

Expertise Finding: Approaches to Foster Social Capital

Andreas Becks¹, Tim Reichling², Volker Wulf^{1, 2}

¹Fraunhofer Institute for Applied Information Technology (FhG-FIT), Germany

²Institute for Information Systems, University of Siegen, Germany

ABSTRACT: The application of information technology can have positive and negative impacts on social capital. In this paper we discuss technologies which have the potential to foster social capital by matching human actors. The matching algorithms are based on the personal data describing the actors' behavior, background, qualification, or interests. Consequently, actors who are little known or even unknown towards each other get aware of each other. We show how these concepts are applied to supplement a learning platform with an expertise matching functionality. Design principles for matching algorithms, a general architecture for an expertise matching algorithm, and an implementation of these functionalities are presented. Future challenges in the field of expertise matching are discussed.

1 Introduction

The term "social capital" has gained importance in the scientific discussion of different disciplines. Bourdieu (1985) provides an early definition of the concept: "Social capital is the aggregate of the actual or potential resources which are linked to possessions of a durable network of more or less institutionalized relationships of mutual acquaintance and recognition ..." (p. 248). Burt (1992, p. 9) understands social capital as friends, colleagues and other personal relationships which offer opportunities to use one's personal or financial capital. Putnam (1993 and 2000) applies the concept of social capital even to cities, regions and whole nations. He understands social capital as a set of properties of a social entity (e.g. norms, level of trust, or social networks) which enables joint activities and cooperation for mutual benefit.

All of these definitions of social capital have a point in common: the creation of social networks requires efforts (investments) and allows their purposeful use later on. Like financial capital invested in machinery or like personal capital gained within educational institutions, social capital increases the productivity of labor. Therefore, the concept has considerable economic relevance.

However, the findings concerning an appropriate structure of the social networks diverge. The mainstream assumes that closely knit social networks are advantageous. On the other side Burt (1992) argues that rather loosely coupled networks, containing structural holes, are best suited to provide appropriate resources. These networks allow human actors to get divers non-redundant information. Putnam (2000) distinguishes between two types of social capital: bonding social capital relates to social networks within an actors own community, bridging social capital reaches beyond the community boundaries.

In this paper we want to focus on building social capital to foster collaborative learning processes. Cohen and Prusak (2001) argue that social capital offers an interesting new perspective to look at knowledge management. While earlier approaches focused on storing and retrieving explicit knowledge represented in documents, new works deal with implicit knowledge ("knowing" according to Polanyi (1958)), as well. So, research has to be centered around problem-solving capabilities of individual actors and social entities, e.g. communities (Ackerman, Pipek, and Wulf 2003). Contrary to knowledge management, the theories of learning have focused on institutional settings whose primary purpose is knowledge transfer (e.g. schools, or universities). Within the field of learning theories, socio-cultural approaches which focus on knowledge acquisition within communities of practice gain in importance (Lave and Wenger 1991, Wenger 1998). They complement or even replace approaches which focus on individual learning (e.g. behaviorism or

cognitivism). From the point of view of management science, Nahapiet and Goshal (1998) have tried to link the two lines of thought: communities of practice and social capital. Following Schumpeter's (1934) model of knowledge creation, they argue that social capital fosters the generic processes of combining and exchanging knowledge.

Given the importance of social capital for knowledge management and learning, applied computer science needs to take this perspective into account. One way to do this is to investigate how computer applications may contribute to increase social capital. One can think of different roles computer applications may play in increasing social capital:

1. Analysis of existing social capital: Algorithms may be capable to detect inter-personal relation (e.g.: analysis of the frequency of mail exchange). Appropriate visualizations ease the mutual understanding on the current state of a social entities' social capital. These visualizations may also be the starting point of interventions to improve the social capital of a social aggregate.
2. Finding of (unknown) actors: To encourage bridging social capital, algorithms may make actors aware of each other, who have similar or complementary backgrounds, interests, or needs. Therefore, personal profiles have to be created and updated either manually by the actors or automatically by appropriate algorithms.
3. Communication among actors: While actors are often dislocated or need to communicate asynchronously, appropriate applications for computer-mediated communication are needed. This refers specially to the linking between communication channels and artifacts the communication refers to.
4. Building of trust within social entities: To establish and maintain social relations, trust plays an essential role. A computer application may open additional channels among the actors to encourage trust-building.
5. Maintaining social relations: Bridging social capital is often characterized by rather infrequent personal relationships (e.g. among school or university friends, former colleagues). Within highly dynamic environments, there is an ongoing danger that these relationships may fade away (e.g. new addresses after changing jobs). Computer applications may help actors to stay informed about news concerning their old friends (e.g. address changes).

With regard to each of the different roles computer applications may play, one has to reflect critically which may be an appropriate mixture between technologically-mediated and technologically-non-mediated activities.

This paper concentrates mainly on the second aspect. The challenge is to make potentially fitting actors aware of each other in the virtual or in the real space. So these functionalities offer opportunities to introduce actors towards each other by matching or visualizing aspects of their behavior, background, qualification, expertise, or interests. Therefore these functionalities need to grasp, model and evaluate relevant personal data. These data can be either put in manually by the user, they can be automatically grasped, or they can be imported from other applications.

After reviewing the current state of the art, we will present an application which is supposed to foster social capital within an e-learning platform. E-learning platforms allow users to access content structured in a hierarchical way. When grasping personal data automatically, the hierarchical content structure of the e-learning application eases the adding of semantics.

