Building Expert Recommenders from Email-Based Personal Social Networks

Verónica Rivera-Pelayo[1], Simone Braun[2], Uwe V. Riss[3], Hans Friedrich Witschel[3], Bo Hu[3]

[1]Facultat d'Informàtica de Barcelona, Universitat Politècnica de Catalunya,
Jordi Girona, 31
08034 Barcelona, Spain
veronica.rivera@est.fib.upc.edu
[2]Forschungszentrum Informatik,
Haid-und-Neu-Str. 10-14
76131 Karlsruhe, Germany
braun@fzi.de
[3]SAP AG, Dietmar-Hopp-Allee 16
69190 Walldorf, Germany
{Hans-Friedrich.Witschel, Bo01.Hu, Uwe.Riss}@sap.com

Abstract In modern organisations there is the necessity to collaborate with people and establish interpersonal relationships. Contacting the right person is crucial for the success of the performed daily tasks. Personal email corpora contain rich information about all the people the user knows and their activities. Thus, an analysis of a person's emails allows automatically constructing a realistic image of the surroundings of that person. This chapter aims to develop ExpertSN, a personalised Expert Recommender tool based on email Data Mining and Social Network Analysis.

ExpertSN constructs a personal social network from the email corpus of a person by computing profiles – including topics represented by keywords and other attributes such as recency of communication – for each contact found in the emails and by extracting relationships between people based on measures such as co-occurrence in *To* and *CC* fields of the emails or reciprocity of communication. Having constructed such a personal social network, we then consider its application for people search in a given work context. Through an analysis of several use cases, we have derived requirements for a query language that allows exploiting the personal social network for people search, taking into account a variety of information needs that go well beyond classical expert search scenarios known from the literature. We further discuss the application of the people search interface in a personal task management environment for effectively retrieving collaborators for a work task. Finally, we report on a user study undertaken to evaluate the personal social network in ExpertSN that shows very promising results.

Keywords: Egocentric Social Networks, Expert Search, Social Network Query Language, Email Mining

1 Introduction

Currently we witness a significant increase in the uptake of social network technologies that have almost become an omnipresent phenomenon. The impact of such networks on organisations has become a vividly discussed topic. While initially extra-organisational social networks such as Facebook[1] or LinkedIn[2] were used for organisational purposes [41], we now observe a certain trend of implementing purposely designed intra-organisational social networks such as IBM's Beehive [13]. The motivation behind the introduction of social network applications into large organisations was to foster new ways of communication and collaboration between the employees and motivate them to share work-related personal experience. The importance of social network support has been further increased by the growing agility requirements of organisations and their workforce, which continuously raise the need for IT support in collaboration and expert identification [15,35].

In the enterprise environment, a widely accepted use of social networks is the "expert search" that is finding the right person with the right expertise. In such a field, there is, however, a particular challenge resulting from the fact that collaboration always takes place in a restricted social context and not in a global network [6]. As a consequence the willingness to collaborate often depends on the personal proximity to the respective expert. Beside the use of extra- and intra-organisational social networks, which bring about various privacy issues [1,20], recently the use of so-called egocentric social networks (ESNs) has entered the discussion. They serve similar purposes as global public networks but are tailored to and controlled by the individual users [17]. To this end they take the personal communication (e.g. emails, blogs, etc.) of the users into account to help them find a potential expert. On this basis ESNs allow exploring the relations to and between other people within and outside the user's organisation. Among other communication channels, email as the most extended and accepted way of communication in and between organisations provides a rich reservoir of data, of which the full potential has not been exploited yet. Emails are being used for various purposes that go far beyond their original aim of mere communication [48]. Since especially in large organisations people can only sporadically be reached at their desks, emails are used for all kinds of purposes such as requesting answers to questions, discussing problems, summoning up meetings, managing projects, aligning work tasks and so on. Each email establishes connections of varying intensity depending on the nature of the recipients, e.g. who was directly addressed or who was included via a distribution list. The connections may also be regulated by other characteristics of emails, e.g. time interval, subject, etc. All this information is, however, hidden in the massive body of personal emails and thus not accessible to the users. The situation is aggravated due to a lack of suitable visualisers that assist us to harness the vast amount of information.

[1] http://www.facebook.com/
[2] http://www.linkedin.com/

In this chapter, we present an approach that has been developed at SAP Research and consists in the design, implementation and evaluation of an expert search tool, called ExpertSN, based on a Personal Social Network (PSN). ExpertSN extracts and analyses the hidden treasure in the personal email corpus to build up the PSN that is focused on the owner of the email corpus. Hence, individual users have full control over the resultant PSN without jeopardising data privacy and safety. The analysis of this corpus provides information about all contacts that appear in the email and is used as a data pool for the expert recommender. The aim of this recommender is to identify potential collaborators for specific tasks. Identifying the experts is normally not the end of process, it is also important to select the most appropriate one judged by other criteria [31]. This is facilitated by additional information about the particular foci of expertise that the respective expert is bringing along and their general availability.

A particular characteristic of the current approach is that the entire data management and persistence is based on semantic technologies. This approach opens up opportunities to exploit the analysis results for a semantically integrated Personal Information and Task Management framework that we have started to develop [19,38,39]. Here the aim is to provide a seamless infrastructure that allows users to get desktop-wide access to personal information resources as well as to the corresponding metadata and offer analytic tools to exploit the resulting semantic network so as to enrich the PSN.

In order to simplify the interaction with the ExpertSN system, we developed a Social Network Query Language (SNQL) for expressing the users' search requirements in the PSN. This query language helps exploit the email data pool and thus supports users to find suitable collaborators for their daily tasks.

In summary, the design and development of ExpertSN was directed towards the following objectives:

- Analysis of actual work situations that require expert search;
- Data extraction from the user's email corpus;
- Development of use cases and requirements for such extraction;
- Construction of a social network graph of persons from the email corpus;
- Keyword extraction methodology that automatically annotates persons with keywords found in emails;
- Definition of a relationship between two persons based on email corpus;
- Recommendations based on both keywords and network proximities.

Before we explain the ExpertSN approach in more detail, we would like first give an overview of the current state of the art in email based network development and expert search in Section 2. In Section 3 we derive the requirements of an expert search system as they appear in the context of social networks. In Section 4 we describe the chosen approach, turning to technical aspects of architecture and implementation. In Section 5 we detail the experience of using the system and the lessons learned. We conclude the chapter in Section 6 where we briefly summarise the results and envision the prospective further research and development of the ExpertSN system.

2 State of the Art

In this section we review the related work of using email corpora as knowledge source. This is followed by a survey of existing approaches to social network analysis (SNA) and expert recommendation systems.

2.1 Email Corpora as Knowledge Source

Email systems provide a rich source of information and the analysis of email communication [16,34,36] has been a quite popular research topic in social and computer science for many years [17,28,45,47]. An email is not only a simple text message but also yields information on interactions and user behaviour [42] and knowledge transfer among people within a specific context.

Clustering and classification techniques are used to analyse the email content that might be trained with manual user input [33,43]. Carvalho and Cohen [8] showed methods for automatically identifying the different parts in a plain-text email message, e.g. of significant importance are signature blocks and reply lines. Aalst and Nikolov [45] aimed to discover the interaction patterns and processes in the email corpora of a company's employees to support business process management. They created an organisation wide sociogram from such information. Culotta et al. [11] extracted one's social network from his or her email inbox and enriched the network with expertise and contact information for each person obtained from the Web.

Balog and de Rijke [2] found out that (i) the fielded structure of email messages can be effectively exploited to find pieces of evidence of expertise, which can then be successfully combined in a language modelling framework, and (ii) email signatures are a reliable source of personal contact information. Campbell et al. [7] compared two algorithms for determining expertise from email: a content-based approach that mines email text and a graph-based ranking algorithm adapting HITS (Hyperlink-Induced Topic Search [27]). Their results suggest that HITS makes more specific or targeted predictions than the simple algorithm. However better algorithms are needed to capture more knowledge.

