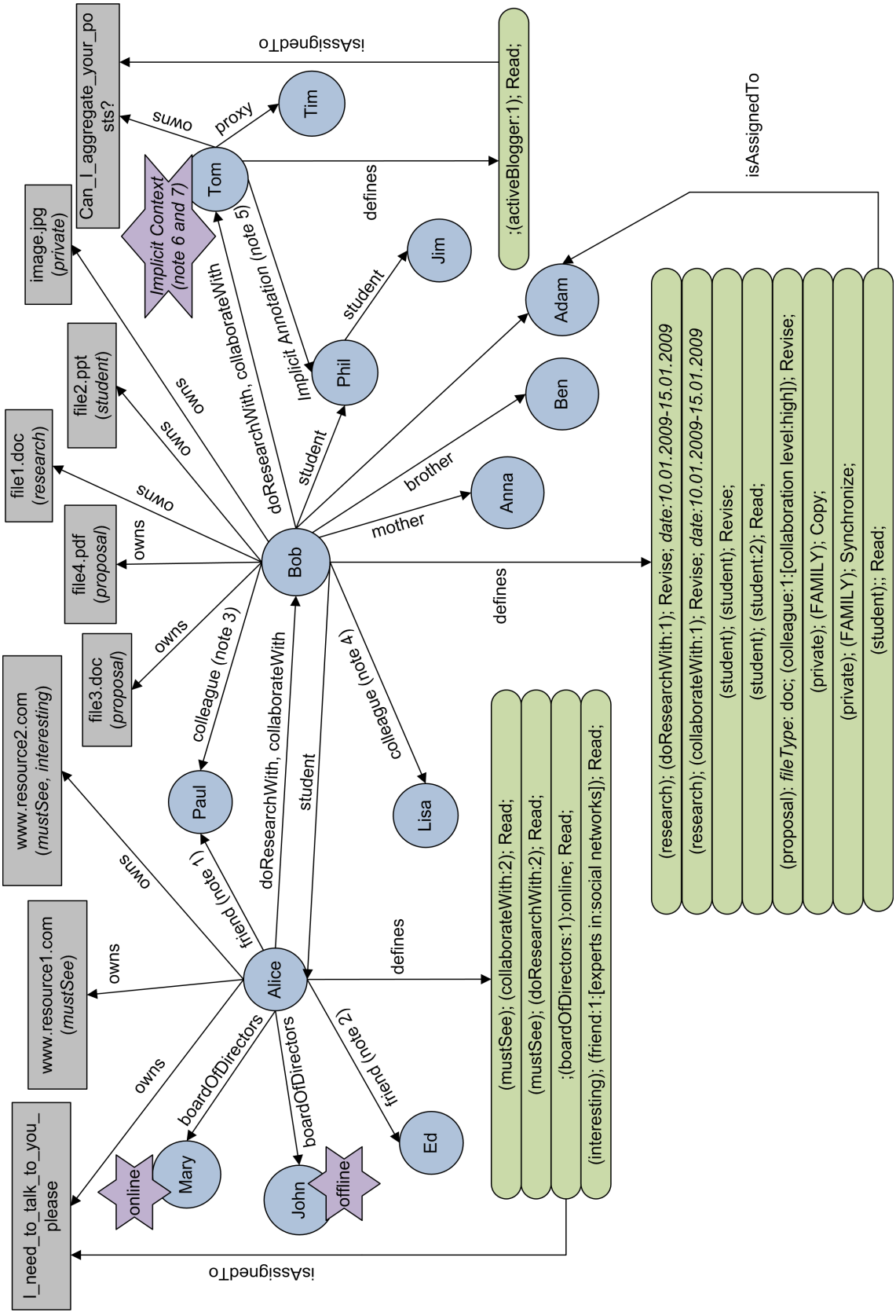

Figure 5.2: Use case scenario. Users annotate their contacts and resources and define access control policies for sharing their resources.

in the office). Tim will not have access to Tom's private resources.

Adam has Read access to *file2.ppt* via Bob.

If we consider the scenario presented in Section 5.1.1, we can realise that Bob's requirements can be expressed using AnBAC model policies. This would involve defining suitable explicit and implicit annotations and context information.

Note that here we do not discuss technical details of our model , however, the technologies for enabling (and implementing) implicit annotations, rules, context elements, commonly-agreed vocabularies, etc. were already developed and are available.

5.6 Simplified Annotation-Based Access Control

The model presented in Section 5.3 is comprehensive to address a wide range of user requirements. It is possible that not all features of the model (e.g., implicit annotations and context elements) are required for common use cases, such as sharing an announcement with contacts at a distance of two. Thus, it is useful to have a simplified version of the model, in which basic elements for realising a people-tag-based access control model are presented. To this end, we simplified the AnBAC model. In the simplified version, we have three main entities and two main concepts. The entities include *Person*, *Resource* and *Policy* and the concepts include *Annotation* and *Distance*. The description of these entities and concepts are the same as detailed specification. The other main difference is that a Policy has one Annotation and that is Person Annotation. Figure 5.3 demonstrates simplified version of the Annotation-Based Access Control model.

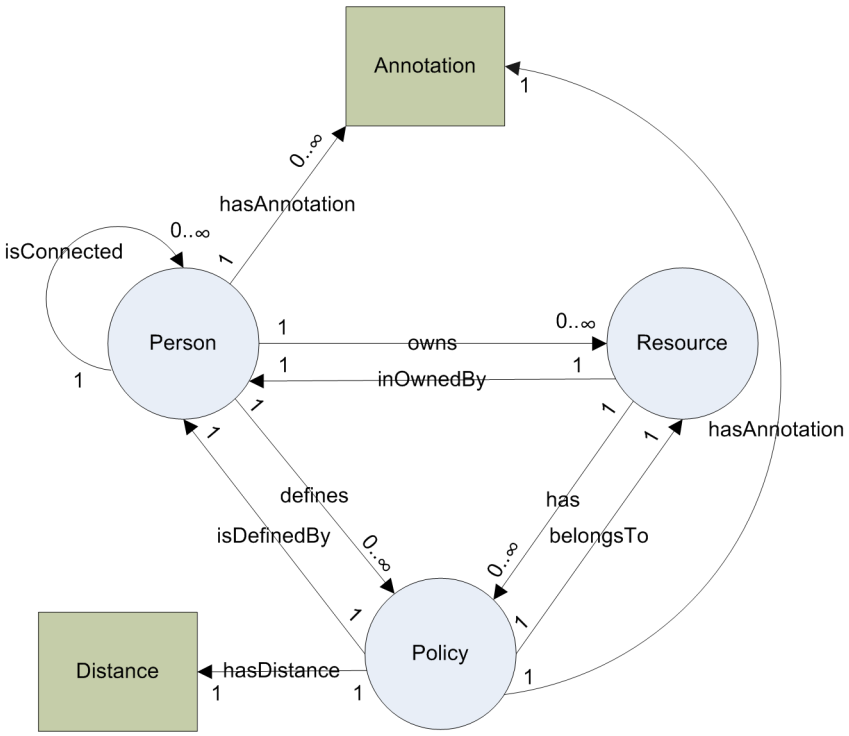

Figure 5.3: Simplified version of the Annotation-Based Access Control (AnBAC) model.

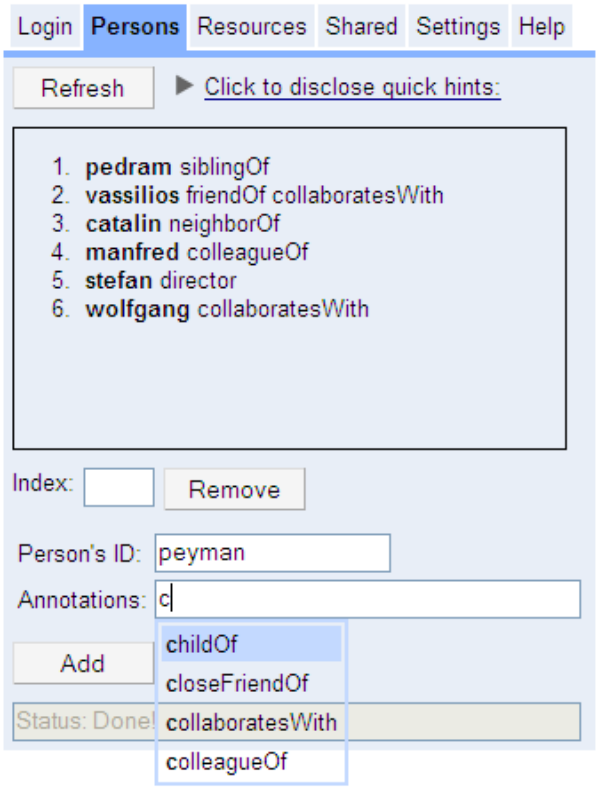

Figure 5.4: A snapshot of Uncle-Share: a tool for enacting the AnBAC model

5.7 Prototype

In order to evaluate our model, we developed a prototype called Uncle-Share. Note that current version of the prototype does not support all features and components of the model presented in Section 5.3. Uncle-Share has been developed as a gadget. Having this application as a gadget enables end users to use Uncle-Share together with other applications, something that increases the tool's usability, as users don't have to launch a new application or browse a new web page to utilise Uncle-Share. In particular, we decided to use iGoogle for developing our gadget, as Google provides sufficient documentation and support for developing gadgets. However, our gadget can be embedded into any other widget/gadget platform or website. The only client-side requirement is that the browser should support JavaScript.

Figure 5.4 demonstrates how the main user interface of the Uncle-Share gadget looks like.

The Uncle-Share gadget has six main tabs: Login, Persons, Resources, Shared, Settings, and Help.

- All users should subscribe first via the Login tab. The subscription is pretty straightforward. The current registration requires providing only a full name, a username and a password.

- Under the Persons tab, registered users are able to add contacts, annotate them, and remove some of their contacts.

- Under the Resources tab, registered users are able to add various resources (URLs/URIs/short messages) and assign different sharing policies to them.
- Under the Shared tab, users are able to see the resources that have been shared by others. They can set the distance to increase or decrease the scope of the shared resources.
- Under the Settings tab, end users are able to change the server and change their passwords. Uncle-Share server (SOA server) is a Java WAR file which can be installed on any machine and end users can have their own instances of Uncle-Share SOA server.
- Under the Help tab, there exists a link to the tutorial video[5] and some technical and contact information regarding the platform.

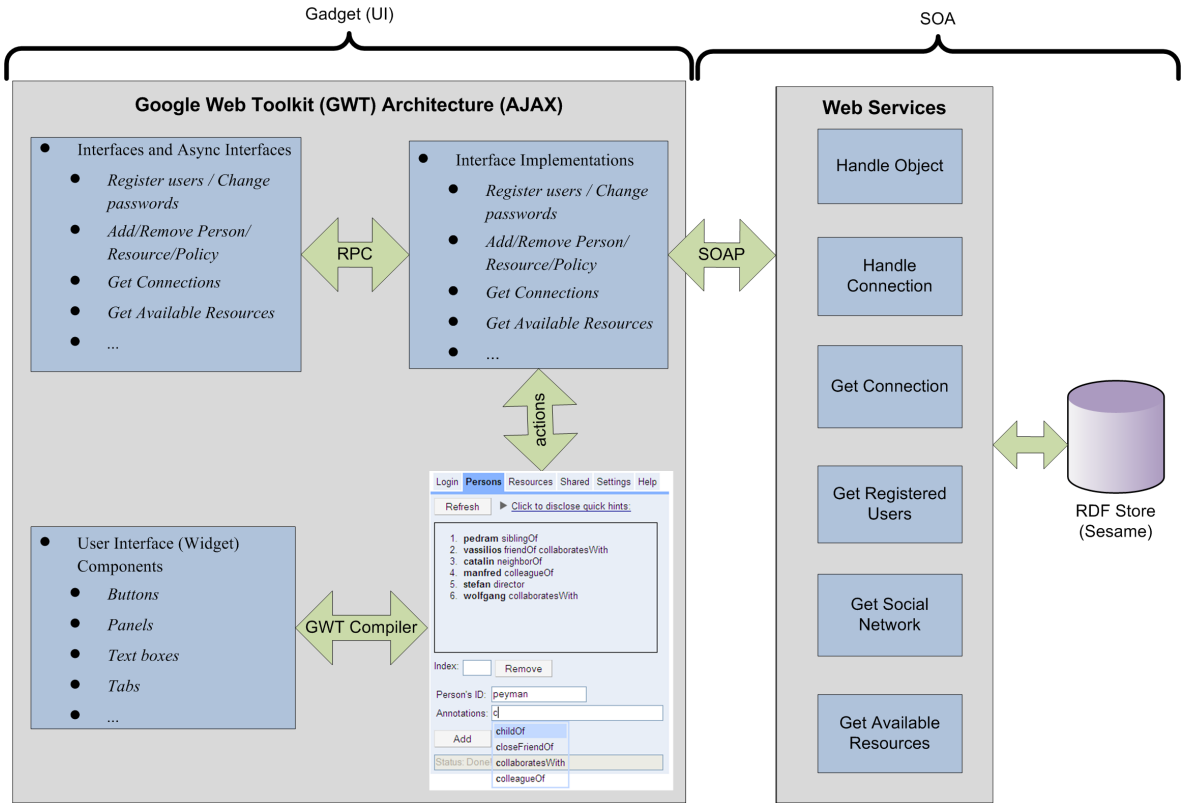

Figure 5.5: Overall architecture of the Uncle-Share. The User Interface (UI) communicates via SOAP messages with the SOA backbone.

All functionalities of Uncle-Share (registration, changing password, adding persons and resources, fetching shared resources, etc.) are wrapped as Web services. This approach enables developers to utilise all Uncle-Share's functionalities within their own separate applications, ensuring reusability and interoperability with various platforms. Currently, Uncle-Share provides the following services:

- Handle Object: This service enables end users to register themselves to the system and/or change their passwords.

[5] http://www.youtube.com/watch?v=eMDniFQ_-h0

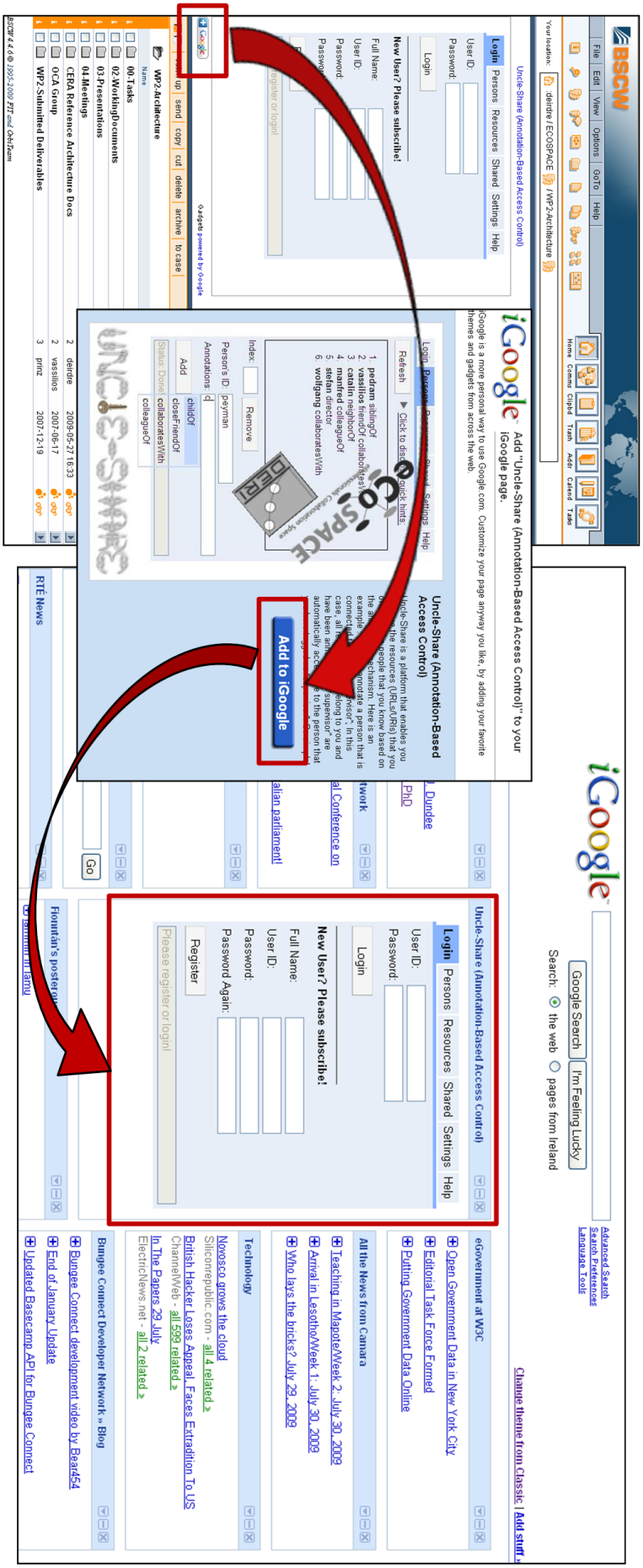

Figure 5.6: Embedding Uncle-Share gadget into iGoogle and OrbiTeam BSCW online shared workspace.

- Handle Connection: This service enables registered users to add connections between persons; persons and resources; and persons and policies. This service enables also registered users to annotate the connections between persons with closed and/or open terms. We used the RELATIONSHIP ontology [Davis and Vitiello, 2005] as a commonly-agreed closed vocabulary. This ontology is an extended version of FOAF and a set of terms for describing general relationships between people. The RELATIONSHIP ontology can be simply replaced with CoVoc – see Appendix I – or similar ontologies that model the relationships among people.
- Get Connection: This service enables end users to get the information on the connections shared by a specific person.
- Get Registered Users: This service returns the list of the registered users on the system.
- Get Social Network: This service returns the social network of the authenticated user in RDF.
- Get Available Resources: This service returns the available resources to a specific person based on the Distance input.

As illustrated in Figure 5.6, we have successfully embedded Uncle-Share into iGoogle and OrbiTeam BSCW shared workspace. Current version of Uncle-Share and its services do not support Context(s), Implicit Annotation(s), Action(s) and Attributes(s). The SOA backbone is based on Apache CXF[6] which eases the development of Web services. For building the AJAX-based gadget, we used Google Web Toolkit[7] (GWT). The GWT which is a free Java package gives us the basic useful elements of the UI, such as text boxes, buttons, tabs, etc. Figure 5.5 demonstrates the overall technical architecture of Uncle-Share. The Uncle-Share prototype is accessible online[8].

5.8 Comparison and Evaluation

The key point of the AnBAC model is to enable users to annotate their contacts and define access control policies by exploiting these annotations. Like many social media websites such as *flickr.com* and *delicious.com*, the model enables users to annotate their resources as well. Before discussing how the main idea behind the AnBAC model (i.e., annotating contacts) differs from similar approaches, we need to clarify the concepts of "group" and "role". A group is a named collection of users and possibly other groups [Sandhu, 1996]. A role differs from a group, as a role is either a named collection of users, permissions, and possibly other roles; or a named collection of permissions, and possibly other roles [Sandhu, 1996].

Common social networking sites (like *facebook.com*) enable users to assign their contacts to various groups. This is similar to the annotation mechanism. In this way, annotations may be used to create virtual groups. But annotation-based access control differs from group-based access control. First, we introduce the concept of implicit annotations which enable users to set the annotations at run-time. Second, we apply the Distance criterion which

[6] http://cxf.apache.org/
[7] http://code.google.com/webtoolkit/
[8] http://purl.oclc.org/projects/uncle-share-gadget-standalone

increases or decreases scope of the annotations in the policies. Third, users may benefit from explicit and implicit context information of their contacts and resources in order to assign more flexible policies. Fourth, an annotation can be tuned using attribute and value pairs. Finally, due to its definition, a group is usually non-empty and typically has at least two members [Sandhu, 1996], whereas an annotation can be freely assigned to just one person.

On the other hand, our approach for access control looks relevant to Role-Based Access Control (RBAC) [Ferraiolo and Kuhn, 1992, Sandhu et al., 1996], Generalized Role-Based Access Control (GRBAC) [Moyer and Ahamad, 2001] and other family members of RBAC. In brief, in RBAC, a user is assigned one or more roles. Each role has some defined permissions. Users will receive desired permissions through their roles or they inherit the permissions through the role hierarchy. RBAC is a quite successful access control method and is used in many platforms (operating systems, databases, etc.) and organisations. In GRBAC [Moyer and Ahamad, 2001], the authors extend RBAC by introducing subject roles, object roles and environment roles. RBAC, GRBAC and other family members of RBAC work well in well-structured (and perhaps hierarchy of) roles, permissions (and resources). The main difference between RBAC and our approach is that in RBAC, the roles are already defined by a role engineer, while in our approach, we have decentralised concepts (i.e., annotations) which are not necessary roles (from the semantics point of view). Even more importantly, it is the user that defines his/her own annotations and assigns them to his/her contacts which is more user-centric. On the other hand, annotations differ from roles, as an annotation is not directly linked with permissions. From the RBAC perspective, our model can be seen as an extension to RBAC through assigning user-centric or bottom-up versus top-down roles (i.e., annotations) without any permissions to a person's contacts and resources. The other main difference is the concept of Distance which increases or decreases the scope of policies in sharing resources, as people are connected together in a graph-like rather than hierarchy-like manner. Moreover, our model is more dynamic and flexible through introducing implicit concepts (i.e., implicit annotation, implicit context and implicit values of attributes). RBAC can be very useful in large and well-structured organisations, while our approach fits well for defining access control policies in ad-hoc social networks as these dramatically emerge through social platforms and CWEs.

Although, there are approaches that are similar to ours, like [Carminati et al., 2006b, Kruk et al., 2006], they exhibit certain differences: First, they do not completely support various attributes and values for annotations. As an example, instead of using percentages for expressing the trust level (e.g., friend 80%) like in [Kruk et al., 2006], in our model end users can express degrees of friendship in a more natural way with an annotation like *closeFriendOf*. The model becomes in this way more realistic and expressive, as in real-life we don't assign numerical values and percentages to our friends and relationships. Second, they do not support implicit and explicit annotations. Third, they do not use any explicit or implicit context information for defining access control policies. We enable users to use open as well as closed vocabulary to keep our model as practical and close as possible to real-life. We calculate the distance between two persons taking into account annotation values. This is important because annotations build a graph among people which may contain several paths between two persons and it is important to consider all paths when we want to reach a target person.

As Uncle-Share was developed in the context of the Ecospace project, the evaluation of Uncle-

Share was performed through the Ecospace Frascati Living Lab[9]. The Frascati Living Lab consists of several online shared workspaces (in our case OrbiTeam BSCW online shared workspace), where real end users try various features of a tool and provide feedback based on various criteria such as usability and usefulness of the tool and whether such tools help to improve collaboration and communication among e-professionals.

At least three end users of Frascati Living Lab tried Uncle-Share to share bookmarks with their contacts. Analysing their feedback reports showed that there was a consensus indicating that propagation of policies (i.e., distance criterion) brings interesting dimensions for information dissemination. They were also concerned regarding several issues. We analysed their concerns and generalised them as follows: Policies in the basic AnBAC model are built on top of annotation and distance. Both annotation and distance are sources of ambiguity and imprecision as a user must choose them in an ad-hoc fashion. Sources of imprecision for annotation include:

- Users may not have any idea of coverage.
- There may be synonymous annotations which the user may have omitted.
- Cold-start problem: Suitable annotations could be proposed to end users at the beginning. Moreover, there may exist errors in assigning annotations to people in the first place.
- Annotations may be too fine-grained or too coarse-grained.
- Annotation drift.
- Users may forget the annotations that they assign to other users or themselves.

Sources of imprecision for distance include:

- The user may not have any sense of how far information may propagate over the network. As an example, the difference between distance two and distance three is not evident.
- Non-IT users may face difficulties in understanding the concept of distance.

These points are valid points that we are currently trying to address. For example, as we presented in the previous chapter, our expertise extracting tool can be used to address the so-called cold-start problem. In other words, we propose expertise elements that are the result of applying information mining techniques to be used by users for annotating their contacts. In the next chapter, we show an approach to assist users to adopt AnBAC in an online Web 2.0 platform taking into account the aforementioned comments.

5.9 Conclusion and Future Work

In this chapter, we presented an annotation-based access control model for social and collaborative platforms which is based on people-tags. We also presented a supportive tool called

[9] https://frascatilivinglab.eu/

Uncle-Share to realise the AnBAC model. Our approach is applicable in both Web-based collaborative information spaces like Web 2.0 social platforms and Collaborative Working Environments (CWE). Our model can be seen as an extension to role-based access control, where people are able to define their own roles and assign them to others in a user-centric manner. The AnBAC model aims to provide an intuitive and real-life-oriented approach for a user-centric access control for personal (online) material.

Annotation and distance are two main elements of the AnBAC model. Both annotation and distance are sources of imprecisions due to annotation drift and synonyms, granularity issues, distance ambiguity, etc. Therefore, the model needs a "policy advisor" to help people drafting their policies for sharing online resources. The goal of the policy advisor is to help users share relevant resources with relevant people while considering factors such as user feedback. In the next chapter, we present our policy advisor to address aforementioned drawbacks.

Chapter 6

People-Tag-Based Policy Advisor[1]

> I do not fear computers. I fear the lack of them.
>
> Isaac Asimov

In the previous chapter, we presented our access control model that is based on people-tags. Our evaluators suggested that such access control models require a policy advisor to assist users in drafting information sharing policies, as otherwise, this may lead to information overload and information shortage in collaborative environments. In this chapter, we present fundamental elements of such policy advisors. Our policy advisor helps users to avoid information overload and information shortage in a network of people-tag-based profiles. It provides optional real-time advice on whether to send a particular piece of information to the network. Moreover, it also recommends well-connected topic-sensitive users who may act as hubs for broadcasting a piece of information to a larger relevant audience. Each piece of advice is supported by declarative explanations. We enact our policy advisor using micro-blogging which is a Web 2.0 service that is currently well adopted by many enterprises (so-called enterprise 2.0) for propagating information. In particular, we focused on Twitter[2], the fastest growing micro-blogging service on the Web. Recently, Twitter enabled a feature called Twitter Lists which can be perceived as a list of tagged people and this gave us real-world data to evaluate our policy advisor. The evaluation suggested that our approach helps users to know their network better and also to find better hubs for propagating community-related information.

6.1 Introduction and Motivation

The AnBAC model as presented in Chapter 5 requires a policy advisor for helping users to share relevant information with relevant people. In other words, the main mission of the policy advisor is to ensure that whoever that receives the information is actually interested to access that information and also people who do not receive a particular information are not interested to access that. As our model aimed to be a trade-off between usability and

[1] This chapter is mainly based on [Nasirifard and Hayes, 2011].
[2] http://twitter.com/

security for mass personal information diffusion, we do not discuss eligibility or authority of information recipients here. The role of the policy advisor becomes more evident with a simple scenario. In our example, Bob is the main actor. He works in a large research-oriented corporation and wants to share a conference announcement on *marketing* with his colleagues who have similar research agenda to the *marketing* topic. Either Bob sends the announcement to all of his colleagues regardless of their interests, or he may send the announcement to a group of colleagues that he is aware of their interests and ask them to further circulate the announcement. Both approaches have considerable drawbacks. The drawback of the first approach is that it may lead to information overload, as uninterested people will also get the announcement. The second approach has three main drawbacks: The first one is that circulating an announcement through friend of a friend may not reach all (relevant) colleagues and hence they face information shortage. It could also lead to information overload, as some people may receive multiple copies of the same announcement. This approach also brings additional overhead for people who are asked to further forward the announcement. To avoid such drawbacks, Bob decides to share the conference announcement using the AnBAC model. To do this, he drafts an AnBAC policy like (marketing:2). This policy shares the announcement with people who have been tagged with *marketing* and have a maximum distance of two to Bob. Bob looks for some advice (perhaps with convincing explanations) to check if this policy is efficient enough to keep information overload and information shortage at a minimum level in his corporation. This is the task of the policy advisor which we address in this chapter.

The remainder of the chapter is structured as follows. In Section 6.2, we describe our approach for the policy advisor. In this part, we describe fundamental elements of a people-tag-based policy advisor. In Section 6.3, we present our arguments regarding our decisions to use Twitter for evaluating our policy advisor. We also present a brief overview of micro-blogging and Twitter system including the terminology that is used in Twitter. In Section 6.4, we present related work and our arguments to support advantages of filtering at source initiative for Twitter. Next, in Section 6.5, we introduce our novel Twitter assistant called Tadvise by describing its components in detail. In Section 6.6, we present our experiment to evaluate our policy advisor and also how me measured information overload and information shortage. Our experiment was conducted through personalised online surveys and each question of the survey was equipped with an optional comment box. The analysis of comment boxes is presented in Section 6.7. Finally, in Section 6.8, we conclude and have an overview on future work.

6.2 Policy Advisor

There is research on applying information filtering techniques at information recipients' side, such as Tapestry [Goldberg et al., 1992]. Filtering at the target level is an effective approach for controlling information flow that reaches the target person. The drawback of such approaches is that there may exist more information that the target person is interested in receiving, but such information never finds its path to the recipients, as the information sender is not aware of such nodes in the network. For example, there may exist a perfect candidate for collaboration at a friend of friend (i.e., a distance of two) level, but perhaps these people will never find each other. Filtering at source is an approach to address such drawbacks.

Our policy advisor acts as an assistant to help users in drafting appropriate policies at information sender side. In other words, the aim is to send community-related information to users who should or are interested to receive it. In the previous chapter, we defined $Users(\Psi(u_i, p_j)) = Users(\Psi(u_i, (t, t', d, a)))$ which returns eligible users that have an access of type a to associated resources of u_i (based on p_j) and is defined as $\{(u_j, r_k, a) \,|\, Tags(u_j) \subseteq t' \wedge Tags(r_k) \subseteq t \wedge \Delta(u_i, u_j, t') \leq d)\}$. The goal of the policy advisor is to keep a balance between information overload and information shortage in a network of people-tag-based profiles (i.e., result of $Users(\Psi(u_i, p_j))$).

The two main responsibilities of the policy advisor are: a) analysing social network of a user at distances of one and two in order to determine if majority of the user's contacts are interested to receive the information, and b) recommending several topic-sensitive hubs who may propagate the information further in the network.

Our policy advisor uses a traffic light metaphor. In other words, there exist three different kinds of advice: a) the green light advice, b) the amber light advice, and c) the red light advice. The green light advises users to freely propagate the resource in the network as there exist many users in the network that might be interested in receiving it. The amber light prompts that the resource might not be relevant to majority of the users, however, there may exist users who might be interested in the resource. The red light prompts that the ratio of relevant to non-relevant users in the network is very low and thus, it is better not to propagate the resource further in the network. None of these light colours was aimed to prohibit users from generating novel contents, but instead give a hint on the relevancy of a resource to contacts of a user at distances of one and two.

The policy advisor has several components. Figure 6.1 illustrates an overall view of the policy advisor components. Before proceeding further, we formally define a system which the policy advisor operates on. Let S be a system with n nodes (users) $U = \{u_1, u_2, ...u_n\}$, where there exists a set of unidirectional relationships R between users, so that if u_i makes a relationship ($r_{ij} \in R$) with u_j, we call u_i a follower of u_j and u_j a followee of u_i. We denote this relationship with $u_i \rightarrow u_j$. We assume that the system S is open, so that any user can make relationships with other users. The set of followees and followers of u_i are denoted by U_i^{fr} and U_i^{fo} respectively. User u_i can assign zero or more tags ($\{t_1, t_2, ...t_m\}$) to each of his/her followees. We define a function $lists(u_j)$ that takes a user u_j as input and returns (u_i, t_k) pairs meaning that u_i has assigned t_k to u_j.

