Recommendations in Online Discussion Forums for E-Learning Systems

Fabian Abel, Ig Ibert Bittencourt, Evandro Costa, Nicola Henze, Daniel Krause, and Julita Vassileva

Abstract—In this paper, we outline the importance of discussion fora for e-learning applications. Due to a weak structure or size of the discussion forum, recommendations are required in order to help learners finding relevant information within a forum. We present a generic personalization framework and evaluate the framework based on a recommender architecture for the e-learning focused discussion forum Comtella-D. In the evaluation, we examine different sources of user feedback and the required amount of user interaction to provide recommendations. The outcomes of the evaluation serve as source for a personalization rule, which selects the most appropriate recommendation strategy based on available user input data. We furthermore conclude that collaborative filtering techniques can be utilize successfully in small data sets, like e-learning related discussion fora.

Index Terms—Information filtering, user profiles and alert services, online information services.

✦

1   INTRODUCTION

CURRENT e-learning systems focus on supporting the creation and presentation of learning materials. The communication between the learners, which is also an important factor for a successful learning experience [1], is not taken into account seriously enough in today's E-Learning systems.

Discussion fora provide unique communication features, which make them a perfect candidate for handling communication in an e-learning environment: Forums allow for asynchronous communication, and hence, do not force learners to negotiate for specific communication times as this would be required in chats. Furthermore, as all exchanged knowledge is stored within a forum, it can be explored or searched by all users at any time. This could possibly give an immediate answer for frequently asked questions.

Another strong benefit of discussion fora is the tree-like structure. While a discussion forum usually has an overall topic, user can further divide the forum into subfora where specific subtopics can be discussed. Within the subfora, different discussions can be distinguished by so-called threads. This structure enables learners to browse through discussion topics quickly and further to navigate directly to relevant topics. Thus, users are not overwhelmed by unrelated information as this could happen in mailing lists where users can only decide to opt-in and receive all mails or opt-out and receive none of the mails.

● *F. Abel, D. Krause, and N. Henze are with IVS—Semantic Web Group, Leibniz University of Hannover, Appelstr. 4, D-30167 Hannover, Germany. E-mail: {abel, krause, henze}@kbs.uni-hannover.de.*
● *I.I. Bittencourt and E. Costa are with the Computing Institute, Federal University of Alagoas, BR 104 Norte km 97, 57072-970 Maceío, AL, Brazil. E-mail: ibert@dsc.ufcg.edu.br, evandro@fapeal.br.*
● *J. Vassileva is with the Department of Computer Science, University of Saskatechwan, 110 Science Place, Saskatoon SK S7N 5C9, Canada. E-mail: jiv@cs.usask.ca.*

Drawbacks of the structure arise when 1) users start a discussion in a wrong thread, 2) a topic would fit in multiple threads, or 3) the forum becomes so big that the structure can not be overlooked by the users. In such a case, learners could possibly miss relevant information or have to spend much time to find all relevant information. In such a situation keyword-based search, which is implemented in most of the current discussion board systems, is not an appropriate solution as most users can hardly express their interests through keyword-based queries [2].

A promising approach to match users and relevant threads is to use collaborative filtering techniques. However, the e-learning domain is very different from other domains where recommender systems perform well: As most e-learning systems (like our Comtella-D [3] system) are used to support university courses, the number of users is relatively small in comparison to typical recommender scenarios, like large online stores. Hence, recommender systems need to create recommendations based on a small amount of input data, and hence, may be inappropriate to generate high-quality recommendations.

Furthermore, users in online communities, like fora, are not homogeneous [4]. There are some users who actively contribute new content while other users seldomly or never publish their own content. Those who never publish content on their own may or may not rate at least the content of other users. With such heterogeneous kind and amount of input data from the users, the question arises whether a single recommendation algorithm can be appropriate to generate recommendations.

In this paper, we evaluate different recommendation strategies for the Comtella-D system, which is a discussion board with e-learning specific extensions. We show that the behavior of the users, and hence, the characteristics of the user profiles influence the quality of different recommendation algorithms. We show in our experiments, based on data from a lecture "Ethics and IT," given at the University of Saskatchewan in spring 2006, how an e-learning system can create the most appropriate recommendations by selecting

different recommendation techniques in a flexible, rule-based manner.