2 State of the Art in Expertise Finding

Research in the field of Computer Supported Cooperative Work (CSCW) and Artificial Intelligence (AI) has created applications which can be understood as technical support for building social capital. Traditionally, CSCW research focused on the support of small working groups containing already a high level of social capital. However, some of the techniques developed in this context can also be used to support the creation of bridging social capital. Recently, CSCW research has also focused on the support of less well connected communities. Social capital can provide an interesting perspective on building applications for such communities. The AI community can contribute to foster social capital because it created algorithms which allows to detect pattern of similarity within unstructured data. These similarities can be used to match actors or make them aware of each other.

Here we want to discuss five research directions in more detail which can contribute to encourage the building of social capital:

- research in expertise profiling systems,
- research in topic oriented communication channels,
- research in discussion and annotation systems,
- research in collaborative/social filtering/recommender systems,
- research on mutual awareness.

The core question in the field of expertise profiling systems is: how to make explicit and implicit knowledge held by individuals visible and accessible to others. In the standard approach to personal profiling systems the actors are asked to input the data describing their expertise or interests by themselves (e.g. yellow pages). However, the creation and maintenance of personal profiles suffer from a couple of difficult problems. First, a common understanding of the different attributes of a personal profile has to be given (Ehrlich 2002). If the profiles are created and updated manually, the different human actors need to have a joint understanding of each attribute. Only in this case their input can be matched automatically. Second, the actors need to be motivated to input and update their personal profiles. Especially the ongoing necessity to update these profiles, threatens their validity (Pipek, Hinrichs and Wulf 2002). Therefore, these data may be complemented by automatically generated data, derived for instance from analysing an actors' home-page or mail traffic. However, automatically generated profiles aggregate data whose semantics are not clear. So it is doubtful whether these data really represent the actors' competencies and interests.

The core question in topic-oriented communication channels is: how do electronic media change communication and social interaction among the actors. There are different approaches to realize topic-oriented communication channels such as newsgroups, mailing lists, MUDs, or MOOS. While the first ones are based on a purely content-oriented structure, the last ones apply a spatial metaphor to structure communication. Experiences demonstrate that topic-oriented communication channels are able to create virtual communities of mutual support (cf. Rheingold 2000; Hafner 2001). In cases where the actors communicate by revealing their personal identity, social relationships - even beyond the virtual space - may be established. In the domain of web-based training, Wessner and Pfister (2001) have proposed that learning platforms may be supplemented with topic-oriented communication channels. To structure and focus users' contribution in learning environments, the authors introduce the concept of Intended Points of Cooperation (IPoCs), i.e. starting points for communication in a learning unit that are defined by the authors of the unit. Specific communication channels (e.g. chat, video conferencing, shared whiteboards) are used to support different types of communication. In the sketched learning environment the critical process of group formation can be performed manually by a tutor (supported by a tool that displays course- and class-related information) or automatically by matching learners who participate in the same course and have not yet completed common IPoCs. So the learners are matched without their direct involvement.

The core question in discussion and annotation systems is: how to support the development or refinement of a mutual understanding on a certain topic by means of computer-mediated discourses, typically in textual form. There are many approaches which combine the presentation of content with integrated functionality to annotate or discuss (e.g. Buckingham Shum 1997; Pipek and Won 2000; Stahl 2003). In case the different contributions to the discussion can be attributed to individual actors, such applications can support building social capital. Active participation in computer-mediated discourses is required to catch other actors' attention. However, this may not always be an appropriate approach. The discourses are typically restricted to a rather specific issue which makes it difficult to transfer a competency demonstrated in a specific discourse on other topics.

The core question in recommender systems is: how to support actors in selecting an item from a set of rather similar items. Several recommender systems are relevant here because they have been designed to support the finding of human actors (cf. Yiman-Seid and Kobsa 2003). Systems like Who Knows (Streeter and Lochman 1988), the Referral Web (Kautz, Selman, and Shak 1997 and 1997a), Yenta (Foner 1997), or MII Expert Finder and XperNet (Maybury, D'Amore, and House 2003) extract personal data about human interests automatically from documents which are created by the actors. Vivacque and Lieberman (2000) have developed a system which extracts personal data concerning a programmer's skill from the Java code the programmer has produced. Based on these personal data the systems allow to pose queries or to match actors. However, these systems have hitherto dealt with specific matching algorithms for one type of personal data. McDonald (2000) and McDonald and Ackerman (2000) developed a framework of an expertise recommendation system that finds people who are likely to have expertise in a specific area. Contrary to the general approaches to expertise matching mentioned above, the framework allows very specific heuristics to be developed that are tailored to the individual organizational context. Thus it does not focus on an automatic evaluation of

many different documents or programs, but on a context specific heuristic. These heuristics need to be revealed by a preceding ethnographic study in the application field. If found, such a heuristic is probably better suited than an automatic algorithms. Like in the approaches mentioned above the heuristic matches experts with people looking for support.

The core question in the field of mutual awareness is: how to make the activities of distributed actors visible to each other. With their study on the importance of mutual awareness for cooperation, Heath and Luff (1991) have motivated a whole series of design approaches. These approaches tried to capture selected activities of individual actors and made them visible to their cooperation partners (e.g. Rodden 1996; Sandor, Bogdan, and Bower 1997; Fuchs 1997 and 1999; Fitzpatrick et al. 2002). With regard to the data captured one can distinguish between structured and unstructured ones. Structured data record the use of a system's functionality, unstructured data typically consists of video streams. The visualization of these data is supposed to compensate for a lack of visibility of individuals' activities and their context in a distributed setting. Awareness features are typically built for groups which contain a high level of social capital and cooperate intensely. However, awareness data and the resulting histories of interaction can also be applied to match people who are not yet well know to each other. For instance, recent approaches try to apply structured awareness data to make individuals aware of other who access the same WWW-site. The Social Web Cockpit provides awareness data which informs users about the presence of other users at a site of interest. Moreover, it allows for collaborative content rating and recommendation functionalities (Gräther and Prinz 2001). Thus, communities can be set up in a self-organized way based on common interests. However the Social Web Cockpit has to be installed as an additional application on each users' computer before benefiting from the matching functionality.