In the following section, we introduce the main components of the policy adviser in detail.

6.2.1 User Profile Builder

This component of the policy advisor generates people-tag-based user profiles. People-tags need to be pre-processed, ranked and enriched to make them ready for building user profiles and later be used in the policy advisor. The pseudocode of building user profiles is presented in Algorithm 3.

Initially, this component has to enrich people-tags. There exist different approaches such as FLOR [Angeletou et al., 2009] (FoLksonomy Ontology enRichment) that can be used for enriching a tag set. For instance, semantic distances between the terms can be considered to generate more appropriate tags for a given set of tags. As an example, for a given set of tags such as {*Semantic Web*, *DERI*}, other phrases in the domain such as *Linked Data* and *Open*

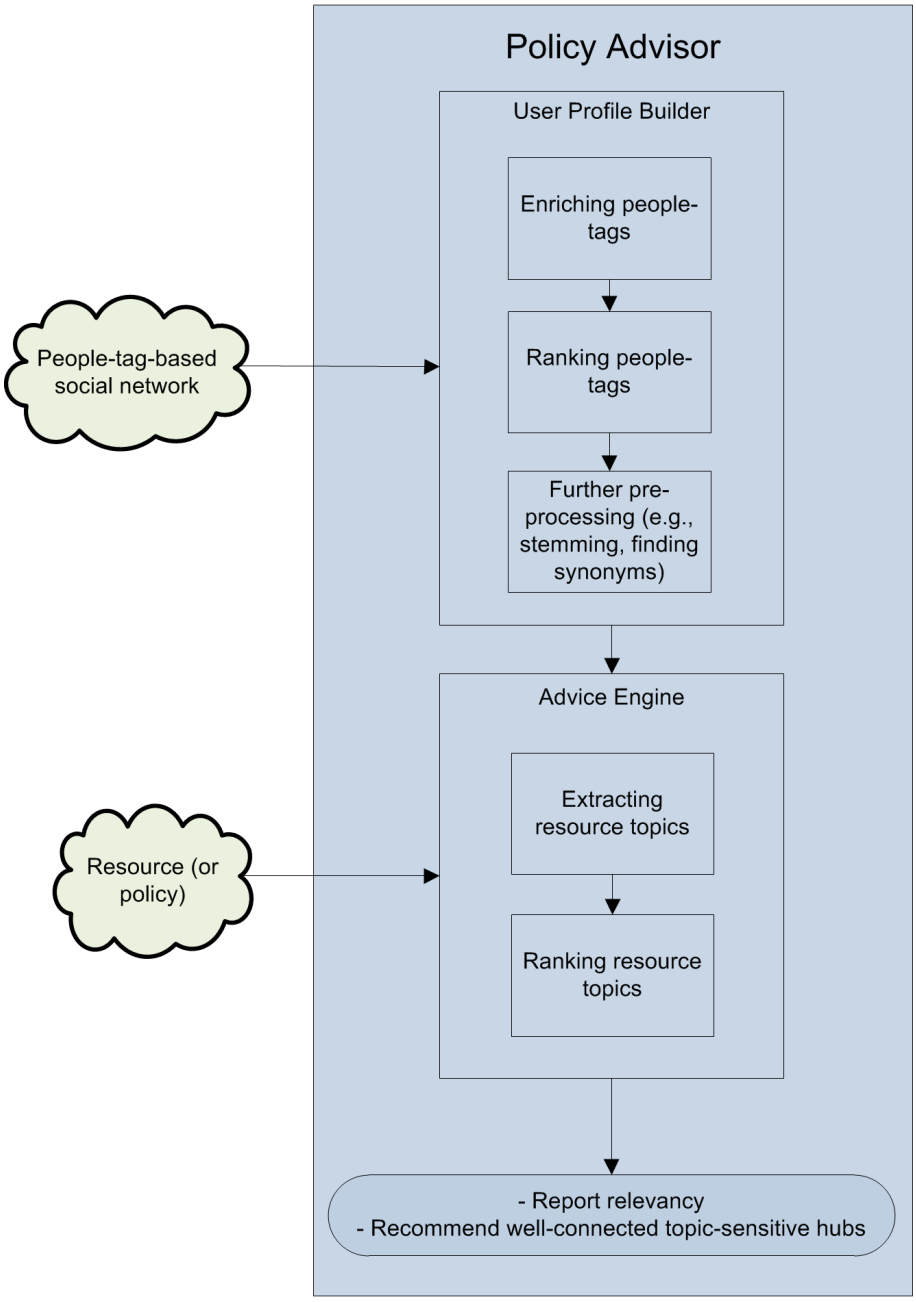

Figure 6.1: An overall view of different components of a people-tag-based policy advisor.

Data can also be considered and added to the initial set.

For building profiles, it is important to identify the most important tags in a profile (i.e., rank the tags). Ranking aims to minimise the effect of the noisy tags such as those tags which are assigned by spammers. In order to rank a tag, a value so-called *weight* of a tag should be assigned or calculated. The weight of a tag is domain dependant and is directly proportional to the rank of the user who assigned it. The rank of a user depends on the rankings of the users who tagged him/her. Figure 6.2 illustrates schematic view of a weighted user profile. It

input : Tags of a user u_i: $lists(u_i)$
output: Profile of the u_i: $profile(u_i)$

$profile(u_i) \leftarrow \emptyset$;
$tags \leftarrow \{t_k \mid (u_j, t_k) \in lists(u_i)\}$ //u_j assigned t_k to u_i;
$tags \leftarrow tags \cup enrichTags(tags)$ //to get more appropriate tags;
foreach $tag \in tags$ **do**
  $rank \leftarrow getRank(tag)$;
  $profile(u_i) \leftarrow profile(u_i) \cup (stem(tag), rank)$;
  $synonyms \leftarrow getSynonyms(tag)$;
  foreach $synonym \in synonyms$ **do**
    $profile(u_i) \leftarrow profile(u_i) \cup (stem(synonym), rank)$;
  end
end
return $profile(u_i)$;

Algorithm 3: Build User Profile (BUP).

is easy to see that the tags assigned by users with higher ranks have larger weights.

As people-tags consist of arbitrary phrases, we need to stem them in order to find root of the terms. For example, consider a tag like *Web Services*. As this phrase is plural, we have to find root of this tag in order to obtain an unambiguous meaning. We also need to find synonyms of the tags to avoid redundant tags and also to enrich the tags. These transformations enable us to build the user profiles which are unambiguous and enriched.

6.2.2 Advice Engine

The advice engine component provides real-time topic-sensitive advice. It gets two items as input: people-tag-based user profiles and a resource such as a URL (or a policy for sharing that resource). Initially, the advice engine tries to extract main topics of the resource. Various methods such as natural language processing (NLP) techniques can be used to extract key topics of a resource. Then based on extracted topics, the relevancy of the topics to users at distances of one and two can be measured. In other words, in order to assess the relevancy of a resource t to a user u_j, weighted user profile of u_j containing metadata of u_j's communities, interests, expertise, etc. that is built in the previous step is being used.

Our system introduced in Section 6.2 follows a one-to-many approach for information diffusion. So a piece of information that is sent by u_i will propagate to all U_i^{fo}. Obviously, it is possible that not all members of U_i^{fo} are interested in a particular topic. The challenge is to advise users to propagate items that can potentially be interesting for the *majority* and at the same time not stopping them of generating novel contents. For detecting the tags that are relevant to majority of the followers, we build aggregated user profiles that comprise user profiles of all followers of a seed at distances of one and two. We represent such aggregated profiles as $followersProfile1(u_i)$ and $followersProfile2(u_i)$ respectively. We cluster the sorted weights of $followersProfile1(u_i)$ and $followersProfile2(u_i)$ into two partitions which represent frequently occurring (thus highly weighted) lists and infrequently occurring lists. Rather than applying a fixed threshold to each profile, we find a *knee point* between the two partitions by applying a clustering algorithm. Algorithm 4 shows this piece of process

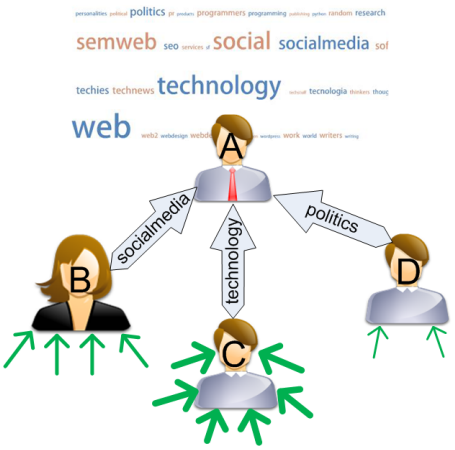

Figure 6.2: Schematic view of user A's weighted profile using a tag cloud notion. Thickness of the green arrows associated with the users B, C, and D symbolises ranking of their contacts. For instance, as user C has more high-ranked contacts in the network, therefore, the tags that this user assigned to user A (e.g., technology) has also larger weights. For simplicity, we did not demonstrate other users who tagged user A in the figure.

for building $followersProfile1(u_i)$.

input : User u_i
output: $followersProfile1(u_i)$: Single clustered profile for U_i^{fo}

$followersProfile1(u_i) \leftarrow \emptyset$
foreach $u_j \in U_i^{fo}$ **do**
  $followersProfile1(u_i) \leftarrow followersProfile1(u_i) \cup BUP(lists(u_j))$;
end
cluster the $followersProfile1(u_i)$ into two clusters, so that the tags within each cluster have similar rankings;
return $followersProfile1(u_i)$;

Algorithm 4: Build Followers Profile (BFP).

The first partition which groups high-ranked tags, represents the source of green light advice and when tags are matched at query time to this partition, we encourage users to propagate the resource further in the network. The second partition represents the source of amber light advice. Tags that are matched at query time to this partition represent interests that have low support among the user's followers. The policy advisor shows a red light, if it is unable to find any representative tags within these partitions.

If a user wants to find well-connected hubs in relation to a topic, the policy advisor makes recommendations of several well-connected topic-sensitive users. Algorithm 5 shows this process. The input to this algorithm is a directed graph g. We build this graph using $followersProfile1(u_i)$. The root of g is the seed u_i. We add all members of U_i^{fo} to g $(u_j \rightarrow u_i)$. The reason is that when u_i propagates a message/resource, all of her followers receive that message/resource, regardless of their interests, and thus can act as potential hubs. Then, we add all interested followers of each follower of u_i to g using $followersProfile2(u_i)$. We pass g to the algorithm. The algorithm finds k hubs in g using in-degree so that the

hubs cover as many interested followers (at a distance of two from u_i) as possible and have as few overlapping followers as possible with each other. The reason that we also consider overlapping followers is to minimise redundancy. The "hub score" in Algorithm 5 indicates the number of interested users who receives a tweet through a hub.

input : Directed graph (g)
Integer k // number of recommended hubs
output : $candidates \subset g$

$candidates \leftarrow \emptyset$;
$covered \leftarrow \emptyset$;
while *size(candidates)!=k* **do**
  sort the nodes in g based on their scores (i.e., hub score);
  node $\leftarrow$ get the node with the highest hub score, so that $followers(node) \cap covered$ is minimum;
  $candidates \leftarrow candidates \cup node$;
  $covered \leftarrow covered \cup followers(node)$;
  $g \leftarrow g - followers(node) - node$;
  if $g == root(g)$ **then** *break*;
end
return candidates;

Algorithm 5: Finding Well-Connected Hubs.

Explaining Our Recommendations

[Schafer et al., 2007] argue that it is useful to persuade users that the provided recommendations are useful. [Herlocker et al., 2000] also provide some good insights into explaining collaborative filtering recommendations and suggest that rating histograms seem to be the most compelling ways to explain a prediction. However, their findings are limited to the system where there is a rating mechanism for items. In order to convince end users that our recommendations are relevant, our approach is capable of providing explanations. As such explanations are domain dependant, in Section 6.5.3, we elaborate further on this.

6.3 Policy Advisor Testbed

We require a real-world dataset to evaluate effectiveness of our policy advisor. To this end, we compared webtop platforms versus desktop ones. Using a webtop platform instead of a desktop one, where more people are involved in – despite their various flavours in selecting operating systems – can potentially give us more real-world users. The other main advantage of using a webtop platform is that end users do not need to download or install any additional software, library or package.

In particular, we decided to use the micro-blogging platform, Twitter, as a webtop platform due to several reasons. The main reason is due to the fact that recently, Twitter has introduced Twitter lists which can be seen as a mechanism for tagging people. In other words, Twitter lists help users to build profiles for their friends in a collaborative manner. By looking at Twitter lists as people-tags, we can spot various characteristics such as affiliations, expertise, interests and topics that people are associated to (from their friends' perspective). Other reasons that affected our decision of using Twitter are listed below: Twitter has shown a

dramatic growth rate in recent years. As a report, more than half of researchers in our institute[3] are using Twitter for collaboration purposes which is a strong supportive argument for the gradual shift towards enterprise 2.0. Most current enterprises such as IBM, Microsoft, etc. use Twitter or have their own internal Twitter-like systems such as BlueTwit. They use micro-blogging to boost collaboration and communication intra- and inter-organisations (see Section 6.3.1 for more details). Moreover, the number of platforms such as Google live search, LinkedIn, Wordpress, etc. that have been integrated with Twitter or provide plug-ins for Twitter is rapidly increasing. Among others, the new release of OrbiTeam BSCW shared workspace (i.e., version 4.5) provides Twitter integration through widgets. In addition to all these arguments, by exporting our analysis results using semantics such as ontologies and RDF, desktop and online platforms within organisations such as OrbiTeam BSCW or KDE[4], can benefit from our results.

6.3.1 Micro-blogging and Twitter

Micro-blogging is a sort of blogging with the difference that micro-blog posts are smaller than normal blog posts and have typically a limited length (e.g., maximum 140 characters). [Ehrlich and Shami, 2010] found that micro-blogging increases visibility of a topic compared to discussing it over email or instant messenger. They also found that the main purpose of micro-blogging is to provide information or to engage in conversation.

Twitter, the micro-blogging service that was launched in 2006, is the fastest growing micro-blogging service on the Web[5]. As by March 2011, statistics shows that Twitter users are tweeting with an average rate of 140 million tweets per day[6]. Scholars, professionals, celebrities, organisations, conferences[7], animals[8], plants[9] and even houses[10] are tweeting. In Twitter jargon, a user is called a *twitterer* or *tweep*. A twitterer u can send short status messages (so-called tweets) to his/her *followers*, the set of other twitterers who explicitly follow u's updates (i.e., tweets). A *followee* of user u is any twitterer whom u chooses to follow[11]. The set of followees and followers of any user may be completely disjoint.

People use Twitter for various purposes [Krishnamurthy et al., 2008], but the main goal is to diffuse a small piece of information or to share a piece of knowledge. An early Twitter analysis by [Java et al., 2007] shows that user intentions of tweeting differ. The main purposes include being able to share information, report news, chat, and have conversations. A later analysis [pearanalytics, 2009] categorised tweets into six main groups: news, spam, self-promotion, pointless babble, conversational, and pass-along values. From another perspective, [Ehrlich and Shami, 2010] categorised tweets into six different groups: sharing status, providing information, retweeting, asking questions, direct posts to someone, and directed question

[3] For a list of these researchers, refer to this Twitter List: http://twitter.com/#!/epeyman/deri

[4] http://nepomuk.kde.org/

[5] http://eu.techcrunch.com/2010/01/26/opera-facebook-largest-mobile-social-network-twitter-fastest-growing

[6] http://blog.twitter.com/2011/03/numbers.html

[7] http://www.chi2010.org/socialmedia/twitter_stream.html

[8] http://www.dailywireless.org/2010/01/26/puppy-tweets/

[9] http://www.botanicalls.com/kits/

[10] http://www.readwriteweb.com/archives/the_tweeting_house_twitter_internet_of_things.php

[11] For a long time, the Twitter website used the term *friend* instead of *followee*. As *friend* is normally perceived as a reciprocal relationship among two persons, we use the term *followee* to indicate a directed relationship.

to someone. A recent analysis by [Brooks and Churchill, 2010] suggest that Twitter users often no longer tweet simple status updates, but rather share/look for information, recommendations and news. As above studies show, sharing information and announcements (e.g., conference calls, events) is one of the main use cases of Twitter. In such cases, people typically tweet announcements and ask their followers to 'retweet' it in order to propagate the message further (by using a phrase like "Please RT"). Retweeting is Twitter slang to describe how the recipient further distributes the message among his/her follower network. [Chiu et al., 2006] show that trust or shared language do not have significant impacts on quantity of knowledge sharing in virtual communities. They mention that people tend to share their knowledge with more people in virtual communities. Due to dramatic growth of Twitter, there is a high probability that users may overlook important and relevant tweets and retweet requests. When twitterers retweet each others' tweet, it is not clear how information propagates in the network via retweeting. If such announcements do not propagate efficiently in the network, some potentially interested followers may miss or overlook important tweets. As such, Twitter (and in general micro-blogging) is yet another dynamic information channel where the user needs assistance to manage incoming and outgoing information streams.

In the next section, besides presenting related work, we also present supportive arguments for effectiveness of filtering-at-source for boosting collaboration and communication in Twitter (and in particular among scholars and professionals).

6.4 Filtering-At-Source and Related Work

The need to group people in Twitter and also an assistant for managing tweets have been suggested by earlier work [Java et al., 2007]. Perhaps, one of the most well-known Twitter assistants is MrTweet[12]. MrTweet analyses twitterers' networks and provides personalised recommendations for new followees. A similar service by Google, called Google Follow Finder[13], also recommends new followees by extracting patterns of followers from the social graph of the user. However, none of above services advise the user whether a (re)tweet is relevant to his/her followers or recommend followers that have the best location in the network to propagate the message to a relevant audience. [Boyd et al., 2010] mentioned that "Retweeting for social action is most successful when the retweeter has a large network and occupies structural holes, or gaps in network connectivity between different communities." [Ehrlich and Shami, 2010] also mentioned the importance of intelligent filtering for widespread use of micro-blogging in the workplace. As a candidate for intelligent filtering, they mentioned to enable grouping in Twitter. [Zhao and Rosson, 2009] also mentioned the importance of filtering and grouping in Twitter.

Typically, filtering involves the removal of unwanted items from an incoming stream of items and the preservation of items that would be of interest to a target recipient. Some approaches and tools (e.g., TweetDeck[14]) exist to filter tweets based on various criteria[15]. While such approaches are undoubtedly useful at blocking *incoming* tweets, they do not address the problem of classifying which *outgoing* tweets/retweets might be of interest to a twitterer's

[12] http://mrtweet.com
[13] http://www.followfinder.googlelabs.com/ – Note: This service has phased out by Google.
[14] http://www.tweetdeck.com/
[15] http://mashable.com/2009/07/03/twitter-filter/

followers. As a user's follower list increases, it becomes more difficult to keep of their interests and to locate the best users who are well-connected topic-sensitive hubs for propagating community-related tweets. As stated, our focus is not to prohibit users of generating novel contents. [Boyd et al., 2010] had an in-depth study on (re)tweeting in Twittersphere. Their study shows that some people pay attention to interests of their followers, when they want to retweet something. Moreover, [Bernstein et al., 2010] found that "sharing is motivated by a perception of what friends would like to see, but held back by concerns about spamming and misreading friends' interests." To this end, they developed a filtering-at-source plug-in for Google Reader, called FeedMe, that recommends relevant friends who might be interested in an item. FeedMe enables users to share relevant items with interested friends via email.

Micro-blogging is heavily used among scholars and also inside enterprises for boosting collaboration and communication. Using micro-blogging within enterprises allows employees to become aware of what their colleagues are working on [Ehrlich and Shami, 2010]. [Zhang et al., 2010] showed that professionals use Yammer[16], an enterprise-oriented micro-blogging service, to find new professional connections and also to learn on what others are doing. [DiMicco et al., 2008] found different patterns of social networking sites usage inside and outside enterprise. Their study shows that professionals inside a large corporate tend to get more familiar with weak ties within social networks. Moreover, professionals are also interested to strategically connect to large audience in order to get more high-quality replies and feedback in relation to work-related items. [Constant et al., 1996]'s finding also supports this hypothesis. They found that it is possible to get qualified answers from weak ties – relationships with acquaintances or strangers. In other words, it is always useful to send a question to a large relevant audience. Filtering-at-source is an approach to broadcast a message further to large relevant audience (e.g., people with specific expertise) and thus reducing information overload and information shortage within communities.

Our novel filtering-at-source Twitter application (called Tadvise[17]) assists users to classify which tweets/retweets are likely to be of interest to their followers and to select which followers would best be able to propagate the message to a relevant audience. The evaluation results showed that Tadvise helps users to know their followers' interests better and can assist users in relation to (re)tweeting. Note that our focus is not to prohibit users of generating and submitting novel contents, but on the other hand, to know their followers' communities better.

In late October 2009, Twitter introduced a feature called *Twitter list* which allows users to assign other users to various lists. Twitter lists are similar to grouping and can be also perceived as a way of *tagging* people. In this chapter, we use people-tag/tag/Twitter list to refer to each other. Our work (Tadvise) uses Twitter lists for building user profiles in order to make recommendations on tweet diffusion. Tadvise is most useful for those Twitter users interested in sharing information, recommendations and news (such as conference announcements and events) with like-minded users in a community. Earlier work [Weng et al., 2010, Java et al., 2007] demonstrated the community (i.e., highly reciprocal network) structure of the Twitter network. As such, the scope of our work is focused on community-related pass-along tweets, as categorised by [pearanalytics, 2009]. For example, tweets like "deadline extended for next drupal conference..." are considered to be in the scope of Tadvise, as they are relevant to a particular interest group. On the other hand, informal status updates such as "having breakfast now..." are out of scope of Tadvise. We analyse the

[16] https://www.yammer.com/

[17] http://purl.oclc.org/projects/tadvise or http://tadvise.net

followers of a seed user (followers at a *distance of one*) plus the followers of the followers of the seed (followers at a *distance of two*) when considering the relevant audience for a (re)tweet. While not actually following the seed, followers at a distance of two may be influenced by or be interested in a seed's community-related tweets due to the dense community structure of the network [Weng et al., 2010, Java et al., 2007] and principle of locality [Chen et al., 2010].

The work by [Kim et al., 2010] is one of the earliest studies on Twitter lists. They applied X^2 feature selection tool to tweets of users who were associated with a Twitter list and concluded that words with high X^2 values are useful for getting informative characteristics related to those users (associated with the Twitter list). [Chen et al., 2010] present a URL recommendation system for Twitter. They build user profiles using tweets and apply similarity measures to see if a user might be interested in a public link or links posted by his/her followees of followees. To the best of our knowledge, we are not aware of any study on – using Twitter lists for – filtering tweets at information sender side.

6.5 Tadvise Overview and Components

Tadvise builds user profiles for twitterers and leverages such profiles to make recommendations on which tweets or retweets could be sent to a twitterer's followers. For each user, it analyses his/her network of followers to a distance of two, as earlier work showed the community structure of Twitter [Weng et al., 2010, Java et al., 2007], and reports if network members would be interested in a particular tweet. Moreover, for each tweet it recommends top-k (i.e., k is end-user defined) relevant, well-connected followers who could propagate the tweet to a larger relevant audience.

To register for Tadvise, a twitterer u simply chooses to follow the Tadvise Twitter account (i.e., @Tadvise). Once notified, Tadvise crawls the social network of u and builds appropriate user profiles. After completing these steps, which are performed offline, Tadvise sends a direct message to u, indicating that it is ready to provide advice. By visiting the Tadvise homepage (see Figure 6.3), u can benefit from advice and/or tweet a message directly to Twitter. As stated in Section 6.2, the policy advisor uses a traffic light metaphor, however, for the implementation, we decided to use more intuitive icons. The reason is that the red light in the traffic light is commonly perceived as a way to stop an ongoing activity. This could prohibit users of posting novel contents as discussed in Section 6.7.7. To this end, we had consultation with experts in user interaction design and they designed the illustrated icons in Figure 6.4 for our use case. The advantage of using these icons is that none of them looks prohibitive from users' perspective.

Tadvise has three main components, namely a *crawler*, a *user profile builder* and an *advice engine*. In the following section, we describe all three components of Tadvise.

6.5.1 Tadvise Crawler

The crawling component of Tadvise gets a seed as input and uses Twitter API and white-listed Twitter accounts for crawling twitterers. The crawling component does its job in two steps. First, it crawls the network of followers at distances of one and two from a seed (i.e., breath-first mechanism). The second step of crawling consists of crawling Twitter lists. This step takes the network of followers from the first step and crawls Twitter lists associated

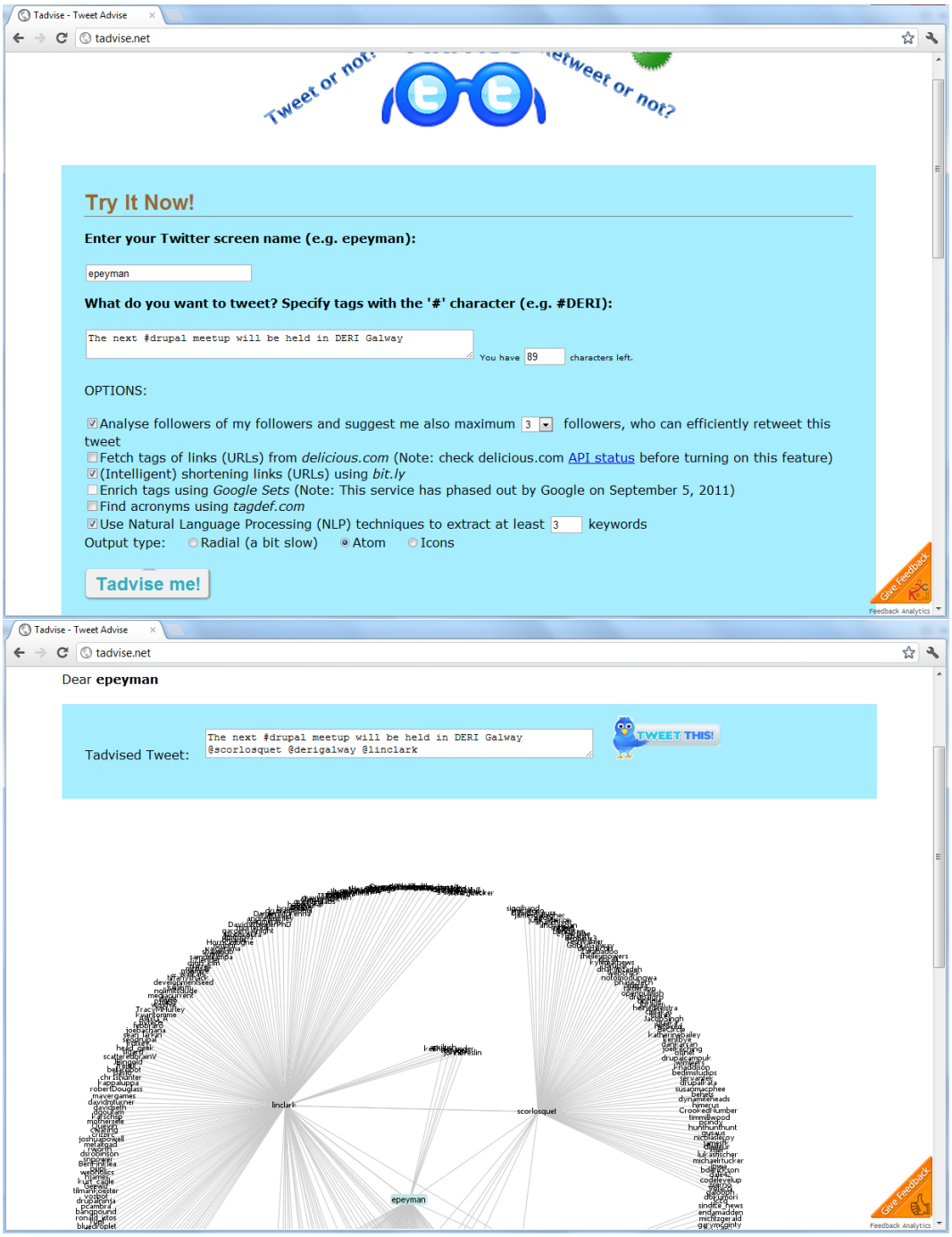

Figure 6.3: Snapshots of Tadvise user interface. Users enter their Twitter screen names and a tweet (upper pane). They have also several available options to tune the advice parameters. After clicking on "Tadvise me!" button, Tadvise will analyse the inputs and users will be redirected to a result page (lower pane) which composes of manipulated or so-called *tadvised* tweet, as recommended hubs are automatically added to the tweet. In the result pane, users can also see text-based as well as graphical explanations. Users have also the option to tweet the *tadvised* message directly from the Tadvise interface.

(a) (b) (c)

Figure 6.4: More intuitive icons for replacing the traffic lights icons. The green light was replaced with Figure 6.4(a). The amber light was replaced with Figure 6.4(b). The red light was replaced with Figure 6.4(c).

with each follower. Each API call returns 20 lists membership of a user. We put a limit (i.e., 300) on the number of Twitter lists associated with a user that we crawl, as 300 tags are reasonably enough for building a high-quality user profile for our purpose. Note that we were not interested in crawling the statuses, but rather the social networks of a seed and the Twitter lists.

6.5.2 Tadvise User Profile Builder

While Twitter does not have a rating system for tweets, it does enable users to create a list of favorite tweets, a feature that is similar to bookmarking a link in a Web browser. However, our analysis on 20 random users has shown that the bookmarking feature is not yet heavily used and is not yet a useful source of data for building a user profile.

As stated, our approach is to use Twitter lists as a crowd-sourcing method for building user profile metadata. In short, each user profile is composed of metadata extracted from the lists (tags) associated with the user by other users. In order to build a weighted user profile, we need to rank the tags that have been associated with a user (i.e., rank the result of $lists(u_i)$.) We do this by ranking the users who assigned the tags. There have been several studies of user ranking on Twitter [Java et al., 2007, Weng et al., 2010, Kwak et al., 2010] with no one technique demonstrating superiority. As such we make use of [Kwak et al., 2010]'s finding that a simple in-degree measure behaves similarly to PageRank on the Twitter network (see Equation 6.1). As Twitter is an open platform and the connections are not necessarily reciprocal and does not require confirmation of the followee-side for public accounts, we do not consider the outgoing links (i.e., followees) for ranking purposes.

$$rank(u_i) = \log(|U_i^{fo}|) \tag{6.1}$$

Note that our ranking method can be generalised to a recursive one (see Equation 6.2). In brief, users who have more high-ranked followers, have higher ranks.

$$rank(u_i) = \sum_{u_j \in U_i^{fo}} rank(u_j) \tag{6.2}$$

$$weight(t_k, u_j) = \sum_{(u_i, t_k) \in lists(u_j)} rank(u_i) \tag{6.3}$$

ID (masked)	#Lists	#Unique lists	#Tweets	#Enriched tags
epe...	13	8	197	202
con..	4	2	46	97
eco...	8	8	12	33
jac...	8	6	25	117
cya...	2	2	230	107

Table 6.1: Result of enriching user profile tags set (i.e., Twitter lists) of five sample users by analysing URLs in their tweets.

Comparison between top-ranked users between our ranking results and the list at [Java et al., 2007] shows that both lists are coherent[18].

The weight of a particular Twitter list for a target user profile is calculated by summing up the ranking of people who assigned that list description to the target person (see Equation 6.3).

As Twitter lists consist of arbitrary phrases, we use the Porter stemming algorithm [Porter, 1980] to stem the Twitter lists for each user profile. For tags that comprise of more than one term, we use the stemmer for each term.