Many e-learning systems cannot use general purpose discussion fora as they would not fit in the e-learning systems' data structure, programming style or due to license issues. We expect that most discussion fora will be created from scratch or adapted with specific extensions, and hence, limit reusability of tightly integrated algorithms. Therefore, we propose a Web Service-based solution that is loosely coupled and encapsulates recommendation functionality in order to make it reusable. Therefore, the concept of so-called Personalization Services form the Personal Reader architecture was implemented. Thus, different discussion fora, as well as other e-learning systems, can benefit from the personalization features, offered by such a framework. Furthermore, by introduction personalization rules, we make the offered functionality adjustable while applications and services can still use their existing ontologies and interfaces.

The paper is structured as follows: In Section 2, we refer to related work in the in the e-learning domain in general and specifically in the area of recommendations for discussion fora. Furthermore, we present different approaches to select or combine multiple personalization techniques based on recommender algorithms. Section 3 describes the Comtella-D discussion board and outlines the advantages of this system for the e-learning domain. We furthermore present the Personal Reader Framework, a generic Semantic Web based framework for building applications that provide personalization and extend it to enable a dynamic discovery and invocation of generic personalization algorithms. We evaluate in Section 4 how an application can be configured to take advantage of using the Personal Reader Framework. Therefore, we create a Comtella-D specific rule that describes the conditions to select the most appropriate personalization algorithm.

2 RELATED WORK

In this section, we describe the current usage of discussion fora and recommender algorithms within the e-learning area. Furthermore, we discuss how to combine different recommendation techniques in a generic way and how to make recommendation algorithms reusable in order to be utilized in different e-learning applications.

2.1 Discussion Fora in an E-Learning Environment

Online discussion fora are a valuable component for every e-learning System. Due to its unique properties, they can foster discussion, cooperation, and collaboration among learners. Some of these properties are:

- *Asynchronous messages:* Learners can decide on their own if and when they consume content, create content on their own, or rate content of other learners [5].
- *Feedback to the teacher:* While some students encounter barriers to address questions directly to the teacher, they might publish them in a forum where learners can help each other. By observing the ongoing

discussions, teachers can immediately detect opaque learning content [6].

- *Motivation for the learners:* Discussion fora motivate learners [7] in two ways. First, active discussion fora have every minute new content, and thus, make it more attractive for a visitor to come back later. Second, whenever a learner expressed his own opinion she tends to defend this opinion against others. Hence, active learners will contribute quite regularly.

This unique combination of features turns discussion fora into a prominent object of research in the e-learning area: In [8] the authors have shown that participation in discussion fora can improve the learning performance while [9] states that discussion fora can foster collaborative learning. Some authors consider discussion fora even as learning resources on their own [10], [6], which outlines the capability of discussion fora to collect, distribute, and archive learning content.

2.2 Recommendations in an E-Learning Environment

In the e-learning domain, different recommendation scenarios are covered by the literature: Recommender systems can be used to recommend appropriate learning resources [11], [12], to recommend actions that a learner should perform within the e-learning environment [13] or even to extend the content of an e-learning system by related learning resources from the Web [14].

However, the number of approaches that are purely based on collaborative filtering, like [15] is limited in the e-learning domain. The reason for this is that either the explicit ratings by the users are missing [13] or that there are not enough users in the system. Hence, systems extend or replace collaborative filtering by clustering, stereotyping, or rule-based techniques.

2.3 Recommendations in Discussion Forums

Recommendations within social structures, which are similar to discussion fora, are created by different approaches: [16] describes collaborative filtering in the Usenet system to select relevant emails while [17] combines collaborative algorithms and NLP to predict emerging trends. However, all of these systems are in domains where a large group of comparable users exists. Hence, these approaches cannot be utilized in the e-learning domain without further clarification.

In [18], first approaches that show how to use collaborative filtering in e-learning can be found. However, they have not yet been implemented. To the best of our knowledge, no e-learning specific discussion forum was extended by recommendations based on collaborative filtering.

2.4 Generic Personalization Frameworks

In fact, different personalization techniques are already combined to overcome the disadvantages of single personalization techniques. In the domain of recommender systems, these combination techniques are known as hybrid recommender systems [19], utilizing for example both, collaborative and content-based recommender systems to overcome the *new item* or *new user* problem. Indeed, there

are several proposals [20], [21], [22], [23], [24] using a hybrid approach. However, all of these approaches are statically, i.e., they are not able to select the most appropriate technique (be it a specific hybrid recommender system or some atomic recommender algorithm) based on available input data.