3 Matching Personal Data with Algorithms

As the discussion above has shown, identifying, collecting and maintaining appropriate user personal data is very difficult. The information kept in a profile may stem from many different sources (e.g. manually created interest statements, professional or personal history). Moreover, a great deal of semantic background knowledge may be necessary to realize an algorithmic match-making of experts based on personal data.

With regard to the creation of personal profiles in learning platforms, we are in a rather advantageous position. The structure of the content represented in the platform provides an ontology of the knowledge domain. This ontology can be used to add semantics to automatically recorded data. Specific features of a learning platform, like the results of tests may allow to update personal profiles automatically. So, learning platforms provide semantics and specific data which ease the automatic identification of expertise. Therefore, it is easier to match expertise within learning platforms than in other applications.

In this paper we want to use histories of interaction and awareness data concerning the production and use of the platform's content to create and update personal profiles. Due to the fact that the content is pre-structured, an automatically capturing and processing of these data seems to be promising. In the following we want to show which data are relevant and how to gain semantic information from these data:

- data concerning the production of learning material: actors who have produced specific content for the platform may be experts in this domain,
- data concerning the update of learning material: actors who have updated or refined specific content for the platform may be experts in this domain,
- data concerning tutoring responsibilities: actors who are doing or have done tutoring tasks concerning specific content of the platform may be experts in this domain,
- data concerning test results: actors who have passed tests concerning specific content of the platform may be knowledgeable in this domain,
- data concerning the actual use of certain material: actors who are navigating through specific content of the platform may be interested in this domain,
- data concerning the history of interaction with certain material: actors who are navigating through specific content of the platform may be interested or even knowledgeable in this domain.

Further aspects of the user's profile can be imported from sources outside the learning platform. Keyword vectors or higher order structures derived from an actor's mail (incoming or outgoing) or document production (letters, papers, slides) can be automatically captured as well as those derived from an actor's homepage (Foner 1997; Streeter and Lochman 1988). Further data may be extracted from an automatic evaluation of aspects of actors' task performance (e.g. elements of a programming language used (cf. Vivacque and Lieberman 2000)). These automatically captured data can be supplemented by profile data entered by the user concerning his personal background, interests, or competences. A cross check between manually and automatically created profile data may reveal inconsistencies. These inconsistencies can be indicated at the user interface to initiate an update of the personal profile.

Now the question arises how to apply these data in matching learners, tutors and content provider. The matching algorithms make use of the ontology given by the hierarchical structure of the content. Whenever a learner looks for support it can be located whenever he browses a specific learning unit. With regard to this learning unit, the system can retrieve data about the production history of the content. The creator of a unit as well as the actor who did the last update can be presented to the learner as well as the one responsible for tutoring. In a similar way histories of passed tests can be applied to match learners with those who have already demonstrated capabilities within a certain time span. Finally the matching algorithms allow to identify those actors which are actually browsing the same learning unit or have done so within a certain period of time.

Prior to presenting our approach to expert finding it is necessary to discuss some further requirements for the matching framework. First, we sketch a direct consequence of the discussion of relevant information sources for user profiles (e.g. information on the professional training status, information gained from produced documents, user context and history of interaction, etc.): Quite clearly, each source of information requires a specific method of matching. Thus, the expert matching framework should consist of modules (with well-defined technical interfaces) that encapsulate a certain information type and then contribute to a global matching result by calculating a degree of matching based on that particular type of data (e.g. similarity in learning or project history, interest profile, etc.). Note that this kind of modularity is a pre-requisite for adapting an expert finding component to different application contexts, e.g. different learning platforms or knowledge management environments: Specific matching modules can be exchanged or adapted according to the relevant requirements.

Second, matching expertise affects privacy issues: Learners – or more general: users of any kind of platform that includes expert finding functionalities – might not be willing to make available any kind of personal information to the public. In order to protect the users' right of informational self-determination each user must know and be able to define which of his personal data is used for matching or publication, respectively.

An important problem of matching personal data is the question of 'information quantity': Histories of interaction, e.g., are collected successively and tend to become more expressive with each history item collected. The completeness of personal data may also vary from user to user due to individual privacy decisions. However, the matching quality will depend on the 'completeness' of information available. The more complete a specific type of personal data, the more reliable one can expect the matching result to be. As a consequence for the algorithmic framework a 'degree of completeness' of the different types of personal data should be measured where possible and be taken into account when calculating a matching degree.

Finally, given that a user agrees to use certain types of personal data for the matching process, he should also be able to adapt the expert matching algorithm to a certain degree: The perception of which modules, i.e. types of personal data, contribute to a good profile matching may vary for different users in different contexts. In order to let the user decide which 'factors' contribute to which degree to the expert matching, we propose to incorporate a factor which weighs the impact a certain module has on the overall matching result into the matching framework.