User Profile Sparsity

The number of lists that are assigned to a user could be small and the list names could be repetitive. On the other hand, tweets that are sent on Twitter are originated from a wide range of topics. Thus, it is possible that we may not be able to find appropriate hubs for a given tweet due to user profile sparsity. One possibility to overcome this drawback is to enrich user profiles by extracting URLs from someone's tweet and then fetching the tags associated with the URLs from online bookmarking services like *delicious.com*. Table 6.1 shows an overview of this approach on five sample users[19]. The first column in the table shows masked Twitter IDs of five different users. The second column is the number of Twitter lists that were assigned to each user. The third column shows number of unique Twitter lists (i.e., non-repetitive) assigned to each user. The fourth column shows the total number of tweets for each user. Finally, the last column shows the total number of unique tags – fetched from *delicious.com* – after analysing URLs in each user's tweets. As supplementary information, Table 6.2 shows an overview of unique Twitter lists of users (third column of Table 6.1) plus top-10 unstemmed enriched tags (fifth column of Table 6.1).

Besides online bookmarking services like *delicious.com*, we also envisioned to use online directories like *http://www.dmoz.org/* to get the main categories of the URLs. Our observation showed that such directories are not updated frequently and thus many links that are shared in Twitter do not have a category.

[18]Our 15 top-ranked users include: google, twitter, BarackObama, nytimes, algore, cnnbrk, johncmayer, TheOnion, iamdiddy, britneyspears, ashleytisdale, aplusk, THE_REAL_SHAQ, TheEllenShow, and RyanSeacrest.

[19]Crawling date: September 2, 2011.

ID (masked)	Unique lists	Top-10 enriched tags (sorted by frequency)
epe...	unfollowed-relisted, derians, deri, derimembers, my-fun-time, research, semweb, ecospace-related	networking, social, business, community, career, web2.0, jobs, network, socialnetworking, api
con..	deri, derimembers	news, science, magazine, technology, politics, magazines, research, culture, media, tech
eco...	innovation, astro, projectmgmt, ecospace-related, collaboration, fit-projects, fit-cscw, fit	video, youtube, videos, entertainment, media, web2.0, social, fun, music, community, xml
jac...	wikimedia, venture-capital, visual-art, hi-tech-eng, deri, business	technology, blog, video, gadgets, news, youtube, tech, videos, web2.0, blogs
cya...	design-creativity, blogging-social-networking	photos, flickr, photography, news, web2.0, photo, sharing, video, blog, technology

Table 6.2: Unique Twitter lists along with top-10 unstemmed tags fetched from *delicious.com* by analysing URLs in users' tweets.

6.5.3 Tadvise Advice Engine

Given a tweet and a user u_i, first we extract tags from the tweet. On Twitter platform, twitterers use the hashtags to specify tags in a tweet (e.g., #drupal). For a fairly detailed overview on hashtags and its differences with common tagging within most Web 2.0 platforms see [Huang et al., 2010]. We extract such tags from the tweet and enrich them using Google Sets[20]. Our analysis suggests that Google Sets provide more contextually relevant suggestions than lexical databases such as WordNet [Miller, 1995]. We also analyse the URLs within a tweet. Using regular expressions, we extract HTTP and FTP URLs from a tweet. Then we use the *delicious.com* API[21] to retrieve the tags associated with each link. We do not enrich the tags extracted from *delicious.com*, as *delicious.com* recommends already sufficient tags for a given URL. We then merge the tags from *delicious.com* and Google Sets.

It is common that users use acronyms as hashtags in a tweet. For example, they may use *#sn* to represent the *social network* phrase or *#semweb* as a shorter form of the *semantic web* phrase. Obviously, understanding acronyms or hashtag summarisations could help us to provide better recommendations. In order to understand meaning of a hashtag, we used an online popular service offered by *http://www.tagdef.com* to analyse hashtags and get real meaning of the tags. The main advantage of this service is that in a short time (normally with no required registration) users can define new acronyms such as tags representing events (e.g., conferences) that are used in Twitter. Such definitions become *live* and can be accessed immediately. The other advantage of this service is its built-in voting mechanism for filtering irrelevant definitions or acronyms. As we use this public service for analysing acronyms and hashtags, it is already sufficient, if only one user provides more information about a specific hashtag on *http://www.tagdef.com*. Browsing the repository of *http://www.tagdef.com* showed

[20] http://labs.google.com/sets – Note: This service has phased out by Google on September 5, 2011.
[21] http://delicious.com/developers

that most commonly-used acronyms such as *#sn*, *#semweb*, etc. were already defined by users on this site.

Generally, depending on hashtags to identify key topics of a tweet has one major drawback. It is possible that some users never use hashtags in their tweets and thus Tadvise may fail to recommend appropriate hubs for tweets that contain no hashtags. To address this drawback, we used a third-party NLP service [QasemiZadeh et al., 2012] that was designed to extract keywords or key phrases from short texts like tweets. For each extracted keyword or key phrase, the NLP package is capable of returning a confidence value, a metric that indicates to what extent the extracted keyword or key phrase could be a potentially relevant item. We used confidence values to get top-k relevant keywords (k is end-user defined via Tadvise interface). Besides extracting hashtags, using this NLP package helps us to understand important keywords or key phrases in a tweet and then enrich the keywords. This could give us new metadata for recommending appropriate hubs.

We used Algorithm 4 to build *followersProfile1()* and *followersProfile2()* of user u_i. In order to find a knee point between two partitions in the algorithm, we applied the k-means clustering algorithm with k=2.

As tweets are 140-characters in length, we extended Algorithm 5 to consider the length of screen name of a hub for making recommendations. That means when two hubs disclose a tweet further with n users, we choose the hub who has shorter length of screen name. We add the recommended candidates automatically to the tweet by mentioning their screen names after the '@' sign and enable the seed to tweet it directly from the Tadvise interface. Note that we do not explicitly ask the hubs to retweet a tweet, but our goal is to attract their attention. A brief literature review that supports this hypothesis follows: The use of @ sign for addressing users in Twittersphere was originated from IRC practices [Boyd et al., 2010]. [Ehrlich and Shami, 2010] state that "the use of @ sign to direct a post to someone may improve communication by initiating a brief dialog". [Honeycutt and Herring, 2009] studied functions of @ sign in Twittersphere using several random samples of tweets. They found that roughly 30% of tweets contain @ sign. They speculated that the use of Twitter for collaboration (besides conversation) is rapidly increasing. Moreover, their analysis showed that more than 90% of tweets with @ sign was used for addressivity (addressing an individual). More interestingly, they claimed that "a conservative measure of public responses posted within a one-hour period indicated that about 31% of tweets with @ [sign] received a response".

Tadvise Explanations

Tadvise provides simple text-based as well as graphical explanations. Our explanations originate from the processes that we use for providing advice. In other words, we show the list of potentially interested Twitter users at distances of one and two from a seed and also justify how our recommended hubs can propagate a tweet further in the network. We also present a ranked list of potentially interested Twitter users at a distance of two from the seed who can not receive the tweet via the recommended hubs. The seed can freely add such (top-ranked) Twitter users to the tweet (i.e., direct message), in order to attract their attention.

Unlike blog posts, micro-blog posts are short in size and users might need fast explanations to see what happens behind Tadvise recommendations. Reading long textual explanations is a bit time-consuming for users. It is often said that *a picture is worth a thousand words*. In other words, it is a well-accepted hypothesis that images are better for representing spatial structures, locations, and details, whereas words are better for representing procedural infor-

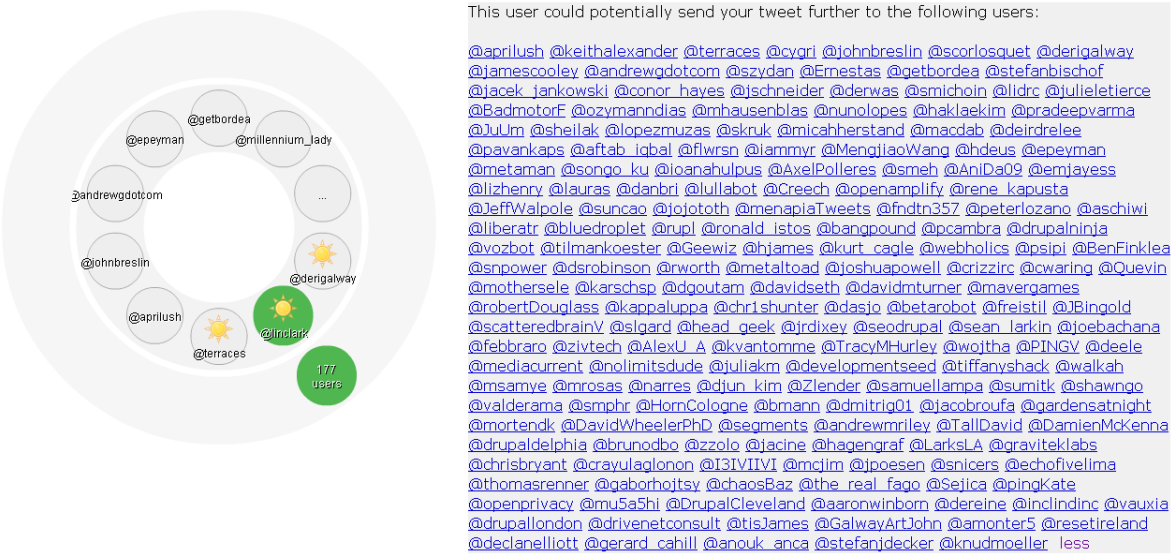

Figure 6.5: Interactive atom-based interface of Tadvise explanations. By clicking a Twitter ID that has a hub indicator (i.e., the sun icon) in the left pane, Twitter IDs of those users who receive the tweet via the hub will be demonstrated in the right pane.

mation and abstract verbal concepts [Ware, 2004]. To this end, we equipped our text-based explanations with three interactive visual explanations as follows:

`Graph-based visualisation`: This visualisation technique presents Tadvise explanations in an interactive graph-like interface. By hovering the mouse icon on Twitter IDs, a user can see who may propagate a tweet further to whom, as the colour of Twitter IDs who may receive the tweet will change. By looking at this visualisation, a user can grasp immediately how well-connected hubs may propagate a tweet further in the network.

`Radial-based visualisation`: This visualisation technique is similar to graph-based visualisation with the difference that it visualises the tweet propagation path in a more clear radial layout. This interface can help users for grasping an immediate perception of how many users at different layers (e.g., followers of followers) can get a tweet and who the most potentially interested and well-connected followers are. This visualisation is also interactive. By clicking a Twitter ID, the ID moves to the centre of the radial interface.

`Atom-based visualisation`: This visualisation technique is similar to radial-based visualisation with the difference that it loads faster than the other two, as its technology is based on Adobe Flash and not Java applet. The other difference between atom-based interface and the other two interfaces is that in atom-based interface, a limited number of users at the first layer (i.e., direct followers) will be shown to end users. This is due to the space limitations of atom layouts, however, we ensure that all recommended hubs are included in the first layer. The hubs have an indicator (i.e., the sun icon) to distinguish them with other users. This visualisation is also interactive. By clicking a Twitter ID that has a hub indicator, the ID will be highlighted and end users can get a list of Twitter IDs at a distance of two who receive the tweet via the highlighted hub. Figure 6.5 demonstrates a snapshot of this visualisation. For more information on the atom interface see [Samp and Decker, 2010].

We used Prefuse[22], a set of open source software tools for creating interactive data visualisations, for making graph-based and radial-based visual explanations. End users can choose between above visualisation techniques via Tadvise user interface. Our evaluation shows that our explanations are convincing for end users.

6.6 Evaluation and User Study

We evaluated the following main hypothesis: our policy advisor reduces information overload and information shortage by spotting well-connected topic-sensitive hubs who may propagate community-related and even non-community-related information further in the network. However, we were also interested to study two more issues/aspects: a) if people-tags on social media (e.g., Twitter) assist users to know each other better, and b) if building enriched people-tag-based profiles helps users to know their social networks (as a whole) better by identifying their communities, interests, expertise, etc. The latter is important, as this can help users to boost communication and collaboration opportunities and may encourage users to propagate community-related information more often.

Our focus is not to force well-connected topic-sensitive hubs to propagate a message, but rather to attract their attentions. The propagation step is an optional step for the recommended hubs, however, as stated in Section 6.5.3 [Honeycutt and Herring, 2009] mention that in Twittersphere (i.e., our evaluation testbed) about 31% of tweets with @ [sign] received a response within a one-hour period.

6.6.1 Experiment – Design

[Schafer et al., 2007] mention that there is no global well-accepted metric that can evaluate important criteria for recommender systems. In order to provide support for our hypothesis and study the two issues that we mentioned in Section 6.6, we designed a survey that was personalised for each participant. For the survey design we studied the design recommendations of [Shneiderman and Plaisant, 2004] and the well-known Questionnaire for User Interaction Satisfaction[23] (QUIS). The survey had five main steps with a number of questions in each step. Most of questions in the survey had five possible replies: strongly agree, agree, neutral, disagree, and strongly disagree.

`Step 1: General Questions` – In the first step, the goal was to study: a) Whether participants agree with the Twitter lists assigned to them, b) Whether the lists that were assigned to them fall into certain categories, and c) Whether the lists they assign(ed) to others fall into certain categories.

The aforementioned categories refer to common people-tagging categories discovered in a large-scale analysis of tagging behaviour [Muller et al., 2007]. They are as follows: *Characteristic* (e.g., friendly, cool), *Interest and Hobby*, *Affiliation* (e.g., IBM), *Working Group*, *Location*, *Name* (e.g., Peter, Mary), *Project*, *Role* (e.g., boss), *Skill and Expertise*, *Sport*, and *Technology* (e.g., drupal, semantic-web).

`Step 2: Usefulness of Twitter Lists/People-Tags` – Steps 2–4 were presented in a game-like fashion with the participant having to guess or choose from a set of answers. Each

[22] http://prefuse.org/
[23] http://lap.umd.edu/quis/

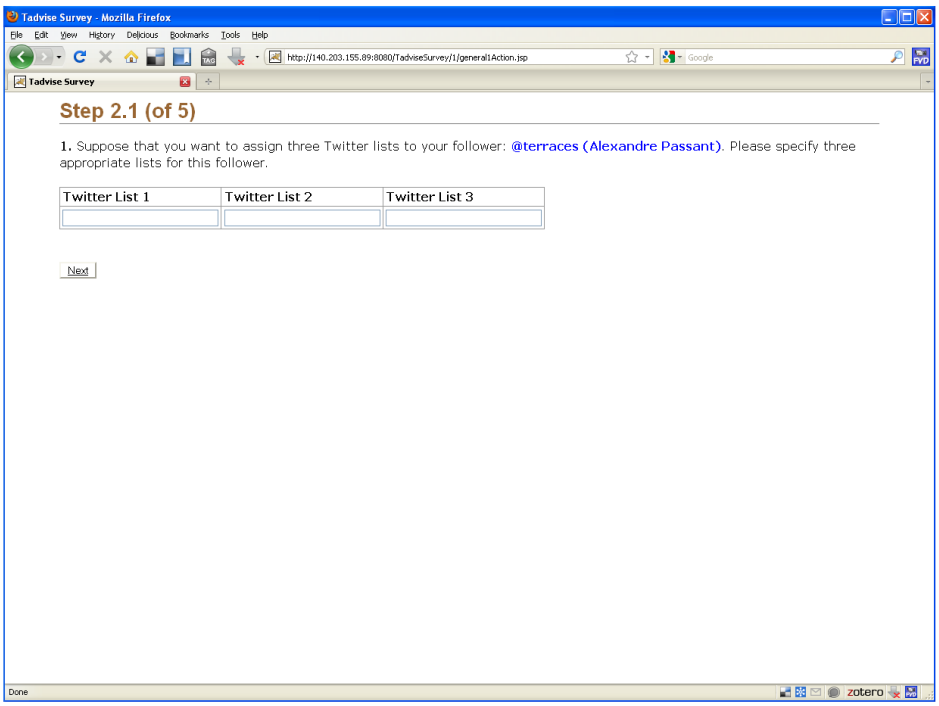

Figure 6.6: Step 2 of the survey: asking a participant to assign three Twitter lists to one of his/her follower who was chosen randomly.

step had four sub-steps. In Step 2, we collected data on the usefulness of Twitter lists. For the first three sub-steps of Step 2, we picked one random follower who had been assigned to at least three Twitter lists by any user and was also a followee of the participant, Then, we asked the participant to assign three Twitter lists to the follower (see Figure 6.6). After the participant clicked the submit button, we fetched the real Twitter lists assigned to the follower and asked the participant whether the result was useful in knowing the follower better (see Figure 6.7). In Sub-step 2.4, we focused on the community of the participant and asked him/her to guess three Twitter lists that fit the majority of his/her followers. After submitting the result, we showed our analysis result (i.e., all Twitter lists of first partition of the $followersProfile1(participant)$) to the participant and asked, if it helps to know the community of his/her followers better.

`Step 3: Knowledge of Followers` – Step 3 of the survey measured how well participants know their followers. In each sub-step, we showed a random Twitter list (fetched from $followersProfile1(participant)$) to the participant and asked two questions: 1) Approximate percentage of the followers who were assigned to that list, and 2) The followers (from twenty random followers) who were assigned to that Twitter list (see Figure 6.8).

In Sub-steps 3.1 and 3.2, we picked a random Twitter list from the first partition of the $followersProfile1(participant)$ and ensured that at least 50% (if possible) of the 20 random followers are correct answers. In Sub-steps 3.3 and 3.4, we picked a random Twitter list from the second partition. We enabled the participants to skip a Twitter list (maximum three

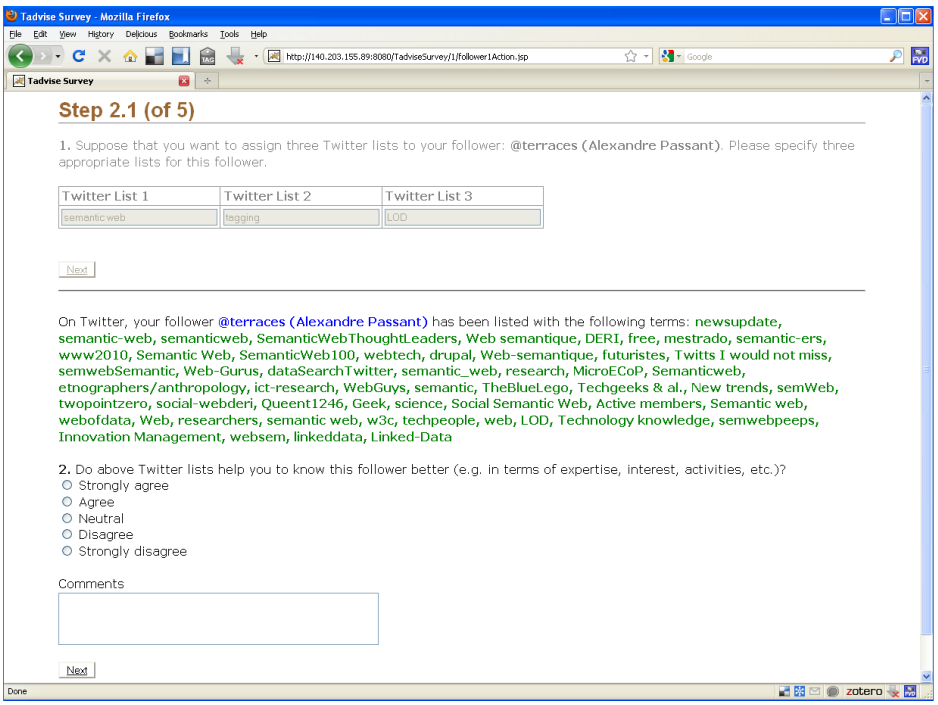

Figure 6.7: Step 2 of the survey: retrieving real Twitter lists of the random follower and asking the participant if the retrieved lists are informative.

times in each sub-step), if they could not understand its meaning. In order to prevent the participants selecting all followers, we put a maximum limit on the number of followers that could be selected. After submitting the result, we showed correct percentages and the missing followers from the list and asked the participants whether this information helped in knowing their followers/community better (see Figure 6.9).

`Step 4: Usefulness of Recommendations` – In Step 4, we investigated whether participants found Tadvise recommendations of well-connected followers to be useful. This step had four sub-steps. In Sub-steps 4.1 and 4.2, we showed a random Twitter list as a topic from the first partition of the $followersProfile1(participant)$ and asked the participant to select two well-connected followers who could propagate a tweet about the topic to a broader audience. We enabled the participants to select two followers from drop-down boxes, each containing twenty random followers, two of which were the correct answers (see Figure 6.10). For Sub-steps 4.3 and 4.4, we carried out the same experiment, but with the Twitter lists from the second partition. After submitting the result, we presented the participants with our recommended hubs and provided explanations to justify our recommendations. Participants were asked whether they were sufficiently convinced to use the recommendations (see Figure 6.11).

`Step 5: General Questions` – In the final step, we asked participants several general questions. Among others, we asked the participants if they would find it useful to receive advice on whether their followers may be interested in a particular tweet. We also asked the partici-

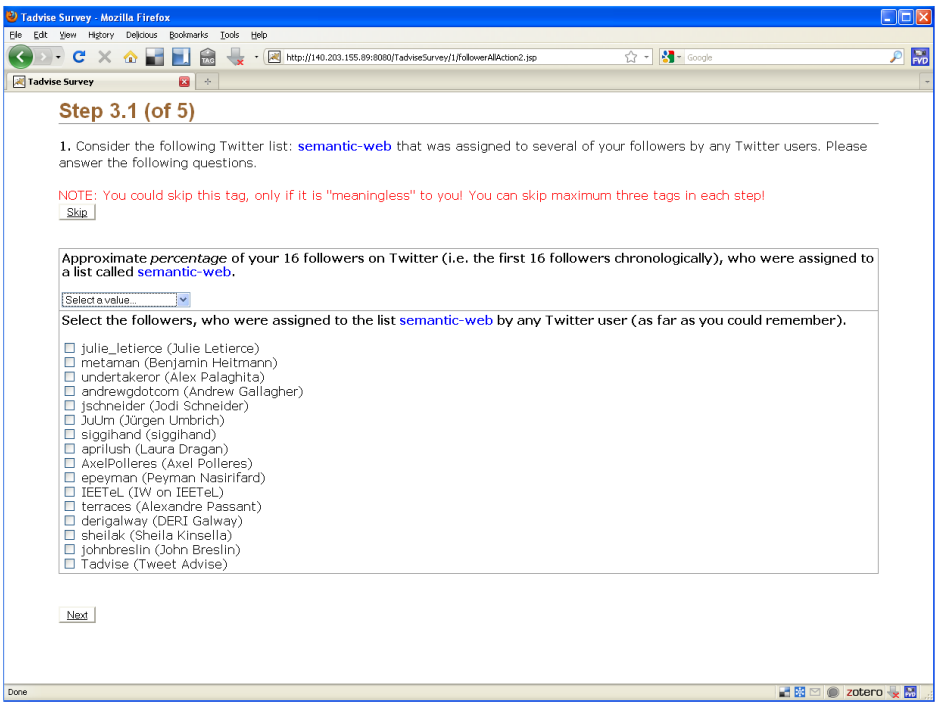

Figure 6.8: Step 3 of the survey: showing a random Twitter list to the participant and asking him/her two questions: 1) Approximate percentage of the followers who were assigned to that list, and 2) Select the followers who were assigned to that Twitter list.

pants if they would find it useful to receive advice about the most effective and well-connected hubs.

We used word-of-mouth to propagate the Tadvise system and the survey. We asked people to refer their friends for participating in our survey and we gave a small prize to the person who referred the most friends.

Twitter users have on average 120-130 followers[24]. We wanted to evaluate three groups of users: 1) Users who are not being followed by many people, 2) Users who are followed by an average number of followers, and 3) Users who are community hubs and are followed by thousands of followers. Thus, we ensured that we had evaluators from all these three main groups.

6.6.2 Experiment – Result

Participants Overview

We made personalised online surveys for 112 Twitter candidates, among them 11 candidates did not fulfill our requirements for the survey – Each participant had to have at least three followers that were assigned to at least three Twitter lists, and who were also followees of

[24] http://www.guardian.co.uk/technology/blog/2009/jun/29/twitter-users-average-api-traffic

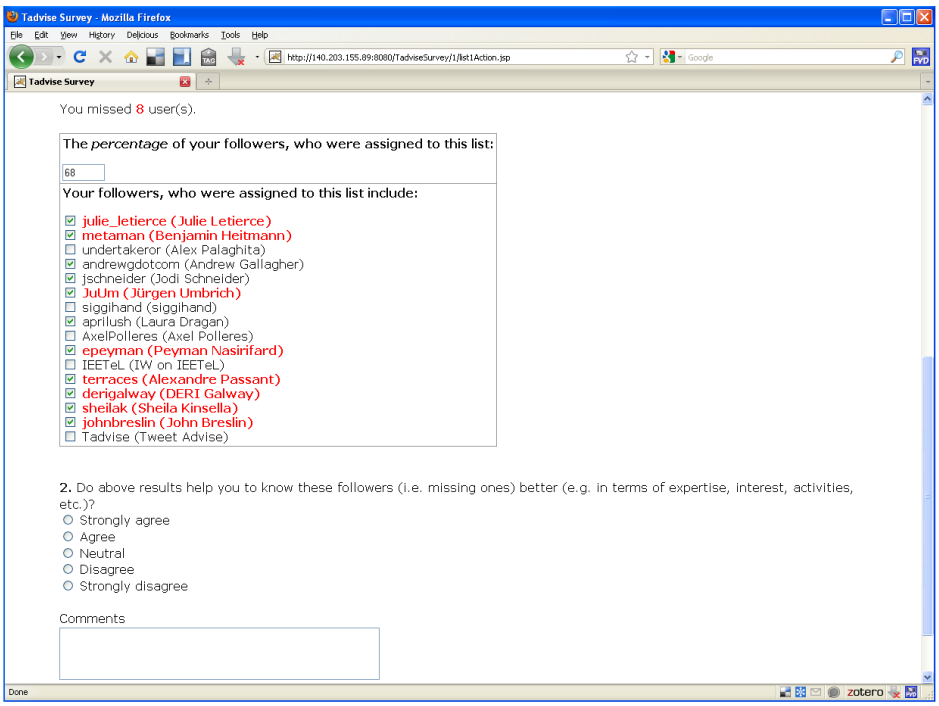

Figure 6.9: Step 3 of the survey: showing calculated percentage of the followers who were assigned to that random Twitter list along with those followers in red colour who were not selected by the participant. We asked the participant if the results are informative.

the participant (i.e., reciprocal link). The survey was online for four weeks and we asked all 101 eligible candidates via email, instant messaging or direct tweet to participate in our survey. In total, 76 eligible candidates participated in our survey, among them 66 participants completed the survey. 47% of participants who completed the survey (i.e., 31 participants) had 100 or more followers, among them twelve participants had more than 500 followers. Four participants had 1000 or more followers. 53% of participants who completed the survey (i.e., 35 participants) were members of our institute with different backgrounds. Generally, our participants were from different domains such as computers and IT, marketing, chemistry, math, physics, economy, etc. and with different job titles like students, researchers, marketers, technicians, professors, community fundraisers, etc. Several participants mentioned that they use Twitter mainly in conferences and scientific events for broadcasting key points of the interesting talks and papers. In order to encourage the candidates to participate in our study, we made a small donation to a national charity on behalf of each participant.

Our participants were a mixture of active and non-active Twitter users. 27.7% of participants mentioned that they check their Twitter accounts once per hour, while 83.1% of participants mentioned that they do it at least once per week. 3% of participants mentioned that they tweet at least once per hour, while 64.7% of participants mentioned that they tweet at least once per week.

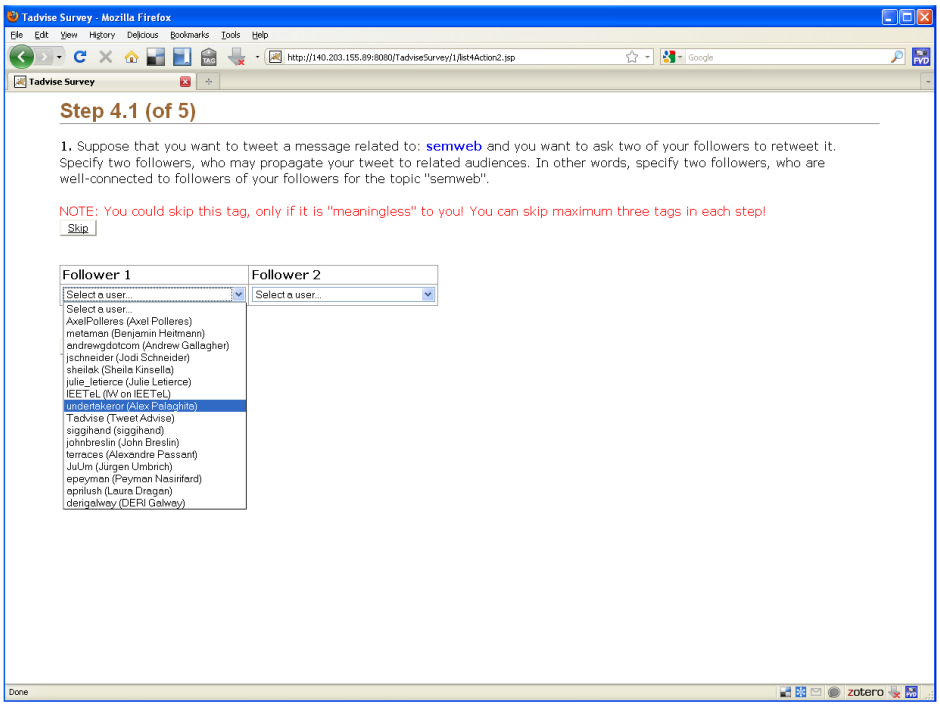

Figure 6.10: Step 4 of the survey: showing a random topic to the participant and asking him/her to select two followers who are well-connected hubs in relation to that topic. The participant had to select his/her suggestions from drop-down boxes, each containing twenty random followers, two of which were the correct answers.

Results

The results show that 79.1% of participants who were assigned to one or more Twitter lists mentioned that Twitter lists represent them correctly. Only 1.6% of participants claimed that they were incorrectly assigned to a list (see Section 6.7 for their arguments). Whether assigning lists or being assigned to lists, participants indicated that 96% of the lists came from the following categories: Affiliation: 24.3%, Technology: 14.6%, Interest and Hobby: 15.9%, Skill and Expertise: 13.8%, Working Group: 9.2%, Location: 8.4%, Characteristic: 6.3%, Project: 3.8%, Role: 1.7%, Name: 1.3%, and Sport: 0.8%.

