FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Measuring User Expertise in Online Communities

DISSERTATION

zur Erlangung des akademischen Grades

Doktor der Sozial- und Wirtschaftswissenschaften

eingereicht von

Martin Hochmeister
Matrikelnummer 9825597

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung: Ao.Univ.Prof. Dipl.-Inf. Dr.-Ing. Jürgen Dorn

Diese Dissertation haben begutachtet:

(Ao.Univ.Prof. Dipl.-Inf. Dr.-Ing. Jürgen Dorn)	(Assoc. Prof. Dipl. Ing. Dr. Hilda Tellioglu)

Wien, 31.05.2012

(Martin Hochmeister)

FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

Measuring User Expertise in Online Communities

DISSERTATION

submitted in partial fulfillment of the requirements for the degree of

Doktor der Sozial- und Wirtschaftswissenschaften

by

Martin Hochmeister
Registration Number 9825597

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: Ao.Univ.Prof. Dipl.-Inf. Dr.-Ing. Jürgen Dorn

The dissertation has been reviewed by:

(Ao.Univ.Prof. Dipl.-Inf. Dr.-Ing. Jürgen Dorn)

(Assoc. Prof. Dipl. Ing. Dr. Hilda Tellioglu)

Wien, 31.05.2012

(Martin Hochmeister)

To my wonderful parents, Franziska and Gerhard.

Erklärung zur Verfassung der Arbeit

Martin Hochmeister
Puchsbaumplatz 11/41, 1100 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

(Ort, Datum) (Unterschrift Verfasser)

Acknowledgements

This doctoral work would not have been possible without the great support of my advisor Prof. Jürgen Dorn. He always stayed calm when I was impatient. He encouraged me many times to finally apply for a dissertation fellowship that in the end provided me with the appropriate environment to accomplish my mission. Besides, I am very grateful for all the spontaneous meetings, which I certainly did not take for granted, and the valuable advice emerging from them. It is unlikely that I would have ever seen Australia without Prof. Dorn raising the idea to visit a former colleague down there to discuss my research issues with. Furthermore, I realized that it is of extraordinary value to know that there is someone who backs you up when others smash your work into pieces.

While focussing on my dissertation studies I found myself canceling a lot of social events. In this regard, I want to express my deep gratitude to my family, especially to my parents Franziska and Gerhard, my sisters Petra and Isolde, and my nephew Lukas for their appreciation as well as for their persistent and outstanding support. I also like to thank my friends for their limitless patience.

During the time I worked on this thesis I enjoyed being a member of the EC Group at the Vienna University of Technology. I would like to thank in particular: Prof. Hannes Werthner who consistently reminded me that the most important goal of a PhD student is "to finish", Prof. Dieter Merkl who repeatedly found time to give valuable feedback, and my colleagues Christoph Grün, Michael Pöttler, Nick Tahamtan and Thomas Motal who were always available for proof-reading papers and discussing common PhD issues.

In the course of my visit to Australia I met a few people who finally turned out to be highly crucial for my undertaking. I would like to thank Prof. Markus Stumptner from the University of South Australia for his offer to stay for a couple of months at his department and share my ideas with him and his colleagues Georg Grossmann, Wolfgang Mayer, Andreas Jordan and Gavin Smith. I also would like to thank Prof. Judy Kay from the University of Sydney who not only provided her feedback on parts of this thesis but also became a great partner in co-authoring papers. Moreover, I extremely appreciate the support of Prof. Ulrike Gretzel from the University of Wollongong who invited me to give a talk at her department and introduced me to a lot of fellow researchers down there.

Last but not least, I want to thank the Österreichische Forschungsförderungsgesellschaft (FFG) for funding my work and thus giving me the opportunity to focus on writing papers, giving talks at conferences and meeting up with other researchers for the exchange of ideas.

Abstract

The thesis at hand addresses the challenge to identify and measure expertise of individuals. This task is highly relevant since the location of individuals' expertise is crucial to organizations in order to assign the most appropriate people to given tasks. Such effective assignments support organizations in sustaining competitive advantage as well as in fostering innovation. However, the elicitation of expertise is challenging since knowledge resides first and foremost in the heads of individuals and thus is inherently elusive.

We iteratively develop a method to quantify users' expertise based on their submissions to online communities. An online community offers a communication platform to its users that facilitates the informal exchange of knowledge. As a consequence, when people share their experiences in problem-solving contexts, they demonstrate expertise regarding certain topics. The proposed method aggregates data obtained from such an online community and automatically generates users' expertise models containing expertise topics along with users' expertise levels. Thereby, expertise levels correspond to numerical values on an absolute scale. Expertise levels mapped on an absolute scale allow to compare one's expertise with others' as well as to staff teams according to the expertise levels needed.

To evaluate the proposed method we conduct a series of experiments with students at our university. Since the method constitutes a composite of various calculation steps, each experiment covers either a specific step or several steps of the proposed method. We set up hypotheses that are based on each other to systematically explore both the characteristics of the method and the value of users' submissions to reliable expertise calculation. The method's calculation accuracy is measured by comparing the calculated expertise levels with the participants' self-assessments.

Kurzfassung

Die vorliegende Arbeit beschäftigt sich mit der Identifikation und Messung von individueller Expertise. Unternehmen, die präzise über die Expertise ihrer Mitarbeiter Bescheid wissen, können diese effektiv bestimmten Unternehmensaufgaben zuordnen. Der optimale Einsatz von Wissen im Unternehmen ermöglicht den Ausbau und die Wahrung von Wettbewerbsvorteilen. Der Zugriff auf individuelles Wissen ist jedoch nicht trivial, da Wissen in erster Linie personenbezogen ist und nicht direkt beobachtet werden kann.

Im Rahmen dieser Dissertation entwickeln wir iterativ eine Methode zur Quantifizierung von Expertise basierend auf den Beiträgen von Nutzern in einer Online Community. Online Communities repräsentieren eine Plattform zum informellen Austausch von Wissen. Die Mitglieder einer Online Community demonstrieren ihre Expertise im Zuge der gemeinsamen Lösung von Problemen. Die vorgestellte Methode bedient sich dieses Wissensaustausches und generiert daraus individuelle Expertenprofile. Die Expertise zu einem bestimmten Fachthema wird dabei mit einem berechneten Expertenniveau assoziiert. Die Bestimmung von Expertenniveaus ermöglicht das Vergleichen von Experten als auch die gezielte Besetzung von Stellen basierend auf gegebenen Anforderungsprofilen.

Die Methode zur automatisch Berechnung von individueller Expertise wird anhand mehrerer Experimente mit Studenten evaluiert. Der Prozess zur Berechnung von Expertenniveaus gliedert sich in mehrere Schritte. Die durchgeführten Experimente beziehen sich entweder auf einen spezifischen Schritt der Berechnung oder auf die Evaluierung mehrerer Schritte. Aufeinander aufbauende Hypothesen bilden die Grundlage für die systematische Untersuchung der Eigenschaften der Methode. Zudem dient die Bearbeitung der Hypothesen zur Bestimmung der Wertigkeit von bestimmten Nutzerbeiträgen zur akkuraten Berechnung von Expertenniveaus. Die Berechnungsgenauigkeit der präsentierten Methode wird auf Basis der Selbstbewertungen der Studenten ermittelt.

Contents

CHAPTER 1

Introduction

If you can not measure it,
you can not improve it.

William Thomson - Lord Kelvin
(1824 - 1907)

Knowledge is well recognized as a crucial resource to sustain competitive advantage [Davenport and Prusak, 1998] and to stimulate innovation [Du Plessis, 2007]. This is in particular true in knowledge-intense domains where organizations compete in uncertain and dynamic environments [Miller and Shamsie, 1996]. In order to maintain competitive advantage, organizations must efficiently and effectively create, locate, capture, and share the organization's knowledge and expertise [Zack, 1999]. Basically, two types of knowledge are distinguished, i.e., knowledge of individuals and organizational knowledge. An individual's knowledge consists thereby of a theoretical part (knowledge not being applied yet) and a practical part (knowledge based on experience). On the contrary, organizational knowledge constitutes knowledge of individuals applied in an organizational context to accomplish tasks of various kinds to reach the respective organization's goals [Tsoukas and Vladimirou, 2001]. Hence, even though knowledge resides on an individual as well as organizational level it is highly interconnected. [Reinhardt and North, 2003] suggest the need to systematically integrate these levels in favor of a goal-oriented utilization of knowledge. In resource-based theory, sustained competitive advantage is derived from an enterprise's internal resources as long as they add value, are unique or limited and are difficult to imitate by competitors [Foss and Knudsen, 1996].

Due to the importance of knowledge for business and industry, management frameworks have emerged that efficiently exploit knowledge to achieve enterprises' business goals. From this, a unique discipline arose called Knowledge Management. [Probst et al., 2006] identify six core processes for knowledge management, i.e., knowledge identification, acquisition, development, distribution, utilization and knowledge storage. These processes are designed to handle knowledge on both levels, the individual as well as the organizational one.

Knowledge is a Complex Construct

Knowledge represents a complex and multi-faceted concept. In the past, researchers raised several perspectives on knowledge, which, for instance, distinguish a tacit and explicit kind of knowledge [Nonaka and Takeuchi, 1995]. [Spender, 1996] suggests further aspects besides tacit and explicit knowledge, i.e., individual and collective knowledge. However, all the aforementioned authors build on the influential work of [Polanyi, 1966], who hypothesized that "we can know more than we can tell". In this sense, current research understands tacit knowledge as knowledge that is not easily communicated and only exists in people's minds. Tacit knowledge is demonstrated in people's actions, experience and involvement in specific contexts [Alavi and Leidner, 2001]. In contrast, explicit knowledge is captured and explained quite easily, like knowledge explicated in textbooks or certain procedures describing how to achieve something.

Firms proactively managing their employees' tacit and explicit knowledge for solving corporate problems have a major competitive advantage [Smith, 2001]. Therefore, to efficiently allocate knowledge resources, information systems facilitating knowledge management should also consider the identification of tacit knowledge [Alavi and Leidner, 2001]. However, this is challenging since tacit knowledge is inherently elusive. Two approaches are distinguished to locate tacit knowledge. The first one refers to the process of making tacit knowledge explicit whereas the second approach is based on knowledge about specific persons who possess the tacit knowledge needed to accomplish a certain task. The process of making knowledge explicit has its roots in the early days of Artificial Intelligence where so called Expert Systems were supposed to behave in a problem-solving setting like a human expert would do [Waterman, 1986]. Hence, this requires the system to possess theoretically the same knowledge as the human expert has available. Specifying rules for such a knowledge base is challenging because of different reasons, e.g., human experts might approach a problem in different ways or engage in different thought processes during problem-solving. In order to process tacit knowledge electronically, we need to make it explicit. [Stenmark, 2000] describes the challenges of such a process. First of all, people are not necessarily aware of their tacit knowledge. Secondly, when applying tacit knowledge, we do not need to make it explicit. And lastly, tacit knowledge is a personal asset to retain competitive advantage with respect to other people working in the organization.

> *PROBLEM: Tacit knowledge is a driver of competitive advantage, but it is difficult to measure.*

Systems Supporting Knowledge Management

Organizations managing their knowledge effectively need to (1) understand their strategic knowledge requirements, (2) devise a knowledge strategy that is aligned with their business strategy, and (3) implement an organizational and technical architecture that suits the firm's knowledge-processing needs [Zack, 1999]. As required by the latter point, information systems play an important role in supporting knowledge management processes. Not only as a repository of

knowledge but also to facilitate knowledge-sharing amongst people [Sharratt and Usoro, 2003]. To know who knows what in an organization is crucial for effective knowledge management. For instance, during the design of the knowledge strategy, organizations need to know if strategically required knowledge is already available amongst the staff members or needs to be developed by conducting certain training activities. In such cases, it is crucial for systems to identify, index and distribute knowledge of individuals appropriately. Two examples for such systems are so called Expert Finders and Intelligent Tutoring Systems.

Expert finding is a crucial task for corporations to sustain competitive advantage. In particular, expert finders help people with their need to seek the best suitable candidates to either perform given tasks or to simply act as sources of information [Seid and Kobsa, 2003]. Such systems support users in discovering subject matter experts and thus make organizations more efficient and effective in that they help to accelerate research and development as well as to enable a rapid staffing process of teams [Maybury, 2006]. Reliable and accurate user expertise models are essential for expert finders to effectively locate experts.

In case training is needed to acquire new or enhance existing knowledge, staff members change their roles from users seeking others for help to learners studying new topics. As a kind of adaptive educational systems, intelligent tutoring systems adapt learning resources to learners based on their learner models. Learning resources include learning content, learning paths that may help navigating through appropriate learning resources or relevant peer-learners, with whom collaborative learning may take place [Manouselis et al., 2011]. [Berio et al., 2005] underline the need for knowledge management systems to consider an e-learning component to support the process of competence acquisition. However, similar to expert finders, these systems perform poorly until they collect sufficient information about learners. Thus, expert finders as well as intelligent tutoring systems may improve their services by exploiting more comprehensive and accurate learner models.

> *PROBLEM: Information systems supporting knowledge management suffer from inaccurate and incomplete representations of the underlying user models.*

The Scope of this Thesis

In this thesis, we address the aforementioned problems by means of indirectly locating tacit knowledge that is indexed in online communities, in favor of gaining richer user models by which knowledge management systems may improve their services. We aim to provide a method towards the automatic measurement of expertise to lessen the burden of users engaged in time-consuming and tedious self-assessments. The proposed method is related to the knowledge identification process as mentioned earlier.

While most adaptive systems gather detailed information about users in their particular application domain, they forget that the same users are also involved in other digital environments such as social network sites or online communities. Information systems may enhance their user models with information from external data sources. In this regard, systems are required to understand the user more as a person with manifold attributes other than relying on application-specific data [Liu and Maes, 2005].

Communities of practice (CoP) [Lave and Wenger, 1991] seem to represent a promising source to gain additional user data for profiling. CoPs are self-organizing systems comprising people that are united in action. Such CoPs are informal structures where people are glued together by their specific shared problems or interests. A company's competitive advantage is largely embedded in the intangible, tacit knowledge of its employees and this knowledge is strictly bound to the people's minds [Dougherty, 1995]. However, [Horvath and Sternberg, 1999] have observed that people use tacit knowledge while telling stories to peers. Based on this, [Ardichvili et al., 2003] suggest to help people sharing tacit knowledge by allowing them to talk about their experiences and also exchange knowledge while solving problems together. In contrast to team members, people in CoPs can offer advice on a project without the risk to get entangled in it. [Wenger et al., 2002] found out that "many of the most valuable community activities are the small, everyday interactions [and] informal discussions to solve a problem."

Lately, we observe the emergence of numerous types of online communities adopting the notion of CoPs mediated by information systems. For instance in Community-driven Question Answering (CQA), community members respond online to a posted problem by sharing what they know. Examples of such CQA communities are online platforms like Yahoo! Answers[1], Answerbag[2] and StackExchange[3]. In general, a crucial factor for an online community's success is its members' motivation to actively participate in knowledge-sharing activities. [Ardichvili et al., 2003] explore possible motivations and barriers for members' active contribution. They found that employees are reluctant to contribute out of fear of criticism. This is mainly caused through their belief that contributions may be not as important than others', they might be not completely accurate/wrong or even not of the community's interest. On the other hand, employees actively participate to establish themselves as experts. [Wasko and Faraj, 2005] suggest that people contribute when they have the experience to share and when they feel to be part of the network. They also suggest that contributions occur without expecting reciprocity from others.

Individuals' expertise is highly dependent on tacit knowledge, and "it can often only be observed and recognized through its resulting actions" [Stenmark, 2000]. Given this relationship

[1] http://answers.yahoo.com/

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

[3] http://www.stackexchange.com/

between expertise and tacit knowledge, we aim at measuring users' expertise in online communities based on their contributions representing users' experience in real-world situations. We will focus especially on online communities where members gather to collaborate in problem-solving tasks. Within this joint work, people help others by explaining how they would successfully solve certain issues. In particular, they explicate knowledge that they would not describe in such detail if they had to solve the given task on their own. Thus, we assume that these informal communications allow for the elicitation of tacit user knowledge. At least to an extent that may help to model users' expertise in a more accurate and comprehensive way.

1.1 Research Questions

In this thesis, we aim to explore users' expertise applicable to specific work within a certain domain referred to as *technical tacit knowledge* [Alavi and Leidner, 2001]. In particular, we address and evaluate the need to describe one's knowledge by means of expertise levels [McDonald and Ackerman, 1998] [Alavi and Leidner, 2001] [Berio et al., 2005]. Modeling user expertise is a challenging task because of several reasons such as the lack of access to information about users' past performances as well as due to the lack of standards specifying the necessary criteria to reach a certain level of expertise. Furthermore, expertise continuously changes over time, which has to be considered in the long run. The main research question guiding this thesis reads as follows:

> *Can we reliably quantify users' technical expertise based on their contributions in an online community?*

In particular, we explore ways to quantify expertise as well as to present expertise to the users for scrutiny. Based on the main research question we derive a set of more specific questions as listed below.

Q.1: Can we consistently quantify users' expertise levels on an absolute scale?

Q.2: Can we determine a confidence level to express the reliability of expertise predictions?

Ontological user models provide valuable information about the relationship between users' attributes. Given this structural information, we ask further:

Q.3: Can we determine a user's expertise in topic Y based on the user's expertise in topic X by exploiting the direct or indirect linkage given in the competence ontology?

1.2 Main Contributions

Figure 1.1 illustrates the big picture of the research conducted in this thesis.

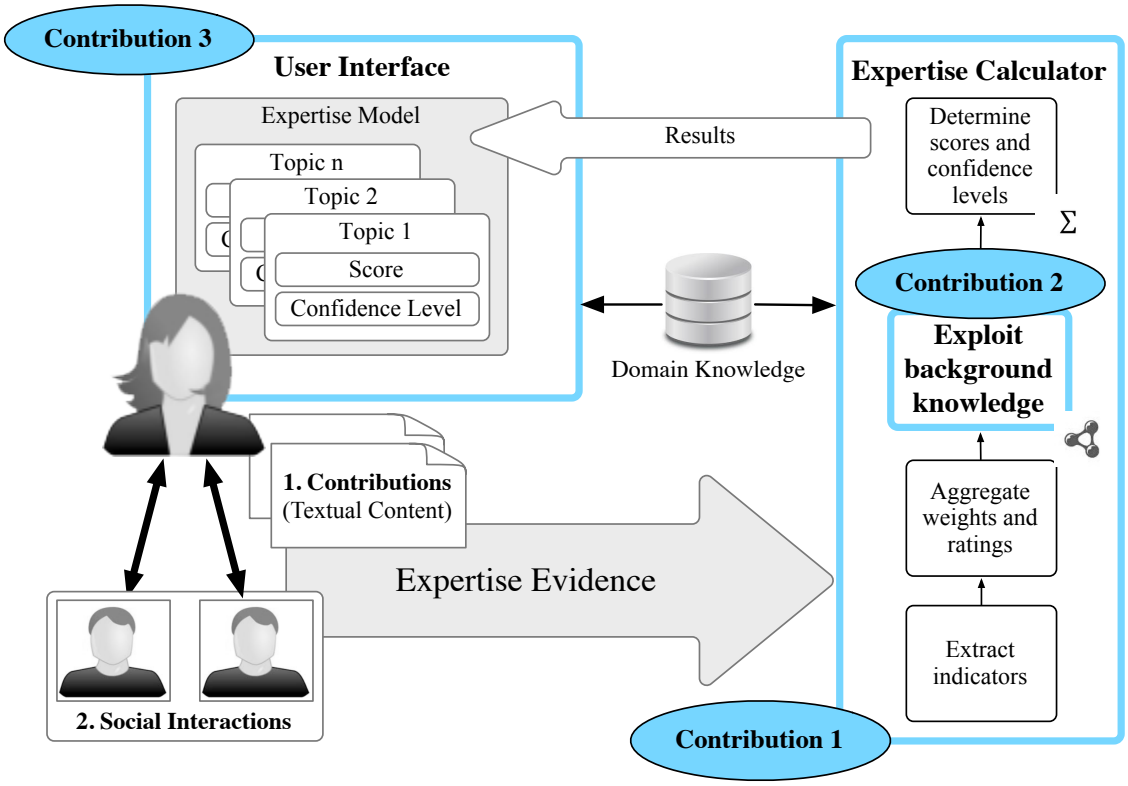


Figure 1.1: Calculating users' expertise based on their contributions and social interactions in online communities.

Toward the goal of measuring users' expertise based on their submissions to an online community, we make three main contributions:

1. We devise and implement a method called Expertise Calculator displayed on the right in Figure 1.1. The Expertise Calculator couples various types of contributions with information obtained from users' interactions in order to calculate users' expertise models. These expertise models are built of expertise topics, absolute expertise scores ranging from 0 to 100 points and values representing the trust in these scores.

2. Expertise topics differ regarding their level of abstraction, i.e., some topics are rather general whereas others have a specific nature. To align the score levels amongst expertise topics, we propose a score propagation algorithm exploiting the structure of a domain ontology. This algorithm is part of the Expertise Calculator, but can also be used in other application contexts.

3. Systems modeling users' expertise, need to open these models for both to gain user acceptance and to collect user feedback in order to improve model quality. We introduce an interface (top left in Figure 1.1) allowing users to scrutinize their expertise models. This is in particular challenging since the more expertise topics are available in the domain, the more difficult it is for users to keep the overview. Furthermore, we enhance this user interface with an expertise prediction feature supporting users in maintaining their models.

A strength of this thesis is certainly its extensive empirical focus. In the course of our research, we iteratively develop the Expertise Calculator. Each version is evaluated with a dedicated experiment adopting students as subjects. The same applies to the proposed score propagation and to the presented user interfaces, all of them are evaluated by conducting separate experiments.

1.3 Methodology

From the methodical point of view, we conduct our research by following mainly the design science paradigm as proposed by [Hevner et al., 2004]. This particular research approach represents a framework comprising IT artifacts, processes focusing on these artifacts and a set of research guidelines. The conceptual framework is based on the notion that within information systems research, IT artifacts are built and evaluated given a relevant problem as shown in Figure 1.2. Thereby, the conducted research is based on existing scientific knowledge and at the same time contributes back to this knowledge base.

As already mentioned in the previous section, we mainly contribute three artifacts throughout this thesis, i.e., the Expertise Calculator, the Score Propagator and the User Interface (including a variant). According to the terminology by [Hevner et al., 2004], all these artifacts correspond to *methods*, where their implementations constitute *instantiations*. We iteratively build,

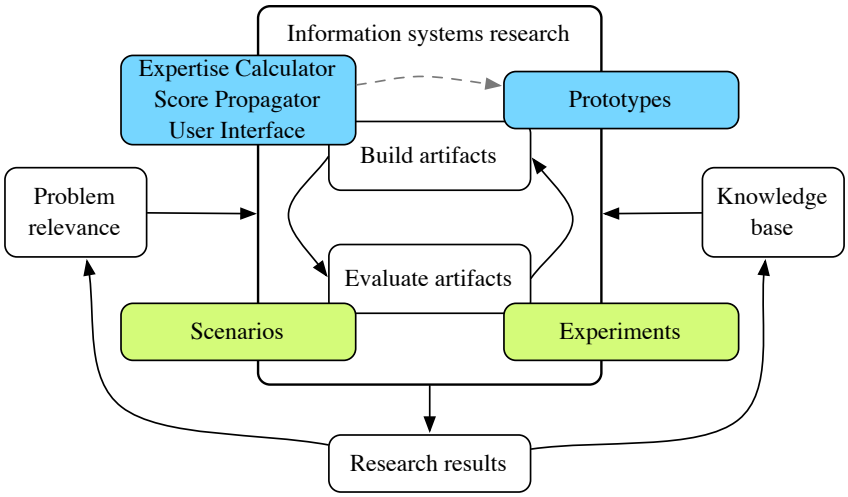


Figure 1.2: Research methodology framework.

implement and evaluate the Expertise Calculator. Given the various versions of the Expertise Calculator, we conduct several controlled experiments for evaluation. All of these experiments take place in a university environment with participants represented by students. We aim to test the Expertise Calculator's attributes in a real environment as well as estimate the validity of its results by means of participants' self-assessments.

The Score Propagator is developed in a similar process. We start with describing its concept and proceed with implementing its prototype. The Score Propagator is going to be evaluated by two independent experiments. In the first experiment, we set up scenarios and display the propagation results based on these scenarios to human experts. The second experiment demonstrates the application of the proposed Score Propagator in another application context where we are able to test its performance based on participants' self-assessments.

Regarding the development of the proposed user interfaces, we follow the behavioral science paradigm. In particular, we expose our interfaces to users and analyze how they respond. In one case, we explore the perceived usefulness of the interface. In the other case, we look at how expertise models evolve regarding certain characteristics when users are supported by expertise predictions during self-assessment.

1.4 Structure of the Thesis

The present thesis addresses the measurement of expertise within an online community. As we mentioned earlier, we focus on online communities where members work together on problem-solving tasks. To evaluate our research, we conduct several experiments with university students. We chose this environment since it guarantees the availability of experimental subjects and because it gives us full control and flexibility over the experimental setting. As a consequence, we will regularly refer to literature in the field of educational information systems and thus talk about learners rather than employees in a company. However, in the scope of our research, it does not matter if we look at students solving problems or if we look at employees doing the same even if the emotional context is partly different.

In the following chapter, we introduce the terminology used throughout this thesis. Terms such as expertise and skills are often understood as representing the same concepts, but in fact, this is not true. We also review the various ways in which expertise is commonly validated. As competence ontologies serve as the background knowledge for the proposed Expertise Calculator, we briefly explain the structure of this special kind of ontology. Since we aim to generate users' expertise models, we need to choose a suitable representation form for them. And finally, we present some background knowledge regarding the notion of online communities and its variants. Furthermore, we survey related research works on expertise modeling. In particular, we review the approaches of different types of systems, e.g., competence management systems and expert finders. We look at the sources of evidence that are used to capture users' expertise and which of them seem more promising than others. We explore the techniques used by existing approaches to extract expertise from digital artifacts. In particular, we are interested in how competence ontologies are utilized in this regard. We further study approaches directly related to our research, i.e., approaches generating expertise models based on information gained from online communities.

In Chapter 3, we outline the fundamental structure of the Expertise Calculator. First, we introduce the knowledge sharing platform being used for experimenting and especially for collecting the data. Given the iterative design procedure, we develop three different versions of the Expertise Calculator in total. The first two versions including their evaluation experiments are covered in this particular chapter. In addition, we introduce a user interface to open the generated expertise models to our participants for scrutiny.

The Expertise Calculator as presented in Chapter 3 applies a simple propagation method to spread expertise scores in users' expertise models. In Chapter 4, we address the shortcomings of this simple propagation approach and devise a sophisticated method exploiting the structure of the background knowledge in a more advanced way. We evaluate the novel method with the help of human experts. To do so, we design scenarios and execute score propagation. The propagation results are then displayed to the experts, who examine score validity by comparing the scores generated by the more sophisticated approach with those of the simple one. In the third version of the Expertise Calculator, we replace the former simple approach with the proposed novel method. This is exactly the setting that we extensively evaluate later on in Chapter 6.

But before conducting the final evaluation, we are interested in how the novel propagation approach performs in another application setting. Thus, we utilize the novel method in Chapter 5 to support users in constructing and maintaining their expertise models by means of expertise predictions. Besides analyzing the accuracy of score predictions by comparing them with participants' self-assessments, we explore how the nature of expertise models as well as the participants' behavior in self-assessment change when offering predictions to users. For testing these issues, we conduct an experiment where we separate the participants into two groups: one group working with predictions and the other group without prediction support.

We evaluate the final version of the Expertise Calculator in Chapter 6. The final version is mainly based on the Expertise Calculator in Chapter 3, except for the method used for propagating scores. In this regard, the simple approach is replaced by the novel method introduced in Chapter 4. During the evaluation we mainly measure the Expertise Calculator's score accuracy and the validity of calculated confidence levels. We explore which attributes of the Expertise Calculator contribute best to calculating valid expertise scores. In addition, we analyze participants' feedback across all experiments.

Chapter 7 concludes this thesis by revisiting the initial research questions and answering them based on our contributions and results. In addition, we discuss the limitations of our research and raise some issues for future work.

1.5 Grounding Material

The content of this thesis is based on a number of publications. Please refer to the Bibliography to obtain full details about the listed publications.

Parts of Chapter 3 build on work presented in:

- [Dorn and Hochmeister, 2009]: *TechScreen: Mining Competencies in Social Software*, KGCM2009.
- [Hochmeister, 2011]: *Mining User Knowledge in Learning Networks*, BIR2011.
- [Hochmeister and Daxböck, 2011]: *A User Interface for Semantic Competence Profiles*, UMAP2011.
- [Hochmeister, 2012a]: *Calculate Learners' Competence Scores and Their Reliability in Learning Networks*, BIR2011.

Some parts of the content covered in Chapter 4 were published in:

- [Hochmeister, 2012b]: *Spreading Expertise Scores in Overlay Learner Models*, CSEDU2012.

Parts of the material presented in Chapter 5 were published as:

- [Hochmeister et al., 2012]: *Using Expertise Predictions to Facilitate Self-regulated Learning*, ITS2012.

CHAPTER 2

Related Work

This chapter introduces the terminology used throughout this thesis and reviews related research works regarding expertise modeling. We address in particular our understanding of expertise, the commonly used ways to assess expertise and how expertise is modeled in current research works. While surveying literature we realized that users' expertise is primarily determined by information systems focussing on finding experts, supporting learners and managing competences in organizations. Thus, we particularly analyzed existing approaches in these certain fields. After a brief description of these systems, we review the most common sources of evidence by which systems infer users' expertise. Then, we explore systems using ontologies for expertise profiling as well as systems extracting expertise in an online community environment. For all approaches being reviewed, we were especially interested in how they represent expertise levels that is either on a qualitative or quantitative scale.

2.1 The 'Fuzzy' Notion of Competence

A controversial debate is running in both the research community and the professional field about how to precisely define an individual's ability in accomplishing certain tasks in real-world situations [Weinert, 2001] [Le Deist and Winterton, 2005]. This is also true for individuals' theoretical knowledge about concepts in which they have, if any, rather limited experience. The diversity in interpretations regarding terms like skills, competences, expertise, qualifications and knowledge is a rather broad one. Therefore, we aim to briefly review some interpretations of these concepts in literature and deduce a working definition of expertise serving as a foundation for this thesis.

2.1.1 Towards a Definition of Competence

[Burke, 1989] delineates the competence concept as "being able to perform" work roles, rather than just having specific skills or knowledge. Performance being shown is measured against standards expected in employment "with all the associated pressures and variations of real work."

Table 2.1: Explicit vs. tacit knowledge modified after [Ellstrom, 1997] and [Smith, 2001]

	Explicit (know-what)	**Tacit (know-how)**
Knowledge base	Theoretical/academic	Practical/experience-based
Situation	Well-defined	Ill-defined/complex
Information for action	Certain Emotionally neutral	Uncertain Emotionally colored
Mode of action	Problem-solving-in-thought	Problem-solving-in-action
Information processing	Analytical	Intuitive
Mode of learning	Formal education/instructions	Informal learning in everyday practice Situated learning

[Ellstrom, 1997] explores the difference between competence, qualification and skill in the professional context. Basically, he defines competence as individuals' capacity to successfully handle certain situations and accomplish certain tasks respectively. Following this definition, the term occupational competence refers to the relation between individuals' capacity and certain task requirements. This capacity reflects a complex function comprising, amongst others, different types of knowledge, personality traits as well as social skills.

On the other hand, the notion of qualification is a much more restricted and evident one. It describes competences that are actually required by the working task and prescribed by the employer. Following the distinction between occupational competence and qualification, individuals may possess competences that are not qualifications as they may not be prescribed in a work's task description.

Viewing competence as an individual attribute workers bring into their job, we can distinguish between formal competences (like years of schooling completed) and actual competences including learning experience and informal, everyday activities at the working place [Ellstrom, 1997]. Thus, one can not use formal competences as a base to infer actual competence. This would simply ignore qualitative differences amongst educational institutions.

Another perspective on the competence concept considers on the one hand the theoretical, explicit aspect of knowledge and on the other hand the experienced-based, tacit aspect [Polanyi, 1966] [Smith, 2001] [Stenmark, 2000] [Ellstrom, 1997]. Table 2.1 shows the main characteristics by which explicit and tacit knowledge are distinguished.

Tacit knowledge can be further divided into cognitive and technical tacit knowledge [Smith, 2001] [Alavi and Leidner, 2001] where the former is understood as the individuals' mental models, beliefs and perceptions. Technical tacit knowledge, however, represents the know-how applied to a specific task. While working on a task, people know something so well that they are mostly unaware what finally contributed to successful task completion. For instance, programmers building new software are not aware of the techniques they apply in solving problems that occur while working on their development tasks.

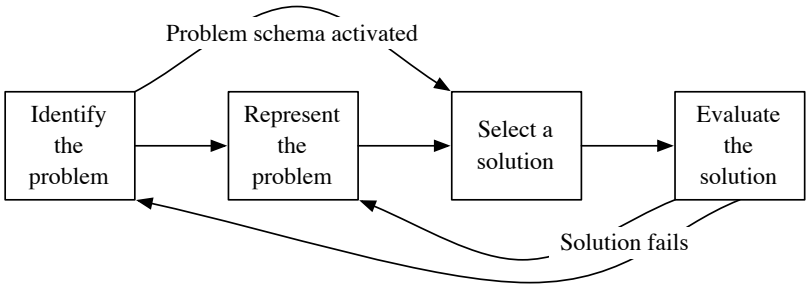


Figure 2.1: Example problem solving process after [Schraw et al., 2006].

In the Oxford English Dictionary [1] *expertise* described as the "skill or expertness in a particular branch of study". The concept of an *expert* is explained by someone who gained skills from experience. In psychological science, an individuals' expertise is defined as "the possession of a large body of knowledge and procedural skill" [Chi et al., 1982]. A prominent area of research in cognitive psychology is problem solving. Researchers in this area mainly differ experts from novices. Basically, problems fall into two types, i.e., classroom problems and real-world problems [Chi and Glaser, 1985]. Real-world problems, we encounter in our everyday experience, are often the most important and difficult problems we seek to solve other than classroom problems. One of the key characteristics of real-world problems is their ill-defined nature, i.e., several aspects of the problem are not well-defined, confer Table 2.1. Therefore, it is highly uncertain which specific actions we have to take for reaching a solution. In such a case, problem solvers have to add information to the problem situation, which largely depends on their domain knowledge and experience, confer Figure 2.1.

More recently, [Le Deist and Winterton, 2005] reviewed the understanding of competence across various countries including the USA, UK, France, Germany and Austria. Their results show that even within countries there is an apparent difference in approaching competence, not to speak of the differences amongst countries. However, they can recognize a trend where one-dimensional frameworks of competence give way to multi-dimensional frameworks. Therefore, [Le Deist and Winterton, 2005] propose a holistic typology of competences. Basically, their approach is centered around a key competence referred to as *meta-competence* facilitating the acquisition of other substantive competences including cognitive, functional and social competences.

2.1.2 A Working Definition of Expertise

An expert is widely understood as an individual with outstanding *expertise* in a certain field, which is largely based on experience. Throughout this thesis, we use the technical term *expertise* referring to user knowledge that is applied in the context of solving a real-world problem. Besides, this is the term commonly used in related literature we will review later in this chapter. We also agreed to use *expertise* since it inherently suggests concepts like experience and difficulty and thus might prevent readers' confusion with other competence concepts.

[1] http://www.oed.com

The potential of explicit knowledge as a source for extensive expertise modeling seems rather limited. As shown in Table 2.1, explicit knowledge is based on well-defined tasks and more importantly has theoretical nature. Therefore, we assume that technical tacit knowledge provides a better source to address our ultimate goal to improve the qualities of users' expertise models.

Furthermore, a major objective of our research is to measure expertise levels on an absolute scale. The importance of grading expertise is reflected by a number of existing research works. For instance, [Cheetham and Chivers, 2005] refers to competence as the effective performance within a domain (context) at different levels of proficiency. According to [De Coi et al., 2007], a competence consists of three dimensions including competency (meaning skill), context and proficiency level. Please note that at some points in this thesis we make still use of the term *competence* in order to precisely refer to related works. However, we always follow the notion of users' knowledge that is applied to a more or less complex, real-world situation far away from being largely focussed on theoretical issues.

2.2 Ground Truth for Expertise Evaluation

In this section, we explore various ways to evaluate the validity of expertise statements originating either from individuals themselves or from a system in the form of predictions about individuals' expertise. In the following, we review several approaches for expertise validation including self-assessment, peer-assessment, expert-assessment and multi-source assessments.

2.2.1 Self-assessment

Self-assessment is an intrinsically difficult task. Even though considerable research suggests that learners are able to accurately describe their expertise [Blanche and Merino, 1989], errors in self-assessment occur due to various reasons.

According to [Boud and Falchikov, 1989], self-assessment is defined as "the involvement of learners in making judgements about their own learning, particularly about their achievements and the outcomes of their learning." Several psychological mechanism contribute to faulty self-assessment [Dunning et al., 2004]. They propose to sort these mechanisms into two classes. First, erroneous self-assessments occur because people rarely have all the information necessary to make profound assessments and secondly they often overlook what they do not know. In addition to the latter aspect, people neglect to incorporate relevant information they do have in hand.

The complexity of self-assessment does even increase when moving from rather well-defined to ill-defined expertise concepts. For instance, it is rather easy to define top expertise in math performance, a very well-defined domain. In math, specific right answers are available in advance and techniques to obtain the solutions are clearly defined. This is rather different in ill-defined domains. In these domains, numerous skills themselves are ill-defined in that many different criteria can be argued to be relevant for them. People tend to overestimate on skills that are ill-defined, but not on skills with a rather clear outlined definition. Based on a skill definition that is more constrained, students fail to rate themselves too positively and their ratings are somehow similar to those of others [Dunning et al., 2004]. For instance, students' self-assessed grades

were slightly more related to their teachers' evaluations when the exam's subject matter is more well-defined [Falchikov and Boud, 1989]. Furthermore, students' and teachers' grades tend to correspond more in advanced classes than in introductory courses, i.e., students in higher-level classes better predict their performance than students in lower-level classes [Boud and Falchikov, 1989]. More recently, this phenomenon was acknowledged in a real world setting. Even when poor performers were given incentives for particularly cautious assessment, the accuracy did not improve [Ehrlinger et al., 2008]. However, although more experienced students achieve a higher agreement with their teachers, their self-assessments are still far from being perfect. The accuracy of students' self-assessment improves over time and is further enhanced when teachers give students feedback on their self-assessments [Dochy et al., 1999].

[MacIntyre et al., 1997] examine students' perceived competence in second languages. They found that anxious students who have little faith in their capacities tend to underestimate their competences whereas less anxious, self-confident students are prone to overestimate themselves. However, their study results reveal that deviations from actual performance (judged by experts) show a clear tendency for both groups of students implicating a systematic bias in this regard.

2.2.2 Peer-Assessment

Peer-assessment is defined as "the process through which groups of individuals rate their peers" [Falchikov, 1995]. [Dochy et al., 1999] defines a combined notion of self- and peer-assessment where students assess peers "but the self is also included as a member of the group and must be assessed." In the following we use the latter definition.

Data on peer-assessment indicate that peers provide more accurate assessments about their fellows' abilities than their fellows' own estimates [Topping, 1998]. More specifically, evaluations by peers highly correlate with those of teachers where grades coming from peers tend to be lower than those from teachers [Falchikov and Goldfinch, 2000]. Studies mainly argue the increased value of peer-assessment with the fact that individuals can identify good and bad performances, but are unable or not willing to apply the same standards to their own performance [Ward et al., 2002]. Besides, peer-assessment is not without shortcomings. For instance, it can raise anxiety [Topping, 1998] and similar to the earlier mentioned aspect, poor students are not able to provide such accurate assessments as the more skilled students do [Dochy et al., 1999]. However, peer-assessments become more valid when based on a larger number of evidence and on a broader scope of skills. The more well-defined the matter and the more peers are involved in the assessment procedure, the more reliable the assessment [Dunning et al., 2004].

2.2.3 Measuring the Quality of Self-assessment

[Ward et al., 2002] review existing approaches measuring the quality of self-assessments and examines methodological issues impeding this measurement. The most common approach involves correlation analysis. Herein, a self-assessed score and a score usually based on experts' estimates are generated for each individual in the group. Self-assessments are correlated with expert ratings based on the entire score pairs in the group and result in a single correlation value. This correlation value finally represents the quality of the group's self-assessment. Another

methodological approach involves the comparison of self-assessed scores with an external standard. Similar to the correlation analysis, this approach performs a comparison considering the self-assessments in the group as a whole with the external standard based on average means. In the following we refer to three methodological issues as presented by [Ward et al., 2002] that plague either of these approaches.

First, both of the approaches assume that expert estimates represent the golden standard by which to measure all aspects of competences. However, only a few studies examined the reliability of the golden standard and they suggest inconsistency among expert assessors. Thus, the unshakeable notion of the golden standard that is grounded on expert evaluations must be handled carefully while interpreting score correlations. Furthermore, experts have to agree on a valid measure of the aspects they are asked to evaluate to ensure that they measure what has to be measured. The more ill-defined the aspect, the harder it is to find valid measures. One way to tackle this issue is that experimenters should attempt to achieve a high rater reliability by means of multiple expert raters.

Secondly, the correlation approach performs comparison across all pairs of self- and expert estimates in the given group. It seems improbable that all group members share the same understanding of the dimensions of performance. Assuming the rating scale has been optimized regarding its reliability, even an highly elaborated scale remains subject to individual interpretation. To cope with the problem of using scales inconsistently experimenters may provide explicit anchors for evaluation criteria, e.g., by introducing benchmarks of performance. For instance, a benchmark describes the performance of a top expert in java programming. However, finding such benchmarks for ill-defined expertise descriptions still remains a challenge.

Lastly, even if experts provide reliable evaluations and self-assessments are based on the same interpretation of scales, the correlation calculated on group-level remains problematic. It assumes that every individual in the group is equally able in self-assessing their performance. A low correlation suggests that the whole group can not self-assess effectively and vice versa. In this sense, the correlation measure is vulnerable to yet a few outliers that may spoil correlation results.

2.2.4 360-degree Assessment

Multi-source feedback aggregates the previously mentioned techniques into one measure. The most frequent used method in this regard is the 360-degree feedback. It constitutes a quantitatively, competence-based survey that is filled in by the full range of working relationships of the ratee including subordinates, peers and bosses [Toegel and Conger, 2003]. It seems obvious that people working with the ratee are generally able to provide a more comprehensive picture of the ratee's behavior and performance than the ratee's supervisors by themselves. This is in particular crucial when supervisors do not have the opportunity to inspect all areas of the ratee's performance. However, the 360-degree assessment is not without shortcomings. Given the extensive amount of people that might be involved in the rating process, the feedback tends to be costly to implement, complex to manage and time-consuming.

Today, organizations attempt to measure nearly everything. Thus, while originally used only for employees' personal development purposes, the 360-degree feedback is nowadays increasingly included in strategies to measure the performance of employees as well [Maylett,

2009]. Such performance appraisals can have considerable effects on employees as they constitute input to administrative decisions, e.g., to determine compensation. One has to consider that the purpose of 360-degree assessment causes different motivational responses from participants. [Maylett, 2009] reports that when employees know that the feedback they receive will be used solely for their personal developmental benefit, they tend to be more receptive regarding the provided feedback. In contrast, once feedback is determined to trigger administrative consequences, e.g., possible layoffs, employees may perceive feedback more likely as a threat rather than accept it. As a consequence, raters may less likely provide frank feedback when they know that it may affect others' situation negatively.

2.2.5 Summary

Any of the aforementioned approaches for expertise validation has its pros and cons. When carefully considering the context and the settings of the study while choosing the validation method, each of them can contribute to meaningful research. During the review of related work for this thesis, we found that a considerable amount of research works rely on self-assessments to validate predicted expertise levels. For instance, [Vivacqua and Lieberman, 2000] present a system that calculates individuals' expertise levels in a programmer community for the purpose of expert finding. They determine the accuracy of measured expertise levels by comparing them with the self-assessments of users being modeled. The deviation of levels is expressed in percentage rates calculated across the whole group. [Wasko and Faraj, 2005] ask for users self-assessment to explore the correlation between users' expertise and the amount of users' contributions in an online community. [Balog et al., 2007] propose methods aimed at finding expertise relations between topics in documents and people. To evaluate their results they rely on people's self-assessment selecting topics for their profile. [McLure Wasko and Faraj, 2000] derive users' self-assessments from open-ended comments in order to examine a possible correlation between expertise and the willingness to participate in online communities and why users help others anyway.

In the course of this work, we make use of both user self-assessment and expert-assessment to validate predicted expertise scores. Apart from validation, we adopt the notion of peer-assessment as part of the proposed expertise measure.

2.3 Ontologies

Ontologies have achieved an important role with respect to the advancement of established information systems, of systems for data and knowledge management or of systems for collaboration and information sharing [Staab and Studer, 2009]. In this section, we briefly review the fundamentals of ontologies. We especially focus on issues regarding their structural forms. After that, we take a look at ontologies from literature used to represent individuals' competences.

2.3.1 Ontology Fundamentals

Various authors provide their definitions on ontologies, however, all of them share to some extent the same attributes that characterize an ontology. [Neches et al., 1991] presents an ontology

as defining the "basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary." [Gruber et al., 1993] provides a definition that views an ontology as "an explicit specification of a conceptualization". Thereby, a conceptualization "is an abstract, simplified view of the world that we wish to represent for some purpose." Most importantly, [Gruber et al., 1993] understands the knowledge being modeled as *shared knowledge*. *Shared* in the sense that ontologies are intended to be portable between information systems. [Swartout et al., 1996] refers explicitly to the type of structure an ontology is built on, namely, "an ontology is a hierarchically structured set of terms for describing a domain that can be used as a skeletal foundation for a knowledge base." [Guarino, 1998] suggests a more advanced view regarding the representation of an ontology, i.e., a "set of logical axioms designed to account for the intended meaning of a vocabulary."

Ontologies simply consisting of concepts and only one type of relation are referred to as *lightweight ontologies* [Uschold and Gruninger, 2004]. Concepts in such ontologies are mostly organized in taxonomies and do not include any logical axioms. On the other hand, so called *heavyweight ontologies* are semantically rich representations with formal axiomatizations. However, it is hard to say whether simple representations are necessarily of less value than the more advanced ones or vice versa. It mainly depends on the field of application that sometimes requires low computational cost and sometimes powerful reasoning capabilities.

Taxonomies, as a kind of lightweight ontologies, are hierarchical structures for categorizing classes of things in real world. Things are represented by nodes, which are related with an *is-a* relationship. The meaning of this particular type of relationship is manifold and often depends on the application context. Hence, to understand the proper meaning of a relation, one has to examine what is on either ends of the relation. [Brachman, 1983] investigates the various uses of *is-a* relations. One specific kind of interpretation is referred to as *conceptual containment*. In this case, the intent of the *is-a* relation is to express that one description includes another. [Brachman, 1983] provides an example with the general node *king* and the node *king of France*. Thereby, the general description is used to build the other node's description.

A grading between the two extremes, i.e., lightweight and heavyweight ontologies, is proposed by [Lassila and McGuinness, 2001]. The simplest notion of an ontology is a controlled vocabulary representing a finite list of terms, for example, a catalog. The next possible type for defining an ontology is a glossary, i.e., a list of terms including their meanings. Thesauri introduce semantics to the relations between terms. Typically, they do not provide an explicit hierarchy, however, based on narrower and broader term specifications a hierarchy can be constructed anyway. The next two types of ontologies are characterized by their explicit hierarchical structure utilizing is-a relationships, where the latter specifies this relation in a strictly formal way. The remaining types of ontologies include the more formal logical constructs the closer they are located towards the end of the line in Figure 2.2.

Another scheme for ontology classification considers, amongst others, general ontologies, domain ontologies and application ontologies [Gomez-Perez et al., 2004]. They mainly differ regarding their possible reusability. Thus, general ontologies are reusable across several domains, domain ontologies are reusable within the domain they are built for (e.g., the enterprise ontology by [Uschold et al., 1998]), whereas application ontologies work only in the specific context of an application.