4 A Modular and Adaptable Expert Matching Approach

This section presents an algorithmic framework for expert matching that takes into account the requirements discussed above. In the following we make use of the following terminology: Expert finding means matching a prototype set of personal data (i.e. profile of a certain user in the application environment or a query profile) against a collection of other actors' personal profiles in order to determine a ranking of fitting actors. We distinguish between two modes of using an expert finding component:

- **Filter functionality:** In this mode a user applies the expert finder system in order to find other users with personal data that are similar to his own (or relevant parts of his own data, respectively). This functionality is not only relevant in a learning environment where a learner wants to find other learners with similar backgrounds, interests and knowledge in order to build a learning group. It also applies to organizational knowledge management scenarios: Consider an enterprise environment where an expert in a certain field wants to set up an expert network of people with similar project background in order to share experiences. Alternatively, the user can pose a query (in terms of a user-defined profile) to the expert finder system in order to find people that match *explicitly defined* needs of the user. As an example consider an enterprise environment where an employee needs to find an expert in a certain field who is able to solve a specific problem.
- **Cluster functionality:** Here, the expert finder system is used to cluster the profiles of all users in order to present a “landscape of expertise” for analysis or exploration purposes. Consider an enterprise environment where a project manager tries to identify the expertise of those members of the staff who could potentially take over a certain subtask.

The term “personal data” is deliberately kept very general: With regard to the discussion in section 3 data subsets may include an actor’s professional and training status, interest statements, self-assessment of abilities, certain kinds of history information (learning history in a learning environment, project history in a company’s expert database, etc.) and similar data that describes the user’s expertise depending on the application context.

We now describe the algorithmic matching framework of our expert finder system in a more formal way. We point out the elements for making matches of profiles and define their constraints. Let P denote the set of all possible personal data (i.e. user profiles or query profiles). These data consist of the full set of information available for each user or the query, respectively, in a certain application environment (i.e. a certain learning platform, an enterprise expert database, ...).

As discussed above, there are different kinds of data that help to determine the degree of expertise. According to the nature of the data subsets, different algorithms for matching subsets (e.g. self-assessment, history) have to be applied. In the following this is expressed by the notion of modules: Each module contains functions for matching the respective relevant subsets of personal data (which contain the data items used for matching). Intuitively, each module realizes a criterion for expert matching (e.g. one module for matching histories of interaction, another for matching the training status of users).

Formally, each module \mathcal{M}_i consists of a **matching function** $m_i : P \times P \rightarrow [0,1]$ which determines the degree of similarity¹ for each pair (p_a, p_b) of personal data collections (e.g. the similarity of learning histories) and a **completeness function** $c_i : P \rightarrow [0,1]$ which measures the degree to which relevant data for \mathcal{M}_i is available in a profile p (e.g. the amount of interaction history in the profile, see also section 3).

The matching function is realized for each module. The realization of this function has to meet the following requirements: The more similar two profiles are, the higher the value calculated by m_i (with 1 representing a ‘perfect’ match). Furthermore, matching identical profiles should produce a perfect match (which is quite intuitive), i.e. $m_i(p_a, p_a) = 1$ for all $p_a \in P$. In cases where the clustering functionality is used we also assume m_i to be symmetric, i.e. $m_i(p_a, p_b) = m_i(p_b, p_a)$ for all $p_a, p_b \in P$.

The completeness function takes into account the quantity of data available for module \mathcal{M}_i ’s matching process, where $c_i(p) = 0$ means that no data is available and $c_i(p) = 1$ corresponds to the maximal degree of completeness. For example, assume a module \mathcal{M}_i that matches profiles based on certain history data of a user. If the user is new to the platform in which the expert finder is embedded only few (or even no) history data may be available in p . In this case the matching function m_i might yield a high matching value based on the considered pair of profiles, but this matching would be based on sparse data. In such a case $c_i(p)$ should yield a low value².

In order to allow the user to adapt the matching process he can adjust the influence the different matching criteria have on the overall matching result. Formally, this is done by assigning weights to the single modules: For each module \mathcal{M}_i a

¹ Note that dissimilarity (distance) measures can be used as well since their results can be converted to similarity values.

² Of course certain matching functions may take sparse data into account and yield corresponding matching values. In such a case consider $c_i(p)=1$.

weight $w_i \in [0, 1]$ is given where $w_i = 0$ means that module \mathcal{M}_i is switched off and $w_i = 1$ means that module \mathcal{M}_i has full influence. All values of w_i between 0 and 1 correspond to a more or less strong influence of module \mathcal{M}_i .

We also need to take care of privacy issues: If a user does not want the system to use certain parts of his profile for match-making he should be able to switch off the corresponding modules. Formally, we introduce flags $priv_{ai}, priv_{bi} \in \{0, 1\}$ for all users a and b and each module \mathcal{M}_i which indicate whether the respective user wants module \mathcal{M}_i to be used or not.

Now we can define the overall matching result for profiles p_a and p_b from users a and b :

$$m(p_a, p_b) =_{def} \frac{1}{k} \sum_{i=1}^k priv_{ai} \cdot priv_{bi} \cdot w_i \cdot \min(c_i(p_a), c_i(p_b)) \cdot m_i(p_a, p_b).$$

Summing up, the overall matching value of two profiles p_a and p_b is based on the matching degree of each individual module. A module can only contribute to the overall result if both, user a and user b , agree that their profile may be used for match-making by setting the respective flag to 1. Furthermore, the contribution of each module depends on the completeness (and thus, as we claim, trustworthiness) of the respective profile data and the user-defined weighting for that module.