We used the results of Sub-steps 2.1, 2.2, 2.3, 3.3 and 3.4 to see if people-tags (i.e., Twitter lists in this study) assist users to know each other better. We further used Sub-steps 2.4, 3.1, and 3.2 to see if building enriched people-tag-based profiles helps users to know their followers (as a whole) better by identifying their communities, interests, expertise, etc. We then used Sub-steps 4.1, 4.2, 4.3, and 4.4 for measuring the usability of our main hypothesis (see also Section 6.6.3 for statistical measurement). As Figure 6.12(a) illustrates, 58.1% of participants agreed that people-tags (i.e., Twitter lists) assist them to know individuals (e.g., their followers) better (67.7% expressed agreement in Sub-steps 2.1, 2.2 and 2.3, whereas 43.5% expressed agreement in Sub-steps 3.3 and 3.4), while 18.6% disagreed (8.1% expressed disagreement in Sub-steps 2.1, 2.2 and 2.3, whereas 34.4% expressed disagreement in Sub-steps 3.3 and

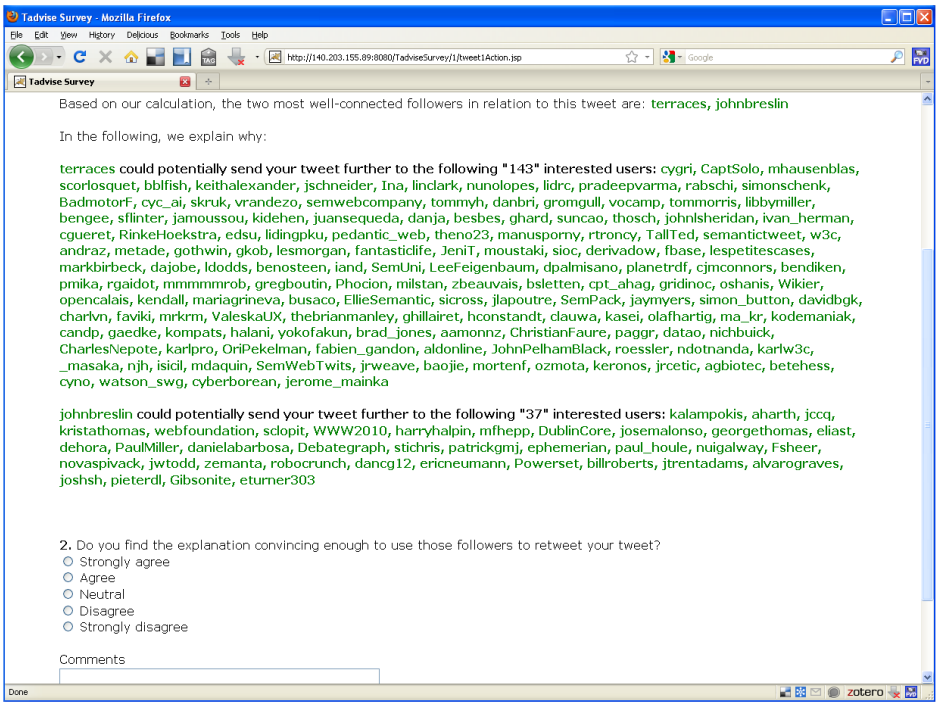

Figure 6.11: Step 4 of the survey: showing to the participant the two recommended followers who are potential topic-sensitive hubs along with explaining why these two followers are well-connected hubs. We asked the participant if the provided explanations are convincing.

3.4) (see also Section 6.7 for their justifications). Figure 6.12(b) illustrates that 57.4% of participants agreed that building enriched people-tag-based profiles helps them to know their followers/communities better, whereas 17.3% disagreed (see also Section 6.7). Figures 6.12(c) and 6.12(d) are related to measuring usability of our main hypothesis. Figure 6.12(c) shows the result of Sub-steps 4.1 and 4.2. In total, 72% of participants found our explanations convincing enough to use the recommended hubs for propagating their community-related information (e.g., tweets), whereas 13.7% disagreed. Finally, Figure 6.12(d) shows the result of Sub-steps 4.3 and 4.4. The result shows that 49.3% of participants found our recommendations for propagating non-community-related information convincing, whereas 18.2% disagreed (see also Section 6.7).

In Step 5, 48.4% of participants were positive about being advised, if a tweet is interesting for majority of the followers, whereas 28.1% of participants were negative. The rest (23.5%) selected the *Undecided* option. In relation to this case, participants mentioned interesting comments and arguments which will be discussed in Section 6.7. In total, 78.1% of participants were positive about being recommended hubs that could effectively propagate a topic, whereas only 7.8% of participants found it not useful. The rest (14.1%) selected the *Undecided* option.

In brief, the analysis suggests that participants were mainly interested in being recommended hubs that can effectively retweet their messages and they found our recommendations for (mainly) community-related tweets useful and convincing.

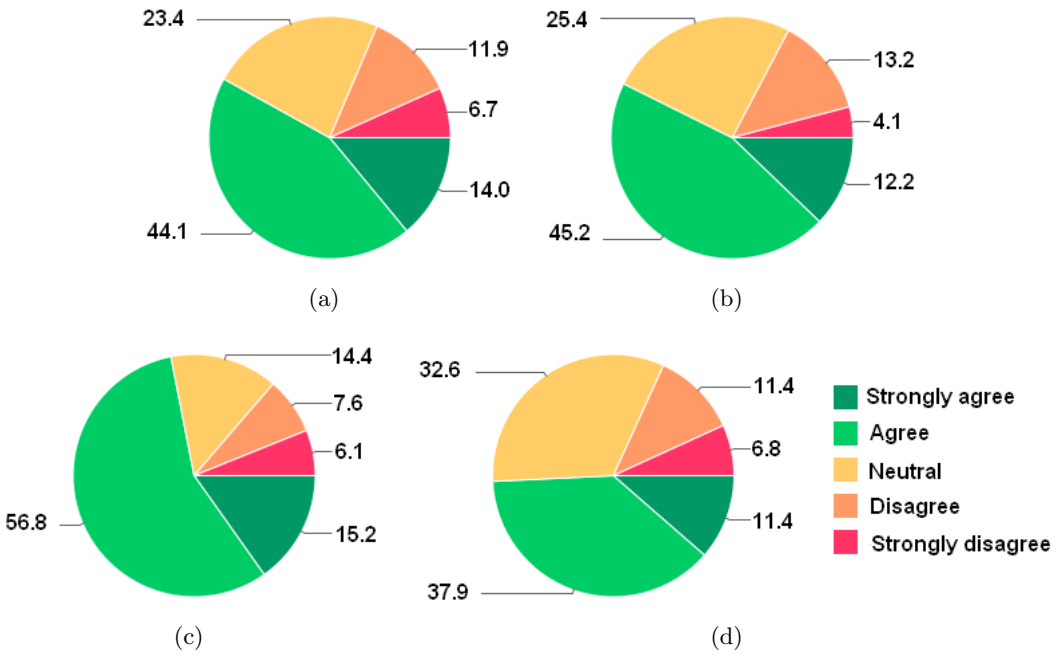

Figure 6.12: Figure (a) demonstrates that 58.1% of participants agreed that people-tags (i.e., Twitter lists) assist them to know individuals (e.g., their followers) better, whereas 18.6% disagreed; Figure (b) demonstrates that 57.4% of participants agreed that building enriched people-tag-based profiles helps them to know their followers/communities better, whereas 17.3% disagreed; Figures (c) and (d) are related to usability of our main hypothesis: 72% of participants found our recommendations and explanations for propagating *community-related* topics convincing, whereas 13.7% disagreed (see Figure (c)); moreover, 49.3% of participants found our recommendations and explanations for propagating *non-community-related* topics convincing, whereas 18.2% disagreed (see Figure (d)).

6.6.3 Measuring Information Overload and Information Shortage

In order to measure how our policy advisor can reduce both information overload and information shortage within a network of people-tag-based profiles, we further compared our participants' results with our recommendations using a pairwise t-test[25]. In particular, we measured two different parameters: 1) How many *interested* users at a distance of two from a user will get a tweet related to a topic through our recommended hubs versus given participants' suggestions, and 2) What the flooding ratios of our hub recommendations versus participants' suggestions are. The flooding ratio is calculated as follows: Initially, we count x: the total number of users at a distance of two who receive a tweet (regardless of their interests). Then, we count y: the total number of users who are at a distance of two and have interests in the topic (i.e., were tagged with the same or similar topics). Then the flooding ratio can be calculated as x divided by y. We studied the two above parameters using community-oriented as well as non-community-oriented topics.

Figure 6.13 compares our recommendations with the participants' suggestions for community-related topics. A pairwise t-test showed that our policy advisor approach can propagate

[25]These measurements were performed with a new crawl of the Twitter network which was performed approximately one month after the evaluation.

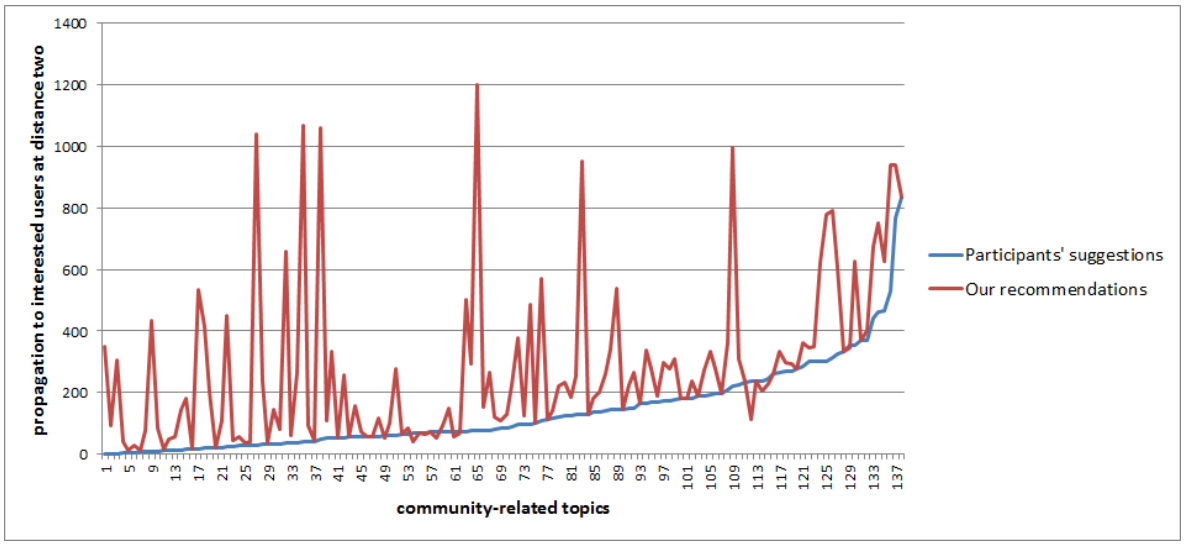

Figure 6.13: Propagating community-related topics to interested users at a distance of two from a seed. A pairwise t-test showed that our approach can send a topic-related message to more interested users and at the same time keeping the flooding ratios to a minimum.

information to more interested users at a distance of two (p-value = 0.0). A pairwise t-test also showed that the flooding ratios of the participants' suggestions are higher than the flooding ratios of our recommendations and this result is also statistically significant (p-value = 0.029).

Figure 6.14 compares our hub recommendations with the participants' suggestions for non-community-related topics. A pairwise t-test showed that like community-related topics, our approach can send tweets to more interested users at a distance of two from a user (p-value = 0.0). A pairwise t-test also showed that the flooding ratios of the participants' choices are higher than the flooding ratios of our recommendations (p-value=0.003). Despite the fact that our participants mainly preferred our recommendations for propagating community-oriented topics (see the previous section), our measurements showed that even for non-community-oriented topics, we can recommend more relevant and well-connected hubs for propagating a topic.

In brief, our analysis suggests that our recommendations can forward a topic to a larger relevant audience (i.e., to address information shortage) and at the same time with reduced flooding ratios (i.e., information overload), compared to participants' suggestions.

6.7 Discussion and Analysing Comments

Each question in the survey had an optional *comment* box that enabled participants to provide comments. Through comment boxes, we received lots of valuable feedback reflecting the behaviour of Twitter users in relation to Twitter lists and retweeting. These comments provide design implications for recommender systems that are built on top of Twitter lists and in general for systems used to tag and group people. In the following section, we have an overview on a selective list of the comments that we received. Based on content of the comments, we categorised them into several groups. For privacy purposes, we masked real

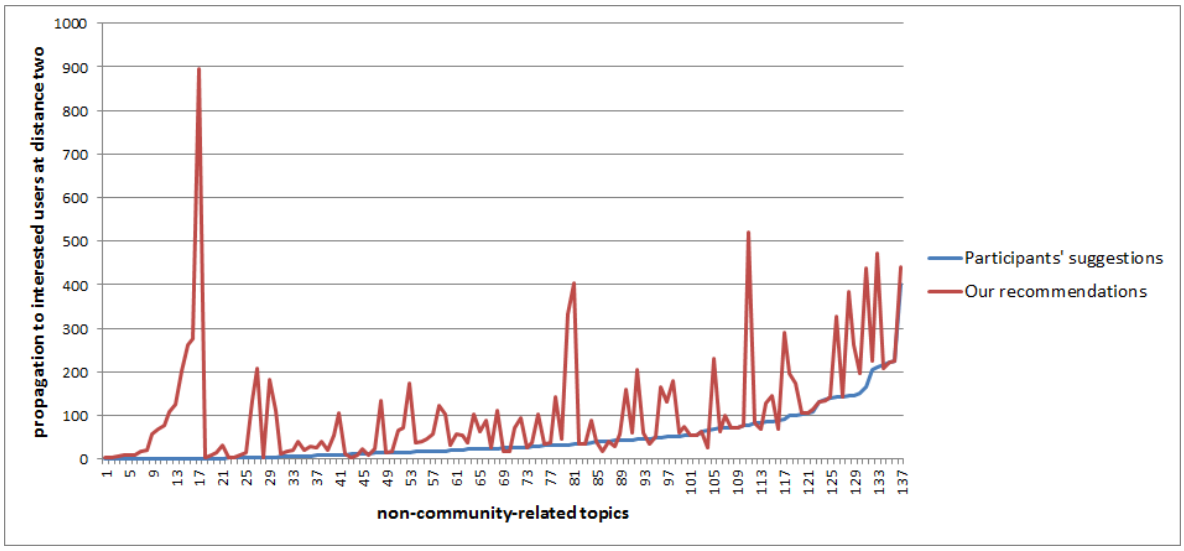

Figure 6.14: Propagating non-community-related topics to interested users at a distance of two from a seed. A pairwise t-test showed that even for non-community-related topics our approach can send a topic-related message to more interested users and at the same time keeping the flooding ratios to a minimum.

Twitter screen name of users who gave the comments with *@user*.

6.7.1 Ambiguity of Twitter Lists

Several participants mentioned the ambiguity of Twitter lists/tags. As a result, the participants could not assign such tags (e.g., media-basket, SAURABH JOSHI, colegas, gender, methodology, conference lists, In-an-artwork-yes, means of association) to any aforementioned categories (i.e., affiliation, technology, etc.). The reasons for the ambiguity can be categorised as follows:

- `Ambiguity due to a lack of meaning`: Some tags were meaningless or did not have a clear meaning from participants' point of view: *@user: I don't understand "media-basket" term – I write something about basketball sometimes, but I don't know if this term has this meaning.*

- `Ambiguity due to auto-generated tags`: There exist several third-party applications (e.g., MrTweet, Twibes) that assist users to find new followees or help users categorise their followees by automatically building groups of like-minded users using Twitter lists. As a result, such tags could not be categorised: *@user: All fine, but what is "conversationlist"?*

- `Ambiguity due to subjective tags`: Some Twitter lists get their meanings in relation to someone or something else. For example, a Twitter list like *friend* or *People I've met* gets its meaning in relation to the list creator. This issue was also reflected by several participants: *@user: Yes - Sustainable behaviour and behavioural economics were particularly useful. Misc, vvreading, tweeps I've met, etc weren't.*

6.7.2 Validity Period of Twitter Lists

Some Twitter lists (e.g., related to affiliation) may lose their validity after a while: *@user: Additionally, we've got here time and continuity dilemma: I'm not in DERI anymore and 2/3 of the terms are about DERI.* As a Twitter list is created by a single user, updating the list should be also handled by that user. This may be neglected by some users and thus making validity-related issues. To address this drawback, one participant mentioned that creating Twitter lists could be a collaborative practice and users could collaborate in order to enrich a Twitter list rather than creating their own list from the scratch. This could be a good practice, however, there is a need for moderation in order to update a Twitter list.

6.7.3 Incomprehensiveness of Twitter Lists

Many participants complained that Twitter lists associated with themselves or with their (mutual) friends were not comprehensive: *@user: Still much a lot to describe this follower.* As Twitter lists is a new feature by Twitter, we envision that this feature will evolve after some time. We also speculate that unlike tagging people which aims to define a single person in a more fine-grained fashion, Twitter lists aim to group more than one Twitter user, hence Twitter lists are usually coarse-grained and aim to group like-minded people (e.g., in terms of affiliation, interest, expertise). As stated in Section 6.5.2, to address this issue and for building user profiles we enrich Twitter lists by extracting tags from someone's tweets. Another possibility would be to enable users to update or vote for suitable tags associated with themselves or (mutual) friends.

6.7.4 Redundancy of Twitter Lists

As we showed Twitter lists without any pre-processing (e.g., taking into account case-sensitivity, redundancy, etc.) to the participants, several participants reflected this issue in their comments: *@user: There's a lot of redundancy.* However, redundant tags were also perceived by some participants as `important` tags: *@user: Connection with local people is their real strong point, being on so many "Northern Ireland" lists confirms this.* As explained, we use this fact (i.e., redundancy) to rank the tags.

6.7.5 (Semantic) Linking of Twitter Lists

Many users use a shorter version of a word or phrase for creating their Twitter lists. For example, they may use "tweeps", "ppl" or "peopl" to refer to the same concept (i.e., people). Several participants mentioned the importance of linking the tags – in order to unify user profiles: *@user: too many different tags, need more synthesis.*

6.7.6 Remembering Mutual (Twitter) Friends

Participants who had lots of followers found it more difficult to recognise their (mutual) Twitter friends than participants who did not have many followers. In relation to survey questions regarding to knowing followers, a participant with 669 followers said: *@user: I don't remember why I follow him, sorry!* As an another report, a participant with 467 followers

mentioned: *@user: I've not been aware of this follower until now.* A participant with 1132 followers mentioned that *@user: very hard quiz which explains me that I do not know my social network on twitter.* On the other hand, to the question of Sub-step 2.4, a participant with 91 followers commented that: *@user: no, it's what I knew anyway.*

6.7.7 Interest of Followers

Around 28% of participants did not want to be advised if their followers were interested in a particular tweet. Among others, we highlight some of their arguments that carry interesting messages: *@user: I don't see why I should do this? Maybe for others twitting is a mission, but I'm simply twitting my moods or something that is interesting to me and I have never thought that I do it to please others...*; *@user: Usually I don't tweet only about topics in which my followers are interested, I tweet about things that I myself find interesting*; *@user: I would be glad to keep the interests of my followers in mind. yet i tweet what i find important to tweet, not necessarily targeting a particular audience*; *@user: As the point also of Twitter is to be surprised by news we won't expect. I won't like to receive only tweets which are supposed to be interested for me. I will miss the entertaining part of Twitter.*

6.7.8 Asking for Retweet

Asking people for retweeting is not unusual [Boyd et al., 2010], but people mainly request for retweeting by adding a phrase like "please RT". 64% of our participants mentioned that they never explicitly asked anybody to retweet their tweets, however, 67% of participants mentioned that if somebody asks them to retweet a tweet, they will do so. Apparently, the Twitter culture is not shaped (so far) by people mentioning explicit users for a retweeting request: *@user: I have never asked for a retweet before*; *@user: I only use @xxx when I address a message, not using it as an incentive to retweet.* Note that the mission of Tadvise is not to explicitly ask recommended hubs to retweet a tweet, but rather to attract their attention.

6.7.9 Well-Connected Hubs

Around 78% of participants were positive about being recommended with well-connected hubs: *@user: Sometimes I am trying to send/spread a message. In that case, this could be helpful*; *@user: For certain types of tweets... e.g. a petition that I would like people to sign.* Several participants mentioned also other advantages of tools like Tadvise: *@user: I think this tool [Tadvise] could be a good idea not necessarily to ask people to RT, but to suggest you new followers through your current followers and according to your interest.*

6.7.10 Tadvise Recommendations and Explanations

Several participants mentioned that our recommendations and explanations were not convincing and they claimed that they know their followers better: *@user: I feel I have better connected followers.* Such issues can be justified as follows: our mission is to recommend well-connected hubs in relation to a topic in order to propagate a tweet to more relevant audience at a distance of two. In other words, having a well-connected follower is not an issue,

but rather having a well-connected follower in relation to a topic is important. For example, a user with 100 followers could have more interested followers in relation to a specific topic (e.g., drupal) than a user with 500 followers.

The importance of monitoring who might actually retweet a tweet was also raised by some participants: *@user: aviobiletes is a user for tweets about cheep airline tickets. not likely it will retweet any of my content unless related to their topic.* We have to find ways to distinguish between personal and corporate Twitter users. Moreover, we have to enable a feature in our system to remember any decision made by users for influencing future advice.

To give advice on potential retweeting candidates, we currently consider all followers of a person, regardless of their interests due to Twitter's one-to-all architecture. A participant mentioned that besides connectivity, we should also consider the interest of users who we recommend as retweeting candidates: *@user: Another consideration is whether the person is interested in the topic (these guys are) so likely to retweet it.* Several participants mentioned that we should only propose those candidates who are also friends: *@user: One of those followers I barely know, I'm not going to ask him to retweet something. the other, yeah sure.*

6.7.11 Survey Design Issue

Several participants mentioned that the limitations on selecting Twitter users in Step 3 did not allow them to select people who were actually correct answers (from their perspective) and thus they `disagreed` with the results: *@user: I could not select enough people... sorry. I "missed" only those I couldn't select... Otherwise, I agree, of course.* This is a valid point, however, we designed the survey so that the participants were not able to select all options of a list, as we thought some participants may select the options randomly (without thinking).

6.8 Conclusion and Future Work

In this chapter, we presented our policy advisor. Our policy advisor aims to help users reduce information overload and information shortage in a network of people-tag-based profiles. The policy advisor approach uses people-tag-based social networks as input and builds enriched user profiles. Then it uses these profiles to find how relevant a topic is to the users at different distances. Moreover, it may recommend several users who are well-connected topic-sensitive hubs for propagating a message. To enact the policy advisor, we focused on Twitter, a micro-blogging platform that is currently used by many professionals. We developed Tadvise, a system for helping users to manage the flow of messages in Twitter. We conducted a personalised evaluation survey which indicated that users are interested in receiving advice on topic-relevant hubs who might retweet a message. Our measurements suggested that our approach can control information overload as well as information shortage within a network of people-tag-based profiles. We also provided some insight in relation to Twitter lists (i.e., people-tagging on Twitter), as it is relatively a new feature of Twitter.

As our approach is based on the wisdom of the crowds (i.e., user lists), spam accounts will not be considered in our analysis, as normally people do not assign spam accounts to any lists. Celebrities are also out of scope, as they are followed by millions of twitterers who are from various communities. Moreover, Twitter has a feature that enables users to automatically follow back users who follow them. This feature is used by some celebrities, and thus may

create noise for ranking tags and building user profiles, as these celebrities are not actually following the updates of millions of users, but rather follow them as a courtesy. Another consideration is the fact that normally people do not ask celebrities to act as a hub.

To give better advice and also to solve the cold-start problem of our policy advisor, we plan to enable users to enrich and vote for tags of people whom they know in order to receive better recommendations. Sentiment analysis of resources (e.g., tweets, URLs) can be also considered as a future step. We also plan to monitor the retweet network on Twitter, as some users may tend not to act as a hub. Our other goal is to define a refractory period for users, so that not all (attention) requests be targeted in a short space of time to specific hubs.

Part III

Conclusion

Chapter 7

Summary and Future Work

> I know enough to realise that I know nothing.
>
> Abu-Shakour Balkhi

In this thesis, we focused on the people-tagging. People-tagging aims to help users to build fine-grained user profiles by making this practice collaborative. Such fine-grained user profiles can be used for various purposes such as expert finding and information filtering which aim to improve collaboration and communication among knowledge workers. Receiving a tag by a tagger (e.g., a colleague) may incentivise the taggee – a user who is tagged by others – to mutually tag the tagger in return [Bernstein et al., 2009], however, user motivations for tagging others as well as themselves may decrease due to additional overhead that this practice may offer. Developing (semi-) automated approaches to assist users to tag themselves and others can improve the people-tagging habit among knowledge workers and additionally address the cold-start problem of people-tag-based systems. In this thesis, we developed an approach to extract, rank and assign expertise-based people-tags to users based on their interaction and contribution history within collaborative platforms. Users can then maintain the suggestions and remove inappropriate tags. We further use people-tag-based profiles for developing a more user-centric information propagation model within social and collaborative platforms. Our model together with a policy advisor enables users to propagate community-related information more effectively in a network of people-tag-based profiles.

The remainder of this chapter proceeds as follows. In Section 7.1, we present a summary of the conceptual as well as the applied contributions of this thesis. In Section 7.2, we elaborate on open research questions and future work; and finally with concluding remarks in Section 7.3 we conclude the chapter and this thesis.

7.1 Contributions

We have made the following insights and contributions to the social and collaborative software communities.

- Earlier work studied people-tagging behaviour in a large corporation and classified the most common tags that are used in a closed environment into several categories such as technology, expertise and so on [Muller, 2007, Muller et al., 2007]. Yet, people-tagging outside organisations and in open platforms such as blogs and social networks has not been well studied. The reason is perhaps because public social platforms with embedded people-tagging features are not as popular as other social platforms with embedded tagging feature applicable to online items such as URLs. Obviously, a better understanding of pubic people-tags will complement a general understanding of people-tagging practices and increase the usability of applications which use profiles based on people-tags. To this end, we gathered people-tags from four different websites, aggregated the tags and classified them using a tagging classification scheme inspired from [Sen et al., 2006]. We observed that people-tags, extracted from online social blogs, are more subjective than tags assigned to bookmarks or even organisational people-tags [Muller, 2007, Muller et al., 2007]. People-tags in our corpus reflected mainly personal opinions about mostly behaviourial and physical characteristics of taggees. The reason for this major difference between people-tagging inside and outside organisations could be that in an organisation, there exists some sort of trust or professional behaviour among co-workers which may affect people-tagging practices [Raban et al., 2011].

  We also analysed Twitter lists, a Twitter feature that enables users to tag each other, as presented in Chapter 6. Due to a gradual shift of enterprises towards enterprise 2.0 [McAfee, 2006], we envision that more knowledge workers will adopt micro-blogging services such as Twitter. Unlike online social blogs, people-tags on Twitter were mainly objective and could be categorised into several major categories that were also observed internally within large organisations [Muller, 2007, Muller et al., 2007]. Twitter lists can be perceived as community names within Twitter, as earlier work showed existence of the communities in Twitter [Weng et al., 2010, Java et al., 2007].

- Recently, several content management systems and online shared workspaces such as OrbiTeam BSCW enabled people-tagging as an internal feature[1]. Current people-tagging practices within such platforms are manual tasks in which users have to explicitly assign tags to themselves or others. This, however, may decrease user motivations for tagging as well as may lead to a cold-start problem for people-tag-based systems. To overcome this limitation, we presented our approach for extracting, ranking and assigning expertise-based people-tags to users based on their contribution and interaction history within online platforms such as question–answering (Q–A) forums and online shared workspaces. The threads that are discussed in Q–A forums as well as documents that are stored within online shared workspaces contain rich metadata that reflect the expertise and interests of users who discuss and collaborate on them. In order to develop our approach, we chose to focus on Q–A forums due to the availability of large data corpora that could be used for evaluation purposes. However, our approach can be extended to other collaborative platforms too.

  It is a well accepted hypothesis that the most frequently occurring phrases in a technical and mature (finished) document such as a scientific deliverable or a Q–A thread reflect important terms of the document – see Chapter 4. Initially, we extract such important terms from forum threads using various methods such as Latent Dirichlet Allocation

[1] http://www.bscw.de/bscw44.html

(LDA). The extracted phrases are ranked and assigned to a knowledge worker based on his/her contribution and interaction level within Q–A forums. For example, the more issues related to *Web services* someone solves, the higher the rank of *Web services* in his/her profile will be. We used a training set to build expertise profiles and a test set to evaluate our approach. Given a question in the test set, we used the expertise profiles to return a ranked list of experts who can answer that question. We evaluated our approach by using the title and the body of a new question. The evaluation showed that for both the title and the body of a new question, we can recommend a ranked list of experts who could address the question.

- Current access control models that are used in social and collaborative platforms are not flexible enough to express fine-grained access control policies [Hart et al., 2007]. This may be because access control models within social platforms were inspired by role-based models that are most effective in closed environments such as operating systems. Such models are not flexible enough to allow user-generated contents to be shared in a user-centric manner in a network of connected users [Sari et al., 2007]. To address this drawback, we further investigated usage of people-tag-based profiles in an information sharing use case and developed a novel access control model called Annotation-Based Access Control (AnBAC). The AnBAC model enables users to define policies on top of people-tags in order to share online resources such as bookmarks. Defining appropriate sharing policies enables users to share a resource with people who were tagged with specific phrases and are also located in a particular scope (i.e., distance) of the user who defines the policies. The distance criterion is a positive integer value that determines how far (or how may hops) an item should propagate in the network. In the advanced version of our model, a policy may have more parameters. For example, a user may assign tags to an item[2] and such item tags can be used together with people-tags in the policies. Our model is generic and we did not bind users to a specific technology, however, we provided general implementation as well as privacy guidelines for users who want to adopt our approach within their applications. Our model is well suited for sharing and propagating online items such as news and announcements in communities – network of connected people with similar interests, expertise and so on.

  People-tags and numeric values (i.e., distances) are two main elements of policies in our model. Both elements may be sources of imprecision due to various reasons such as annotation drift. Thus, users may require assistance in drafting policies in order to effectively propagate the community-related items in the network. Without an appropriate policy advisor, our model may lead to broadcasting too much information (i.e., information overload) or too little information (i.e., information shortage) within social platforms.

- To assist users in drafting policies, we provide an approach based on people-tag-based profiles which can advise end users that a resource that they are going to share might be relevant to their contacts (at distances of one or two) or not. The policy advisor takes an item (such as a URL) and a network of people-tag-based profiles as input and initially ranks people-tags based on ranking of the taggers. By analysing the input item, the policy advisor is able to extract the main topics of the item and by matching the topics of the item with ranked people-tag-based profiles, it can provide information

[2]Tags can be extracted from an item as well.

sharing advice. Moreover, in order to effectively propagate a resource further (e.g., a distance of two from a user) in the network and to minimise information overload and information shortage, our approach can recommend one or more topic-sensitive hubs who may propagate a relevant resource such as an announcement further in the network.

Our policy advisor analyses both a user's direct contacts and the direct contacts' contacts (i.e., a distance of two). The reason that we focused on two distances (i.e., one and two) in a network of connected users is due to dense structure of communities [Weng et al., 2010, Java et al., 2007]. In other words, users of a community are typically connected either directly to each other (i.e., a distance of one) or through a hub (i.e., a distance of two). The evaluation showed that our policy advisor assists users in propagating their community-related resources more effectively in a network of people-tag-based profiles.

In order to facilitate the evaluation of our approaches, we developed several software prototypes. In the following part, we have an overview of the tools that we developed.

- We developed a tool called Sapport that was customised for Q–A forums of a large corporate (i.e., SAP Community Network Forums). Sapport enables end users to search for a ranked list of relevant experts in relation to a topic as well as browse detailed expertise profile of a user. In order to convince end users that the list of recommended experts is relevant, Sapport provides explanations in two formats: graphical (in the form of pie chart) as well as text-based. The explanations are based on history of the threads that an expert contributed to (e.g., solving an issue or providing (very) helpful answers). Besides Sapport which was mainly developed for Q–A forums, we also developed a tool for the OrbiTeam BSCW online shared workspace to analyse stored documents and build expertise profiles using OrbiTeam BSCW log files. The OrbiTeam BSCW log files contain document-based events such as read, write, revise and so on.

- To realise our access control model, we developed a tool called Uncle-Share that enables end users to share bookmarks or short messages with their contacts using the AnBAC model. Uncle-Share enables users to add contacts to their social networks, tag them using arbitrary terms and define access control policies on top of such tags. Users who are eligible to see shared items can access them when they successfully authenticate themselves. They can change the distance parameter via the user interface in order to increase or decrease the scope of available resources (e.g., direct contacts, contacts of direct contacts, etc.).