Several studies have been carried out in order to analyse personal relationships via email messages. The study by Whittaker et al. [49] has shown that frequency, reciprocity, recency, longevity and affiliation of email interaction are strong predictors of the importance of email contacts. Similarly, the study by Ogata et al. [36] reveals that the social relationship is strong if email between two persons is exchanged frequently, recently and reciprocally. Van Reijsen et al. [46] propose to extract email based social networks (who-knows-who) from email header data as well as knowledge (who-knows-what) from email bodies. They have developed tools, ESNE (email social network extractor) and EKE (email knowledge extractor) and discuss an integration of both.

Carvalho and Cohen [9] and Pal and Mccallum [37] approached the problem of recipient prediction; i.e. identifying and suggesting potentially intended recipients when composing the email. The prediction is based on the written content. This problem is closely related to the expert finding approach, since it involves

identifying people within an organisation or social network who are working on similar projects, dealing with similar issues or who have relevant skills.

2.2 Social Network Analysis

There is a wide range on SNA tools that support examining social networks and performing common SNA routines like *UCINET*[3] or *Pajek*[4] or *Vizster*[5] for visualisation. *InFlow 3.1*[6] maps and measures knowledge exchange, information flow, emergent communities, networks of alliances and other networks within and between organisations and communities. The *Email Mining Toolkit* ([42]) and *EmailNet* ([44]) analyse email usage patterns at the individual and group level and compute behaviour profiles or models of user email accounts.

Farnham et al. [16] used public corporate mailing lists to automatically approximate corporate social networks. They have shown that co-occurrence in mailing lists provided a good predictor of who works with whom. With their *Point to Point* tool, they let users explore these networks and decide whom they want to contact. Based on this, the authors had also found that the network visualisation had a meaningful impact on users' ratings with respect to the similarities between others and themselves – people appeared less similar with the presence of the visualisation. Rowe et al. [40] introduce an algorithm for extracting a special kind of social relationships, namely hierarchial ones, from email communication.

Flink by Mika [32] is a system for extracting, aggregating and visualising online social networks, specifically of researchers together with their interests and research topics. It employs semantic technologies for reasoning with personal information extracted from different electronic information sources including web pages, emails, publications and FOAF profiles. FOAF[7] is intended to create a Web of machine-readable pages describing people, the links between them and the things they like. SIOC[8] (Semantically-Interlinked Online Communities) is an ontology for semantically representing Social Web data in order to integrate the information from online communities. Similarly, RELATIONSHIP[9] is a vocabulary for describing relationships among people with a specific nomenclature of terms like "ancestor of", "colleague of" or "employer of".

As has been mentioned before, the analysis of egocentric social networks has become a more active area of research due to the consideration of privacy. Fisher [17] makes a first attempt to view social networks from an individual's perspective and to understand how people manage group communication. He uses data from email or newsgroups [17,18] to construct egocentric social networks, other

[3] http://www.analytictech.com/ucinet
[4] http://pajek.imfm.si/doku.php
[5] http://hci.stanford.edu/jheer/projects/vizster
[6] http://www.orgnet.com/inflow3.html
[7] http://www.foaf-project.org/
[8] http://sioc-project.org
[9] http://vocab.org/relationship/.html

approaches focus on mobile call networks [50]. In [22], the authors make an attempt to compare public global social networks with egocentric ones derived from email.

2.3 Expert Systems for People Recommendation

A key issue of expert recommender systems is the determination of expert profiles, which Balog and de Rijke [3] described as the record of topical profile (types and fields of skills) plus social profile (collaboration network). Determining a user's expertise is usually based on and often limited to topic extraction from documents. Becerra-Fernandez [4] provided a review on expertise locator systems from a knowledge and human resource management perspective. Balog and de Rijke [3] presented a formalisation in order to automatically determine an expert profile of a person from a heterogeneous corpus of an organisation. Hofmann and Balog [24] explored in a pilot study how contextual factors identified by expertise seeking models can be integrated with topic-centric retrieval performance. Based on the vision of the social semantic desktop [21] that includes the extraction of metadata about people and the extraction of content from the desktop files, Demartini and Niederée [12] presented a new system for finding experts in the user's desktop content.

Recently expert recommendation systems have attracted strong interests in major enterprises. For instance, the social networking web application *SmallBlue* by IBM Research [15,29] aims to locate knowledgeable colleagues, communities, and knowledge networks in companies based on analysing a user's outgoing email and instant messages. Different components provide functionalities such as expertise search, profile information, social search showing the social distance and combining it with social bookmarking and web search, as well as displaying the social network of top experts associated with a topic.

In conclusion, email corpora provide a rich source of information. The extraction of knowledge from these email corpora is a new trend in the field of expert recommender systems, in contrast with the conventional methods for searching experts based on documents. Meanwhile, most of the approaches that offer expert recommendation are only based on content information, neglecting the importance of relationships among people. The approach of combining the content and metadata extraction from emails with the characterisation of the relationships between the people based on network analysis is one of the innovative aspects that differentiates our approach from apparently similar ones. In other words: previous research has concentrated on either email-based expert search or analysis of email-based social networks, but there is few work that brings both together. Our main motivation for doing so is the possibility to cover a broader variety of information needs in expert search – i.e. to be able to search for facets, introducing e.g. a distinction between finding (results known largely in advance) and searching (completely open results) – and thus to go beyond the simplistic notion of "expertise" that is so widely used in the existing expert search literature. The systematisation and formalisation of the varying

information needs via the Social Network Query Language (SNQL) is another distinctive feature of our research.

3 Information Needs in Expert Search

The task of finding collaborators in an organisational environment – which we refer to as "people retrieval" hereinafter – can involve more aspects than a person's expertise. Therefore, a prerequisite of designing an expert recommendation system is to gain an understanding of different information needs that support different use case of people retrieval.

Categorisation of information needs is extensively studied in Information Retrieval research, namely in the area of web search, where tasks like homepage finding or on-line service location were identified in addition to the traditional topic relevance searches. Taxonomies of search tasks have been brought forward e.g. based on the number of results that a searcher expects (see the TREC web track [23]). Similar to web search, information needs in people retrieval can also be generally classified into two groups: *finding* and *searching* based on both the goals of the user and the expected results. In *people finding*, the result set is known in advance. In other words, the user is looking for a particular person or a group of persons whose existence is already known. He/She is seeking to enrich the known information, e.g. contact details and sometimes name(s). The search criteria in the people finding use case are very clear and sometimes well-defined for the user. In *people search*, the situation is different: the user is looking for people with specific characteristics. He/She does not know if such people exist or, in case they do exist, how many candidates there are. The search criteria in the people searching use case are rather vague. Typically, retrieved experts will match the criteria to different extents and thus ranking become necessary.

The rest of this section provides a detailed analysis of information needs in the context of people finding and people search, which are compilations of situations that the authors have experienced in their daily work activities. The first of these use cases is centred around a researcher, Philipp, who – having worked in a research project for some years – now joins a new project that started two years ago. Philipp needs to catch up with the work that has been done and, even more importantly, get acquainted with the people who are working on the same project and who can help him with his new tasks. We will use the task of writing a project report as an example. The second use case is that of a project manager, Jan, who has taken up a new position as software release manager where he needs to communicate with a large number of people, including developers, testers and customers. Jan needs to release the new software EasyWay and wishes to set up a meeting with all key persons involved in the production. Fortunately, he had been involved since the early stages of the EasyWay project.

3.1 People Finding

There are several situations that can lead to looking for a person or a group of persons of whose existence is known. Based on our two use cases, we have extracted the following categories:

Substitute (single person). This case occurs when one of the collaborators in the past is no longer available, e.g. due to a job change. In this case, it is desirable to find a replacement of that person. In the example of our first use case, imagine that Philipp, who needs to work on a software prototype that is part of his new project, finds the name of a programmer in the source code of the prototype. Let us assume that this person has left the project and the company. Philipp then has to search for the substitute.