Furthermore, most systems integrate personalization functionality directly in the application. Thus, every application needs to reimplement personalization functionality. For specific domains, like the e-learning domain, there exist approaches that realize such a decoupling [25]. Approaches that aim on decoupling personalization functionality in general are proposed in the Personal Reader Framework [26], [27], which introduce the concepts of encapsulated Personalization Services.

In [28], a framework for automatic online personalization is proposed. The idea is to provide interfaces to personalization aspects of the interface, site features, message boards, or even other users through the use of recommender systems and also to provide data sets to integrate several artificial intelligence techniques in recommendation. Nevertheless, they do not provide any kind of semantics to provide the automatic discovery, composition, and invocation of those data sets.

In [29], rule-based recommender systems are described. The goal is to handle three issues: data source, data modeling, and recommendation strategy. However, the system is not decoupled and therewith does not provide the same degree of extensibility and flexibility as our approach.

3 THE RULE-BASED THREAD RECOMMENDER

In this section, we briefly describe the used discussion forum, called Comtella-D, and discuss characteristics that turn Comtella-D into a perfect candidate for creating recommendations. In the section part, we utilize the Personal Reader Framework in order to create a rule-based architecture to generate recommendations in a flexible manner based on available user data.

3.1 The Comtella-D System

Comtella Discussions (Comtella-D) [3] is an online community for discussing the social, ethical, legal, and managerial issues associated with information technology and biotechnology. Moreover, it represents a mechanism for motivating participation in interest-based online communities, which engages noncontributing members by modeling and visualizing the asymmetrical relations formed when reading, evaluating, or commenting other community member's contributions. Indeed, it was used to support the coursework related to a 4th year undergraduate class on Ethics and IT taught in the spring of 2006 at the University of Saskatchewan. Access to content is restricted to registered members. Members are relatively anonymous because they are identified just by their alias. The purpose of using Comtella-D in the class was sharing and discussing information (Internet publications, popular magazine, paper, respectively) related to the course topics. The students had to share at least one link to an online paper related to the weekly topic and summarize the paper in a
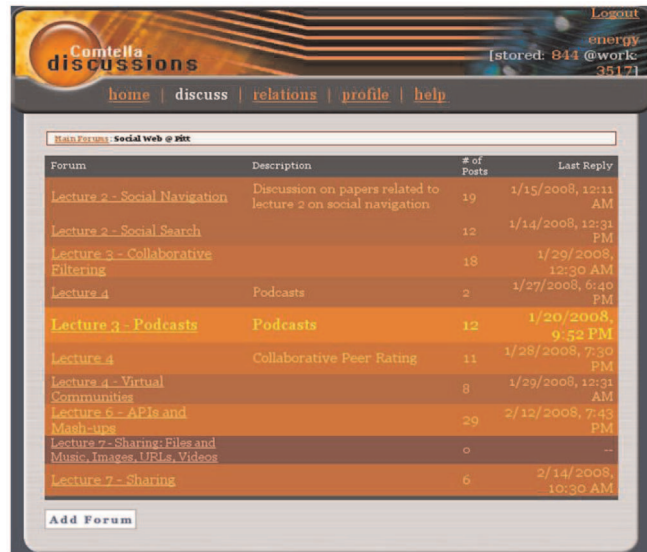

Fig. 1. Screenshot of the Comtella Application—a light color represents actively discussed threads, i.e., energy has been assigned recently to posts within the thread.

way that it stimulates discussion. As a part of their coursework, the students also had to reply/discuss two of their colleagues' postings each week. In parallel with the students of the Ethics and IT class, (4th year Computer Science students), the Comtella-D system was used in a class on Ethics and Technology offered by the Philosophy department. These students used the system only as an additional resource, recommended by the instructor. The system was not related to their coursework, and it was used entirely voluntary.

In the context of Comtella-D, a forum is an initial theme related to a course topic (usually weekly), defined and created by the instructor. A thread is started when a student contributes a link (URL) of a paper related to the topic of the forum. The first post in a new thread contains the URL and a summary of the paper (usually half a page). Further posts in the thread are added as other students respond to/discuss the first post of the thread. Each post can be commented. A comment is usually a very specific local comment to the post rather than to the entire thread. In Comtella-D, comments were used mostly by the marker to give feedback on the quality of arguments raised in the students posts. Fig. 1 presents a thread view in Comtella-D which can be accessed by registered users to follow the discussion. For each thread, the users can view the name of the forum, a description, the number of posts, and the last reply.