5 Realization of the Expert Finder

Based on the methodical approach presented in this paper we built an expert finder system for the Fraunhofer e-Qualification framework³ – an e-learning environment that offers extensive technical, methodical and didactical support for both, authors of Web-based trainings and learners. We therefore adapt the generic architecture of our system to the e-Qualification platform and provide a user interface as presented in section 5.2 which mainly focuses on the filter functionality of our framework. In section 5.3 we sketch another application of our framework. In this case the actors' personal data are input for a cluster functionality and an advanced visual interface for exploring a "landscape of expertise".

5.1 Architecture

The architecture (cf. figure 1) we have chosen for our expert finding system has been designed in order to keep the system flexible and easily adaptable to different application platforms. The current architecture requires that the application platform provides a Web-based interface. The expert finding system consists of three mayor parts: The *expert finder* itself (which realizes the algorithmic matching framework described above), the *connection to the application environment* (e.g. a learning platform) and the *connection to the client* (i.e. Web browser). The expert finding system contains internal databases where information about the user and the content of the application environment is 'cached'. This is done to have quick access to those information that are frequently used by the expert finder, and to store additional data about users and content which is not kept in the main-databases of the application environment. The connection to the learning-environment is done by 'adapter objects' that translate personal data and content information from the application environment's proprietary format into the data structures used by the expert finder. The connection to the client is realized by a Java™ Servlet that forwards requests to the expert finder and generates HTML-codes from the computed matching results.

³ <http://www.e-qf.de>

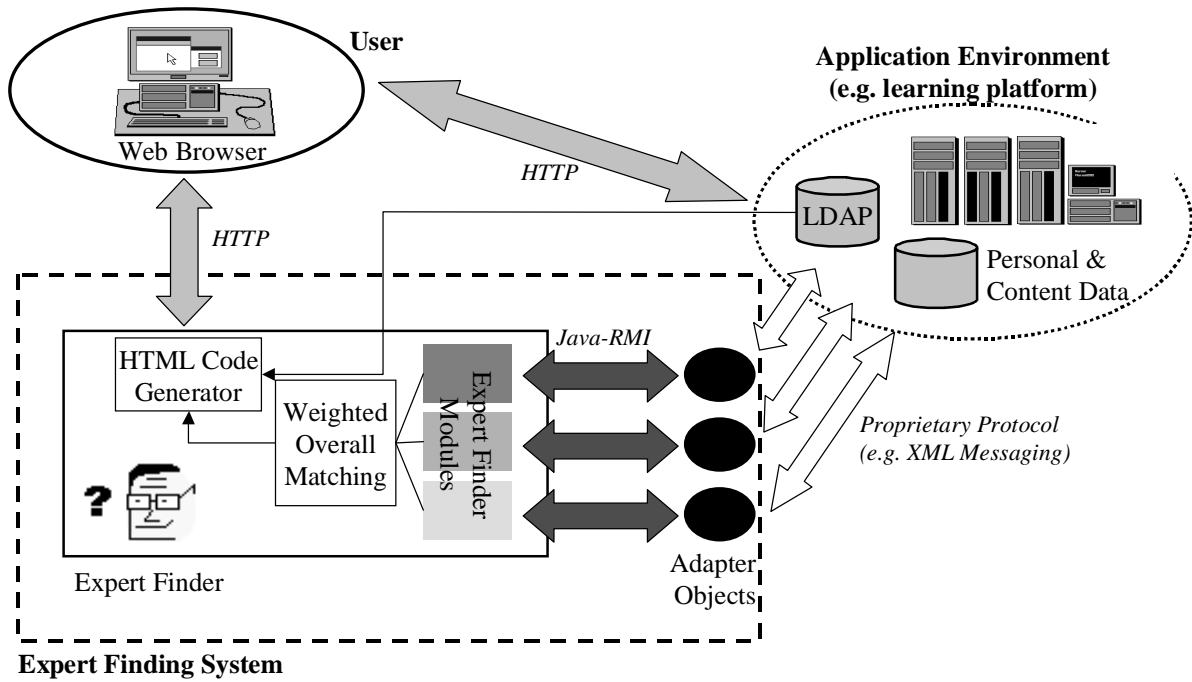


Figure 1: Architecture of the expert finding system

5.2 User Interface

Figure 2 depicts the user interface of the expert finder within the Fraunhofer e-Qualification platform: Users of the platform who are logged in as learners can use the expert finder system in order to contact authors, tutors and suitable co-learners. For learners, the main window of the expert finder is available during the complete training session (either as a separate window or integrated in the training web pages as a frame layout). Learners can seek advice from the author of their whole session or specifically assigned tutors by clicking on the respective symbol in the main window. In both cases the result of the user request is the name, e-mail address and telephone number of the author(s) or tutor(s), respectively.