Representative (single person). Sometimes one faces the need to contact a person in another department or organisation in order to acquire answers to a specific question. In such a situation, one needs to find the right representative in the target group. Imagine, for example, that Philipp would like to access a versioning system as a repository for important project documents. He knows which partner organisation is hosting the system and also knows that there is a dedicated contact person, but is lacking name and contact details of that person.

Group members (group of persons). Frequently, one needs to contact all members of a certain group, e.g. to set up a meeting or distribute an announcement. In the case where no mailing lists already exist for the given group, finding all members of a group becomes challenging. In the example of our second use case, we can imagine that Jan is trying to set up a meeting with all members of the EasyWay project.

3.2 People Search

As mentioned above, people search has more uncertainty in the result set than people finding. In addition, it often involves less clear criteria, e.g. the personal relationships among people. We have discovered the following categories of criteria that characterise information needs:

Expertise. This is the conventional expert search where people with a particular expertise are to be identified. However, it is possible to further sub-categorise this kind of information need with respect to the kind of knowledge that one wishes to obtain from the expert – by distinguishing between procedural and factual knowledge:

Procedural expertise: Often, one needs to find a person that has (successfully) performed a certain task in order to draw from their experience – in the form of advice or resources that have applied or produced. For instance, Philipp might need to write a report describing the outcomes of the accomplished work in the project. He suspects that someone has done this task before and thus he can use as a guide the templates and examples that were used by other before. In order to find such information, he needs to look for a person who has actually performed this task in the past.

Factual expertise: This corresponds to the topical relevance in document retrieval where a document conveys the information about a certain topic. It occurs in cases where one is searching for persons interested and knowledgeable in a certain topic in order to collaborate with them or engage in discussions. Here, the primary focus is not necessarily on sharing concrete experience, but rather on common interests and factual knowledge that can lead to a fruitful collaboration. For instance, we can imagine that Jan would like to contact people from another department – which he has not previously worked with – in order to discuss a potential future collaboration aiming at extending EasyWay with new functionalities. The system should recommend collaborators from the department who are working on or interested in topics that are relevant to EasyWay.

Closeness. Sometimes a person (P) that one usually works with or whom one knows to be knowledgeable about something is unavailable or too busy to respond. In such cases, the user may be interested in contacting a person who is close enough to P. We can further distinguish two types of closeness:

Close collaboration: This refers to persons who work closely together with P – it could be that they are in the same department and project or share an office. In our example, let us imagine that Philipp is looking for people who know a certain software; he knows that John is among the few people who do and is the most knowledgeable one about the software. Unfortunately, John is on vacation. Philipp would like the system to propose from those who collaborate closely with John, with the hope that there is another expert among these. Moreover, Philipp would prefer those who know John personally because he has a good relationship with John and believes that in this way he is more likely to get good answers.

Close expertise: Here, we are looking at cases where the user knows potentially rather little about P, except the fact that John is a knowledgeable person on a certain topic. The user would like to find other people with similar expertise. That is, there is no strict need for the retrieved person to know P, apart from using P's profile as an exemplary case. In our example, consider Jan looking for someone in the Change Management Department who could help him with a particular problem. He knows that Michael has the necessary expertise, but is unavailable. Therefore he needs to find other people in Change Management who have a matching profile with Michael's.

Availability. The (regular) availability of a person may be an important criterion for fruitful collaboration: someone can be a very renowned expert in a field; but if he/she never has time, there is no hope for good collaboration. For instance, assume Philipp needs the expertise of an analyst for his report. He finds out that the head analyst of the project is not responsive. He would like to find

another analyst, e.g. using *close collaboration* as a criterion with an additional constraint on availability.

It is typical for people retrieval that two or more of the criteria mentioned above are combined. For instance, it is often possible to restrict the search for a person with a certain expertise to a specific group of persons (Group members) or it may be desirable to combine expertise with availability.

3.3 General Ranking Criteria

In addition to analysing and categorising the information needs mentioned above, we propose the following general considerations about how the results of people retrieval should be ranked – taking into account the *personal relationship* between the user and the contacts that are retrieved:

Affluence of communication. Depending on the situation, a high or low affluence can be desirable. In many situations, a user would prefer well-known persons to be ranked close to the top of the list because it is easier to get in contact and collaborate with these. On the other hand, a searcher might be more interested in *novelty*, i.e. retrieving contacts of whom he or she would otherwise not have thought. Hence, it is important for a retrieval system to be able to retrieve persons with whom he/she has never communicated before.

Time. The date of last and first activity or communication between a retrieved contact and the user may be an important ranking criterion. It is often desirable to rank experts at a higher position if they have been in contact with the user recently and/or for a long period of time.

3.4 A Social Network Query Language

The means to express both the "atomic" information needs described above and combinations thereof is provided by a Social Network Query Language (SNQL).

In the field of web search, considerable effort has been invested into automatically classifying queries according to search task categories (e.g. homepage finding versus topical relevance, see e.g. the TREC-2004 web track [10] with its query classification task), in order to apply different ranking algorithms based on the outcome of classification. Since we are not sure whether an automatic query classification (differentiating people finding and people search) is possible for people retrieval and since this is beyond the scope of this work, we will present SNQL for the *explicit* formulation of all information needs that we have identified. The SNQL intends to offer a suitable mean to specify in which conditions and contexts a user is seeking collaborators to do a specific task.

Table 1 summarises the atomic constructs from which SNQL queries can be composed. In addition to the explanation in the second column, the table also lists the types of information needs (see Section 3) that are satisfied by each constructs. The SNQL is formalised using the Extended Backus-Naur Form (EBNF) in Figure 1. For simplicity, we do not further detail the syntax of the

```
query = keywords, { ";", [ "NOT", blank ], person },
        [ ";REG AVAILABILITY" ], [ options ]
        | "REPRESENTATIVE", blank, query;
options = closeness | unavailable | compare | substitute | compound;
keywords = words, { ",", words };
closeness = ";", "CLOSE", blank, person;
unavailable = ";", "UNAVAILABLE";
compare = ";", "COMPARE", blank, words;
substitute = ";", "SUBSTITUTE", blank, person;
compound = [ recency ], { relationship } , [ order ];
recency = ";", "RECENT", [ blank, date];
relationship = ";", "REL", blank, person;
order = ";", "ORDER", blank, [ "weight" | "reciprocity" | "co-occurrence" ];
person = "name=", words | "email=", address;
words = word, { blank, word };
date = day, month, year;
blank=" ";
```

Figure 1. SNQL in EBNF

non-terminals word, address, day, month, and year, for which we assume standard definitions.

In the next section, we will describe ExpertSN and show how an egocentric social network, gained from an analysis of a person's emails, can be exploited for satisfying many of the discussed information needs and thus how that social network can help implementing SNQL.

4 ExpertSN – a Personalised Expert Recommender

ExpertSN system is an Expert Recommender based on email mining and social network analysis including the construction of an egocentric Personal Social Network. The general architecture of ExpertSN is based on two main modules, as shown in Figure 2. The first module involves the construction of the egocentric Social Network from the email corpora and consequently has the email corpora as input and the PSN as output. The second module constitutes the Expert Recommender, taking as input the PSN and generating a personalised expert recommendation.