- In order to develop our policy advisor framework and evaluate it, we decided to focus on a platform with real-world people-tags which contains real-time information flow. In particular, we focused on Twitter as a micro-blogging platform which is currently well-adopted by many organisations and knowledge workers. Unlike the early days of launching Twitter, nowadays, the Twitter culture is shaped so that knowledge workers use it to propagate community-related information such as conference announcements and news. Moreover, Twitter lists (i.e., people-tags) are similar to those tags that are used internally in a large organisation, as they could be categorised into similar groups (e.g., affiliation, expertise, etc. – see Chapter 6). Our Twitter assistant, called Tadvise, helps users to know the community of their followers better. It also assists people

who tweet in identifying strategically well-connected and topic-sensitive followers who may propagate a tweet further in Twitter, ensuring that a community-related tweet reaches the most appropriate and interested people related to that community. Our policy advisor is equipped with various graphical (e.g., radial, icons) and text-based explanations, in order to convince end users about recommendations.

From the infrastructure point of view, we used Semantic Web technologies and standards to make our approaches interoperable between various platforms. Several prototypes (UncleShare and BSCW Expert Finder) were developed based on the SOA paradigm and the rest can be also easily extended to provide Web services. Where possible, we developed our prototypes as widgets or gadgets in order to increase their reusability. We used open source and free tools, libraries and packages for developing our prototypes.

7.2 Open Questions and Future Work

The people-tagging practice opened a new area of research, as it enabled users of a system or personnel within an organisation to tag themselves or others. Most studies in this domain are limited to people-tagging behaviour inside large organisations [Muller et al., 2007, Muller, 2007, Raban et al., 2011]. The reason that public social people-tags were not well studied was perhaps due to the fact that websites that enable users to explicitly tag each other are not (or perhaps were not) as popular as other Web 2.0 services with embedded tagging feature like *delicious.com* or *flickr.com*. As tagging is an embedded feature of many online shared workspaces, extending this activity to tag users and embedding this feature into content management systems (CMS) and online shared workspaces do not require much implementation overhead for application developers. Recently, several collaborative software vendors such as OrbiTeam[3] incorporated people-tagging into their products. We envision that in upcoming years, people-tagging will become more popular in medium-to-large organisations. However, the main goal of people-tagging is not to make personal tags publicly accessible, but to use them for expert finding and information sharing use cases. Anonymising organisational people-tags and releasing them to the public domain could feed further research on organisational people-tag corpora. Perhaps by analysing people-tags, ongoing collaboration patterns and internal trends among knowledge workers can be extracted and be utilised.

Our conceptual approaches as well as our prototypes can be improved in several ways. To this end, we envision studying other aspects of our approaches:

- The trade-off between usability and security: Our information filtering approach is a trade-off between usability and security in which users are able to share personal resources with others in a user-centric manner. Our model suits well for users who seek to share user-generated contents within online communities, however, it may not be suitable for sharing critical information with people who are located at a distance of two and beyond. A question that might raise is where the border between usability and security should be placed. Does it make sense to adopt our approach for platforms with high security demands (e.g., semantic desktop platforms like KDE)? If so, what characteristics of such environments should be considered for achieving this goal?

[3] http://www.bscw.de/unternehmen.html

- Supporting temporal aspects of people-tags: Tags related to specific categories such as expertise, affiliation, hobby, etc. may have a limited life span. Technology evolves rapidly and trends appear quickly, so that they may influence life span of people-tags. Obviously, a user can explicitly maintain his/her tag-based profile, however, we envision developing approaches to reduce the overhead of maintaining tag-based profiles.
- Smart gadgets: From the application point of view, nowadays, many knowledge workers are equipped with smart gadgets such as smart phones and tablets. We plan to extend the tools that we developed to make them compatible and more accessible to smart devices. Moreover, by accessing the location of knowledge workers and also their calendars, we may automatically fetch their context information to provide better advice.

As tagging is an embedded feature of many online collaborative platforms, extending this activity to humans does not require much implementation overhead. Thus, we highly encourage application developers to embed this feature into their products. This will undoubtedly unleash new use cases and will improve collaboration and communication among knowledge workers.

7.3 Concluding Remarks

Contributions of this thesis are both conceptual and applied. From the conceptual point of view, we provided (i) insight on people-tagging behaviour of users within online social platforms; (ii) a novel approach for extracting and assigning (ranked) people-tags (i.e., expertise elements) to users of Q–A forums as well as online shared workspaces; (iii) a novel information filtering model (i.e., AnBAC) based on two main elements (i.e., people-tags and a distance criterion) that can be used in many current social and collaborative platforms; and (iv) a novel approach based on people-tag-based user profiles to assist users for propagating community-related information within a network of connected people based on the AnBAC model.

From the implementation point of view, we provided (i) two expert finder systems that extract and assign expertise elements to users who collaborate within question–answering forums as well as in online shared workspaces; (ii) a widget that can be embedded into iGoogle or any other widget platform to enable users to benefit from the AnBAC model; and (iii) a policy advisor called Tadvise that assists users in propagating community-related information more effectively in the network.

Part IV

Addendum

Chapter 8

Appendix I – Collaboration Vocabulary (CoVoc)

8.1 CoVoc Ontology

People-tagging is a novel and collaborative approach for building user profiles and this could be a difficult practice for some users and they may require some help/suggestions to find appropriate tags. Moreover, in a Collaborative Working Environment (CWE), there exist several common professional relationships among people such as co-authoring a report. Identifying and documenting these common relationships in a formal way will help interoperation and make the integration among applications easier. Due to these reasons and as our focus is also on enterprise domains, we decided to build a vocabulary to help IT-related enterprise users to tag each other. We chose this narrow domain, as it was related to the domain that we work in and we had experience and knowledge regarding domain activities.

For building our vocabulary, we used Semantic Web technologies and ontologies. In Chapter 2, we introduced advantages of using ontologies, as they enable software developers to utilise them within their applications with a little effort using current existing software libraries.

Among ontological consideration of human relationships [Matsuo et al., 2004, Gan et al., 2004], the FOAF project[1] is perhaps the most used and adapted ontology on the Web. This vocabulary enables users to express their contact information and social networks in RDF. The main advantage of FOAF (and other ontologies) is perhaps the extensibility; the fact that they can be easily extended to meet new requirements. For example, FOAFCorp [Brickley, 2006] extends FOAF to describe the structure and interconnections of corporate entities in more detail; or [Gan et al., 2004] provide several terms as FOAF extensions to cover the often-changing variables in FOAF. The latter can be considered as providing context information for FOAF profiles.

The other advantage of such vocabularies is that a snippet of them can be used to model a new domain. For example, SIOC [Breslin et al., 2006] which is a common vocabulary and is used to interlink online communities uses FOAF to address user-related terms. As an another example, the RELATIONSHIP [Davis and Vitiello, 2005] and the REL-X [Carminati et al., 2006a]

[1] http://www.foaf-project.org/

ontologies are built on top of FOAF. Both RELATIONSHIP and REL-X ontologies model general interpersonal relationships. The former is based on RDFS[2], whereas the latter is based on OWL[3]. Similar to RELATIONSHIP and REL-X, efforts like XFN[4] (XHTML Friends Network) tries to embed social networks and human relationships using hyperlinks like HTML. The XFN initiative also proposes several general terms for social acquaintances.

Above efforts contain the terms that capture the `general` relationships and social acquaintances among people, however, we are not aware of any vocabulary that targets specifically the annotations of knowledge workers within collaborative working environments (specially research-related environments). Our work is a first attempt to model human relationships in collaboration-related context.

Our vocabulary is called CoVoc (COllaboration VOCabulary). CoVoc is a simple but general vocabulary that focuses on collaboration and can be used for tagging knowledge workers in an IT-related collaborative environment. In brief, CoVoc is a set of terms that address various collaborative relationships and social acquaintances that exist between individuals (knowledge workers) in a collaborative working environment (mainly IT-related research environments).

8.1.1 CoVoc Fundamentals and Sources

For developing CoVoc, we followed a bottom-up approach, meaning that by looking at different sources, we selected an initial set of terms that reflect collaborative behaviour in an IT-related enterprise. For example, if two researchers published a paper together, we implied the term co-author to represent the relationship between them. In particular, we used the following sources:

- We looked at detailed Curriculum Vitae (CV) of senior researchers, Ph.D. and M.Sc. students (in total 30 profiles) to determine what they have performed and/or are performing in their professional (research) lives. The total number of 30 profiles were enough for our case, as activities of IT-related knowledge workers overlapped and we could not identify any more new terms.
- We surveyed the following ontologies from SchemaWeb[5], as they appeared relevant to collaboration: Wordnet Project, Wordnet Person, Wordnet Organization, Wordnet Document, Wordnet Agent-3, WebScripter Person, vCard, VANN, SIOC, SKOS Core, SKOS Extensions, SKOS Mapping, Software Project, Software Tools Ontology, Resume, Reading List Schema, Presentations, PRJ Project Vocabulary, Project Management, Publishing, Pervasive Computing Standard Ontologies, FOAF, FOAF Extensions, FOAFCorp, ESWC2006 Conference Ontology, eBiquity Ontologies, Dublin Core Ontologies, ConOnto Ontologies, Conference Ontology, DOAC, DOAP, Competency-Oriented Human Resource Development Ontology, Charette Relationship Set, CERIF (Common European Research Information Format), Blogger Code, BIO, and Annotea. We also looked at some other ontologies like Enterprise ontology [Uschold et al., 1998] and TOVE ontology [Fox, 1992]. We picked those terms that were relevant to collaboration behaviour in IT domain.

[2] http://www.w3.org/TR/rdf-schema/
[3] http://www.w3.org/TR/owl-ref/
[4] http://gmpg.org/xfn/
[5] http://www.schemaweb.info/

After coming up with an initial set of 112 terms, we manually categorised them. We came up with the following five main categories:

- Project-Related Collaboration: An IT-related knowledge worker collaborates in different projects (e.g., writing deliverables, writing proposals, etc.)
- Collaborative-Organised Events: An IT-related knowledge worker participates in various collaborative-organised events (e.g., conferences, workshops, etc.) and publishes various material (e.g., papers, books, etc.)
- Academic Collaboration: An IT-related knowledge worker may be part of the university board (e.g., lecturer, adjacent lecturer, etc.)
- Industrial Collaboration: An IT-related knowledge worker may be involved in industry (e.g., CEO, CTO, etc.)
- Online Social Collaboration: An IT-related knowledge worker has various online social activities (e.g., reading blogs, watching online videos, listening to podcasts, etc.)

For each category, we assigned relevant terms from our set of 112 terms. It is important to consider that some terms may suit for more than one category. As an example, the term `supervisor` can be used in both academic and industrial environments. Note that this categorisation is mainly utilised for a better `internal` classification of the terms and end users of such terms will just use the terms and not the categories alone for tagging each other, however, the categories may be used to provide context (i.e., super tag) for the tags. The CoVoc terms have been selected as far as possible to match the natural use of English words by people collaborating together. The CoVoc terms together with their descriptions are presented in Section 8.2. The terms included in the current version of CoVoc follow on two broad categories:

- Terms which are related to relationships between persons. These are the terms that describe the actual relationships between two persons that collaborate together (e.g., writeDocumentWith). From an ER modelling [Chen, 1976] perspective, these are possible relationships that may exist amongst entities.
- Terms that are related to personal characteristics that acquire interest for the users in a collaborative context (e.g., supervisor). Again, from an ER perspective, these are attributes of the entities that somehow influence the relationship of the entity with other external entities.

The latter category should not be part of a Collaboration Vocabulary, as it covers personal characteristics that exist at the user profile and not at the relationship level. Ideally, these characteristics should be accessed through user profiles. But due to a lack of such profiles, we included these terms in CoVoc in order to allow users to annotate their relationships using them. These terms are marked with (*) in this appendix.

8.1.2 CoVoc Schema

We developed an RDF Schema for CoVoc. The schema composes of one *RDFS class*, called `Collaboration`, and all CoVoc terms are considered as *RDF properties*. For compatibility purposes, the schema extends the RELATIONSHIP ontology and utilises the FOAF vocabulary. The RDF properties of CoVoc terms are sub-properties of some properties of RELATIONSHIP or sub-properties of other CoVoc terms' properties. The following properties belong to the RELATIONSHIP ontology: `knowsOf`, `hasMet`, `wouldLikeToKnow`, `collaboratesWith`. Listing 8.1 shows a snippet of the CoVoc schema (`writeDeliverableWith` property). The listing shows that both `rdfs:domain` and `rdfs:range` of `covoc:writeDeliverableWith` are `foaf:Person`. Moreover, `covoc:writeDeliverableWith` is sub-property of `covoc:writeDocumentWith`.

Listing 8.1: One property of the CoVoc ontology.

```
<rdf:Property rdf:about="http://purl.oclc.org/vocabulary/covoc#writeDeliverableWith">
        <rdfs:subPropertyOf rdf:resource="http://purl.oclc.org/vocabulary/covoc#writeDocumentWith"
             rdfs:label="Write Document With"/>
        <rdfs:label xml:lang="en">Write Deliverable With</rdfs:label>
        <rdfs:comment xml:lang="en">A person writes a deliverable with this person.</rdfs:comment>
        <rdfs:domain rdf:resource="http://xmlns.com/foaf/0.1/Person" rdfs:label="Person"/>
        <rdfs:range rdf:resource="http://xmlns.com/foaf/0.1/Person" rdfs:label="Person"/>
        <rdfs:isDefinedBy rdf:resource="http://purl.oclc.org/vocabulary/covoc#"/>
</rdf:Property>
```

The schema of CoVoc is accessible online[6]. We assigned a Persistent URL (PURL) to the namespace of CoVoc which can be used by software engineers; for example, `http://purl.oclc.org/vocabulary/covoc#doResearchWith` is the valid reference for the `doResearchWith` term.

8.1.3 CoVoc Usage

CoVoc has been mainly developed with the aim that it can be utilised for tagging knowledge workers in mainly IT and research-related social networks. Like RELATIONSHIP, CoVoc can be simply utilised in FOAF profiles and also within HTML/XHTML documents (using `rel` and `rev` attributes, like the XFN approach). Listing 8.2 shows a sample FOAF profile that uses CoVoc for annotating people. In natural language, Listing 8.2 shows that a Person by the name `Peyman` does research with a Person by the name `Vassilios` and met a Person by the name `Wolfgang` in a conference.

Listing 8.2: Using CoVoc in the FOAF profiles.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
      xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
      xmlns:foaf="http://xmlns.com/foaf/0.1/"
      xmlns:covoc="http://purl.oclc.org/vocabulary/covoc#"
      >
  <foaf:Person rdf:nodeID="peyman">
    <foaf:name>Peyman</foaf:name>

    <covoc:doResearchWith>
      <foaf:Person>
        <foaf:name>Vassilios</foaf:name>
      </foaf:Person>
    </covoc:doResearchWith>

    <covoc:metInConference rdf:nodeID="wolfgang"/>
  </foaf:Person>

  <foaf:Person rdf:nodeID="wolfgang">
```

[6] http://purl.oclc.org/vocabulary/covoc/

```
    <foaf:name>Wolfgang</foaf:name>
  </foaf:Person>

</rdf:RDF>
```

Listing 8.3 shows a snippet of a HTML file that uses CoVoc for linking people. This listing presents the same scenario as explained for listing 8.2.

Listing 8.3: Using CoVoc in the HTML files.

```
<html>
  <head profile="http://purl.org/vocab/relationship/">
    <title>CoVoc in HTML</title>
  </head>
  <body>
    <p>I know these folks:</p>
      <ul>
        <li><a href="http://www.deri.ie/about/team/member/vassilios_peristeras/"
               rel="doResearchWith">Vassilios</a></li>
        <li><a href="http://wolfgang.com/"
               rel="metInConference">Wolfgang</a></li>
      </ul>
  </body>
</html>
```

8.1.4 Discussion

As CoVoc is a bottom-up open vocabulary and not a fixed set of terms, it will be evaluated mainly with `usage`. In order to initially evaluate our vocabulary, we had various discussions and interviews with senior researchers, Ph.D. and M.Sc. students within our institute which led to current version of this vocabulary. We tried to keep CoVoc as general and simple as possible for IT-related collaboration terms. Based on this principle, terms such as `participate_in_W3C_standardisation` which was proposed by a senior researcher, sounds to be too specific to be used in CoVoc. Following this approach, we did not include many enterprise-related (e.g., intra- or inter-company) collaboration terms [Uschold et al., 1998, Fox, 1992] in CoVoc, as end users of such terms may refer to the original ontologies. It is also important to consider that many individual-related online activities of people (e.g., uploading photos, downloading media, etc.) do not take place in the last category of CoVoc – see Section 8.2, as they reflect individual-based activities, rather than collaboration-based activities. Our goal is to capture collaboration behaviour between IT-related knowledge workers.

The CoVoc ontology is easily extensible to meet new requirements and terms. Software engineers are able to easily extend it and add special collaboration-based terms for specific purposes (e.g., `participate_in_W3C_standardisation`).

8.1.5 CoVoc-Based Visualisation

As CoVoc can be used to facilitate agreement among knowledge workers, such agreements can be useful in many agenda, such as visualising social networks. It is a well-accepted hypothesis that visualisation leads to immediate grasp of new concepts. Visualising inter- and intra-organisational social networks can assist authorised entities such as people at HR department to easily understand the internal network among employees. To this end, we have developed Who-With-Whom which is a simple prototype that visualises the annotated social networks based on CoVoc terms. We used `Graph Gear`[7] for visualising the graphs which is based on Adobe Flash. Who-With-Whom fetches the RDF triples that are related to a

[7] http://www.creativesynthesis.net/blog/projects/graph-gear/

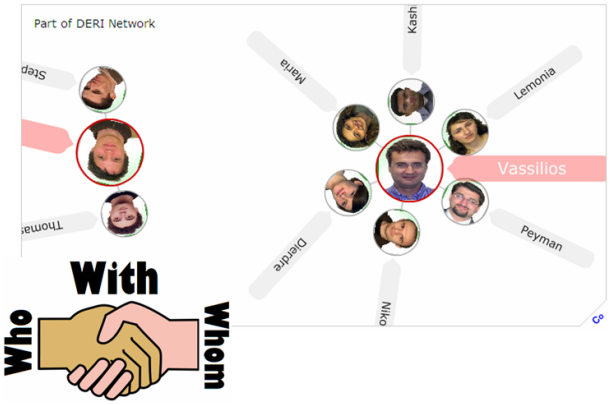

Figure 8.1: A snapshot of Who-With-Whom: a tool for visualising social networks based on CoVoc terms.

specific CoVoc term and transforms them to the XML files which feed Graph Gear. If users' photos were already stored in the repository, they will be shown in the graph as well. Figure 8.1 demonstrates a snapshot of Who-With-Whom.

Who-With-Whom is also able to export the social network based on each CoVoc term as RDF. The generated RDF is based on FOAF and CoVoc and currently contains real name of users plus their CoVoc-based relationships. As an example, Listing 8.4 shows a snippet of the network of people who are *writing deliverables* (i.e., `covoc:writeDeliverableWith`) together, generated by Who-With-Whom. We verified the generated RDF with W3C RDF Validator[8].

Listing 8.4: A snippet of a collaboration network in RDF.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
        xmlns:foaf="http://xmlns.com/foaf/0.1/"
        xmlns:covoc ="http://purl.oclc.org/vocabulary/covoc#"
        xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
        xmlns:rdf ="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:openOntology = "http://nowhere.org#">

<foaf:Person rdf:about="http://uncle-share.com/persons#Vassilios">
        <covoc:writeDeliverableWith rdf:resource="http://uncle-share.com/persons#Peyman"/>
        <covoc:writeDeliverableWith rdf:resource="http://uncle-share.com/persons#Kashif"/>
        <covoc:writeDeliverableWith rdf:resource="http://uncle-share.com/persons#Nikos"/>
        <covoc:writeDeliverableWith rdf:resource="http://uncle-share.com/persons#Maria"/>
        <covoc:writeDeliverableWith rdf:resource="http://uncle-share.com/persons#Lemonia"/>
        <covoc:writeDeliverableWith rdf:resource="http://uncle-share.com/persons#Dierdre"/>
        <foaf:name>Vassilios</foaf:name>
</foaf:Person>

<foaf:Person rdf:about="http://uncle-share.com/persons#Axel">
        <covoc:writeDeliverableWith rdf:resource="http://uncle-share.com/persons#Stephane"/>
        <covoc:writeDeliverableWith rdf:resource="http://uncle-share.com/persons#Thomas"/>
        <foaf:name>Axel</foaf:name>
</foaf:Person>

<foaf:Person rdf:about="http://uncle-share.com/persons#Peyman">
        <covoc:writeDeliverableWith rdf:resource="http://uncle-share.com/persons#Vassilios"/>
        <foaf:name>Peyman</foaf:name>
</foaf:Person>

<foaf:Person rdf:about="http://uncle-share.com/persons#Thomas">
        <covoc:writeDeliverableWith rdf:resource="http://uncle-share.com/persons#Axel"/>
        <foaf:name>Thomas</foaf:name>
</foaf:Person>

.
.
.
```

[8] http://www.w3.org/RDF/Validator/

```
38
39 </rdf:RDF>
```

Who-With-Whom is available online and can be accessed[9].
In the following section, we have an overview of the CoVoc terms.

8.2 CoVoc Terms

8.2.1 Project-Related Collaboration

Table 8.1 lists the terms that belong to this category.

Table 8.1: Project-related collaboration properties.

Term (Property)	Description	Sub-property
writeDocumentWith	A person writes a document with this person.	collaborateWith
reviseDocumentWith	A person revises a document with this person.	collaborateWith
reviseDocumentOf	A person revises a document that has been created by this person.	collaborateWith
collaborateCloselyWith	A person has a close collaboration with this person.	collaborateWith
collaborateWith	A person collaborates with this person.	collaboratesWith
writeDeliverableWith	A person writes a deliverable with this person.	writeDocumentWith
writeProposalWith	A person writes a proposal with this person.	writeDocumentWith
writeReportWith	A person writes a report with this person.	writeDocumentWith
reviseDeliverableWith	A person revises a deliverable with this person.	reviseDocumentWith
reviseProposalWith	A person revises a proposal with this person.	reviseDocumentWith
reviseReportWith	A person revises a report with this person.	reviseDocumentWith
reviseDeliverableOf	A person revises a deliverable that has been written by this person.	reviseDocumentOf
reviseProposalOf	A person revises a proposal that has been written by this person.	reviseDocumentOf
reviseReportOf	A person revises a report that has been written by this person.	reviseDocumentOf
reviewer*	A person knows this person who is a reviewer.	knowsOf
coordinator*	A person knows this person who is a coordinator.	knowsOf
leader*	A person knows this person who is a leader.	knowsOf
workingPackageLeader*	A person knows this person who is a working package leader.	leader
taskLeader*	A person knows this person who is a task leader.	leader
deliverableLeader*	A person knows this person who is a deliverable leader.	leader
groupLeader*	A person knows this person who is a group leader.	leader
workingPackageContributor*	A person knows this person who is a working package contributor.	knowsOf
taskContributor*	A person knows this person who is a task contributor.	knowsOf
deliverableContributor*	A person knows this person who is a deliverable contributor.	knowsOf
projectManager*	A person knows this person who is a project manager.	knowsOf
developer*	A person knows this person who is a developer.	knowsOf
designer*	A person knows this person who is a designer.	knowsOf

Continued on next page

[9] http://purl.oclc.org/projects/who-with-whom

Table 8.1 – continued from previous page

Term (Property)	Description	Sub-property
assembler*	A person knows this person who is an assembler.	knowsOf
tester*	A person knows this person who is a tester.	knowsOf
partner*	A person knows this person who is a (professional) partner.	knowsOf
industrialPartner*	A person knows this person who is an industrial partner.	knowsOf
academicPartner*	A person knows this person who is an academic partner.	knowsOf
member*	A person knows this person who is a member of an organisation.	knowsOf
presenter*	A person knows this person who is a presenter.	knowsOf
hadMeetingWith	A person had a meeting with this person.	hasMet
metInProjectMeeting	A person met this person in a project meeting.	hasMet
hadSocialEventWith	A person had a social event (e.g., dinner, lunch) with this person.	hasMet
hadConfCallWith	A person had a confcall with this person.	hasMet
invited	A person has invited this person (e.g., to a project).	knowsOf
invitedBy	A person has been invited by this person (e.g., to a project).	knowsOf
wouldLikeToCollaborateWith	A person would like to collaborate with this person.	wouldLikeToKnow
wouldLikeToWriteProposalWith	A person would like to write a proposal with this person.	wouldLikeToCollaborateWith
wouldLikeToHaveMeetingWith	A person would like to have a meeting with this person.	wouldLikeToCollaborateWith
wouldLikeToHaveConfcallWith	A person would like to have a confcall with this person.	wouldLikeToCollaborateWith
wouldLikeToContactWith	A person would like to have contact with this person.	wouldLikeToCollaborateWith

8.2.2 Collaborative-Organised Events

Table 8.2 lists the terms that belong to this category.

Table 8.2: Collaborative-organised events and publications properties.

Term (Property)	Description	Sub-property
metInConference	A person met this person in a conference.	metInScientificEvent
metInWorkshop	A person met this person in a workshop.	metInScientificEvent
metInScientificEvent	A person met this person in a scientific event.	hasMet
metInIndustrialEvent	A person met this person in an industrial event.	hasMet
hadConferenceDinnerWith	A person had a conference dinner with this person.	hasMet
hadCoffeeBreakWith	A person had a coffee break with this person.	hasMet
exchangedBusinessCardWith	A person exchanged a business card with this person.	hasMet
askedHimHerQuestion	A person asked a question from this person.	hasMet
askedMeQuestion	This person asked a question from a person.	hasMet
answeredHisHerQuestion	A person answered this person's question.	hasMet
answeredMyQuestion	This person answered a person's question.	hasMet
presentedPaper*	A person knows this person who presented a paper in an event.	knowsOf
programCommittee*	A person knows this person who is a member of a program committee.	knowsOf
hadInvitedTalk*	A person knows this person who had an invited talk in an event.	knowsOf

Continued on next page

Table 8.2 – continued from previous page

Term (Property)	Description	Sub-property
editor*	A person knows this person who is an editor.	knowsOf
patentHolder*	A person knows this person who holds a patent.	knowsOf
specialThanksTo	A person sends his/her special thanks to this person.	knowsOf
organiser*	A person knows this person who is an organiser.	knowsOf
chair*	A person knows this person who is a chair.	knowsOf
generalChair*	A person knows this person who is a general chair.	chair
co-chair*	A person knows this person who is a co-chair.	chair
viceChair*	A person knows this person who is a vice chair.	chair
proceedingsChair*	A person knows this person who is a proceedings chair.	chair
developerChair*	A person knows this person who is a developer track chair.	chair
programmeChair*	A person knows this person who is a programme chair.	chair
tutorialChair*	A person knows this person who is a tutorial chair.	chair
workshopChair*	A person knows this person who is a workshop chair.	chair
demoChair*	A person knows this person who is a demo chair.	chair
posterChair*	A person knows this person who is a poster chair.	chair
panelChair*	A person knows this person who is a panel chair.	chair
phDSymposiumChair*	A person knows this person who is a PhD symposium chair.	chair
sponsor*	A person knows this person who is a sponsor.	knowsOf
administrator*	A person knows this person who is an administrator.	knowsOf
isInterestedInMyWork	A person's work is interesting for this person.	knowsOf
amInterestedInHisHerWork	A person is interested in this person's work.	knowsOf
writePaperWith	A person writes a paper with this person.	writeDocumentWith
author*	A person knows this person who is an author.	knowsOf
co-authorWith	A person co-authors something with this person.	writeDocumentWith
reviewPaperWith	A person reviews a paper with this person.	reviewDocumentWith
reviewPaperOf	A person reviews a paper that has been written by this person.	reviewDocumentOf
wouldLikeToMeetInEvent	A person would like to have a meeting with this person in an event.	wouldLikeToKnow
wouldLikeToWritePaperWith	A person would like to write a paper with this person.	wouldLikeToCollaborateWith
wouldLikeToInvite	A person would like to invite this person (e.g., to an event).	wouldLikeToCollaborateWith

8.2.3 Academic Collaboration

Table 8.3 lists the terms that belong to this category.

Table 8.3: Academic collaboration properties.

Term (Property)	Description	Sub-property
supervisor*	A person knows this person who is a supervisor.	knowsOf
mentor*	A person knows this person who is a mentor.	knowsOf
postDoc*	A person knows this person who is a postdoc.	knowsOf

Continued on next page

Table 8.3 – continued from previous page

Term (Property)	Description	Sub-property
phDStudent*	A person knows this person who is a Ph.D. student.	graduate
mScStudent*	A person knows this person who is a M.Sc. student.	graduate
intern*	A person knows this person who is an intern.	knowsOf
researchVisitor*	A person knows this person who is a research visitor.	knowsOf
bScStudent*	A person knows this person who is a B.Sc. student.	undergraduate
doResearchWith	A person does research with this person.	knowsOf
lecturer*	A person knows this person who is a lecturer.	knowsOf
memberOfAcademicBoard*	A person knows this person who is a member of an academic board.	knowsOf
undergraduate*	A person knows this person who is an undergraduate student.	student
graduate*	A person knows this person who is a graduate student.	student
adjunctLecturer*	A person knows this person who is an adjunct lecturer.	lecturer
student*	A person knows this person who is a student.	knowsOf
researchAssistant*	A person knows this person who is a research assistant.	knowsOf
teachingAssistant*	A person knows this person who is a teaching assistant.	knowsOf

8.2.4 Industrial Collaboration

Table 8.4 lists the terms that belong to this category.

Table 8.4: Industrial collaboration properties.

Term (Property)	Description	Sub-property
boardOfDirectors*	A person knows this person who is part of a board of directors.	knowsOf
corporateOfficer*	A person knows this person who is a corporate officer.	knowsOf
president*	A person knows this person who is a president (of an organisation).	knowsOf
strategyOfficer*	A person knows this person who is a strategy officer.	knowsOf
channelOfficer*	A person knows this person who is a channel officer.	knowsOf
financialOfficer*	A person knows this person who is a financial officer.	knowsOf
visionaryOfficer*	A person knows this person who is a visionary officer.	knowsOf
operatingOfficer*	A person knows this person who is an operating officer.	knowsOf
informationOfficer*	A person knows this person who is an information officer.	knowsOf
informationSecurityOfficer*	A person knows this person who is an info. sec. officer.	knowsOf
marketingOfficer*	A person knows this person who is a marketing officer.	knowsOf
analyticsOfficer*	A person knows this person who is an analytics officer.	knowsOf
administrativeOfficer*	A person knows this person who is an Adm. officer.	knowsOf
networkingOfficer*	A person knows this person who is a networking officer.	knowsOf
dataOfficer*	A person knows this person who is a data officer.	knowsOf
technicalOfficer*	A person knows this person who is a technical officer.	knowsOf
technologyOfficer*	A person knows this person who is a technology officer.	knowsOf
scienceOfficer*	A person knows this person who is a science officer.	knowsOf

Continued on next page

Table 8.4 – continued from previous page

Term (Property)	Description	Sub-property
legalOfficer*	A person knows this person who is a legal officer.	knowsOf
chiefStrategyOfficer*	A person knows this person who is a chief strategy officer.	strategyOfficer
chiefChannelOfficer*	A person knows this person who is a chief channel officer.	channelOfficer
chiefFinancialOfficer*	A person knows this person who is a chief financial officer.	financialOfficer
chiefVisionaryOfficer*	A person knows this person who is a chief visionary officer.	visionaryOfficer
chiefOperatingOfficer*	A person knows this person who is a chief operating officer.	operatingOfficer
chiefInformationOfficer*	A person knows this person who is a chief information officer.	informationOfficer
chiefInformationSecurityOfficer*	A person knows this person who is a chief inf. sec. officer.	informationSecurityOfficer
chiefMarketingOfficer*	A person knows this person who is a chief marketing officer.	marketingOfficer
chiefAnalyticsOfficer*	A person knows this person who is a chief analytics officer.	analyticsOfficer
chiefAdministrativeOfficer*	A person knows this person who is a chief adm. officer.	administrativeOfficer
chiefNetworkingOfficer*	A person knows this person who is a chief networking officer.	networkingOfficer
chiefDataOfficer*	A person knows this person who is a chief data officer.	dataOfficer
chiefTechnicalOfficer*	A person knows this person who is a chief technical officer.	technicalOfficer
chiefTechnologyOfficer*	A person knows this person who is a chief technology officer.	technologyOfficer
chiefScienceOfficer*	A person knows this person who is a chief science officer.	scienceOfficer
chiefLegalOfficer*	A person knows this person who is a chief legal officer.	legalOfficer
managingDirector*	A person knows this person who is a managing director.	knowsOf
secretary*	A person knows this person who is a secretary.	knowsOf
founder*	A person knows this person who is a founder of an initiative.	knowsOf
co-Founder*	A person knows this person who is a co-founder of an initiative.	knowsOf

8.2.5 Online Social Collaboration

Table 8.5 lists the terms that belong to this category.