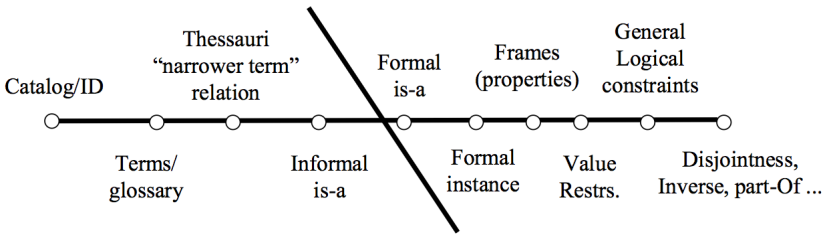


Figure 2.2: Types of ontologies.

Various methodological approach exist for building ontologies, yet it seems there is no completely mature proposal for building ontologies out so far [Fernández-López and Gómez-Pérez, 2002]. However, a widely used and comprehensive methodology for developing ontologies is presented by [Uschold and Gruninger, 1996]. When building an ontology they first recommend to define its future purpose and its scope on the modeled domain. The next step concerns the capture of concepts describing the domain as well as the relations linking these concepts. Once everybody agreed on the collected concepts, they are made explicit using a representation language. During the building process, there is the question on how to consider or even integrate existing ontologies, which is in general a very difficult problem. On the one hand, it is relatively easy to define synonyms for existing concepts or add new concepts where no similar concepts readily exist. On the other hand, though, once there are obviously similar concepts available, it is hard to decide how and whether such concepts will be integrated anyway. The ontology being built is evaluated against so called *competency questions* in order to test if the ontology can actually give the particular answers it was originally built for. Finally, the assumptions taken while setting up the ontology need to be documented for later revision or reuse.

Interestingly, a broadly based review on interfaces regarding the visualization of ontologies reveals that these interfaces mostly focus on hierarchies, implying that this is currently a widely used ontology structure in various application domains [Katifori et al., 2007].

2.3.2 Competence Ontologies

Once a user successfully demonstrated a certain competence in a real-world situation, we say the user has expertise in the given subject matter. When defining an ontology supporting knowledge management systems to measure expertise, we determine the competences describing the respective domain as well as the relations amongst them. However, these competences are not inherently related to expertise. In fact, this relation is established once users are associated with competences in the ontology based on their actual performances. Thus, we speak of *competence* ontologies rather than an ontology holding expertise topics per se.

[Schmidt and Braun, 2008] distinguish three levels of formality regarding the modeling of expertise. The accuracy to which users' expertise can be described depends strongly on the these formality levels. The first and most simple variant of formality represents a flat list of topics regarding a certain subject matter. The second level considers taxonomic relationships allowing different levels of abstraction. Lastly, the third level constitutes the most accurate form

to represent expertise. It extends the hierarchy from the previous level in so far that it introduces different degrees of expertise fulfillment, i.e., the level of expertise given a certain topic. It seems quite obvious that information systems can provide more sophisticated services, the more fine-grained information about users is available.

We surveyed competence ontologies used by existing applications with varying purposes. We realize that competences are mainly structured in hierarchies or at least are based on hierarchical structures [Liao et al., 1999] [Mohamed et al., 2006] [Biesalski and Abecker, 2005] [Tarasov et al., 2007] [Pernici et al., 2006] [Colucci et al., 2007] [De Coi et al., 2007] [de Vasconcelos et al., 2009]. Regarding competences in the field of computer sciences, the ACM (Association for Computing Machinery) together with the IEEE Computer Society provide guidance on developing respective curricula at approximately ten-year intervals. In their most recent published guidelines [ACM, 2008], they suggest a body of knowledge comprising 14 knowledge areas such as Programming Fundamentals, Information Management and Operating Systems. Knowledge areas are described by means of more specific topics and each of these topics is further described by even more specific topics and so on. Eventually, this leads also to a hierarchy of learning goals and different kinds of computer science expertise respectively.

In general, the proper size of an ontology depends on its purpose. As mentioned in the previous section, whether an ontology meets the application's requirements can be evaluated, for instance, by means of competency questions. Thus, as long as these competency questions can be answered, the given ontology obviously holds a sufficient amount of concepts (and relations). In terms of a competence ontology, it seems somehow clear that the more concepts are defined the more accurate and fine-grained users' expertise can be described. However, this implies huge efforts in building such an almost "perfect" ontology and its usability will suffer equally. On the other hand, an ontology representing a given domain on a trivial level is perhaps easy to handle and built quickly, but on the downside it may not provide enough concepts to gain a meaningful statement describing one's expertise.

2.3.3 Spreading Activation

Spreading activation is a technique to process networked data such as an ontology. It was first introduced in the field of psychology [Anderson, 1983]. Computer sciences adopted spreading activation in various areas, for instance, in information retrieval [Crestani, 1997]. Basically, spreading activation activates topics in an ontology and passes the level of these topics to adjacent topics as shown in Equation 2.1, where the level depends also on the link connecting two topics.

$$I_j = \sum_i O_i \cdot \omega_{ij} \ . \tag{2.1}$$

where I_j represents the activation level of topic j received from topic i depending on the relation weight ω_{ij}. Various approaches exist to determine relation weights [Pirrò, 2009]. However, one simple way to configure relation weights is the use of a decay factor, which consistently attenuates the activation level during spreading activation [Liu and Maes, 2005] [Cantador et al., 2008].

Spreading continues until all topics in the network are activated. In fact, this is the main drawback of pure spreading activation. Introducing rules adjusting spreading activation helps to

gain control of this undesired behavior. Constrained spreading activation considers such rules (constraints) that limit the number of activations in the network. These rules include distance constraints, fan-out constraints, path constraints and activation constraints [Crestani, 1997].

One of the most cited and pioneering systems using spreading activation is GRANT [Cohen and Kjeldsen, 1987]. This system facilitates the search for funding bodies based on research proposals. For that reason, GRANT relies on an ontology representing research topics. Research proposals as well as funding agencies are associated with ontology topics. The system starts searching by activating topics obtained from research proposals and spreads activation through the ontology until funding agencies, linked to the ontology's topics, are activated as well. Thereby, activation is restricted to prevent activation of possibly irrelevant funding bodies.

[Crestani and Lee, 2000] retrieve information from the web by means of spreading activation. Their web search system offers users an autonomously navigation through web pages based on hyperlinks connecting these pages. The relevance of a page in this navigation process is computed by spreading activation. Web pages linked to a page the user showed interest in, will only be considered for navigation if they comply with certain constraints.

[Liu et al., 2005] adopt spreading activation for the purpose of ontology extension. They first augment a seed ontology with terms obtained from a collection of news media sites. The relation weights are set depending on the type of relation between terms found in the web documents. Finally, spreading activation yields the most promising terms, which are then suggested to experts as candidates for ontology extension.

[Sieg et al., 2007] utilize spreading activation to propagate interests in a hierarchically structured user model. They determine relation weights by a measure of containment. Ontology topics are associated with documents. The more equal the document term vectors of topics, the higher the relation weight. A similar approach using a hierarchy is proposed by [Schickel-Zuber and Faltings, 2007]. The amount of scores propagated to a parent topic depends on the features shared by the parent and the descendants in its subtree.

[Hussein and Ziegler, 2008] learn user interest models for building context-adaptive web applications using spreading activation. Both domain knowledge and context factors are represented by means of single ontologies. The aggregation of these ontologies allows the inference of user interests in a given context. The context, e.g. location, is captured and associated with a topic in the context ontology. Context topics activated in this manner, spread their activation levels through the ontology network and thus activate topics from the domain ontology. This activation process is restricted by the number of activated nodes as well as the number of processed nodes. While users browse a website, the system adjusts the relation weights based on users' feedback about recommended content.

[Kay and Lum, 2005b] apply spreading activation to propagate a user's expertise scores in an overlay user model. They define the relation weight of a parent topic as the reciprocal value of the total amount of its children. To our knowledge, this is the only directly related work to our approach as it is related to a similar context, thus we decided to use it for the second version of our Expertise Calculator.

2.4 Modeling Users

In this section, we review the features by which a user is commonly modeled. In this regard, we especially focus on users' knowledge.

2.4.1 User Expertise

[Brusilovsky and Millán, 2007] provides a list of the most popular features that are commonly modeled in the field of adaptive web systems. These include users' knowledge, their interests, goals and tasks, background, and individual traits. The authors also mention the individuals' context of work as a relatively new feature drawing the attention of researchers.

Amongst the aforementioned user features, knowledge appears to be the most important one. Users' knowledge is changing over time, i.e., it can either increase or decrease. Adaptive systems have to consider this particular development and need to make sure to keep information about users' knowledge up to date. The simplest form to represent domain knowledge is the scalar model that estimates domain knowledge by means of a single value on either a quantitative or qualitative scale. Knowledge is typically provided by users themselves or by objective testing, if applicable. However, scalar models have a major shortcoming that is its low precision. This is because the scalar model averages the user knowledge of a certain domain rather than also describing specific parts of the domain. This problem is solved by so called structural models, such as the overlay model.

Overlay modeling has its roots in the design of a tutoring system. [Carr and Goldstein, 1977] introduce a model holding learners' skills compared to an expert standard model. They propose a tutoring system utilizing a set of hypotheses, called overlay, to estimate the confidence that learners possess certain skills. A unique overlay is assigned to each learner, i.e., the learner's model. Based on these overlays the system is able to adapt explanations to learners' knowledge levels and thus allow efficient learning. Basically, [Carr and Goldstein, 1977] understand overlays as *a perturbation on the expert's structure*. Hence, an overlay holding a subset of the expert standard model represents a simplification in that it does not consider learners' incorrect or even alternative skills. Despite of this limitation, [Carr and Goldstein, 1977] argue the models usefulness with the fact that a human tutor preparing explanations is not fully aware of a learners skill portfolio either.

The basic idea of overlays was transferred to ontology-based user models. In this type of models, learners' expertise is modeled as a subset of topics from a domain ontology representing the expert standard. The underlying network structure of the domain ontology allows for reasoning over the topics in learners' models. Today, this kind of user models constitutes the dominant representation of users in adaptive educational systems.

For example, [De Bra et al., 2003] propose an architecture for an adaptive hypermedia system based on overlay user models. Their work originates from the idea to support an online course with additional user guidance by refining explanations and methods of link hiding. Each web page is associated with some of the domain topics. In order to improve adaption of web pages, they exploit topic links to propagate a user's knowledge (triggered through a web page visit) to other topics in the ontology. This propagation mechanism generates new knowledge to learner models and helps refining the adaption process.

2.4.2 User Modeling Approaches

In the following, we briefly review the most common approach that is used in today's systems for user modeling. However, when modeling user knowledge, we often deal with information that is uncertain. Thus, we shortly present approaches dealing with that particular issue as well.

Feature-based Modeling

The currently dominant approach to user modeling is the feature-based approach [Brusilovsky and Millán, 2007]. Feature-based models describe users by means of their features, for instance, their knowledge. As we mentioned earlier, features use to change over time, thus the system has to make sure to adapt users' models appropriately. Another modeling approach associate users with stereotypes [Rich, 1979]. The system treats users associated to a certain stereotype in the same way. A stereotype contains a mixture of features, however, these features are ignored in modeling, instead the stereotype is used as a whole. Although stereotype user modeling has been proposed over three decades ago, it is still of importance when combining it with the feature-based approach. To tackle the problem of new users in the system, users' feature-based models are initialized with the features given by a particular stereotype.

Uncertainty-based Modeling

When capturing knowledge about a user, there always remains some extent of uncertainty and inaccuracy. For instance, if a learner fails to answer a question, we *most-likely* know that this learner does not possess the necessary competence. Similarly, in case a user was engaged in learning a concept for *a rather a long* time, we have to deal with inaccuracy. Numeric uncertainty measures tackle these kinds of issues. [Jameson, 1995] reviews three approaches to uncertainty management in user modeling, i.e., Bayesian networks (BN) [Pearl, 1988], the Dempster-Shafer theory of evidence [Shafer, 1976] and fuzzy logic [Zadeh, 1994]. In the following, we review the idea to use Bayesian networks and fuzzy logic since these two represent the most commonly used techniques for uncertainty-based user modeling, even though, only a few studies report the use of these approaches [Brusilovsky and Millán, 2007].

Bayesian networks are probabilistic models providing a network model comprising nodes (possibly multi-valued) and relations linking these nodes. In particular, these links represent the probabilistic relationship between a pair of nodes. Let us consider these concepts by means of an example. Assuming we have two competences C_1 and C_2 (represented by nodes in the BN) that are associated with a link. In terms of C_1, we have evidence for a certain user that suggests a certain expertise level for this competence. By means of the probability figure that describes the relationship between the two competences, we can now estimate the user's probable expertise in C_2 (for example, a probability of 0.2 corresponds to beginners where a probability of 0.6 to intermediates). The construction of such a network model consists of two steps: First, we define the nodes and links of the network (the qualitative model) and secondly, we need to determine the link probabilities. Basically, these conditional probabilities can be either obtained from domain experts' estimates or (semi-) automatically learned from empirical data. It seems obvious that large and reliable Bayesian networks are hard to create, which is actually their main disadvantage. Thus, the cost of creating almost complete models needs to be carefully balanced with

models' usability and the usefulness in terms of the particular task. [Zapata-Rivera and Greer, 2004] propose an interface that helps students and teachers to engage in a negotiated assessment process. Negotiation happens by means of a Bayesian student model representing learning topics associated with the students' level of knowledge. Both students and teachers give their estimates about the probabilistic relationships linking the various learning topics. The system finally aggregates these estimates and thus determine its beliefs about students' knowledge.

Fuzzy logic. Consider this statement: "Jane is rather advanced, so she is most-likely be able to accomplish this task quite well." We often use vague concepts in our reasoning. Fuzzy logic techniques facilitate to mimic this human style of reasoning. Thus, it is especially easy for users to understand and maintain the reasoning of systems adopting a fuzzy logic approach. Fuzzy logic includes concepts like *linguistic variables*, *fuzzy sets* and *fuzzy if-then rules*. [Chin, 1989] provides an example for a fuzzy treatment. That is, a linguistic variable may represent likelihoods by means of 6 discrete values, e.g., "somewhat likely", "likely", "very likely". Assuming that expertise is represented on four levels, i.e., novice - beginner - intermediate - top, and a knowledge concept has two difficult levels: simple and complex. Given these attributes, we can set up fuzzy logic rules like:

> If Jane is a *beginner* and the concept C is *simple*,
> then it is *likely* that Jane knows C.

These rules are similar to the probabilities used in Bayesian networks, but they explicitly state the uncertainty of the system. Thus, for designers coding uncertainty, the fuzzy logic approach might be more intuitive than determining conditional probabilities for links in Bayesian networks. The system takes care about expertise changes based on observations by using another fuzzy logic rule:

> If the concept C is *simple* and Jane knows C,
> then it seems *more likely* that Jane is an expert in C.

The question of where the numbers come from seems to be the most crucial one when thinking about the adoption of uncertainty-based modeling [Jameson, 1995]. This is especially true for Bayesian networks, where usually experts need to determine numerical probabilities. As for fuzzy logic approaches, determining qualitative labels for certain variables and developing reasoning rules similar to human reasoning is one side of the coin. In the end, even this linguistic variables need to be mapped to numbers for internal representation. And exactly this mapping constitutes the other side of the coin since it also demands human experts.

2.5 Online Communities

Due to the prevalence of the internet and corporate intranets people increasingly share knowledge by means of digital artifacts. Platforms, where people meet online to discuss various topics, are called online communities. According to [Plant, 2004], an online community is "a collective group of entities, individuals or organizations that come together either temporarily or permanently through an electronic medium to interact in a common problem or interest space". Topics

in such communities include issues like professions, interests or products. Online communities have emerged as a major platform for people to seek and share knowledge [Zhang et al., 2007]. The shared knowledge represents substantial evidence with respect to the authors' expertise.

Web-based communities are rather social and dynamic and have different forms, e.g., online chat forums, blogs, problem-solving communities and social networks. We are particularly interested in problem-solving communities since we expect that users' contributions to this type of community comprehensively reflect users' expertise. In recent years, so called Question and Answer (Q&A) websites became very popular not only for help-seeking people and eager experts but also for researchers working on various aspects of Q&A, see [Rodrigues et al., 2008], [Sun et al., 2009], [Blooma et al., 2010], [Pal and Konstan, 2010].People use this kind of websites to exchange knowledge given certain knowledge categories. More specifically, some users post questions related to a category where others provide answers to posted questions. [Harper et al., 2008] identify three types of Q&A sites including *Digital Reference Services* (traditional library reference services where expert researchers help people to find useful information), *Ask an Expert Services* (experts in topic categories provide answers, less structured and formal procedure than in digital reference services) and *Community Q&A Sites* (leverage the time and effort of everyday users, little structural or role-based organizations, include newer features to facilitate user interactions such as tagging and rating). While some Q&A services are free to use, commercial Q&A websites have emerged lately where askers submit their questions and experts compile and sell their answers to the askers. [Harper et al., 2008] found that the quality of answers is higher in fee-based Q&A sites and interestingly, the less structured and open Q&A sites like Yahoo! Answers outperform sites that depend on specific individuals.

2.6 Systems Mining Expertise

Competence management systems (CMS) play an important role in corporate efforts to ensure the achievement of strategic goals and thus gain sustainable competitive advantage. The major task of a CMS is the provision of information describing an individual's expertise. This information is used to support tasks like expert finding or workforce planning [Draganidis and Mentzas, 2006]. A user's competence information is also used for personalizing services. For instance, in learning management, recommendations for future learning activities are adapted to users' expertise. To gain user acceptance for a CMS, it is necessary to leave the ultimate control of profiles to the users [Lindgren et al., 2004]. Even though competences may be derived implicitly, the users should always be able to scrutinize them. A review of CMSs [Draganidis and Mentzas, 2006] reports that employees are increasingly supplied with self-service portals to maintain their competence profiles.

Locating expertise in order to solve difficult problems collaboratively is a crucial issue for an organization's effective performance. When seeking experts people are interested in "Who knows about topic X?", "How much does someone know about X?" or "How does someone compare to others with respect to topic X?". [Seid and Kobsa, 2003] identified two main motives for seeking an expert: (1) as a source of information and (2) as someone who can perform a given organizational or social function. The larger the company and the more geographically distributed, the more important the task of expert finding. Some people assist others to find

experts that possibly help them out on certain problems by means of referrals. Expert finder systems are designed to automate this process. According to [Mockus and Herbsleb, 2002], such systems need to meet the following requirements:

- Identify experts quickly and easily while not overloading a few individuals.
- Allowing users to find alternatives when some experts are not available.
- Support users in or even automate the construction of their expertise models as well as gather information about users' social networks.

In order to be able to provide effective expert finding, systems need to identify experts either via self-assessment and/or automatic analysis of expert communications, publications and activities. They further need to measure the type and level of people's expertise as well as validate its breadth and depth [Maybury, 2006]. The main task of expert finders comprises two steps: First, expert finders extract individuals' expertise profiles and secondly, they provide users with a list of candidate experts based on the users' expertise queries [Balog and De Rijke, 2007]. As for the first step of expert profiling, [Becerra-Fernandez, 2006] reports that in most cases expertise is just identified rather than measured gradually, although measuring expertise levels may improve expert finder results since they could execute more detailed comparisons of users' expertise profiles. In addition, the introduction of expertise levels can serve another purpose. Namely, users profiled by an information system want to be adequately represented especially with respect to their expertise level [Reichling and Wulf, 2009]. Furthermore, expert profiling is mostly based on users' self-assessments [Becerra-Fernandez, 2006]. On the one hand, employees' self-assessments facilitate a quick establishment of a company's expertise repository. On the other hand, self-assessments are inherently subjective and thus a comparison between users becomes difficult since users apply their own standards to self-assessment. Besides that, describing and maintaining one's expertise profile is perceived as annoying and frustrating [Mockus and Herbsleb, 2002].

2.7 Sources for Expertise

In this section, we review several approaches to expertise modeling which are distinguished by the source of expertise evidence they use. For instance, [Razmerita et al., 2003] extract users' expertise based on usage data such as number of contributions to the system or the number of documents read by users. However, the most commonly used sources of evidence today are human assessments, documents and network structures as we briefly describe in the following.

2.7.1 Human Assessments

Users' expertise is mostly determined explicitly, i.e., either by users themselves or by other people. For instance, [Reichling and Wulf, 2009] present an approach where users self-assess and publish their expertise in an organization's yellow pages. Self-assessment is generally widely

used, confer [Pernici et al., 2006] [Razmerita et al., 2003] [Harzallah et al., 2002]. Users' self-assessments are sometimes also combined with other people's assessments. For instance, [Davenport and Prusak, 1998] gather users' self-assessments as well as rates by their superiors within an iterative process. Similarly, the social search engine Aardvark [Horowitz and Kamvar, 2012] indexes people's expertise by gathering self- and peer-assessments.

[Schmidt and Braun, 2008] propose an approach to collaborative competence management. Instead of following the traditional top-down approach where ontologies are developed by domain experts in formal, regular meetings, they suggest a bottom-up approach where everybody in the organization describes others by simply tagging them. These tags are supposed to describe people's expertise. Their study results show that it is indeed possible to retrieve expertise from tags and that the process of people-tagging supports reflection on individuals' expertise as well as on organizational expertise. A similar work with regards to people-tagging indicates its value for the collective maintenance of community members' interests and expertise profiles [Farrell et al., 2007]. In addition, the authors found that none of the people's tags observed during their study was inappropriate nor offensive to the people being tagged.

2.7.2 Documents

Documents are written by individuals. Thus, they provide potential sources to extract expertise information about their authors. There are several strategies for associating documents and people to generate expertise models. Some of them are:

- Documents holding a person's name: [Zhu et al., 2005], [Balog and De Rijke, 2007].
- Emails sent or received by a person: [Campbell et al., 2003], [Ehrlich et al., 2007].
- Research publications written by a person: [Taylor and Richards, 2009], [Song et al., 2005], [Rodrigues et al., 2006].
- Web pages authored by a person such as content for Wikipedia [Demartini, 2007].
- Software code written by a person [Vivacqua and Lieberman, 2000] or change history data in software version control systems [Mockus and Herbsleb, 2002] [McDonald and Ackerman, 2000].
- A person's curriculum vitae [Harzallah et al., 2002].
- Project documents produced by a person: [Sure et al., 2000], [Ley and Albert, 2003].

In the following, we will briefly describe a selection of approaches relying on users' documents to give an idea how documents can be exploited to determine users' expertise. [McDonald and Ackerman, 2000] propose a flexible architecture for an expert recommender system. This system includes a component that deploys heuristics for associating people with certain expertise. They conducted a field study at a software company with participants represented by developers. Two systems constitute the sources for expertise evidence, i.e., the version control system and the support database. Hence, developers are either associated with the explicit changes they

made to some parts of the code or with problems they solved in the course of a support activity. Both code changes and customer problems are attached with various metadata.

In the context of software development, [Mockus and Herbsleb, 2002] propose a quantitative approach to measure expertise based on data obtained from a software change management system. This kind of systems records changes to a specific part of software including information about time, author, motivation and changed code lines. Changes are associated with users' expertise and can be distinguished based on various meanings such as fixing a problem or adding new functionalities to a code module. Depending on the type of changes programmers earn numbers of expertise atoms (EAs). Their level of expertise is then measured by the number of EAs to specific deliveries. Users' expertise levels given a certain delivery artifact are calculated by summing up collected EAs.

[Zhu et al., 2005] argue that documents are a primary resource for discovering information about peoples' expertise and associations. Documents such as web pages and reports reflect day to day activities within an organization. The authors present an approach to build people's expertise models by extracting named entities from these documents. In this sense, named entities represent persons as well as subject matter terms which build a matrix of co-occurrences in the given documents. Extracted subject matter terms are presumed to indicate expertise. Each of these person-expertise pairs holds a value corresponding to the frequencies of co-occurrences found among all documents. Based on these figures, experts are ranked in a list given a certain subject matter.

[Balog and De Rijke, 2007] devise two profiling algorithms enhancing the performance of state-of-the-art expert finding. Their first method automatically constructs users' expertise models based on the top n documents retrieved from a query related to a certain expertise area. Documents are associated with users. Users, identified from the retrieved documents, are then described with the given knowledge area where the expertise levels are determined by summing up the relevance scores of the retrieved documents associated with users. This particular method does not differentiate between the roles of users or the extent of contribution users may have made to documents. In their second method, the authors use keyword similarity of users' expertise models and knowledge areas. To do so, this method extracts the top 20 keywords for each document by means of the TF-IDF measure. Then, all keywords from these documents are associated with the given knowledge area. Similarly, users are indexed with the keywords extracted from documents associated with their names. Based on the sets of keywords for knowledge areas as well as users, the method estimates the users expertise levels by means of the ratio of co-occurring keywords and the set of total keywords in the knowledge area. A system using one of the proposed methods responds to a query about a certain knowledge area with a ranked list of experts sorted by their expertise levels.

2.7.3 Network Structures

Besides documents, the links between people as well as the links between documents became popular for users' expertise modeling. [Campbell et al., 2003] present a method to rank experts based on email communication. Emails contain precious information about users' attributes such as activities, interests and priorities. Another valuable aspect of exploring emails to identify expertise is that emails naturally represent the change of someones attributes over time, a major

challenge in expertise modeling. The proposed algorithm starts with collecting emails regarding a certain topic. It then extracts people involved in these communication data and apply the HITS (Hyperlink-Induced Topic Search) algorithm by [Kleinberg, 1999]. By means of HITS they calculate scores depending on whether a person acts as an authority or as a hub in the network. They assume that an expert in the network will reflect an authority rather than a hub. Experts are finally ranked to a given topic based on these authority scores.

[Demartini, 2007] suggests a similar approach that applies HITS on Wikipedia[2] articles. The authors of articles are ranked according to their authority scores. Besides applying HITS to Wikipedia content, [Demartini, 2007] explores the cites in Wikipedia articles based on the assumption that authors who cite another article are somehow competent in this cited article. In particular, a cite in Wikipedia is represented as a HTML link. To expand users' expertise model, a number of N words directly surrounding such a link are added to the model.

[Song et al., 2005] extract users' expertise based on a collection of research papers. They build an *ExpertiseNet* where nodes represent expertise categories. To begin with, research papers are classified to the expertise categories. The level of users' expertise (the authors of the papers) in a certain topic is calculated by the number of their publications in the given topic. [Song et al., 2005] incorporate citation information to describe the relations between expertise categories. Thereby, citations are considered in two directions, i.e., outgoing citation links (a publication of a user influences another publication/user) and incoming links (the publication of a user is influenced by others' publications). When seeking for experts, the system starts with identifying people having the expertise of interest. Then it evaluates if certain relational patterns between a user's expertise topics exist that might refine the user's ranking in comparison to others.

2.8 Mining Expertise Using Ontologies

In the expertise modeling field, ontologies are basically used to represent users' profiles (confer 2.4.1), to expand incomplete definitions of expertise [Colucci et al., 2003] or to integrate expertise with other sources [Liao et al., 1999], for instance, relating a user's expertise with a certain project in a company. Ontologies support the matching of users' profiles with either a query or with other profiles [Thiagarajan et al., 2008] . For the latter, user profiles are mostly compared with others that represent expertise required to handle certain tasks, for instance, to find appropriate people staffing a project team. In this section, we will review approaches that exploit ontologies during expertise extraction and expert finding respectively.

[Vivacqua and Lieberman, 2000] introduce an approach that automatically generates user models based on Java source files for the purpose of expert finding. The proposed system periodically reads through users' Java source files to determine the users' expertise about certain Java concepts and classes. In particular, the system verifies what constructs are used, how often and how extensively, and compares these figures to the usage levels of peers in order to establish levels of expertise. This is rather similar to the TF-IDF measure in that the more users work with classes that are not generally used, the more relevant these are to their expertise models. The expertise model represents a list of classes and corresponding expertise levels. Constructs

[2] http://www.wikipedia.org/

in Java are hierarchically structured and organized in packages according to certain application fields. The system exploits this background knowledge to match keywords entered by a help-seeking user by exploring Java concepts that are similar to the given keywords. They found that users were mostly underestimated by the system where on average, deviation of the calculated expertise levels from users self-assessments amounts to 43% given expertise levels ranging from 0 to 100%.

[Sure et al., 2000] present two systems supporting organizational skill management. One refers to the matching of employees'/applicants' skill profiles with current positions' requirements. The second system concerns the extension of individuals' skill profiles stored in the database. By means of metadata that is annotated to documents generated in the organization's environment (e.g., project documents), they draw inferences to extend skill profiles. In particular, this inference mechanism exploits the structure and rules given by an ontology that was exclusively designed by experts from human resources. The ontology serves as the source of metadata by which documents are annotated. For instance, a rule that extends data about a programmer in the profile database reads as: "If a programmer worked for a project, in which a specific programming language has been used, than this programmer has at least some experience with the language." Skills being inferred with such rules are simply added to the profile database with the value "beginner". A similar approach using annotations is that of [Harzallah et al., 2002]. They help job applicants to annotate expertise described in their curriculum vitae with concepts defined in a domain ontology. Using this shared vocabulary does not only facilitate the matching process of e-recruiting services, but also allows to exploit ontology relations for reasoning.

[Oliveira et al., 2006] present a knowledge management system to support scientific communication within research centers and universities. An essential part of their approach is a competence-mining module that measures expertise from different types of documents, e.g., project definitions, blog posts, emails, personal web pages. To identify expertise the system uses text mining in conjunction with a lightweight ontology that is manually maintained by domain experts. This ontology is mainly used to tailor the terms gained from text mining to the given domain. Besides text extraction, the system gathers additional information about the interests of users. This is achieved through a web mining facility. They hypothesize that interests may also indicate some degree of competence in a certain environment.

The *eCompetence* management tool by [Pernici et al., 2006] allows users and domain experts equally to manage the system's ontology via a graphical user interface. The authors argue this procedure with the fact, that ICT competences evolve faster than their formal codification. The system's main task is to analyze the gaps between user profiles and standard profiles. For this analysis, the competences in users' profiles are mapped to concepts in the ontology. A standard profile consists of a set of required competences represented by certain concepts in the ontology. Hence, to measure the gap of profiles, the system compares the set of required competences with the set of users' competences previously mapped to the ontology. In this case, the ontology provides valuable information about the relationship between competences to go beyond the matching of exact competence terms. For instance, a user being able to program in *C* but does not explicitly know *Java* will be declared (by means of concept relations) as being able "to program in a programming language". The latter competence is part of the required competences and thus

represents a full match. Similar to this approach, [Liao et al., 1999] use a competence ontology to empower a knowledge-based system to effectively find persons to accomplish a given task. Persons are represented with their user models holding a set of instances from the underlying domain ontology. Due to the relations between competences in the ontology, it is possible to infer additional knowledge about users as well as expand the scope to identify certain expertise.

Linked Open Data (LOD) is a database initiated by the W3C Semantic Web Education and Outreach Interest Group [3]. Its basic notion is to extend the Web with a data commons by publishing various open data sets as RDF on the Web and by setting RDF links between data items from different data sources. As of today, the database consists of 31 billion RDF triples, which are interlinked by around 504 million RDF links. In other words, this database represents a huge ontology. [Stankovic et al., 2010] evaluated this database whether it provides a valuable source for finding experts. Therefore, they tested traditional expertise hypothesis such as "If a user wrote a scientific publication on topic X than he might be an expert on topic X" given the data in the LOD database. The idea behind this is mainly that expert finders operating on LOD can provide a more complete picture of the profiled users than expert finders based on closed systems (e.g., email program). However, since expert search often relies on data that is inherently private, e.g., emails and content in corporate intranets, LOD does not constitute a perfect "all-rounder". Thus, they conclude with recommendations to LOD publishers to make their data even a better source of expertise evidence.

2.9 Expertise Extraction in Online Communities

Online communities provide a rich source of evidence for expertise. This is in particular true for communities where people share their experience while collaboratively work on problem-solving tasks. Thus, a considerable amount of research has been done lately that pay attention to such communities, see [Agichtein et al., 2008] [Harper et al., 2008] [Rodrigues et al., 2008] [Sun et al., 2009] [Lu et al., 2009] [Jiao et al., 2009] [Blooma et al., 2010] [Pal and Konstan, 2010]. In this section, we give a brief review of selected approaches exploiting information given in online communities.

[Zhang et al., 2007] seek to enhance online communities with expert finders using graph-based algorithms exploiting social networks. They present a method to generate a ranked list of experts sorted by their levels of expertise. These experts are members of the online community which communicate amongst each other by means of posting questions on the one side and providing answers on the other. From these social interactions, a post/reply-network emerges that models the relationships between the users of the online community. To exploit these post/reply-network, [Zhang et al., 2007] propose *ExpertiseRank*. The intuition of this algorithm is that if person B is able to answer A's question, and C answers B's question then C's expertise rank should be boosted, not only because C was able to answer a question, but because C answered a question of B who still showed expertise by answering someone other's question. Besides *ExpertiseRank*, they also propose a method called *Z-score* that simply considers that amounts of posts and replies of users whereby users that reply more than they post possibly have higher expertise than users that primarily ask questions. They evaluated these measures by means of data

[3] http://www.w3.org/wiki/SweoIG

in an online forum, namely, the Java Forum. Human experts provided the ground truth based on users' contributions. Both algorithms did very well compared with the expert estimates. However, their approach only estimates expertise on a rather general level, e.g., whether users are Java beginners or top experts. They neither explore more specific knowledge nor do they measure absolute levels. Thus, they interpret users' top expertise in relation to others' expertise, but that not necessarily mean the former users have actual top expertise in Java.

[Kao et al., 2010] suggest a hybrid approach to find experts in a Q&A community. Users provide answers to posted questions related to a particular topic category. These answers reflect answerers' expertise about topics in the given category. The proposed method to find experts is based on various aspects, i.e., the users' subject relevance (relevance of users' domain knowledge to target questions), users' reputation (amount of best answers in a given category) and users' authority (link analysis). In order to build users' expertise models, they consider users' textual submissions as well as quality measures (e.g. peer votes) of the users' historical question-answer pairs. As for peer votes, they assume that the more votes answers receive the more important they are and thus the higher the users' expertise levels for topics assigned to a category. Expertise levels are associated with expertise topics extracted from the answer body by means of TF-IDF. However, the difficulty level of a question-answer pair is not considered in their approach. Consequently, the measure can be used for ranking experts but not to find experts by means of absolute expertise description such as "beginners". Their results show that peer votes as well as considering the time factor can improve the quality of computing user knowledge profiles.

[Haselmann et al., 2011] measure skill profiles in online social networks. People publish their expertise with the purpose of advertising themselves to other members of the community network. The authors' main concern is the trustworthiness of such profiles. Hence, they devise a conceptual model where users specify their expertise together with the evidence confirming their experience. Users self-assess their experience by assigning a proficiency level (novice:1, advance:2, expert:3). The system calculates users' expertise scores by considering other users confirmations to the reported experience. Basically, they build the weighted average mean of confirmations (serving as weights) and the given proficiency levels from users' self-assessments. In addition to these scores, they examine credibility of scores by integrating the proficiency levels of users confirming other users' experience. The essential character of their approach is that users self-assess their expertise first. Then others confirm this expertise, however, these people are not able to alter the users' original expertise estimates. Their first experiment, conducted with a small group of people, suggest a closer integration of expertise scores and its credibility measure. They observed that users, stating their expertise, might have too much influence on their scores.

2.10 Summary

[Mockus and Herbsleb, 2002] emphasize the need to quantify expertise so that (1) potential experts can be compared with one another in terms of their expertise levels and (2) so that experts can be searched based on a required distribution of expertise, i.e., generalists vs. specialists. The more advanced approaches we reviewed calculate expertise scores to rank experts. Ranking

implies that users' expertise levels are calculated relative to the levels of others and thus do not reflect the users' absolute expertise levels. However, ranking of experts is useful though, as long as we look for the best candidates available in an organization. A shortcoming of ranking approaches is that they can not determine whether a candidate has the required proficiency level to accomplish a particular task, for instance, when staffing a SW project team that requires intermediate Java programmers rather than Java top experts. In addition, it is not desirable to contact the best ranked candidates all the time, since they could be better employed in more complex tasks rather than helping out on simple problems.

We primarily focussed on existing research works that automatically calculate user expertise. [Ley and Albert, 2003] raise the issue that automatic expertise modeling needs to be justified by human actors such as human resources managers, knowledge engineers or even by employees themselves. Thus, they propose a semi-automatic method to determine individuals' expertise. They confront employees with the documents they created based on their work assignments and systematically ask which competences they applied to accomplish their work. A more recent approach to semi-automatic modeling is proposed by [Reichling and Wulf, 2009]. They present an expert recommender that identifies users' expertise based on various types of document files located in their personal folders. In addition, the users' expertise models are extended by their self-descriptions published in an organization's yellow pages. Finally, they subsume terms gained from text mining performed on these sources into users' expertise models. However, their motivation for a semi-automatic method is not primarily that humans need to correct the system's modeling results, but to consider the privacy of individuals being modeled.

CHAPTER 3

Measuring and Displaying User Expertise

In this chapter, we propose both a method to measure users' expertise and a user interface to open calculated expertise models to users for scrutiny. Basically, the expertise calculation method is based on two assumptions:

> ASSUMPTION 1: Users demonstrate their expertise while authoring contributions in online communities regarding their individual experiences. In particular, the words and phrases people use serve as indicators for their actual performance.

> ASSUMPTION 2: People use different kinds of interaction when they meet in an online community to collaborate in problem solving tasks. Information about these interactions can be leveraged to determine and qualify users' expertise.

Our approach to calculate users' expertise consists of several steps as illustrated in Figure 3.1. First of all, we selectively extract topics from users' contributions. In the second step, we determine the value of extracted topics by means of the contributions they originate from. Next, we exploit user ratings given to contributions in order to further qualify the values of expertise topics. Since topics can either be of a general or specific kind, we make use of an ontology to

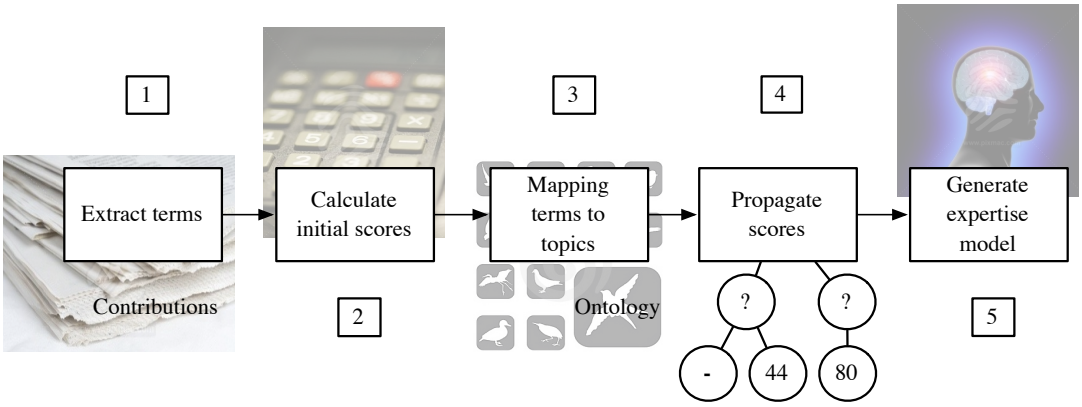


Figure 3.1: Steps during expertise calculation.

align these topics regarding their abstraction levels. Finally, we assign a certain subset of topics to users' expertise models, which are finally presented by means of an Expertise Cockpit.

During the design of the calculation method, we iteratively focussed on the various steps of the algorithm. As a consequence, this thesis is characterized by three versions of the algorithm each accompanied with its own evaluation cycle. Evaluation is conducted in the form of user experiments where the subjects are represented by students at university. The present chapter is dedicated to the first two versions of the algorithm. The third version is an aggregate comprising mainly this chapter's work as well as the work done in Chapter 4. The details and the evaluation of the third version are covered by Chapter 6.

To begin with, the following section describes the knowledge management portal students used to share knowledge amongst each other. Section 3.2 introduces the first version of the algorithm acting as a pilot in order to test basic functionality of the algorithm's very first steps. Furthermore, this section also covers the second version that is based on the pilot design but introduces enhancements such as utilizing the background knowledge more thoroughly. More importantly, the second version of the algorithm introduces absolute expertise scores to model users' performance for the first time as well as a reliability measure to estimate the trust in calculated expertise predictions. Section 3.3 presents a user interface opening the calculated expertise models to individual users. First, we evaluate various interface elements that may be supportive in scrutinizing expertise models. Based on the findings, we devise an Expertise Cockpit used for evaluating the third version of the proposed score calculation method.

3.1 Sharing Experience with TechScreen

In order to design and evaluate a method for expertise calculation, we agreed on providing our own knowledge sharing system called *TechScreen*[1]. This comes with the advantage of having full control of the environment later used for experimentation. On the downside, we perhaps

[1] https://techscreen.tuwien.ac.at/

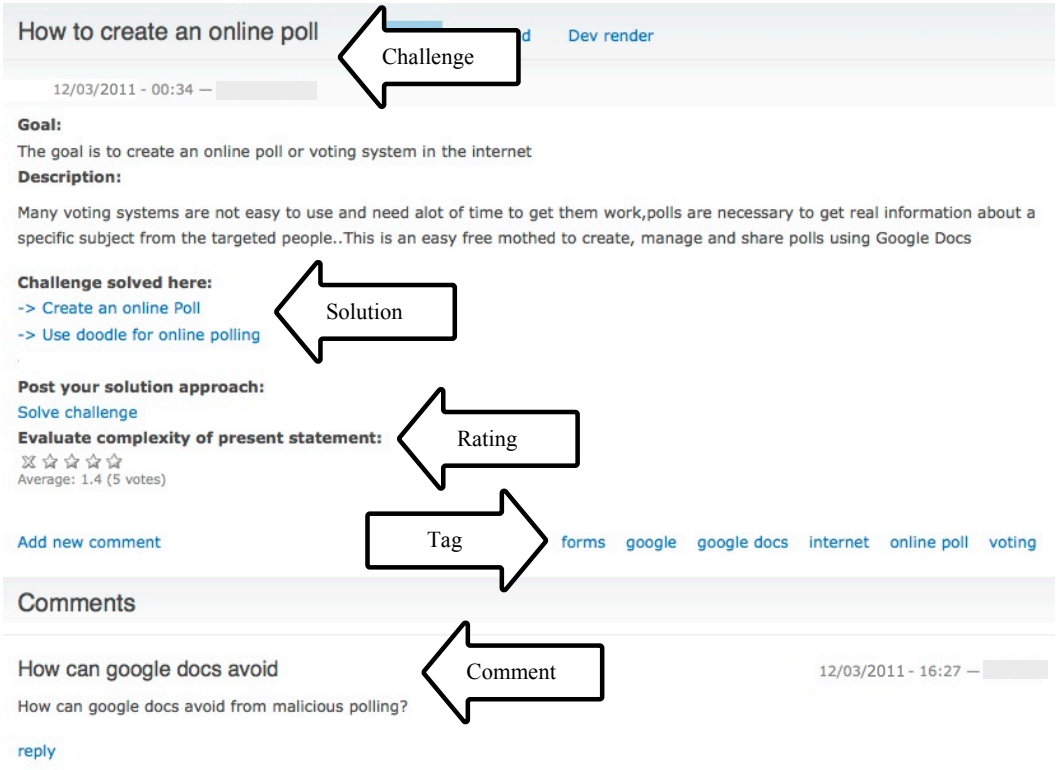


Figure 3.2: Display of available contribution types on the example of a user's challenge.

have to struggle with collecting sufficient data, which is certainly a crucial point when doing research regarding users submitting content to online communities.

Online communities provide members with different types of artifacts to share knowledge. Considering online communities with the purpose of solving problems collaboratively, we found that communication artifacts share certain commonalities. We analyzed community-driven question-answering services like *Microsoft TechNet*[2] and *Yahoo! Answers*[3], forums like *Informatik Forum*[4], but also an online community sharing bookmarks called *Delicious*[5]. On these platforms, knowledge is mostly shared by simple text structures including a title and a text body. Such artifacts may be tagged as well as rated by peers. In the context of particular issues, users are engaged in discussions by posting comments or even longer texts.

3.1.1 Contribution Types

In the following, we refer to these commonly used artifacts as contribution types. So, based on these contribution types, we set up an online community for the purpose of collaborative learning. The community members are represented by master students, who share challenges they face during their day to day activities related to internet technologies. Such challenges mostly arise from situations students have to cope with regarding a particular learning content. However, students are also encouraged to report on challenges they face in private contexts. To do so, students post challenges and build or refine solutions to these challenges by working together with their peers. We assume that terms used by students in their contributions as well as the terms they use in later discussions about these contributions serve as indicators about their expertise.

Figure 3.2 illustrates a challenge stored in the system as it is presented to users. The top part shows the description of the challenge comprising its title, goal and actual content. This particular challenge is already associated with solutions from two peers as displayed in the middle part of the Figure. To take a view on these solutions, please refer to Figures A.1 and A.2 in the appendix. In case other peers have additional ideas on how to solve this challenge they can follow the respective hyperlink located below the current list of solutions. Users are encouraged to rate the challenge's difficulty level. The more difficult the challenge, the more expertise is necessary to solve it as well as to formulate its problem description. Furthermore, people can associate tags with the challenge and start a debate on it.

It seems obvious from Figure 3.2 that contribution types are linked with each other. These links allow to combine the texts behind individual contributions. We will later exploit this combined information for expertise calculation.

3.1.2 Architecture and Technologies

TechScreen is a service installed on a dedicated server located in the university's computer network. Figure 3.3 illustrates its main connections to the outside world including services being offered in the university's intranet as well as services available on the public internet. TechScreen provides the facilities to share knowledge online by means of contribution types as described in the previous section. In addition, it offers search capabilities that help to locate interesting content and it accommodates a forum where users can discuss issues besides their technical contributions. However, in the context of this thesis, we focus on our method to calculate users' expertise as presented in the next sections. Therefore, we only describe those parts of TechScreen that are related to the proposed calculation method. For instance, the user interface we refer to on top in Figure 3.3 does not represent all the components that are actually provided to the user, but only the Expertise Cockpit. For details on the architecture of the user interface please refer to Section 3.3.2.

[2] http://social.technet.microsoft.com/Forums/

[3] http://answers.yahoo.com/

[4] http://www.informatik-forum.at/

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

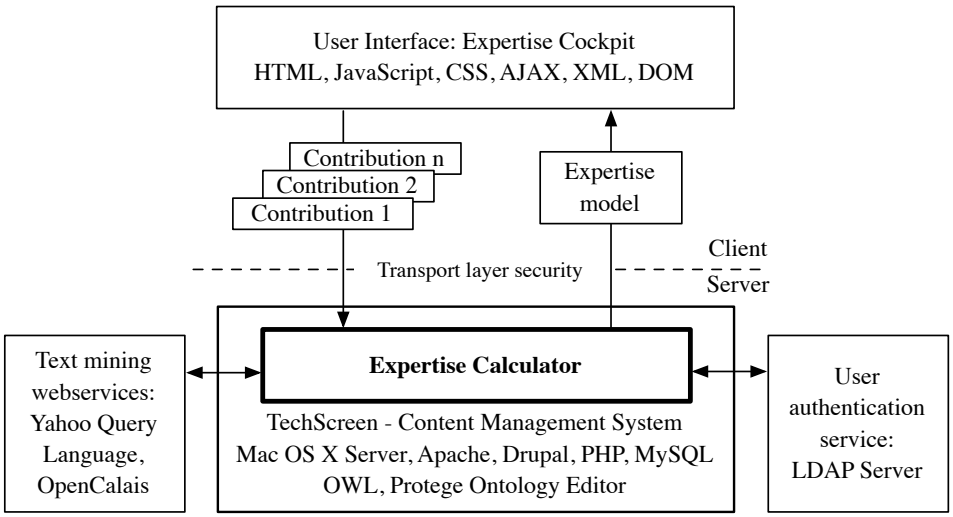


Figure 3.3: System architecture.