4.1 ExpertSN Architecture

Technically, ExpertSN is composed of:

- A presentation layer, through which users interact with the system, comprising the visualisation of the PSN and the console terminal that allows the communication with the Expert Recommender (e.g. entering SNQL queries).
- A logic layer consisting of modules that implement the steps in figure 2, i.e. crawl the raw data, create the PSN and allow to access the PSN data

Table 1. Constructs of the SNQL

Constructs	Explanation	Information need / ranking criterion
SUBSTITUTE *person*	find the substitute for *person*	Substitute
REPRESENTATIVE *q*	Given a set of results for SNQL query *q*, find a representative person	Representative
keywords	set of keywords associated to a contact describing his/her expertise	Factual Expertise
NOT *person*	allows the exclusion of a person from the result set	
REG AVAILABILITY	find a contact with availability to establish a continuous collaboration	Availability
REL *person*	find a contact with a relationship with *person*	(Close) collaboration
CLOSE *person*	find a contact close to *person*	Close collaboration
UNAVAILABLE *person*	indicates that *person* is unavailable, the user needs a contact with similar characteristics	Close expertise
COMPARE *keywords*	search for contacts close to the current collaborator in a field described by *keywords*	Close expertise
RECENT *date*	retrieve contacts with evidence of activity more recent than *date*	Time factor
RECENT	rank by recency of communication	Time factor
ORDER *criteria*	order candidates following the *criteria*	Affluence of communication

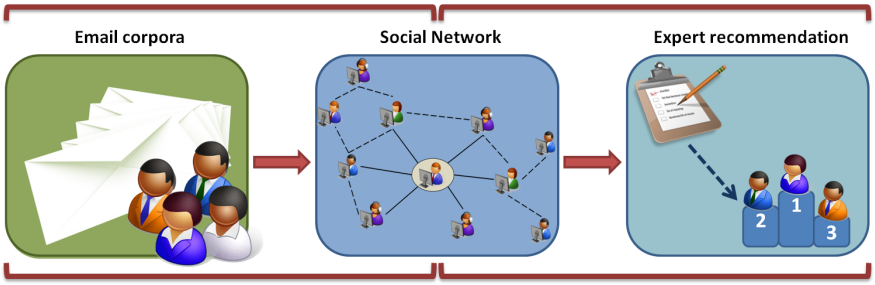

Figure 2. The two ExpertSN modules

for use in the presentation layer. Crawling of emails is performed by the *Outlookcrawler* of the *Aperture* Framework[10], which interacts with Microsoft Outlook (the main email management tool being used at SAP Research) to extract all email content and metadata into RDF[11]. Another important part of the logic layer consists of the extraction of keywords from the body of an email and the co-occurrence analysis of contacts from its header, handled by the *TE*[5] and *TinyCC*[12] modules, respectively (see below for details). Finally, the *Render* module in the logic layer helps to visualise the PSN through its interaction with *Protégé*[13] and the *Expert Recommender* module performs all the necessary tasks to generate a personal recommendation from the PSN based on the user's request/query.

- A data layer that contains the persistent data of the ExpertSN system, namely the email corpora, the RDF repository storing the email corpora, and the RDF repository storing the Social Network. We used the Sesame triple store[14] in conjunction with the *RDFReactor*[15] and *Jena*[16] libraries for the actual storage and retrieval of the RDF data – thus, all issues such as optimisation of queries, scalability etc. are handled by these components.

To describe the design of ExpertSN we will use one running example. Suppose that Jan is a user who has installed ExpertSN in his system. In the next two sections we will explain the construction of the Personal Social Network and the Expert Recommender using the context of Jan.

4.2 Construction of the Personal Social Network

The construction of the PSN provides the basis that can be fed into the Expert Recommender to discover answers to user's queries.

The first step towards constructing the PSN consists of crawling the email corpus, technically implemented by Outlookcrawler from the Aperture Framework. Outlookcroawler connects to the MS Outlook instance. In the ExpertSN system, we leverage mainly two types of information provided by the Outlookcrawler, i.e. the emails and contacts. The raw RDF file output by Outlookcrawler may contain invalid XML characters and can also have multiple string representations for the same email address. In order to guarantee the right performance of the next stages, this raw file must be cleaned.

The nodes of the resultant PSN are persons, including the user (owner of the email corpus, i.e. Jan) and the contacts, whereas the edges are undirected representing the relationships between pairs of persons. The relationships are classified into two categories according to the nodes that define the edge: if the

[10] http://aperture.sourceforge.net/
[11] http://www.w3.org/TR/PR-rdf-syntax/
[12] http://wortschatz.uni-leipzig.de/~cbiemann/software/TinyCC2.html
[13] http://protege.stanford.edu
[14] http://www.openrdf.org/
[15] http://semanticweb.org/wiki/RDFReactor
[16] http://jena.sourceforge.net/

user is one of the nodes defining the edge, the relationship is classified as *first* (first level); if both nodes are contacts from the network, it is denoted as *second* (second level).

The attributes that define each of the PSN nodes and edges were derived from the formalisation of the requirements. The formalisation allowed us to know which information is necessary in the PSN to enable the Expert Recommender. The attributes of each contact, first relationship, and second relationship were defined as follows: where $\mathcal{C}$ denotes the Contact nodes and $\mathcal{R}$ the relationship – note that we do not differentiate first and second level relationships when defining the attributes.

$\mathcal{C}$.**Frequency:** the total number of messages exchanged between the contact and the user, divided by the longevity of their relationship.
$\mathcal{C}$.**Availability:** the probability of keeping a constant contact, based on the frequency of communication of the most recent days.
$\mathcal{C}$.**Longevity:** the number of days since the first communication of the contact with the user.
$\mathcal{C}$.**Recency:** the number of days since the last communication of the contact with the user.
$\mathcal{C}$.**Keywords:** the list of words from the email bodies related to each contact, represented as pairs (t,w) where t is the term and w its significance.

$\mathcal{R}$.**Reciprocity:** the measurement of the act of returning/responding emails (only for *first* relationships).
$\mathcal{R}$.**Co-occurrence:** the joint appearance as receivers in emails.
$\mathcal{R}$.**Weight:** strength of the relationship based on the attributes of the adjacent contacts and the previous two attributes.

The cleaned RDF graph that represents the email corpus will be queried using SPARQL[17] to extract the following information: the sent date, the sender, the receivers in *To* and *CC* fields, the message subject and the message body. This information is analysed to construct the PSN as follows.

A link is created between the body of the email and each contact that appears on it in order to perform the subsequent extraction of keywords. Regarding the relationships between the contacts of the email, we can distinguish between the emails that the user has sent and the emails that he/she has received (including the list emails). In case the email is sent by the user, the first and second relationships are added to the PSN between: (1) the user and the receivers of the email (first relationship) and (2) all the receivers in To and CC (second relationship). In case the email is received by the user, the relationships are added to the PSN between: (1) the user and the sender (first relationship) (2) the user and the other receivers, i.e. co-receivers (first relationship) (3) the sender and all the receivers in To and CC (second relationship) (4) all the receivers in To and CC, excluding the user (second relationship).

[17] http://www.w3.org/TR/rdf-sparql-query/

The following Source Code 1.1 shows two SPARQL query examples used to obtain all this information. The first query would obtain the body content and the second query would obtain all the recipients in *To* field of an email identified by 'mailID'.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX nmo: <http://www.semanticdesktop.org/ontologies/2007/03/22/nmo#>
PREFIX nco: <http://www.semanticdesktop.org/ontologies/2007/03/22/nco#>

SELECT ?body
WHERE {?email rdf:type nmo:Email.
FILTER regex(str(?email), 'mailID', 'i' ).
?email nmo:plainTextMessageContent ?body. }