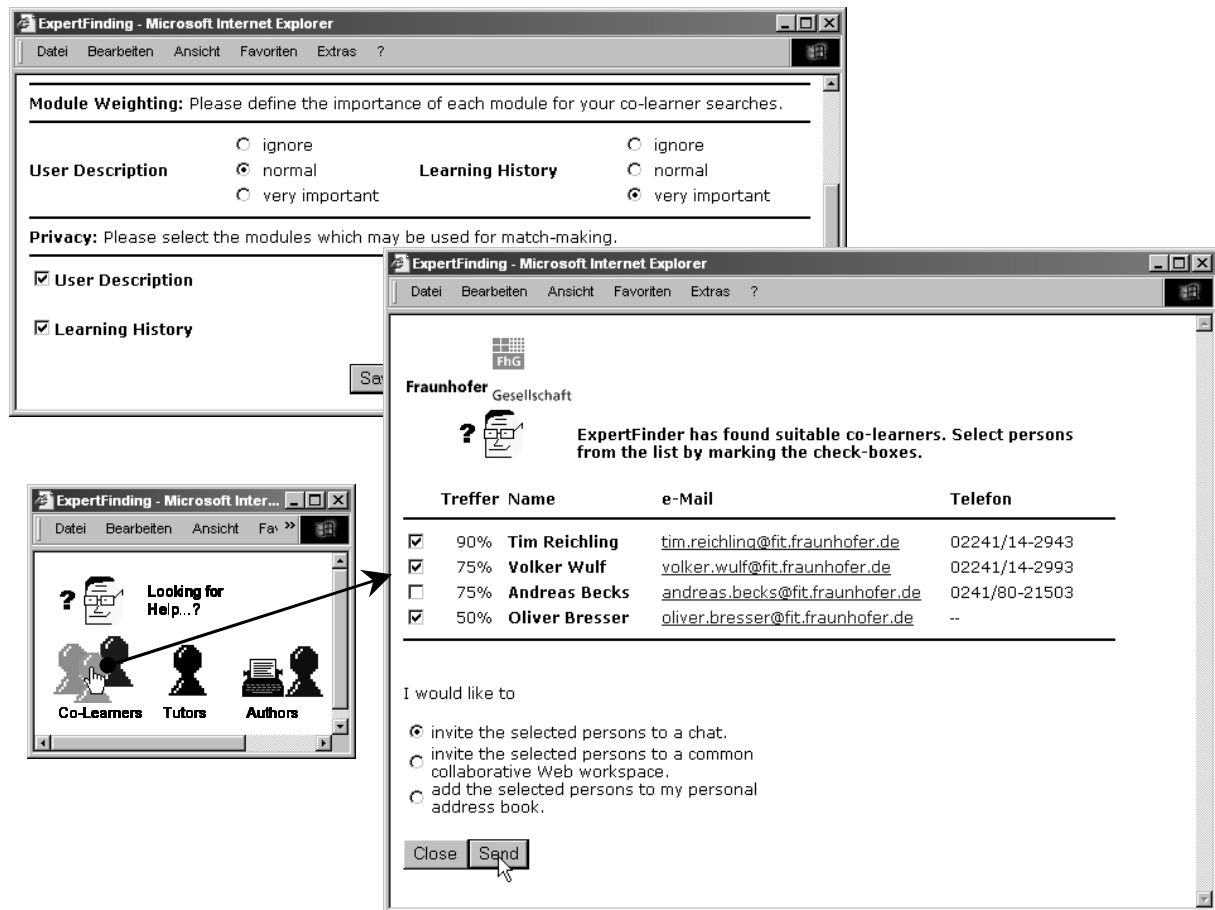


Figure 2: The expert finder interface of the e-Qualification platform: main window of the expert finder (bottom left), result list of co-learner search (bottom right), options window for configuring modules for co-learner search (top left)

When the learner looks for potential co-learners he can click on the corresponding icon in the main window (or expert finder frame). As a result of this request the learner directly receives a list of suitable co-learners, ranked by the matching degree of his own profile and the profile of each candidate co-learner (thus realizing the filter functionality of our framework, cf. section 4). The user can then select potential co-learners from the list and either invite them to a chat (as a means of synchronous communication), ask them to join a common collaborative workspace within the learning platform (as a means of asynchronous communication), or just add them to his personal address book for later reference.

In a specific options window the user can define a relative weighting of the modules used to assess the suitability of other users of the platform as co-learners, thus determining the influence each module has on the matching result. In this case the modules "learning history" and "user description" are available. The latter contains, among others, educational status and interest statements of each learner subscribed to the platform. The option window also allows to adjust the learner's individual privacy flags. The privacy flags are used globally, i.e. as soon as a learner deactivates a flag, the corresponding profile data will not be used by the expert finder at all. The module weights, on the other hand, are used locally: If a learner selects "ignore" for one of the modules, this only affects his co-learner searches. For the searches of other learners the respective personal data is still available.

5.3 Expert Maps: Clustering Profiles for Exploration

In this section we sketch another application of our expert finder framework where the calculated similarity information of personal profiles is used to automatically calculate a graphical "landscape of expertise": In larger traditional enterprises or in virtual organizations similar or complementary expertise is usually spread over different departments or sites, making it difficult to get an overview of the areas of competence available in the organization. We conducted informal interviews in several medium-sized companies which were interested in advanced knowledge management technology. The study revealed that a graphical presentation of an "expertise landscape" would be of high value. We therefore

propose to combine the clustering functionality of our framework with a technique for visualizing the structure of complex information spaces, originally developed for data and document mining (Becks et al. 2000). Given a degree of similarity for each pair of actors (e.g. the similarity of each pair of personal profiles), the visualization approach automatically generates an overview map (cf. figure 3) which intuitively displays the relationships of actors which are represented as points in the map; similar collections of personal data are grouped as neighbored points – the more similar, the closer they appear in the display. In addition, the shades of gray indicate additional similarity information: Groups of similar actors can be found in the bright-shaded areas. Dark borders separate these groups. The darker the border, the more dissimilar are the respective profile groups.

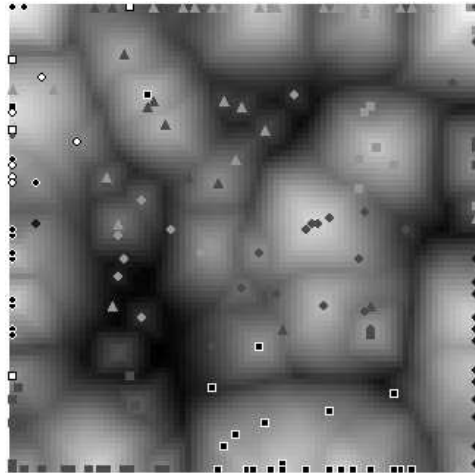


Figure 3: An expert map which displays the similarity structure of all pairs of personal profiles can be used for navigating through and exploring an "expert landscape".