The open source content management system Drupal[6] builds the technological heart of TechScreen. In our setting, Drupal is installed on an Apache web server running on a Mac OS X Server operating system. In general, a Drupal installation consists of a mix of core and contributed modules. Thus, the Expertise Calculator algorithm is realized as a set of contributed modules written in the programming language PHP[7]. These modules rely on a MySQL[8] database for persistent data storage. Within its framework for building dynamic web sites, Drupal offers metadata functionalities using controlled vocabularies. Based on these functionalities we were able to integrate a competence ontology structured by means of the web ontology language OWL[9]. Prior to its integration, this ontology was constructed using the open source ontology editor Protégé[10]. For the system's interaction with the user, we applied technologies commonly used in web applications such as HTML, JavaScript, Cascading Style Sheets (CSS), AJAX, the Document Object Model (DOM) and XML.

Students at the Vienna University of Technology receive for the time of their studies a unique account that allows them to use online services either provided by the university or from external university partners, e.g., free access to scientific works published by online libraries. TechScreen constitutes yet another service that can be accessed by using student credentials. Therefore, we connected our Drupal installation to the university's authentication service as displayed on the right in Figure 3.3. On user login, Drupal sends a request to the authentication server via the Lightweight Directory Application Protocol (LDAP). On successful authentication, TechScreen receives the registration number as well as the student's email address from the server and pro-

[6] http://www.drupal.org
[7] http://www.php.net
[8] http://www.mysql.com
[9] http://www.w3c.org/owl
[10] http://protege.stanford.edu/

vides access to the user.

Since we associate users' textual submissions with their expertise, we need to analyze the terms they use to explicate their experience. To do so, we utilized free text mining services available on the internet, i.e., the *OpenCalais Service*[11] and the *Yahoo! Query Language*[12]. Using natural language processing, machine learning and other methods, both services offer a broad range of text mining features including named entity recognition, the extraction of facts or even events. Importantly, text mining can be restricted to a certain domain, which is in our case the domain of internet technologies. We tested both services with different set of texts. The results showed that both services are able to determine topics that are relevant to the requested domain. However, extracted topics mostly differ from each other, which is most-likely caused by different vocabularies working in the background of each individual service. Thus, we agreed on aggregating the results from both services yielding a richer set on topics describing a contribution's subject matter.

3.2 Calculating Expertise Scores and Reliability

In this section, we propose a method to determine users' expertise represented as expertise scores. An expertise score is associated with an expertise topic and shows a value between 0 and 100 points. This numerical range covers expertise areas ranging from a novice to a beginner level, from beginner to intermediate and from intermediate to the top expertise level. Expertise scores are based on different types of evidence, some of them are less and some of them more reliable for calculation. Hence, for each calculated expertise score, we compute a confidence level representing the trust in this score. We calculate for each user an expertise model comprising a set of topics, its scores and confidence levels. After that, we devise a user interface opening these models to the users for two reasons. First, to let the users scrutinize their models, which is an important characteristic of user modeling systems in order to gain users' acceptance. And secondly, for the reason to collect users' feedback regarding their calculated expertise. Based on users' feedback, we later evaluate the accuracy of the proposed score calculation method.

The remainder of this section is organized as follows. In Section 3.2.1, we conduct a pilot experiment to test if we are able to extract proper contexts from user contributions and whether users are satisfied with the provided features for sharing their experience. We also use this pilot run to construct a solid base ontology and to perform a first experiment with a rather simple approach of expertise calculation. We proceed in Section 3.2.2 with finding weights for the individual contribution types representing their value during expertise calculation. Based on this weighting model, we devise a method to actually measure expertise scores on an absolute level as described in Section 3.2.3. In addition to expertise scores, we design a measure to calculate the confidence in these scores (Section 3.2.4) and perform a first evaluation in Section 5.2. We summarize and conclude our findings in Section 3.2.6.

[11] http://www.opencalais.com/

[12] http://developer.yahoo.com/yql/

3.2.1 Pilot Experiment

The basic idea to calculate users' expertise is displayed in Figure 3.1. A key ingredient of the algorithm constitutes the background knowledge used to identify and align topics extracted from contributions. We use an ontology for representing this knowledge. Even though ontologies are supposed to be a shared description of concepts within a domain, we realized that it is still hard to find an existing ontology covering the domain of internet ontologies. Despite the fact that constructing an ontology with a considerable amount of concepts is known to be tedious, we decided to generate an ontology on our own. Furthermore, we extract terms from users' contributions that are later mapped to ontology topics. Although we have already run first tests regarding the performance of text mining services, we still need to apply them in a real environment given authentic user contributions. Lastly, since we establish a new platform for sharing user knowledge, we are curious whether the provided features are convenient enough to satisfy users needs and achieve user acceptance. For these reasons, we conduct a pilot experiment aimed at the following goals:

- Generate a base ontology describing the domain of internet technologies.
- Apply text mining services and map terms to ontology topics.
- Test the usability of our knowledge sharing platform.

The main focus of our research lies not on the knowledge sharing platform introduced in Section 3.1. In fact, TechScreen is just a means that provides an environment to collect user data supporting the design and evaluation of the proposed expertise calculation algorithm. Thus, running a pilot experiment meaningful to our research, does not only mean to test usability of the knowledge sharing platform and to examine certain steps of the future calculation method independently, even if these issues are undoubtedly important. In fact, it does also mean that we aim to design at least a simple approach to capture users' expertise in order to get a first feeling about particular challenges in determining expertise. Moreover, it allows to explore users' general acceptance with respect to expertise predicted by a system.

An Ontology Modeling Internet Technologies

[Golemati et al., 2007] present an ontology that incorporates concepts and properties used to describe the user model. Their particular aim is "to create a general yet extendable ontology that will be able to adapt to the needs of every application". This ontology emphasizes the need to represent expertise by its breadth, depth and finesse. As for the latter, they mean scores or levels of expertise.

In this thesis, expertise models are represented by ontology overlays. An overlay is understood as a subset of topics from a domain ontology. This overlay is then associated with users expertise showing expertise levels in particular topics. In the course of our research, we examine how to calculate expertise in the field of internet technologies. Therefore, we constructed a competence ontology holding expertise topics related to this domain. As we already described in Section 2.3, such ontologies are predominately structured in hierarchies, i.e., the more general/specific a topic, the higher/lower its place in the hierarchy.

In order to design the ontology, we followed the bottom-up as well as the top-down approach. We started with the top-down approach and thus defined fields of expertise in which we expect that, for instance, a web engineer needs to be competent in. With the help of various resources like the categories used in Wikipedia[13] and the computer science curricula guidelines published by the ACM [ACM, 2008], we agreed on the following eight expertise fields subsumed under the root topic internet technologies:

1. Programming
2. Databases
3. Web Concepts
4. Web Development
5. Network
6. Security
7. Application Software
8. Operating Systems

To begin with, we identified and assigned topics to each of these expertise fields based on the aforementioned resources and with the support of domain experts at university. We further enhanced the ontology by following the bottom-up approach, i.e., after collecting the first sets of contributions from students, we examined which terms they used to describe their experience as well as which terms they used for tagging contributions. We explored the term use on the hand by manual text analysis and on the other hand with the support of text mining services. More specifically, we applied the following steps for each contribution:

1. Discard terms that are not related to the target domain at all.
2. Discard terms actually related to the domain but having a too general notion.
3. Find relationships amongst terms and determine synonyms.
4. Integrate terms and synonyms with the actual ontology .

As a consequence, we obtained a competence ontology holding the most indicative terms regarding knowledge about internet technologies. At this stage, the competence ontology contains 454 topics and 223 synonyms. Expertise topics are linked via a *is-a* relationship commonly used in traditional hierarchy structures. In Chapter 4, we introduce a more specific type of relationship that allows to differentiate the degree of similarity between topics. So far, the ontology holds only expertise topics and the relations amongst these topics. Since we aim to calculate expertise scores for individual users, we need to enhance the current ontology with a user and a score concept. Figure 3.4 illustrates a snippet of the ontology including these new concepts. When using an ontology, one often distinguishes between a concept class and its instances. In our context, we refer to topics in the domain ontology as classes, whereas instances are represented by topics associated with a user and estimated with a certain expertise level.

A crucial point for any ontology concerns the strategy on how to keep the represented knowledge up to date. This is especially true when modeling a domain such as internet technologies

[13] http://en.wikipedia.org/wiki/Portal:Contents/Categorical_index

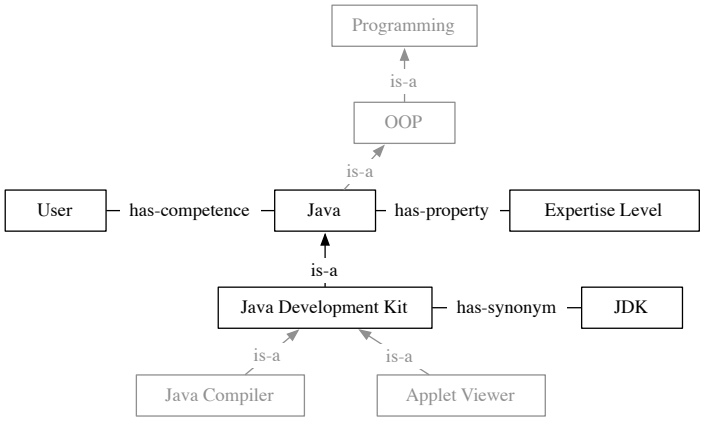


Figure 3.4: An example snippet showing the structure of the competence ontology.

where new topics emerge rather quickly as well as existing topics more or less disappear unnoticed. For example, such strategies may involve the regular maintenance by ontology engineers representing domain experts or the system just stores currently unknown topics into a pool that is later evaluated by domain experts. However, due to the short cycles in which the present ontology is being used in this thesis, the issue of currentness is not as crucial as for field settings. Yet, we have continuously revised the ontology while moving from one experiment to the other.

A Simple Approach to Expertise Measurement

Besides testing the basic features of TechScreen to facilitate knowledge sharing, we also devise and evaluate a first version of our Expertise Calculator. However, we will not measure any scores yet, but attempt to calculate users' strengths given particular expertise fields. It is likely the case that during expertise calculation we will determine one or more expertise fields for each user. However, if we can not determine any expertise from users' contributions, no expertise field will be added to users' expertise models. Figure 3.1 already sketched the designated sequence of our expertise calculation approach. In the following we will devise a simple measure according to this sequence of steps.

Expertise calculation starts with gathering all contributions associated with an individual user who is about being modeled. Next, we apply text mining on the user' textual contributions and thus extract terms that serve as indictors for the user's expertise. After text mining we obtained a set of terms describing the user's documents that is referred to as bag-of-words representation [Hotho et al., 2005]. As already mentioned in Section 3.1.2, in order to extract terms from contributions we utilize online text mining services. Besides traditional text processing techniques such as Tokenization, Filtering and Stemming, these services make also use of advance techniques like Part-of-Speech tagging, Word Sense Disambiguation and they even adopt semantic dictionaries for term extraction.

Once a user's bag-of-words is identified, the terms will be mapped to expertise topics in the ontology. This is known to be a non-trivial task [Tsujii and Ananiadou, 2005]. One of the major

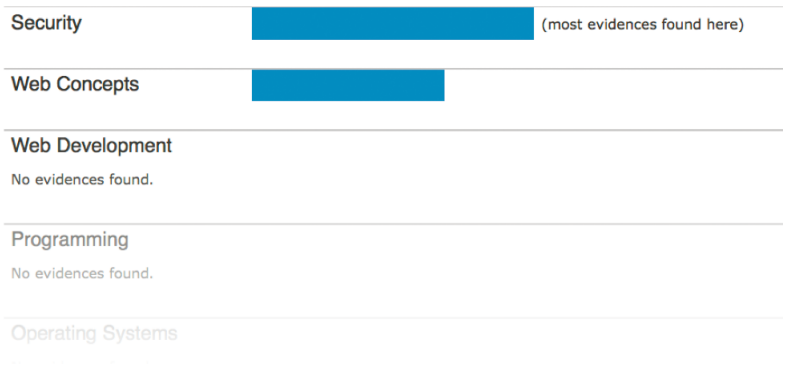


Figure 3.5: Indicating an individual's expertise using expertise fields.

problems that needs to be resolved in this regard is that of term ambiguity. In our specific case, the text mining services use state-of-the-art disambiguation features, e.g., they evaluate term co-occurrences and combine this results with background knowledge to determine the term's semantics and domain belonging respectively. However, a certain chance for mapping failures still remains.

The mapping of terms to ontology topics can be accomplished by means of various techniques. One way is to compare the labels of ontology topics with that of extracted terms. There are numerous variants available to do that. A simple one just compares labels if they are literally equal thus representing an exact match. Others use string similarity measures such as the Hamming distance [Hamming, 1950] or Levenshtein's edit distance [Levenshtein, 1966]. More advanced techniques may consider the term's co-occurrences as well as the adjacent topics of a candidate topic in the ontology. As for the first version of Expertise Calculator we rely on exact matches of candidate topics and ontology topics.

After mapping the terms to ontology topics, we count the number of topics assigned to each expertise field. An expertise field will be activated once it contains at least one topic successfully matched with an extracted term. Figure 3.5 displays the Expertise Cockpit as it is presented to users. Basically, the cockpit represents a list of expertise fields. The list is in descending order where the highest ranked expertise field corresponds to the field where the highest number of topics could be found. The bar length of the subsequent expertise fields is calculated in relation to the number of topics contained in the top-ranked field. By means of this Expertise Cockpit, users can reflect on their strengths and weaknesses even though the system's expertise predictions are indicating rather coarse-grained levels.

Pilot Evaluation and Findings

In the course of a tutorial on knowledge management, we conducted a pilot experiment with 31 master students enrolled in a computer science program at the Vienna University of Technology. Consequently, the participants of our study are supposed to have at least basic knowledge in the domain of internet technologies. We asked participants to share their experience amongst

Table 3.1: Data collected during pilot experiment

Participants	31
Contributions submitted	92 Challenges 101 Solutions 65 Comments 453 Tags 160 Ratings
Feedbacks submitted	23
Expertise field accuracy	53,44%

Figure 3.6: Users self-assess their expertise in certain expertise fields. Blue-colored fields indicate the system's beliefs about the user's expertise.

each other using TechScreen. They were encouraged to extensively use the features provided by TechScreen such as posting challenges and solutions, start discussions around these contributions as well as tagging and evaluating them. While participants were engaged in sharing their experience, we already started to analyze users' contributions in order to set up the competence ontology as presented in Section 3.2.1.

Table 3.1 displays the data we collected in a four week period. After this period we activated a new button in the TechScreen user interface by which participants could calculate their expertise models. Once participants inspected their expertise models, they were asked to provide feedback separated in two parts. In the first part, we asked participants to evaluate the ranking of expertise fields in their model. Figure 3.6 shows the feedback form we provided to the participants where expertise fields being calculated were blue-colored. Now, participants chose those expertise fields that were the closest to the contexts of their submissions. In the second part of the feedback, we asked participants mainly about likes and dislikes concerning the usability of TechScreen as well as the construction of their expertise model.

We had originally 31 participants taking part in the experiment whereas 8 participants quit before the experiment was over. Thus, Table 3.1 displays only the data regarding the remaining 23 participants. We measured the accuracy of calculated expertise by determining the percentage of correctly identified fields against the total amount of fields participants reported in their feedback. We built the average mean across all participants' accuracy figures and found that in approximately half the cases expertise fields were assigned accurately and this without eliminating potential outliers. This is a quite promising figure given the simplicity of the applied expertise measure. However, we are confident to improve accuracy by (1) using string similarity measures for ontology mapping, (2) leveraging the structural information provided by the ontology for topic alignment, (3) introducing weights facilitating the construction of a term vector model and lastly, (4) by exploiting peer ratings.

Besides accuracy results, we were mainly interested in how participants were satisfied with the usability and the set of features provided by TechScreen. Therefore, we evaluated participants' response to open questions asking after the likes, dislikes and desire for improvements. At this point we will only focus on the main issues we identified from participants' feedback. First of all, participants complained about a missing statement describing how the data is used by the system regarding privacy concerns. Most of the participants were not satisfied with the provided options to search and navigate content. Some participants raised the desire to be able to attach images and documents to contributions. They said this might help to describe one's subject matter more precisely. Because TechScreen had no former content to offer, participants in the pilot experiment struggled in the beginning with their motivation to contribute to an "empty community". However, this attitude changed the more content became available. Another desire for improvement refers to the publication status of contributions. Participants demand full control over their contributions including the option to mark a submission either as private or public. In terms of user acceptance, we consistently received positive responses that acknowledge the potential of the proposed expertise calculation method. In the course of our research, we conducted three evaluation cycles with different groups of participants. Each cycle included a closing feedback step asking practically equal questions across all evaluation cycles. Thus, for more details about qualitative feedback please refer to the summary given in Section 6.5.

To sum up, the measured expertise accuracy figures suggest that we were able to capture considerable parts of contributions' contexts. However, there is still room for improvement. By means of various techniques we will address some of them in the course of the upcoming sections. The results of the pilot experiment revealed issues that need to be implemented in order to improve the usability of TechScreen for future experiments. We experienced that the negotiation about which topics and relations will actually take part in the ontology is a challenging task.

3.2.2 Contribution Weighting Model

We understand the terms extracted from users' contributions as indicators of their expertise. Terms from one contribution type may reflect a higher and more reliable value for expertise calculation than terms originating from others. Therefore, we systematically examine each contribution type according to the questions listed in Table 3.2. Given its value for expertise calculation, we assign each contribution type a weight ranging from 1 to 5.

Table 3.2: Criteria for examining contribution types

Questions	Heuristics
1. How far does the contribution originate from experience?	The more a contribution originates from experience, the more valuable it is for expert profiling.
2. How promising is the contribution regarding the calculation of a maximum competence score?	The more action in problem-solving is involved and the more significant the occasion of contribution, the higher the level of expertise to measure.
3. How costly is the contribution to fake?	The harder a contribution is to to fake, the more valuable it is for expert profiling.
4. How likely is the contribution of high-quality?	The higher the quality of a contribution is, the more competent the author must be.

Question Q.1 is based on the assumption that people demonstrate expertise when they apply certain skills to perform an action in a real-world situation. As for Q.2, we explore users' involvement in the problem-solving process. For instance, we consider the authors of solutions to be more involved in problem-solving than users tagging a contribution. [Shami et al., 2009] introduce the principle of signal theory to estimate users' expertise based on digital artifacts like blog posts, a self-description or other information summarized in an online profile.They found that certain signals in various social software are much harder to fake than others and thus, are more reliable indicators of expertise. Therefore, we examine in Q.3 how easily a contribution type is to fake. In Q.4, we address the quality of contributions by means of their textual information. In this regard, we came across approaches, which measure the quality of *Wikipedia* articles by considering their structure and integrity [Lim et al., 2006] [Wöhner and Peters, 2009] [Hu et al., 2007]. For instance, the number of words contained in articles proved to be a good indicator of their quality [Blumenstock, 2008] [Harper et al., 2008] [Agichtein et al., 2008]. However, the robustness of such a metric seems not promising, i.e., users can easily pretend expertise by just copy and pasting texts from other sources. More recently, Wikipedia released the Article Feedback Tool [14] to engage readers in the assessment of others' article quality. Readers can rate articles regarding their trustworthiness, objectivity, completeness and writing style. During the present thesis, we test the quality of users' contributions by whether the contribution can be rated by peers.

In the following, we estimate each contribution type according to the questions listed in Table 3.2. We use arrow symbols on a four-point-scale to represent our estimates as shown in Table 3.3. For instance, the chance that *solutions* originate from experience (Q.1) is very high whereas the chance to assume experience behind a *comment* is very low.

Users post *challenges* based on problems they experience in their daily routine. While authoring challenges, users need to reflect the problem space profoundly. However, they are not

[14] http://en.wikipedia.org/wiki/Wikipedia:Article_Feedback_Tool

Table 3.3: Contribution weighting scheme

Question	Challenge	Solution	Comment	Tag	Rating
Q.1: Experience	$\uparrow$	$\Uparrow$	$\Downarrow$	$\uparrow$	$\Uparrow$
Q.2: Max Score	$\downarrow$	$\Uparrow$	$\Downarrow$	$\downarrow$	$\uparrow$
Q.3: Fake	$\uparrow$	$\uparrow$	$\downarrow$	$\downarrow$	$\uparrow$
Q.4: Quality	$\Uparrow$	$\Uparrow$	$\Downarrow$	$\Downarrow$	n/a
Weights	$\omega_{Ch} = 3$	$\omega_S = 5$	$\omega_{Co} = 1$	$\omega_T = 2$	$\omega_R = 4$

Probability: $\Uparrow \ldots$ very high, $\uparrow \ldots$ high, $\downarrow \ldots$ low, $\Downarrow \ldots$ very low

able to solve the challenge, hence it is not possible to measure a maximum competence score by only considering challenges. Users may easily fake challenges by copying and pasting text, but most of these cases will be revealed by peers' ratings.

While constructing *solutions* users reflect the problem as well as the solution space. Users solving others' problems indicates that solvers may have superior expertise than the users who post problems [Zhang et al., 2007]. Therefore, we assume that a solution allows to measure the maximum possible expertise score. A solution is rated by others and thus very costly to fake. Its quality with respect to completeness and accuracy is qualified by ratings as well.

Users comment on others' contributions to help them refining their contributions, ask questions or just showing their opinion. Since the motivation behind *comments* is not definitely clear, they contain lots of noise that makes them difficult to interpret [Almeida et al., 2010]. Since comments can not be rated, they represent an unreliable source for expertise calculation.

If users find certain contributions appealing, they can assign *tags* to them. This indicates that they must be somehow competent within the given topic, but we can not determine to which extent. Tags appear to be the most significant descriptive feature regarding multimedia content [Almeida et al., 2010]. However, tags can not be rated, which makes them easy to fake.

Aggregated *ratings* can be used to judge the quality of contributions [Blooma et al., 2010]. From the rater's view, ratings are easy to fake though. We assume that the majority of users only rate others' contributions if they have strong self-confidence regarding their own experience in the given topic. Ratings are very costly to fake especially the higher the number of raters is. Users being rated have to show true expertise by posting complete and accurate contributions otherwise users will respond with low ratings.

3.2.3 Calculating Absolute Expertise Scores

In this section, we devise a measure to calculate expertise represented by expertise scores. Expertise scores range from 0 to 100 points. In contrast to approaches, which rank users according to their expertise level regarding a certain subject matter, the proposed Expertise Calculator uses an absolute scale. An expertise score of 0 simply displays no expertise whereas a score of 100 points represents users' top expertise. Top expertise means that users' have achieved a

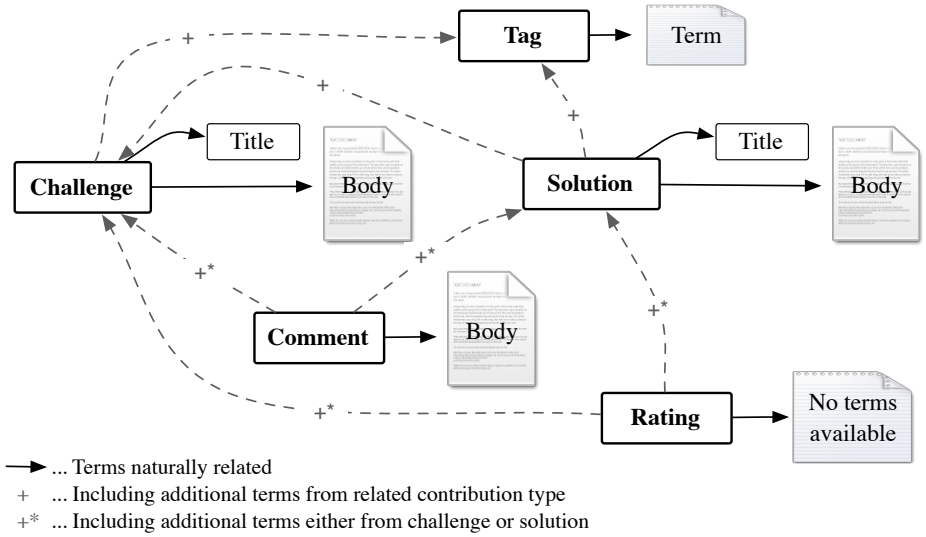


Figure 3.7: Terms associated with each contribution type. Text mining is primarily based on the directly related terms (solid arrows). However, the corpus of certain contribution types will be enhanced by terms from associated contribution types (dotted arrows) before text mining starts.

very high degree of problem-solving capability. Given this top expertise, such users are able to solve complex real-world problems where highly developed expertise regarding a certain topic is necessary. Eventually, absolute scores allow a more accurate selection of experts under particular circumstances, e.g., when seeking explicitly for Java professionals or Java learners. In order to make this absolute scale more transparent to users, it can be divided into various ranges, each labeled with a description concerning the respective expertise level. For instance, Zhang et al. [Zhang et al., 2007] introduced five levels of expertise that range from a newbie level up to the top expert level, confer Figure A.5.

The calculation of expertise scores takes several steps as shown in Figure 3.1. To begin with, we make use of online text mining services to extract terms from users' contributions. Figure 3.7 illustrates the set of words used as the input for text mining regarding each contribution type. After extracting the topics from users' contribution, each user is associated with a set of topics representing candidate expertise topics.

After topic extraction, each term gets a weight assigned, which corresponds to the weight of the contribution the term was extracted from. For instance, we assign the weight ω_{Ch} to a topic we obtained from a challenge. Equation 3.1 shows the calculation of the initial expertise score for topic t associated with user u.

$$sc_{init}(u,t) = \omega_{contribType} \cdot r_{factor} \ . \tag{3.1}$$

where r_{factor} is the contribution's rating score normalized to [1,2]. The rating score represents the average mean of rates the contribution received from peers in the community. For instance, when using a 4-point rating scale, the highest possible rating score 4 is converted into $r_{factor} =$

2, whereas the lowest possible rating corresponds to $r_{factor} = 1$. A topic associated with a weight of 5 (contribution weight) and the maximum average rating score receives an initial expertise score of 10. We transform this value to our absolute scale, i.e., a maximum initial score of 10 is transformed to a final score of 100 points. Terms originating from contributions not being rated are further processed by using default rating values. Default rating values are constant values set in the system to substitute missing peer ratings. Additionally, default rating values facilitate to overcome the cold-start problem where users are new to the community. Such users submit their contributions and want to calculate their expertise model instead of waiting until peers provide votes for their submissions.

Due to this procedure, one and the same topic may obtain initial scores from contributions of different types. For instance, a topic originates from a comment as well as from a challenge. Consequently, this topic is associated with two initial scores, one calculated with respect to the comment (with ω_{Co} and default rating value) and one based on the challenge (with ω_{Ch} and the average score of peer rates). In this case, we only assign the highest one of the two scores to the topic. However, we do not dismiss any information concerning the lower calculated score but consider it for confidence calculation described later on in Section 3.2.4.

At this stage, two problems occur. First of all, we can not distinguish topics originating from the same contribution with respect to their level of abstraction. One topic might indicate specific expertise where the other expertise topic is of a more general nature. Anyway, so far both topics show the same initial scores. Secondly, we might have identified topics that are not relevant to the domain of interest. We consider both issues within the third step of our algorithm by introducing background knowledge gained from a lightweight ontology as introduced in Section 3.4. This ontology links expertise topics in a given domain and organizes them in hierarchical order. By exploiting the ontology's structural information we are able to align a user's expertise topics. Hence, we now map these topics to ontology topics as shown in Equation 3.2. An expertise topic t is mapped to an ontology topic o_t. This allows us to eliminate topics not relevant to the domain of our interest.

$$\mathcal{T} \rightarrow \mathcal{O} : t \mapsto sim_{Levenshtein(\%)}(t, o_t) > tr_{sim} \ . \tag{3.2}$$

where $t \in \mathcal{T}$ and $o_t \in \mathcal{O}$. $\mathcal{T}$ is the set of extracted topics and $\mathcal{O}$ the set of topics contained in the ontology. Expertise topics are successfully mapped to ontology topics if they show some degree of similarity. The threshold tr_{sim} specifies the degree of similarity topics have to exceed in order to be considered for further processing. Those topics with similarity values below this threshold will be discarded. To calculate topic similarity, we adopt Levenshtein's string distance measure and customize it to our needs. By means of the original distance measure we calculate the similarity between the extracted topic s_1 and the ontology topic s_2 based on their edit distance, i.e., the minimum number of point mutations required to change one topic string into the other. A point mutation involves either a change, an insertion or a deletion of characters. We aim to express topic similarity with a percentage rate, thus we adapted the original distance measure as shown in Equation 3.3.

$$sim_{Levenshtein(\%)}(s_1, s_2) = 1 - (d_{Levenshtein}(s_1, s_2) / \max(|s_1|, |s_2|)) \ . \tag{3.3}$$

where $\max(|s_1|, |s_2|)$ returns the number of characters of the string with the greatest length. The similarity function allows to further refine the initial score function defined in Equation 3.1 to

the function shown in Equation 3.4.

$$sc_{init}(u,t) = \omega_{contribType} \cdot r_{factor} \cdot sim_{Levenshtein(\%)} \ . \tag{3.4}$$

In the last step of score calculation, we address the issue regarding the different abstraction levels of topics. By leveraging the ontology's hierarchy, we can align expertise scores by propagating them from lower levels to higher levels. For score propagation we adopt the approach presented by Kay and Lum [Kay and Lum, 2005b]. Consequently, the final expertise score $sc(u,t)$ is calculated by means of the weighted sum of its children's scores as shown in Equation 3.5.

$$sc(u,t) = sc(u,t) + (1 - sc(u,t)) \frac{\sum_{child \in \mathcal{C}_p} sc_{init}(u, child)}{|\mathcal{C}_p|} \ . \tag{3.5}$$

where $\mathcal{C}_p$ is the set of children of topic t. The scores are propagated level by level starting with the lowest topics up to the hierarchy's root level.

3.2.4 Determining a Score's Confidence Level

Every modeling task intrinsically has a degree of uncertainty, so does the calculation method proposed in the previous section. Therefore, we compute for each expertise score a corresponding confidence level to further qualify the score. We propose two independent measures to estimate a score's confidence level. These measures are finally aggregated into the score's overall confidence level.

The first measure is built on the assumption that only top experts can accurately rate other top experts. Figure 3.8 illustrates the procedure that eventually delivers the score confidence levels displayed on the right side. Jane submitted various contributions related to *Java* and *WLAN*. These topics including their calculated expertise scores were assigned to her expertise model as shown on the left. Jane's contributions were rated by peers estimating the contributions' difficulty levels. As for calculating the confidence in Jane's expertise regarding *Java*, we follow the previously stated assumption and obtain the raters' expertise scores given the topic *Java* and build the average mean of these individual expertise scores. The higher the raters' expertise in *Java*, the higher the confidence in Jane's *Java* capability. Against this background, Equation 3.6 shows the calculation of topic t's confidence level based on raters' average expertise scores.

$$conf_{raters}(u,t) = \frac{1}{|\mathcal{R}_t|} \cdot \sum_{r \in \mathcal{R}_t} score(r,t) \ . \tag{3.6}$$

where $\mathcal{R}_t$ is the set of raters, which evaluated contributions of user u containing topic t.

The second confidence measure assumes that the more diverse the contributions of users are, the higher the confidence in their calculated expertise. For instance, the confidence in a calculated expertise score is higher if a user demonstrates this expertise in both a challenge and a solution rather than only submitting a challenge. Equation 3.7 formulates the calculation of this aspect utilizing the contribution weights determined in Section 3.2.2. The higher the contribution's weight is, the higher is the level of confidence.

$$conf_{diversity}(u,t) = \frac{\sum_{contrib \in \mathcal{C}_{u,t}} getWeight(contrib)}{\sum_{\omega \in \mathcal{W}} \omega} \ . \tag{3.7}$$

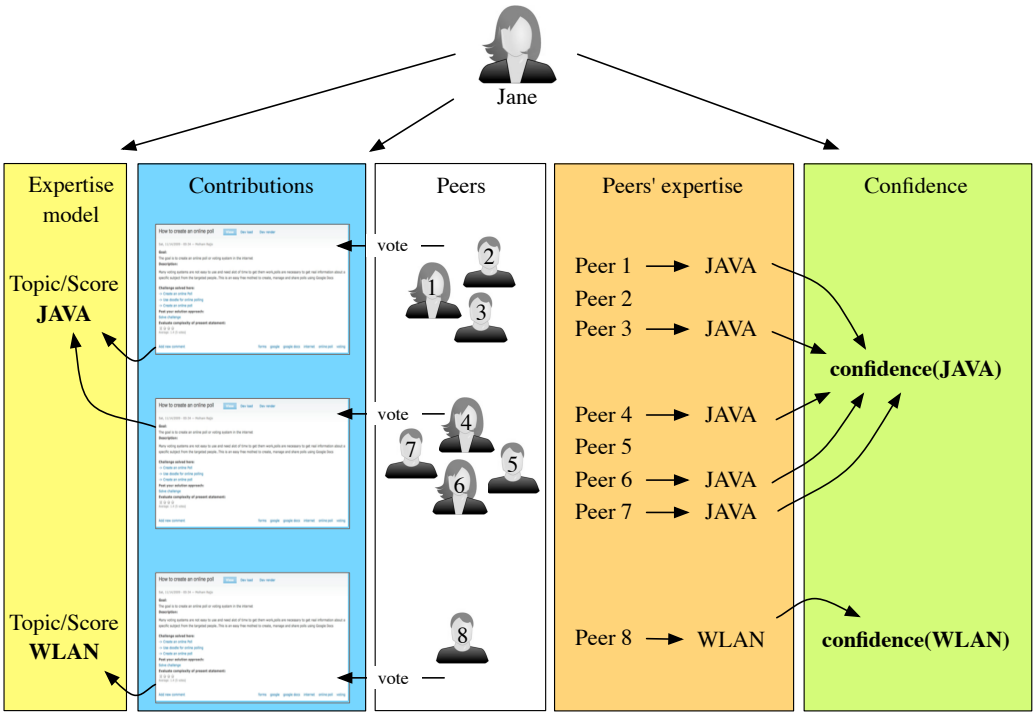


Figure 3.8: Confidence in Jane's expertise topics based on peers' expertise.

where $\mathcal{C}_{u,t}$ is the set of contributions submitted by user u and associated with topic t. $\mathcal{W}$ is the set of contributions weights.

As shown in Equation 3.8, we now combine the two confidence measures into the overall confidence level regarding the score calculated for topic t.

$$confidence(u,t) = \lambda \cdot conf_{raters}(u,t) + (1-\lambda) \cdot conf_{diversity}(u,t)\ . \tag{3.8}$$

where λ controls the balance between the independent confidence measures.

3.2.5 Evaluation

In this section, we conduct an experiment with 14 students to evaluate the second version of our Expertise Calculator. The students participating in the experiment are enrolled in a master program on computer science at the Vienna University of Technology. In the course of a tutorial about knowledge management, we started a four-week exercise dedicated to test score calculation. We encourage students to participate in a learning network and share their experience related to internet technologies. To make sure to collect sufficient data for evaluation, we asked participants to submit at least three challenges and three corresponding solutions regarding problems they recently faced in their daily routine, e.g., in certain exercises or during their work in case of part-time students. Participants were also encouraged to submit solutions to challenges

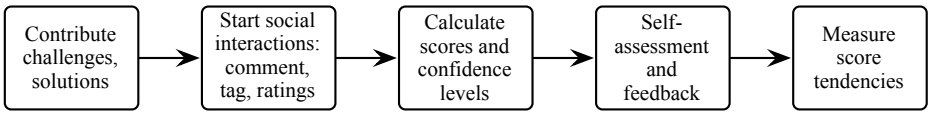


Figure 3.9: Evaluation procedure (Second experiment).

posed by other users. After submitting these initial contributions our participants took part in discussing their submissions, tagging them and evaluate contributions with their ratings. Figure 3.9 illustrates the steps taken during evaluation.

Once participants submitted a certain amount of contributions, they were able to invoke the calculation of their expertise model. We opened the expertise models to the participants for inspection and self-assessment. The left side in Figure 3.10 shows a snippet of the expertise model as presented to participants. Expertise topics are displayed by a tree view according to their relations given in the ontology. Participants can expand and collapse tree elements. Expertise topics are accompanied by their calculated scores and confidence levels. Besides numerical expertise scores, we used qualitative labels to support quick orientation and overview of expertise levels.

After four weeks, participants gave feedback regarding the scores contained in their expertise model as as well as on the possible potential they assume in the automatic calculation of expertise models. We then evaluated the tendencies of calculated expertise scores based on participants' self-assessments they collected during their feedback. Predicted expertise scores are either accurately calculated or under-/overestimate participants' actual performance. We refer to this deviation as score tendencies. For the current experiment, this measure represents the score accuracy of the proposed expertise calculation algorithm. In the third and last experiment as described in Chapter 6, we apply a much more detailed accuracy measure. However, at the moment we need to know if the algorithm is able to reliably predict scores on a coarse-grained level anyway. Thus, we examined score predictions whether they are calculated on a (1) *lower* (2) *equal* or (3) *higher* level by means of participants' self-assessments. The right side in Figure 3.10 depicts the feedback form as presented to participants. It shows a list of their expertise topics together with the algorithm's calculations. If participants feel to be more competent than the system believes, they would select the option *more*. In this particular case, we conclude that the system is underestimating the participant's actual performance.

Besides score tendencies, we evaluated if our algorithm captures the proper context of contributions, i.e., if we extract the appropriate topics to describe a contribution's actual subject matter. For that reason, participants could opt for *wrong* when self-assessing topics in their expertise models. In these cases, we interpret topics associated with such particular feedbacks as false positives, i.e., the algorithm assigned topics to the expertise model although they are not related to any of the participants' contributions. At least participants do not perceive them as such.

As for the first evaluation of the proposed confidence measure, we assume that a valid calculation of confidence levels will result in higher amounts of participants' feedback regarding the score tendency *exact* in contrast to the score tendencies *less* and *more*. Confidence levels for

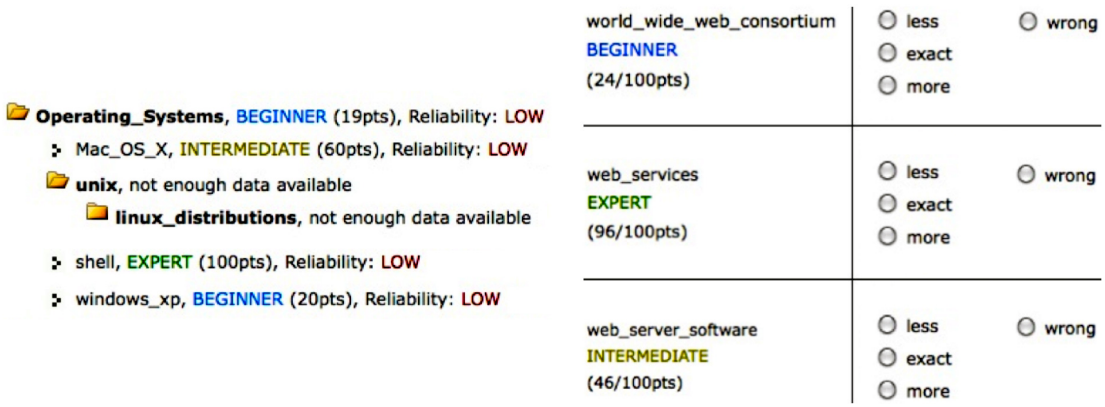


Figure 3.10: A user's expertise model (left) and self-assessment (right).

expertise topics marked as *wrong* will be excluded from evaluation.

We set the various parameters of the algorithm as follows. The contribution weight settings are taken from Table 3.3. We determined that a contribution has to be rated by at least two peers otherwise we will use default rating values, i.e., r_{factor} will be 1. Comments, tags and ratings can not be rated, thus these contribution types also rely on the default rating value r_{factor} set to 1. The topic similarity threshold regarding the mapping of extracted topics to ontology topics is set to $tr_{sim} = 90\%$. As for the aggregation of the two confidence sub-measures as shown in Equation 3.8, we set the factor balancing these measures to $\lambda = 0.7$. Thus, we assume that confidence being determined on the base of peer votes may be more valuable for a valid overall confidence level.

Prior calculation of expertise models showed that on average the amount of expertise topics contained in participants' models is relatively high (above 90 topics per model). We do not want to annoy participants by displaying unacceptable long lists of expertise topics for self-assessment. This may lead to the effect that some participants just *click through* the list rather than reflecting their expertise given the calculated topics. Hence, we only displayed topics with predicted scores exceeding 20 points (maximum score: 100 points) during self-assessment.

Results and Findings

Table 3.4 shows the data we collected during our four-week experiment. The amount of submitted solutions is slightly higher than that of challenges implying that some challenges were solved by more than one participant. We actually thought to observe more intensive discussion reflected by a higher amount of comments. On average, we calculated 93 expertise scores per model where 18 expertise scores were displayed to participants for self-assessment. Figure 3.11 displays the results of participants' self-assessments. Participants felt accurately assessed in 134 of 246 total score predictions which amounts to an accuracy rate of 54%. As for the rest of the calculated scores, we observe that the algorithm mostly underestimated participants' expertise. More specifically, this is true in 80% of deviations excluding topics falsely associated with

Contributions submitted	
Challenges	59
Solutions	78
Comments	88
Tags	359
Ratings	243
Total	827
Total scores calculated	1301
Scores self-assessed	246

Table 3.4: Data statistics

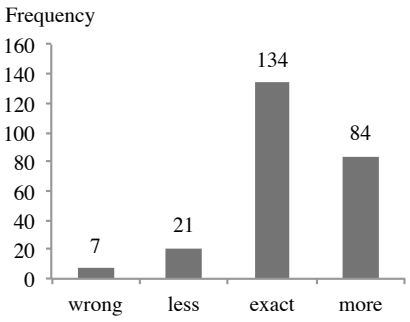


Figure 3.11: Feedback results.

participants.

We consider following reasons for this algorithm behavior. First of all, according to [Dunning et al., 2004] people usually tend to overestimate themselves. This is especially true for poor performers that lack insight into their shortcomings even on the promise to receive incentives the harder they work on their self-assessments [Ehrlinger et al., 2008]. Despite the fact that students participating in advanced courses are said to perform significantly better than students from basic courses [Falchikov and Boud, 1989] - and our participants are represented by master students - we sort of anticipated the trend to underestimation. Thus, we asked our participants to orally present their contributions in a closing session of our tutorial and let them argue why they self-assessed the way they did. Two human experts followed these presentations and provided their estimates. On the one side, these estimates considered the participants' expertise levels as perceived based on their presentation performance. On the other side, experts considered also the topics generated to the participants expertise models as well as their self-assessments. The final presentation session lasted two hours in total. In this time frame we had 14 participants presenting their contributions including occasional discussions between experts and presenters to clarify provided self-assessments. In summary, given this quite compact session, we observed that experts tend to agree with participants self-assessments. However, experts said that it was hard to follow several presenters in such a short time frame and to evaluate their performance by associating expertise score with presenters' topics as they speak. To conclude, given the expert assessments suggesting that self-assessments are mostly viable, it seems that underestimation is not primarily caused by overconfident self-assessments.

Another reason for underestimation may be that the algorithm only considers the highest weighted contribution to determine a topic's final expertise score. At the moment, predicted scores obtained from lower weighted contributions are discarded. Furthermore, we intentionally set the default rating values to very low levels. This pessimistic attitude may have contributed to underestimation as well. Thus, we need to examine different values for default ratings in subsequent experiments.

A shortcoming of the current experiment is that on average, we only collected 1.8 peer ratings for either challenges or solutions. Hence, most expertise scores were calculated based

on default rating values (pessimistic approach, low values). This is possibly another reason why participants felt mostly underestimated by the system. Insufficient rating data has also influenced the calculation of confidence levels. Since we emphasized the sub-measure relying on peer votes by means of the balance factor λ, the overall confidence levels show consistently low values as shown in the expertise model in Figure 3.10.

As shown in Figure 3.11, 7 of 246 displayed expertise topics were identified as false positives. More specific, this means that 17 of 18 topics were properly assigned to participants' expertise models. This is a very promising figure which indicates that the text mining web services we integrated for score calculation are well suited to extract candidate topics.

Once participants submitted their self-assessment, they reported their ideas regarding the potential of automatic expertise modeling. Participants said that they can imagine to use their generated expertise model as *a personal knowledge base* they can regularly reflect on. In addition, they suggest to integrate the proposed algorithm with the university's existing course register in order to recommend future courses based on their personal expertise. Others think of using expertise models as the fundament of a *competence marketplace* where companies and students get in touch regarding different kinds of collaboration. Moreover, participants guess that our method can facilitate the gathering of students into learning groups.

3.2.6 Summary and Next Steps

In the present section we proposed a method to calculate absolute expertise scores of users based on their contributions and social interactions in a learning network. We systematically determined weights for the various types of contributions building the base for expertise predictions. Our algorithm computes expertise scores as well as confidence levels to express the reliability of scores.

We conducted an experiment with 14 university students to evaluate score accuracy, to identify topics falsely assigned to expertise models and to test participants acceptance of automatic expertise modeling. We found that 97% of topics were identified properly and 54% of competence scores were accurately calculated compared to participants self-assessments. Most of the scores that were not exactly calculated showed the trend to underestimate participants. As for testing the calculation of confidence levels, we did not collect enough data for a profound interpretation and thus need to rethink the study design for future experiments. Responses from participants' feedback indicate that basically expertise scores are perceived to be useful for recommending future courses as well as for the formation of learning groups.

Based on the present results, we are in the position to redesign and adjust our method for further, more detailed evaluation. More specifically, we aim to test different contribution weight settings as well as default rating values. The adoption of a more sophisticated approach for score propagation may improve score accuracy as well. As for a profound evaluation of predicted scores, we need to collect user self-assessments on a fine-grained scale. It seems obvious that the quality of self-assessment improves once we expose the ontology to the users. Thus, in the next section, we introduce an interface for user self-assessment facilitating the navigation through a competence ontology, the assignment of fine-grained scores to expertise topics and an extensive view of the expertise model with various options to seek details regarding a certain topic.

3.3 A User Interface for Overlay Expertise Models

In this section, we aim to design an interface for expertise models consisting of a subset of topics from a domain ontology. By means of this interface, we collect users' expertise self-assessments on a point-wise scale. Such fine-grained self-assessments allow to explore the algorithm's score calculation behavior on a more detailed level than just considering score tendencies as realized in the previous section. Besides, users can navigate through the ontology and inspect the various expertise topics as well as their relationships. Bull and Kay [Bull and Kay, 2010] describe the trend to open profiles to users in the field of intelligent tutoring systems. Giving learners greater control over their learner models may aid learning by supporting learners' self-reflection and it can help them planning future learning activities. Thus, we assume that exploring the domain knowledge provides users not only with a better understanding of the domain but it might also increase users' self-assessment quality. For instance, users can scrutinize a certain expertise score by exploring its relationship with adjacent topics.

Competence ontologies are mostly very large in both breath and depth. Navigating such ontologies as well as presenting expertise models based on these ontologies constitute major challenges in the design of user interfaces [Crowder et al., 2009] [Bakalov et al., 2010]. As for navigation, a conventional tree view of topics is cumbersome to handle. A user starts at the top of the tree and navigates to the bottom. If navigation leads to a path in which users are not interested, they must go back all the way to the point where they started. Regarding the presentation of an expertise model, users may quickly lose their sense of the big picture as more topics are available in the model. Thus, we aim to address the following questions in order facilitate expertise self-assessment:

- How can we support users in navigating a large competence ontology, selecting ontology topics and associate expertise score with these topics?
- How can we achieve a useful presentation of expertise models?

In answering these questions, we propose a user interface comprising (1) a navigation and (2) a presentation component. The navigation component supports users in selecting topics from the competence ontology, associate an expertise score with selected topics and finally store them to the users' expertise models. On the other side, the presentation component aims to provide a comprehensive view of users' expertise topics as well as several options to adapt this view to users' personal preferences.