SELECT ?name ?mID
WHERE {?email rdf: type nmo:Email.
FILTER regex(str(?email), 'mailID', 'i' ).
?email nmo:to ?to.
?to nco:fullname ?name. ?to nco:hasEmailAddress ?mail.
?mail nco:emailAddress ?mID.}
```

Source Code 1.1. SPARQL query examples

Besides, all the receivers (in both *To* and *CC* fields) of each email are added to a file to serve as input for the TinyCC module. TinyCC 1.5 is used to calculate the frequency of joint occurrence between two co-receivers (sentence-based co-occurrence) and the co-occurrence significance, i.e. the log-likelihood ratio of joint occurrence. The computation of this significance is based on a null hypothesis that states that the occurrence of two items A and B (e.g. two email receivers) is probabilistically independent, i.e. $p(AB) = p(A)p(B)$; the likelihood ratio [14] indicates how much the data deviates from the null hypothesis.

The body of each email undergoes a special process to identify forwarded or replied emails, which appear as simple strings without any RDF identification of its components (e.g. sender, receiver or date). This requires a string identification process to find the potential headers of embedded emails. Implementation is done in two languages: German and English. The analysis of these headers allows finding information about contacts who have not exchanged (send, received or co-receive) emails with the user but appear in forwarded or response emails of him/her.

For each contact, we extract keywords from the body of emails that are related to it. The keyword extraction is performed using TE, which extracts terminologically relevant terms and phrases from short documents based on a large background corpus. This extraction is again based on a statistical test using a likelihood ratio [14]. Here, however, the null hypothesis is that the probability of a term occurring in the emails of a person is the same as that of occurrence in the large background corpus. That means that the weights associated to extracted terms indicate how strongly a term's relative frequency in the emails deviates from its relative frequency in the background corpus.

Figure 3 shows the morphology of the Personal Social Network through an example of the PSN that we would obtain from the email corpus of Jan. We use solid lines for the first relationships, i.e. relationships between Jan and the people who appear in emails exchanged with him. For the second relationships, dashed lines are used to link contacts that appear together in the emails exchanged with Jan or contacts who appear in forwarded/replied emails found in the body of

Jan's emails. These last contacts do not have a relationship with Jan because they have not directly exchanged emails with him (e.g. that would be the case of the contact A, who would appear in a thread that B has emailed to Jan).

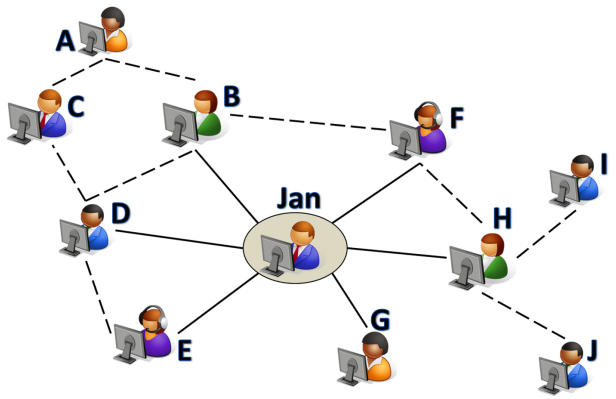

Figure 3. Morphology of a Personal Social Network

After the analysis of the emails and construction of the network, the calculation of all the measures that characterise a PSN is performed. For each contact, the following values are calculated:

Availability (Θ) is derived from the summation of frequencies of emails sent each day. Recent days are more important and thus the weight of each day is modified by an exponential factor:

$$\Theta = \sum_i f_i \cdot w_i, \qquad \text{with } w_i = e^{-\left(\frac{x_i}{\alpha}\right)^2}, \tag{1}$$

where f_i is the total number of emails on the ith day, x_i the number of passed days since the ith day, and α the relevance period. Currently, α is set to 4 weeks.

Longevity (Δ_{lg}) is the total number of days since the first message exchanged between the contact and the user (*c.f.* [49]):

Recency (Δ_{rec}) is the total number of days since the last message exchanged between the contact and the user (*c.f.* [49]):

Frequency (Φ_{freq}) is the total number n_{exch} of messages exchanged between the contact and the user divided by the duration of their relationship (*c.f.* [49]):

$$\Phi_{\text{freq}} = \frac{N_{\text{exch}}}{\Delta_{\text{lg}} - \Delta_{\text{rec}}} \tag{2}$$

The *relationship* is weighted in the following way:

Reciprocity (ϕ_{recip}) results from the comparison between sent and received messages with respect to a contact (for first relationships, *c.f.* [49]):

$$\phi_{\text{recip}} = 1 - \left| \frac{n_{\text{sentBy}} - n_{\text{sentTo}}}{n_{\text{exch}}} \right|, \tag{3}$$

where n_{sentBy} is the number of emails sent by the user and n_{sentTo} the number of emails sent to the user.

First Relationship Weight: (W_{firstRel}) General strength of the relationship based on the other specific measures and weighted according to their importance and influence on a relationship:

$$\begin{aligned} W_{\text{firstRel}} = {} & w_{\text{high}}(\phi_{\text{freq}} + \phi_{\text{to}} + \phi_{\text{recip}}) + \\ & w_{\text{low}} \left(e^{-w_{\text{rec}} \cdot \Delta_{\text{rec}}} \cdot e^{-\left(\frac{w_{\text{rel}}}{\Delta_{\text{lg}} - \Delta_{\text{rec}}} \right)} + \phi_{\text{cc}} + \phi_{\text{ct}} \right) \end{aligned} \tag{4}$$

where $w_{\text{high}} = 1$ weights the most important contributing factors, $w_{\text{low}} = 0.5$ weights the less important factors, $w_{\text{rec}} = w_{\text{rel}} = 1$ scale the recency and the inverse difference of longevity and recency, ϕ_{freq} is the normalised frequency, ϕ_{to} is the number of emails received from the user in *To* field divided by the total number of received emails, ϕ_{cc} is the number of emails received from the user in *CC* field divided by the total number of received emails, and ϕ_{ct} is the centrality of the related person defined as the number of this person's relations by the total number of persons in the network (excluding the user). $\Delta_{\text{lg}} - \Delta_{\text{rec}}$ defines the duration of the relationship with the user. ϕ_{freq}, ϕ_{to} and ϕ_{recip} are considered the most important measures when evaluating a relationship whereas Δ_{lg}, Δ_{rec}, ϕ_{cc} and ϕ_{ct} indicate in a lower level a relationship and therefore have a lower weight. The number of emails sent with contacts in the *To* field have a higher weight than those appearing in the *CC* field, due to the consideration that the relation is stronger in the first case than in the latter. All the parameters of the equation have been chosen and experimented empirically to have the desired behaviour i.e. obtain a normalised value that rewards when the contacts exchange more emails, the exchange is reciprocal and they have a long recent relationship, and penalizes otherwise.

Second Relationship Weight: ($W_{\text{secondRel}}$) General strength of the relationship based on the other specific measures, weighted according to their importance and influence on a relationship:

$$W_{\text{secondRel}} = \; w_{\text{exch}} \cdot \phi_{\text{exch}} + w_{\text{ct}} \cdot \phi_{\text{ct}} + w_{\text{co}} \cdot \phi_{\text{co}} \tag{5}$$

where $w_{\text{exch}} = 1$, $w_{\text{co}} = 0.7$ and $w_{\text{ct}} = 0.5$ are weights for the individual factors, while ϕ_{co} is the co-occurrence divided by the highest co-occurrence in the network. The *co-occurrence* is determined according to [5] by the TinyCC module. The used weight factors reflect the importance of the individual contributions and were determined empirically.

After computing the value, both first and second relationship weights are normalised in the network, divided by the highest value among all the first and second relationships respectively.

The structure of the PSN is stored in an RDF repository. *RDFReactor* allows the creation and management of new instances, following the ontology that defines the domain of the PSN.

The visualisation of the resultant PSN is too complicated for an RDF visualisation tool due to the amount of information that the network contains. A new representation of the PSN, therefore, was needed to allow the user visualise his PSN using Protégé. As a result, the ontology of the network was simplified to show the key point of the network: the relationships between the people. Still, in order to make the visualisation more manageable, the user can decide what relationships should be included in the user interface.

4.3 Expert Recommender Based on the PSN

Once the PSN is constructed, the Expert Recommender can use it as input to make its personalised recommendations. The Social Network Query Language (SNQL) was designed and implemented to give the user a mean for expressing his search request. For each SNQL construct, we translate it according to the PSN structure and its attributes. The formalisation of the requirements allowed us to perceive which constructs of the SNQL could be implemented from the email data pool. As a result, *substitute* and *representative* were not practical and thus were not implemented. In the following, we present the translation of SNQL constructs in terms of metrics of the PSN.

<keywords> Sum of significance values of the matching keywords in each person adjusted by the BM25 score of the person.

NOT <name/email> list of candidates - Person(name|email).

REG AVAILABILITY attribute availability higher than the arithmetic mean of the availabilities of the candidates.

REL <name/email > weight of relationship with Person(name|email).

CLOSE <name/email> weight of relationship and coincidence in keywords and co-occurrence, all the attributes related to Person(name|email).

UNAVAILABLE <name/email> NOT Person(name|email) and maximise co-occurrence of the contacts with Person(name|email).

COMPARE <keyword> compare significance values of the keyword(s) between the current collaborator and the candidates.

RECENT minimise (Today.date - Person.recency).

RECENT <date> Person.recency >date.

ORDER <criteria> order final list of candidates according to criteria. Possible values for criteria: weight, reciprocity or co-occurrence.

BM25 [25] is a ranking function used by search engines to rank matching documents according to their relevance to a given search query that is specified by keywords. The ranking is done based on the query terms appearing in each

document. For ExpertSN, a document is tantamount to all the sent and received emails of a person together in the persons *profile*.

Given a search query Q, containing keywords q_1,...,q_n, the BM25 score of a person P is shown in the Equation 6.

$$\text{Score}(P, Q) = \sum_{i=1}^{n} \text{IDF}(q_i) \cdot \frac{f(q_i, P) \cdot (k_1 + 1)}{f(q_i, P) + k_1 \cdot \left(1 - b + b \cdot \frac{|l_{\text{profile}}|}{L_{\text{avg}}}\right)} \tag{6}$$

where $f(q_i, P)$ is q_i's term frequency in the profile of person P, $|l_{\text{profile}}|$ is the length of the profile of P (number of words), L_{avg} is the average profile length in the persons collection from which persons are drawn (i.e. the social network), $k_1 = 2.0$ and $b = 0.75$. IDF(q_i) is the IDF (Inverse Document Frequency) weight of the query term q_i and is computed as follows:

$$\text{IDF}(q_i) = \epsilon + log \frac{N - n_{q_i} + 0.5}{n_{q_i} + 0.5}; \tag{7}$$

where N is the total number of persons in the network, $n(q_i)$ is the number of persons containing q_i as related keyword, ϵ is a floor constant to avoid negative values without ignoring common terms at all.

A query can be composed of more than one SNQL construct. Different constructs have different behaviour when the results are interpreted. Generally, we classify SNQL constructs into the following groups:

1. Group A: the return values must be calculated and then the retrieval function is applied. The constructs in this group are: keywords, relationship, closeness, recency, unavailable and compare.
2. Group B: the return values present a restriction, directly filtering out some candidates from the recommendation set. The constructs of this group are: not, forced recency and regular availability.
3. Group C: the action of the return value is made at the end, when the candidate collaborators are already chosen. The construct order is part of this group.

The retrieval function applied to the measures of group A is the Arithmetic Mean as given in Equation 8:

$$A = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{8}$$

where n is the total number of return values specified in the query; $x_1, x_2, \ldots, x_n$ would be the return value of the interpretation of each construct of the query.

We chose the Arithmetic Mean because it weights all the constructs of the query equally. Besides, experimentation to compare the Arithmetic Mean and Harmonic Mean was carried out and it was concluded, that the first function gives the desired behaviour (there is no necessity to avoid the influence of outliers).

In order to get a concrete idea of the construction and results of a query, we present below a query that Jan would do. Suppose that Jan is looking for someone in the Change Management Department who has worked with Mary, who is actually on holidays, and who has a regular availability to work with him. Jan would introduce the following query:

```
Task Description: change management department
Search query: REL name=Mary Sandens; REG AVAILABILITY
```

Jan would receive a list of recommended candidates, ordered in increasing order of punctuation:

```
Hope, Peter hope@sap.com 0.0
Winter,Martin winter@sap.com 0.031849
Schmidt,Maxwell schmidt@sap.de 0.854
```

Functionalities of the Expert Recommender Generally speaking, the final ExpertSN system supports two types of high level queries posted to the PSN:

1. "Who can be a collaborator for a certain task?" – the user posts a query to the system and receives a list of candidates to collaborate with him/her. Moreover, the system creates a network that is a part of the PSN, containing the recommended candidates and their relationships.
2. "What are the connections between person X and Y?" – the user can specify which relationship he/she wants to see between himself/herself and a contact or among contacts. The system shows the measures that qualify the relationship.

5 Discussion of the Results

The evaluation of the ExpertSN system includes firstly feedback of end users regarding the usability of the people search interface in a personal task management environment. We then report on a user study that evaluates the quality of the Personal Social Network extracted by ExpertSN.

5.1 Analytical Evaluation of the Search Interface

In order to study the usability of the people search interface, an analytical evaluation of ExpertSN was carried out, which uses a variation of the Activity-Oriented Evaluation Method (AOEM). This practical and analytical evaluation method utilises activity theory and was developed by M. Jurisch [26].

The main objective of ExpertSN's analytical evaluation was to assess users' experience about the interaction with the system and the user acceptance of the system. According to the description from Jurisch, AOEM intends to be a semi-structured method, which is flexible enough to adapt to the situation under investigation according to specific goals and context.

For the user acceptance evaluation of ExpertSN, the main objective was to understand what the user is expecting from the system and how the users interact with ExpertSN when searching for personal collaboration in their daily tasks. This allows evaluating the overall design principle of ExpertSN. The outcome of the acceptance evaluation is to answer whether the ExpertSN tool supports the users in selecting persons to collaborate with them in their daily tasks.

Due to practical reasons (imposed by the parent company), the testing of the tool within the system of each user was not possible. It was substituted by a qualitative description of the tool itself and the use of the SNQL. This description was materialised in a detailed presentation of the interface, functionalities and examples of ExpertSN in the framework of collaborators search. We introduced the tool to the subjects of the evaluation and showed the available functionalities and features. We designed a questionnaire based on the defined main goals and assessed it by the decomposition of the Activity System done in the previous steps. We additionally evaluated contradictions found during the design of the ExpertSN system with the questionnaire. Most of the items in the questionnaire were open-ended in order to permit the respondents to elaborate their opinions.

We performed the analytic evaluation with eleven participants, who were knowledge workers from three different companies. The results of the questionnaire confirmed most of the contradictions previously identified and addressed in the questionnaire. Regarding the necessity of computer-based recommenders, a majority of the answers were positive, pointing out the functionalities as an additional help, as a mean of saving time and having extra/new information about people. Three of the participants gave negative response in term of "unnecessary" or "had doubts about their performance". All the participants could identify expected benefits of using a personalised Expert Recommender and the possibility of adapting ExpertSN to a task management tool, especially because of usability, low turnaround time and faster performance of their tasks. Overall, the users had a positive opinion about the system and specified benefits like "easy access to potential collaborators", "semantical crawling of email corpora", "access to information they currently do not have" or "taking into account new measures e.g. availability or co-occurrence". However, one participant found that emails and keywords do not tell him how much a person really knows about a topic while another participant expressed that the location should be considered.

Referring to the features of ExpertSN, there was unanimity that the visualisation of the PSN would help the users choose a potential collaborator, if the visualisation can be easily configured according to certain attributes, measures or queries. The feature of consulting a relationship in the Social Network created diversity of opinions: some did not know what to answer while others found it useful and suitable to have a perception. However, one participant stated that he could not imagine that PSN based on a personal email corpus can provide the real quality of relationships. Regarding using Microsoft Outlook as an input source for ExpertSN, the participants in general show agreement, though two participants stated that they do not use Outlook.

Among the ExpertSN Recommender design features, the ones that sparked more controversy were *regular availability* and *relationship*. For some participants, it was clearly evident that the activity in email communication can be used to approximate the availability because in many organisations emails play an important role in communication and sent and received emails were informative enough to show the availability of a person. On the other hand, other users suggested that it depends on the context and calendars should be used for this purpose. Additionally, most of the participants thought that *compare* could be useful in situations like comparing skills in a concrete field, looking for specific competencies or impossibility to collaborate with the current collaborators.

The search criteria *relationship* created some doubts among the participants due to the limitation of having only email communication to approximate a real life aspect. Eight of the eleven participants affirmed being more likely to contact someone with a high *reciprocity*, because this contact is more reliable suggesting that it is more certain to obtain an answer and it indicates a higher participation level of the contact. Finally, the importance of contacting *recent* people depends on the situation, as asserted by the majority (eight out of eleven).

In general, discovering information in the email corpus of the same user triggered discussion in most of the evaluations. Some participants even affirmed it in their answers. However, it does not mean that all the participants were convinced. Two emphasised the *reciprocity* and the *co-occurrence* as promising measures to evaluate relationships between people, although one of them pointed out the problems that administrative emails can deviate the results in the case of co-occurrence.

Table 2 summarises the results of the questionnaire. It shows the percentage of negative, neutral and positive answers given to the questions that were not open.

Table 2. Quantitative results of the Questionnaire

Question	Negative	Neutral	Positive
Benefits/Limitations of ExpertSN	22%	0%	78%
Perception of the feature relationship in the PSN	55%	27%	18%
Visualization would help to choose a collaborator	0%	18%	82%
Integration with Outlook	27%	18%	55%
Explicit exclusion of collaborators with the SNQL	9%	18%	73%
Regular availability as a criteria to choose a collaborator	36%	0%	64%
Email as an aproximation of regular availability	27%	27%	45%
Email as an aproximation of a personal relationship	9%	64%	27%
Reciprocity as a factor to characterize a relationship	9%	9%	82%
Recency as a factor to characterize a relationship	0%	73%	27%

5.2 Quality of the Personal Social Network

The evaluation of the ExpertSN PSN (and its characterising measures) involves obtaining feedback from the end users regarding the accuracy of the extracted social network graph. For this purpose, ExpertSN was executed with the email corpus of the main author. The obtained PSN was examined against a manually crafted social network. The email corpus contained 581 mails in a 160 day window and the resulting PSN was formed with 145 contacts.

In the manually crafted version, the owner of the email corpus divided her contacts into three classes according to their role in her work life. Class A contains the professionals and tutors that advised/supervised her during her time in SAP Research. Class C contains all the students and colleagues that shared their time with her. Class B was formed according to the importance and closeness of friendship with a subgroup of the students and colleagues. Afterwards, the author indicated the membership of each the class. This classification was validated by those who understood the surrounding of the author very well.

In order to allow a comparison between the constructed PSN and the classes defined by the user, there should be an equivalent categorisation of the contacts in the network. This is done as follows: 1) the contacts were sorted in a decreasing order according to their edge strengths; and 2) a filtering rule was applied grouping the contacts in three categories:

```
if(strength >= 67%)              Class A
if(strength < 67% and >=33%)     Class B
if(strength < 33%)               Class C
```

The manually crafted social network was compared against the categorised results obtained from the system. First of all, the size of the classes followed the same pattern as the classes obtained from the system. In both categorisations, Class A contained a reduced number of contacts (3-4 persons); Class B had slightly more contacts than Class A; and finally Class C was the class which contains more members.

Regarding the class membership, the first disagreement was the non-overlapping part between the contacts that the user mentioned and the contacts that are really part of the ExpertSN PSN. This was specially evident in Class C, where the user only mentioned 11 contacts, whereas the ExpertSN PSN had 124 contacts. This supports one of the motivations for this research work: enriching and revealing the existence of new collaborators and increasing the user's awareness with respect to these contacts.

Secondly, we made the following observations when analysing the exact members in each class: Class A in the PSN contains all the three members given by the user. Interestingly, the PSN Class A contains one extra member, who was placed by the user in class B for being a student who has a close relationship with the user. The majority of contacts specified by the user as members of Class B were already contained in the corresponding PSN Class. There is one exception, who is a member of Class B according to the user, but was classified into Class C according to the PSN. This could be explained by the fact that this

contact, besides being part of the most trusted students, only started working with the user on the same project two months ago and thus has a lower level of activity in her email communication. Finally, the people identified by the user as members of Class C were included in the corresponding Class produced based on ExpertSN PSN. It, however, must be noticed (as stated before) that the entire set of email contacts were relatively small and Class C was served as the final catchment of all the contacts not classified to Class A and B by the ExpertSN system. After the analysis of class C, we noticed that there were new contacts whose existence the user ignored. These contacts mainly came from long threads and forwarded/replied emails which were embedded in emails of the user.

6 Conclusion and Future Work

With the deepening of globalisation and virtualisation, it becomes increasingly important to establish the right teams of specialities for consultation, coordination, and collaboration. Social network, as the signature of Enterprise 2.0 [30], was leveraged to assist the discovery and management of the expertise landscape in enterprises. In this paper, we focused on the enterprise social network solely based on one's work email. We developed the ExpertSN system which parses a given email corpus and constructs the user-centred egocentric social network covering all the contacts that the user may benefit from. The resultant social network then serves as the basis for identifying and discovering experts when special needs arise. The preliminary evaluation results of ExpertSN are promising: automatically generated social network largely agreed with inputs manually crafted by human users.

Building social network solely with work email is based on both practical and theoretical considerations. On the one hand, work email is normally available under organisational regulations and has less restrictive privacy and safety concerns than private email. This ensures that a further large scale evaluation can be performed. On the other hand, work email is tightly bound with one's daily work activity. A personal network constructed therefrom can fully align with one's work duties and thus facilitate a well-focused and well-targeted expert recommendation mechanism.

There are several points that need further investigation. The crux of our immediate future work lies in the evaluation of ExpertSN in a larger scale. Feedbacks from users can then serve to optimise the proposed mining and recommendation algorithms and to improve system usability. Modern enterprises generally have multiple internal information systems in parallel with the email system. The information contained in these systems can enhance and refine ExpertSN personal networks as well as validate the accuracy of such networks. We, therefore, will investigate what information sources are particularly useful for ExpertSN and how they can be effectively incorporated. Finally, though expert recommendation is the most evident application of ExpertSN personal networks, there are other potential use cases that can exploit the results of ExpertSN. For instance,

work-based personal network can be the basis of sophisticated SNA methods to derive patterns in employee behaviours and organisational structures.

Acknowledgements

This work is supported by the European Union IST fund through the EU FP7 MATURE Integrating Project (Grant No. 216356).

References

1. A. Acquisti and R. Gross. Imagined Communities: Awareness, Information Sharing, and Privacy on the Facebook. In *Proceedings of 6th Workshop on Privacy Enhancing Technologies*, 2006.
2. K. Balog and M. de Rijke. Finding experts and their details in e-mail corpora. In *15th International World Wide Web Conference (WWW2006)*, 2006.
3. K. Balog and M. De Rijke. Determining expert profiles (with an application to expert finding). In *IJCAI'07: Proceedings of the 20th international joint conference on Artifical intelligence*, pages 2657–2662, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.
4. I. Becerra-Fernandez. Searching for experts on the web: A review of contemporary expertise locator systems. *ACM Transactions on Internet Technologies*, 6(4):333–355, 2006.
5. C. Biemann, U. Quasthoff, G. Heyer, and F. Holz. Asv toolbox – a modular collection of language exploration tools. In *Proceedings of the 6th Language Resources and Evaluation Conference (LREC) 2008*, 2008. "http://wortschatz.uni-leipzig.de/~cbiemann/software/toolbox/index.htm".
6. J. S. Brown, P. Duguid, and P. Duguid. *The Social Life of Information*. Harvard Business School Press, March 2000.
7. C. S. Campbell, P. P. Maglio, A. Cozzi, and B. Dom. Expertise identification using email communications. In *CIKM Š03: Proceedings of the twelfth international conference on Information and knowledge management*, pages 528–531. ACM Press, 2003.
8. V. R. Carvalho and W. W. Cohen. Learning to extract signature and reply lines from email. In *CEAS-2004 (Conference on Email and Anti-Spam), Mountain View, CA*, July 2004.
9. V. R. Carvalho and W. W. Cohen. Recommending Recipients in the Enron Email Corpus. Technical report, Carnegie Mellon University, 2007.
10. N. Craswell and D. Hawking. Overview of the TREC-2004 Web Track. In *Proceedings of TREC-2004*, 2004.
11. A. Culotta, R. Bekkerman, and A. Mccallum. Extracting Social Networks and Contact Information from Email and the Web. In *CEAS-1*, 2004.
12. G. Demartini and C. Niederée. Finding experts on the semantic desktop. In *Personal Identification and Collaborations: Knowledge Mediation and Extraction (PICKME 2008) Workshop at ISWC 2008, Karlsruhe, Germany*, October 2008.