In addition, Comtella-D allows students to rate posts by adding or removing energy to or from it. A user can rate every post once, but only if there is free energy in the system available. The system provides a limited number of energy units, depending on the level of activity in the system. Fig. 2 shows two posts with different colors. The post with the lightest color represents the contribution of the user that received the most positive attention from the other users. In other words, the more energy is assigned to a posts the lighter the color of this post becomes. The sum of energy that is available within an online community

Fig. 2. Screenshot of the Comtella Application—users can increase or decrease the energy level of every post by up and down buttons.

measures the current level of contributions/activity in the community. When an item (e.g., discussion post, movie review, blog entry, etc.) is contributed, it brings in a default number of new energy units into the system [3].

Thus, also users who are not willing to contribute new content by posting or commenting, can be engaged: As the energy distribution is done by a simple mouse click and shows an immediate effect (the colors changes), we assume that some of the previously passive users will at least become active in a sense that they distribute energy.

As a result, it is possible to evaluate several aspects, such as the proximity of the users, the altruism, empathy, contributions of each students, and others.

Moreover, the number of energy units in the system increases every time when a new post is created (two new units are added), and it decays with time. In this way, the scarcity of energy in the system prevents users from overrating their colleagues' posts and encourages them to carefully read a post before assigning energy to it. This mechanism is described in [3].

As every week several new threads are started and popular threads attract many posts, keeping an overview of the discussion is a time consuming task. A student who does not spend the time to read all new posts could easily miss important topics of his/her interest. Hence, a recommender system is needed which points the student to relevant posts.

Like in other communities, we determined different behavior styles among the users within the discussion forum:

- *Regularly contributing users:* These users contribute new posts regularly. Often, they discuss about their own opinion with other users.
- *Casual contributing users:* These users contribute only seldomly.
- *Regularly rating users:* These users do not contribute content by creating posts but rate posts of other users regularly.
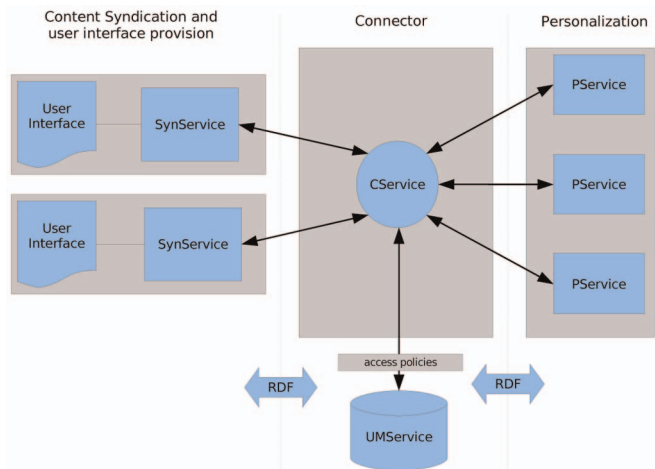

Fig. 3. Architecture of the Personal Reader Framework: SynServices can contact appropriate PService via the CService.

- *Casual rating users:* These users do not contribute content by creating posts and rate posts of other users only seldomly.
- *Passive users:* These users never contribute own posts nor do they rate posts of other users.

These different user types were considered when the recommender system for Comtella-D System was designed. Using a rule-based personalization framework as described in the following section, we can utilize collaborative recommender services to take different user groups into account.

3.2 The Rule-Based Personalization Architecture

We believe that our approach for providing personalization should not focus on one specific application, like Comtella-D. The algorithms behind personalization techniques from the areas of recommender systems [30] or adaptive hypermedia [31] are generic.

In this section, we describe our Semantic Web Service based architecture [32] to enable arbitrary applications to discover and use generic Personalization Services. For our approach, we use the Personal Reader Framework as described briefly in Section 3.2.1. Our generic personalization algorithms are realized as Personal Reader Personalization Services. In Section 3.2.2, we introduce the concept of Data Sources, a semantic description of the Personalization Services and a matchmaking service, which enables the semantic discovery of Personalization Services.