The user interface will consist of several elements. We evaluate the usability of the interface on a combination of these elements. Therefore, we conduct an independent usability study to explore the possible benefits of the interface for its later use in experimenting with our score calculation algorithm. The study takes place with 19 master students in the course of a tutorial held at our university. The participants will use the interface to self-assess their expertise in the domain of internet technologies. Based on the results of this usability study, we devise the interface which is used for the thesis' final experiment, confer Chapter 6.

3.3.1 Inspecting Large Ontologies

We reviewed research works that approach the challenge of visualizing and navigating large ontologies. A survey on ontology visualization techniques reports that ontologies are in most cases structured as hierarchies [Katifori et al., 2007]. Furthermore, ontologies in many domains tend to be quite large and complex, which makes them difficult to explore and display [Storey et al., 2001]. The *Visual Information Seeking Mantra* tackles the problem of representing large data in three steps including overview first, then zoom and filter while showing details on demand [Shneiderman, 2002]. When dealing with large unknown data, the concept of *Information Scents* [Pirolli, 2007] and its application in the form of scented widgets [Willett et al., 2007] improves traditional user interface elements. Information scents provide users with more context and help them to accomplish tasks more efficiently. Crowder et al. [Crowder et al., 2009] make use of content dependent filtering, an autocompletion text box and partial segments using drop-down lists for ontology navigation.

With regards to cognitive support of ontology navigation, d'Entremont and Storey [d'Entremont and Storey, 2009] suggest principles to provide overview and context, reduce the complexity, indicate points of interest and support incremental exploration. They further introduce a plugin for the ontology editor Protégé using these principles in providing *Visual Orientation Cues* for user relevant content. Jambalaya [Storey et al., 2001] is a user interface also based on Protégé, which employs the concept of nested interchangeable views to allow a user to explore multiple perspectives of information at different levels of abstraction. Bakalov et al. [Bakalov et al., 2010] present a rich-interaction interface enabling users to inspect and alter their user profiles. The interface provides an overview of terms representing user interests, allows for zooming/filtering and displays additional term information like a term's relationship with other terms.

To the best of our knowledge, none of the reviewed approaches supports an ontology navigation that allows users to reflect and compare scores amongst topics in an ontology. In addition, the surveyed approaches do not include a clear procedure for the assignment of scores to ontology topics.

3.3.2 System Architecture

Figure 3.12 shows the architecture of our prototype implementation that is based on a three-tier model commonly used for web applications. We iteratively developed the interface elements into more advanced ones for ontology navigation, user self-assessment and the presentation of expertise models. As for navigation, the respective topics are retrieved from the ontology on demand. Thus, the growth of the competence ontology does not affect the interface's performance. For retrieving ontology topics, AJAX-methods effectively take care of providing real-time behavior to users. Once users have assigned expertise topics to their models, the entire model is transferred to the server for data storage.

The right side in Figure 3.12 displays a snippet of the competence ontology as proposed in Section 3.4. An ontology instance describes a user who is competent in one or more topics where each topic is associated with the user's expertise level. Some of the topics are related with synonyms. We leverage these synonyms for the autocompletion feature supporting ontology navigation as presented in the following section.

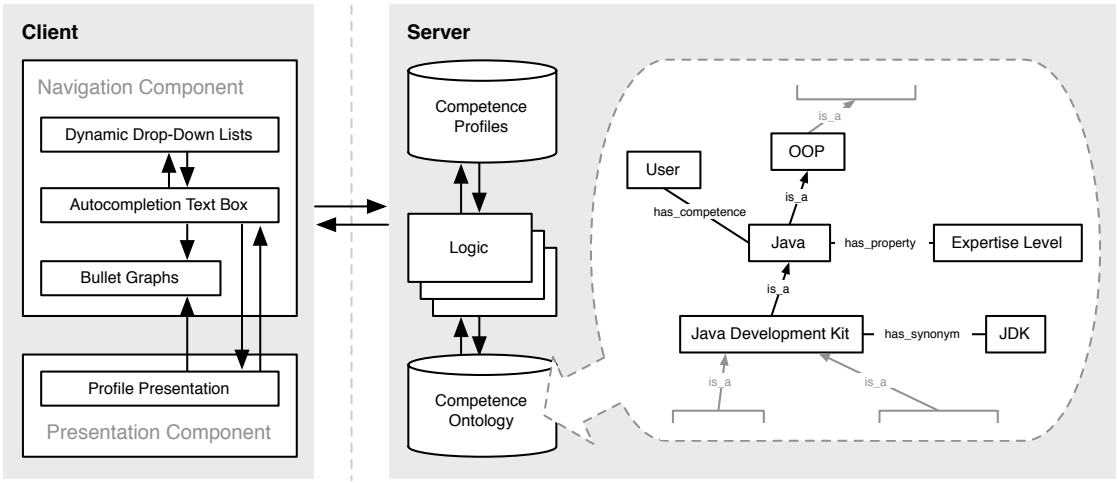


Figure 3.12: System architecture.

3.3.3 Navigation Component

In this section, we assemble the elements that allow users (1) to navigate the competence ontology for the purpose of selecting certain expertise topics and (2) to assign score to selected topics.

Versatile Ontology Navigation

Crowder et al. [Crowder et al., 2009] present autocompletion text boxes and interconnected drop-down lists as means for ontology navigation. We adopt these basic ideas for the design of our interface.

As for autocompletion, users enter words into the text box by which they want to query the topic space. Thereupon, the underlying ontology is queried for topics that match the user's input at best as shown on the top left in Figure 3.13. The query string will be enhanced with wildcards and the result set is further expanded with the topics' descendants obtained from the ontology tree. The resulting list is directly displayed below the text box. We add to each topic in the result list its corresponding expertise score gained from users' self-assessments. Finally, users select the desired topic from the list and continue with assigning their expertise levels as illustrated at the bottom in Figure 3.13.

Besides using word queries for exploring ontology topics, we consider the use of interconnected drop-down lists for navigation. Traditional drop-down elements display available topics in a flat list independent of any relationships between these topics. This implies that we can not display any structural information between topics to users. In contrast, by means of interconnected drop-down lists we can manage the display of the ontology's hierarchy levels, i.e., each level is represented by its own drop-down list. As depicted on the top right in Figure 3.13, users start navigating the ontology by select a topic from the first hierarchy level of the ontology tree.

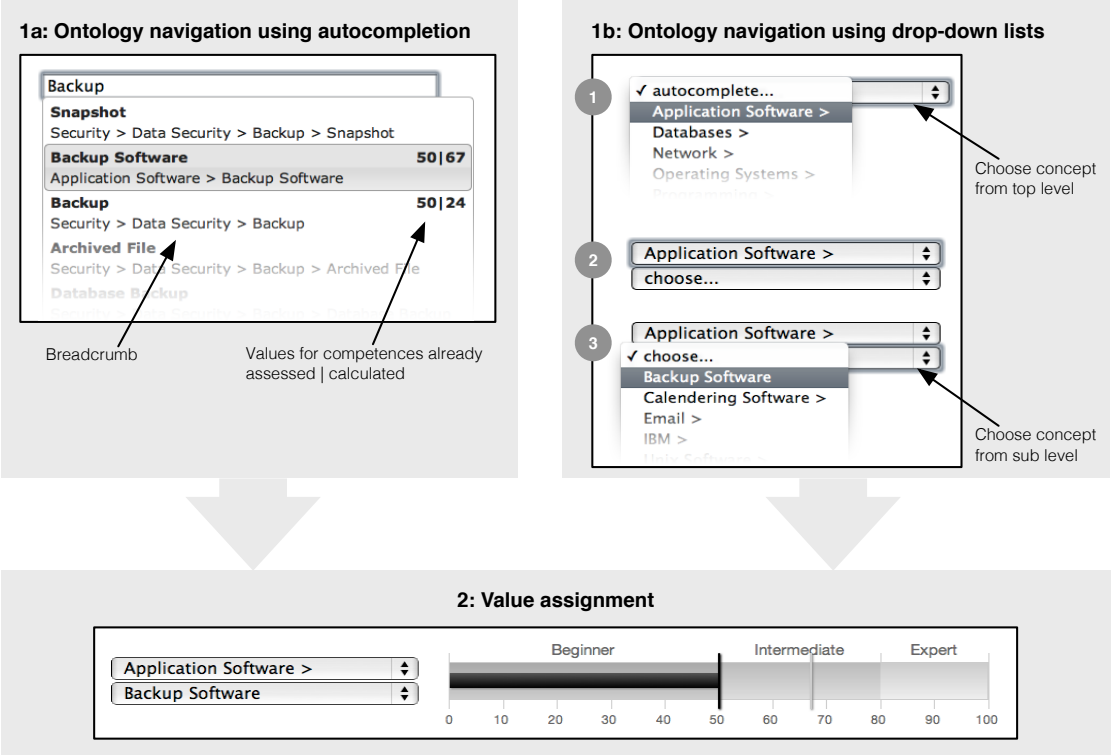


Figure 3.13: Two ways of topic selection both leading to score assignment.

Once a topic from the current level is selected, another drop-down list appears comprising all topics from the subsequent lower levels and so forth. Each time a topic is selected, the area for score assignment is updated and allows users to specify their expertise level.

We want to provide users with versatile way to navigate the ontology. Hence, we integrate the autocompletion text box with the interconnected drop-down lists. This comes with several benefits for both novice and expert users. According to Ernst et al. [Ernst et al., 2005], a top-down approach especially helps users unfamiliar with the ontology. On the other hand, advanced users may want to directly dig into the ontology by selecting a particular topic they assume or they know it exists. By means of this combined approach, users can adapt the way to explore the ontology to their preferences.

The area for expertise score assignment is located right from the elements used for navigation as shown at the bottom in Figure 3.13. We incorporate a graphical element known as *Bullet Graph* to represent scores as well as to alter them. This particular element is described in the following section.

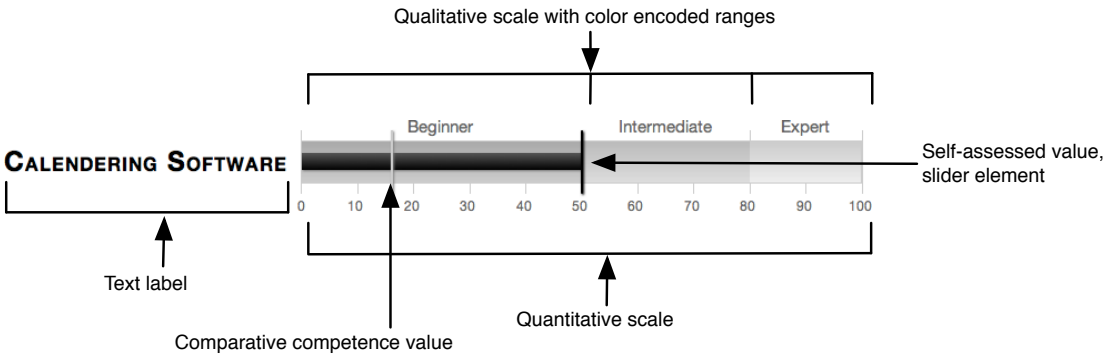


Figure 3.14: Adapted bullet graph for competence self-assessment.

3.3.4 Expertise Score Assignment

During self-assessment, users associate scores ranging from 0 to 100 points with expertise topics. To graphically support this task, we introduce an interface element that is based on *Bullet Graphs* [Few, 2006]. Basically, a bullet graph consists of a content box, which represents a qualitative scale, a quantitative scale and a bar representing a certain value. Additionally, a cross bar can be used to indicate a comparative value that qualifies the actual value displayed by the bar element. Originally, a bullet graph is not intended to be used in a user interface and much less as an interactive element. Therefore, we implemented an interactive bullet graph element based on widgets that allows users to drag the bar to the desired score value representing their expertise. Furthermore, we added labels to describe the fields of the qualitative scale. The comparative value can be used for different reasons, e.g., to show executives' estimates about their employees' expertise. Figure 3.14 depicts the bullet graph including the changes we made.

3.3.5 Presentation Component

In order to display users' expertise models, we propose a table which includes the topics together with their expertise scores as well as the relation amongst topics. Since the competence ontology represents mainly hierarchical relations, we make use of an hierarchical approach for models' presentations using a traditional HTML table. Figure 3.15 illustrates the view of a user's expertise model.

The traditional HTML table was tuned as follows. We integrated the visual information seeking mantra as presented by Shneiderman [Shneiderman, 2002] as well as the idea of information scented widgets [Willett et al., 2007]. Moreover, we consider the principles of cognitive support for ontology navigation by means of visual cues [d'Entremont and Storey, 2009]. With the help of visual cues we highlighted the hierarchical relationship between topics in the expertise model, i.e., we set the intensity of the background color for each topic according to its depth in the ontology tree. A tooltip at the left border of each row shows the path in the ontology leading to the topic in reverse order. For the same purpose, we indented the labels of topics after their path sequence leading to the ontology root. In order to prevent confusion amongst

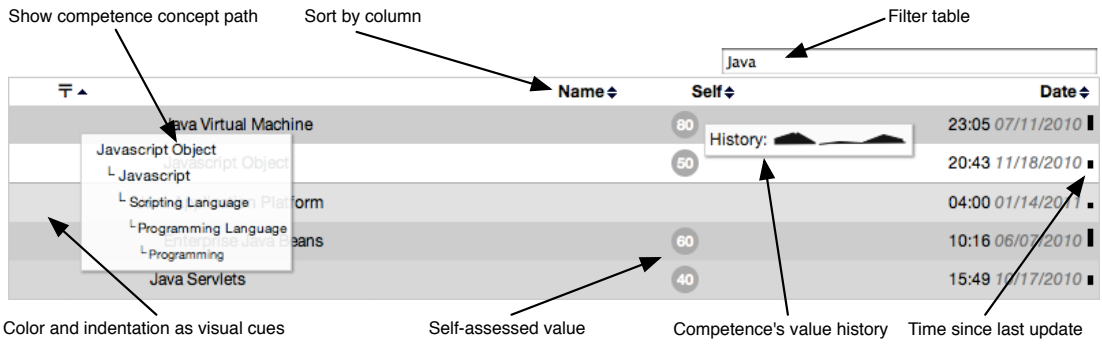


Figure 3.15: Viewing the expertise model.

users regarding adjacent topics in the model that are located on equal levels in the ontology tree but having different ancestors, we separated the respective two rows (topics) with a thicker grey line. Expertise scores are displayed by circled numbers. When moving the mouse over a score, a graphical tooltip visualizes how the value changed over time by means of a filled line chart. The last column of the model table refers to the date of the last alteration together with a bar chart representing the time passed since the last update. Users can personalize their model view by filtering and sorting options. A filter text box allows users to filter topics towards a string in a topic's full path. The users can also sort each column to their personal preferences.

The components for navigation and presentation are integrated and appear on the same screen. That means, users can search for new topics, assign expertise scores and inspect their expertise model simultaneously. The functionalities of either components are linked together. Selecting a topic from the table causes the navigation component to refresh and to display the selected topic.

3.3.6 Testing Interface Usability

As already indicated in the beginning of this section, we conduct an independent usability study to evaluate the various elements of the interface. Given the results of this study, we decide which elements we use to design the competence cockpit suitable for our final experiment as presented in Chapter 6. The usability study is mainly focused on testing user satisfaction by means of quantitative feedback. In addition to that we also provided room for qualitative feedback. All user interactions were logged in order to interpret user behavior and analyze problems that might occur during user testing.

More specific, to evaluate usefulness and satisfaction, we conducted a usability study with 19 master students at university. When speaking about usability, we measure user satisfaction and investigate how efficient users may perform the self-assessment task using our interface. The study took 22 days and was implemented in the course of a tutorial on knowledge management. The service was published on the web, thus participants could easily access the interface as often and as long as they wanted.

We asked participants to build their expertise models by using the proposed interface. Consequently, they had to navigate through the competence ontology, select certain topics and store their self-assessment to the model. We provided a short user guide describing the main features of the interface, however, we did not recommend particular strategies on how to use the interface.

At the end of the study, students had to fill out a questionnaire. Given the responses, we aimed to interpret the following questions:

1. How satisfied are users with navigating the competence ontology and topic selection?
2. How useful is the presentation of user self-assessments using bullet graphs?
3. How useful is the presentation of a user's expertise profile based on a table displaying expertise scores as well as the relations amongst topics?
4. How useful are sorting and filter functions to adapt the model view?

Besides, participants were asked to give their opinion about likes and dislikes of the user interface. The interpretation of open question feedbacks might reveal further details on how the navigation and presentation of competences can be improved.

Results and Findings

We collected 1267 self-assessments in total. Figure 3.16 shows the results regarding the quantitative part of our questionnaire. The majority of participants was mostly satisfied with the interface for ontology navigation and perceived the bullet graph as useful to display expertise scores. As for the presentation of expertise models, participants were predominantly convinced of its usefulness and have also used sorting and filtering functions to customize the model view. The response to open questions mainly complies with the results from quantitative feedback.

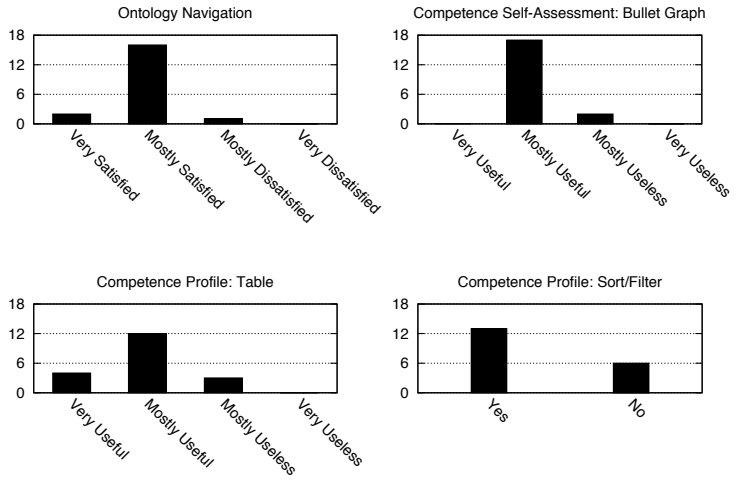


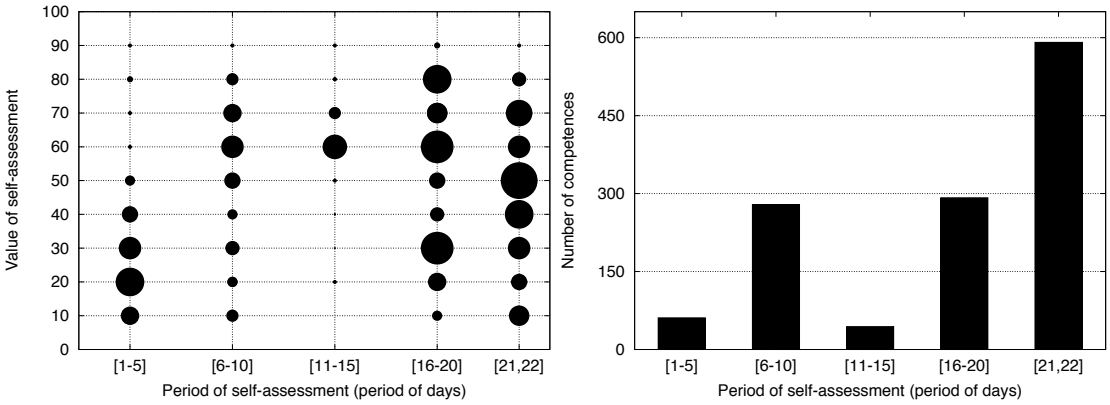
Figure 3.16: Questionnaire results regarding usability and usefulness.

However, some participants said that visual navigation cues used in the model view were not clear to them. Others appreciated the extensive use of AJAX for both navigation and model presentation.

Figure 3.17 puts participants' self-assessments on a timeline. We aggregated the data in time clusters to better show the total number of self-assessments. The size of the dots in Figure 3.17a stands for the number of topics related to a certain expertise score. We observe that participants did not use minimum or maximum scores. We did not expected that participants would not use zero-scores as they were not asked to report on expertise they do not have. As for the maximum score, Figure 3.17a confirms the well-known phenomenon that experts make no use of maximum scores when estimating their personal expertise. It is said that experts know better than less competent people that there is always something else they do not know.

Figure 3.17a as well as Figure 3.17b show that the number of self-assessments increases over the course of the study. Is this enough evidence to prove the interface to be an efficient support for self-assessment? The rise of self-assessments may indicate that the more topics are assessed, the faster the subsequent self-assessments were performed. This interpretation may be supported by the fact that only one task was given to the participants at the beginning of the study. From this point on, participants were free to enter self-assessments in the given time period and they were not asked to process further tasks.

We can rule out a possible bias that participants assessed more topics in favor of getting better grades since they were not required to finish the task with a model containing a certain number of topics. However, there might be another bias causing an increase of topics at the end of the study cycle. That is, participants might have been curious in the first place about how the interface is built up and just started to explore its features. While attending several courses during the study term, participants may have set up a plan on when to finish which task for which course. Such a plan may have led to a larger workload at the end and thus result in an increased activity regarding certain courses. Another limitation is that participants are to some extent familiar with the domain and the notion of ontologies.



(a) Number of topics per expertise score

(b) Total number of topics

Figure 3.17: Analyzing log data to measure efficiency.

Assuming that our results are not significantly biased by previous issues, they suggest that our interface helps to maintain the overview of expertise topics since this would definitely be a challenge the higher the number of topics in expertise models. At the current stage, we can not claim that the interface is a means to efficiently support self-assessment. This issue has to be addressed in future works.

3.3.7 The Expertise Cockpit

We tested various user interface elements in the previous section that may facilitate user self-assessment. Based on these results, we now devise an interface suitable for a detailed evaluation of expertise scores. We found in our previous experiment that some interface elements were rarely or even not used at all. This is especially true for the elements representing time information. Due to the short time frame participants worked with the interface, it makes little sense to display the history of self-assessed scores. That is just because there is not any meaningful history to display. This is quite similar regarding the last updates of expertise topics. Even though this temporal information can make sense on a larger time scale, we will not consider it for the design of the Expertise Cockpit in favor of a clear user interface.

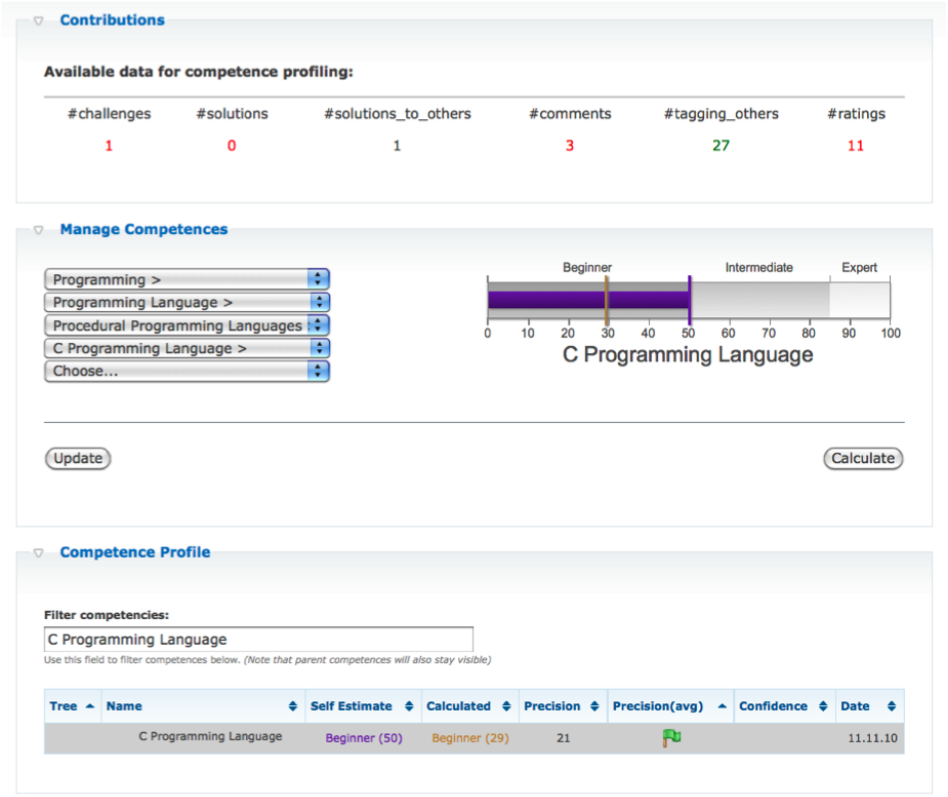


Figure 3.18: Expertise Cockpit including an overview of the user's contributions.

The main information we need to add to the user interface concerns calculated expertise and confidence. Figure 3.18 illustrates the interface as we use it for further evaluation. On top, users find a list of their contributions representing the evidence upon the algorithm calculated their expertise model. The middle part shows the navigation component that is enhanced with a button *Calculate* to initiate score calculation. Participants can add or update topics to the expertise model by means of the button *Update*. The orange-colored cross bar in the bullet graph shows the system's calculated score for the given topic. The bottom of Figure 3.18 displays the presentation component currently showing one topic that already added to the expertise model. We enhanced the topic's information with its calculated score and confidence level. In addition, we calculated precision (absolute deviation of calculated scores and self-assessments) and precision average (average score deviation of topics contained in the subtree of a topic including the topic's own deviation). Precision average values are represented with colored flags whereby green flags concern score deviations up to 30 points, yellow flags up to 50 points and red flags up to 100 points. As for the display confidence levels, we also use labels, namely, no label for levels up to 20%, label *weak* up to 50%, *moderate* up to 80% and *strong* up to 100%.

3.3.8 Summary

Given the problem that large competence ontologies are difficult to navigate, we proposed an integrated user interface allowing users to easily find expertise topics due to various ways of ontology navigation. We utilized bullet graphs for expertise score assignment, which offer a quantitative as well as a qualitative scale to display expertise scores. We further introduced a model view displaying self-assessed topics and their relations to adjacent topics. The proposed components for ontology navigation and model presentation are functionally linked together, which allows users to approach self-assessment in various ways.

The results of our study conducted with 19 master students indicate that participants were mostly satisfied with navigating the competence ontology. They perceived the bullet graph as useful and were also satisfied with the presentation of expertise topics as well as with the options to customize their model view. We were not able to prove whether the proposed interface provides efficient self-assessment, i.e., speeding up the process of self-assessment. Based on these results, we built an Expertise Cockpit allowing us to elicit fine-grained expertise self-assessments to evaluate the algorithm's performance more thoroughly.

CHAPTER 4

Spreading Expertise Scores in Ontology Overlay Models

The second version of the proposed Expertise Calculator utilizes a simple propagation method to align expertise scores in users' expertise models, confer Equation 3.5. We expect that a more sophisticated approach exploiting the hierarchy levels of the competence ontology may deliver more valid results. [Kay and Lum, 2005c] suggest the use of lightweight ontologies in favor of saving expert resources to build relatively complete ontologies. They further conclude that simpler inference algorithms suffice for reasoning about topics in the area of adaptive educational systems. Such reasoning algorithms fight sparsity and increase the precision of user models. Thus, our goal is not to enhance our ontology's expressiveness by introducing new types of relations. Instead, we explore a new way to extensively use the information given by the lightweight ontology as well as by users' expertise scores.

In this chapter, we devise a novel algorithm using spreading activation to propagate expertise scores in an overlay model. Thereby, we aim to answer the following research question:

Based on a user's expertise in topic X, how much does the user know about topic Y?

Spreading activation is a technique to process networked data like topics in an ontology. The basic idea is to transfer information between the topics in the network. Following that, we spread users' expertise scores through the network structure of the domain ontology. The novel aspects of our algorithm are:

1. *Coefficient* α is used to alter a topic's while being activated. Thus, it ensures the alignment between a topic and its subtopics.

2. We introduce *relative depth scaling* for calculating relation weights representing the similarity between topics. These weights are used for propagation, for pre-adjusting activation and for comparing calculated scores with the expert standard.

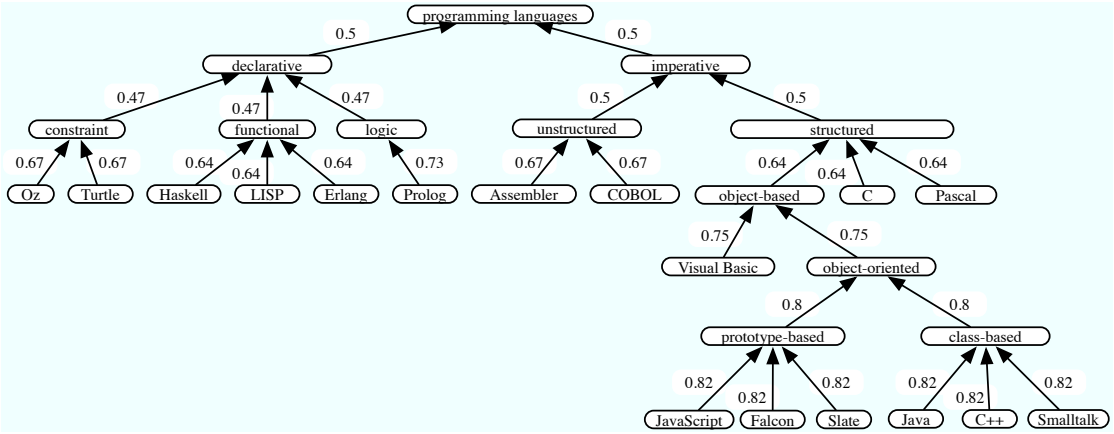


Figure 4.1: A domain ontology modeling topics and their similarities.

We compare our novel method with a baseline approach represented by the propagation method we already incorporated in the previous version of the Expertise Calculator. This chapter is organized as follows. Section 4.1 describes the details of both the baseline and the novel approach. We devise various scenarios to evaluate and compare the performance of either approaches. Section 5.2 presents the evaluation results. We summarize our findings in Section 4.3.

4.1 Expertise Score Propagation

A lot of research work has been done on hierarchical ontologies. This is not surprising since most ontologies are made of *is-a* relationships [Schickel-Zuber and Faltings, 2007]. Many adaptive systems claim to utilize ontologies. In fact, they use taxonomies that can be considered as lightweight ontologies based on relations like *is-a*, *part-of* or *similarity* [Brusilovsky and Millán, 2007]. Figure 4.1 depicts a simple ontology modeling programming languages and programming paradigms. We built this ontology by hand based on descriptions from Wikipedia. The links represent the similarities of topics ranging from 0 to 1. All scores calculated in this chapter are based on this ontology.

Spreading activation is made of a sequence of iterations [Crestani, 1997]. One iteration follows the other until a certain termination condition occurs. Each iteration is made of one or more pulses, where a pulse represents the process of spreading activation from one single topic to another. A pulse consists of a pre-adjustment and post-adjustment phase (see Figure 4.2), which allow to attenuate previous pulses and control activation. We apply spreading activation in a hierarchical ontology. This implies that activation is only allowed on the shortest path leading to the root topic. An iteration consists of pulses that propagate activation starting from lower hierarchy levels upwards. Before any activation starts, initially activated topics (see Table 4.1) will be sorted in descending order by their hierarchy levels. Topics not being activated will

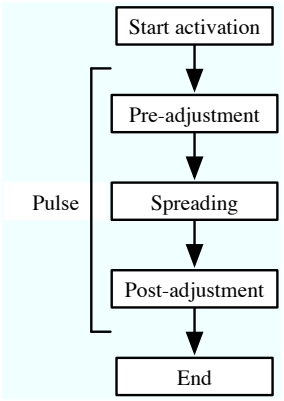


Figure 4.2: Steps of activating a topic.

receive the activation level 0. The first iteration starts with propagating expertise scores on the lowest level. This process terminates at the root level.

In case a topic about being activated has already an activation level greater than 0 (this happens when initial activation concerns topics on different hierarchy levels), we make use of the pre-adjustment phase to prevent possible distortion of activation levels. For instance, in scenario 3 the topic *object-oriented* has an initial score and will also be activated by topic *Smalltalk*.

4.1.1 Baseline Approach

[Kay and Lum, 2005b] propose an algorithm to infer the scores of higher level topics from topics on lower levels where direct evidence is available. We already adopted their approach for the second version of our Expertise Calculator in Section 3.2.3. We are now interested if the novel score propagation method we present in Section 4.1.3 performs better than the approach we used so far. Thus, we defined the score propagation according to [Kay and Lum, 2005b] as the baseline approach.

4.1.2 Semantic Similarity

Prior to the introduction of the novel approach, we briefly outline the results of a literature survey we conducted on semantic similarity measures in the context of ontologies. The goal of this survey was to get a notion about available options for calculating relation weights in hierarchically structured ontologies. We seeked for a weighting method that is more sophisticated than the one used in the baseline approach. However, we also explored measures operating on non-hierarchical ontologies [Maguitman et al., 2005] for possible future work.

In literature, similarity regarding ontologies is interpreted twofold. On the one hand, there exist measures to calculate similarity between single ontologies [Maedche and Staab, 2002] [Doan et al., 2003]. On the other hand, similarity measures focus on the similarity between topics within a single ontology. We aim to adopt an approach from the latter ones. In addition, “similarity” is not equal to “relatedness”. That is because semantic similarity is a “special case of

semantic relatedness" [Resnik, 1995] and thus only considers topic relationships of the type *is-a* (hyponymy). For example, the topics *flower* and *plant pot* are strongly related but interpreted as less similar. Therefore, literature provides measures regarding semantic relatedness [Mazuel and Sabouret, 2008] [Hirst and St-Onge, 1998] as well as semantic similarity of topics. The characteristics of the latter are described in the following.

Basically, similarity measures aim to estimate a score for a pair of nodes by exploiting some information sources. Hence, these measures can be classified based on the source of information they exploit. We distinguish mainly edge-based, node-based and hybrid similarity measures.

Methods focussing on the edges of the ontology [Rada et al., 1989] [Resnik, 1995] constitute the simplest and most intuitive measures. They just count the edges on the shortest path connecting two nodes and assume that the lower the edge count (the distance), the higher the similarity of these nodes. This kind of approaches have two major drawbacks. First, they require a consistent and rich ontology to work properly, i.e., an ontology where the leap between general nodes and that between specific ones have practically the semantic distance. And secondly, edge-based approaches consider the distance uniform on all edges, i.e., the distance between two directly related nodes is always equal, no matter where they reside in the ontology nor how many nodes are related to them. Later on, edge-based measures were improved by integrating information about the depth of nodes in the hierarchy [Wu and Palmer, 1994] [Sussna, 1993] .

Node-based similarity measures are primarily based on the notion of "Information Content" (IC) [Shannon, 2001] associating probabilities to each node in the ontology based on word occurrences calculated in large corpora. These probabilities are aggregated level by level from more specific nodes to more general ones. Hence, IC is steadily decreasing as we move up the ontology to the root level. In fact, the root node has the maximum word frequency count, since it represents the word counts of every other node in the ontology tree. [Resnik, 1995] was the first to adopt this idea for similarity measurement where the similarity between two nodes is the information content of their lowest common ancestor. The shortcoming when using IC for similarity calculation is that it requires a time-consuming analysis of corpora in advance and that IC scores may depend on the type of the underlying corpora as well. Other node-based measures adopt the approach of feature similarity [Tversky, 1977] where similarity of nodes is calculated on the features they share. In particular, this metric compares two nodes' vectors in terms of the number of exact feature matches. More recently, [Pirrò, 2009] presented an approach combining the notion of IC with feature similarity.

Hybrid similarity approaches [Jiang and Conrath, 1997] [Othman et al., 2008] represent a combination of the aforementioned measures. For instance, the measure proposed by [Jiang and Conrath, 1997] integrates the idea of edge-based methods with the nodes' information content.

4.1.3 Novel Approach

In this section, we propose a novel algorithm for propagating expertise scores using constrained spreading activation. By means of relative depth scaling as introduced by [Sussna, 1993], we assign weights to the ontology's relations. Equation 4.1 shows activation, where topic p is activated by topic c. The overall score $S(p)$ is the sum of scores received from activated subtopics.

Scores are propagated level by level starting with the lowest activated topics up to the root.

$$S(p) = \alpha \cdot S(p) + \frac{\sum_{c \in \mathcal{C}_p} S(c) \cdot \omega_{Sussna(p,c)}}{n_{ExpertStandard}(p)} \cdot \gamma \ . \tag{4.1}$$

where α is a coefficient for generalization and $\omega_{Sussna(p,c)}$ the weight of the link connecting topic p and c. The decay factor γ controls the intensity of activation. In the following, we provide a detailed description of each term in Equation 4.1.

Relation Weights

In our context, a relation linking two topics represents the similarity between these topics. Based on the literature survey in Section 4.1.2, we adopt the edge-based distance measure proposed by Sussna [Sussna, 1993] for calculating relation weights. Our decision is grounded on following reasons: First of all, we have no further information about topics on hand except their labels and scores. This rules out IC-related similarity measures. Calculating meaningful IC scores is practically not possible because of the small size of corpora we are dealing with. Secondly, Sussna supports our idea to integrate additional relation types in future work and is designed to work on hierarchies. And lastly, this measure considers the depth of a topic as well as the number of subtopics while calculating similarity and thus represents more finesse than the weighting used in the baseline approach.

Sussna interprets the relation between two topics by means of two inverse relations. Each of the two relations has its own weight. Basically, these weights are calculated based on the links leaving the respective topic. Our ontology does not support multi-inheritance, i.e., subtopics have only one topic they belong to. Therefore, the directed relations from subtopics to their topics have always equal weight. In contrast to that, directed relations from a topic to their immediate subtopics change according to the number of subtopics. Equation 4.3 shows the calculation of a topic's directed relation to its subtopics.

$$\omega_{Sussna(p,c)} = 1 - \frac{1 + \omega(p,c)}{2 \cdot depth \cdot distance_{max}} \ . \tag{4.2}$$

given

$$\omega(p,c) = 2 - \frac{1}{|\mathcal{C}_p|} \ . \tag{4.3}$$

In the next step, we build the arithmetic mean of the two inverse relations as shown in Equation 4.2. The relation weight between two topics is then divided by the $depth$ of the topic located at the lower level. This is called relative depth scaling. It is based on the assumption that topics in lower levels are closer related than topics in higher levels. Sussna calculates the distance between topics. However, we want to model similarity, where $similarity = 1 - distance$. We need to normalize calculated similarities to gain values between 0 and 1, confer [Billig et al., 2010]. To calculate similarities, we first compute the distance of all topic pairs in the ontology. We then divide each distance by $distance_{max}$, which is calculated at the root level. Thus, the root topic shows a distance of 1 towards its subtopics. Since the similarity at the root level will result in 0, we replace these weights by $\frac{1}{|\mathcal{C}_r|}$, where $\mathcal{C}_r$ is the set of children of the root topic.

Normalize to Expert Standard

We define the expert standard by assuming that an ontology almost models the entire knowledge of a given domain and that top experts in a topic have also top expertise in its subtopics. When spreading a score to the target topic we need to normalize the score against the top expert level. We define the expert standard for topic p as shown in Equation 4.4.

$$n_{ExpertStandard}(p) = \sum_{c \in \mathcal{C}_p} 100 \cdot \omega_{SussnaRoot} \ . \tag{4.4}$$

where $\mathcal{C}_p$ is the set of topic p's children. Top expertise is associated with scores of 100 points. In Equation 4.1, we normalize with $n_{ExpertStandard}$. In case we calculate $n_{ExpertStandard}$ based topic's weight being processed (say a topic at level 5), we drop relative depth scaling and the weight in Equation 4.1 is reduced to $\frac{1}{|\mathcal{C}_p|}$. Instead, we use the weight at the root level. As a consequence, for specific topics located on very low levels, a user does not have to show top expertise in all of the subtopics to reach the maximum score. In this case, it is probably sufficient to show nearly top expertise in the sibling topics to reach 100 points in the higher-level topic.

Coefficent α

The coefficient α alters a topic's initial score as shown in Equation 4.5.

$$\alpha = \frac{1}{(1 + |\mathcal{C}_{active}|) \cdot \omega_p \cdot \omega_f} \ . \tag{4.5}$$

where $\mathcal{C}_{active}$ is the set of active topics propagating to topic p. ω_p is the outgoing relation weight of p. ω_f is the outgoing relation weight of the farthest active descendant in p's subtree, where activation originally started. For instance in scenario 3, we calculate α for the topic *object-oriented* with $|\mathcal{C}_{active}| = 1$, $\omega_p = 0.75$ and $\omega_f = 0.82$. Coefficient α prevents inaccuracies due to possibly coarse-grained source information in higher levels. We assume that expertise scores of specific topics are more reliable than that of general topics. For instance, a user's self-assessment in a general topic is possibly more biased than in a specific topic, which is usually easier to self-assess. Therefore, the more information from specific topics is available, the higher the loss of the general topic. In addition, the higher the level of a topic being activated, the higher is the attenuation of its initial score by means of ω_p and ω_f. The maximum score a topic may receive is limited to the maximum score of its children. For instance, three topics with scores of 90, 80 and 70 points activate topic p. Then, the maximum score of p is limited to 90 points.

4.2 Evaluation

To measure the performance of the novel approach against the baseline approach we set up various scenarios serving as calculation tasks for both algorithms. We then calculated expertise scores for each scenario and asked experts to assess the scores by means of an online survey. We had 29 participants completing the survey, including professors, lecturers and post-docs teaching programming courses at university.

Table 4.1: Test scenarios

Scenario	Initial Scores (points)				Topics to Estimate
1	Java: 80	C++: 30	-	-	object-oriented
2	Prolog: 50	COBOL: 90	object-oriented: 20	-	programming
3	Smalltalk: 30	object-oriented: 50	-	-	structured
4	LISP: 10	Erlang: 60	Prolog: 30	-	declarative
5	C++: 70	Java: 40	Falcon: 30	JavaScript: 80	object-oriented
6	Java: 90	C++: 60	Visual Basic: 30	-	object-based
7	Smalltalk: 60	class-based: 30	-	-	class-based
8	Prolog: 40	logic: 70	-	-	logic

Table 4.2: Expertise scores calculated for the given scenarios

Scenario	1	2	3	4	5	6	7	8
Baseline Approach	27.5	20.4	8.8	17.8	36.7	27.5	44.0	82.0
Novel Approach (γ)								
(0.70)	25.3	9.0	5.0	11.3	45.2	33.9	39.3	56.5
(0.75)	29.1	11.1	6.0	12.9	48.5	39.2	41.0	58.9
(0.80)	33.1	13.5	7.0	14.7	51.7	45.1	42.8	61.4
(0.85)	37.3	16.3	8.2	16.6	54.9	50.9	44.5	63.8
(0.90)	41.9	19.4	9.6	18.6	58.1	57.1	46.3	66.3
(0.95)	46.6	22.9	11.1	20.7	61.4	63.6	48.1	68.7
(1.00)	51.7	26.8	12.7	23.0	64.6	70.5	49.8	71.2

4.2.1 Test Scenarios

Table 4.1 shows the scenarios we defined to test the algorithms in different hierarchy levels and at different topic densities. Due to relative depth scaling, we expect the novel algorithm to perform significantly better in scenarios with a high density of topics located in lower levels (covered by scenarios 1, 5, 6). On the other side, we expect rather similar behavior the more general and the more scattered the topics are (Scenarios 2, 4). We also investigate the propagation of scores on the same path testing different path lengths (Scenarios 3, 7, 8).

4.2.2 Settings and Score Calculation

Before we started calculation, we experimented with settings for the decay factor γ. It seems reasonable to us that a one to one relationship of two topics should nearly result in equal scores for both topics. We performed propagation with varying decay factors and found that scores of the topics *Prolog* and *logic* are nearly equal (Prolog: 50, logic: 52) at $\gamma = 0.85$. The baseline approach works equally regarding a one to one relationship. Table **??** shows the propagated scores given our scenarios. As we expected, scenarios 2, 3, 4 and 7 show almost identical results and scores are the closest at $\gamma = 0.85$. The difference in scores for scenarios 1, 5, 6 and 8 are

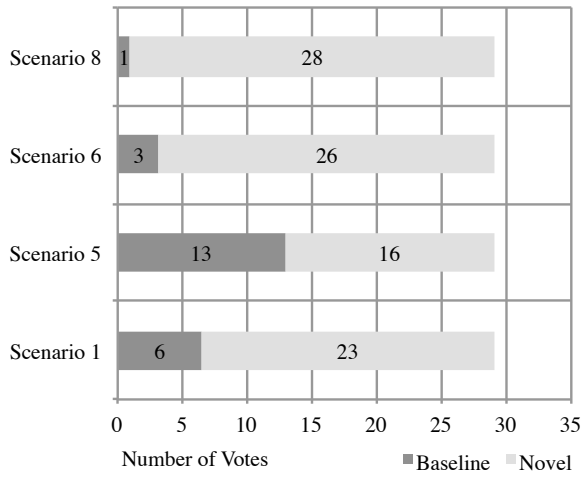


Figure 4.3: Survey results.

worth to notice. We were interested, which scores would be chosen by experts, if they had to vote for a score showing the more accurate tendency.

4.2.3 Expert Survey

We set up an online survey and asked experts for their estimates. For details on the survey forms, please refer to Figures A.3, A.4, A.5, A.6, A.7, A.8, A.9 and A.10 in the appendix. In particular, we were interested in how experts evaluate the scores in scenario 1, 5, 6 and 8 since these scenarios showed a clear difference in score results. After a brief description on how a beginner is distinguished from a top expert, we displayed for each scenario the initial scores and the two calculated scores, one coming from the baseline the other from the novel approach. Experts were asked: "Please choose the score that in your opinion reflects the better tendency for expertise . . . ". Both the ontology and the source of scores were hidden from the participants. Since the scenarios' initial scores are scaled in ten steps, we carefully converted the result scores to the same scale. We assume that this might facilitate the decision-making of participants and thus reducing participants' subjective bias. Scores were converted as follows: Scenario 1 with scores of 27.5/37.3 rounded to 30/40, scenario 5 rounded to 40/60, scenario 6 rounded to 30/50 and scenario 8 rounded to 60/80.

4.2.4 Results and Findings

Scenario 1 was intended to test the algorithms' behavior in lower levels with moderate topic density. 78% of the domain experts perceived the scores coming from the novel approach as more accurate. Scenario 5 aimed to test at lower levels with higher density of topics. In this scenario 56% voted for the novel approach. In scenario 6 we observed the algorithms' behavior in lower levels propagating several levels towards the top given a moderate topic density. Results

show that 89% of the experts found the novel approach's score more accurate. Finally, scenario 8 was intended to test the influence of coefficient α on a topic's initial score. The more specific information available, the more the initial score is attenuated. In contrast, the baseline approach attenuates a propagated score more, the higher the topic's initial score is. 97% of the experts favored the score calculated by the novel approach.

An expert's assessment is inherently subjective and thus the occurrence of bias is unavoidable. However, we aimed at reducing subjective bias while compiling the sequence of scenarios as well as the sequence of response items. Regarding the former, two of our scenarios considered the same target topic to estimate (*object-oriented*) even though the given topics were different. We separated these two scenarios in order to not appear one after the other to prevent possible priming effects. Concerning the sequence of response items, the baseline and novel scores changed place over the scenarios, meaning that the baseline was not always the first option to choose and vice versa. A limitation to our survey design is that it only provides two options to choose from for each scenario, i.e., the result of the baseline vs. the novel result. Such an either/or-decision certainly represents a harder cognitive challenge than a higher amount of options. However, our results seem relatively clear to state claims upon them.

In summary, the novel approach outperforms the baseline approach the lower the topics reside in the hierarchy. Only the result of scenario 5 weakens this claim. However, results of scenario 5 does not significantly speak for the baseline either. Scenario 5 is the one with the most given scores in the task description, which possibly makes expert assessments more difficult and thus leads to a broader distribution of estimates. The results also suggest that the coefficient α is useful for altering initial scores. Despite these promising results, our study is not without shortcomings, i.e., the small size of the ontology as well as the small amount of scenarios tested so far. However, a strong point is certainly the empirical assessment by means of professors, lecturers and post-docs teaching programming courses at university.