13. J. DiMicco, D. R. Millen, W. Geyer, C. Dugan, B. Brownholtz, and M. Muller. Motivations for social networking at work. In *CSCW '08: Proceedings of the 2008 ACM conference on Computer supported cooperative work*, pages 711–720, New York, NY, USA, 2008. ACM.

14. T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19:61–74, 1993.
15. K. Ehrlich, C.-Y. Lin, and V. Griffiths-Fisher. Searching for experts in the enterprise: combining text and social network analysis. In *GROUP '07: Proceedings of the 2007 international ACM conference on Supporting group work*, pages 117–126, New York, NY, USA, 2007. ACM.
16. S. Farnham, W. Portnoy, and A. Turski. Using email mailing lists to approximate and explore corporate social networks. In S. F. D. McDonald and D. F. (eds.), editors, *Proceedings of the CSCW'04 Workshop on Social Networks*, 2004.
17. D. Fisher. Using egocentric networks to understand communication. *IEEE Internet Computing*, 9(5):20–28, 2005.
18. D. Fisher, M. Smith, and H. T. Welser. You Are Who You Talk To: Detecting Roles in Usenet Newsgroups. *Hawaii International Conference on System Sciences*, 3, 2006.
19. O. Grebner and U. V. Riss. The social semantic desktop in an enterprise environment - integrating personal and organizational knowledge for an enterprise research department. In *9th European Conference on Knowledge Management*, pages 233–240, 2008.
20. R. Gross and A. Acquisti. Information revelation and privacy in online social networks. In *Proceedings of the 2005 ACM workshop on Privacy in the electronic society*, WPES '05, pages 71–80, New York, NY, USA, 2005. ACM.
21. T. Groza, S. Handschuh, K. Moeller, G. Grimnes, L. Sauermann, E. Minack, C. Mesnage, M. Jazayeri, G. Reif, and R. Gudjonsdottir. The nepomuk project – on the way to the social semantic desktop. In *Proceedings of the Third International Conference on Semantic Technologies (I-SEMANTICS 2007), Graz, Austria*, 2007. "http://nepomuk.semanticdesktop.org/xwiki/bin/view/Main1".
22. I. Guy, M. Jacovi, N. Meshulam, I. Ronen, and E. Shahar. Public vs. private: comparing public social network information with email. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, CSCW '08, pages 393–402, 2008.
23. D. Hawking. Overview of the TREC-9 Web Track. In *Proceedings of TREC-9*, 2000.
24. K. Hofmann, K. Balog, T. Bogers, and M. de Rijke. Integrating contextual factors into topic-centric retrieval models for finding similar experts. In *SIGIR 2008 Workshop on Future Challenges in Expertise Retrieval (fCHER)*, pages 29–36, 2008.
25. K. S. Jones, S. Walker, and S. E. Robertson. A probabilistic model of information retrieval: development and comparative experiments. In *Information Processing and Management*, pages 779–840, 2000.
26. M. Jurisch. User acceptance of a complex knowledge management tool. Master's thesis, University of Konstanz. Faculty of Law, Economics and Politics. Department of Politics and Management, 2009.
27. J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.
28. V. Krebs. Social capital: the key to success for the 21st century organization. *IHRIM Journal*, XII, No. 5:38–42, 2008. orgnet.com.
29. C.-Y. Lin, N. Cao, S. X. Liu, S. Papadimitriou, J. Sun, and X. Yan. SmallBlue: Social Network Analysis for Expertise Search and Collective Intelligence. In *ICDE '09: Proceedings of the 2009 IEEE International Conference on Data Engineering*, pages 1483 –1486, mar. 2009.
30. A. P. McAfee. Enterprise 2.0: The dawn of emergent collaboration. *MIT Sloan Management Review*, 47(3):21–28, 2006.

31. D. W. McDonald and M. S. Ackerman. Just talk to me: a field study of expertise location. In *Proceedings of the 1998 ACM conference on Computer supported cooperative work*, CSCW '98, pages 315–324, New York, NY, USA, 1998. ACM.
32. P. Mika. Flink: Semantic web technology for the extraction and analysis of social networks. *Journal of Web Semantics*, 3:211–223, 2005.
33. K. Mock. An experimental framework for email categorization and management. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 392–393, New York, NY, USA, 2001. ACM.
34. B. A. Nardi, S. Whittaker, E. Isaacs, M. Creech, J. Johnson, and J. Hainsworth. ContactMap: Integrating Communication and Information Through Visualizing Personal Social Networks. *Communications of the ACM*, 45:89–95, 2001.
35. B. A. Nardi, S. Whittaker, and H. Schwarz. It's not what you know it's who you know. 5(5), 2000.
36. H. Ogata, Y. Yano, N. Furugori, and Q. Jin. Computer supported social networking for augmenting cooperation. *Comput. Supported Coop. Work*, 10(2):189–209, 2001.
37. C. Pal and A. Mccallum. Cc prediction with graphical models. In *CEAS 2006 - The Third Conference on Email and Anti-Spam*, Mountain View, California, USA, July 27-28 2006.
38. U. V. Riss, O. Grebner, P. Taylor, and Y. Du. Knowledge work support by semantic task management. *Computers in Industry*, 61(8):798–805, October 2010.
39. U. V. Riss, M. Jurisch, and V. Kaufman. E-mail in Semantic Task Management. *IEEE Conference on Commerce and Enterprise Computing (CEC '09)*, pages 468–475, 2009.
40. R. Rowe, G. Creamer, S. Hershkop, and S. J. Stolfo. Automated social hierarchy detection through email network analysis. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 109–117, 2007.
41. M. M. Skeels and J. Grudin. When social networks cross boundaries: a case study of workplace use of Facebook and LinkedIn. In *GROUP '09: Proceedings of the ACM 2009 international conference on Supporting group work*, pages 95–104, New York, NY, USA, 2009. ACM.
42. S. J. Stolfo, S. Hershkop, K. Wang, O. Nimeskern, and C.-W. Hu. Behavior profiling of email. In *ISI*, pages 74–90, 2003.
43. E. Udoh. Mining e-mail content for a small enterprise. In *Innovations and Advanced Techniques in Computer and Information Sciences and Engineering*, pages 179–182. Springer Netherlands, 2007.
44. M. van Alstyne and J. Zhang. EmailNet: A System for Automatically Mining Social Networks from Organizational Email Communications. In *North American Association for Computational Social and Organizational Science (NAACSOS), Pittsburgh*, 2003.
45. W. M. van der Aalst and A. Nikolov. EMailAnalyzer: An E-Mail Mining Plug-in for the ProM Framework. BPM Center Report BPM-07-16, BPMCenter.org, 2007.
46. J. van Reijsen, R. Helms, T. Jackson, A. Vleugel, and S. Tedmori. Mining E-Mail to Leverage Knowledge Networks in Organizations. In *Proceedings of the 10th European Conference on Knowledge Management*, pages 870–878, 2009.
47. S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications.* Cambridge University Press, New York, USA, 1994.
48. S. Whittaker, V. Bellotti, and J. Gwizdka. Email in personal information management. *Commun. ACM*, 49(1):68–73, January 2006.

49. S. Whittaker, Q. Jones, and L. Terveen. Contact management: identifying contacts to support long-term communication. In *CSCW '02: Proceedings of the 2002 ACM conference on Computer supported cooperative work*, pages 216–225, New York, NY, USA, 2002. ACM.
50. Q. Ye, B. Wu, D. Hu, and B. Wang. Exploring Temporal Egocentric Networks in Mobile Call Graphs. In *6th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD '09)*, pages 413–417, 2009.