The map in figure 3 presents a landscape of expertise based on personal data of employees in a medium-sized company. The color and shape of profile points in the map represent the department or site the respective employee (who is represented by the personal profile) works in. Applications of such expert maps (as discussed with our industry partners) would include the identification of competence areas and their distribution within the company or the identification of only sparsely covered areas of expertise. In other applications project managers would use the map for identifying suitable team members by browsing through the competence landscape, and innovation managers would apply the map for identifying groups of people with related or complementary personal profiles in order to set up networks of experts for experience exchange and creative brainstorming sessions. Initial discussions of the concept with our industry partners yielded positive feedback.

6 Conclusion

The concept "social capital" has recently gained importance. Social capital seems to have an impact on different aspects of the performance of social entities. The application of information technology is likely to influence the level of social capital. In this paper we have looked at computer applications to encourage human learning. We assume that a high level of social capital among the learners will have a positive effect on their performance. Given the fact that actors within e-learning platforms will not know each other in general, we have developed a framework for match-making. Contrary to face-to-face learning environments, e-learning environments do not allow users to meet in person. So many of the real-world clues which enable the establishment of initial forms of social relationships among learners are missing (e.g. behavior, gesture, facial expressions, or eye contact). The question comes up whether there can be virtual substitutes which may ease the establishment of social relationships. In this paper we have explored ways to match the learners' personal data. We have developed a framework which allows to match different subsets of personal data in a weighed manner. Depending on the subset of personal data, the matching algorithms may favor similarity or complementarity.

Many research issues discussed here need further investigation. First of all an empirical evaluation of our approach is required. In particular, a key question concerns the value added by automatic match-making. This question has to be addressed in a number of smaller steps. First, quite clearly, the quality of the matching needs to be assessed by the target users. For this, we plan to apply the match-making strategy in an in-house study based on profiles of our colleagues. A following series of interviews will then be conducted in order to find out whether the suggested partners

are regarded to be an appropriate choice. Second, given a sufficient quality, an essential question is whether users are willing to establish social contacts based on the recommendations of computer algorithms. This can only be assessed in current learning environments. We will therefore integrate our expert finding system to several learning environments and collect data. Possible information can be extracted from log files which show the communication activities (e.g. chat, e-mail) started in response to the system's recommendations. More important, semi-structured interviews with platform users need to be conducted in order to learn from the users' reactions. Moreover, refinements of the different matching algorithms are required. We have to find out which are the right subsets of personal data and which algorithms are best suited to match a certain subset of data. As the application context will have a strong influence on the choice of the appropriate algorithms, the expert finding components should be highly tailorable. Appropriate concepts to tailor these applications have to be developed.

Further research is required to find out how to support the building of relationships after having matched the actors. Questions arise such as: which private data should become visible to the matched partners, what are the appropriate media for communication, which role may face-to-face meetings play in such an environment? Another research issue refers to the preconditions necessary to successfully match actors. Probably a somehow common language and culture is an important prerequisite to establish contact between the actors. These factors cannot be grasped easily by means of personal data.

We have approached the problem of how to support the establishment of social relationships in the virtual world where many traditional clues for self-selection among the actors are missing. Our approach to expertise matching in an e-learning environment needs empirical evaluation and further refinements. However, the study indicates that there is potential to support the building of social capital by technical media. A thorough investigation of potential risks and opportunities is required.

References

- Ackerman, M. S.; Pipek, V.; Wulf, V. (eds): *Expertise Sharing: Beyond Knowledge Management*, MIT Press, Cambridge MA 2003
- Becks, A., S. Sklorz, M. Jarke: *Exploring the Semantic Structure of Technical Document Collections: A Cooperative Systems Approach*. Fifth IFICIS International Conference on Cooperative Information Systems (CoopIS'2000), Eilat, Israel, September 6-8, 2000
- Bourdieu, P.: *The forms of capital*, in: J.G. Richardson (ed.): *Handbook for Theory and Research for the Sociology of Education*, 1985, pp. 241 – 258
- Buckingham Shum, S. : *Negotiating the Construction and Reconstruction of Organisational Memories*; in: *Journal of Universal Computer Science (Special Issue on IT for Knowledge Management)*, 3 (8), 1997, 899-928,
- Burt, R.S.: *Structural Holes: The Social Structure of Competition*, Harvard University Press, Cambridge 1992
- Cohen, D.; Prusak, L.: *In Good Company: How Social Capital makes Organizations Work*, Harvard Business School Press, Boston 2001
- Ehrlich, K.: *Locating Expertise: Design Issues for an expertise locator system*, in: Ackerman, M.; Pipek, V.; Wulf, V. (eds): *Expertise Sharing: Beyond Knowledge Management*, MIT-Press, Cambridge MA 2003, pp. 137 - 158
- Fitzpatrick, G.; Kaplan, S.; Mansfield, T.; Arnold, D.; Segall, B.: *Supporting Public Availability and Accessibility with Elvin: Experiences and Reflexions*, in: *Computer Supported Cooperative Work: The Journal of Collaborative Computing (JCSCW)*, Vol. 11, No. 3-4, 2002, pp. 447 - 474
- Foner, L. N.: *Yenta: A Multi-Agent, Referral-Based Matchmaking System*, in: *First International Conference on Autonomous Agents (Agent'97)*, ACM-Press, New York 1997, pp. 301 - 307
- Fuchs, L.: *Situationsorientierte Unterstützung von Gruppenwahrnehmung in CSCW-Systemen*, PhD-Thesis, Department of Computer Science, University of Essen, 1997