4.3 Summary

We proposed a novel algorithm to propagate expertise scores in an ontology overlay model based on constrained spreading activation and relative depth scaling. We compared the algorithm's performance with a baseline. 29 experts evaluated the calculated expertise scores given various scenarios. Thereby, our algorithm outperforms the baseline approach in half of the test scenarios. For the remaining scenarios both algorithms propagate almost equally. These results suggests that the calculation of user expertise utilizing constrained spreading activation and relative depth scaling can lead to more accurate user models.

CHAPTER 5

Predicting Expertise in Open Learner Modeling

The Expertise Calculator as proposed in this thesis relies on a score propagation method to align expertise scores using knowledge from a competence ontology. However, such a propagation method can be embedded in various kinds of applications. This chapter is mainly motivated by our work in the previous chapter where we introduced a new approach to score propagation in ontology overlay models. The first evaluation of the proposed method involved human experts estimating the validity of propagation results. Thus, given its potential use in other applications and the interest to further evaluate our score propagation approach with users instead of unconcerned experts, we test our propagation method in a new environment, i.e., open learner modeling.

In recent years, learner models have been increasingly opened to learners allowing them to scrutinize and update information stored in the system [Bull, 2004, Mabbott and Bull, 2006, Bull and Kay, 2007]. One of the potential benefits of this approach is to gain more accurate and extensive learner models. This enables adaptive systems such as intelligent tutoring systems to provide more effective personalized tutoring. Furthermore, the active involvement of learners in building and maintaining their models may contribute to learning [Kay et al., 2007, Bull and Kay, 2012].

To use open learner models to elicit learner's expertise, we need to find ways to support learners in estimating their expertise effectively? If we aim to support a learners' reflection and achieving high quality self-assessments, more guidance is an important ingredient [Zapata-Rivera and Greer, 2004]. A prerequisite for guidance is interaction. Systems that support learners in building their models rely on intense interaction between learners and the system. Indeed, one approach involves learners and the system working together by negotiating their beliefs [Bull and Pain, 1995] [Dimitrova, 2003]. We hypothesize that expertise predictions have the potential to serve an important role in guiding learners in self-assessing their knowledge to quickly create rich learner models. While learner self-assessment may not necessarily be accurate, there is considerable evidence that bias may be systematic [Kleitman, 2008] and so it can be valuable.

Furthermore, students in advanced courses seem to achieve more accurate self-assessment than students in basic courses [Falchikov and Boud, 1989].

In this chapter, we mainly ask two questions:

1. How does the prediction of expertise affect the process of learners' self-assessment?

   More specifically:

   a) Will learners prefer a specific level/range of expertise predictions to be displayed during the interaction to elicit a learner model?

   b) Will learners attempt to align their own expertise scores between topics in their model?

   c) How accurate will predicted scores match learner self-assessment?

2. How will expertise predictions affect the characteristics of learner models?

   In particular:

   a) Which levels/range of expertise scores do learners assign to topics selected for their models? Will learners focus on their weaknesses and strengths equally?

   b) How is the density of a learner model affected when learners are supported with expertise predictions?

In order to calculate expertise predictions, we employ the score propagation algorithm presented in Chapter 4. In this way, we are able to evaluate our propagation approach for a second time but with a different type of subjects, namely, user's of the system. Our first evaluation involved experts assessing the accuracy of expertise predictions. In contrast to that, we now seek to compare the predicted scores with learners' self-assessments.

To examine possible effects of expertise predictions we conduct an experimental study with students separated into two groups. One group will use an interface featuring expertise predictions (Prediction Group) and the other group works without predictions (Control Group). An expertise prediction is represented by a topic and its score value ranging from 0 to 100 points, like *programming:75*. Predictions are calculated based on learners' self-assessments as they were reported to the system. Thus, these self-assessments constitute the initial values for score propagation. As soon as learners update their models, predicted scores will be promptly recalculated and displayed.

Our research aims to *elicit a rich user model* as a basis for subsequent personalization of the learning environment. It does this by creating an interface to the model of the learner's knowledge. This builds upon the growing body of work on *Open Learner Models* (OLMs). In our work, the OLM interface and associated inference mechanisms were designed to enable learners to *self-assess* their knowledge, a core metacognitive skill.

Open learner modeling research has explored the main ways that a user model can be usefully be made available to the learner. These include improving the accuracy of the model, navigation within an information space and supporting metacognitive processes such as setting goals, planning, self-monitoring, self-reflection and self-assessment [Bull and Kay, 2007]. We

build our work on the last of these, for the purpose of quickly creating a learner model. At the same time, the process of self-assessment provides a valuable way to self-reflect and this is valuable for improving learning [Boud, 1985].

There are many forms of interfaces to open learner models [Bull and Kay, 2007]. Some of the earliest and simplest take the form of a 'skill meter' that is tightly linked to a single teaching system [Corbett and Anderson, 1994]. More recently, there has been exploration of the value of opening a learner model that is independent of any single application [Kay, 2008, Bull and Kay, 2012, Bull and Gardner, 2009]. Notably, such independent open learner models can be useful for learning in supporting reflection [Bull and Gardner, 2009] and can serve as the basis for learners identifying their own learning goals [Mabbott and Bull, 2006]. We continue this trend, as we explore the creation of an interface to support self-assessment.

In the case of large learner models, there is a need for particular care in the design of the interface and the support for effective interaction. The VlUM interface aimed to support reflection, planning and navigation based on suitable interfaces onto large learner models [Apted et al., 2003]. This could also be incorporated into learning systems, for example to support reflection in a programming subject [Kay et al., 2007]. It showed an overview of the learner model. Each concept was color coded, with green indicating a concept was known and red that it was not known. The color intensity indicated the knowledge score, with the brightest green for higher positive values for the modeled concept. The interface could be configured with a user control to set the threshold for these colors. So, for example, a learner may decide that they only want concepts to appear green if their score is above 80% [Apted et al., 2003]. In order to support the creation of richer learner models, this interface was augmented with ontological inference [Kay and Lum, 2005a]. This was used to take fine grained data, based on the learner's interaction with each task in the teaching system, then it inferred the value of more general concepts in the learner model. It also inferred finer grained concepts from data about general concepts using grades on larger assessment tasks.

5.1 Experimental Study Design

In order to examine possible effects of expertise predictions in open learner modeling, we conducted an experimental study with Masters students in a computer science program. In the course of a lecture on knowledge management, participants were randomly separated into two groups representing the Control Group and the Prediction Group. The Control Group was exposed to a user interface without predictions whereas the other group was supported by predictions. We put both interface variants online and notified the participants to start building their learner models. We chose the domain of *software engineering* since our participants are supposed to have some expertise in this area from their previous studies. After constructing their models, we asked them to complete an online questionnaire. In particular, we asked how useful participants found the predictions and we invited free comments about likes, dislikes and possible improvements to the prediction feature.

To explore predictions' effects in open learner modeling, we time-stamped and recorded all participants' interactions with the interface. This allows us to reconstruct a learner's model for any time in the model's construction process. Each estimated topic score in a learner's

model is associated with its source indicating whether it was originally selected and estimated by the participant or it came from the prediction engine. After collecting the data, we designed measures for each of our four research questions.

We asked the participants to build learner models from scratch and to finish within two weeks time. Participants were provided with a brief manual on how to use the interface. They completed this task as a one-off with no consequences (either benefits nor negative effects) for poor self-assessments. However, we informed them about the university's plans to create a tutoring system for recommending lectures in the future. We explained that with their help we aimed to improve the self-assessment process needed to make it as easy and effective as possible for students.

User Interface

We provided our two study groups with slightly different interfaces for building their learner models. As a starting point for both variants, we adopted the interface devised in Section 3.3 designed to maintain a users' competence profiles represented as overlays.

Figure 5.1 illustrates the adapted interface for the Prediction Group. We changed the previous interface to give a wider range in the expertise scale granularity (from a granularity of 10 to 5 points). The previous interface used a so called *bullet graph* to represent score values by combining both a qualitative (indicating ranges for beginners, intermediates and experts) and a quantitative scale. We removed the qualitative scale since we did not want to influence learners with predefined ranges like "this is the range for intermediates and I would say my expertise is somewhere in the middle". Learners should be invited to think about finer grades of their expertise. In the upper part, learners select topics from a hierarchically structured domain ontology (we used the one devised in Section 3.2.1), estimate their expertise scores and add the expertise to their model shown in the table below.

In order to obtain predictions, we draw on the algorithm proposed in Chapter 4 exploiting the ontology's network structure to propagate expertise scores through related topics. The algorithm's scores are integrated with the learner model as shown in the bottom right part of Figure 5.1. The top left shows the selection of the topic (1). Learners can either enter a topic in the top text box (1a) or select one of the hierarchy of topics, such as Programming (1b). A selected topic then appears on the top right, where the learners assign their self-assessments (2) and Add/Update their scores to the model illustrated at the bottom. The prediction engine dynamically calculates scores based on the scores shown in column *self* and updates the model table. The learner can customize the model's display (3) by filtering the model according to a specific string and by setting a score threshold (ranging from 10 to 100 points in steps of 5) to restrict the display of predicted scores below the threshold value. We intentionally set the lowest possible threshold value to 10 points since we believe that lower scores lack expressiveness and might annoy learners. Learners can now scrutinize (4) their model by inspecting its structure and scores. They can alter their self-assessments by clicking on a topic in the model, which loads the topic in the top view as it is the case in Figure 5.1 for the topic *Procedural Programming Languages*. Participants working with predictions had to prime the model with five initial scores, so enabling the prediction engine to respond with reasonable scores right from the start.

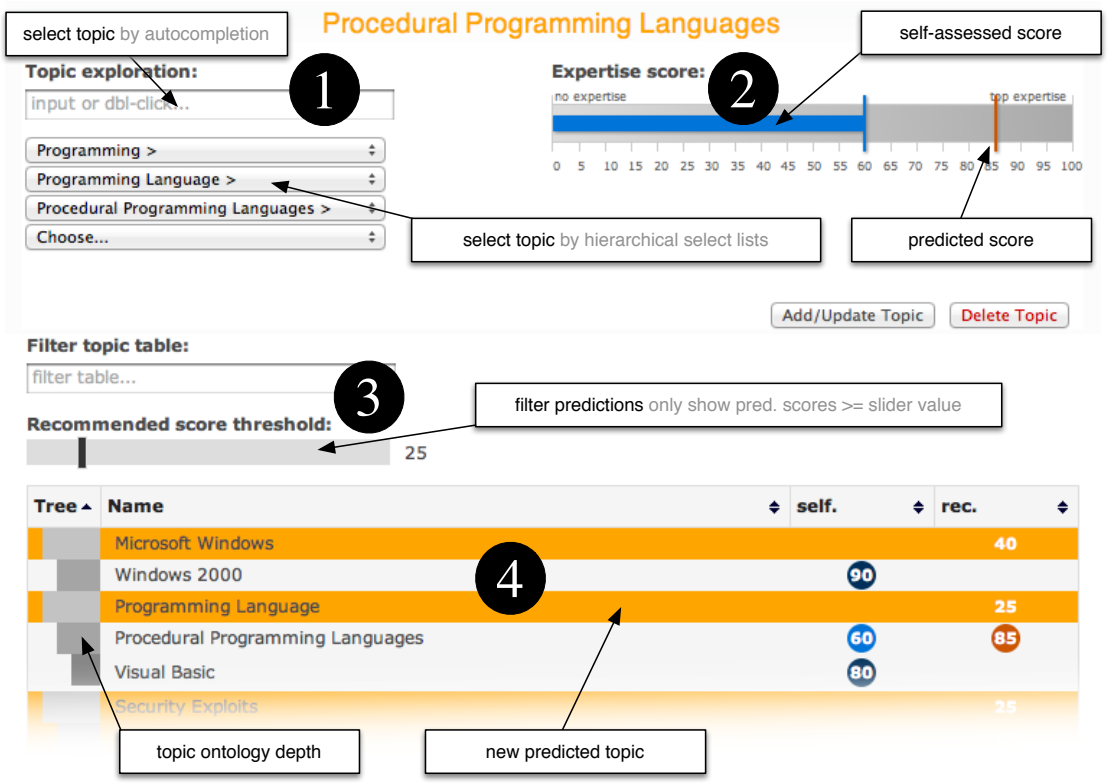


Figure 5.1: Building the Learner Model Utilizing Expertise Predictions.

The interface for the Control Group looks basically the same except for the expertise predictions in the model and the slider element.

5.2 Evaluation and Results

During the study we collected 21 complete datasets from students in the Control Group and 29 from students working with predictions. It is essential that predicted scores show an almost uniform spread in their distribution. If the prediction engine had mainly suggested high-level scores, this may have encouraged learners to focus more on their strengths. This may also have also caused the unwanted effect that the predictions might drive learners to overestimate their scores.

Table 5.1 displays the distribution of scores in our study. A uniform spread is present with an interquartile range ($iqr = Q_1 - Q_3$) of 50 points and a median average deviation (mad) of 25 points. We see that the actual predicted scores with $iqr = 40$ and $mad = 29.65$ are close to a perfect uniform spread. However, we observe that the distribution is slightly skewed as indicated by its median.

Table 5.1: Distribution of scores computed by the prediction engine

	n	min	Q_1	median	mean	sdev	Q_3	max	mad
Predicted scores	1115	0	40	60	59.59	28.09	80	100	29.65

5.2.1 Preferred Levels for Expertise Predictions

In this section, we tackle the following research question:

1.a: Will learners prefer a specific level/range of expertise predictions to be displayed during the interaction to elicit a learner model?

We were interested to assess whether learners prefer a certain level of scores for expertise predictions. Table 5.2 shows the data we collected during the study. On average, participants

Table 5.2: Statistics of the score level threshold data

moves	min	Q_1	median	mean	sdev	Q_3	max	range
509	10	20	35	40.88	24.38	55	100	90

moved the slider 18 times while completing the task. The average mean is 40 points with a standard deviation of 25 points. Hence, these data do not suggest that participants prefer a certain level of predicted scores. But we found that 50% of the slider values are located between 20 and 55 points. Combining these data, we see that participants used a range of approximately 20 to 60 points to customize their display of predicted scores. This suggests that learners choose predictions about their strengths (scores > 60) over predictions about their weaknesses.

From the questionnaire results, 83% participants reported the threshold was useful to restrict the lower bound of predictions. In addition, participants said that they tried to understand how the prediction engine calculates expertise scores and were curious about which scores would

show up next. The responses show further that 66% had fun while constructing their learner model. To sum up, what we seemed to observe was a behavior where learners were 'playing' with threshold values to gain understanding of the calculation of predictions as well as satisfying their curiosity.

5.2.2 Alignment of Expertise Scores

We assume that the effort that learners make in aligning expertise scores amongst related topics encourages them to reflect on their knowledge. In this regard, we aim at answering the following question:

1.b: Will learners attempt to align their own expertise scores between topics in their model?

To examine learners' alignment behavior, we devised a measure based on topics revisited several times during the model's construction process. The measure uses the following steps:

(1) Create a list containing all topics in a learner model revisited more than once. For each topic in the list, we perform the subsequent steps.

(2) Get the timestamp of topic *A*'s second visit.

(3) Scan *A*'s related topics within the model and located within a maximum distance of 2 in the ontology tree (includes parents, children and siblings).

(4) Test if related topics have been altered by the learner after the second visit of *A* as determined in Step 2.

We interpret the related topics as identified in Step 4 to be influenced by topic *A*. Such influenced topics represent learners' attempts to align expertise scores. Because of the relatively small learner models and the limited time frame of our study, we could only collect sparse data to test the alignment behavior. However, Table 5.3 summarizes the results of our analysis.

A few participants had not revisited more than one topic, which reduces our datasets' size. The average model size in the Control Group is 22 and models in the Prediction Group were approximately double this size. On average, 6 topics per model were revisited in the Control Group in contrast to 9 topics in the Prediction Group. It is notable that about half of the revisited topics in the Prediction Group were topics originally predicted by the engine. This suggests that the availability of the predicted scores may have helped motivate the participants to work on more of their model. The average number of actual attempts to align topics (Topics influenced) is higher in the Control Group. But it is interesting that 75% of influenced topics in the Prediction Group originate from predictions, which indicates that predicted scores might motivate alignment.

5.2.3 Accuracy of Predicted Scores

The score propagation method as presented in Chapter 4 predicts expertise scores given a set of initial scores. We conducted the first evaluation of this algorithm by means of scenarios, confer

Table 5.3: Statistics regarding participants' attempts to align expertise scores

Control Group (reduced to 13 students)						
	median	mean	sd	min	max	range
Model size	20.00	21.85	11.29	7	47	40
Topics revisited	4.00	6.46	4.59	2	17	15
Topics influenced	4.00	4.38	4.21	0	14	14

Prediction Group (reduced to 21 students)						
	median	mean	sd	min	max	range
Model size	34.00	38.76	11.83	30.0	72.00	42.00
Topics revisited	8.00	9.43	8.38	1.0	32.00	31.00
Topics revisited (origin=self)	4.00	5.76	6.14	0.0	20.00	20.00
Topics revisited (origin=rec)	2.00	3.62	5.28	0.0	20.00	20.00
Topics influenced	1.00	2.10	2.59	0.0	9.00	9.00
Topics influenced (origin=self)	0.00	0.52	1.63	0.0	7.00	7.00
Topics influenced (origin=rec)	1.00	1.57	1.96	0.0	6.00	6.00

Table 4.1. In this section, we explore learners' responses to expertise predictions represented by learners' self-assessments. We ask in particular:

1.c: How accurate will predicted scores match learners' self-assessments?

A limitation to our first evaluation attempt in validating score accuracy was certainly the small size of the underlying domain ontology. In contrast, the ontology we use in the present study consists of 454 topics. It has an average topic depth of 3.63 and a maximum topic depth of 7.

In the present study, we determine score accuracy as follows. First of all, learners select a new predicted topic from their model. For example, in Figure 5.1, the user clicks on the topic *Programming Language*. Then, the topic shows up at the top right, ready for self-assessment. The self-assessed score is initialized with the predicted score, thus the long bar element and the cross bar show equal scores. We now observe if learners adopt the predicted score directly, that is when they just update the topic to their model without altering the long bar representing their self-assessment. We interpret scores directly adopted by learners as scores they perceive as accurate. In addition, for scores not directly adopted but altered before stored to the model, we measure the average deviation of the self-assessment from the originally predicted score level.

We collected 1115 self-assessments from participants in the Prediction Group. The system stores each topic of a learner model together with its self-assessed and predicted score. Due to the score range given by the slider element, learners' can only inspect predicted scores greater or equal than 10 points. Even though these scores are not displayed to the user, they are stored in the system. Therefore, we had to remove all predicted scores from the dataset with scores below the minimum slider threshold ($>= 10$). This reduces the dataset from 1115 to 1055 items.

Table 5.4: Participants directly adopting predicted scores

Directly adopted n	0	1	2	3	4	5	6	7	9	10	12	14	18	19	26	36
Participants n	6	2	4	1	1	3	2	1	2	1	1	1	1	1	1	1

The collected data show that participants adopted 485 topics originally coming from the prediction engine whereas the remaining 570 origin from participants' own reflections. We found that 204 of 485 adopted scores were directly adopted meaning that these scores were not altered by participants before being added to their models. As shown in Table 5.4, six participants did not directly adopt any predicted scores at all. 17 participants directly adopted predictions from one up to 10 times whereas the remaining 6 participants accepted expertise scores up to 36 topics. For the total number of adopted scores, the average mean deviation of predicted scores from participants' self-assessments is 14.16 points.

We now determine Pearson's correlation coefficient on the full data set (1055 items). The data comprise items that either origin from predictions or from participants self-reflections. In the latter, participants add topics to their model that were not predicted previously. However, after these topics are added to the models, the system augments these topics with a predicted score. Thus, each topic in the model features two expertise scores. Measuring the correlation coefficient across these data pairs results in $r = 0.8133$ with $p < .000$ $(2.2e-16)$. This signifies a strong positive correlation between self-assessments and predictions. Figure 5.2 shows the regression line considering this relationship. From regression calculation we obtain a residual standard error $\sigma = 13.67976$. σ describes the spread from the regression line, i.e., how far away typical predicted scores will be from the regression line. Figure 5.2 illustrates the 204 directly adopted scores sitting on the dotted line representing a "perfect" prediction. We observe that scores ranging from 10 to 90 points indicate a linear relationship as already suggested by Pearson's correlation coefficient. It is notable that predictions associated with the highest score level show a larger spread from the regression line than others.

In summary, we observed that 485 predicted scores were adopted from participants. In almost half the cases, participants accepted scores without alteration and added them to their model. The rest of the scores were on average altered by 14 points. Linear regression calculated on the full dataset showed an average error of expertise predictions around 14 points as well. Participants were asked in the closing questionnaire, if they were satisfied with the levels of predicted scores. The collected answers suggest a neutral preference in this regard, although none of the participants explicitly stated clear satisfaction (satisfied:0, mostly satisfied:12, mostly dissatisfied:16, dissatisfied:1). This is an unexpected response because a large amount of predicted scores was directly adopted where the rest was just marginally altered. However, we have no evidence about predicted topics, like *Programming Language* in Figure 5.1, that have not been selected and finally added to learners' models.

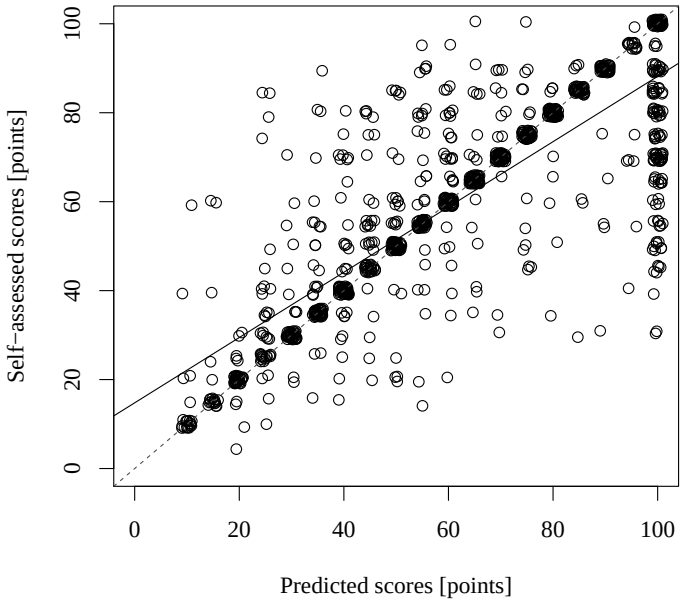


Figure 5.2: Linear Regression. The solid line fits the self/predicted data pairs best whereas the dashed line represents the theoretical perfect fit. (Both variables jittered)

5.2.4 Levels and Range of Self-assessments

Learners' self-reflections focussing rather equally on their strengths and weaknesses will result in a more extensive learner model than in case learners would only prefer to think of their strengths alone. In this regard, we ask:

2.a: Which levels/range of expertise scores do learners assign to topics selected for their models? Will learners focus on their weaknesses and strengths equally?

To tackle this question, we explore the levels and ranges of expertise scores learners used while building their models. We examined the distribution of scores for either groups as illustrated in Figure 5.3.

Table 5.5 shows that scores in the Control Group are skewed, meaning that participants tend to assign higher scores. Similar to the evaluation method for the predicted scores in Table 5.1, we measure in the Control Group distribution values of $iqr = 30$ and $mad = 14.83$. It is interesting that participants of the Control Group were reluctant to assign scores up to the maximum value.

Table 5.5: Distribution of learners' self-assessments

	n	min	Q_1	median	mean	sdev	Q_3	max	mad
Control Group	411	5	40	60	52.94	21.31	70	90	14.83
Prediction Group	1115	5	40	60	58.09	24.29	80	100	29.65

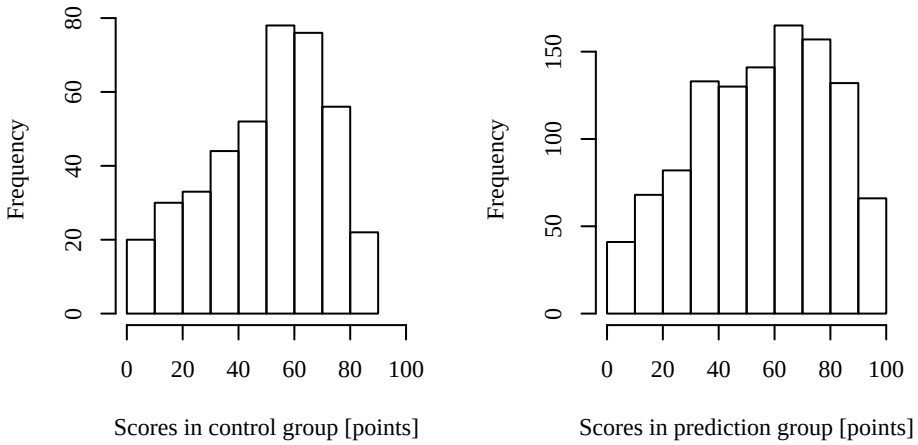


Figure 5.3: Distribution of participants' self-assessed scores.

This is especially for the interval of 90 to 100 points. All of the participants avoided assigning scores above 90 points.

Self-assessments in the Prediction Group are also skewed but to a smaller degree than in the Control Group. This becomes clear when we look at the distance between the median and average mean which is remarkable smaller (1.91) than the distance in the Control Group (7.06). Furthermore, comparing iqr and mad we see that the scores used in the Prediction Group are closer to the perfect uniform distribution standard ($iqr = 50, mad = 25$) than those of the Control Group. Importantly, we observe that the Prediction Group was willing to use high expertise scores.

We now consider the origin of topics, whether they were initially selected by the participants or suggested by the prediction engine. There were 356 (32% of 1115 total) predicted topics accepted by participants with the rest of topics originating from participants' own reflections. We found that 45 topics were self-assessed with the maximum score of 100 points. Interestingly, 34 of these topics origin from the prediction engine. This finding represents a clear distinction to the Control Group where none of the participants self-assessed topics with 100 points.

In summary, the results suggests that participants in the Prediction Group focused their expertise scoring on a somewhat larger part of the model. Hence, this suggests that predictions help learners to explore their model more broadly, reflecting on both their strengths and weaknesses. However, we note that this may have been influenced by the novelty of the system. At the same time, participants seemed to think the predictions were helpful, with 66% indicating it was fun to work with predictions and 62% that predictions shorten the time building their learner models. We know from questionnaire results that many of the participants were curious about how the prediction engine works. Together, this suggests the predictions may have led to a higher level of motivation to use the system. This could be very important for maintaining the model over a longer period.

5.2.5 Model Density

A model's density describes the distance between self-assessed topics in a learner model. In this section, we aim to answer the following question:

2.b: How is the density of a learner model affected when learners are supported with expertise predictions?

We expect higher densities in the Prediction Group since learners are provided with predictions related to their self-assessments. This possibly increases the extent of self-reflection on related topics. Equation 5.1 shows our measure for density. First, we calculate the mean of the shortest paths between each pair of topics. Since we regard the density of a model to be higher the more topics the model contains, the mean of the shortest paths is multiplied by the proportion of topics in the model and the total number of topics in the domain ontology.

$$density(m) = \frac{\sum_{pair \in \mathcal{M} \times \mathcal{M}} shortest\_path(pair)}{|\mathcal{M} \times \mathcal{M}|} \cdot \frac{|\mathcal{M}|}{|\mathcal{O}|} \ . \tag{5.1}$$

where $\mathcal{M}$ is the set of topics contained in the model m and $\mathcal{O}$ the set of topics represented in the domain ontology. The maximum value for a model's density occurs when the model contains the entire set of topics in the domain ontology. Hence, as shown in Equation 5.2, we normalize the $density$ to the maximum possible value obtaining a final $density$ between 0 and 100%.

$$density_{norm}(m) = \frac{density(m)}{density(ontology)} \ . \tag{5.2}$$

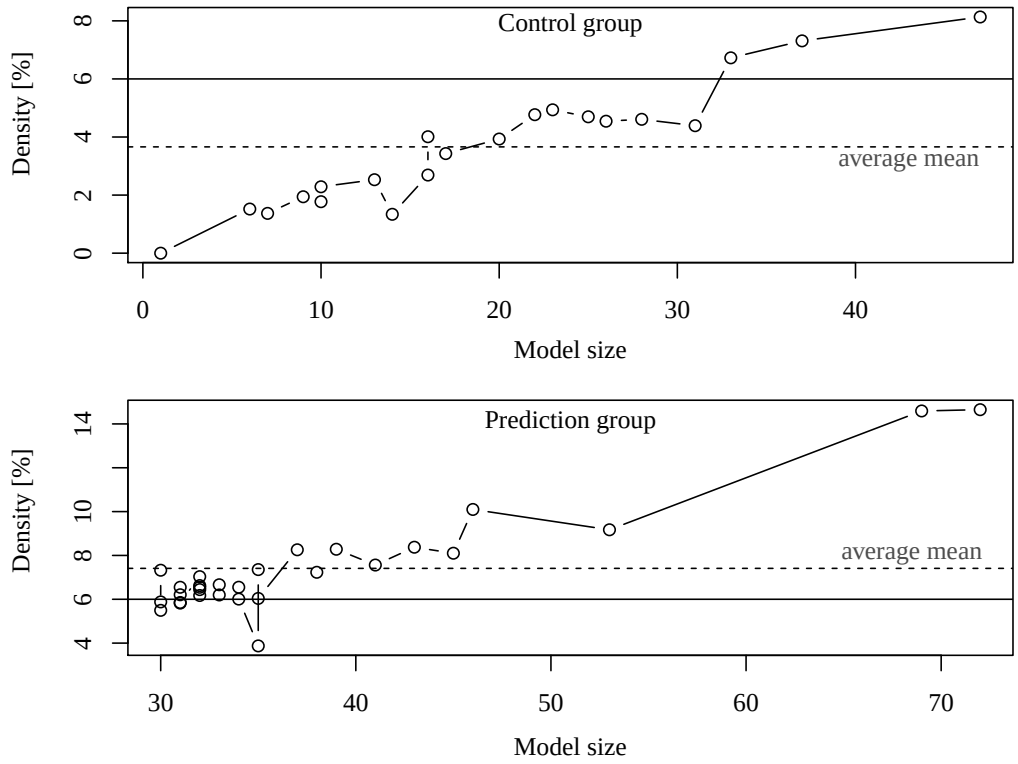


Figure 5.4: Model densities at increasing model size. Within the interval of 30 to 35 topics the densities in both groups amount to approximately 6 %.

Since the size of learner models differs significantly between the two groups, it is hard to compare densities. Figure 5.4 illustrates the development of densities at increasing model sizes. Based on these data, we can only compare density values between 30 and 35 topics. A restriction to this sample window reduces the Control Group data to three items, which is relatively little to state strong claims. However, we observe that for a model size ranging from 30 to 35 topics the density is about 6% for both groups.

According to Equation 5.1, a model's density increases with the number of topics in the model. We want to make sure that our measure does not excessively depend on either the average path length or the model size. Hence, a valid measure would show a trend of rising density values at increasing model sizes, but the rise of density would not necessarily be steady. Figure 5.4 shows that densities not only depend on the size of the model but also on the shortest paths between its topics. Density values rise and fall even though the models' sizes increase.

5.2.6 Feedback

We asked the participants to complete an online questionnaire (for details refer to A.11 and A.12) after building their models. Since we focus on the effects of predictions, we only report on the feedback of the Prediction Group. Figure A.13 illustrates responses to closed questions

where 62% of participants liked the predicted scores although 38% rated them mostly useless (useful:2, mostly useful:16, mostly useless:11, useless:0). 83% found the slider element to be useful to limit the display of predicted scores (useful:13, mostly useful:11, mostly useless:3, useless:2). 62% believe that a prediction feature shortens the time to build a learner model (yes:18, no:11). And finally, 66% said that it was fun to work with predictions (yes:18, no:11).

From the open questions about likes, dislikes and improvements, it seems that participants found it challenging to decide what it means to be an expert. Selected quotes are:

> *When is someone an expert and when not?*
>
> *I got a very good in Artificial Intelligence. But am I an expert in this topic?*
>
> *Someone else might say that he has used Java for 10 years but he still feels that there are better people than him, so he gives himself 80%.*
>
> *Further I don't know the reference point of the scores. (e.g. 'all people', students of informatics, ...?)*

Even though we declared the expert level as having problem-solving capability in the respective topic, participants experienced difficulties. This is part of a broader challenge in defining what an expert level means.

Another finding concerns self-reflection. Selected quotes are:

> *It was interesting to think about questions i did not have in mind before (what is my expertise).*
>
> *It helps to find mistakes and makes me rethink my self assessment.*
>
> *Was interesting to see how the software thinks my expertise is.*

These statements suggest that predictions can trigger mechanisms to think about one's expertise in more detail as well as scrutinize one-selves believes.

Lastly, participants expressed the wish after a more transparent prediction process:

> *I dislike the present interface because I don't understand how the predicted score is calculated.*
>
> *The system should reason (comment) its predictions.*
>
> *It would be nice to be able to get a short explanation from the system on how the score was derived.*
>
> *Scores were irritating, because I don't know how they are determined.*

5.3 Summary

In this chapter, we examined the effects of expertise predictions on learners' expertise models as well as on the process of their self-assessments. Our study results indicate that predictions can have a positive influence on learners' motivation. This appears to be one reason that models for the Prediction Group were almost double the size of those for the Control Group. Furthermore, predictions appear to help learners to broaden their focus to include both their strengths and weaknesses. [Dunning et al., 2004] argue that people who carefully consider what they know and do not know may improve the accuracy of their self-assessments. Therefore, a broader scope on one's expertise might lead, for instance, to a more effective planning of further learning activities. The majority of participants appreciated the system's expertise predictions and also think that they shorten the time effort in building their models. Although we have not tested the validity of participants' self-assessments, our study represents a critical precursor before incorporating this class of interfaces into broader contexts, e.g., long term learner modeling. Moreover, tendencies to bias in self-assessments are likely to be consistent [Kleitman, 2008] and over the long term, changes in these self-assessments could be valuable for learners' reflection on their progress.

We used the study conducted in this chapter as yet another evaluation to test our score propagation method. We analyzed participants' responses to expertise predictions generated by the proposed score propagation approach. Since we exploit the participants' self-assessments to validate predictions' score accuracy, it is hard to compare the results of this study with our first evaluation setting that relies on human experts preferring expertise predictions to others. However by means of this study, we have applied score propagation to a substantially larger domain ontology compared to the one used in previous evaluation. Furthermore, we considered propagated scores from a different perspective, that is of users being "assessed" by a system's predictions. In general, this personal involvement of individuals represents a crucial aspect regarding the acceptance of systems that might incorporate automatic expertise calculation. In so far, it was interesting to see that participants were not clearly enthusiastic about the predicted score levels, even though the expertise scores came very close to their self-assessments.

CHAPTER 6

Evaluation

In the course of this thesis, we conducted three experiments to explore the validity of the proposed Expertise Calculator regarding the prediction of users' expertise based on their contributions in an online community. The key aspect distinguishing these experiments is the maturity of the underlying concept to measure expertise. Accordingly, the very first experiment was designed as a pilot run. The goals of the pilot run were to test various text mining approaches, to construct an initial competence ontology, mapping the terms obtained from text mining to topics in the ontology and to ensure usability of the prototype. The calculated expertise was displayed to participants in an aggregated form, i.e., expertise was expressed by means of competence fields rather than single expertise topics, recall Figure 3.5. 23 participants gave feedback on how well the assigned competence fields match with their actual expertise.

The second evaluation aimed at testing an improved version of the Expertise Calculator extracting expertise from different contribution types while incorporating a score propagation method to align expertise topics according to their abstraction levels. We had 14 students participating in our second evaluation cycle. Expertise predictions were assigned to participants' expertise models and the participants evaluated the algorithm's predictions by means of their self-assessments. In fact, participants were asked if the calculated scores represent a perfect match of their expertise or if the algorithm tends to either under- or overestimate their actual performance. The first as well as the second experiment were closed by collecting participants' feedback regarding interface usability and user acceptance. For details on the second evaluation cycle refer to Chapter 3.

The experiments conducted so far served mainly as a foundation for both improving the score calculation method and testing users' acceptance regarding such kind of mining approaches. In this chapter, we continue with a further experiment. We evaluate the Expertise Calculator in its third version, i.e., we take the Expertise Calculator as it was presented in Chapter 3 and replace its score propagation method by the one proposed and evaluated in Chapter 4 and 5. This time, we explore the characteristics of the integrated algorithm design in more detail, i.e., we measure the accuracy of expertise scores beyond determining over- and underestimation and examine the confidence metric more thoroughly. Similar to the previous experiments, participants had to fill

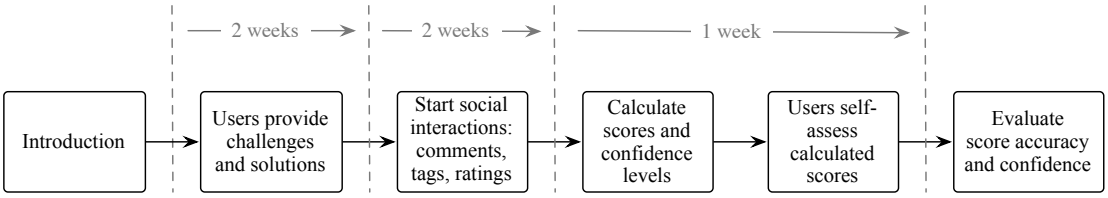


Figure 6.1: Experiment procedure.

in a closing questionnaire after completing the given task. The response to this questionnaire together with our interpretation of feedback results complete this chapter.

6.1 Experiment Design

We conducted an experiment with students at our university enrolled in a master program on computer science. In the course of a tutorial in knowledge management, students were asked to participate in our experiment. This section describes the various steps of the experiment including the measures to validate predicted scores as well as the evaluation of the confidence metric supporting expertise predictions.

6.1.1 Task and Procedure

Figure 6.1 illustrates the main steps of the experiment. We opened the tutorial with an introduction session where we provided students with the key aspects and goals of expertise mining. We told them about the university's initiative concerning the evaluation of various ways to gather students' expertise in order to improve their e-learning services, e.g., recommending new courses to students based on their expertise models. In this introductory class, we discussed the potential of various contribution types representing different levels of knowledge for the task of expertise modeling. After a lively discussion within the group, we presented the task students had to complete during the next few weeks. We need to emphasize that students received no grades based on the quality of their contributions. That was clearly communicated to the students before they started to work on the task.

The given task had a quite similar design compared with the tasks of the previous experiments. We provided an online platform to share knowledge, which students were able to access at any time and as often they want. Still in the introduction phase, participants were encouraged to get familiar with the competence cockpit as illustrated in Figure 3.18. They should get an idea how to navigate the competence ontology, select topics and save their self-assessments to their expertise model.

Within the next two weeks, participants were asked to provide challenges they had recently faced in the context of software engineering. If they were able to solve these challenges themselves, they also submitted a corresponding solution. For another two weeks, we asked students to explore the challenges and solutions submitted by peers. While inspecting others' contri-

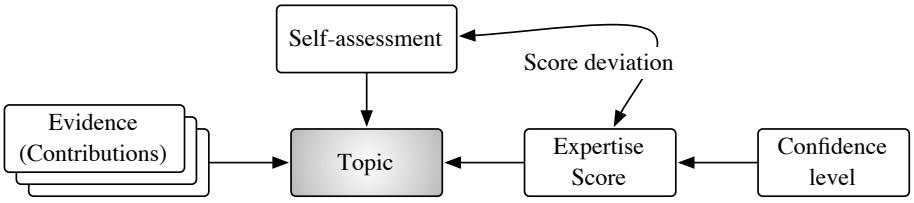


Figure 6.2: Relation of concepts used for evaluation.

butions different interaction mechanisms took place: (1) Participants contributed solutions to open challenges. (2) They also submitted alternative solutions to already solved challenges and discussed open issues by means of comments. Furthermore, (3) participants used tags to mark interesting contributions as well as (4) rated others' contributions regarding their complexity. A contribution's complexity is understood as an aggregated construct evaluating characteristics such as the extent of the contribution, its structure and the approximate expertise needed to author the contribution.

In the closing week of the experiment, we activated the calculation of expertise models and participants started to scrutinize their expertise models generated by the system. These models were not published but only presented to the individuals themselves. While scrutinizing the models' topics and expertise scores, participants evaluated each of the predicted scores with their self-assessments. Once each topic was associated with a self-assessed score, participants were asked to fill in a closing questionnaire.

6.1.2 Evaluation Measures

The behavior of the proposed score calculation algorithm depends on various parameters. Each parameter has a certain influence on expertise predictions. Similarly, the measure to determine the reliability of predicted scores is based on two independent sub-measures that affect the overall confidence level.

Figure 6.2 shows the main concepts involved in our evaluation work. During the experiment, we generate individual expertise models for each of the participants. These models represent the system's belief about the participants' expertise. An expertise is related to a certain subject matter, i.e., the expertise topic. An expertise topic is extracted from participants' contributions serving as the evidence for expertise. For each topic the system calculates an expertise score as well as a confidence level which expresses the trust in the predicted score. Furthermore, participants self-assess the topics assigned to their individual models. We refer to the absolute difference between expertise scores and self-assessments as score deviation.

In this chapter, we test whether the aforementioned relationships between independent and dependent variables actually exist. Furthermore, we evaluate the validity of both the predicted scores and their confidence levels. The validity of predictions is reflected by how close calculated scores are to individuals' self-assessments. We measure score accuracy by calculating the correlation coefficient (Pearson's r) between self-assessments and the algorithm's expertise calculations. As shown on the left side in Figure 6.3, a positive correlation $r = 1$ cor-

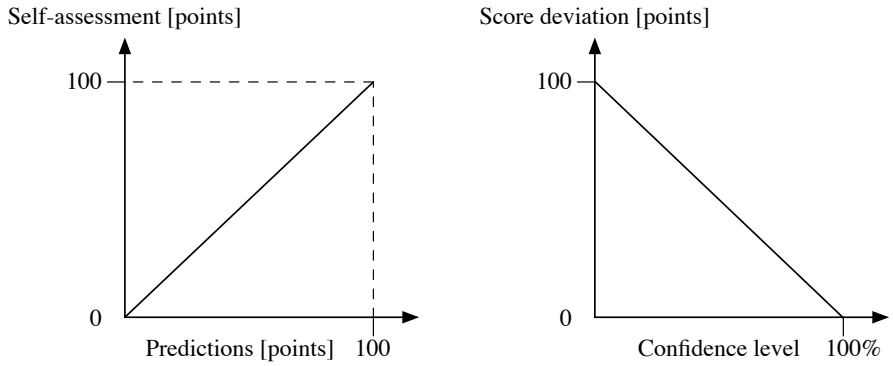


Figure 6.3: Positive correlation of scores and negative linearity in scores' confidence.

responds to a perfect prediction behavior, i.e., all calculated scores exactly match individual's self-assessments.

Regarding the validity of confidence levels, we expect that small score deviations, i.e., the gap between predictions and self-assessments, yield high confidence levels whereas large score deviations result in low confidence levels. Hence, in contrast to the positive linearity we expect for score accuracy, confidence levels will be negatively correlated ($r = -1$) as presented on the right side in Figure 6.3. Based on the collected data, we aim at exploring how close both of the proposed measures can get to perfect linearity.

6.1.3 Collected Data

19 students participated in the experiment. We calculated 1683 expertise scores in total. However, we only displayed expertise scores to the participants if they exceeded the limit of 10 points. Consequently, 1060 calculated scores were shown to the participants and qualified through their self-assessments. Table 6.1 shows some characteristics of the data we collected. In the following sections, we examine the ability of our algorithm to accurately estimate students' expertise based on the data collected during the experiment.

Table 6.1: Data collected from 19 participants

Contribution type	n	# Words (avg)	# Extracted terms (avg)
Challenge	104	108.3	17.6
Solution	129	122.95	19.3
Comment	231	20.83	7.2
Tag	1587	n/a	n/a
Rating	629	n/a	n/a

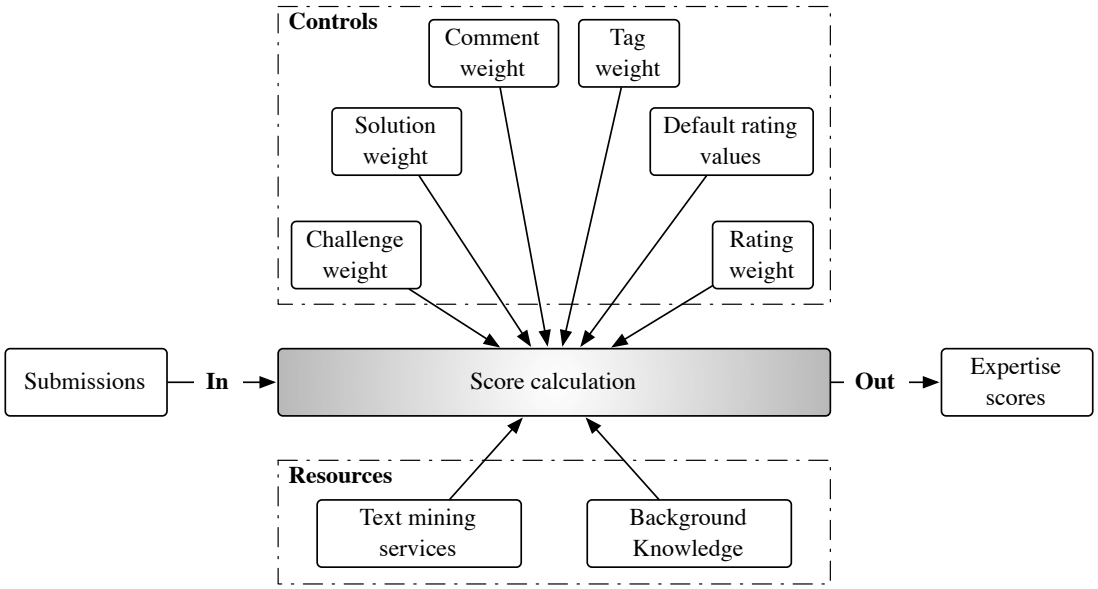


Figure 6.4: Variables affecting expertise score calculation.

6.2 Prediction Accuracy

To examine the accuracy of expertise predictions, we explore the effect of contribution weights and default rating values (the independent variables) on the calculated expertise scores (the dependent variable), if there is one anyway. Figure 6.4 illustrates the relations between these variables. In this section, we mainly aim to determine the influence of the various independent variables on the deviation of predicted scores from participants' self-assessments. For instance, the examination results may reveal that certain contribution types may prove to be a better source for expertise predictions than others. As for default rating values, we may discover that they cause an overestimation of expertise scores that contradicts with the original intent to follow a pessimistic approach.

6.2.1 The Influence of Single Contribution Types

We measure the accuracy of predictions by calculating the correlation between participants' self-assessments and the algorithm's calculated scores across all participants. Basically, contribution weights range from 1 to 5, recall Section 3.2.2. To begin with, we analyze the influence of each single contribution on the predicted scores separately. Table 6.2 shows the weight settings we applied to test a contribution's individual influence on score accuracy. We hypothesize that:

> *Certain contribution types have a stronger influence on score accuracy than others.*

Table 6.2: Weight settings to determine single contributions' effect

Setting	ω_{Ch}	ω_S	ω_{Co}	ω_T	ω_R
1	5	1	1	1	1
2	1	5	1	1	1
3	1	1	5	1	1
4	1	1	1	5	1
5	1	1	1	1	5

The results of 1060 expertise scores calculated on these weight settings are shown in Table 6.3. The first three columns Mean, Median and SD refer to the absolute deviation of predicted scores from self-assessed scores. We observe rather low variation concerning the mean and standard deviation values. The median, though, presents a different picture showing that the first three weight settings cause a much smaller deviation from self-assessments than the last two settings.

The next two columns in Table 6.3 represent results from correlation analysis and linear regression. The Pearson correlation coefficient measures the degree to which self-assessments and predicted scores are associated. In addition to correlation, we performed linear regression, which determines the linear line that best fits the points on the graph of average score predictions. The column named R.M.S. shows the values of the residual standard error describing the spread from the regression line.