3.2.1 Personal Reader Framework

The Personal Reader Framework [26], [27] (see Fig. 3) enables a fast development of interoperable, personalized Semantic Web applications. Therefore, applications are divided into different parts, which are encapsulated in reusable Web Services. The Framework distinguishes mainly three different kinds of services: 1) Personalization Services, 2) Syndication Services, and 3) a Connector Service.

The Syndication Services (*SynServices*) contain the business logic of an application and maintain the user interface. To receive (personalized) data, SynServices invoke Personalization Services (*PServices*), which allow for personalized
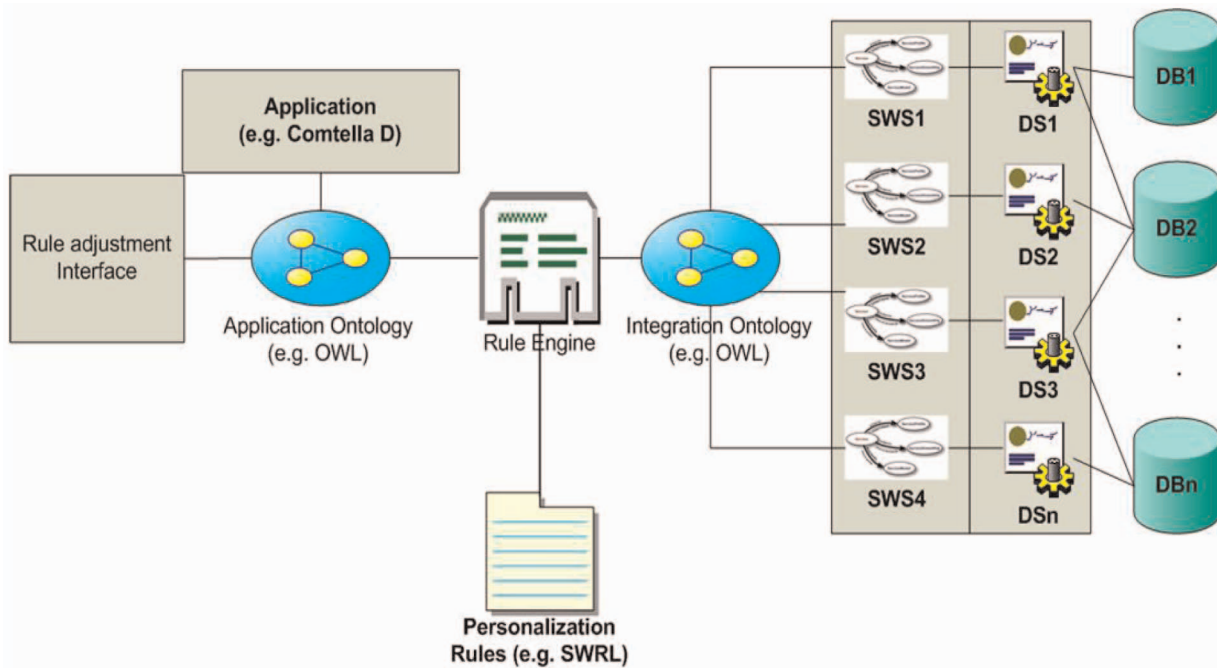
Fig. 4. Architecture of the System—personalization rules map requests from the application, expressed in the Application Ontology vocabulary to the Data Source Services and their Integration Ontology.

access to a specific part of the Semantic Web. Therefore, the SynService passes the application context, e.g., required information, the actual task, etc., to the PService which utilizes the applications context, user data from a centralized repository [33], and the knowledge of a specific domain of the Semantic Web.

The encapsulation of personalization functionality into PServices is the key innovation of the Personal Reader Framework. Applications can therewith be enriched with personalization features in a plug-and-play manner. For example, the Personal Reader Framework provides a service that constructs personalized RSS feeds. The service 1) detects feeds that are relevant to a given query and the user's profile, 2) filters the content of the detected feeds according to the user profile and her context, and 3) aggregates the relevant items into a new feed. The RSS feed personalization service is utilized in different applications. The *MyEar* system [34] applies the service to generate personalized podcasting feeds in order to provide a personal Web radio to the MyEar users while the *MyNews* system [34] integrates the RSS feed personalization service to present personalized news to the users. Both systems can make use of a link annotation service that was primarily developed to annotate links for Personal Readers in the e-learning domain [35].