- Fuchs, L.: AREA: A Cross Application Notification Service for Groupware, in: *Proceedings of the Sixth European Conference on Computer Supported Cooperative Work - ECSCW '99*, Kluwer, Dordrecht, September 1999, pp. 61 - 80.
- Gräther, W.; Prinz, W.: The Social Web Cockpit: Support for Virtual Communities. Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work, Boulder, Colorado, USA, 2001
- Hafner, K.: *The Well*, Carroll & Graf, New York 2001
- Heath, C; Luff, P: Collaborative Activity and Technological Design: Task Coordination in London Underground Control Rooms, in: *Proceedings of the Third European Conference on Computer Supported Cooperative Work - ECSCW '91*, Kluwer, Dordrecht, Amsterdam, September 1991, pp. 65 - 80.
- Kautz, H. A.; Selman, B.; Shak, M.: Referral Web: Combining Social Networks and Collaborative filtering, in: *Communication of the ACM*, Vol. 40, No. 3, 1997, pp. 63 - 65
- Kautz, H. A.; Selman, B.; Shak, M.: The hidden web, in: *AI Magazine*, Summer, 1997a, pp. 27 - 36
- Lave, J. und E. Wenger: *Situated Learning: Legitimate Peripheral Participation*. Cambridge University Press, Cambridge 1991
- Maybury, M.; D'Amore, R.; House, D.: Automated Discovery and Mapping of Expertise, in: in: Ackerman, M.; Pipek, V.; Wulf, V. (eds): *Expertise Sharing: Beyond Knowledge Management*, MIT-Press, Cambridge MA 2002, pp. 359 - 382
- McDonald, D. W. : Supporting Nuance in Groupware Design: Moving from Naturalistic Expertise Location to Expertise Recommendation, PhD-thesis, University of California, Irvine 2000
- McDonald, D. W.; Ackerman, M. S.: *Expertise Recommender: A Flexible Recommendation System and Architecture*, in: *Int. Conf. on CSCW*, ACM Press, New York 2000, pp. 231-240
- Nahapiet, J.; Ghosal, S.: Social capital, interlectual capital, and the organizational advantage, in: *The Academy of Management Review*, , Vol. 23, No. 2, 1998, pp. 242 - 266
- Pipek, V.; Hinrichs, J.; Wulf, V. : Sharing Expertise: Challenges for Technical Support, in: Ackerman, M.; Pipek, V.; Wulf, V. (eds): *Expertise Sharing: Beyond Knowledge Management*, MIT-Press, Cambridge MA 2002, pp. 111 – 136
- Pipek, V.; Won, M.: *Communication-oriented Computer Support for Knowledge Management*. *Informatik/Informatique - Magazine of the Swiss Informatics Societies* 2002(1): 39-43.
- Polanyi, M.: *Personal Knowledge: Towards a Post-Critical Philosophy*, London, Routledge & Kegan 1958
- Putnam, R.: The prosperous community: social capital and public life, in: *American Prospect*, Vol. 13, 1993, pp. 35 – 42
- Putnam, R.: *Bowling alone: The collapse and Revival of American Community*, Simon & Schuster, New York 2000
- Rodden, T.: Populating the Application: A Model of Awareness for Collaborative Applications, in: *Int. Conf. on CSCW 1996*, ACM Press, New York 1996, pp. 87 - 96
- Rheingold, H.: *The Virtual Community*, MIT-Press, Cambridge MA, 2000
- Sandor, O.; Bogdan, C.; Bowers, J.: Aether: An Awareness Engine for CSCW, in: *Proceedings of the Fifth European Conference on Computer Supported Cooperative Work - ECSCW '97*, Kluwer, Dordrecht, September 1997, pp. 221 - 236
- Schumpeter, J. A.: *The theory of economic development: An inquiry into profits, capital, credit, interest and the business cycle*, Harvard University Press, Cambridge 1934
- Stahl, G.: Building Collaborative Knowing- Elements of a Social Theory of Learning, in: Strijbos, Kirschner, and Martens (eds): *What We Know About CSCL in Higher Education*, Kluwer, Dordrecht 2003 in press

Streeter, L. A.; Lochman, K. A.: An expert/expert location system based on an automatic representation of semantic structure, in Proceedings of the Forth Conference on Artificial Intelligence Applications, San Diego, CA, March 14 - 18, 1988, ACM-Press, New York 1988, pp. 345 - 350

Vivacque, A.; Lieberman, H.: Agents to assist in finding help, in: Proceedings in the Conference on Computer Human Interaction (CHI 2000), ACM-Press, New York 2000, pp. 65 - 72

Wenger, E.: Communities of Practice, Cambridge University Press, Cambridge 1998

Wessner, Martin, Hans-Rüdiger Pfister: Group Formation in Computer-Supported Collaborative Learning, Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work, Boulder, Colorado, USA, 2001

Yiman-Seid, D.; Kobsa, A.: Expert Finding Systems for Organizations: Problem and Domain Analysis and the DEMOIR Approach, in: Ackerman, M. S.; Pipek, V.; Wulf, V. (eds): Expertise Sharing: Beyond Knowledge Management, MIT-Press, Cambridge MA 2003, pp. 327 - 358