Interestingly, setting 2 specified to test the relative importance of solutions to score calculation, yields the highest correlation coefficient as well as the least score deviation (Median: 20 points). As for setting 5, we observe a significant lower association between self-assessments and predicted scores in contrast to all other settings. Setting 5 is designed to examine the effect of ratings on calculated scores. In combination with the median value for setting 5, our preliminary results suggest that ratings are less valuable for reliable expertise calculation than other contribution types. In the next section, we will proceed with testing different combinations of weights that may increase score accuracy and correlation respectively.

We proceed with exploring possible trends of expertise predictions, meaning whether our calculation approach tends to either under- or overestimates participants compared to their self-

Table 6.3: Accuracy of expertise scores calculated with weight settings from Table 6.2

Setting	Mean	Median	SD	Correlation	R.M.S.	n
1	27.24811	21	19.72645	0.2835136	27.18037	1060
2	25.19528	20	20.56062	0.3298118	26.75745	1060
3	25.75283	21	18.30301	0.2687011	27.30099	1060
4	29.43868	26	19.65210	0.2243468	27.62086	1060
5	29.11509	27	19.28216	0.0243724	28.33493	1060

Table 6.4: Trends of expertise scores calculated with weight settings from Table 6.2

Setting	# Correct	# Underestimates	# Overestimates	# False positives
1	29	664	367	287
2	35	543	482	287
3	40	602	418	287
4	25	715	320	287
5	38	486	536	287

assessments. Table 6.4 shows the results of this analysis. We see that predictions mainly underestimate participants. Basically, the level of predicted scores is limited by the rating value associated with the contribution upon score calculation, see Equation 3.1. Ratings associated with challenges and solutions originate from peers whereas comments, tags and ratings can not be rated per se. For the latter, we carefully chose default values (pessimistic approach) with the aim to prevent overconfident predictions. Thus, raising these default rating values represent a way to possibly increase predicted scores in general. This may result in a more uniform distribution of predictions, i.e., a distribution where the numbers of under- and overestimates are almost equal.

As for ratings associated with challenges and solutions, we set a minimum number of peers which have to rate these contributions, otherwise we use default rating values for calculation. At the moment, the minimum raters count is set to 2, i.e., at least two peers have to rate the contribution. From the 1060 predicted expertise scores in our sample, 707 (67%) were calculated based on default ratings (less than 2 peer votes). 81 topics received only one vote. The remaining 353 topics were rated by 4.84 peers on average . We do not expect a significant boost by lowering the required number of minimum raters since this would only bring 81 more votes and besides, these votes would reflect the opinion of only a single peer. It is highly doubtable, whether a single peer is able to provide reliable estimates anyway, confer Section 2.2.2. Hence, given the data we collected, we later proceed by experimenting with default rating values rather than relying on a single peer's vote.

False positives, as listed in the last column of Table 6.4, refer to topics in which participants believe to have no expertise at all. Thus, 27% (287) of the topics were wrongly assigned to participants models. In the first step of our mining approach, terms are extracted from contributions by means of text mining techniques. Therefore, false positives are clearly related with this particular part of the algorithm. Furthermore, considering the contribution types from which false positives primarily originate, we see that, on average, false positives are associated with ratings (1.955 ratings per false positive), comments (0.5889), solutions (0.5157), challenges (0.4913) and tags (0.06272). As we described in Figure 3.7, in order to calculate expertise scores based on ratings, they are associated with the contribution body being rated. We have to further examine whether this association is indeed supportive or rather introduces undesired bias. Finally, it should not go unmentioned that we also have to deal with subjective bias inherently coupled with false positives.

Table 6.5: Accuracy of expertise scores calculated on the reduced data set

Setting	Mean	Median	SD	Correlation	R.M.S.	n
1	30.91850	30	19.59347	0.3276332	18.99719	773
2	25.11384	20	19.09107	0.3664084	18.70864	773
3	26.79819	24	18.05219	0.2766072	19.32249	773
4	35.20569	35	19.07024	0.2747977	19.33292	773
5	23.92109	20	16.20546	0.1397165	19.90978	773

Table 6.6: Trends of expertise scores calculated on the reduced data set

Setting	# Correct	# Underestimates	# Overestimates	# False positives
1	29	664	80	0
2	35	543	195	0
3	40	602	131	0
4	25	715	33	0
5	38	486	249	0

Since topics identified as false positives do not bear any valuable information regarding the evaluation of prediction accuracy, we eliminate these topics from the sample data. Table 6.5 shows the results calculated on the reduced data set. As a consequence of reducing the data, the correlation values improve significantly compared with the previous ones in Table 6.3. We can also recognize a remarkable decrease of the residual standard error. However, we lost a few topics after cleaning the data. The new data sample includes 773 topics. Looking at the trend of predicted scores based on the new data set, see Table 6.6, we observe a much clearer trend towards underestimation than our previous trend results have shown. Across all settings, the number of underestimated scores surpasses the amount of overestimates.

Table 6.7 shows the average score deviation of expertise predictions from self-assessments for each individual participant. One of the participants differs significantly from the other participants. Descriptive statistics deliver a standard deviation $SD = 4.54$ given an average mean of 22.17 and quartiles of $Q_1 = 20.34$, $Q_2 = 21.48$, $Q_3 = 24.84$. Commonly used methods to detect outliers like the interquartile range (considers only the data located within the range of Q_1 and Q_2) or the three-sigma rule (values that are around 3 standard deviations away from the mean are referred to as outliers), more or less suggest to eliminate participant 5 from the data set.

The recalculation of correlation values while ignoring the scores associated with participant 5 yields even better results compared with the previous ones in Table 6.5. While the trend to underestimation is still present, the correlation and R.M.S. considerably improve as all other values do equally, see Table 6.8. However, we are aware that in our particular context, we can not raise a grounded assumption that any of the participants self-assessments are more valid than

Table 6.7: Detecting outliers amongst the participants

Participant	Avg score deviation
1	24.2504
2	20.2525
3	21.4819
4	20.7395
5	32.1614
6	20.4281
7	22.5139
8	22.4592
9	28.3950
10	21.2994
11	16.2415
12	15.1835
13	21.1778
14	25.4236
15	25.5143
16	27.9154
17	14.4450
18	18.0250
19	23.3996

Table 6.8: Accuracy of expertise scores ignoring participant 5

Setting	Mean	Median	SD	Correlation	R.M.S.	n
1	29.81844	28.0	18.95701	0.3587696	18.48452	716
2	24.34218	20.0	18.62073	0.4000394	18.14929	716
3	25.90084	22.0	17.63563	0.3001260	18.88995	716
4	34.42598	33.5	18.61953	0.2891170	18.95717	716
5	22.93855	20.0	15.34158	0.1694566	19.51648	716

those of others. Even though we decided to continue our evaluation by excluding participant 5 from the data set. Thus, the new data set contains 716 items.

So far, we found that contribution types differ in their value for reliable score calculation. In particular, ratings seem to have the least influence on score accuracy. Concerning the sample data, we eliminated scores classified as false positives as well as one specific participant, who significantly differs to the rest of the participants regarding the deviation of self-assessment from score predictions. All further calculations are based on this cleaned data set.

6.2.2 Combining Contribution Types

After exploring the influence of individual contribution types on expertise predictions, we now analyze their effect on score accuracy when we combine them. We hypothesize that:

> *The combination of contribution types leads to a higher score accuracy than considering contribution types separately.*

We defined 50 different combinations of contribution types to test this hypothesis. The impact of each contribution type on the resulting scores is determined by its respective weighting. We carefully chose the weight levels across the various settings. That is because we need to make sure that contribution types are considered equally across all settings. We do not want to favor a particular contribution type over another, i.e., a specific contribution type is steadily higher weighted than others.

To ensure a uniform distribution of weight levels across our test settings, we calculated the average mean for each single weight as shown by the columns 2-6 in Table 6.9. This table lists a subset of the total settings. To get the details on all settings, please refer to Table A.1 in the appendix. The average mean calculated on each single weight column yields 3.3 for weight ω_{Ch}, 3.4 for ω_S, 3.44 for ω_{Co}, 2.82 for ω_T and 2.68 for ω_R. Although the average means of contribution weights are rather close, we set a focus on our previous results obtained in Section 6.2.1, which suggest that solutions, challenges and comments may have a stronger influence on score accuracy than other contribution types.

After calculating the scores for each of the 50 settings, we sorted the results by the correlation coefficient in descending order. Table 6.9 shows the weight settings yielding the highest correlation values (Top-10). We observe that all correlation values from the top-10 ranked weight settings exceed the best correlation value obtained while considering the contribution types separately, confer Table 6.8. Similarly, the average deviation from the regression line also improves across all settings when using combined weights.

The results from initial weight analysis in Section 6.2.1 already suggested that certain contribution types may be more valuable than others for expertise score calculation. However, except for ratings, we were not able to state a valid claim regarding the order of contribution types sorted by their importance to accurate score predictions. The difference between the results obtained with initial settings in terms of score correlation was just too little. With our current data on hand based on 50 different weight settings, we will explore our initial thought once more and hypothesize that:

> *A particular assembly of contribution weights characterizes both high and low score accuracy.*

Table 6.9: Top-10 ranked weight combinations yielding highest score accuracy

Rank	ω_{Ch}	ω_S	ω_{Co}	ω_T	ω_R	Correlation	R.M.S. error	n
1	5	5	4	3	2	0.4251923	17.92364	716
2	5	5	3	2	1	0.4246628	17.92856	716
3	4	5	3	2	1	0.4244265	17.93076	716
4	4	5	3	1	1	0.4236703	17.93777	716
5	4	4	3	3	1	0.4232447	17.94171	716
6	4	5	2	1	3	0.4223023	17.95041	716
7	5	5	4	4	1	0.4215501	17.95735	716
8	5	5	4	2	1	0.4205940	17.96614	716
9	5	5	4	1	1	0.4197691	17.97370	716
10	4	5	3	1	3	0.4173946	17.99537	716

Table 6.10: Average weights for top-ranked and lowest-ranked weight settings

Top-10	ω_{Ch}	ω_S	ω_{Co}	ω_T	ω_R
Median	4.5	5.0	3.0	2.0	1.0
Mean	4.5	4.9	3.3	2.0	1.4
Lowest-10	ω_{Ch}	ω_S	ω_{Co}	ω_T	ω_R
Median	2.0	1.0	3.0	4.0	5.0
Mean	2.2	1.6	3.1	3.6	4.6

We calculated the average weight values for each contribution type of the 10 top-ranked weight settings to test our hypothesis. In addition, we also built the individual weight averages of the 10 lowest-ranked settings. The results are shown in Table 6.10 and indicate a rather clear order of weights contributing best to accurate score predictions. In contrast, the order gained from the averages of the lowest-ranked settings just acknowledges the obtained weight order for best score predictions.

Consequently, these results suggest a clear ranking about which contribution is more valuable than others. Equation 6.1 depicts the weighting rule we derived from our empirical data set supporting accurate expertise predictions.

$$\omega_S \geq \omega_{Ch} > \omega_{Co} > \omega_T > \omega_R \tag{6.1}$$

Scores predicted within the top-ranked weight settings clearly underestimate (100%) participants' expertise as shown in Table 6.11. When considering the complete set of weight settings, we observe a similar trend where 86% of predicted scores underestimate people.

We now seek to determine possible reasons for this particular behavior of under- and overestimation. To begin with, we investigate the origin of topics that finally received underestimated

Table 6.11: Trends of expertise scores calculated on the top-10 ranked weight settings

Rank	# Correct	# Underestimates	# Overestimates	n
1	31	394	291	716
2	36	442	238	716
3	38	446	232	716
4	38	446	232	716
5	42	521	153	716
6	60	375	281	716
7	35	416	265	716
8	34	418	264	716
9	34	418	264	716
10	59	370	287	716

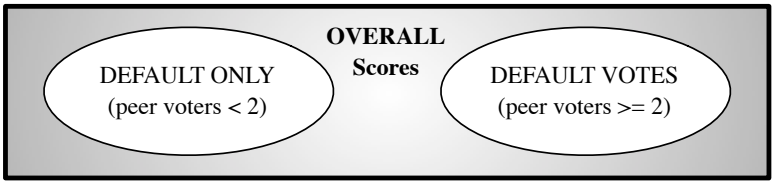


Figure 6.5: Three different setups regarding the use of rating values.

scores. In addition, we examine how many peers were involved in rating these underestimated topics. Ratings have a major influence on the levels of calculated expertise scores. Whenever a topic's score is determined the system either takes a rating value given by peers or a predefined default rating value to calculate an intermediate score. To systematically explore the way how the system is using these rating data, we need to distinguish various calculation setups as illustrated in Figure 6.5. In the DEFAULT ONLY configuration we just consider the set of scores calculated on default rating values. Expertise scores determined based on ratings from peers (concerning challenges and solutions) and default rating values used for comments, tags and ratings are covered by the DEFAULT VOTES setup. The aggregated set of scores is represented by the OVERALL setup.

Approaches designed to exploit peer rating data often suffer from the sparsity of ratings [Sun et al., 2009] [Manouselis et al., 2011]. We hypothesize that:

> *The use of default rating values substituting missing rating data contributes positively to overall score accuracy.*

Table 6.12: Score accuracy based on different setups (average values based on Top-10 ranks)

Setup	Corr	R.M.S.	Score Dev	# Correct	# Under	# Over	n
	max	median	median	mean	mean	mean	
OVERALL	0.4252	17.95	16	40.70	424.6	250.7	716
DEFAULT ONLY	0.3353	18.31	20	33.40	302.8	103.8	440
DEFAULT VOTES	0.3822	17.13	13	7.60	134.6	133.8	276

Table 6.13: Average amount of contributions behind score tendencies (OVERALL)

Correct predictions (contribution average means)					
Weight setting	Challenge	Solution	Comment	Tag	Rating
Rank 1	2.774	1.548	0.7419	0.03226	2.065
Rank 1-50	2.128	1.264	0.8287	0.08512	2.046
Underestimates (contribution average means)					
Weight setting	Challenge	Solution	Comment	Tag	Rating
Rank 1	1.345	1.325	1.099	0.1091	2.584
Rank 1-50	1.651	1.665	1.174	0.1321	2.55
Overestimates (contribution average means)					
Weight setting	Challenge	Solution	Comment	Tag	Rating
Rank 1	1.718	1.88	1.234	0.189	2.436
Rank 1-50	1.29	1.413	1.123	0.1588	2.486

We calculated score accuracy figures for all setups presented in Table 6.12. Calculations are based on the 10 top-ranked weight settings. The best correlation coefficient is gained in the OVERALL setup. Looking at the numbers of underestimated scores we realize that the most unbalanced ratio occurs in the DEFAULT ONLY setup whereas the DEFAULT VOTES setup shows a nicely balanced distribution of under- and overestimates. The clear trend to underestimation in the DEFAULT ONLY setup suggests to increase default values to gain a better balance figure.

Expertise scores are associated with actual contributions of various types. Table 6.13 shows the average amount of contribution types associated with correct, underestimated and overestimated predictions considering the OVERALL set of scores. We calculated these figures once for the top-ranked weight setting (Rank 1) and once over the total set of weight settings (Rank 1-50). The figures are read as follows: For instance, an underestimated topic originates, on average, from 2.6 ratings. As for correct predictions, they mostly origin from 2.8 challenges. We have to carefully interpret figures related to correct predictions since these can change quickly as soon as scores vary by only one point up or down (correct predictions represent exact score matches with respect to users' self-assessments).

Table 6.14: Average votes per predicted expertise score (topic)

Weight setting	Correct		Underestimates		Overestimates	
	Median	Mean	Median	Mean	Median	Mean
Rank 1	0.0000	0.7097	0.00	1.51	2.000	2.832
Rank 1-50	0.000	1.083	0.000	1.971	1.000	2.245

Table 6.15: Total number/percentage rate of expertise scores calculated with default rating values

Weight setting	Correct	Underestimates	Overestimates
Rank 1	26 / 84%	280 / 71%	134 / 46%
Rank 1-50	1533 / 78%	13720 / 62%	6747 / 57%

When we take a closer look at the dominant source of underestimated scores, we realize that these scores are mostly associated with topics originating from ratings. Once participants rated a contribution, for instance a peer's challenge, they are associated with the content of this challenge via their rating action, see Figure 3.7. Interestingly, predictions that overestimate people as well as correctly assess them considerably origin from ratings as well. The figures associated with solutions, comments and tags do not significantly change across tendency classes. In principle, this is also true for challenges, except for the figures representing correct estimates.

Default rating values are used for expertise calculation if the given contributions are rated by less than two peers or can not be rated at all (in case of comments, tags and ratings). Table 6.14 shows the average amount of peer votes associated with expertise topics for each tendency class. Predicted scores overestimating people are associated with approximately three peer votes. Predictions either matching self-assessments exactly or underestimate participants are largely calculated based on less than two votes. Consequently, correct and underestimated scores are mostly calculated on default rating values rather than peer votes. Table 6.15 depicts the number of scores only calculated on default rating values classified by score tendency. The figure for underestimations supports our previous observation that underestimated scores are mainly calculated on default rating values, i.e., almost 3 of 4 scores. From this view and considering previous analysis, it seems promising to increase the default rating values for topics originating from ratings to boost scores that are currently underestimated. So far, the default rating value was set to 1. That means, a candidate topic can develop a maximum value of 50 points during expertise calculation (pessimistic approach). The maximum default rating value is 2, which allows expertise scores up to 100 points. By increasing the default rating value we expect a more uniform distribution of predicted scores amongst the underestimate/correct/overestimate classes. In particular, the scores currently belonging to the classes correct and underestimate may partly move to the overestimate class.

We test the effect of altering the default rating value for ratings on predicted scores based on the top-ranked weight setting in Table 6.9. The default rating value is changed to 1.5. Con-

Table 6.16: The effect on score accuracy while testing different default values for ratings

Default value	Mean	Median	SD	Corr	R.M.S.	#Correct	#Under	#Over
1	18.46	13.0	15.35	0.4252	17.92	31	394	291
1.5	17.35	13.0	14.56	0.4197	17.97	50	344	322

sequently, topics originating from ratings can now get scores up to 75 points. Table 6.16 shows the results calculated with the increased default rating value. In fact, previously underestimated scores move to both the correct and the overestimate class, implying a more uniform tendency of score predictions.

Default rating values are supposed to substitute missing peer ratings. Our intention is not to change default rating values until we find the best configuration for the collected data, instead the experiment demonstrates that default values represent a viable option to optimize the algorithm's accuracy. Learning approaches may adapt default rating values automatically, e.g., by exploiting users' relevance feedback given by their self-assessments.

6.2.3 Prediction Accuracy in Different Prediction Score Ranges

We determined the quartiles of predicted score data ($Q_1 = 25.75$, $Q_2 = 40$, $Q_3 = 64$) and split the data according to these quartiles into four subsets. We now aim to explore the accuracy of predictions in different score ranges for each of the data subsets. Even though the correlation values are rather small, they steadily increase when moving from one quartile to the next one: $r(Q_1) = 0.0247, r(Q_2) = 0.0721, r(Q_3) = 0.1194, r(Max) = 0.1454$. Splitting the data into two equal halves results in much higher correlation value suggesting that the accuracy of predictions is significantly higher in the upper half of score levels than in the lower half: $r(Q_2) = 0.1149, r(Max) = 0.3940$. Figure 6.6 shows a scatterplot of score predictions with their associated self-assessments for the first half of the data (left side) as well as for the second half (right side). For both charts, the red sketched line approximates the association between self-assessments and score predictions whereas the linear black line represents the regression line.

6.2.4 Accuracy of Newly Generated Expertise

A contribution consists of terms that we interpret as indicators of the author's expertise. While calculating expertise scores for topics gained from mining texts of contributions, only a part of these scores are finally selected to represent an individual's expertise. However, contributions are not the only source of candidate expertise topics. During score propagation we generate new topics in which users' may have expertise. It is notable that only the least common ancestors of those topics that serve as the input for score propagation are promoted to expertise candidate status. We examined the score accuracy of topics gained from determining least common ancestors. On average, models contained two topics newly generated during score propagation (average model size: 40 topics). Even though we have rather small data on hand (36 items), the

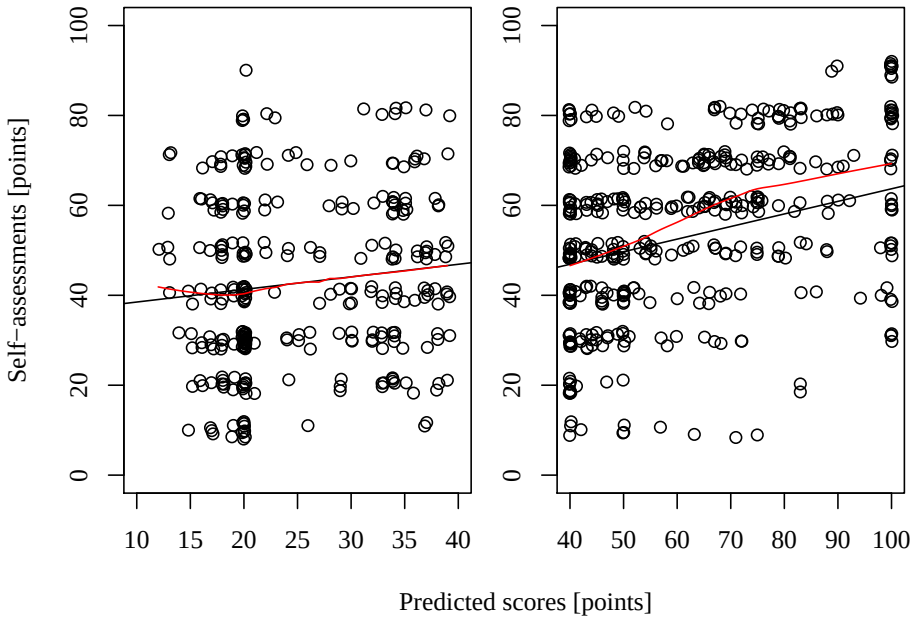


Figure 6.6: Score accuracy in different score ranges.

score accuracy results show a similar picture compared with the results of the top-ranked weight settings in Table 6.9. Namely, most of the 10 top-ranked weight settings comply to the contribution weighting rule proposed in Equation 6.1. With regards to score tendencies, we observe a clear trend to underestimation for all weight settings. Although correlation values are moderate compared with those obtained from calculation over the full set of predictions (Table 6.9), the average score deviation regarding topics originating from least common ancestors amounts to 20 points with a standard deviation of approximately 17 points. For full details on score accuracy results refer to Table A.2 in the appendix.

6.3 Reliability of Expertise Predictions

As introduced in Section 3.2.4, the overall measure for confidence calculation is composed of two sub-measures each having its own pattern for score prediction reliability. One sub-measure assumes that only people who are themselves top experts in the given topic can evaluate (via voting) the top expertise of others regarding this topic. That means, the higher the raters' expertise, the higher the algorithm's confidence in its score predictions based on raters' evaluations. The second sub-measure's follows the premise that the higher the variety in users' submissions (in terms of submitting different kinds of contribution types), the more reliable are the calculated expertise scores. We defined the coefficient λ to control the balance between these two patterns as illustrated by Figure 6.7. The goal of the present section is to examine the validity of the proposed overall confidence measure as well as the performance of each single sub-measure. In general, we pursue the common-sense assumption for valid confidence levels that reads as

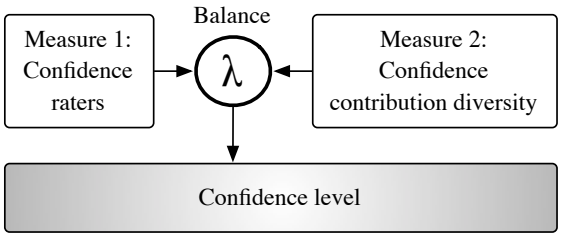


Figure 6.7: Relationship between independent confidence measures and overall confidence.

follows:

> *The lower the score deviation, the higher the reliability of predicted scores.*

To test whether the competence measure complies with this assumption we determined confidence levels for score predictions calculated on the top-ranked weight setting as displayed in Table 6.9. Default rating values for all contribution types are set to 1. We performed score calculations with varying values for λ ranging from 0 to 1 (in steps of tenths). We aim to explore the combined effect of the two sub-measures as well as their individual impact on overall confidence levels. Regarding the cooperative effect, the optimum balance aligned by the balance factor λ will correlate with the highest negative correlation coefficient determined between the overall confidence level and the absolute score deviation. Furthermore, we test the validity of confidence levels in various ranges. We demand negative linearity concerning the relation between score deviations and confidence levels. Therefore, we divided the scale of score deviations into three parts and calculated average confidence levels for each of these parts. As shown in Equation 6.2, we expect that the confidence average mean of the first third is greater than that of the second third and that the average mean of the second third exceeds the mean of the last third.

$$conf_{Third\,1} > conf_{Third\,2} > conf_{Third\,3} \tag{6.2}$$

The maximum score deviation in the sample data amounts to 70 points. Thus, we defined the first range to include score deviations up to 20 points that corresponds to practically the first third of the scale ranging from 0 to 70 points. The second range was set to hold score deviations from 21 to 40 points (the second third) and lastly, the third range contained topics with score deviations reaching from 41 to 70 points (the last third). Table 6.17 shows the calculation results. According to the correlation coefficients at varying balance factors, it is clear to see that there seems to be no association at all between the variables score deviation and confidence level. In addition, the confidence average means for different ranges of score deviations do not follow the expected behavior as expressed in Equation 6.2. The analysis of confidence levels only calculated on default rating values as well as only considering peer votes led to the same result. However, while experimenting with certain variables, we found that confidence levels

Table 6.17: Correlation Score Deviation/Confidence, @self-assessments > 0, n=716

λ	Correlation	Mean Conf Third 1	Mean Conf Third 2	Mean Conf Third 3
0.0	-0.06	40.12	40.23	33.34
0.1	-0.06	37.70	37.83	31.53
0.2	-0.05	35.35	35.54	29.78
0.3	-0.05	32.93	33.17	27.95
0.4	-0.04	30.36	30.70	26.12
0.5	-0.03	28.00	28.41	24.36
0.6	-0.03	25.62	26.07	22.61
0.7	-0.02	23.23	23.71	20.84
0.8	-0.01	20.87	21.41	19.05
0.9	0.00	18.48	19.09	17.27
1.0	0.00	16.23	16.88	15.59

Table 6.18: Correlation Score Deviation/Confidence, @self-assessments > 75, n=79

λ	Correlation	Mean Conf Third 1	Mean Conf Third 2	Mean Conf Third 3
0.0	-0.42	64.61	54.50	33.71
0.1	-0.44	62.20	51.07	31.07
0.2	-0.46	59.94	47.86	28.43
0.3	-0.47	57.67	44.50	25.79
0.4	-0.49	55.33	40.93	22.79
0.5	-0.50	53.00	37.64	20.14
0.6	-0.50	50.78	34.36	17.50
0.7	-0.50	48.43	30.93	14.93
0.8	-0.50	46.29	27.86	12.29
0.9	-0.50	43.98	24.50	9.64
1.0	-0.49	41.84	21.29	7.07

yet correlate with score deviations, namely, at varying levels of participants' self-assessments. We examined correlation coefficients calculated for subsets of expertise topics. Subsets are built based on the levels of self-assessments of expertise topics starting from 0, 25, 50 and 75 points. Table 6.18 displays the correlation coefficients and confidence average means for self-assessments greater than 75 points. Please refer to the details of correlation results concerning topic subsets starting from 25 (Table A.3) and 50 points (Table A.4) in the appendix.

We observe that the higher the participants' self-assessments the more valid the confidence levels. Compared to the results in Table 6.17, we now obtain correlation coefficients up to $r = -50$ depending on the balance factor λ. Similarly, the confidence average means reflect the

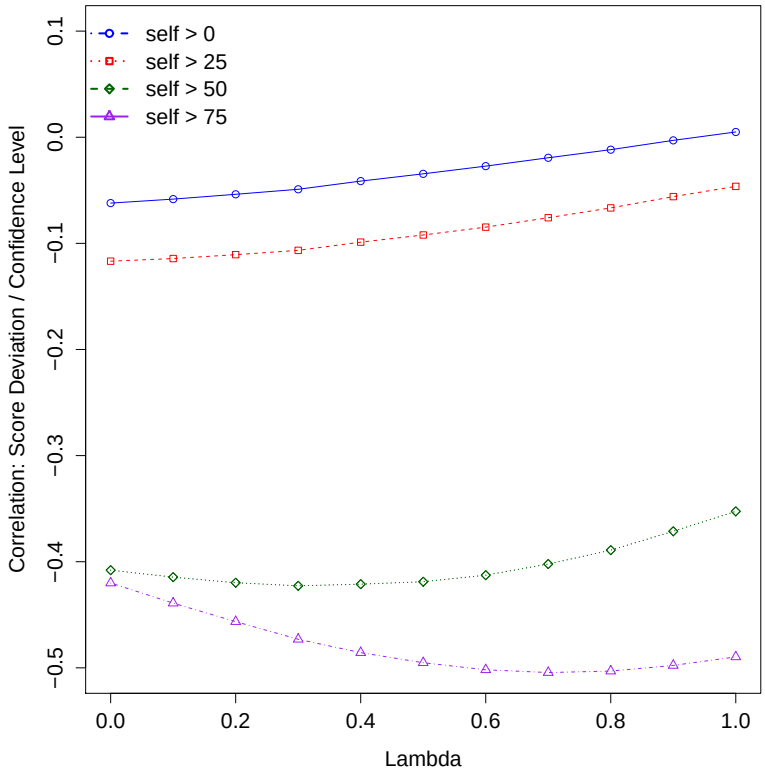


Figure 6.8: Correlation values viewed at different expertise levels and varying balance factor λ.

expected graduation as given in Equation 6.2. Looking at the individual performances of the two sub-measures at $\lambda = 0$ and $\lambda = 1$, we recognize that there seems no clear indication that one of them works better than the other. Figure 6.8 illustrates the development of correlation figures at varying balance factors. It is hard to determine which individual sub-measure outperforms the other. This is not only because of the small numerical difference between correlation figures. But also because of the fact that with increasing levels of self-assessments the measure based on raters' expertise seems to perform better than the diversity measure. However, the latter measure performed better when comparing at lower self-assessments. Searching for the optimum setting of the balance factor λ, the correlation figures show a slight improvement when combining the two sub-measures, however, this very small improvement is practically equal with the performance at $\lambda = 1$.

Besides correlation analysis, we examine whether the validity of confidence levels depends on the contribution types the topics originate from. That means, we may possibly learn that high confidence levels often occur with expertise scores originating from particular contribution types. Table 6.19 shows the average number of contribution types associated with confidence levels for each of the three score deviation ranges. The ratio of the amount of contributions within a range do not change significantly. We observe once more that ratings are considerably high in numbers even though they do not contribute to higher correlation values. Looking across

Table 6.19: Average number of originating contribution types for each score deviation range

Score Deviation Range	Challenge	Solution	Comment	Tag	Rating
First third	1.619	1.665	1.217	0.1313	2.379
Second third	1.494	1.475	1.062	0.1605	2.753
Top third	1.311	1.068	0.7973	0.1351	2.743

the ranges, contribution types essentially tend to lessen in number, except for ratings. However, this is natural since the higher the ranges the smaller the data sets which implies a lower number of total contributions left in the top third range. Thus, based on the collected data, we can not find any evidence that confidence levels correlate with the origin of expertise scores.

The confidence sub-measure relying on raters' expertise does not consider topics originating from comments, tags or ratings. That is because none of these contribution types can be qualified by peer votes. Thus, the confidence levels of topics originating from these contribution types are only calculated by means of the diversity confidence measure. Since the confidence diversity sub-measure builds the weighted average of the types of contributions submitted by participants, confidence levels originating only from comments, tags and ratings show consistently low values. This is especially true for ratings that have the lowest weight assigned. Therefore, topics exclusively calculated on ratings receive very low confidence levels (on average 3%, n=117). To recall our initial assumption we pursue, i.e., the lower the score deviation the higher the confidence level, we realize that there might be a slight contradiction in interpreting confidence. On the one hand we expect higher confidence on low score deviations. On the other hand, though, according to our observations regarding the confidence of topics calculated on ratings, it makes sense that these topics basically have low confidence levels irrespective of their score deviations since ratings are just not as reliable as other contribution types. This is especially obvious from our previous results that repeatedly suggest a low value of ratings in calculating the raters' expertise.

To analyze the possible effect of this reflection on our previous results, especially on our claim that the overall confidence measure is particularly viable when measuring high expertise scores, we recalculated the correlation between the score deviation and confidence level. To be more specific, we determine the correlation coefficients only based on comments, tags and ratings, and for self-assessments greater than 75 points. The results show lower correlation coefficients (around $r = -0.38$, n=26) than for confidence levels calculated on challenges and solutions, for details refer to Table A.5 in the appendix. However, compared with correlation coefficients based on self-assessments less than 75 points, we measured a significant higher correlation. As a consequence, given the yet high correlation coefficient and given that only one third of confidence levels are based on non-rateable contributions (26 of total 76 total, confer Table 6.18), we rule out that our previous results might be distorted significantly and thus, are still valid.

6.4 Quantities of Contributions

Users differ in the amount and extent of contributions they submit to an online community. In this section, we examine whether the amount of words contained in participants' contributions correlate with both score accuracy and confidence levels.

6.4.1 Effect of Word Quantities on Score Accuracy

Expertise topics originate from one or more contributions as illustrated on the left side in Figure 6.2. With regards to the amount of words these contributions are built on, we hypothesize that:

> *The higher the number of words supporting expertise score calculation, the higher the score accuracy.*

In order to test this hypothesis we need to determine the amount of words behind individual topics. Therefore, we add up the words of a topic's associated contributions including challenges, solutions, comments, tags and ratings. From the resulting data set, we eliminate topics that were newly generated during score propagation since they are not related to participants' original contributions. The remaining data set consists of 680 topics. On average, a topic is associated with 553 total words (median: 340, max: 2953) and 92 extracted words (median: 55, max: 521). By *total words*, we mean all words contained in a contribution. Those words, which remain after text mining is applied, are denoted as *extracted words*. We now calculate Pearson's r to evaluate the correlation between the amount of words and the score accuracy. Score accuracy is expressed by the score deviation. The correlation coefficients results in $r = 0.008$ concerning total words and $r = 0.005$ in terms of extracted words. These results suggest that, based on the collected data, the amount of words does not affect score accuracy as also shown on the left side in Figure 6.9.

Our previous results indicate that ratings provide no vital support to valid score calculation. Thus, we were interested whether the correlation coefficients change if we exclude topics originating from ratings. In this context, we refer to ratings from the perspective of the rater, who is associated with terms of the rated contribution. Excluding topics originating exclusively from ratings reduces the data set to 563 topics. However, even if correlation is calculated on this reduced data set (total: $r = 0.043$, extracted: $r = 0.041$), there is still no indication that the amount of contributions' words have any effect on score accuracy.

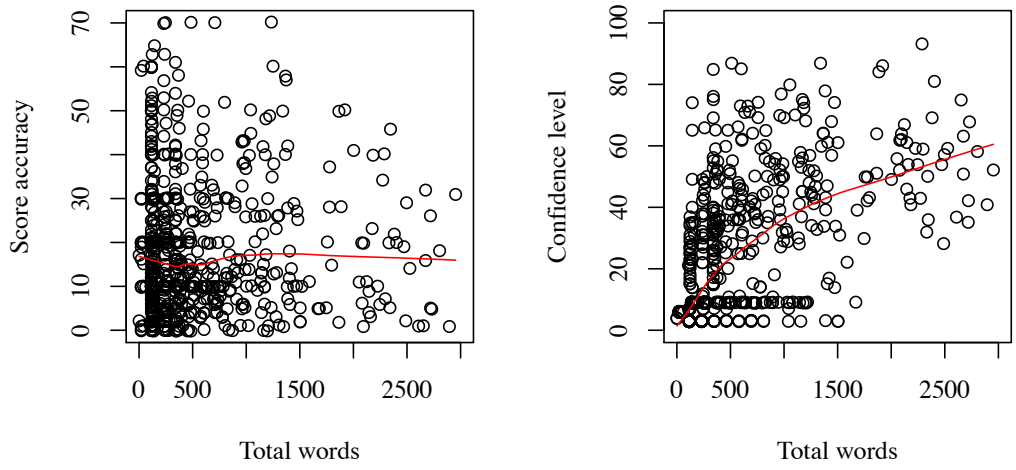


Figure 6.9: Correlating the number of words behind a topic with score accuracy and confidence level.

6.4.2 Word Quantities and Confidence Levels

As for the possible influence of contribution quantities on confidence levels, it seems obvious that the more we know about our participants by means of their contributions and especially the words contained in this contributions, the more confidence we have in calculating their expertise scores. Hence, we hypothesize that:

> *The more words available for score calculation, the higher the confidence in these scores.*

The confidence level represents the reliability of its corresponding expertise score. This particular link indirectly connects the confidence level to the expertise topic and also to the contributions behind this topic as illustrated in Figure 6.2. Once more, we calculate the amount of words behind each topic and examine correlation coefficients. The scatterplot on the right in Figure 6.9 shows a positive correlation between the two variables. Correlation coefficients amount to $r = 0.588$ for total words and $r = 0.586$ for extracted words. The curve (red line) shows the approximate progression of correlation. Taking a closer look at the scatterplot, we see that some of the lower confidence levels stay unchanged at an increasing number of words. More specific, we refer to confidence levels with values of 3% and 9%. These confidence levels correspond to expertise scores that are either calculated exclusively on ratings (3%) or on the combination of comments and ratings (9%), in other words, on contribution types that can not be rated by others. As these confidence levels distort the correlation value we eliminate them and redo calculation. The newly calculated correlation coefficients are lower than the previous

ones, i.e., $r = 0.407$ (total words) and $r = 0.410$ (extracted words). Even though they still suggest a positive linearity between the amount of words and the confidence level.

Recalling the aims of the two confidence sub-measures, which together determine the overall confidence level as shown in Figure 6.7, the *confidence diversity* measure is designed to operate on contribution types rather than actual contribution instances. Consequently, a higher amount of equal contributions and words respectively will not lead to a higher confidence level. In this sense, the diversity measure does not contribute to the positive correlation our data is suggesting. Therefore, to explain an actual relationship between word quantity and confidence, we need to examine the second confidence measure, which incorporates the expertise levels of raters. The *confidence raters* measure calculates confidence for topics extracted from challenges and solutions. This is because only these particular contribution types are associated with peer votes, which are finally used to calculate confidence, confer Figure 3.8. Hence, we determine the correlation coefficients including confidence levels that are only based on challenges and/or solutions. Prior to this, we need to eliminate those confidence levels from the data set that are solely calculated on non-rateable contribution types. The remaining data set consists of 307 topics. Newly calculated correlation results in $r = 0.407$ (total words) and $r = 0.411$ (extracted words). These results show that previous correlation values are practically not affected even if we completely remove topics originating from non-rateable contributions.

These results are somehow surprising since an obvious relation between the amount of words and calculated confidence levels does not exist. There is an indirect connection between contributions and confidence levels, though, when following the link chain as depicted in Figure 6.2. But this indirect linkage does more reflect the common feature of having a relation to the expert topic rather than having a relation among each other. Figure 3.8 illustrates that the confidence levels in Jane's expertise scores depend on peers' expertise levels. The higher their expertise, the higher the scores' confidence levels. In turn, peers' expertise levels depend essentially on ratings (either from peers or specified by system's default values) and terms extracted from their individual contributions. Despite the fact that peers' contributions are not associated with Jane's contributions and thus not relevant to our correlation analysis, the particular amount of extracted words does not influence peers' expertise levels either, confer Equation 3.4. No matter how we look at it, even if the collected data in fact suggest a correlation between word quantities and confidence levels, we are not able to find a profound reason to argue for this.

6.5 Participants' Feedback

Basically, we provided a knowledge sharing platform to participants in each individual experiment including the feature to generate their expertise model calculated on their shared contributions. We collected feedback from participants at the end of each experiment. Given the various aspects of our experiments with regards to the approach of expertise measurement, the closing feedback forms slightly differ from experiment to experiment. In this section we primarily focus on the results concerning the last of our three experiments, which the current chapter is all about. However, where applicable, we will relate the current feedback results to participants' response collected in previous experiments as well.

6.5.1 Sharing Expertise Models

We asked participants whether they are willing to share their expertise models with peers. In this regard, we summarize the responses to this closed question across all three experiments (56 participants in total). The accumulated result shows that the majority of participants (62%) like to share their model to certain peers. 16% do not want that peers can access their expertise model whereas 22% said they are willing to share their expertise with all peers even if they are strangers.

6.5.2 Contributing to Background Knowledge

The presented expertise calculation method relies on background knowledge represented by an ontology used to refine expertise scores. Against this background, we asked participants if they can imagine to contribute new topics to this ontology. Again, we aggregate the figures from all three experiments to this closed question. 64% said they would like to contribute topics to the ontology.

6.5.3 Discovering Expertise Previously Unknown

The first version of our Expertise Calculator predicted competence fields representing technical expertise on a general level. In the following versions, though, the algorithm calculated fine-grained expertise scores for general topics as well as specific ones. We were interested if participants discovered new expertise they were previously unaware of. The responses to this closed questions were different for the second and third release of the Expertise Calculator. Concerning participants working with the second version, 36 % of 14 participants found new topics in which they yet have experience. In contrast to that, the responses regarding the third version were completely different, namely, 95% of 19 participants said that they discovered new expertise.

6.5.4 Possible Fields of Application

We asked participants how they would estimate the potential benefit of automatic expertise calculation in more practical environments than the experimental setting. Participants perceived

the generation of their individual expertise model especially valuable for the purpose of self-reflection. For personal development it is particularly important that students as well as employees regularly scrutinize their expertise towards future tasks and challenges. Participants' responses indicate that when confronted with the system's beliefs about their expertise, they think more profoundly about their abilities. This suggests that automatic expertise modeling might foster and support metacognitive activities, e.g., scrutinize own expertise levels regarding strengths as well as weaknesses, how is one's expertise comparable to others' and how does individual expertise increase or decline in the future. Selected quotes are:

"[...] a student can compare her knowledge to others - to get an idea of her status."
"[...] also a better picture of her interests (and what is not of her interests at all)."
"It helps to figure out your strengths and weaknesses."
"[...] determine their weak / strength points in a specific domain."
"It's a good way to get an overview where you are with your skills."
"The student could find a competence that he already possessed but didn't realize."
"One could see his/her improvement over time."
"Self-evaluation of employees of companies."
"Rate yourself and try to improve your competences."

It is crucial in professional life to be able to articulate one's expertise as detailed as possible. The better employees are marketing themselves, the greater their chances to work on suitable and interesting tasks as well as getting promoted. In this concern, the expertise modeling approach presented in this thesis may support people to prepare for job interviews and help them constructing their CVs as emphasized by following participants' quotes:

"[...] generating professional profiles."
"[...] to think about your skills can be helpful for working life."
"Discover competences which can be added to a CV."
"Maybe it will help you to write a CV because you will find competences you have not thought of till now."

As mentioned at the very beginning, knowledge has become a vital production factor in todays industries and sustains competitive advantage. To utilize knowledge in a productive way, a company has to make sure that the right knowledge is available at the right place in the right time. Expert finding systems assist knowledge workers in effectively finding other individuals giving them support on solving their problems. We collected various responses from participants indicating the use of automatic expertise models for expert finding tasks as well as for team formation. Selected quotes are:

"[...] as a base for corporate expert finding systems."
"Supporting students when establishing teams for group work."
"Matching people with similar or complementary skills."
"Find students with similar interests."
"Potential could be high in finding matching students for learning groups."

Further responses concern the use of expertise models in order to match individuals with other resources, e.g., jobs, project tasks, call for papers and lectures. Selected quotes are:

"Job search - employee search."
"Mostly discovering the abilities of employees, or would be employees."
"[...] finding matching people for recruiting."
"Possible topics for a thesis, based on experience and current research topics."

It seems common sense that users want to be modeled as accurate and (practically) complete as possible. Today, users are increasingly involved in various online communities sharing different kinds of experience. In this regard, integrating possible expertise evidence from independent communities may constitute a means towards a unified, more complete and reliable expertise model. A few participants' feedback go in this direction:

"It's hard to measure all competencies by just focusing on one forum or platform."
"[...] you have registered at sun, chip.de and other forums and in each forum you can say they shall update your competence profile would be quite useful."
"Autocalculated personal profiles would make a great facebook app."

Automatic generated expertise models serve also as a means to increase personalization as suggested by following selected quotes:

"When used in an online portal such as stackoverflow.com it might identify experts in a subject area or filter the list of displayed threads according to your expertise."
"It could suggest other lectures to improve certain competencies"
"[...] recommend you some course to improve you weak competencies."

Driven by the desire to be accurately modeled as well as to establish positive reputation, we observed that participants are encouraged to share experience among each other. This is emphasized by responses like:

"Motivates students to interact with each other."
"Expertise calculation motivates users to contribute and to learn new concepts."
"It encourages you to share your best ideas/solutions with others (by sharing you gain reputation an you can even state authorship)"
"[...] makes one proud of sharing content and stuff (specially the most-active-user view got me)."

6.5.5 Likes

Participants were further asked to express their likings regarding the expertise score calculation based on their shared experiences. The responded subjects relate more or less to the same issues we already observed in the previous section. However, it seems that there is one issue that participants appreciated the most, i.e., the support of metacognitive activities as indicated by statements like:

"It's kind of fun to test the system and to see which competencies it discovers."
"[...] comparing those competencies to my own expectations and my personal estimation is very interesting."
"It offers the opportunity for evaluation of self-competence."
"I found out so many competences, I haven't recognized until yet."
"Moreover, it gave me a few competencies i forgot to add or i thought they would not be important (on beginner level)."
"To figure out competences I have not thought of."
"I was the first time really thinking about my skills and knowledge in detail."
"While thinking on programming language and my skills in that area I didn't thought on things like compiling [...]that also belongs to that field."
"Calculation of competence values is a nice feature cause it shows relations you might not have been thinking yourself about."

6.5.6 Dislikes and Desires for Improvements

Participants are mainly concerned about a misuse of their expertise models. On the one side, a misuse by peer users which pretend to have expertise or manipulate others' expertise models. On the other side, participants fear misuse by authorities providing the system. For instance, managers in a company that obtain information about their employees from expertise models and their levels of activity to determine candidates for dismissals.

"Automatically generated competence profiles probably shouldn't be used for determining the worthiness of an employee."
"Humans tend to belief in such systems in a way, that they don't questioning the result."
"People could copy information from internet modify a little bit and this is all."
"Profiles could be used for bluffing - a bluff detector could be necessary for usage in open system or commercial systems."
"If I would [...] feel more 'secure' if I would not have to care about my profile or those expertise measurements."
"Used in a company [...] it could also have a negative impact on business culture: if it is known that the system is used for competence mining, it could increase competitive behavior (rating!)"

Furthermore, participants criticize the rigid focus on technical expertise and prefer a more extensive consideration of users' expertise. Selected quotes are:

"[...] showing only technical competences."
"The system emphasizes on professional-, neglecting personal competences."
"Maybe there's [...] a too strong focus on a person's hard skills. [...] the personality of a person and some other skills are undervalued or in the worst case totally forgotten."

Another dislike regards the publication of others' expertise models. Participants want to place themselves in the community, thus they require information about peers' expertise as indicated by:

"I am also interested to know how many users are more competent than i in a specific field"
"I like to compare my profile with the profiles of other students (maybe anonymous)."
"[...] that i can't see the reputation of other users."
"When it will be possible to see profiles of other users I would definitely prefer to see informations about their field of study, maybe gender, age and added content."

As mentioned previously, the more diverse the evidence, the greater the chance to accurately model users' expertise. The following quotes refer to the approach of considering individuals' contributions made to various communities.

"The profile will never be completely exact, because you can't write down all of the problems you solved in your life."
"To capture a quite complete profile much more contributions are necessary."
"[...] integrate content from own blogs."
"[...] possibly linking it to other information: LinkedIn / XING profiles for example would provide a valuable source."

Apart from the communication about certain topics, participants would appreciate additional ways to get in contact with their peers.

"Perhaps some kind of chat function for currently online users."
"You can never get in contact with the contributors."
"Some kind of networking features could be nice."
"The feature to send message to other users to contact other users. People who have the same interests may like to know each other more."