Another typical Personal Reader setting, which is implemented within the *Personal Publication Reader* [27], looks as follows: *Personalization Service A provides users with recommendations for scientific publications according to the users' interests. Service B offers detailed information about authors or researchers. By integrating both services via a Syndication Service users can browse publications they like within an embedded context.*

A detailed description of the RDF-based messages passed between the different services is available in [36] and [37].

3.2.2 Recommender Architecture

This section describes the architecture of the rule-based personalization framework. The idea is to provide an architecture that ensures the encapsulation and combination of personalized algorithms. For this reason, we decouple personalization algorithms, data sources, and pre- and postprocessing from each other and allow the creation of rules that describe the interaction of the single components. We have chosen a rule-based approach because rules allow for a flexible combination of components that can be adjusted without the need of changing the components. Furthermore, rules have to be commonly used in the e-learning environment [38], [39], hence, e-learning designers can extend existing rules by personalization functionality or—at least—utilize existing knowledge about rules in order to specify new personalization rules.

We chose Semantic Web Services because we foresee dynamic usage of the Personalization Services that provide recommendations. For example, in Comtella-D, the choice of appropriate recommender services might depend on the particular user (regular versus casual contributing, etc.) or on the particular theme, which might attract a certain amount of (contributing) users. Semantic Web Services can help to automatically solve these tasks [40].

Fig. 4 shows the architecture of the rule-based recommender system. It emphasizes two aspects: First, it assures the integration of different recommendation algorithms based on the use of Web Services. Second, it shows how to integrate external personalization functionality like recommendations [32]. A description of the components of the architecture is listed below.

- *DB*: DB represents all databases that can be used for personalization. An important concept is that they are independent from each other but can be
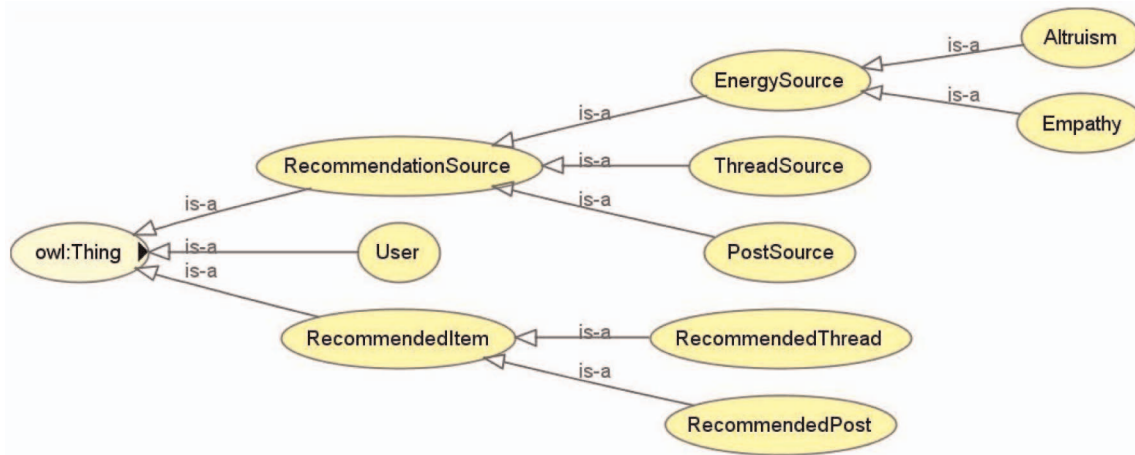
Fig. 5. The Integration Ontology contains concepts to describe the functionality of the Data Sources. It must be fine-grained to distinguish different recommendation services from each other.

combined if necessary. Examples of these databases are user profiles, usage histories or data provided on the Web.

- *DS*: Each data source (DS) represents an encapsulated personalization algorithm implemented as a Personal Reader PService. In other words, these data sources are interfaced by using Web Services. As a consequence of following a Personal Reader-based service-oriented approach, each function is separated into distinct units or services accessible over a network so that users can combine and reuse them when they create new applications [41]. The development of new DS services is easy as the Personal Reader Framework reduces the amount of code that has to be written by the programmer (cf. Section 3.2.1).