Table 8.5: Online social collaboration properties.

Term (Property)	Description	Sub-property
readBlogOf	A person reads blog posts of this person.	readNewsOf
readWebsiteOf	A person reads website of this person.	knowsOf
followMicroBlogOf	A person follows micro-blog posts of this person.	readNewsOf
watchVideoOf	A person watches (uploaded) videos of this person.	knowsOf
listenToPodcastOf	A person listens to podcasts of this person.	knowsOf
seePhotoOf	A person sees (uploaded) photos of this person.	knowsOf
followBookmarksOf	A person follows bookmarks of this person.	knowsOf
playGameWith	A person plays game with this person.	knowsOf
readNewsOf	A person reads news of this person.	readWebsiteOf

Chapter 9

Appendix II – Utilising User-Centric Social Networks

9.1 User-Centric Social Networks

As stated in Chapter 4, we found an interesting use case by creating a user-centric perspective of latent social network among personnel extracted from log files of online shared workspaces. In this case, social relationships among users are composed of events or actions that were performed on objects such as documents. Such extracted social networks can be used in various use cases from information sharing [Goecks and Mynatt, 2004, Mori et al., 2005] and social learning [Chang et al., 2007, Nurmela et al., 1999] to auto-completing address book entries [Culotta et al., 2004] and faceted browsing. Moreover, we can also use the extracted social network for assigning more appropriate people-tags to users. For example, if two users collaborate closely with each other, we may put a default strong connection between them. We can also use such social networks in information sharing approaches which we presented in Chapter 5.

To obtain the desired user-centric social network, we remove the objects and connect people directly together based on the events that the users performed on a specific object. From a data model point of view, we combine the RDF predicates in order to build a user-centric social network, where people are directly connected together. Figure 9.1 shows the overall approach of building a user-centric social network using an object-centric one.

The relationships between people are defined as combinations of the events on the objects. We may also make the event-based relationships transitive by enabling users to traverse across document-centric clouds. Thus, the depth of the event-based relationships is also important (i.e., moving from one cloud to another one, as illustrated in Figure 4.4). As an example, if user A has created a document which has been revised by user B and user B has read another document which has been deleted by user C, the depth of a possible relationship between user A and user B (i.e., CreateEvent;ReviseEvent[1]) is one, whereas the depth of a possible relationship between user A and user C is two (i.e., CreateEvent;ReviseEvent;ReadEvent;DeleteEvent), as two different documents were in the middle which were connected together via user B. Note

[1]We use semicolon (;) as a separator between events.

that all relationships are unidirectional.

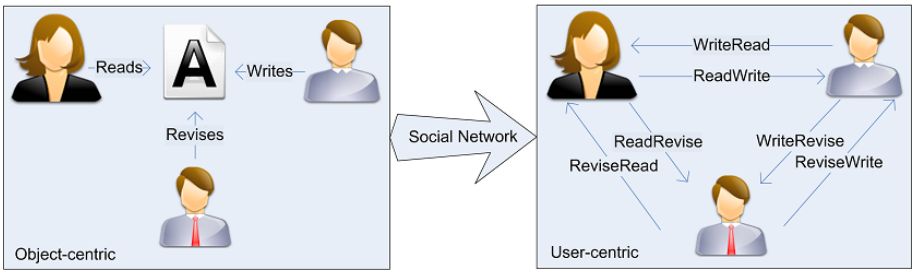

Figure 9.1: From object-centric to user-centric social network.

The extracted user-centric social network needs to be weighted to reflect strength of the links (i.e., collaboration) among personnel. We call each weight of a link in a user-centric social network a *Cooperation Index* (CI), an index that determines how close two people worked together. The higher the index, the stronger the collaboration that happened in the past between the two people. To calculate this index, we assign user-defined weights to the relationships between people and we sum up frequency of the relationships with consideration of weights. Figure 9.2 demonstrates the overall iterative approach of calculating the CIs.

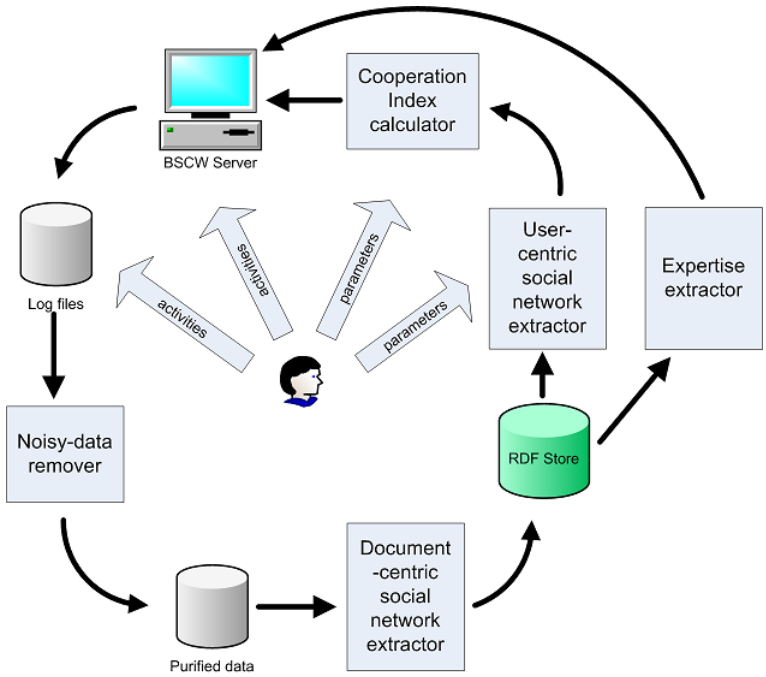

Figure 9.2: Iterative approach for calculating Cooperation Index (CI) and assigning expertise.

In the following section, we present formal definitions and a use case scenario.

9.2 Formal Definition

It is useful to formally define the relationships between people and also the CI. To do so, we need some pre-definitions and assumptions.

We assume that there exists a set of users U. n is the number of users.

$$U = \{u_1, u_2, ..., u_n\}, |U| = n$$

We assume that there exists a set of documents D. m is the number of documents.

$$D = \{d_1, d_2, ..., d_m\}, |D| = m$$

We also assume that there exists a set of events E. k is the number of events.

$$E = \{e_1, e_2, ..., e_k\}, |E| = k$$

The events are those functions that happen on a document within online shared workspace. The assumption is that u_i can perform e_k on d_j. As an example, *user A* can perform a *Read* event on *document B*.

The combination of events for depth k is defined by a cartesian product as shown below:

$$E^k = \underbrace{E \times E \times .... \times E}_{k}$$

For example:

$$E^2 = E \times E = \{e_1e_1, e_1e_2, ..., e_2e_1, e_2e_2, ..., e_ke_1, ..., e_ke_k\}$$

We define a function ($f : U \times D \longrightarrow E$) which returns all events that were performed by a user from the log file. Formally, we can define it as:

$$f(u_i, d_j) = \{e_l \mid e_l \in E\} \text{ for } u_i \in U \text{ and } d_j \in D \tag{9.1}$$

The Relationship function for depth k ($Rel_k : U \times U \longrightarrow E^{2k} (k \geq 1)$) is defined as the following[2]:

$$\forall d_y \in D, f(u_i, d_y) \neq \emptyset \wedge f(u_j, d_y) \neq \emptyset \Rightarrow Rel_k(u_a, u_z) = \underbrace{f(u_a, d_b) \times f(u_b, d_b) \times f(u_b, d_c) \times f(u_c, d_c) \times ... \times f(u_z, d_k)}_{2k} \tag{9.2}$$

For example:

$$Rel_1 : U \times U \longrightarrow E^2,$$

$$Rel_1(u_i, u_j) = Rel(u_i, u_j) = f(u_i, d_b) \times f(u_j, d_b)$$

[2]Note that k is actually the number of documents which locate between users and should be traversed, in order to reach a target user (second parameter of the function) from a source user (first parameter of the function).

Every combination of the events has a predefined or user-defined weight. The weight function ($w : E^k \longrightarrow \mathbb{R}$ for $(k \geq 2)$) is defined as follows:

$$\forall e \in E^k (k \geq 2) : w(e) = i \in \mathbb{R}$$

Using the previous definitions, we introduce a formal definition for the Cooperation Index (CI). The CI between u_i and u_j for depth k is the sum of weights of all relationships between u_i and u_j in depth k. Formally, we can define it as:

$$CooperationIndex_k(u_i, u_j) = \sum_{e \in Rel_k(u_i, u_j)} w(e)(k \geq 1) \tag{9.3}$$

For example:

$$CooperationIndex_1(u_i, u_j) = CooperationIndex(u_i, u_j)$$
$$= \sum_{e \in Rel(u_i, u_j)} w(e)$$

Finally, an optional step could be a further *normalisation*, in order to bound all CIs (e.g., between 0 and 100).

9.3 Use Case

In this section, we provide an example for calculating the CI (for depth one, i.e., *CooperationIndex*) from a sample log file. In our example, we have four users: Alice, Bob, Tom, and Mary. They work together using the OrbiTeam BSCW shared workspace and they take part in some document-based events (e.g., read, create) which then OrbiTeam BSCW exports in CSV format. For simplicity, let us consider 20 records of log file from 10-March-2007 to 19-March-2007. Listing 9.1 shows this piece of the log file.

Listing 9.1: A sample log file.

```
2007-03-10 17:17:55;337777;CreateEvent;11;D1;2;Bob;
2007-03-10 17:18:05;338771;CreateEvent;12;D2;3;Tom;
2007-03-11 17:19:15;333481;CreateEvent;13;D3;4;Mary;
2007-03-11 17:20:35;333281;ReadEvent;12;D2;1;Alice;
2007-03-12 12:17:25;336681;ReadEvent;12;D2;1;Alice;
2007-03-13 13:17:58;334381;ReadEvent;11;D1;1;Alice;
2007-03-14 17:13:52;344423;ReadEvent;12;D2;1;Alice;
2007-03-15 19:17:35;355662;ReadEvent;11;D1;1;Alice;
2007-03-16 09:13:22;335481;ReadEvent;13;D3;2;Bob;
2007-03-17 12:17:56;385481;ReviseEvent;13;D3;2;Bob;
2007-03-17 10:17:45;337431;ReadEvent;12;D2;2;Bob;
2007-03-17 17:19:35;332581;ReviseEvent;12;D2;4;Mary;
2007-03-17 10:10:25;346541;ReadEvent;12;D2;4;Mary;
2007-03-18 13:25:15;312431;ReviseEvent;11;D1;4;Mary;
2007-03-18 18:11:05;323444;ReviseEvent;13;D3;3;Tom;
2007-03-18 08:10:03;332355;ReadEvent;11;D1;3;Tom;
2007-03-19 09:10:06;335534;ReviseEvent;11;D1;1;Alice;
2007-03-19 10:17:24;335325;DeleteEvent;12;D2;1;Alice;
2007-03-19 18:20:09;332345;ReviseEvent;13;D3;3;Tom;
2007-03-19 20:20:09;332395;ReviseEvent;13;D3;1;Alice;
```

Each log record/entry starts with temporal information, followed by event ID, event name, object ID, object name, user ID, and username.

The first step is to extract RDF triples from the log records. Each record is mapped to one RDF triple which we do not present here. Suppose that Mary wants to calculate her CIs (at

a depth of one) with the other three members of the workspace. Using Equation 9.1, we will have:

$$f(Mary, d_i) = \begin{cases} f(Mary, D1) = \{ReviseEvent\} \\ f(Mary, D2) = \{ReadEvent, ReviseEvent\} \\ f(Mary, D3) = \{CreateEvent\} \end{cases}$$

In the same way we can calculate $f(Bob, d_i)$, $f(Alice, d_i)$, and $f(Tom, d_i)$. As long as we have the result of f, we apply Equation 9.2 to come up with the relationships between Mary and the rest. As an example, we present here the relationships between Mary and Bob.

$$Rel_1(Mary, Bob) = \begin{cases} f(Mary, D1) \times f(Bob, D1) \\ f(Mary, D2) \times f(Bob, D2) \\ f(Mary, D3) \times f(Bob, D3) \end{cases}$$

$$f(Mary, D1) \times f(Bob, D1) =$$

$$\{ReviseEvent; CreateEvent\}$$

$$f(Mary, D2) \times f(Bob, D2) =$$

$$\{ReadEvent; ReadEvent, ReviseEvent; ReadEvent\}$$

$$f(Mary, D3) \times f(Bob, D3) =$$

$$\{CreateEvent; ReadEvent, CreateEvent; ReviseEvent\}$$

In the next step, we need to set user-defined weights for the relationships. This allows users to decide what types of relationships are more important depending on the context of the specific common project or collaboration. In other words, it is the user that decides which relationships should have more effect and influence on calculation. In our example, Mary assigns the weights presented in Listing 9.2 to her possible relationships with other members of the workspace.

Listing 9.2: Weights of the relationships.

```
ReadEvent;CreateEvent = 0
ReadEvent;ReviseEvent = 0
ReadEvent;DeleteEvent = 0
ReadEvent;ReadEvent = 0

DeleteEvent;CreateEvent = 0
DeleteEvent;ReviseEvent = 0
DeleteEvent;DeleteEvent = 0
DeleteEvent;ReadEvent = 0

CreateEvent;CreateEvent = 0
CreateEvent;ReviseEvent = 0.4
CreateEvent;DeleteEvent = 0
CreateEvent;ReadEvent = 0

ReviseEvent;CreateEvent = 0.2
ReviseEvent;ReviseEvent = 0.4
ReviseEvent;DeleteEvent = 0
ReviseEvent;ReadEvent = 0
```

As shown in Listing 9.2, some relationships have acquired the weight zero. That means they have no effect on calculating the CI.

Now, based on the relationships between Mary and others and also the weights assigned by her, we calculate the CIs by counting frequency of the relationships between Mary and others

(for depth one) with consideration of weights (see Equation 9.3). As an example, we calculate here the CI between Mary and Bob.

$CooperationIndex_1(Mary, Bob) =$
$w(ReviseEvent; CreateEvent) + w(ReadEvent; ReadEvent) +$
$w(ReviseEvent; ReadEvent) + w(CreateEvent; ReadEvent) +$
$w(CreateEvent; ReviseEvent) = 0.6$

Listing 9.3 shows the CIs.

Listing 9.3: The calculated CIs between Mary and others.

```
1 Between Mary and Tom: 0.6
2 Between Mary and Bob: 0.6
3 Between Mary and Alice: 0.8
```

The result shows that Mary and Alice had stronger collaboration, in comparison to Mary and Tom and also Mary and Bob.

9.4 Prototype

As a proof-of-concept, we developed Holmes. Holmes extracts user-centric social networks from OrbiTeam BSCW log files and calculates the CIs between users. We do not store the user-centric social network in the RDF repository as these are built *on-the-fly* using SPARQL queries. For example, Listing 9.4 demonstrates the SPARQL query for returning all people who are connected to a fictional user with ID 151, based on *CreateEvent;ReviseEvent*, i.e., people have revised a document which has been created by a user with ID 151.

Listing 9.4: A sample SPARQL query for building social network.

```
1 PREFIX deri: <http://deri.ie/holmes#>
2 CONSTRUCT {deri:151 deri:collaborate ?person}
3 WHERE {
4         deri:151 ?predicate1 ?X.
5        ?person ?predicate2 ?X.
6 FILTER
7 (?predicate1 = deri:CreateEvent && ?predicate2 = deri:ReviseEvent)
8 }
```

Figure 9.3 demonstrates some snapshots of Holmes. OrbiTeam BSCW users are able to select their usernames from a select box, assign desired weights to their relationships and calculate the CIs with the rest of the users. To reduce the overhead for users, we assigned predefined weights to all relationships. In our case, *CreateEvent;ReviseEvent* and *ReviseEvent;CreateEvent* have larger weights.

Holmes provides two Web services as well. The *Calculate Cooperation Index* service enables a user to calculate his/her CIs with others. This service accepts an XML file as input which contains user-defined weights of possible relationships. The service returns the CIs in RDF or XHTML formats. The other service is called *Handle Data* and enables the authenticated users to lodge data into the RDF repository. The Holmes prototype is accessible online[3].

[3] http://purl.oclc.org/projects/holmes

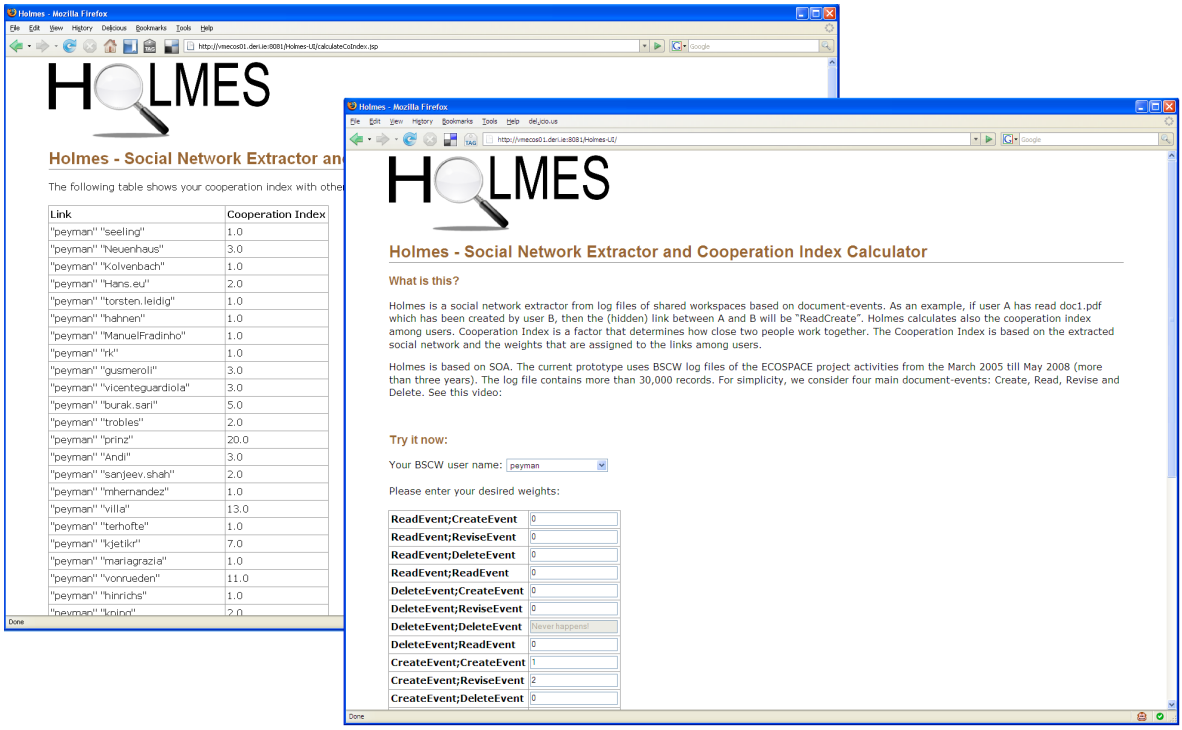

Figure 9.3: Several snapshots of Holmes: a tool for extracting weighted user-centric social networks from log files of an OrbiTeam BSCW.

Chapter 10

Appendix III – Various n-grams

The following figures present the LL-values obtained after applying the LDA with different n-gram settings on our corpus. We used the LL-values to determine the optimal topic detection settings and to compare the effect of those on the performance.

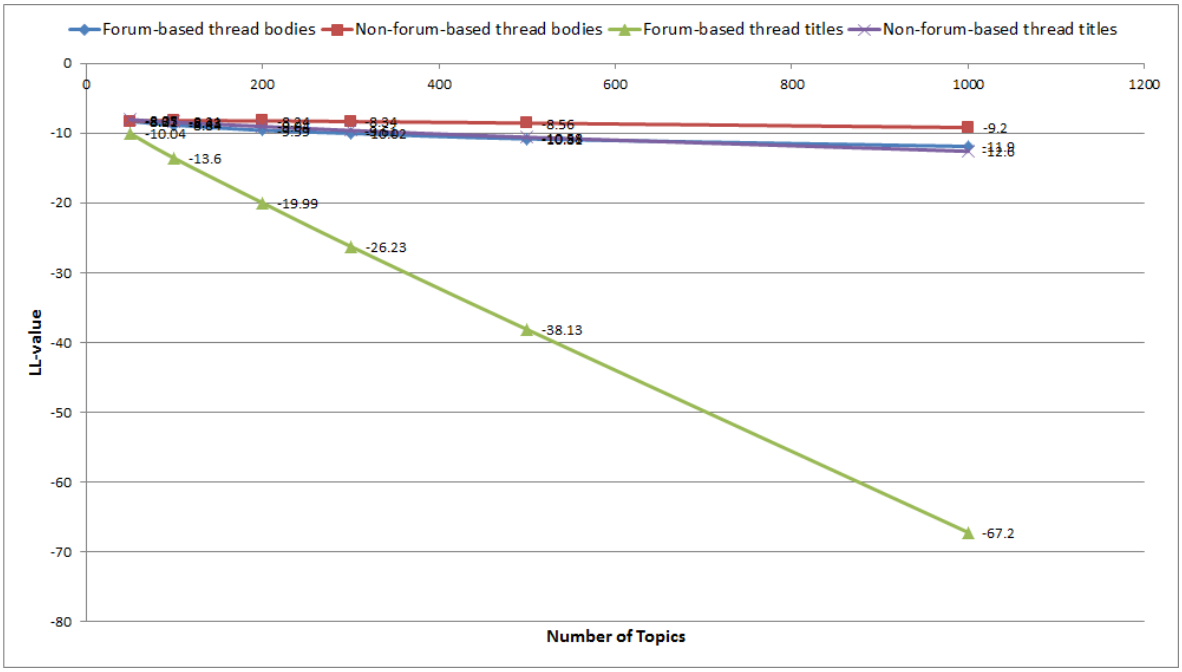

Figure 10.1: The LL-values for different number of topics after applying the LDA to our corpus. For the LDA analysis we considered both the corpus as a whole and also the divided corpus into forums. Thus, our analysis constituted four different settings as follows: forum-based thread bodies: divided thread bodies based on their forums; non-forum-based thread bodies: thread bodies as a whole; forum-based thread titles: divided thread titles based on their forums; non-forum-based thread titles: thread titles as a whole. The X-axis shows number of the topics. The Y-axis shows the LL-value after the 1000th iteration (#n-gram=1, #topics=50,100,200,300,500,1000, #keywords in each topic=20).

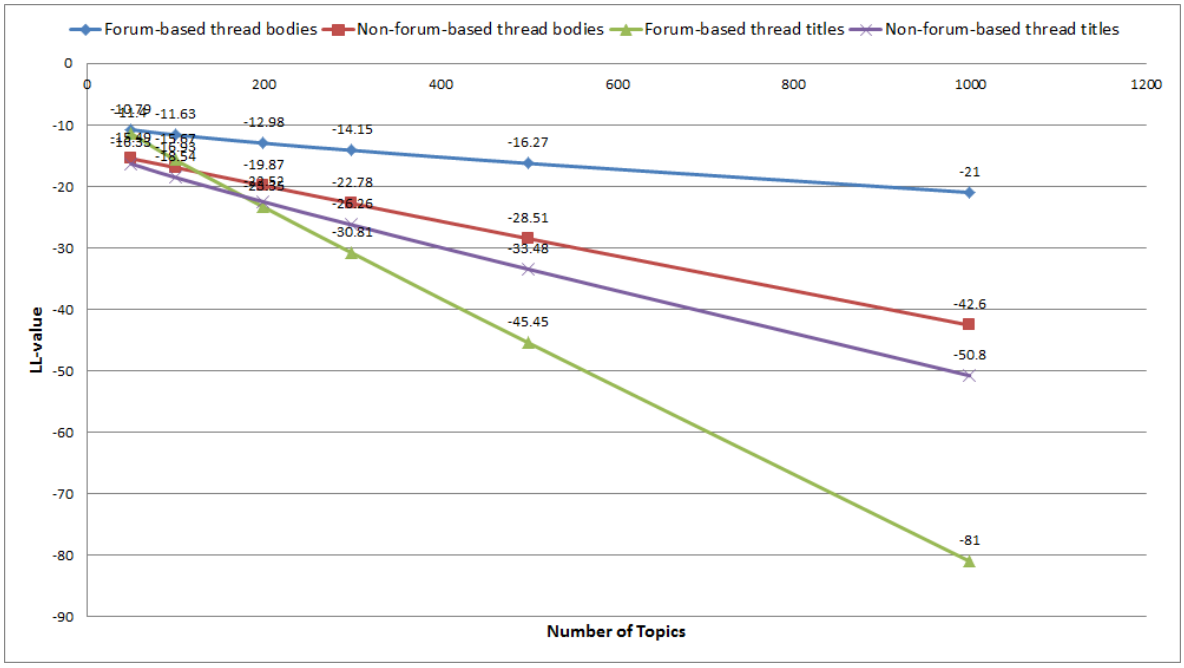

Figure 10.2: The LL-values for different number of topics after applying the LDA to our corpus. For the LDA analysis we considered both the corpus as a whole and also the divided corpus into forums. Thus, our analysis constituted four different settings as follows: forum-based thread bodies: divided thread bodies based on their forums; non-forum-based thread bodies: thread bodies as a whole; forum-based thread titles: divided thread titles based on their forums; non-forum-based thread titles: thread titles as a whole. The X-axis shows number of the topics. The Y-axis shows the LL-value after the 1000th iteration (#n-gram=2, #topics=50,100,200,300,500,1000, #keywords in each topic=20).

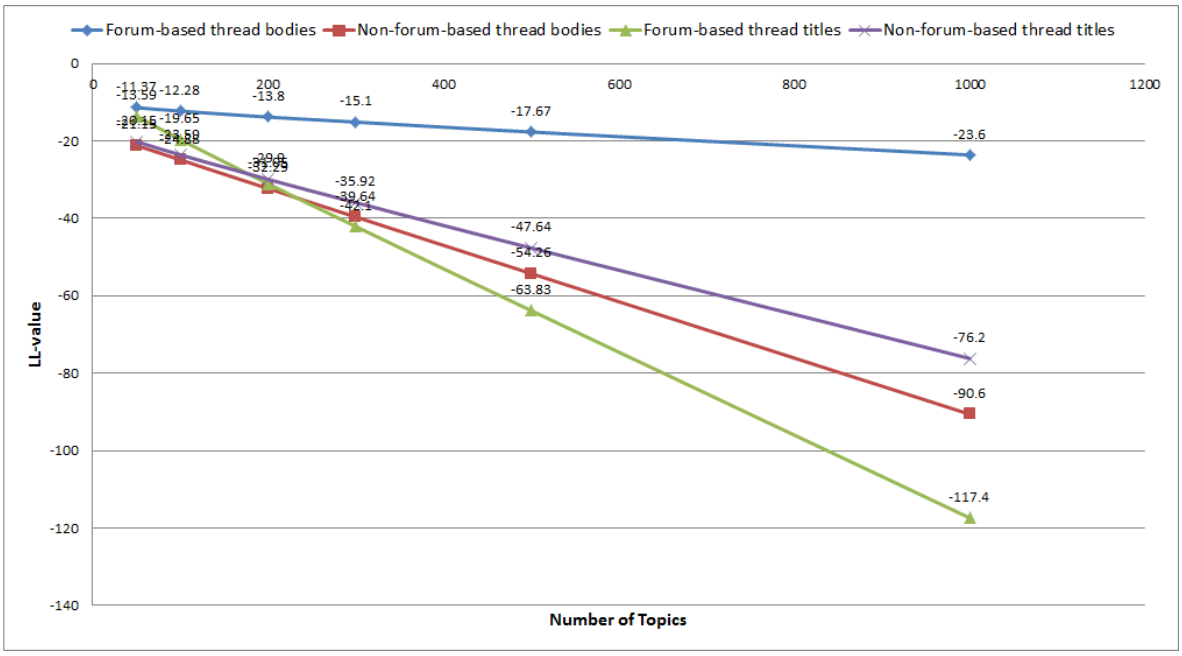

Figure 10.3: The LL-values for different number of topics after applying the LDA to our corpus. For the LDA analysis we considered both the corpus as a whole and also the divided corpus into forums. Thus, our analysis constituted four different settings as follows: forum-based thread bodies: divided thread bodies based on their forums; non-forum-based thread bodies: thread bodies as a whole; forum-based thread titles: divided thread titles based on their forums; non-forum-based thread titles: thread titles as a whole. The X-axis shows number of the topics. The Y-axis shows the LL-value after the 1000th iteration (#n-gram=3, #topics=50,100,200,300,500,1000, #keywords in each topic=20).

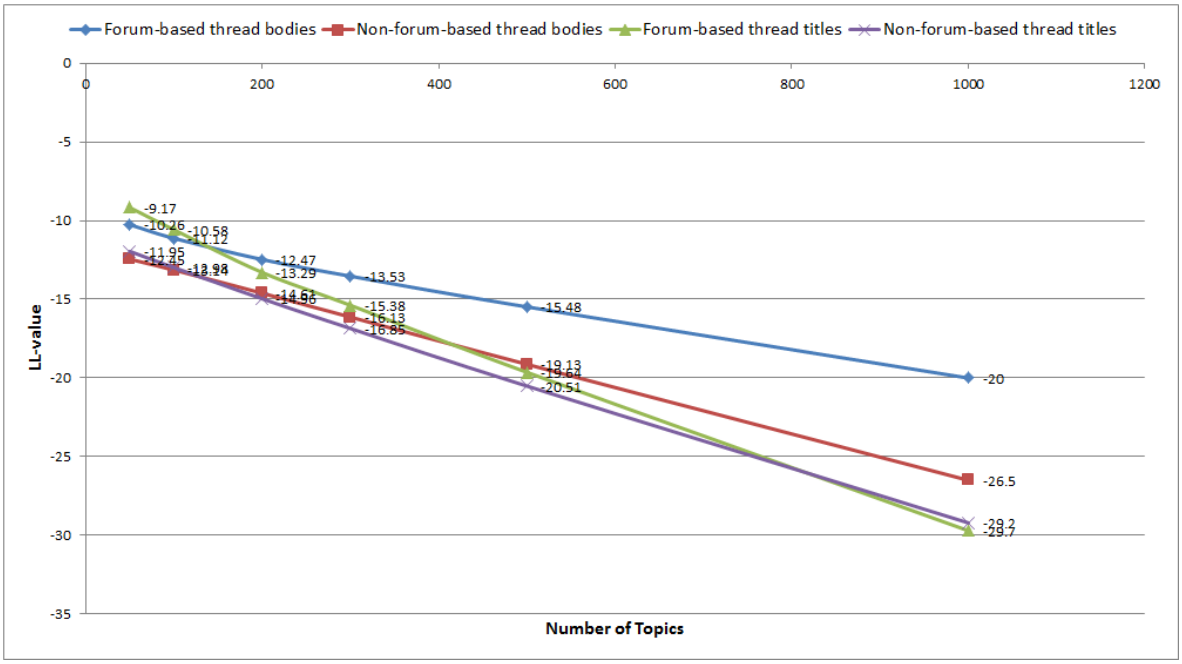

Figure 10.4: The LL-values for different number of topics after applying the LDA to our corpus. For the LDA analysis we considered both the corpus as a whole and also the divided corpus into forums. Thus, our analysis constituted four different settings as follows: forum-based thread bodies: divided thread bodies based on their forums; non-forum-based thread bodies: thread bodies as a whole; forum-based thread titles: divided thread titles based on their forums; non-forum-based thread titles: thread titles as a whole. The X-axis shows number of the topics. The Y-axis shows the LL-value after the 1000th iteration (#n-gram=1,2, #topics=50,100,200,300,500,1000, #keywords in each topic=20).