CHAPTER 7

Conclusion

The main motivation for this thesis was the need to quantify users' expertise in order to (1) compare potential experts with each other in terms of their expertise levels and (2) to determine experts with the needed distribution of expertise, i.e., distinguishing generalists from specialists. We addressed this need by developing an algorithm that identifies and measures users' expertise on an absolute scale. Users' contributions to online communities including their textual submissions and information gained from users' social interactions serve as expertise evidence. This chapter aims to answer the research questions posed in Section 1.1. While answering these questions we briefly recall the main contributions of this thesis and summarize our findings. As the devised method is not only applicable to the specific environment in which we conducted our experiments, we provide a concise list of issues to customize the method for its use in other application environments. We close with presenting open issues and directions for future work.

7.1 Answers to Research Questions

Our work was guided by the following main research question:

> *Can we reliably quantify users' technical expertise based on their contributions in an online community?*

With respect to this main question, we explored ways to quantify users' expertise and present the calculated expertise scores to the users for scrutiny. This is essential for two reasons. Firstly, users need to know what the system stores about them, otherwise the majority of users will not trust the system. In addition, we observed that users also want to know how their user models are generated. The second reason why we opened the models to the users was to gather their feedback about the expertise levels calculated for them. We approached the main research question by means of the following subquestions.

7.1.1 Question 1.

Q.1: Can we consistently quantify users' expertise levels on an absolute scale?

In Chapter 3, we proposed the Expertise Calculator which constitutes a hybrid method to identify expertise topics from users' contributions and calculates an expertise score for each of these topics. In the course of this thesis, we iteratively designed the Expertise Calculator producing three different versions. We evaluated each of these versions by conducting separate experiments, each involving master students sharing their experience in the field of Software Engineering online.

We aimed at validating the Expertise Calculator's score predictions in comparison to participants' self-assessments. We found that contribution types differ in their value for reliable score calculation. Our results show that ratings have the least influence on score accuracy. This is surprising, because we actually assumed that ratings will play a much more important role, which was also suggested by related literature as surveyed in Section 3.2.2. In addition, our discussions with students in terms of the quality of certain contribution types to serve as expertise evidence clearly supported our initial assumption about the potential importance of ratings for score calculation.

However, in this regard, we probably need to rethink the association of raters' expertise with the texts they evaluate. We came across a few indications that let us doubt the usefulness of relating peer votes with the text corpora of the rated contributions. To begin with, ratings are given with a rather low effort compared to the provision of other contribution types, e.g., authoring a challenge. This is not at least suggested by a significant higher amount of ratings than challenges collected in the course of the experiment as shown in Table 6.1. Besides, it seems obvious that participants' personal involvement during voting is not as high as when contributing a challenge participants struggle with. In order to qualify predicted scores we asked participants for their self-assessments. They were supposed to provide self-estimates against the background of their personal contributions. Given the aforementioned easiness of rating a contribution, participants might not identify themselves with topics associated with their votes, at least not as strongly as they identify themselves with more costly contributions. In the end, this might result in higher score deviations (due to biased self-assessments) and lower correlation values respectively.

A further sign that suggests a particular careful handling of the votes-text relation is the fact that false positives, expertise topics wrongly associated with participants, mainly originate from ratings. On the one hand, this may support our previous assumption that participants do not feel as related to texts originating from their ratings as they identify themselves with their own textual contributions. On the other hand, rating another's contribution can occasionally cause a considerable amount of expertise topics the raters are suddenly associated with, just by giving a quick and easy vote. Probably, a more selective approach is needed that adopts new terms from rated contributions as candidates for raters' expertise. One participant said in the closing feedback: "I rated some challenges/solutions which I thought looked very complicated and now I am an expert in Typo3 I hardly know."

Another finding concerns the combination of contribution types. As soon as contribution types are combined with each other, certain combinations yield higher score accuracies than

those score calculations focussing on single contribution types. Contribution types take part in the score calculation process by means of their contribution weight. We found out that a certain assembly of weights amongst the contribution types leads to higher score correlation. Based on this examination, we established a weighting rule as presented in Equation 6.1.

As illustrated in Figure 6.4 expertise predictions are influenced by contribution weights as well as default rating values. For the latter, we explored their effect on score tendencies by altering the default rating value for ratings from 1 to 1.5. Not only did we see that default rating values indeed have an effect on expertise scores, the results from score calculation based on the changed default rating value revealed a better balance regarding score tendencies than the previous setting. Furthermore, we addressed the problem of rating data sparsity and showed that considering default rating values in expertise score calculation leads to a better overall score accuracy than only calculating expertise scores based on peer votes.

Given our results showing an average score deviation of 18 points when comparing calculated scores with users' self-assessments, we can say that the proposed Expertise Calculator is able to quantify users' expertise.

7.1.2 Question 2.

While calculating expertise predictions a certain extent of uncertainty always remains. We addressed this issue by means of the following question:

Q.2: Can we determine a confidence level to express the reliability of expertise predictions?

In Section 3.2.4, we introduced two independent confidence measures. The first one considers expertise levels of peer users whereas the second one regards the variety of contributions users submitted to the online community. We evaluated each of these measures separately and further examined their combination leading to an overall confidence level. We assumed that the lower the score deviation of predicted scores from users' self-assessments, the higher the confidence in these scores. Although our sub-measures pursue different strategies to measure the reliability of score predictions, we did not find any evidence that one would outperform the other. On the contrary, none of the measures did work as we expected.

However, after exploring the scatterplots of various variables we realized that there exists a moderate linear relationship between score deviations and confidence levels. This moderate correlation was measured at a certain range of participants' self-assessment. More specifically, for score predictions associated with high self-assessments we observed a significant higher correlation coefficient than for lower self-assessments. This can be valuable in situations when seeking people with particularly advanced expertise, e.g., when looking for candidates to moderate a forum. In sum, regarding the answer to the research question, we can say that the proposed confidence measure is only partly suitable to calculate the confidence in expertise scores.

7.1.3 Question 3.

The proposed Expertise Calculator represents user models as ontological overlay models. Given the relations between competence concepts in the underlying competence ontology, we asked the following:

> **Q.3:** Can we determine a user's expertise in topic Y based on the user's expertise in topic X by exploiting the direct or indirect linkage given in the competence ontology?

In Chapter 4, we presented a novel approach to spread expertise scores in ontology overlay models. To express the various abstraction levels of competence concepts in the ontology (general vs. specific competences), we adopted a measure from literature exploiting the ontology's hierarchical levels to determine the similarity of competence concepts. Based on these similarity links, we applied an adapted spreading activation algorithm to propagate expertise scores through the ontology network. We evaluated this algorithm by means of expert assessments as well as users' self-assessments. As for the former, we found that compared with a simple baseline, our approach performs significantly better without introducing further efforts regarding configuration, e.g., enhancing the ontology with new competences does not imply any additional human effort but only the automatic recalculation of new similarity links. In another experiment (see Section 5.2.3), we integrated the novel approach with a user interface that supports users in constructing their expertise models. In particular, the spreading activation algorithm was used to provide users with expertise predictions based on their self-assessments. We found that on average, the score deviation of predicted scores from users' self-assessments amounts to approximately 15 points.

Our results suggest that similarity measures can be effectively used for the alignment of scores associated with general and specific expertise topics. In addition, the assumption we made that information about specific expertise receives a higher priority than information about more general expertise topics seems practicable. In the context of users' self-assessments, this might be caused by the fact that specific expertise (rather well-defined) is easier to self-assess than general expertise (rather ill-defined). Given our observations and results we can answer our research question with a "yes".

7.1.4 Summary

The results of our experiments suggest that the proposed Expertise Calculator is able to determine users' expertise levels. On average, the deviation of expertise scores from the users' self-assessments was approximately 18 points. This seems to be a very promising figure. In the first place, a qualitative scale for expertise such as one ranging from "novice" to "expert" may be more intuitive than expertise levels ranging from 0 to 100 points. It seems obvious that information systems can benefit from more detailed user information in that they can adapt their services to users more accurately. Besides that, we were wondering whether such fine-grained expertise scores make sense to humans as well. At least we knew from literature that people want their expertise represented as accurately as possible. Our experiment results showed that participants maintaining their expertise models make use of the full range of expertise scores, for

instance, participants carefully decided whether to describe their expertise topics with 55 points or with 60 points. In fact, this indicates that people do care about fine-grained expertise levels. However, once people's expertise is automatically determined by a system they show a strong demand to know how expertise is calculated, the more detailed the better. Finally, we observed that participants perceive the task of self-assessment less tedious when supported by a system providing them with expertise predictions.

The data collected during the various experiments can be requested from the author for further consideration or even to replicate the results presented in this thesis. The data set is forwarded to other researchers in the form of an SQL database dump and does not include any personal information about the users who participated in our experiments. The SQL dump consists of database tables used in a standard installation of Drupal 6.

7.2 Application

The main contribution of this thesis is a method and its prototypical implementation for calculating user expertise in an online community. We iteratively constructed the method within a particular environment, i.e., a knowledge-sharing platform where students exchange their experience with issues related to Software Engineering. However, the proposed method is also applicable to other domains as long as the target environment includes the storage of users' contributions, relationships between users and their contributions as well as rating capability. In case the Expertise Calculator is applied to other environments, the following customization steps are required:

- Competence ontology: The expertise topics of the target domain need to be modeled by means of a competence ontology. The topics have to be arranged in a hierarchy and stored either in RDF or OWL file format. Each topic may be associated with one or more synonyms. The similarity between expertise topics will be calculated automatically once the ontology is uploaded to the system.

- Contribution types: Users can exchange information by various types of contributions. These contributions need to be weighted according to their perceived value for reliable expertise calculation. In our research, we derived a weighting rule for commonly used contribution types in online communities. This weighting rule can serve as a base for the determination of weights for similar and new contribution types respectively.

- Configuration parameters: The proposed method can be adjusted by means of a few parameters in order to optimize its performance in the target domain. For instance, the default rating values substituting missing peer ratings can be utilized to prevent a trend to overestimate users' expertise. In the course of our experiments, we tested different settings of these parameters for both the prediction of expertise scores and the calculation of confidence levels. The effect of these settings on the algorithm's performance can guide the process of finding suitable parameters in other environments.

Application environments not supporting rating capabilities may use a lightweight version of the Expertise Calculator. That means, expertise score calculation is then solely based on users' textual contributions which causes contribution weights to play a more important role. We have not experimented with such a specific setting so far. But we learned from working with default rating values that there is potential to measure expertise without any peer ratings; however, in this case expertise scores showed lower accuracy figures. As for calculating the trust in predicted scores, the lack of rating data reduces the overall measure to only consider the variety of users' submissions to the online community.

7.3 Future Work

In the course of our research, we introduced an approach to quantify users' expertise in online communities. However, there are still a few open issues remaining. Moreover, we identified starting points for improvements of measuring expertise. Thus, future work can be conducted on the following aspects.

Field-based document weighting models exploit the structure of documents and associate their fields with individual weights. Applying such a model to our context means that we would consider textual submissions to the community in more detail, e.g., viewing a posted challenge as having a title, a goal and a main body. [Macdonald and Ounis, 2006] propose an approach for expert finding based on documents. They found that weighting the body and the title separately can improve the performance of expertise retrieval significantly. Thus, exploiting the structures of contribution types more thoroughly seems promising to further refine the calculation of expertise scores. Another issue for improvement concerns the calculation of confidence levels. The existing overall confidence measure may be enhanced with a metric that keeps track of the half life period of users' expertise given an environment where long term data is available.

Although current human-edited competence ontologies are mostly structured hierarchically, it should not go unmentioned that ideas for future work also include the spreading of expertise scores in non-hierarchical ontologies as well as ontologies built on a mixture of hierarchical and non-hierarchical structures. Considering additional transversal relations may increase the ontology's expressiveness. Intuitively, this might implicate more accurate expertise scores being propagated. Currently, the proposed score propagation method only considers hierarchical structured competence ontologies. Thus, an improved version may also consider multi-inheritance of topics as well as integrate additional relation types such as *part-of* relationships. As for the latter, the calculation of relation weights could be based on both relative depth scaling and the link type. The aggregation of link semantics, e.g., with *similarity* and *part-of*, may improve the accuracy of expertise scores. In such a case, we could assign constant weights to labeled link types as *part-of* $= 0.5$ and *is-a* $= 1$. Then, the total link weight could be calculated by $\omega_{total} = \omega_{linktype} \cdot \omega_{Sussna}$. Another option to improve the value of relations might be that of adopting the notion of Bayesian networks, which consider probabilities between competences in the ontology. As we already mentioned, the larger an ontology gets, the harder the process is to keep the link probabilities up to date by human experts. To exploit empirical data for this purpose, we may evaluate users' feedback to calculated expertise scores and learn link probabilities given these data. [Mockus and Herbsleb, 2002] found that people sometimes want to

locate an expert in a particular technology, for instance, an expert in databases. In addition, they observed a frequent need for an expert in a specific part of a product, e.g., someone who knows the OA&M interface for a certain network component. Based on that, how can we integrate expertise about particular products with expertise not directly related to products? What might be the implications for expertise score propagation?

We presented a user interface featuring expertise predictions to facilitate users' self - assessment. The calculation of predicted topics related to those a user is already familiar with is not without shortcomings. To be specific, one aspect is that users are not encouraged to reflect on topics for which they currently have no knowledge at all. It would be interesting to explore the enhancement of predictions with scores gained from collaborative filtering based on users' similar expertise. This will introduce new topic areas to users since they are not based on topics users explicitly stated. This might help them to explore more new areas over familiar ones.

In this thesis, we evaluated the calculation of users' expertise levels in a controlled environment. We conducted our main experimental work with master students attending a tutorial on knowledge management. The tutorial was regularly held in the winter term with changing participants. Consequently, the experiment durations were rather short and may have limited the interpretation of our results. In this regard, it would be interesting to see how the proposed Expertise Calculator performs on data gathered over a longer period of time. At the same time, the robustness (How vulnerable is the model in terms of deliberate user attacks?) of expertise predictions could be further explored as well. Since the design of the Expertise Calculator allows for application in various domains (given the presence of a respective domain ontology) with different kinds of textual user submissions, future research may consider data obtained from one of the well established Question and Answer communities such as *Yahoo! Answers*.

We leveraged information about users' social interactions such as peer ratings to calculated users' expertise scores. However, we have completely ignored exploiting social relations amongst users. Social relations may contain useful information to predict more accurate and reliable expertise of users. For instance, we can ask who shows interest in whose contributions or does someone answer challenges from specific peers more frequently? By exploring these questions, we might discover a latent social network where users are connected amongst each other. Based on such a network, we could examine raters' closeness to users being indirectly rated via their contributions. On the one hand, close relationships between raters and users may help in precisely assessing a contribution's complexity since the rater knows more about the user and might interpret the user's contribution more precisely. On the other hand, however, close relations may lead to unjustified ratings in order to win favor or because of dislike towards the user being rated.

Currently, the Expertise Calculator only considers users' technical expertise. Besides that, data in online communities show potential to measure further user attributes such as personal characteristics or social expertise. The aggregation of various user attributes can not only give a broader picture of users but might equally yield more accurate user information. Furthermore, [Lindgren et al., 2003] suggest to consider the interests of users in organizational competence management. For instance, if users frequently show interest in certain contributions it is highly possible that they have a certain degree of expertise with respect to the contributions' topics.

List of Figures

List of Tables

APPENDIX A

Additional Figures, Forms and Tables

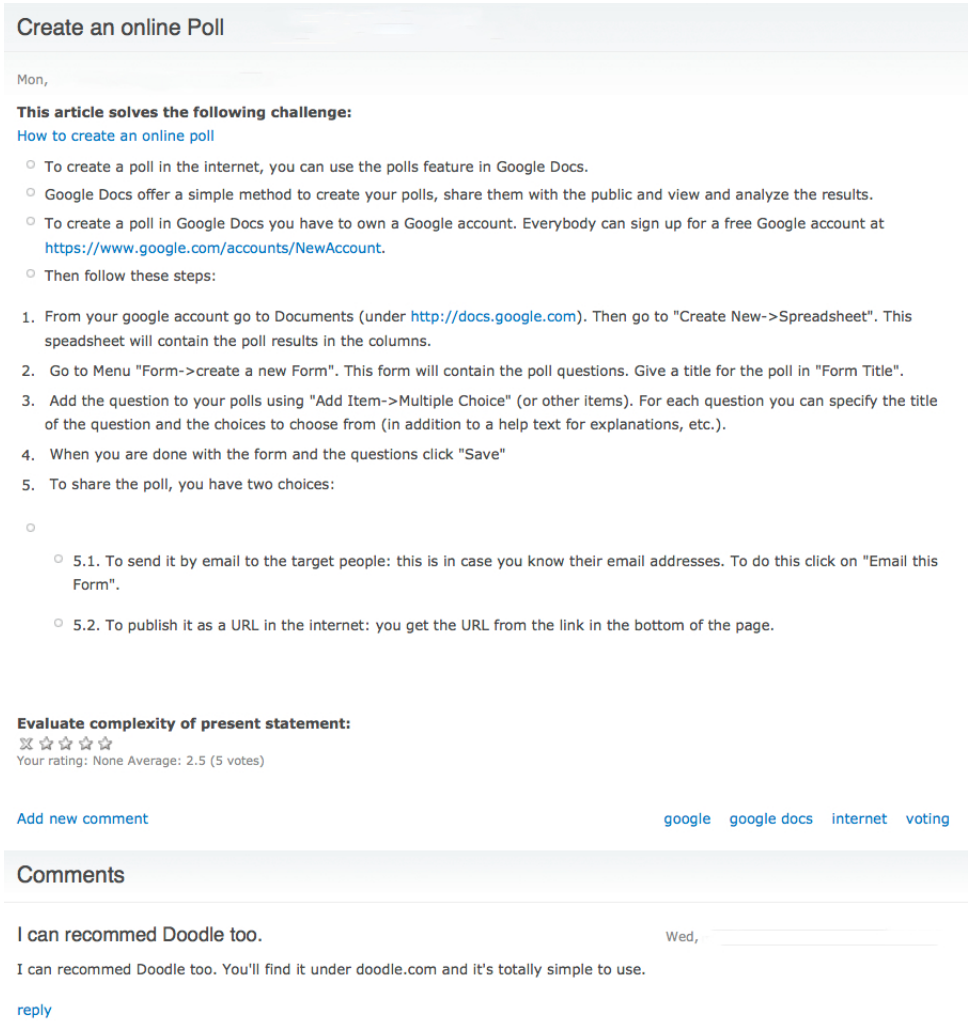


Figure A.1: Display of an example solution 1

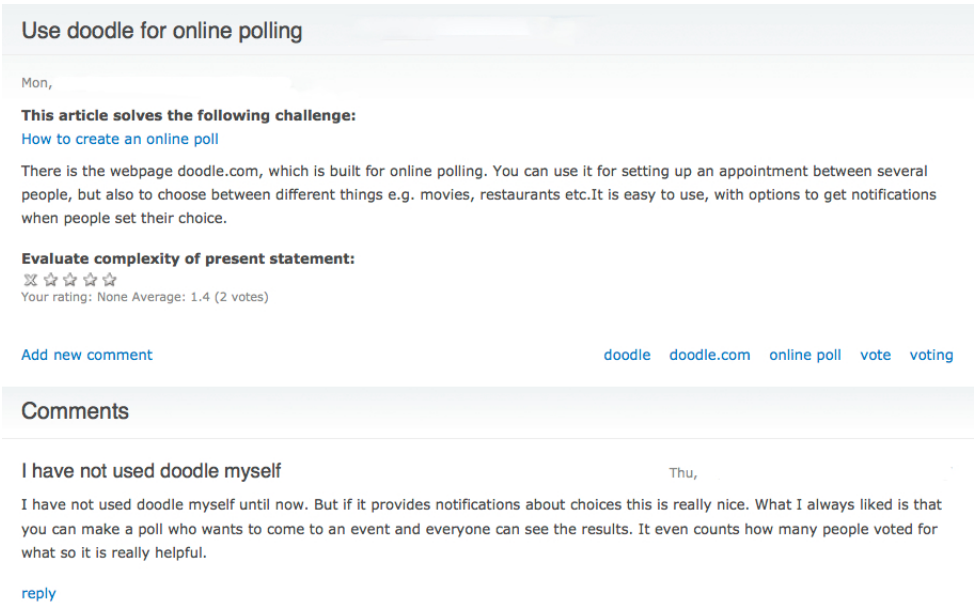


Figure A.2: Display of an example solution 2

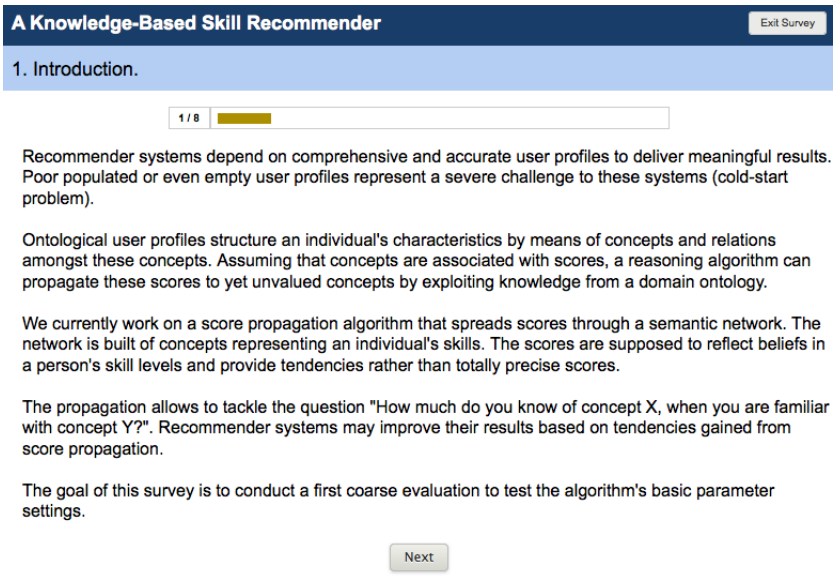


Figure A.3: Score propagation evaluation survey, Part 1.

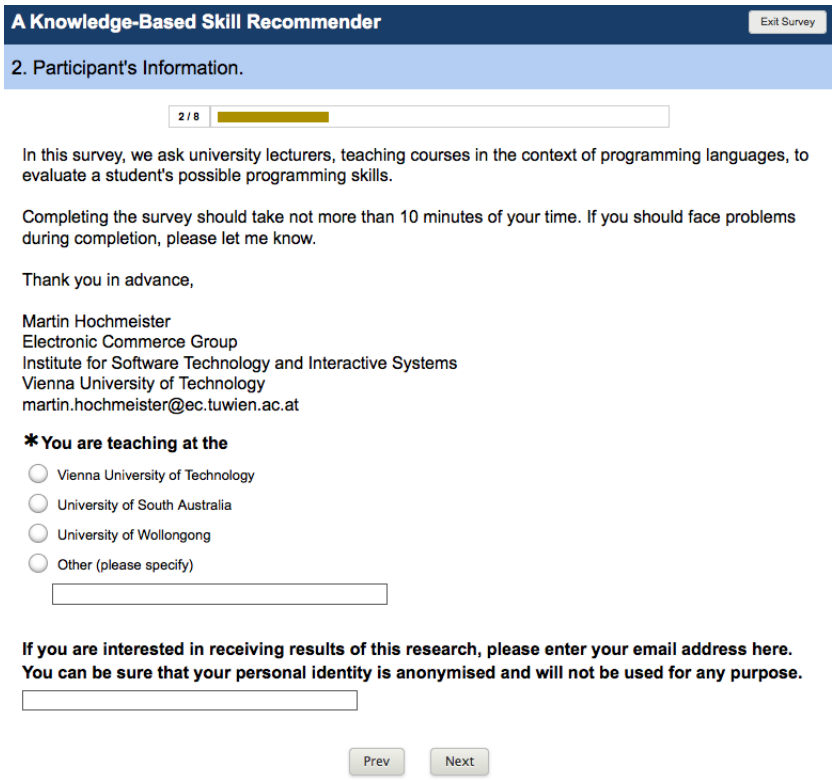


Figure A.4: Score propagation evaluation survey, Part 2.

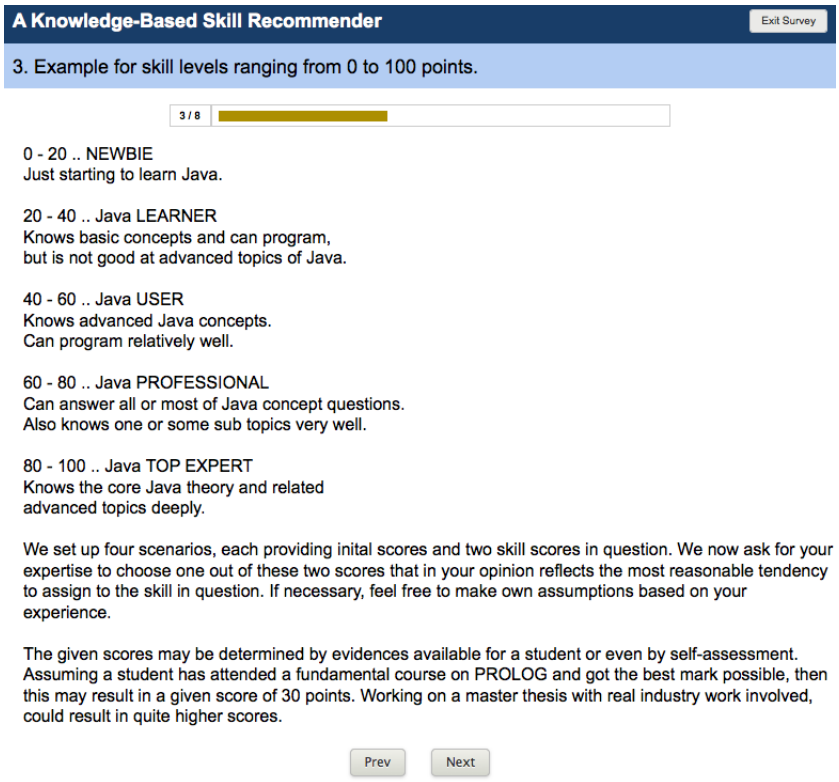


Figure A.5: Score propagation evaluation survey, Part 3.

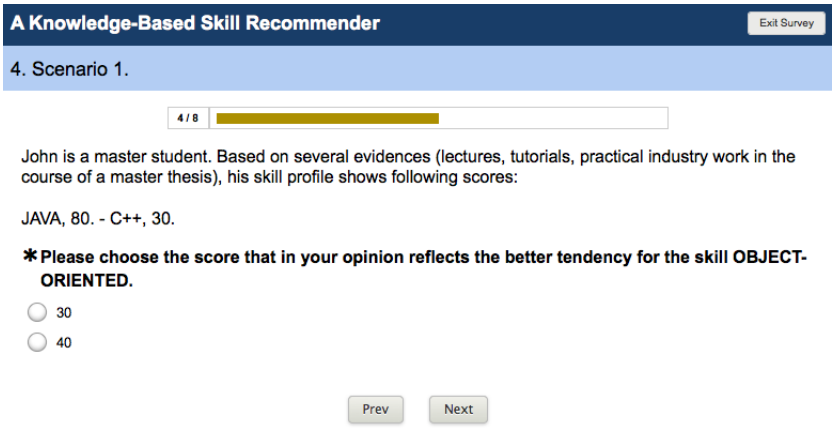


Figure A.6: Score propagation evaluation survey, Part 4.

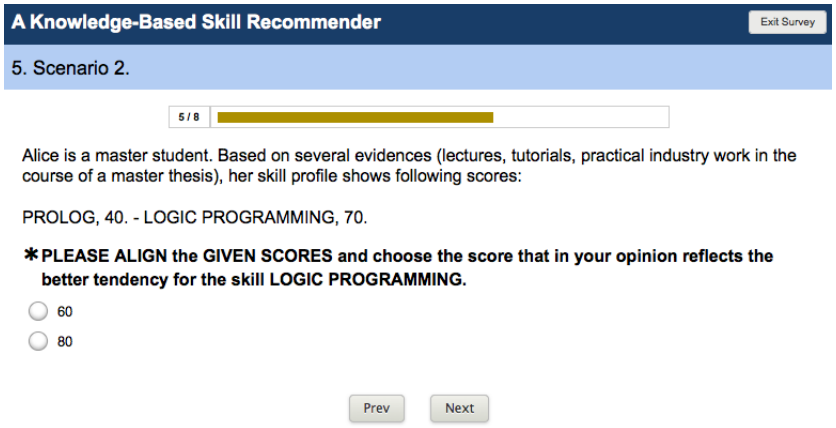


Figure A.7: Score propagation evaluation survey, Part 5.

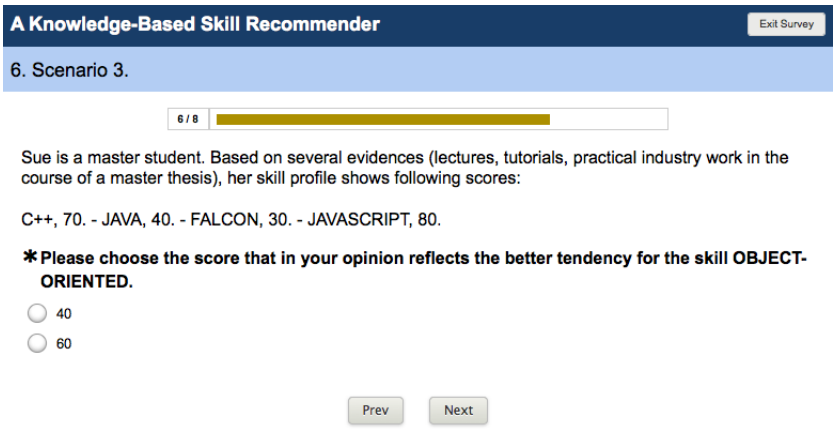


Figure A.8: Score propagation evaluation survey, Part 6.

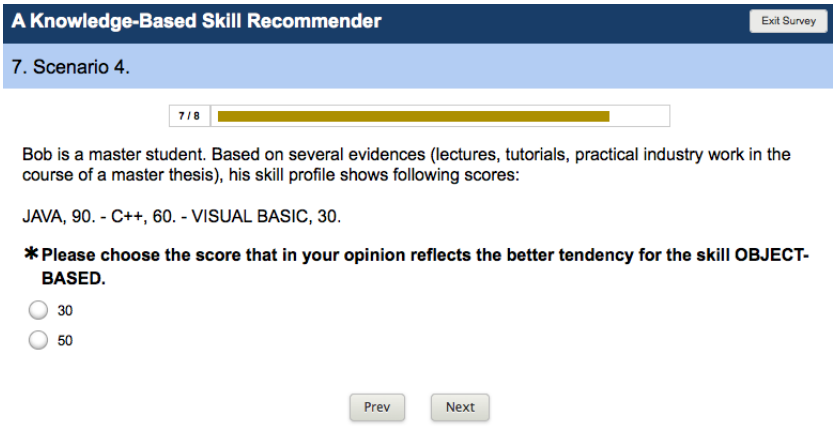


Figure A.9: Score propagation evaluation survey, Part 7.

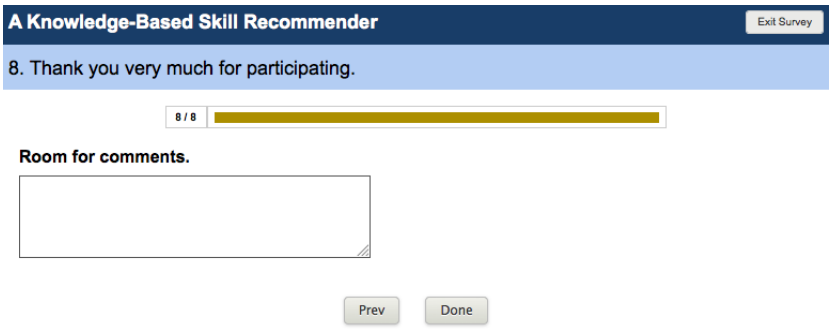


Figure A.10: Score propagation evaluation survey, Part 8.

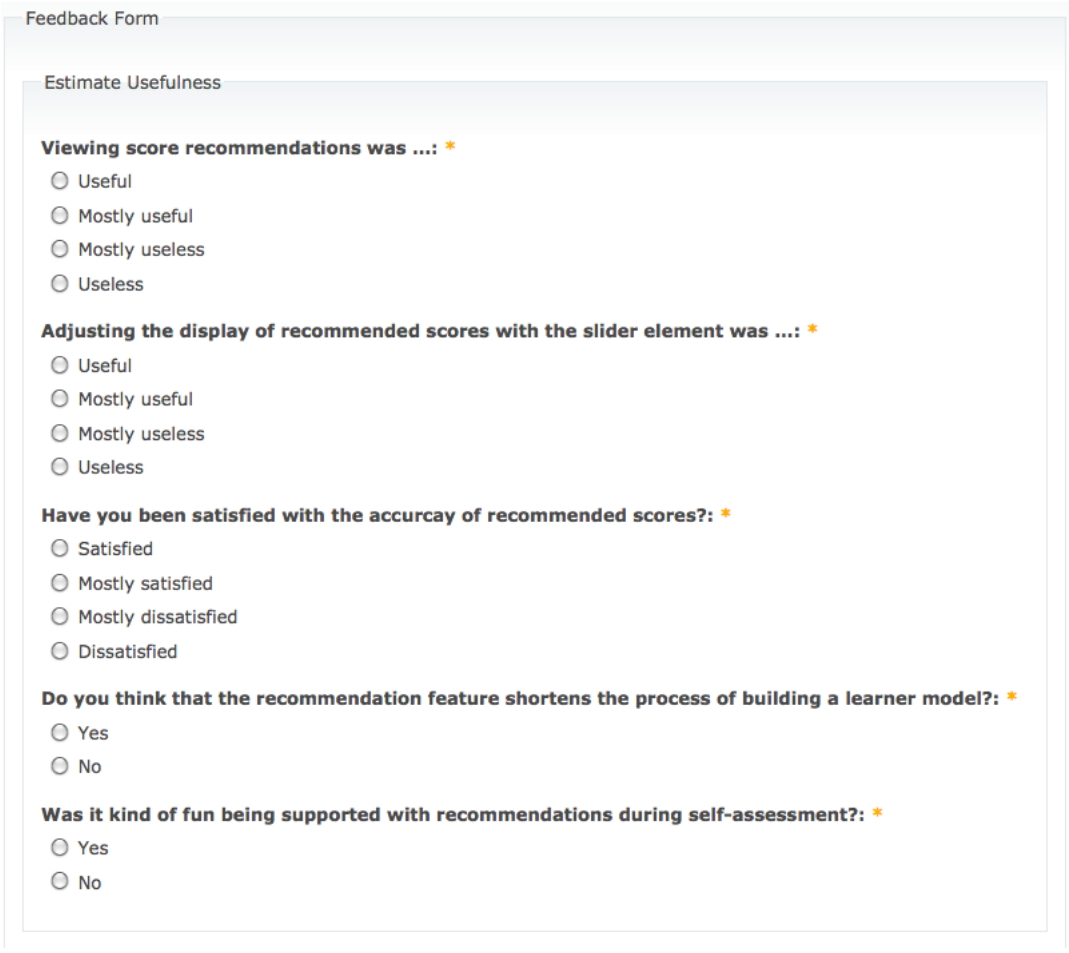


Figure A.11: Feedback form, Part 1.

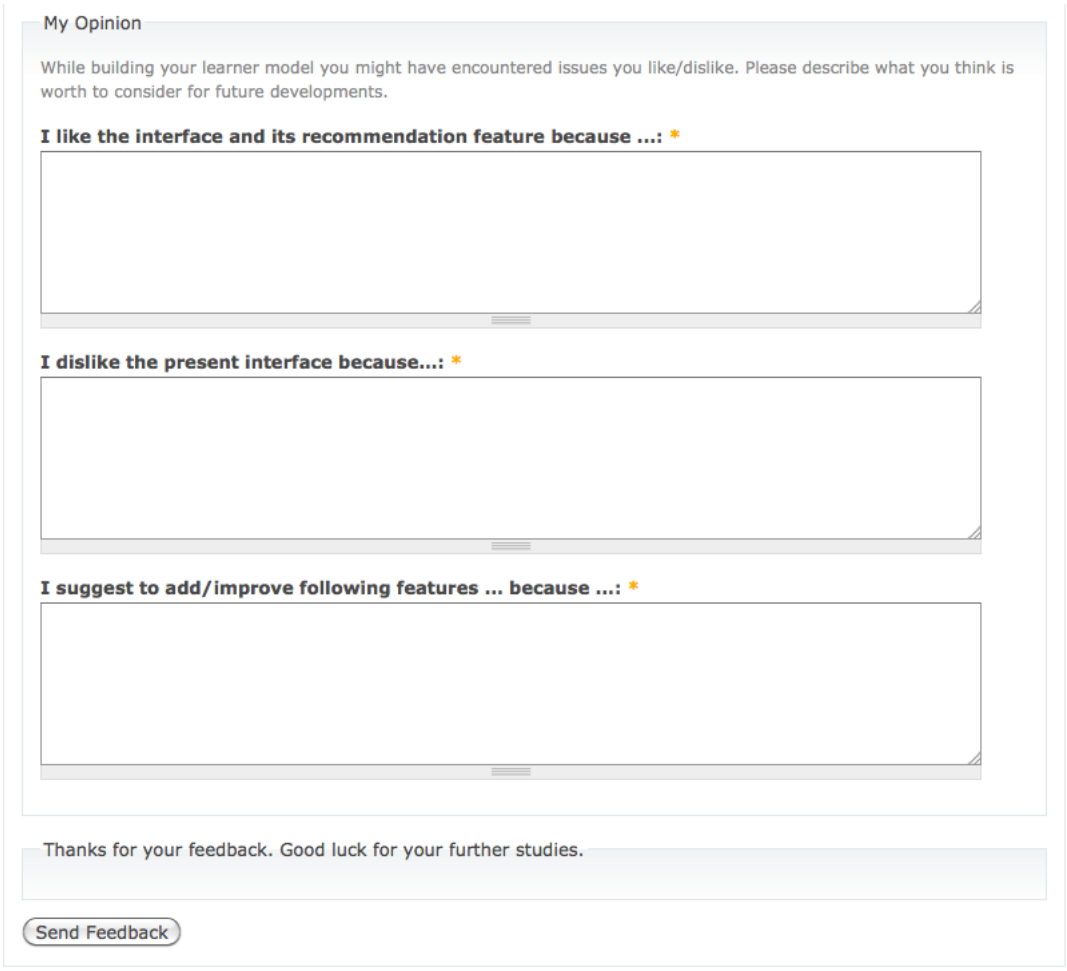


Figure A.12: Feedback form, Part 2.

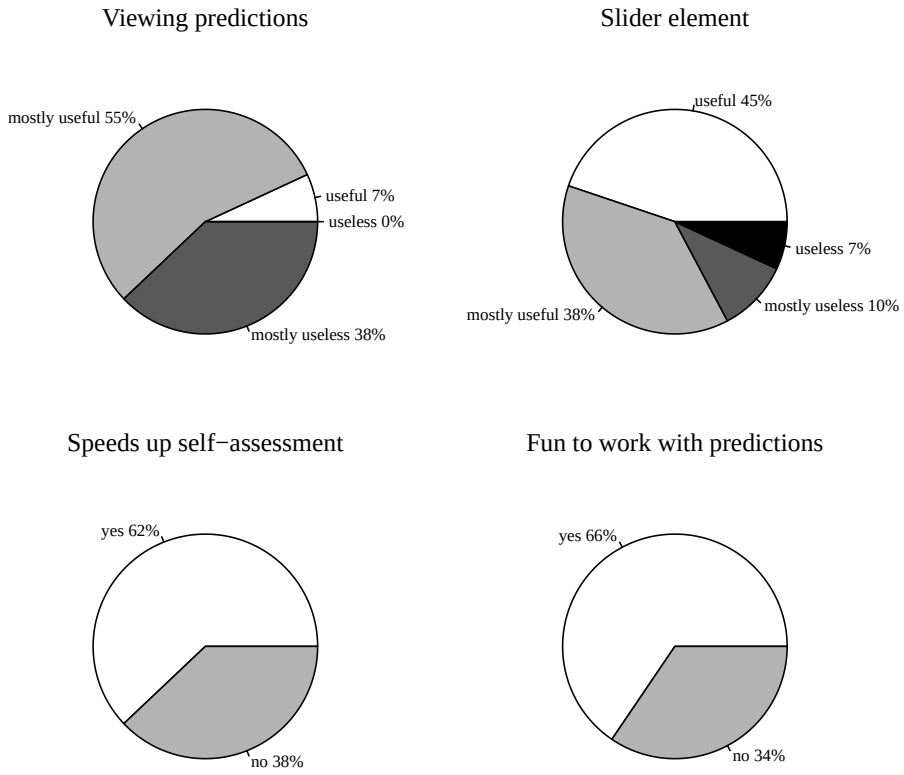


Figure A.13: Results from quantitative student feedback.

Table A.1: Score accuracy calculated on different weight combinations, $n = 716$

ω_{Ch}	ω_S	ω_{Co}	ω_T	ω_R	Mean	Median	SD	**Correlation** ↓	R.M.S.	# Correct	# Under	# Over
5	5	4	3	2	18.45670	13.0	15.34652	0.4251923	17.92364	31	394	291
5	5	3	2	1	20.97905	17.0	16.87828	0.4246628	17.92856	36	442	238
4	5	3	2	1	20.93855	17.0	16.86322	0.4244265	17.93076	38	446	232
4	5	3	1	1	20.98743	17.5	16.89593	0.4236703	17.93777	38	446	232
4	4	3	3	1	21.32402	19.0	16.22675	0.4232447	17.94171	42	521	153
4	5	2	1	3	17.50279	12.0	15.21735	0.4223023	17.95041	60	375	281
5	5	4	4	1	20.31844	16.0	16.48205	0.4215501	17.95735	35	416	265
5	5	4	2	1	20.41061	16.0	16.49409	0.4205940	17.96614	34	418	264
5	5	4	1	1	20.45950	16.0	16.52867	0.4197691	17.97370	34	418	264
4	5	3	1	3	17.51257	12.0	15.14872	0.4173946	17.99537	59	370	287
5	5	5	4	2	18.54749	14.0	15.06394	0.4168638	18.00020	34	365	317
2	5	3	1	3	17.57123	12.0	15.19769	0.4152999	18.01437	57	372	287
5	5	5	4	3	17.49162	14.0	14.33627	0.4126899	18.03787	52	316	348
5	5	5	5	1	20.22207	17.0	16.30152	0.4119064	18.04489	34	389	293
5	5	5	4	1	20.23883	17.0	16.31431	0.4116429	18.04725	35	390	291
5	5	5	1	1	20.37989	17.0	16.36159	0.4097200	18.06440	34	392	290
3	5	5	1	3	17.49581	14.0	14.35450	0.4083516	18.07655	52	324	340
3	5	2	1	4	17.20950	13.0	13.85028	0.4077263	18.08209	42	318	356
5	5	3	2	4	17.25559	14.0	13.78670	0.4075328	18.08380	41	309	366
3	3	3	3	1	24.09078	21.0	15.84510	0.4071761	18.08695	32	594	90
1	5	1	1	1	24.34218	20.0	18.62073	0.4000394	18.14929	32	497	187
3	3	3	5	1	23.71369	20.5	15.89917	0.3958206	18.18553	32	584	100
5	4	5	1	1	20.28492	17.5	15.80714	0.3933894	18.20620	39	445	232
5	1	3	5	1	24.16760	20.0	16.96297	0.3933227	18.20677	36	558	122
4	5	5	1	4	17.45531	14.0	13.71598	0.3912646	18.22415	42	289	385
2	4	2	5	4	17.48184	14.0	13.26844	0.3677580	18.41512	49	395	272
1	1	1	1	1	37.95950	40.0	18.38005	0.3674087	18.41785	17	694	5
4	4	4	4	4	17.46788	15.0	13.23527	0.3595686	18.47843	47	385	284
3	3	3	3	3	20.51955	20.0	14.61672	0.3593313	18.48024	53	521	142
5	5	5	5	5	18.31564	16.0	13.68193	0.3592473	18.48088	28	266	422
5	5	5	3	5	18.30587	16.0	13.69886	0.3589726	18.48297	28	266	422
3	3	3	1	3	20.55168	20.0	14.62162	0.3589180	18.48339	52	522	142
5	1	1	1	1	29.81844	28.0	18.95701	0.3587696	18.48452	26	613	77
2	2	2	2	2	27.80447	28.0	16.67596	0.3550404	18.51273	20	644	52
1	3	5	5	1	22.01676	20.0	15.59397	0.3500644	18.54986	39	514	163
1	3	5	5	3	19.14106	18.0	13.77901	0.3147430	18.79643	57	441	218
1	1	5	1	1	25.90084	22.0	17.63563	0.3001260	18.88995	37	552	127
1	1	1	5	1	34.42598	33.5	18.61953	0.2891170	18.95717	22	664	30
3	1	5	1	3	20.51676	20.0	15.09971	0.2811235	19.00425	58	457	201
3	1	3	5	3	22.48464	20.0	16.11258	0.2807540	19.00640	52	530	134
1	2	3	5	4	21.19274	20.0	14.24154	0.2287497	19.27780	41	466	209
3	2	3	4	4	21.08659	20.0	14.14859	0.2272644	19.28469	39	469	208
2	3	2	4	5	19.69693	20.0	13.13167	0.2199314	19.31801	41	406	269
3	3	5	3	5	19.56564	20.0	13.07108	0.2027781	19.39146	37	386	293
3	1	3	3	5	20.77514	20.0	14.05488	0.1860586	19.45709	37	414	265
2	2	4	3	5	20.68017	20.0	13.74125	0.1780369	19.48650	38	406	272
3	1	4	4	5	20.63268	20.0	14.04411	0.1772747	19.48922	35	404	277
2	1	3	3	5	21.39525	20.0	14.44348	0.1713360	19.51004	37	421	258
1	1	1	1	5	22.93855	20.0	15.34158	0.1694566	19.51648	33	439	244
1	1	3	5	5	21.77654	20.0	14.85436	0.1594161	19.54962	38	415	263

Table A.2: Score accuracy of new topics originating from score propagation, $n = 36$

ω_{Ch}	ω_S	ω_{Co}	ω_T	ω_R	Mean	Median	SD	**Correlation ↓**	R.M.S.	# Correct	# Under	# Over
1	1	1	1	1	40.33333	39.0	14.82854	0.3499656231	14.92483	0	36	0
1	1	1	5	1	36.69444	35.0	15.50451	0.2931232058	15.23252	0	36	0
1	5	1	1	1	24.30556	23.0	16.31444	0.2802884292	15.29372	1	30	5
4	5	3	1	1	21.38889	20.0	16.94070	0.2074627860	15.58571	3	23	10
5	5	3	2	1	21.66667	20.0	16.91998	0.2061394701	15.59017	3	23	10
4	5	3	2	1	21.33333	20.0	16.95540	0.2057322064	15.59153	3	23	10
4	4	3	3	1	22.33333	19.0	17.76835	0.1893540958	15.64412	1	28	7
4	5	2	1	3	19.33333	16.5	16.28321	0.1892717899	15.64437	2	24	10
4	5	3	1	3	19.22222	16.0	16.33771	0.1815935784	15.66746	3	22	11
2	2	2	2	2	32.08333	31.0	16.03456	0.1783721258	15.67685	0	36	0
2	5	3	1	3	19.02778	15.0	16.45511	0.1729041701	15.69239	3	22	11
5	5	4	1	1	22.58333	20.0	15.16269	0.1650892704	15.71374	0	22	14
5	5	4	2	1	22.52778	20.0	15.19114	0.1633977876	15.71823	0	22	14
5	5	4	4	1	22.38889	20.0	15.29384	0.1589680160	15.72975	0	22	14
5	1	1	1	1	31.50000	28.0	17.71279	0.1587837627	15.73023	1	32	3
3	3	3	3	1	25.72222	21.5	18.16372	0.1475165136	15.75805	2	33	1
5	5	4	3	2	21.55556	19.5	15.10619	0.1468883350	15.75954	0	22	14
3	3	3	5	1	25.13889	21.0	18.47957	0.1410882869	15.77298	2	33	1
5	5	5	1	1	23.58333	22.5	14.04152	0.1231892867	15.81100	0	22	14
5	5	3	2	4	19.52778	17.5	15.22308	0.1221645033	15.81302	1	20	15
5	5	5	4	1	23.38889	22.5	14.19714	0.1172481439	15.82246	0	22	14
5	5	5	5	1	23.33333	22.5	14.25883	0.1154792758	15.82576	0	22	14
3	5	2	1	4	19.05556	17.0	15.25768	0.1142837297	15.82797	0	21	15
3	3	3	1	3	23.38889	19.0	17.91771	0.1024262109	15.84856	2	33	1
5	5	5	4	2	22.50000	22.5	14.29985	0.1024219361	15.84857	0	22	14
3	3	3	3	3	23.25000	19.0	18.01170	0.0932066095	15.86300	2	33	1
5	5	5	4	3	21.27778	20.0	14.67283	0.0869064976	15.87207	2	20	14
5	4	5	1	1	24.02778	23.5	15.17985	0.0847669819	15.87501	0	23	13
3	5	5	1	3	20.88889	19.5	14.63416	0.0750265312	15.88745	1	21	14
4	4	4	4	4	18.94444	16.0	16.82506	0.0659590580	15.89766	1	21	14
5	1	3	5	1	26.16667	22.0	19.45618	0.0653913042	15.89825	1	29	6
4	5	5	1	4	20.41667	19.0	14.58840	0.0646433993	15.89903	1	19	16
2	4	2	5	4	19.30556	15.0	16.51635	0.0543506597	15.90880	1	23	12
5	5	5	3	5	20.91667	19.0	14.00281	0.0490298346	15.91319	0	18	18
5	5	5	5	5	21.02778	19.0	13.90475	0.0448756685	15.91630	0	18	18
3	1	3	5	3	25.05556	22.0	19.37737	0.0171820695	15.93000	1	34	1
1	3	5	5	1	24.58333	22.5	17.26495	-0.0001490487	15.93235	1	28	7
1	1	5	1	1	29.11111	28.0	17.93604	-0.0264560029	15.92678	1	30	5
1	3	5	5	3	22.22222	19.5	17.79745	-0.0460370637	15.91546	1	26	9
1	2	3	5	4	23.11111	20.0	18.82821	-0.0539289090	15.90917	1	30	5
3	1	5	1	3	23.77778	23.0	19.06846	-0.0655342294	15.89810	2	28	6
3	2	3	4	4	22.77778	18.0	19.19689	-0.0691349701	15.89423	3	30	3
3	3	5	3	5	21.16667	17.5	17.58165	-0.0828150581	15.87762	0	23	13
1	1	3	5	5	23.72222	23.0	18.14168	-0.0843056556	15.87563	0	27	9
1	1	1	1	5	24.63889	23.0	18.70852	-0.0897253334	15.86809	1	29	6
3	1	4	4	5	21.80556	17.5	18.47287	-0.0905482462	15.86690	1	25	10
2	3	2	4	5	21.47222	16.5	17.42820	-0.0912001398	15.86596	0	24	12
3	1	3	3	5	21.97222	16.5	18.64478	-0.0924507157	15.86412	0	26	10
2	1	3	3	5	22.91667	19.5	18.50154	-0.0954635060	15.85959	1	27	8
2	2	4	3	5	22.27778	19.5	18.36085	-0.0978610519	15.85588	2	25	9

Table A.3: Correlation Score Deviation/Confidence, @self-assessments > 25, n=636

λ	Correlation	Mean Conf Third 1	Mean Conf Third 2	Mean Conf Third 3
0.0	-0.12	43.19	40.30	33.15
0.1	-0.11	40.67	37.95	31.21
0.2	-0.11	38.24	35.71	29.35
0.3	-0.11	35.72	33.40	27.40
0.4	-0.10	33.06	31.00	25.44
0.5	-0.09	30.61	28.75	23.56
0.6	-0.08	28.14	26.47	21.68
0.7	-0.08	25.65	24.17	19.78
0.8	-0.07	23.20	21.93	17.90
0.9	-0.06	20.71	19.65	16.00
1.0	-0.05	18.38	17.49	14.18

Table A.4: Correlation Score Deviation/Confidence, @self-assessments > 50, n=309

λ	Correlation	Mean Conf Third 1	Mean Conf Third 2	Mean Conf Third 3
0.0	-0.41	56.36	45.55	21.12
0.1	-0.41	53.46	42.94	19.31
0.2	-0.42	50.70	40.51	17.51
0.3	-0.42	47.81	37.95	15.69
0.4	-0.42	44.83	35.23	13.73
0.5	-0.42	42.04	32.78	11.90
0.6	-0.41	39.22	30.28	10.10
0.7	-0.40	36.36	27.74	8.29
0.8	-0.39	33.60	25.30	6.49
0.9	-0.37	30.76	22.81	4.67
1.0	-0.35	28.09	20.41	2.88

Table A.5: Correlation Score Deviation/Confidence, @self-assessments > 75, n=26

λ	Correlation	Mean Conf Third 1	Mean Conf Third 2	Mean Conf Third 3
0.0	-0.39	26.44	26.57	18.4
0.1	-0.38	23.67	24.00	16.6
0.2	-0.38	21.11	21.43	14.8
0.3	-0.38	18.56	18.86	13.0
0.4	-0.38	15.33	15.43	10.8
0.5	-0.38	12.78	12.86	9.0
0.6	-0.38	10.22	10.29	7.2
0.7	-0.38	7.67	7.71	5.4
0.8	-0.38	5.11	5.14	3.6
0.9	-0.38	2.56	2.57	1.8
1.0	NA	0.00	0.00	0.0

Bibliography

[ACM, 2008] ACM (2008). *Computer Science Curriculum 2008: Computer Science Computer Science Curriculum 2008: An Interim Revision of CS 2001*. ACM and IEEE Computer Society.