- *Semantic Web Services*: Although Web Services separate each data source, which should be combined and reused across the applications, they do not provide an expressive language to describe the services semantically. For this reason, Semantic Web Services (SWS) provide platform functionalities in order to ensure the automatic discovery, composition, and invocation of the services by the system (see [42]). Semantic Web Services provide a number of different recommender algorithms transforming a static collection of information into a distributed information space on the basis of Semantic Web technology, which makes content within the Web machine-processable and machine-interpretable [43].

- *Integration Ontology*: This ontology holds the structure of the users and personalization algorithms to be used by the Semantic Web Services. For this reason, matchmaking algorithms [44], [45], [46] use this ontology to discover, compose, and invoke the Semantic Web Services that are used according to the user specification in the rule-based recommender interface. In addition, this ontology can be extended by the developers without causing any problems to Semantic Web Services, which have been implemented before the ontology was extended. The class hierarchy of this ontology is presented in Fig. 5. The ontology describes three main concepts:

-   *RecommendedItem*: It represents the kind of item considered in the recommendation. In other words, based on the ontology the algorithms can recommend *Posts* or *Threads* in a discussion forum.
-   *User*: Description about the users that receive the recommendation of the algorithms.
-   *RecommendationSource*: This concept defines the kind of source used in the recommendation. For example, the algorithms can take into account the post, threads, or even the *energy* (rating) of a discussion provided by a user (cf. Section 3.1).

- *Application*: It represents applications that can be used by the recommender architecture. In this paper, Comtella-D was used as application.

- *Rule-Adjustment Interface*: This interface is used to specify personalization rules according to the application used. Section 4.7 gives an example of such a rule.

- *OWL Application*: This ontology holds the description of the configuration of the recommendation and the users. The concepts of this ontology are shown in Fig. 6.

- *SWRL (Semantic Web Rule Language)*: SWRL is settled on a combination of OWL DL and the Unary/Binary RuleML sublanguages of the Rule Markup Language [47]. The rules are formed in an antecedent-consequent fashion formed by zero or more atoms. Based on the former ontologies, SWRL rules were created to map instances of one ontology into the other and to ensure the relation between similar concepts with different descriptions (e.g., *person*
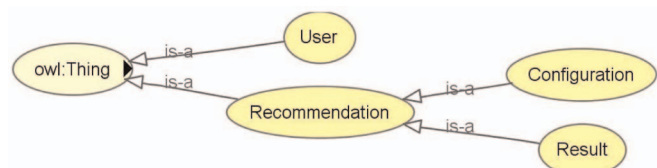

Fig. 6. The Application Ontology contains concepts that are needed to request recommendations for the Data Sources.

concept and *user concept*). It is used to specify the conversion of information between the Comtella-D application and the data sources.

To semantically combine the application with recommender services, we map applications' and services' ontologies to each other in order to enable every component to use its own vocabulary. In the example of the Comtella-D system, the ontology is comparatively small so that a mapping was defined by hand. For larger mappings and the semiautomatic creation of mappings, we recommend to use the SILK framework [48].

In this architecture, rules have three main purposes that enable a flexible coupling of applications and services:

1. Rules define a clear syntactic interface by catching requests from applications and transform them into requests that are submitted to the services.
2. Rules map between applications' and services' ontologies, and hence, ensure integration on the ontology level by maintaining appropriate mappings.
3. Rules use services as bricks for offering complex functionality. Hence, for adjusting the functionality, it is mostly sufficient to modify or adjust the rules instead of changing the services.

3.2.3 Implementation

The technologies used in the development of the generic personalization system were

1. *Protégé*,[1] which is an ontology editor and knowledge-based framework and has been used as the main tool for building OWL ontologies,
2. *SWRL* [47], which represents the Semantic Web rules-language proposal and also combines the rules with the OWL sublanguages. SWRL has been used to improve the expressivity of OWL languages to allow for rules in the domain ontologies and to provide a way for automatic mapping between OWL ontologies. We, furthermore, utilized,
3. *OWL-S Editor*,[2] which is an editor used to build Semantic Web Services. In fact, it provides an easy way to develop *OWL-S* (OWL for Services) and also to map *WSDL* (Web Service Description Language) to OWL-S, and
4. *MindSwap*,[3] which is an OWL-S API that was developed in Java for manipulation of OWL-S ontology-based through objects.

It provides read, write, and execute service ontologies. In addition, it supports execution of services whose groundings are based on WSDL. For the implementation of the different services the Personal Reader Framework was used (cf. Section 3.2.1).