Chapter 11

Appendix IV – Detailed Evaluation Results of the First Experiment

The following tables show detailed result of our first experiment based on various methods. The first column in each table corresponds to a top-x-y% range. The second column shows the total number of threads that we could recommend a relevant expert (i.e., an expert who can provide a very helpful answer or solve an issue) in a top-x-y% range. The third column shows percentage of the number of threads in the second column against the total number of the test set threads. The fourth column shows a top-k position. The fifth column shows the total number of threads that we could recommend a relevant expert at a top-k position. The last column shows percentage of the number of threads in the fifth column against the total number of the test set threads.

As an example, consider Table 11.1. The first row in the table shows that for 274 threads (out of 492) the relevant expert popped up in top 1% position in our ranked list of relevant experts that we recommended for those threads. That is actually 55.69% of all test set threads in this case. This row in the table also shows that for 97 threads (out of 492), the relevant expert popped up in top-1 position in our ranked list of relevant experts. That is actually 19.72% of all test set threads in this case. This table also shows that for 492-453=39 threads, we could not recommend any relevant experts.

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	274	55.69%	Top 1	97	19.72%
Top 1-5%	353	71.75%	Top 5	144	29.27%
Top 1-10%	397	80.69%	Top 10	163	33.13%
Top 11-20%	27	5.49%	Top 50	272	55.28%
Top 21-30%	11	2.24%	Top 100	320	65.04%
Top 31-40%	8	1.63%	Top 200	364	73.98%
Top 41-50%	5	1.02%	Top 300	389	79.07%
Top 51-60%	2	0.41%	Top 400	411	83.54%
Top 61-70%	2	0.41%	Top 500	420	85.37%
Top 71-80%	0	0.0%	Top 600	424	86.18%
Top 81-90%	1	0.2%	Top 700	429	87.2%
Top 91-100%	0	0.0%	Top 800	432	87.8%
Total	453 / 492	92.07%	Top 900	433	88.01%
			Top 1000	434	88.21%
			Top >1000	453 / 492	92.07%

Table 11.1: Evaluation result based on NLP (i.e., linear classification) for the first post of the 492 test set threads.

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	190	38.62%	Top 1	89	18.09%
Top 1-5%	277	56.3%	Top 5	133	27.03%
Top 1-10%	320	65.04%	Top 10	168	34.15%
Top 11-20%	23	4.67%	Top 50	243	49.39%
Top 21-30%	14	2.85%	Top 100	288	58.54%
Top 31-40%	5	1.02%	Top 200	325	66.06%
Top 41-50%	6	1.22%	Top 300	345	70.12%
Top 51-60%	10	2.03%	Top 400	354	71.95%
Top 61-70%	3	0.61%	Top 500	364	73.98%
Top 71-80%	0	0.0%	Top 600	367	74.59%
Top 81-90%	2	0.41%	Top 700	370	75.2%
Top 91-100%	1	0.2%	Top 800	372	75.61%
Total	384 / 492	78.05%	Top 900	373	75.81%
			Top 1000	374	76.02%
			Top >1000	384 / 492	78.05%

Table 11.2: Evaluation result based on NLP (i.e., linear classification) for the title of the 492 test set threads.

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	301	61.18%	Top 1	98	19.92%
Top 1-5%	405	82.32%	Top 5	148	30.08%
Top 1-10%	436	88.62%	Top 10	160	32.52%
Top 11-20%	11	2.24%	Top 50	265	53.86%
Top 21-30%	7	1.42%	Top 100	308	62.6%
Top 31-40%	5	1.02%	Top 200	373	75.81%
Top 41-50%	4	0.81%	Top 300	394	80.08%
Top 51-60%	3	0.61%	Top 400	423	85.98%
Top 61-70%	0	0.0%	Top 500	431	87.6%
Top 71-80%	1	0.2%	Top 600	436	88.62%
Top 81-90%	0	0.0%	Top 700	439	89.23%
Top 91-100%	0	0.0%	Top 800	439	89.23%
Total	467 / 492	94.92%	Top 900	439	89.23%
			Top 1000	444	90.24%
			Top >1000	467 / 492	94.92%

Table 11.3: Evaluation result based on LDA for the first post of the 492 test set threads (n-gram=1,2,3, #topics=50, #keywords in each topic=20).

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	297	60.37%	Top 1	98	19.92%
Top 1-5%	400	81.3%	Top 5	147	29.88%
Top 1-10%	433	88.01%	Top 10	162	32.93%
Top 11-20%	12	2.44%	Top 50	262	53.25%
Top 21-30%	8	1.63%	Top 100	305	61.99%
Top 31-40%	4	0.81%	Top 200	368	74.8%
Top 41-50%	6	1.22%	Top 300	389	79.07%
Top 51-60%	1	0.2%	Top 400	424	86.18%
Top 61-70%	1	0.2%	Top 500	428	86.99%
Top 71-80%	0	0.0%	Top 600	435	88.41%
Top 81-90%	0	0.0%	Top 700	438	89.02%
Top 91-100%	0	0.0%	Top 800	439	89.23%
Total	465 / 492	94.51%	Top 900	441	89.63%
			Top 1000	442	89.84%
			Top >1000	465 / 492	94.51%

Table 11.4: Evaluation result based on LDA for the first post of the 492 test set threads (n-gram=1,2,3, #topics=100, #keywords in each topic=20).

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	300	60.98%	Top 1	99	20.12%
Top 1-5%	394	80.08%	Top 5	146	29.67%
Top 1-10%	433	88.01%	Top 10	160	32.52%
Top 11-20%	10	2.03%	Top 50	264	53.66%
Top 21-30%	9	1.83%	Top 100	309	62.8%
Top 31-40%	2	0.41%	Top 200	368	74.8%
Top 41-50%	4	0.81%	Top 300	388	78.86%
Top 51-60%	2	0.41%	Top 400	418	84.96%
Top 61-70%	2	0.41%	Top 500	429	87.2%
Top 71-80%	1	0.2%	Top 600	436	88.62%
Top 81-90%	0	0.0%	Top 700	437	88.82%
Top 91-100%	0	0.0%	Top 800	437	88.82%
Total	463 / 492	94.11%	Top 900	439	89.23%
			Top 1000	440	89.43%
			Top >1000	463 / 492	94.11%

Table 11.5: Evaluation result based on LDA for the first post of the 492 test set threads (n-gram=1,2,3, #topics=200, #keywords in each topic=20).

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	295	59.96%	Top 1	99	20.12%
Top 1-5%	399	81.1%	Top 5	148	30.08%
Top 1-10%	434	88.21%	Top 10	161	32.72%
Top 11-20%	9	1.83%	Top 50	260	52.85%
Top 21-30%	10	2.03%	Top 100	314	63.82%
Top 31-40%	3	0.61%	Top 200	369	75.0%
Top 41-50%	2	0.41%	Top 300	392	79.67%
Top 51-60%	3	0.61%	Top 400	423	85.98%
Top 61-70%	2	0.41%	Top 500	427	86.79%
Top 71-80%	0	0.0%	Top 600	433	88.01%
Top 81-90%	1	0.2%	Top 700	438	89.02%
Top 91-100%	0	0.0%	Top 800	439	89.23%
Total	464 / 492	94.31%	Top 900	440	89.43%
			Top 1000	442	89.84%
			Top >1000	464 / 492	94.31%

Table 11.6: Evaluation result based on LDA for the first post of the 492 test set threads (n-gram=1,2,3, #topics=300, #keywords in each topic=20).

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	298	60.57%	Top 1	97	19.72%
Top 1-5%	399	81.1%	Top 5	148	30.08%
Top 1-10%	428	86.99%	Top 10	163	33.13%
Top 11-20%	11	2.24%	Top 50	263	53.46%
Top 21-30%	10	2.03%	Top 100	310	63.01%
Top 31-40%	8	1.63%	Top 200	370	75.2%
Top 41-50%	3	0.61%	Top 300	391	79.47%
Top 51-60%	1	0.2%	Top 400	420	85.37%
Top 61-70%	2	0.41%	Top 500	424	86.18%
Top 71-80%	0	0.0%	Top 600	431	87.6%
Top 81-90%	0	0.0%	Top 700	434	88.21%
Top 91-100%	0	0.0%	Top 800	434	88.21%
Total	463 / 492	94.11%	Top 900	434	88.21%
			Top 1000	435	88.41%
			Top >1000	463 / 492	94.11%

Table 11.7: Evaluation result based on LDA for the first post of the 492 test set threads (n-gram=1,2,3, #topics=500, #keywords in each topic=20).

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	299	60.77%	Top 1	98	19.92%
Top 1-5%	402	81.71%	Top 5	149	30.28%
Top 1-10%	437	88.82%	Top 10	158	32.11%
Top 11-20%	10	2.03%	Top 50	266	54.07%
Top 21-30%	10	2.03%	Top 100	308	62.6%
Top 31-40%	7	1.42%	Top 200	370	75.2%
Top 41-50%	1	0.2%	Top 300	392	79.67%
Top 51-60%	0	0.0%	Top 400	426	86.59%
Top 61-70%	1	0.2%	Top 500	431	87.6%
Top 71-80%	1	0.2%	Top 600	435	88.41%
Top 81-90%	0	0.0%	Top 700	439	89.23%
Top 91-100%	0	0.0%	Top 800	440	89.43%
Total	467 / 492	94.92%	Top 900	440	89.43%
			Top 1000	442	89.84%
			Top >1000	467 / 492	94.92%

Table 11.8: Evaluation result based on LDA for the first post of the 492 test set threads (n-gram=1,2,3, #topics=1000, #keywords in each topic=20).

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	293	59.55%	Top 1	98	19.92%
Top 1-5%	398	80.89%	Top 5	143	29.07%
Top 1-10%	437	88.82%	Top 10	160	32.52%
Top 11-20%	8	1.63%	Top 50	258	52.44%
Top 21-30%	11	2.24%	Top 100	297	60.37%
Top 31-40%	5	1.02%	Top 200	364	73.98%
Top 41-50%	1	0.2%	Top 300	391	79.47%
Top 51-60%	3	0.61%	Top 400	420	85.37%
Top 61-70%	1	0.2%	Top 500	433	88.01%
Top 71-80%	0	0.0%	Top 600	437	88.82%
Top 81-90%	3	0.61%	Top 700	441	89.63%
Top 91-100%	0	0.0%	Top 800	443	90.04%
Total	469 / 492	95.33%	Top 900	444	90.24%
			Top 1000	444	90.24%
			Top >1000	469 / 492	95.33%

Table 11.9: Evaluation result based on LDA for the title of the 492 test set threads (n-gram=1,2,3, #topics=50, #keywords in each topic=20).

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	293	59.55%	Top 1	99	20.12%
Top 1-5%	399	81.1%	Top 5	143	29.07%
Top 1-10%	430	87.4%	Top 10	157	31.91%
Top 11-20%	13	2.64%	Top 50	258	52.44%
Top 21-30%	7	1.42%	Top 100	299	60.77%
Top 31-40%	5	1.02%	Top 200	366	74.39%
Top 41-50%	4	0.81%	Top 300	389	79.07%
Top 51-60%	1	0.2%	Top 400	423	85.98%
Top 61-70%	2	0.41%	Top 500	428	86.99%
Top 71-80%	1	0.2%	Top 600	433	88.01%
Top 81-90%	1	0.2%	Top 700	436	88.62%
Top 91-100%	1	0.2%	Top 800	438	89.02%
Total	465 / 492	94.51%	Top 900	439	89.23%
			Top 1000	442	89.84%
			Top >1000	465 / 492	94.51%

Table 11.10: Evaluation result based on LDA for the title of the 492 test set threads (n-gram=1,2,3, #topics=100, #keywords in each topic=20).

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	287	58.33%	Top 1	98	19.92%
Top 1-5%	387	78.66%	Top 5	140	28.46%
Top 1-10%	428	86.99%	Top 10	158	32.11%
Top 11-20%	11	2.24%	Top 50	256	52.03%
Top 21-30%	8	1.63%	Top 100	297	60.37%
Top 31-40%	6	1.22%	Top 200	359	72.97%
Top 41-50%	2	0.41%	Top 300	387	78.66%
Top 51-60%	2	0.41%	Top 400	410	83.33%
Top 61-70%	0	0.0%	Top 500	424	86.18%
Top 71-80%	1	0.2%	Top 600	431	87.6%
Top 81-90%	2	0.41%	Top 700	433	88.01%
Top 91-100%	0	0.0%	Top 800	435	88.41%
Total	460 / 492	93.5%	Top 900	435	88.41%
			Top 1000	436	88.62%
			Top >1000	460 / 492	93.5%

Table 11.11: Evaluation result based on LDA for the title of the 492 test set threads (n-gram=1,2,3, #topics=200, #keywords in each topic=20).

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	286	58.13%	Top 1	98	19.92%
Top 1-5%	386	78.46%	Top 5	142	28.86%
Top 1-10%	427	86.79%	Top 10	157	31.91%
Top 11-20%	16	3.25%	Top 50	258	52.44%
Top 21-30%	9	1.83%	Top 100	298	60.57%
Top 31-40%	4	0.81%	Top 200	364	73.98%
Top 41-50%	3	0.61%	Top 300	393	79.88%
Top 51-60%	1	0.2%	Top 400	421	85.57%
Top 61-70%	1	0.2%	Top 500	430	87.4%
Top 71-80%	0	0.0%	Top 600	432	87.8%
Top 81-90%	3	0.61%	Top 700	437	88.82%
Top 91-100%	0	0.0%	Top 800	439	89.23%
Total	464 / 492	94.31%	Top 900	440	89.43%
			Top 1000	442	89.84%
			Top >1000	464 / 492	94.31%

Table 11.12: Evaluation result based on LDA for the title of the 492 test set threads (n-gram=1,2,3, #topics=300, #keywords in each topic=20).

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	280	56.91%	Top 1	95	19.31%
Top 1-5%	384	78.05%	Top 5	142	28.86%
Top 1-10%	416	84.55%	Top 10	160	32.52%
Top 11-20%	16	3.25%	Top 50	259	52.64%
Top 21-30%	7	1.42%	Top 100	299	60.77%
Top 31-40%	5	1.02%	Top 200	356	72.36%
Top 41-50%	2	0.41%	Top 300	381	77.44%
Top 51-60%	1	0.2%	Top 400	407	82.72%
Top 61-70%	1	0.2%	Top 500	419	85.16%
Top 71-80%	2	0.41%	Top 600	426	86.59%
Top 81-90%	3	0.61%	Top 700	429	87.2%
Top 91-100%	0	0.0%	Top 800	430	87.4%
Total	453 / 492	92.07%	Top 900	431	87.6%
			Top 1000	432	87.8%
			Top >1000	453 / 492	92.07%

Table 11.13: Evaluation result based on LDA for the title of the 492 test set threads (n-gram=1,2,3, #topics=500, #keywords in each topic=20).

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	280	56.91%	Top 1	96	19.51%
Top 1-5%	377	76.63%	Top 5	139	28.25%
Top 1-10%	414	84.15%	Top 10	152	30.89%
Top 11-20%	14	2.85%	Top 50	256	52.03%
Top 21-30%	4	0.81%	Top 100	304	61.79%
Top 31-40%	7	1.42%	Top 200	355	72.15%
Top 41-50%	5	1.02%	Top 300	384	78.05%
Top 51-60%	1	0.2%	Top 400	406	82.52%
Top 61-70%	0	0.0%	Top 500	420	85.37%
Top 71-80%	1	0.2%	Top 600	423	85.98%
Top 81-90%	1	0.2%	Top 700	424	86.18%
Top 91-100%	2	0.41%	Top 800	426	86.59%
Total	449 / 492	91.26%	Top 900	427	86.79%
			Top 1000	430	87.4%
			Top >1000	449 / 492	91.26%

Table 11.14: Evaluation result based on LDA for the title of the 492 test set threads (n-gram=1,2,3, #topics=1000, #keywords in each topic=20).

Chapter 12

Appendix V – Detailed Evaluation Results of the Second Experiment

The following tables show detailed result of our second experiment based on various methods. The first column in each table corresponds to a top-x-y% range. The second column shows the total number of threads that we could recommend a relevant expert (i.e., an expert who can provide a very helpful answer or solve an issue) in a top-x-y% range. The third column shows percentage of the number of threads in the second column against the total number of the test set threads. The fourth column shows a top-k position. The fifth column shows the total number of threads that we could recommend a relevant expert at a top-k position. The last column shows percentage of the number of threads in the fifth column against the total number of the test set threads.

As an example, consider Table 12.1. The first row in the table shows that for 1596 threads (out of 2828) the relevant expert popped up in top 1% position in our ranked list of relevant experts that we recommended for those threads. That is actually 56.44% of all test set threads in this case. This row in the table also shows that for 562 threads (out of 2828), the relevant expert popped up in top-1 position in our ranked list of relevant experts. That is actually 19.87% of all test set threads in this case. This table also shows that for 2828-2549=279 threads, we could not recommend any relevant experts.

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	1596	56.44%	Top 1	562	19.87%
Top 1-5%	1997	70.62%	Top 5	869	30.73%
Top 1-10%	2269	80.23%	Top 10	1037	36.67%
Top 11-20%	135	4.77%	Top 50	1556	55.02%
Top 21-30%	53	1.87%	Top 100	1763	62.34%
Top 31-40%	31	1.1%	Top 200	1946	68.81%
Top 41-50%	14	0.5%	Top 300	2080	73.55%
Top 51-60%	14	0.5%	Top 400	2193	77.55%
Top 61-70%	9	0.32%	Top 500	2294	81.12%
Top 71-80%	11	0.39%	Top 600	2347	82.99%
Top 81-90%	6	0.21%	Top 700	2385	84.34%
Top 91-100%	7	0.25%	Top 800	2413	85.33%
Total	2549 / 2828	90.13%	Top 900	2433	86.03%
			Top 1000	2446	86.49%
			Top >1000	2549 / 2828	90.13%

Table 12.1: Evaluation result based on NLP (i.e., linear classification) for the first post of the 2828 test set threads.

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	981	34.69%	Top 1	483	17.08%
Top 1-5%	1348	47.67%	Top 5	754	26.66%
Top 1-10%	1526	53.96%	Top 10	893	31.58%
Top 11-20%	143	5.06%	Top 50	1324	46.82%
Top 21-30%	75	2.65%	Top 100	1483	52.44%
Top 31-40%	41	1.45%	Top 200	1614	57.07%
Top 41-50%	28	0.99%	Top 300	1685	59.58%
Top 51-60%	18	0.64%	Top 400	1728	61.1%
Top 61-70%	16	0.57%	Top 500	1777	62.84%
Top 71-80%	13	0.46%	Top 600	1808	63.93%
Top 81-90%	10	0.35%	Top 700	1827	64.6%
Top 91-100%	16	0.57%	Top 800	1836	64.92%
Total	1886 / 2828	66.69%	Top 900	1840	65.06%
			Top 1000	1847	65.31%
			Top >1000	1886 / 2828	66.69%

Table 12.2: Evaluation result based on NLP (i.e., linear classification) for the title of the 2828 test set threads.

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	1708	60.4%	Top 1	555	19.63%
Top 1-5%	2091	73.94%	Top 5	872	30.83%
Top 1-10%	2372	83.88%	Top 10	1019	36.03%
Top 11-20%	106	3.75%	Top 50	1543	54.56%
Top 21-30%	35	1.24%	Top 100	1746	61.74%
Top 31-40%	21	0.74%	Top 200	1960	69.31%
Top 41-50%	11	0.39%	Top 300	2093	74.01%
Top 51-60%	11	0.39%	Top 400	2220	78.5%
Top 61-70%	7	0.25%	Top 500	2334	82.53%
Top 71-80%	9	0.32%	Top 600	2389	84.48%
Top 81-90%	6	0.21%	Top 700	2418	85.5%
Top 91-100%	9	0.32%	Top 800	2443	86.39%
Total	2587 / 2828	91.48%	Top 900	2457	86.88%
			Top 1000	2466	87.2%
			Top >1000	2587 / 2828	91.48%

Table 12.3: Evaluation result based on LDA for the first post of the 2828 test set threads (n-gram=1,2,3, #topics=200, #keywords in each topic=20).

Top-x-y%	#Threads	Percentage	Top-k	#Threads	Percentage
Top 1%	1660	58.7%	Top 1	556	19.66%
Top 1-5%	2042	72.21%	Top 5	887	31.36%
Top 1-10%	2322	82.11%	Top 10	1009	35.68%
Top 11-20%	145	5.13%	Top 50	1532	54.17%
Top 21-30%	39	1.38%	Top 100	1746	61.74%
Top 31-40%	26	0.92%	Top 200	1959	69.27%
Top 41-50%	10	0.35%	Top 300	2081	73.59%
Top 51-60%	11	0.39%	Top 400	2179	77.05%
Top 61-70%	7	0.25%	Top 500	2318	81.97%
Top 71-80%	4	0.14%	Top 600	2370	83.8%
Top 81-90%	11	0.39%	Top 700	2413	85.33%
Top 91-100%	10	0.35%	Top 800	2426	85.79%
Total	2585 / 2828	91.41%	Top 900	2439	86.24%
			Top 1000	2455	86.81%
			Top >1000	2585 / 2828	91.41%

Table 12.4: Evaluation result based on LDA for the title of the 2828 test set threads (n-gram=1,2,3, #topics=200, #keywords in each topic=20).

Chapter 13

Appendix VI – AnBAC Guidelines

In the following, we present implementation and privacy guidelines of the AnBAC model.

- Guideline 1: Action(s) may originate from an ontology that models the possible events (functions) that can happen on a resource.
- Guideline 2: It is up to the implementer to decide how to enable the Action(s).
- Guideline 3: Annotation(s) may not be case-sensitive.
- Guideline 4: Policy(ies) may support mathematical operators (e.g., ternary operators, logical operators, relational operators).
- Guideline 5: The shortest path from one Person to another Person may be calculated using various methods and algorithms (e.g., Dijkstra's algorithm).
- Guideline 6: It is up to the implementer to define the skeleton and framework in which an Implicit Annotation may be defined and/or fetched.
- Guideline 7: It is up to the implementer to enable Person(s) to *prioritise* the Policy(ies) since the order in which Policy(ies) are considered may be important for the users.
- Guideline 8: It is up to the implementer to choose (an) appropriate and domain-specific vocabulary(ies) that can be used for Person Annotation. It is also up to the implementer to recommend or propose suitable Annotation(s) from the vocabulary(ies) to the users.
- Guideline 9: It is up to the implementer to choose an appropriate mechanism to prompt users for the availability of a new Resource (e.g., RSS feed, instant messaging).
- Guideline 10: A Policy can be expressed or represented using the following syntax: *(Resource Annotation: [Attribute: Value,...]).... : Resource Context,... ; (Person Annotation: Distance: [Attribute: Value, ...]).... : Person Context, ... ; Action,... ; Policy Context,....* In the above syntax, "..." denotes that the same element may occur more than once. For simplicity, mathematical operators are not included in the Policy representation in the above syntax; nonetheless, mathematical operators may be used in Annotation(s), Action(s), Context(s), Attribute(s) and Value(s). Note that policy

representation languages are not our contribution and the above syntax is just a suggestion and can be replaced with existing policy languages, such as PROTUNE framework [Coi et al., 2008]. Policy(ies) can be merged or aggregated to simplify the representation of those Policy(ies). A simplified Policy may have more than two Annotations(s).

- Guideline 11: If the Value of an Attribute has not been explicitly mentioned in a Policy, then the minimum Value may be considered.
- Guideline 12: Semantic match-making of Annotation(s) may be considered in the Policy evaluation (specially for commonly-agreed vocabulary(ies)) (e.g., a *closeFriend* is a *friend* as well).
- Guideline 13: It is recommended to disable Attribute(s) for Distance(s) greater than one. If the implementer decides to enable Attribute(s) for Distance(s) greater than one, then it should be considered for only commonly-agreed vocabulary(ies) with commonly-agreed meanings, as Distance is enabled for only commonly-agreed vocabulary(ies). In this case, it is necessary to agree on Attribute(s) as well as their meanings. When users agree on Annotation(s) and Attribute(s), it is the responsibility of the implementer to select the appropriate algorithm to calculate Value(s) in transitive relationships.
- Guideline 14: It is up to the implementer to define a domain-dependant context model.
- Guideline 15: It is up to the implementer to define the skeleton and framework in which an Implicit Context may be defined and/or fetched.
- Guideline 16: Distance(s) and Action(s) could also have Explicit or Implicit types (i.e., values). However, it may not make sense to "dynamically" assign Action(s) and Distance(s), as this may increase the complexity of the model and also decrease the security and privacy.
- Guideline 17: The default Distance for Person Annotation(s) in the Policy(ies) may be "one".
- Guideline 18: The default Action may be "Read".
- Guideline 19: The default mathematical operator between Attribute(s) and Value(s) may be "equality".
- Guideline 20: A user may have the choice to make a Resource publicly available for the whole network (i.e., all Person(s) with any Annotation(s)) within the scope of a specific Distance.
- Guideline 21: It is up to the implementer to remove some part of the Annotation-Based Access Control model (e.g., Attribute and Value) for minor requirements – see Section 5.6 as an example.
- Guideline 22: A Person may opt-out for himself/herself or his/her contacts of being included in the shortest path algorithm. In other words, a Person can control scope of the resources that are shared with friends of friends through friends via Distance.

- Guideline 23: All Resource Annotation(s) and part of the Person Annotation(s) that are not based on a commonly-agreed vocabulary should be private, unless a Person agrees to share them with others. The reason that commonly-agreed Person Annotation(s) can not be private is due to the fact that there is a so-called "hack" to somehow reveal them. For example, suppose that Person X has annotated Person Y with an Explicit Annotation K (originated from a commonly-agreed vocabulary). Person Y has also annotated Person Z with an "unknown" Annotation. If Person X shares Resource W with those Person(s) that have been annotated with K and are in a Distance of two (from Person X), then if Person Z has access to Resource W via Person X, we may conclude that Person Z has been also annotated with Explicit Annotation K by Person Y.
- Guideline 24: Unidirectional connection between two Person(s) may require acceptance confirmation by the target Person.

Bibliography

[Adamic et al., 2008] Adamic, L. A., Zhang, J., Bakshy, E., and Ackerman, M. S. (2008). Knowledge sharing and yahoo answers: Everyone knows something. In *proceedings of WWW '08*, pages 665–674. ACM.

[Agarwal, 2009] Agarwal, A. (2009). Web 3.0 concepts explained in plain english. http://www.labnol.org/internet/web-3-concepts-explained/8908/. Online - Last visited: 30-Jan-2012.

[Alotaiby and Chen, 2004] Alotaiby, F. T. and Chen, J. X. (2004). A model for team-based access control (TMAC 2004). In *ITCC '04: Proceedings of the International Conference on Information Technology: Coding and Computing*, volume 2. IEEE Computer Society.

[Ames and Naaman, 2007] Ames, M. and Naaman, M. (2007). Why we tag: Motivations for annotation in mobile and online media. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 971–980. ACM.

[Anderson, 2007] Anderson, P. (2007). What is web 2.0? ideas, technologies and implications for education. Technical report, JISC, UK.

[Angeletou et al., 2009] Angeletou, S., Sabou, M., and Motta, E. (2009). Folksonomy enrichment and search. In *ESWC '09*, pages 801–805. Springer-Verlag.

[Balog et al., 2006] Balog, K., Azzopardi, L., and de Rijke, M. (2006). Formal models for expert finding in enterprise corpora. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 43–50. ACM.

[Balog and de Rijke, 2006] Balog, K. and de Rijke, M. (2006). Finding experts and their details in e-mail corpora. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 1035–1036. ACM.

[Barkley et al., 1999] Barkley, J., Beznosov, K., and Upppal, J. (1999). Supporting relationships in access control using role based access control. In *proceedings of the Fourth ACM Workshop on Role-Based Access Control*, pages 55–65. ACM.

[Bazire and Brézillon, 2005] Bazire, M. and Brézillon, P. (2005). Understanding context before using it. In *proceedings of the 5th International and Interdisciplinary Conference on Modeling and Using Context*, pages 29–40. Springer.

[Bergman et al., 2003] Bergman, O., Beyth-Marom, R., and Nachmias, R. (2003). The user-subjective approach to personal information management systems. *Journal of the American Society for Information Science and Technology*, 54(9):872–878.

[Bernal, 2009] Bernal, J. (2009). *Web 2.0 and Social Networking for the Enterprise: Guidelines and Examples for Implementation and Management Within Your Organization*. IBM Press.

[Berners-Lee et al., 2001] Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The semantic web, a new form of web content that is meaningful to computers will unleash a revolution of new possibilities. *Scientific American*, 284(5):34–43.

[Bernstein et al., 2009] Bernstein, M., Tan, D. S., Smith, G., Czerwinski, M., and Horvitz, E. (2009). Collabio: A game for annotating people within social networks. In *UIST '09: Proceedings of the 22nd annual ACM symposium on User interface software and technology*, pages 97–100. ACM.

[Bernstein et al., 2010] Bernstein, M. S., Marcus, A., Karger, D. R., and Miller, R. C. (2010). Enhancing directed content sharing on the web. In *CHI '10*, pages 971–980. ACM.

[Bischoff et al., 2008] Bischoff, K., Firan, C. S., Nejdl, W., and Paiu, R. (2008). Can all tags be used for search? In *CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management*, pages 193–202. ACM.

[Blei et al., 2003] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(January):993–1022.

[Blood, 2002] Blood, R. (2002). *The Weblog Handbook: Practical Advice on Creating and Maintaining Your Blog*. Basic Books.

[Bordea, 2010] Bordea, G. (2010). Concept extraction applied to the task of expert finding. In *proceedings of PhD symposium, in conjuction with Extended Semantic Web Conference (ESWC '10)*, pages 451–456. Springer.

[Boyd et al., 2010] Boyd, D., Golder, S., and Lotan, G. (2010). Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. In *43rd Hawaii International Conference on Systems Science*, pages 1–10. IEEE Computer Society.

[Breslin et al., 2006] Breslin, J. G., Decker, S., Harth, A., and Bojars, U. (2006). SIOC: an approach to connect web based communities. *International Journal of Web Based Communities*, 2(2):133–142.

[Brickley, 2006] Brickley, D. (2006). FOAFCorp - corporate friends of friends. http://rdfweb.org/foafcorp/intro.html. Online - Last visited: 30-Jan-2012.