[Agichtein et al., 2008] Agichtein, E., Castillo, C., Donato, D., Gionis, A., and Mishne, G. (2008). Finding high-quality content in social media. In *Proceedings of the international conference on Web search and web data mining*, pages 183–194. ACM.

[Alavi and Leidner, 2001] Alavi, M. and Leidner, D. (2001). Review: Knowledge management and knowledge management systems: Conceptual foundations and research issues. *MIS Quarterly*, 25(1):107–136.

[Almeida et al., 2010] Almeida, J., Gonçalves, M., Figueiredo, F., Pinto, H., and Belem, F. (2010). On the quality of information for web 2.0 services. *Internet Computing, IEEE*, 14(6):47–55.

[Anderson, 1983] Anderson, J. (1983). A spreading activation theory of memory. *Journal of verbal learning and verbal behavior*, 22(3):261–295.

[Apted et al., 2003] Apted, T., Kay, J., Lum, A., and Uther, J. (2003). Visualisation of ontological inferences for user control of personal web agents. In *IV '03*, pages 306–311. IEEE.

[Ardichvili et al., 2003] Ardichvili, A., Page, V., and Wentling, T. (2003). Motivation and barriers to participation in virtual knowledge-sharing communities of practice. *Journal of knowledge management*, 7(1):64–77.

[Bakalov et al., 2010] Bakalov, F., König-Ries, B., Nauerz, A., and Welsch, M. (2010). Introspectiveviews: An interface for scrutinizing semantic user models. *User Modeling, Adaptation, and Personalization*, pages 219–230.

[Balog et al., 2007] Balog, K., Bogers, T., Azzopardi, L., De Rijke, M., and Van Den Bosch, A. (2007). Broad expertise retrieval in sparse data environments. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 551–558. ACM.

[Balog and De Rijke, 2007] Balog, K. and De Rijke, M. (2007). Determining expert profiles (with an application to expert finding). In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2657–2662.

[Becerra-Fernandez, 2006] Becerra-Fernandez, I. (2006). Searching for experts on the web: A review of contemporary expertise locator systems. *ACM Transactions on Internet Technology (TOIT)*, 6(4):333–355.

[Berio et al., 2005] Berio, G., Harzallah, M., et al. (2005). Knowledge management for competence management. *Journal of Universal Knowledge Management*, pages 21–28.

[Biesalski and Abecker, 2005] Biesalski, E. and Abecker, A. (2005). Human resource management with ontologies. *Professional Knowledge Management*, pages 499–507.

[Billig et al., 2010] Billig, A., Blomqvist, E., and Lin, F. (2010). Semantic matching based on enterprise ontologies. *On the Move to Meaningful Internet Systems 2007: CoopIS, DOA, ODBASE, GADA, and IS*, pages 1161–1168.

[Blanche and Merino, 1989] Blanche, P. and Merino, B. (1989). Self-assessment of foreign-language skills: Implications for teachers and researchers. *Language Learning*, 39(3):313–338.

[Blooma et al., 2010] Blooma, M., Chua, A., and Goh, D. (2010). Selection of the best answer in cqa services. In *2010 Seventh International Conference on Information Technology*, pages 534–539. IEEE.

[Blumenstock, 2008] Blumenstock, J. (2008). Size matters: word count as a measure of quality on wikipedia. In *Proceedings of the 17th international conference on World Wide Web*, pages 1095–1096. ACM.

[Boud, 1985] Boud, D. (1985). *Reflection: Turning experience into learning*. Routledge.

[Boud and Falchikov, 1989] Boud, D. and Falchikov, N. (1989). Quantitative studies of student self-assessment in higher education: a critical analysis of findings. *Higher education*, 18(5):529–549.

[Brachman, 1983] Brachman, R. (1983). What is-a is and isn't: An analysis of taxonomic links in semantic networks. *Computer*, 10.

[Brusilovsky and Millán, 2007] Brusilovsky, P. and Millán, E. (2007). User models for adaptive hypermedia and adaptive educational systems. *The Adaptive Web*, pages 3–53.

[Bull, 2004] Bull, S. (2004). Supporting learning with open learner models. In *Proceedings of the 4th Hellenic Conference in Information and Communication Technologies in Education*, pages 47–61, Athens, Greece.

[Bull and Gardner, 2009] Bull, S. and Gardner, P. (2009). Highlighting learning across a degree with an independent open learner model. In *Artificial Intelligence in Education*, pages 275–282.

[Bull and Kay, 2007] Bull, S. and Kay, J. (2007). Student Models that Invite the Learner In: The SMILI:() Open Learner Modelling Framework. *IJAIED*, 17(2):89–120.

[Bull and Kay, 2010] Bull, S. and Kay, J. (2010). Open learner models. *Advances in Intelligent Tutoring Systems*, pages 301–322.

[Bull and Kay, 2012] Bull, S. and Kay, J. (2012). *Open Learner Models*, page to appear. Springer.

[Bull and Pain, 1995] Bull, S. and Pain, H. (1995). "Did i say what i think i said, and do you agree with me?": Inspecting and questioning the student model. In *AIED*, pages 501–508.

[Burke, 1989] Burke, J. (1989). *Competency based education and training*. Falmer Press.

[Campbell et al., 2003] Campbell, C., Maglio, P., Cozzi, A., and Dom, B. (2003). Expertise identification using email communications. In *Proceedings of the twelfth international conference on Information and knowledge management*, pages 528–531. ACM.

[Cantador et al., 2008] Cantador, I., Szomszor, M., Alani, H., Fernández, M., and Castells, P. (2008). Enriching ontological user profiles with tagging history for multi-domain recommendations. In *1st Int. Workshop on Collective Intelligence and the Semantic Web (CISWeb 2008)*.

[Carr and Goldstein, 1977] Carr, B. and Goldstein, I. (1977). Overlays: A theory of modelling for computer aided instruction. *Artificial Intelligence Memo 406, Massachusetts Institute of Technology, Cambridge, Massachusetts*.

[Cheetham and Chivers, 2005] Cheetham, G. and Chivers, G. (2005). *Professions, competence and informal learning*. Edward Elgar Publishing.

[Chi and Glaser, 1985] Chi, M. and Glaser, R. (1985). *Problem-solving ability*. Learning Research and Development Center, University of Pittsburgh.

[Chi et al., 1982] Chi, M. T. H., Glaser, R., and Rees, E. (1982). *Expertise in problem solving*, volume 1, pages 7–75. Erlbaum, Hillsdale, NJ.

[Chin, 1989] Chin, D. (1989). Knome: Modeling what the user knows in uc. *User models in dialog systems*, pages 74–107.

[Cohen and Kjeldsen, 1987] Cohen, P. and Kjeldsen, R. (1987). Information retrieval by constrained spreading activation in semantic networks. *Information processing & management*, 23(4):255–268.

[Colucci et al., 2007] Colucci, S., Di Noia, T., Di Sciascio, E., Donini, F., and Ragone, A. (2007). Measuring core competencies in a clustered network of knowledge. In *Knowledge management: innovation, technology and cultures: proceedings of the 2007 International Conference on Knowledge Management, Vienna, Austria, 27-28 August 2007*, page 279. World Scientific Pub Co Inc.

[Colucci et al., 2003] Colucci, S., Noia, T., Sciascio, E., Donini, F., Mongiello, M., and Mottola, M. (2003). A formal approach to ontology-based semantic match of skills descriptions. *J. UCS*, 9(12):1437–1454.

[Corbett and Anderson, 1994] Corbett, A. and Anderson, J. (1994). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278.

[Crestani, 1997] Crestani, F. (1997). Application of spreading activation techniques in information retrieval. *Artificial Intelligence Review*, 11(6):453–482.

[Crestani and Lee, 2000] Crestani, F. and Lee, P. (2000). Searching the web by constrained spreading activation. *Information Processing & Management*, 36(4):585–605.

[Crowder et al., 2009] Crowder, R., Wilson, M. L., Fowler, D., Shadbolt, N., Wills, G., and Wong, S. (2009). Navigation over a large ontology for industrial web applications. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. DETC2009-86544.

[Davenport and Prusak, 1998] Davenport, T. H. and Prusak, L. (1998). *Working Knowledge: How Organizations Manage What They Know*. Harvard Business School Press, Boston, MA.

[De Bra et al., 2003] De Bra, P., Aerts, A., Berden, B., De Lange, B., Rousseau, B., Santic, T., Smits, D., and Stash, N. (2003). Aha! the adaptive hypermedia architecture. In *Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*, pages 81–84. ACM.

[De Coi et al., 2007] De Coi, J., Herder, E., Koesling, A., Lofi, C., Olmedilla, D., Papapetrou, O., and Siberski, W. (2007). A model for competence gap analysis. In *Proceedings of the Third International Conference on Web Information Systems and Technologies: Internet Technology / Web Interface and Applications*. INSTICC Press.

[de Vasconcelos et al., 2009] de Vasconcelos, J., Kimble, C., Miranda, H., and Henriques, V. (2009). A knowledge-engine architecture for a competence management information system. In *UK Academy for Information Systems Conference Proceedings 2009*, page 14.

[Demartini, 2007] Demartini, G. (2007). Finding experts using wikipedia. In *Proceedings of the Workshop on Finding Experts on the Web with Semantics (FEWS2007) at ISWC/ASWC2007*, Busan, South Korea.

[d'Entremont and Storey, 2009] d'Entremont, T. and Storey, M.-A. (2009). Using a degree of interest model to facilitate ontology navigation. In *Visual Languages and Human-Centric Computing, 2009. VL/HCC 2009. IEEE Symposium on*, pages 127 –131.

[Dimitrova, 2003] Dimitrova, V. (2003). Style-olm: Interactive open learner modelling. *International Journal of Artificial Intelligence in Education (IJAIED)*, 13:35–78.

[Doan et al., 2003] Doan, A., Madhavan, J., Dhamankar, R., Domingos, P., and Halevy, A. (2003). Learning to match ontologies on the semantic web. *The VLDB Journal*, 12(4):303–319.

[Dochy et al., 1999] Dochy, F., Segers, M., and Sluijsmans, D. (1999). The use of self-, peer and co-assessment in higher education: a review. *Studies in Higher education*, 24(3):331–350.

[Dorn and Hochmeister, 2009] Dorn, J. and Hochmeister, M. (2009). Techscreen: Mining competencies in social software. In *Proceedings of the 3rd International Conference on Knowledge Generation, Communication and Management (KGCM)*, pages 115–126, Orlando, FLA.

[Dougherty, 1995] Dougherty, D. (1995). Managing your core incompetencies for corporate venturing. *Entrepreneurship Theory and Practice*, 19(3).

[Draganidis and Mentzas, 2006] Draganidis, F. and Mentzas, G. (2006). Competency based management: a review of systems and approaches. *Information Management & Computer Security*, 14(1):51–64.

[Du Plessis, 2007] Du Plessis, M. (2007). The role of knowledge management in innovation. *Journal of Knowledge Management*, 11(4):20–29.

[Dunning et al., 2004] Dunning, D., Heath, C., and Suls, J. (2004). Flawed self-assessment. *Psychological science in the public interest*, 5(3):69.

[Ehrlich et al., 2007] Ehrlich, K., Lin, C., and Griffiths-Fisher, V. (2007). Searching for experts in the enterprise: combining text and social network analysis. In *Proceedings of the 2007 international ACM conference on Supporting group work*, pages 117–126. ACM.

[Ehrlinger et al., 2008] Ehrlinger, J., Johnson, K., Banner, M., Dunning, D., and Kruger, J. (2008). Why the unskilled are unaware: Further explorations of (absent) self-insight among the incompetent. *Organizational behavior and human decision processes*, 105(1):98–121.

[Ellstrom, 1997] Ellstrom, P. (1997). The many meanings of occupational competence and qualification. *Journal of European Industrial Training, 21*, 6(7):266–273.

[Ernst et al., 2005] Ernst, N., Storey, M., and Allen, P. (2005). Cognitive support for ontology modeling. *International Journal of Human-Computer Studies*, 62(5):553–577.

[Falchikov, 1995] Falchikov, N. (1995). Peer feedback marking: developing peer assessment. *Programmed Learning*, 32(2):175–187.

[Falchikov and Boud, 1989] Falchikov, N. and Boud, D. (1989). Student self-assessment in higher education: A meta-analysis. *Review of Educational Research*, 59(4):395.

[Falchikov and Goldfinch, 2000] Falchikov, N. and Goldfinch, J. (2000). Student peer assessment in higher education: A meta-analysis comparing peer and teacher marks. *Review of educational research*, 70(3):287–322.

[Farrell et al., 2007] Farrell, S., Lau, T., Nusser, S., Wilcox, E., and Muller, M. (2007). Socially augmenting employee profiles with people-tagging. In *Proceedings of the 20th annual ACM symposium on User interface software and technology*, pages 91–100. ACM.

[Fernández-López and Gómez-Pérez, 2002] Fernández-López, M. and Gómez-Pérez, A. (2002). Overview and analysis of methodologies for building ontologies. *The Knowledge Engineering Review*, 17(2):129–156.

[Few, 2006] Few, S. (2006). *Information dashboard design: the effective visual communication of data*. O'Reilly Media, Inc.

[Foss and Knudsen, 1996] Foss, N. J. and Knudsen, C. (1996). *Towards a competence theory of the firm*. Routledge, London.

[Golemati et al., 2007] Golemati, M., Katifori, A., Vassilakis, C., Lepouras, G., and Halatsis, C. (2007). Creating an ontology for the user profile: Method and applications. In *Proceedings of the First RCIS Conference*, pages 407–412.

[Gomez-Perez et al., 2004] Gomez-Perez, A., Fernández-López, M., and Corcho, O. (2004). *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*. Springer Verlag.

[Gruber et al., 1993] Gruber, T. et al. (1993). A translation approach to portable ontology specifications. *Knowledge acquisition*, 5(2):199–220.

[Guarino, 1998] Guarino, N. (1998). Formal ontology in information systems. In *Proceedings of the First International Conference on Formal Ontology in Information Systems (FOIS)*, pages 3–15, Amsterdam. IOS Press.

[Hamming, 1950] Hamming, R. (1950). Error detecting and error correcting codes. *Bell System technical journal*, 29(2):147–160.

[Harper et al., 2008] Harper, F., Raban, D., Rafaeli, S., and Konstan, J. (2008). Predictors of answer quality in online q&a sites. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 865–874. ACM.

[Harzallah et al., 2002] Harzallah, M., Leclère, M., and Trichet, F. (2002). Commoncv: modelling the competencies underlying a curriculum vitae. In *Proceedings of the 14th international conference on Software engineering and knowledge engineering*, pages 65–71. ACM.

[Haselmann et al., 2011] Haselmann, T., Winkelmann, A., and Vossen, G. (2011). Towards a conceptual model for trustworthy skills profiles in online social networks. *Information Systems Development*, pages 285–296.

[Hevner et al., 2004] Hevner, A. R., March, S. T., Park, J., and Ram, S. (2004). Design science in information systems research. *Management Information Systems Quarterly*, 28(1):75–106.

[Hirst and St-Onge, 1998] Hirst, G. and St-Onge, D. (1998). Lexical chains as representations of context for the detection and correction of malapropisms. *WordNet: An electronic lexical database*, 13:305–332.

[Hochmeister, 2011] Hochmeister, M. (2011). Mining user knowledge in learning networks. In Niedrite, L., Strazdina, R., and Wangler, B., editors, *Proceedings of the 2nd International Workshop on Intelligent Educational Systems and Technology-Enhanced Learning (INTEL-EDU) at BIR 2011*, pages 267–274.

[Hochmeister, 2012a] Hochmeister, M. (2012a). Calculate learners' competence scores and their reliability in learning networks. In Niedrite, L., Strazdina, R., and Wangler, B., editors, *BIR 2011 Workshops - Revised Selected Papers*, LNBIP 106, pages 171–183, Berlin Heidelberg. Springer-Verlag.

[Hochmeister, 2012b] Hochmeister, M. (2012b). Spreading expertise scores in overlay learner models. In Helfert, M., Martins, M. J., and Cordeiro, J., editors, *Proceedings of the 4th International Conference on Computer Supported Education (CSEDU)*, volume 1, pages 175–180, Porto, Portugal.

[Hochmeister and Daxböck, 2011] Hochmeister, M. and Daxböck, J. (2011). A user interface for semantic competence profiles. In *Proceedings of the 19th international conference on User modeling, adaption, and personalization*, pages 159–170. Springer.

[Hochmeister et al., 2012] Hochmeister, M., Daxböck, J., and Kay, J. (2012). Using expertise predictions to facilitate self-regulated learning. In *Proceedings of the 4th Workshop on Metacognition and Self-Regulated Learning in Educational Technologies in conjunction with the 11th International Conference on Intelligent Tutoring Systems (ITS) 2012*, page to appear.

[Horowitz and Kamvar, 2012] Horowitz, D. and Kamvar, S. (2012). Searching the village: models and methods for social search. *Communications of the ACM*, 55(4):111–118.

[Horvath and Sternberg, 1999] Horvath, J. and Sternberg, R. (1999). Tacit knowledge in the profession. *Sternberg, R. and Horvath, J., Tacit knowledge in professional practice, Laurence Erlbaum, London*.

[Hotho et al., 2005] Hotho, A., Nürnberger, A., and Paaß, G. (2005). A brief survey of text mining. *Machine Learning*, 20(1):19–62.

[Hu et al., 2007] Hu, M., Lim, E., Sun, A., Lauw, H., and Vuong, B. (2007). Measuring article quality in wikipedia: models and evaluation. In *Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management*, pages 243–252. ACM.

[Hussein and Ziegler, 2008] Hussein, T. and Ziegler, J. (2008). Adapting web sites by spreading activation in ontologies. In *Proceedings of International Workshop on Recommendation and Collaboration, New York, USA*.

[Jameson, 1995] Jameson, A. (1995). Numerical uncertainty management in user and student modeling: An overview of systems and issues. *User Modeling and User-Adapted Interaction*, 5(3):193–251.

[Jiang and Conrath, 1997] Jiang, J. and Conrath, D. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the International Conference Research on Computational Linguistics (ROCLING)*, Taiwan.

[Jiao et al., 2009] Jiao, J., Yan, J., Zhao, H., and Fan, W. (2009). Expertrank: An expert user ranking algorithm in online communities. In *New Trends in Information and Service Science, 2009. NISS'09. International Conference on*, pages 674–679. IEEE.

[Kao et al., 2010] Kao, W., Liu, D., and Wang, S. (2010). Expert finding in question-answering websites: a novel hybrid approach. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, pages 867–871. ACM.

[Katifori et al., 2007] Katifori, A., Halatsis, C., Lepouras, G., Vassilakis, C., and Giannopoulou, E. (2007). Ontology visualization methods—a survey. *ACM Computing Surveys*, 39(4):10.

[Kay, 2008] Kay, J. (2008). Lifelong learner modeling for lifelong personalized pervasive learning. *Learning Technologies, IEEE Transactions on*, 1(4):215–228.

[Kay et al., 2007] Kay, J., Li, L., and Fekete, A. (2007). Learner reflection in student self-assessment. In *Proceedings of the ninth Australasian conference on Computing education-Volume 66*, pages 89–95. Australian Computer Society, Inc.

[Kay and Lum, 2005a] Kay, J. and Lum, A. (2005a). Exploiting readily available web data for reflective student models. In *Proceedings of AIED 2005, Artificial Intelligence in Education*, pages 338–345, Amsterdam, The Netherlands. IOS Press.

[Kay and Lum, 2005b] Kay, J. and Lum, A. (2005b). Exploiting readily available web data for scrutable student models. In *Proceedings of the 2005 conference on Artificial Intelligence in Education: Supporting Learning through Intelligent and Socially Informed Technology*, pages 338–345. IOS Press.

[Kay and Lum, 2005c] Kay, J. and Lum, A. (2005c). Ontology-based user modelling for the semantic web. In *Proceedings of the Workshop on Personalisation on the Semantic Web: Per−SWeb05*, pages 15–23.

[Kleinberg, 1999] Kleinberg, J. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632.

[Kleitman, 2008] Kleitman, S. (2008). *Metacognition in the Rationality Debate: self-confidence and its Calibration*. VDM Verlag.

[Lassila and McGuinness, 2001] Lassila, O. and McGuinness, D. (2001). The role of frame-based representation on the semantic web. *Linköping Electronic Articles in Computer and Information Science*, 6(5):2001.

[Lave and Wenger, 1991] Lave, J. and Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge University Press.

[Le Deist and Winterton, 2005] Le Deist, F. and Winterton, J. (2005). What is competence? *Human Resource Development International*, 8(1):27–46.

[Levenshtein, 1966] Levenshtein, V. (1966). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710.

[Ley and Albert, 2003] Ley, T. and Albert, D. (2003). Identifying employee competencies in dynamic work domains: methodological considerations and a case study. *J. UCS*, 9(12):1500–1518.

[Liao et al., 1999] Liao, M., Hinkelmann, K., Abecker, A., and Sintek, M. (1999). A competence knowledge base system as part of the organizational memory. *XPS-99: Knowledge-Based Systems*, pages 125–137.

[Lim et al., 2006] Lim, E., Vuong, B., Lauw, H., and Sun, A. (2006). Measuring qualities of articles contributed by online communities. In *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*, pages 81–87. IEEE Computer Society.

[Lindgren et al., 2004] Lindgren, R., Henfridsson, O., and Schultze, U. (2004). Design principles for competence management systems: a synthesis of an action research study. *MIS quarterly*, 28(3):435–472.

[Lindgren et al., 2003] Lindgren, R., Stenmark, D., and Ljungberg, J. (2003). Rethinking competence systems for knowledge-based organizations. *European Journal of Information Systems*, 12(1):18–29.

[Liu and Maes, 2005] Liu, H. and Maes, P. (2005). Interestmap: Harvesting social network profiles for recommendations. In *Beyond Personalization - IUI 2005*, San Diego, California, USA.

[Liu et al., 2005] Liu, W., Weichselbraun, A., Scharl, A., and Chang, E. (2005). Semi-automatic ontology extension using spreading activation. *Journal of Universal Knowledge Management*, 1:50–58.

[Lu et al., 2009] Lu, Y., Quan, X., Ni, X., Liu, W., and Xu, Y. (2009). Latent link analysis for expert finding in user-interactive question answering services. In *Semantics, Knowledge and Grid, 2009. SKG 2009. Fifth International Conference on*, pages 54–59. IEEE.

[Mabbott and Bull, 2006] Mabbott, A. and Bull, S. (2006). Student preferences for editing, persuading, and negotiating the open learner model. In *Proceedings of ITS*, pages 481–490. Springer.

[Macdonald and Ounis, 2006] Macdonald, C. and Ounis, I. (2006). Voting for candidates: adapting data fusion techniques for an expert search task. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 387–396. ACM.

[MacIntyre et al., 1997] MacIntyre, P., Noels, K., and Clément, R. (1997). Biases in self-ratings of second language proficiency: The role of language anxiety. *Language learning*, 47(2):265–287.

[Maedche and Staab, 2002] Maedche, A. and Staab, S. (2002). Measuring similarity between ontologies. *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, pages 15–21.

[Maguitman et al., 2005] Maguitman, A., Menczer, F., Roinestad, H., and Vespignani, A. (2005). Algorithmic detection of semantic similarity. In *Proceedings of the 14th international conference on World Wide Web, May*, pages 10–14.

[Manouselis et al., 2011] Manouselis, N., Drachsler, H., Vuorikari, R., Hummel, H., and Koper, R. (2011). Recommender systems in technology enhanced learning. *Recommender Systems Handbook*, pages 387–415.

[Maybury, 2006] Maybury, M. (2006). Expert finding systems. *MITRE Center for Integrated Intelligence Systems Bedford, Massachusetts, USA*.

[Maylett, 2009] Maylett, T. (2009). 360-degree feedback revisited: The transition from development to appraisal. *Compensation & Benefits Review*, 41(5):52.

[Mazuel and Sabouret, 2008] Mazuel, L. and Sabouret, N. (2008). Semantic relatedness measure using object properties in an ontology. *The Semantic Web-ISWC 2008*, pages 681–694.

[McDonald and Ackerman, 1998] McDonald, D. and Ackerman, M. (1998). Just talk to me: a field study of expertise location. In *Proceedings of the 1998 ACM conference on Computer supported cooperative work*, pages 315–324. ACM.

[McDonald and Ackerman, 2000] McDonald, D. and Ackerman, M. (2000). Expertise recommender: a flexible recommendation system and architecture. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, pages 231–240. ACM.

[McLure Wasko and Faraj, 2000] McLure Wasko, M. and Faraj, S. (2000). It is what one does why people participate and help others in electronic communities of practice. *The Journal of Strategic Information Systems*, 9(2):155–173.

[Miller and Shamsie, 1996] Miller, D. and Shamsie, J. (1996). The resource-based view of the firm in two environments: The hollywood film studios from 1936 to 1965. *Academy of management Journal*, pages 519–543.

[Mockus and Herbsleb, 2002] Mockus, A. and Herbsleb, J. (2002). Expertise browser: a quantitative approach to identifying expertise. In *Proceedings of the 24th International Conference on Software Engineering*, pages 503–512. ACM.

[Mohamed et al., 2006] Mohamed, A. H., Leeb, S. P., and Salimc, S. S. (2006). An ontology-based knowledge model for software experience management. *International Journal of the Computer, the Internet and Management*, 14(3):79–88.

[Neches et al., 1991] Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Swartout, W., et al. (1991). Enabling technology for knowledge sharing. *AI magazine*, 12(3):36.

[Nonaka and Takeuchi, 1995] Nonaka, I. and Takeuchi, H. (1995). *The knowledge-creating company: How Japanese companies create the dynamics of innovation*. Oxford University Press, USA.

[Oliveira et al., 2006] Oliveira, J., de Souza, J., Miranda, R., Rodrigues, S., Kawamura, V., Martino, R., Mello, C., Krejci, D., Barbosa, C., and Maia, L. (2006). Gcc: a knowledge management environment for research centers and universities. *Frontiers of WWW Research and Development-APWeb 2006*, pages 652–667.

[Othman et al., 2008] Othman, R., Deris, S., and Illias, R. (2008). A genetic similarity algorithm for searching the gene ontology terms and annotating anonymous protein sequences. *Journal of Biomedical Informatics*, 41(1):65–81.

[Pal and Konstan, 2010] Pal, A. and Konstan, J. (2010). Expert identification in community question answering: exploring question selection bias. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1505–1508. ACM.

[Pearl, 1988] Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann.

[Pernici et al., 2006] Pernici, B., Locatelli, P., and Marinoni, C. (2006). The ecco system: an ecompetence management tool based on semantic networks. In *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops*, pages 1088–1099. Springer.

[Pirolli, 2007] Pirolli, P. (2007). *Information foraging theory: Adaptive interaction with information*. Oxford University Press, USA.

[Pirrò, 2009] Pirrò, G. (2009). A semantic similarity metric combining features and intrinsic information content. *Data Knowl. Eng.*, 68(11):1289–1308.

[Plant, 2004] Plant, R. (2004). Online communities. *Technology in Society*, 26(1):51–65.

[Polanyi, 1966] Polanyi, M. (1966). *The tacit dimension*. Doubleday.

[Probst et al., 2006] Probst, G., Raub, S., and Romhardt, K. (2006). *Wissen managen: wie Unternehmen ihre wertvollste Ressource optimal nutzen*. Gabler.

[Rada et al., 1989] Rada, R., Mili, H., Bicknell, E., and Blettner, M. (1989). Development and application of a metric on semantic nets. *Systems, Man and Cybernetics, IEEE Transactions on*, 19(1):17–30.

[Razmerita et al., 2003] Razmerita, L., Angehrn, A., and Maedche, A. (2003). Ontology-based user modeling for knowledge management systems. *User Modeling 2003*, pages 148–148.

[Reichling and Wulf, 2009] Reichling, T. and Wulf, V. (2009). Expert recommender systems in practice: evaluating semi-automatic profile generation. In *Proceedings of the 27th international conference on Human factors in computing systems*, pages 59–68. ACM.

[Reinhardt and North, 2003] Reinhardt, K. and North, K. (2003). Transparency and transfer of individual competencies - a concept of integrative competence management. *J. UCS*, 9(12):1372–1380.

[Resnik, 1995] Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. In *In Proceedings of the 14th International Joint Conference on Artificial Intelligence*.

[Rich, 1979] Rich, E. (1979). User modeling via stereotypes. *Cognitive science*, 3(4):329–354.

[Rodrigues et al., 2008] Rodrigues, E., Milic-Frayling, N., and Fortuna, B. (2008). Social tagging behaviour in community-driven question answering. In *Web Intelligence and Intelligent Agent Technology, 2008. WI-IAT'08. IEEE/WIC/ACM International Conference on*, volume 1, pages 112–119. IEEE.

[Rodrigues et al., 2006] Rodrigues, S., Oliveira, J., and de Souza, J. (2006). Recommendation for team and virtual community formations based on competence mining. *Computer Supported Cooperative Work in Design II*, pages 365–374.

[Schickel-Zuber and Faltings, 2007] Schickel-Zuber, V. and Faltings, B. (2007). Oss: a semantic similarity function based on hierarchical ontologies. In *Proceedings of the 20th international joint conference on Artifical intelligence*, pages 551–556.

[Schmidt and Braun, 2008] Schmidt, A. and Braun, S. (2008). People tagging & ontology maturing: Towards collaborative competence management. In *8th International Conference on the Design of Cooperative Systems (COOP 2008), Carry-le-Rouet*.

[Schraw et al., 2006] Schraw, G., Crippen, K., and Hartley, K. (2006). Promoting self-regulation in science education: Metacognition as part of a broader perspective on learning. *Research in Science Education*, 36(1):111–139.

[Seid and Kobsa, 2003] Seid, D. Y. and Kobsa, A. (2003). *Expert Finding Systems for Organizations: Problem and Domain Analysis and the DEMOIR Approach*, pages 327–358. MIT Press, Cambridge, MA, USA.

[Shafer, 1976] Shafer, G. (1976). *A mathematical theory of evidence*, volume 1. Princeton university press Princeton.

[Shami et al., 2009] Shami, N., Ehrlich, K., Gay, G., and Hancock, J. (2009). Making sense of strangers' expertise from signals in digital artifacts. In *Proceedings of the 27th international conference on Human factors in computing systems*, pages 69–78. ACM.

[Shannon, 2001] Shannon, C. (2001). A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55.

[Sharratt and Usoro, 2003] Sharratt, M. and Usoro, A. (2003). Understanding knowledge-sharing in online communities of practice. *Electronic Journal on Knowledge Management*, 1(2):187–196.

[Shneiderman, 2002] Shneiderman, B. (2002). The eyes have it: A task by data type taxonomy for information visualizations. In *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pages 336–343. IEEE.

[Sieg et al., 2007] Sieg, A., Mobasher, B., and Burke, R. (2007). Web search personalization with ontological user profiles. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 525–534. ACM.

[Smith, 2001] Smith, E. (2001). The role of tacit and explicit knowledge in the workplace. *Journal of Knowledge Management*, 5(4):311–321.

[Song et al., 2005] Song, X., Tseng, B., Lin, C., and Sun, M. (2005). Expertisenet: Relational and evolutionary expert modeling. *User Modeling 2005*, pages 150–150.

[Spender, 1996] Spender, J. (1996). Organizational knowledge, learning and memory: three concepts in search of a theory. *Journal of organizational change management*, 9(1):63–78.

[Staab and Studer, 2009] Staab, S. and Studer, R. (2009). *Handbook on Ontologies*. Springer Publishing Company, Incorporated, 2nd edition.

[Stankovic et al., 2010] Stankovic, M., Wagner, C., Jovanovic, J., and Laublet, P. (2010). Looking for experts? what can linked data do for you. *Proceedings of the Linked Data on the Web (LDOW2010)*.

[Stenmark, 2000] Stenmark, D. (2000). Leveraging tacit organizational knowledge. *Journal of management information systems*, 17(3):9–24.

[Storey et al., 2001] Storey, M., Musen, M., Silva, J., Best, C., Ernst, N., Fergerson, R., and Noy, N. (2001). Jambalaya: Interactive visualization to enhance ontology authoring and knowledge acquisition in protégé. In *Workshop on Interactive Tools for Knowledge Capture (K-CAP-2001)*. Citeseer.

[Sun et al., 2009] Sun, K., Cao, Y., Song, X., Song, Y., Wang, X., and Lin, C. (2009). Learning to recommend questions based on user ratings. In *Proceeding of the 18th ACM conference on Information and knowledge management*, pages 751–758. ACM.

[Sure et al., 2000] Sure, Y., Maedche, A., and Staab, S. (2000). Leveraging corporate skill knowledge-from proper to ontoproper. In *Proceedings of the Third International Conference on Practical Aspects of Knowledge Management. Basel, Switzerland.*

[Sussna, 1993] Sussna, M. (1993). Word sense disambiguation for free-text indexing using a massive semantic network. In *Proceedings of the second international conference on Information and knowledge management*, pages 67–74. ACM.

[Swartout et al., 1996] Swartout, B., Patil, R., Knight, K., and Russ, T. (1996). Toward distributed use of large-scale ontologies. In *Proc. of the Tenth Workshop on Knowledge Acquisition for Knowledge-Based Systems*.

[Tarasov et al., 2007] Tarasov, V., Albertsen, T., Kashevnik, A., Sandkuhl, K., Shilov, N., and Smirnov, A. (2007). Ontology-based competence management for team configuration. *Holonic and Multi-Agent Systems for Manufacturing*, pages 401–410.

[Taylor and Richards, 2009] Taylor, M. and Richards, D. (2009). Discovering areas of expertise from publication data. *Knowledge Acquisition: Approaches, Algorithms and Applications*, pages 218–230.

[Thiagarajan et al., 2008] Thiagarajan, R., Manjunath, G., and Stumptner, M. (2008). Finding experts by semantic matching of user profiles. In *The 7th International Semantic Web Conference*.

[Toegel and Conger, 2003] Toegel, G. and Conger, J. (2003). 360-degree assessment: Time for reinvention. *Academy of Management Learning & Education*, pages 297–311.

[Topping, 1998] Topping, K. (1998). Peer assessment between students in colleges and universities. *Review of Educational Research*, 68(3):249.

[Tsoukas and Vladimirou, 2001] Tsoukas, H. and Vladimirou, E. (2001). What is organizational knowledge? *Journal of management studies*, 38(7):973–993.

[Tsujii and Ananiadou, 2005] Tsujii, J. and Ananiadou, S. (2005). Thesaurus or logical ontology, which one do we need for text mining? *Language resources and evaluation*, 39(1):77–90.

[Tversky, 1977] Tversky, A. (1977). Features of similarity. *Psychological review*, 84(4):327.

[Uschold and Gruninger, 1996] Uschold, M. and Gruninger, M. (1996). Ontologies: Principles, methods and applications. *Knowledge engineering review*, 11(2):93–136.

[Uschold and Gruninger, 2004] Uschold, M. and Gruninger, M. (2004). Ontologies and semantics for seamless connectivity. *ACM SIGMod Record*, 33(4):58–64.

[Uschold et al., 1998] Uschold, M., King, M., Moralee, S., and Zorgios, Y. (1998). The enterprise ontology. *The knowledge engineering review*, 13(1):31–89.

[Vivacqua and Lieberman, 2000] Vivacqua, A. and Lieberman, H. (2000). Agents to assist in finding help. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 65–72. ACM.

[Ward et al., 2002] Ward, M., Gruppen, L., and Regehr, G. (2002). Measuring self-assessment: current state of the art. *Advances in Health Sciences Education*, 7(1):63–80.

[Wasko and Faraj, 2005] Wasko, M. and Faraj, S. (2005). Why should i share? examining social capital and knowledge contribution in electronic networks of practice. *Mis Quarterly*, pages 35–57.

[Waterman, 1986] Waterman, D. (1986). *A guide to expert systems*. Addison Wesley Publishing Company.

[Weinert, 2001] Weinert, F. E. (2001). Concept of competence: A conceptual clarification. In Rychen, D. S. and Salganik, L. H., editors, *Defining and selecting key competences*, pages 45–65. Hogrefe & Huber, Seattle, WA.

[Wenger et al., 2002] Wenger, E., McDermott, R., and Snyder, W. (2002). *Cultivating communities of practice: A guide to managing knowledge*. Harvard Business Press.

[Willett et al., 2007] Willett, W., Heer, J., and Agrawala, M. (2007). Scented widgets: Improving navigation cues with embedded visualizations. *IEEE Transactions on Visualization and Computer Graphics*, pages 1129–1136.

[Wöhner and Peters, 2009] Wöhner, T. and Peters, R. (2009). Assessing the quality of wikipedia articles with lifecycle based metrics. In *Proceedings of the 5th International Symposium on Wikis and Open Collaboration*, WikiSym '09, New York, NY, USA. ACM.

[Wu and Palmer, 1994] Wu, Z. and Palmer, M. (1994). Verbs semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 133–138. Association for Computational Linguistics.

[Zack, 1999] Zack, M. (1999). Managing codified knowledge. *Sloan management review*, 40(4):45–58.

[Zadeh, 1994] Zadeh, L. (1994). Fuzzy logic, neural networks, and soft computing. *Communications of the ACM*, 37(3):77–84.

[Zapata-Rivera and Greer, 2004] Zapata-Rivera, J. and Greer, J. (2004). Interacting with inspectable bayesian student models. *International Journal of Artificial Intelligence in Education*, 14(2):127–163.

[Zhang et al., 2007] Zhang, J., Ackerman, M., and Adamic, L. (2007). Expertise networks in online communities: structure and algorithms. In *Proceedings of the 16th international conference on World Wide Web*, pages 221–230. ACM.

[Zhu et al., 2005] Zhu, J., Goncalves, A., Uren, V., Motta, E., and Pacheco, R. (2005). Mining web data for competency management. In *Web Intelligence, 2005. Proceedings. The 2005 IEEE/WIC/ACM International Conference on*, pages 94–100. IEEE.

Curriculum Vitae

Martin Hochmeister was born on September 21, 1976 in Vienna, Austria. He received his high-school diploma in 1996 and immediately afterwards completed his compulsory military service. In 1997, he started to work as a software designer at Kapsch AG, Vienna, Austria, where he was involved in the development of telecommunication services. In 2000, he became the manager of a joint project implementing the number portability feature for mobile networks together with the North American telecom vendor Nortel Networks. The project was conducted in Ottawa, Canada. After his return in 2001, Martin took over responsibility for developing the product line of Interactive Voice Response systems. As a product manager, he was primarily in charge of the conception and negotiation of new feature sets with existing and potential customers. In 2002, he changed back to R&D and designed solutions for the integration of heterogenous systems in the field of Next Generation Intelligent Networks. From 2006 to 2010, Martin worked as a free consultant in the field of Semantic Web technologies as well as for companies in need of common IT consultation involving tasks such as migrating from ISDN to Voice-over-IP solutions or setting up service level agreements. Since then, he has been employed as a project assistant at the Vienna University of Technology doing research on information extraction, user modeling and online communities.

Besides his professional work, Martin participated in several study programs. He received master's degrees from the University of Applied Sciences Technikum Vienna in *Information and Communication Systems* (2006) as well as in *Information Systems Management* (2008). He also received a master's degree from the Vienna University of Technology in *Computer Science Management* (2008). Since then he has been enrolled in a doctoral program at the Vienna University of Technology. Martin is a member of the Project Management Institute (PMI) and holds the certificate of a Project Management Professional (PMP).

Publications

M. Hochmeister. *Design and implementation of a resource adapter to integrate a next generation intelligent network system with the Microsoft Exchange Server*. Diploma Thesis. University of Applied Sciences Technikum Vienna, Austria, 2006.

M. Hochmeister. *Design aspects of a dynamic skills management system for the computer-aided configuration of teams based on semantic technologies and social network analysis*. Master Thesis. University of Applied Sciences Technikum Vienna, Austria, 2008.

A. Blumauer and M. Hochmeister. *Tag-recommender gestützte Annotation von Web-Dokumenten*. Social Semantic Web, pages 227–243, 2009. Springer-Verlag.

M. Hochmeister. *Bahn frei für ein smarteres Web*. Interview in HR Today Special: HR Systeme, Ausgabe 4/2009.

J. Dorn and M. Hochmeister. *Techscreen: Mining competencies in social software*. In Proceedings of the 3rd International Conference on Knowledge Generation, Communication and Management (KGCM), pages 115–126, Orlando, FLA, 2009.

M. Hochmeister and J. Daxböck. *A user interface for semantic competence profiles*. In Proceedings of the 19th international conference on User modeling, adaption, and personalization, pages 159–170. Springer, 2011. (*including oral presentation*)

M. Hochmeister. *A knowledge-based expertise recommender system*. Invited talk at the Institute for Innovation in Business and Social Research (IIBSoR), Faculty of Commerce, University of Wollongong, Australia, April 2011.

M. Hochmeister. *Mining user knowledge in learning networks*. In L. Niedrite, R. Strazdina, and B. Wangler, editors, Proceedings of the 2nd International Workshop on Intelligent Educational Systems and Technology-Enhanced Learning (INTEL-EDU) at BIR 2011, pages 267–274, 2011. (*including oral presentation*)

J. Patalas-Maliszewska and M. Hochmeister. *Modeling strategic-knowledge-resource management based on individual competencies in SMEs*. Contemporary Economics, 5(2), 2011.

M. Hochmeister. *Calculate learners' competence scores and their reliability in learning networks*. In L. Niedrite, R. Strazdina, and B. Wangler, editors, BIR 2011 Workshops - Revised Selected Papers, LNBIP 106, pages 171–183, Berlin Heidelberg, 2012. Springer-Verlag.

M. Hochmeister. *Spreading expertise scores in overlay learner models*. In M. Helfert, M. J. Martins, and J. Cordeiro, editors, Proceedings of the 4th International Conference on Computer Supported Education (CSEDU), volume 1, pages 175–180, Porto, Portugal, 2012. (*including oral presentation*)

M. Hochmeister, J. Daxböck, and J. Kay. Using expertise predictions to facilitate self-regulated learn- ing. In The 4th Workshop on Metacognition and Self-Regulated Learning in Educational Technologies in conjunction with the 11th International Conference on Intelligent Tutoring Systems (ITS) 2012, page to appear, 2012. (*including oral presentation*)