[Brooks and Churchill, 2010] Brooks, A. L. and Churchill, E. F. (2010). Tune in, tweet on, twit out: Information snacking on twitter. In *CHI '10 Workshop on Microblogging*. ACM.

[Bush, 1945] Bush, V. (1945). As we may think. *Atlantic Monthly*, 176(1):641–649.

[Callison-Burch, 2009] Callison-Burch, C. (2009). Fast, cheap, and creative: Evaluating translation quality using amazon's mechanical turk. In *EMNLP '09: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 286–295. Association for Computational Linguistics.

[Campbell et al., 2003] Campbell, C. S., Maglio, P. P., Cozzi, A., and Dom, B. (2003). Expertise identification using email communications. In *CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management*, pages 528—531. ACM.

[Carminati et al., 2006a] Carminati, B., Ferrari, E., and Perego, A. (2006a). The REL-X vocabulary. http://www.dicom.uninsubria.it/dawsec/vocs/relx. Online - Last visited: 30-Jan-2012.

[Carminati et al., 2006b] Carminati, B., Ferrari, E., and Perego, A. (2006b). Rule-Based access control for social networks. In *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops*, pages 1734–1744. Springer-Verlag.

[Carminati et al., 2007] Carminati, B., Ferrari, E., and Perego, A. (2007). Private relationships in social networks. In *proceedings of the 23rd International Conference on Data Engineering Workshop*, pages 163–171. IEEE Computer Society.

[Carminati et al., 2009] Carminati, B., Ferrari, E., and Perego, A. (2009). Enforcing access control in web-based social networks. *ACM Transactions on Information and System Security*, 13(1):1–38.

[Chang et al., 2007] Chang, Y., Chang, Y., Hsu, S., and Chen, C. (2007). Social network analysis to blog-based online community. In *ICCIT '07: Proceedings of the 2007 International Conference on Convergence Information Technology*, pages 2193–2198. IEEE Computer Society.

[Chen et al., 2006] Chen, H., Shen, H., Xiong, J., Tan, S., and Cheng, X. (2006). Social network structure behind the mailing lists: ICT-IIIS at TREC 2006 expert finding track. In *proceedings of the Fifteenth Text REtrieval Conference*. National Institute of Standards and Technology (NIST).

[Chen et al., 2010] Chen, J., Nairn, R., Nelson, L., Bernstein, M., and Chi, E. (2010). Short and tweet: experiments on recommending content from information streams. In *CHI '10*, pages 1185–1194. ACM.

[Chen, 1976] Chen, P. P. (1976). The entity-relationship model: Toward a unified view of data. *ACM Transactions on Database Systems*, 1(1):9–36.

[Chiu et al., 2006] Chiu, C.-M., Hsu, M.-H., and Wang, E. T. G. (2006). Understanding knowledge sharing in virtual communities: An integration of social capital and social cognitive theories. *Decision Support Systems*, 42(3):1872–1888.

[Coi et al., 2008] Coi, J. L. D., Olmedilla, D., Bonatti, P. A., and Sauro, L. (2008). Protune: A framework for semantic web policies. In *ISWC '08: International Semantic Web Conference (Posters and Demos)*.

[Constant et al., 1996] Constant, D., Sproull, L., and Kiesler, S. (1996). The kindness of strangers: The usefulness of electronic weak ties for technical advice. *Organization Science*, 7(2):119–135.

[Croft et al., 2009] Croft, W. B., Metzler, D., and Strohman, T. (2009). *Search Engines: Information Retrieval in Practice*. Addison Wesley.

[Culotta et al., 2004] Culotta, A., Bekkerman, R., and McCallum, A. (2004). Extracting social networks and contact information from email and the web. In *proceedings of the 1st Conference on Email and Anti-Spam*.

[Davenport, 2005] Davenport, T. H. (2005). *Thinking for a Living: How to Get Better Performances And Results from Knowledge Workers*. Harvard Business Press.

[Davenport and Prusak, 1998] Davenport, T. H. and Prusak, L. (1998). *Working Knowledge: How Organizations Manage What They Know*. Harvard Business Press.

[Davies, 2011] Davies, S. (2011). Still building the memex. *Communications of the ACM*, 54(2):80–88.

[Davis and Vitiello, 2005] Davis, I. and Vitiello, E. (2005). RELATIONSHIP: a vocabulary for describing relationships between people. http://vocab.org/relationship/. Online - Last visited: 30-Jan-2012.

[Decker and Hauswirth, 2008] Decker, S. and Hauswirth, M. (2008). Enabling networked knowledge. Technical report, Digital Enterprise Research Institute, National University of Ireland, Galway, http://www.deri.ie/fileadmin/documents/DERI-TR-2008-06-25.pdf. Online - Last visited: 30-Jan-2012.

[Demartini, 2007] Demartini, G. (2007). Finding experts using wikipedia. In *proceedings of the 2nd International ISWC+ASWC Workshop on Finding Experts on the Web with Semantics*.

[Demchenko et al., 2006] Demchenko, Y., Gommans, L., Tokmakoff, A., and van Buuren, R. (2006). Policy based access control in dynamic grid-based collaborative environment. In *CTS '06: Proceedings of the International Symposium on Collaborative Technologies and Systems*, pages 64–73. IEEE Computer Society.

[DiMicco et al., 2008] DiMicco, J., Millen, D. R., Geyer, W., Dugan, C., Brownholtz, B., and Muller, M. (2008). Motivations for social networking at work. In *CSCW '08*, pages 711–720. ACM.

[Ding et al., 2005] Ding, L., Finin, T., and Joshi, A. (2005). Analyzing social networks on the semantic web. *IEEE Intelligent Systems*, 9(1).

[Dom et al., 2003] Dom, B., Eiron, I., Cozzi, A., and Zhang, Y. (2003). Graph-based ranking algorithms for e-mail expertise analysis. In *DMKD '03: Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pages 42–48. ACM.

[Downes, 2005] Downes, S. (2005). Semantic networks and social networks. *The Learning Organization: An International Journal*, 12(5):411–417.

[Ehrlich and Shami, 2010] Ehrlich, K. and Shami, N. S. (2010). Microblogging inside and outside the workplace. In *proceedings of the Fourth International Conference on Weblogs and Social Media*. The AAAI Press.

[Engelbart, 1962] Engelbart, D. C. (1962). Augmenting human intellect: A conceptual framework. Summary Report AFOSR-3223 - Contract AF49(638)-1024, SRI Project 3578, Air Force Office of Scientific Research, Stanford Research Institute.

[Farrell and Lau, 2006] Farrell, S. and Lau, T. (2006). Fringe contacts: People-tagging for the enterprise. *In proceedings of the WWW 2006 Collaborative Web Tagging Workshop*.

[Farrell et al., 2007] Farrell, S., Lau, T., Nusser, S., Wilcox, E., and Muller, M. (2007). Socially augmenting employee profiles with people-tagging. In *UIST '07*, pages 91–100. ACM.

[Ferraiolo and Kuhn, 1992] Ferraiolo, D. F. and Kuhn, D. R. (1992). Role based access control. In *15th National Computer Security Conference*.

[Finin et al., 2005] Finin, T., Ding, L., Zhou, L., and Joshi, A. (2005). Social networking on the semantic web. *The Learning Organization*, 12(5):418–435.

[Fogel and Nehmad, 2009] Fogel, J. and Nehmad, E. (2009). Internet social network communities: Risk taking, trust, and privacy concerns. *Computers in Human Behavior*, 25(1):153–160.

[Fox, 1992] Fox, M. S. (1992). The TOVE project towards a common-sense model of the enterprise. In *IEA/AIE '92: Proceedings of the 5th international conference on Industrial and engineering applications of artificial intelligence and expert systems*, pages 25–34. Springer-Verlag.

[Freedman et al., 2007] Freedman, D., Pisani, R., and Purves, R. (2007). *Statistics*. W.W. Norton & Co.

[Gan et al., 2004] Gan, J. D., DeLong, B., and Schmidt, C. (2004). MeNow-Document: a FOAF extension for defining often changing variables in FOAF. http://peoplesdns.com/schema/menow/. Online - Last visited: 30-Jan-2012.

[Gates, 2007] Gates, C. E. (2007). Access control requirements for web 2.0 security and privacy. In *proceedings of the IEEE Oakland Web 2.0 Security and Privacy Workshop, held in conjunction with the IEEE Symposium on Security and Privacy*.

[Giunchiglia et al., 2008] Giunchiglia, F., Zhang, R., and Crispo, B. (2008). Relbac: Relation based access control. Technical report, Ingegneria e Scienza dell'Informazione, University of Trento.

[Goecks and Mynatt, 2004] Goecks, J. and Mynatt, E. D. (2004). Leveraging social networks for information sharing. In *CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pages 328–331. ACM.

[Goldberg et al., 1992] Goldberg, D., Nichols, D., Oki, B. M., and Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12):61–70.

[Golder and Huberman, 2006] Golder, S. A. and Huberman, B. A. (2006). Usage patterns of collaborative tagging systems. *Journal of Information Science*, 32(2):198–208.

[Halpin et al., 2007] Halpin, H., Robu, V., and Shepherd, H. (2007). The complex dynamics of collaborative tagging. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 211–220. ACM.

[Harper et al., 2008] Harper, F. M., Raban, D., Rafaeli, S., and Konstan, J. A. (2008). Predictors of answer quality in online q&a sites. In *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 865–874. ACM.

[Hart et al., 2007] Hart, M., Johnson, R., and Stent, A. (2007). More content - less control: Access control in the web 2.0. In *proceedings of the Workshop on Web 2.0 Security and Privacy at the IEEE Symposium on Security and Privacy.*

[Herlocker et al., 2000] Herlocker, J. L., Konstan, J. A., and Riedl, J. (2000). Explaining collaborative filtering recommendations. In *CSCW '00*, pages 241–250. ACM.

[Hofmann, 1999] Hofmann, T. (1999). Probabilistic latent semantic indexing. In *proceedings of the Twenty-Second Annual International SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57. ACM.

[Hogan and Harth, 2007] Hogan, A. and Harth, A. (2007). The ExpertFinder corpus 2007 for the benchmarking and development of Expert-Finding systems. In *1st International ExpertFinder Workshop, co-located with the KnowledgeWeb General Assembly.*

[Honeycutt and Herring, 2009] Honeycutt, C. and Herring, S. C. (2009). Beyond microblogging: Conversation and collaboration via twitter. In *proceedings of the Forty-Second Hawaii International Conference on System Sciences (HICSS '42)*. IEEE Computer Society.

[Hong and Shen, 2008] Hong, D. and Shen, V. Y. (2008). Setting access permission through transitive relationship in web-based social networks. In *Social Web and Knowledge Management Workshop, co-located at the 17th World Wide Web Conference (WWW '08).*

[Hotho et al., 2006a] Hotho, A., Jaeschke, R., Schmitz, C., and Stumme, G. (2006a). Bibsonomy: A social bookmark and publication sharing system. In *proceedings of the Conceptual Structures Tool Interoperability Workshop at the 14th International Conference on Conceptual Structures.*

[Hotho et al., 2006b] Hotho, A., Jaeschke, R., Schmitz, C., and Stumme, G. (2006b). Folkrank : A ranking algorithm for folksonomies. In *LWA '06: Lernen - Wissensentdeckung - Adaptivitaet*, pages 111–114. University of Hildesheim, Institute of Computer Science.

[Hotho et al., 2006c] Hotho, A., Jaeschke, R., Schmitz, C., and Stumme, G. (2006c). Information retrieval in folksonomies: Search and ranking. In *proceedings of the 3rd European Semantic Web Conference*, pages 411–426. Springer.

[Huang et al., 2010] Huang, J., Thornton, K. M., and Efthimiadis, E. N. (2010). Conversational tagging in twitter. In *HT '10: Proceedings of the 21st ACM conference on Hypertext and hypermedia*, pages 173–178. ACM.

[Hussey, 2010] Hussey, T. (2010). *Create Your Own Blog: 6 Easy Projects to Start Blogging Like a Pro*. Sams.

[Irvine, 2008] Irvine, M. (2008). Social networking applications can pose security risks. http://www.pantagraph.com/articles/2008/04/27/news/doc48120a3d97c75432641888.txt. Online - Last visited: 30-Jan-2012.

[Jaeger and Prakash, 1996] Jaeger, T. and Prakash, A. (1996). Requirements of role-based access control for collaborative systems. In *RBAC '95: Proceedings of the first ACM Workshop on Role-based access control*, page 16. ACM.

[Järvelin and Kekäläinen, 2002] Järvelin, K. and Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446.

[Java et al., 2007] Java, A., Song, X., Finin, T., and Tseng, B. (2007). 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*, pages 56–65. ACM.

[Jung et al., 2007] Jung, H., Lee, M., Kang, I., Lee, S., and Sung, W. (2007). Finding topic-centric identified experts based on full text analysis. In *proceedings of the 2nd International ISWC+ASWC Workshop on Finding Experts on the Web with Semantics*.

[Jung and Euzenat, 2007] Jung, J. J. and Euzenat, J. (2007). Towards semantic social networks. In *ESWC '07: Proceedings of the 4th European conference on The Semantic Web*, pages 267–280. Springer-Verlag.

[Kern and Walhorn, 2005] Kern, A. and Walhorn, C. (2005). Rule support for role-based access control. In *SACMAT '05: Proceedings of the tenth ACM symposium on Access control models and technologies*, pages 130–138. ACM.

[Kim et al., 2010] Kim, D., Jo, Y., Moon, I.-C., and Oh, A. H. (2010). Analysis of twitter lists as a potential source for discovering latent characteristics of users. In *workshop on Microblogging at the ACM Conference on Human Factors in Computer Systems (CHI '10)*.

[Kim et al., 2005] Kim, H., Ramakrishna, R., and Sakurai, K. (2005). A collaborative role-based access control for trusted operating systems in distributed environment. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 88(1):270–279.

[Kobie, 2010] Kobie, N. (2010). Conrad wolfram on communicating with apps in web 3.0. http://www.itpro.co.uk/621535/q-a-conrad-wolfram-on-communicating-with-apps-in-web-3-0. Online - Last visited: 30-Jan-2012.

[Krishnamurthy et al., 2008] Krishnamurthy, B., Gill, P., and Arlitt, M. (2008). A few chirps about twitter. In *WOSP '08: Proceedings of the first workshop on Online social networks, co-located at ACM SIGCOMM 2008 Conference*, pages 19–24. ACM.

[Kruk et al., 2006] Kruk, S. R., Grzonkowski, S., Gzella, A., Woroniecki, T., and Choi, H. (2006). D-FOAF: distributed identity management with access rights delegation. In *proceedings of the Asian Semantic Web Conference (ASWC)*, pages 140–154.

[Kwak et al., 2010] Kwak, H., Lee, C., Park, H., and Moon, S. (2010). What is twitter, a social network or a news media? In *WWW '10*, pages 591–600. ACM.

[Li et al., 2005] Li, H., Zhang, X., Wu, H., and Qu, Y. (2005). Design and application of rule based access control policies. In *proceedings of the Semantic Web and Policy Workshop, held in conjunction with the 4th International Semantic Web Conference*, Galway, Ireland.

[Liu et al., 2002] Liu, P., Curson, J., and Dew, P. (2002). Exploring RDF for expertise matching within an organizational memory. In *proceedings of the 14th International Conference on Advanced Information Systems Engineering (CAiSE '02)*, pages 100–116. Springer-Verlag.

[Marlow et al., 2006] Marlow, C., Naaman, M., Boyd, D., and Davis, M. (2006). HT06, tagging paper, taxonomy, flickr, academic article, to read. In *proceedings of the seventeenth conference on Hypertext and hypermedia*, pages 31–40. ACM.

[Maron et al., 1986] Maron, M., Curry, S., and Thompson, P. (1986). An inductive search system: Theory, design, and implementation. *IEEE Transactions on Systems, Man and Cybernetics*, 16(1):21–28.

[Matsuo et al., 2004] Matsuo, Y., Hamasaki, M., Mori, J., Takeda, H., and Hasida, K. (2004). Ontological consideration on human relationship vocabulary for FOAF. In *First Workshop on Friend of a Friend, Social Networking and the Semantic Web, Galway, Ireland.*

[Mayo, 2001] Mayo, A. (2001). *Human Value of the Enterprise: Valuing People As Assets - Monitoring, Measuring, Managing.* Nicholas Brealey Publishing, 1st edition.

[McAfee, 2006] McAfee, A. P. (2006). Enterprise 2.0: The dawn of emergent collaboration. *MITSloan Management Review*, 47(3):21–28.

[Mika, 2004] Mika, P. (2004). Social networks and the semantic web. In *proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence (WI '04)*, pages 285–291. IEEE Computer Society.

[Mika, 2007] Mika, P. (2007). Ontologies are us: A unified model of social networks and semantics. *Web Semantics*, 5(1):5–15.

[Milgram, 1967] Milgram, S. (1967). The small world problem. *Psychology Today*, 1(May):61–67.

[Miller, 1995] Miller, G. A. (1995). Wordnet: A lexical database for english. *Communications of the ACM*, 38(11):39–41.

[Mori et al., 2005] Mori, J., Sugiyama, T., and Matsuo, Y. (2005). Real-world oriented information sharing using social networks. In *proceedings of the 2005 international ACM SIGGROUP conference on Supporting group work*, pages 81–84. ACM.

[Morris, 2005] Morris, M. (2005). How do users feel about technology? Technical report, Forrester Research.

[Moyer and Ahamad, 2001] Moyer, M. and Ahamad, M. (2001). Generalized Role-Based Access Control. In *ICDCS '01: Proceedings of the The 21st International Conference on Distributed Computing Systems*, page 391. IEEE Computer Society.

[Muller, 2007] Muller, M. J. (2007). Comparing tagging vocabularies among four enterprise tag-based services. In *GROUP '07: Proceedings of the 2007 international ACM conference on Supporting group work*, pages 341–350. ACM.

[Muller et al., 2007] Muller, M. J., Ehrlich, K., and Farrell, S. (2007). Social tagging and self-tagging for impression management. Technical report, IBM Watson Research Center.

[Nasirifard, 2007] Nasirifard, P. (2007). Context-aware access control for collaborative working environments based on semantic social networks. In *proceedings of the Doctorial Consortium Workshop at Sixth International and Interdisciplinary Conference on Modeling and Using Context (Context '07)*.

[Nasirifard and Hayes, 2011] Nasirifard, P. and Hayes, C. (2011). Tadvise: A twitter assistant based on twitter lists. In *proceedings of the third International Conference on Social Informatics (SocInfo '11)*, pages 153–160. Springer-Verlag.

[Nasirifard et al., 2010a] Nasirifard, P., Kinsella, S., Samp, K., and Decker, S. (2010a). Social people-tagging vs. social bookmark-tagging. In *proceedings of the 17th International Conference on Knowledge Engineering and Knowledge Management (EKAW '10)*, pages 150–162. Springer.

[Nasirifard and Peristeras, 2007] Nasirifard, P. and Peristeras, V. (2007). Leveraging access control in cscw based on user-defined and hidden semantic social networks. In *proceedings of the CSCW and the Web 2.0 Workshop at the 10th European Conference on Computer Supported Co-operative Work (ECSCW)*.

[Nasirifard and Peristeras, 2008a] Nasirifard, P. and Peristeras, V. (2008a). Annotation-based access control for e-professionals. In *proceedings of the 14th International Conference on Concurrent Enterprising*.

[Nasirifard and Peristeras, 2008b] Nasirifard, P. and Peristeras, V. (2008b). Uncle-share: Annotation-based access control for cooperative and social systems. In *OTM Conferences: Proceedings of the 3rd International Symposium on Information Security (IS '08)*, pages 1122–1130. Springer-Verlag.

[Nasirifard and Peristeras, 2009] Nasirifard, P. and Peristeras, V. (2009). Expertise extracting within online shared workspaces. In *proceedings of the WebSci'09: Society On-Line*.

[Nasirifard et al., 2009] Nasirifard, P., Peristeras, V., Hayes, C., and Decker., S. (2009). Extracting and utilizing social networks from log files of shared workspaces. In *proceedings of the 10th IFIP Working Conference on Virtual Enterprises (PRO-VE '09)*, pages 643–650. Springer.

[Nasirifard et al., 2010b] Nasirifard, P., Peristerasa, V., and Decker, S. (2010b). Annotation-based access control for collaborative information spaces. *Computers in Human Behavior*, 27(4):1352–1364.

[Neumann et al., 2005] Neumann, M., O'Murchu, I., Breslin, J., Decker, S., Hogan, D., and Macdonaill, C. (2005). Semantic social network portal for collaborative online communities. *Journal of European Industrial Training*, 29(6):472–487.

[Nurmela et al., 1999] Nurmela, K., Lehtinen, E., and Palonen, T. (1999). Evaluating CSCL log files by social network analysis. In *CSCL '99: Proceedings of the 1999 conference on Computer support for collaborative learning*. International Society of the Learning Sciences.

[Ojeda-Zapata, 2008] Ojeda-Zapata, J. (2008). *Twitter Means Business: How Microblogging Can Help or Hurt Your Company*. Happy About.

[Oliveira et al., 2008] Oliveira, B., Calado, P., and Pinto, H. S. (2008). Automatic tag suggestion based on resource contents. In *EKAW '08: Proceedings of the 16th International Conference on Knowledge Engineering and Knowledge Management*, pages 255–264. Springer-Verlag.

[Overell et al., 2009] Overell, S., Sigurbjörnsson, B., and van Zwol, R. (2009). Classifying tags using open content resources. In *WSDM '09: Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 64–73. ACM.

[pearanalytics, 2009] pearanalytics (2009). Twitter study - august 2009. http://www.pearanalytics.com/blog/wp-content/uploads/2010/05/Twitter-Study-August-2009.pdf. Online - Last visited: 30-Jan-2012.

[Periorellis and Parastatidis, 2005] Periorellis, P. and Parastatidis, S. (2005). Task-Based access control for virtual organizations. In *proceedings of the 4th international conference on Scientific Engineering of Distributed Java Applications*, pages 38–47. Springer-Verlag.

[Peters, 2009] Peters, I. (2009). *Folksonomies. Indexing and Retrieval in Web 2.0 (Knowledge and Information)*. De Gruyter, 1st edition.

[Porter, 1980] Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, 14(3):130–137.

[Priebe et al., 2006] Priebe, T., Dobmeier, W., and Kamprath, N. (2006). Supporting attribute-based access control with ontologies. In *ARES '06: Proceedings of the First International Conference on Availability, Reliability and Security (ARES '06)*, pages 465–472. IEEE Computer Society.

[QasemiZadeh et al., 2012] QasemiZadeh, B., Buitelaar, P., Chen, T., and Bordea, G. (2012). Semi-supervised technical term tagging with minimal user feedback. In *proceedings of the eighth International Conference on Language Resources and Evaluation (LREC)*.

[Raban et al., 2011] Raban, D. R., Ronen, I., and Guy, I. (2011). Acting or reacting? preferential attachment in a people-tagging system. *Journal of the American Society for Information Science and Technology*, 62(4):738–747.

[Rattenbury et al., 2007] Rattenbury, T., Good, N., and Naaman, M. (2007). Towards automatic extraction of event and place semantics from flickr tags. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 103–110. ACM.

[Razavi and Iverson, 2007] Razavi, M. N. and Iverson, L. (2007). Towards usable privacy for social software. Technical report, Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, Canada.

[Razavi and Iverson, 2009] Razavi, M. N. and Iverson, L. (2009). Improving personal privacy in social systems with people-tagging. In *proceedings of the ACM 2009 international conference on Supporting group work*, pages 11–20. ACM.

[Reed, 2001] Reed, W. J. (2001). The pareto, zipf and other power laws. *Economics Letters*, 74(1):15–19.

[Russell and Gangemi, 1991] Russell, D. and Gangemi, S. G. (1991). *Computer Security Basics*. O'Reilly Media.

[Samp and Decker, 2010] Samp, K. and Decker, S. (2010). Supporting menu design with radial layouts. In *proceedings of the International Conference on Advanced Visual Interfaces (AVI)*, pages 155–162.

[Sandhu, 1996] Sandhu, R. S. (1996). Roles versus groups. In *proceedings of the first ACM Workshop on Role-based access control*. ACM.

[Sandhu et al., 1996] Sandhu, R. S., Coynek, E. J., Feinsteink, H. L., and Youmank, C. E. (1996). Role-based access control models. *IEEE Computer*, 29(2):38–47.

[Santos-neto et al., 2009] Santos-neto, E., Condon, D., Andrade, N., Iamnitchi, A., and Ripeanu, M. (2009). Individual and social behavior in tagging systems. In *HT '09: Proceedings of the 20th ACM conference on Hypertext and Hypermedia*, pages 183–192. ACM.

[Sari et al., 2007] Sari, B. et al. (2007). Deliverable 1.1 – eprofessionals current practice and user requirements report. Technical report, Ecospace Project.

[Schafer et al., 2007] Schafer, J. B., Frankowski, D., Herlocker, J., and Sen, S. (2007). Collaborative filtering recommender systems. In *proceedings of The Adaptive Web*, volume 4321 of *LNCS*, pages 291–324. Springer-Verlag.

[Schmidt and Braun, 2008] Schmidt, A. and Braun, S. (2008). People tagging & ontology maturing: Towards collaborative competence management. In *8th International Conference on the Design of Cooperative Systems (COOP)*.

[Schutz, 2008] Schutz, A. T. (2008). Keyphrase extraction from single documents in the open domain exploiting linguistic and statistical methods. Master's thesis, Digital Enterprise Research Institute, National University of Ireland, Galway.

[Segaran et al., 2009] Segaran, T., Evans, C., and Taylor, J. (2009). *Programming the Semantic Web*. O'Reilly Media, 1st edition.

[Seid and Kobsa, 2003] Seid, D. Y. and Kobsa, A. (2003). Expert finding systems for organizations: Problem and domain analysis and the DEMOIR approach. *Journal of Organizational Computing and Electronic Commerce*, 13(1):1–24.

[Sen et al., 2006] Sen, S., Lam, S. K., Rashid, A. M., Cosley, D., Frankowski, D., Osterhouse, J., Harper, F. M., and Riedl, J. (2006). Tagging, communities, vocabulary, evolution. In *proceedings of the 20th anniversary conference on Computer supported cooperative work*, pages 181–190. ACM.

[Serdyukov et al., 2011] Serdyukov, P., Taylor, M., Vinay, V., Richardson, M., and White, R. W. (2011). Automatic people tagging for expertise profiling in the enterprise. In *ECIR '11: Proceedings of the 33rd European conference on Advances in information retrieval*, pages 399–410. Springer-Verlag.

[Shehab et al., 2008] Shehab, M., Squicciarini, A. C., and Ahn, G. (2008). Beyond User-to-User access control for online social networks. In *ICICS '08: Proceedings of the 10th International Conference on Information and Communications Security*, pages 174–189. Springer.

[Shen and Dewan, 1992] Shen, H. and Dewan, P. (1992). Access control for collaborative environments. In Turner, J. and Kraut, R., editors, *CSCW '92: Proceedings of the ACM conference on Computer-supported cooperative work*, pages 51–58. ACM.

[Shneiderman and Plaisant, 2004] Shneiderman, B. and Plaisant, C. (2004). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Pearson Addison Wesley, 4th edition.

[Sigurbjörnsson and van Zwol, 2008] Sigurbjörnsson, B. and van Zwol, R. (2008). Flickr tag recommendation based on collective knowledge. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 327–336. ACM.

[Slagter et al., 2007] Slagter, R. et al. (2007). Deliverable 1.2: W1.2a – collaboration infrastucture - state of the art, innovation and gaps. Technical report, Ecospace Project.

[Smith, 2008] Smith, G. (2008). *Tagging: People-powered Metadata for the Social Web*. New Riders Press, 1st edition.

[Sood and Hammond, 2007] Sood, S. C. and Hammond, K. J. (2007). Tagassist: Automatic tag suggestion for blog posts. In *proceedings of the International Conference on Weblogs and Social Media (ICWSM)*.

[Stutzman, 2007] Stutzman, F. (2007). Social network transitions. http://chimprawk.blogspot.com/2007/11/social-network-transitions.html. Online - Last visited: 30-Jan-2012.

[Surowiecki, 2004] Surowiecki, J. (2004). *The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations*. Doubleday.

[Tang et al., 2008] Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., and Su, Z. (2008). Arnetminer: Extraction and mining of academic social networks. In *proceedings of the Fourteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 990–998. ACM.

[Tolone et al., 2005] Tolone, W., Ahn, G., Pai, T., and Hong, S. (2005). Access control in collaborative systems. *ACM Computing Surveys*, 37(1):29–41.

[Tootoonchian et al., 2008] Tootoonchian, A., Gollu, K. K., Saroiu, S., Ganjali, Y., and Wolman, A. (2008). Lockr: social access control for web 2.0. In *WOSP '08: Proceedings of the first workshop on Online social networks, co-located at ACM SIGCOMM 2008 Conference*, pages 43–48. ACM.

[Uschold et al., 1998] Uschold, M., King, M., Moralee, S., and Zorgios, Y. (1998). The enterprise ontology. *Knowledge Engineering Review*, 13(1):31–89.

[Wang and Jin, 2009] Wang, Q. and Jin, H. (2009). Selective message distribution with people-tagging in user-collaborative environments. In *CHI '09 Extended Abstracts*, pages 4549–4554. ACM.

[Ware, 2004] Ware, C. (2004). *Information Visualization: Perception for Design*. Morgan Kaufmann, 2nd edition.

[Weng et al., 2010] Weng, J., Lim, E.-P., Jiang, J., and He, Q. (2010). Twitterrank: Finding topic-sensitive influential twitterers. In *WSDM '10: Proceedings of the 3rd ACM international conference on Web search and data mining*, pages 261–270. ACM.

[White et al., 1994] White, J. S., O'Connell, T., and O'Mara, F. (1994). The ARPA MT evaluation methodologies: Evolution, lessons, and future approaches. In *proceedings of the 1st Conference of the Association for Machine Translation in the Americas*, pages 193–205.

[Whitlock and Micek, 2008] Whitlock, W. and Micek, D. (2008). *Twitter Revolution: How Social Media and Mobile Marketing is Changing the Way We Do Business and Market Online*. Xeno Press.

[Xu et al., 2006] Xu, Z., Fu, Y., Mao, J., and Su, D. (2006). Towards the semantic web: Collaborative tag suggestions. In *proceedings of the Collaborative Web Tagging Workshop at WWW '06*.

[Yang et al., 2008] Yang, J., Adamic, L. A., and Ackerman, M. S. (2008). Crowdsourcing and knowledge sharing: strategic user behavior. In *proceedings of the 9th ACM conference on Electronic commerce*, pages 246–255.

[Young, 2008] Young, J. R. (2008). Study raises new privacy concerns about facebook. http://chronicle.com/article/Study-Raises-New-Privacy-Co/465/. Online - Last visited: 30-Jan-2012.

[Yu et al., 2007] Yu, J., Jiang, Z., and Chan, H. C. (2007). Knowledge contribution in problem solving virtual communities: the mediating role of individual motivations. In *proceedings of the ACM SIGMIS CPR conference on Computer personnel doctoral consortium and research conference*, pages 144–152.

[Zhang et al., 2007] Zhang, J., Ackerman, M. S., and Adamic, L. (2007). Expertise networks in online communities: structure and algorithms. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 221–230. ACM.

[Zhang et al., 2010] Zhang, J., Qu, Y., Cody, J., and Wu, Y. (2010). A case study of microblogging in the enterprise: use, value, and related issues. In *CHI '10*, pages 123–132. ACM.

[Zhao and Rosson, 2009] Zhao, D. and Rosson, M. B. (2009). How and why people twitter: the role that micro-blogging plays in informal communication at work. In *GROUP '09*, pages 243–252. ACM.