4 EVALUATION

In this section, we use our architecture, as described in Section 3.2.2, to optimize the quality of recommendations in

the e-learning specific discussion forum Comtella-D. We, furthermore, outline that flexible combined collaborative filtering techniques are appropriate for the e-learning domain. The results of our evaluations are used to define a personalization rule that utilizes generic recommender services to generate high-quality recommendations.

To adjust the personalization rule, we used a database snapshot from Comtella-D, which was used complementary to a lecture (see Section 4.1) given in 2006 at the University of Saskatchewan. From the snapshot, we identified representative users in Section 4.2 and extracted relevant research questions to determine the best recommender algorithm based on available input data.

4.1 Data Set

Based on the features of Comtella-D, there are different possibilities on which input data such a collaborative recommender can perform:

1. recommendations based on explicit feedback gained from the user's energy distribution and
2. recommendations based on implicit feedback gained from coposting in the same thread.

For the evaluation, we took a snapshot of the Comtella-D system of the *Ethics and Computer Science* course 2006. Overall, there were 110 registered users. From these users only 36 contributed actively by posting at least one message in the discussion forum. Users rated other users 183 times and posted 756 messages in 173 threads over a time period of approximately three months. In these three months, the lectures dealt every week with a new topic.

4.2 Scenario

Assume three users A, B, and C: A is a very active user, she create new posts regularly and rates posts of other users as well. B is a user who was some weeks ago active, from then on did not use the system and now requests recommendations from the system. C has used the system rarely and has contributed only two posts.

To define a personalization rule which recommends threads, we need to find a rule that takes all the different behavior patterns into account. We need to know, for user A if all information that we have, shall be taken into account when recommendations are generated. Can we still use the possibly outdated information from user B and is C's contribution sufficient to generate recommendations?

From this scenario, we derived the following four research questions that we would like to answer in this section:

1. How much training data is required to generate precise recommendations (Section 4.3)?
2. What kind of input data (explicit or implicit) gives the best quality to recommend threads (Section 4.4)?
3. Does the behavior of users in the discussion forum change over time (Section 4.5)?
4. Are active users, i.e., users who have posted frequently and hence are more experienced, more reliable as source for recommendations (Section 4.6)?.

	T1	T2	T3	T4	T5	T6	T7	T8
Training User 1	X		X	X		X		
Training User 2		X	X		X		X	
Test User	X			X		**X**		

Fig. 7. Division of the data set into training data (week 1, containing threads T1-T4) and test data (week 2, containing threads T5-T8).

In particular, questions 1, 3, and 4 are of special interest within an e-learning tool: As e-learning environments, like Comtella-D, are often used as a supplement for a given university course, the number of participants is small compared to other domains, where collaborative filtering techniques are used (e.g., online stores). Hence, also the available amount of input data is very limited. As the learner extends the own knowledge quickly, we assume that a learner will also change her opinion with additional information frequently. Thus, old opinions and interests cannot be used to predict current interests. Regarding question 4, we search for domain experts and assume that these experts can be found within the most active learners.

For all of the following measurements, we used the recommender library RenkGround,[4] which implemented the collaborative recommender algorithm described in [49], [50].

4.3 Required Amount of Training Data

To determine how much training data is required to generate precise recommendations (first question), we divided our data set into weeks corresponding to the different topics of the lectures. Afterwards, we iterated over the weeks, selecting every week x as training set. Then, a test user was selected for whom we tried to calculate the posts that this user will create in week $x + 1$.

For example in Fig. 7, week 1 (containing threads T1-T4) is used as training set to find similar users to the test user. Afterwards, all posts from week 2 (containing threads T5-T8), which is considered as test data, are removed from the test user (bold cross). Finally, post recommendations for week 2 are generated from the posting behavior of the similar users in week 2. A hit is achieved if the recommendations contain the original post (bold cross) of the test user.

To ensure that users have contributed sufficient input to generate appropriate recommendations, we classified the users into different classes. These classes contain sets of users who have posted at least y posts in different threads in the training period and at least one post in the test time.

To compare our results, we used a nonpersonalized baseline algorithm. Therefore, we recommend the top-k threads containing the most posts. This baseline algorithm seems fair as the overall data set is comparatively small and top-k lists can thus generate good recommendations for the users.

Our research hypothesis is that the more data from a user is available in the training set, the more precise the recommendation for the test set are.
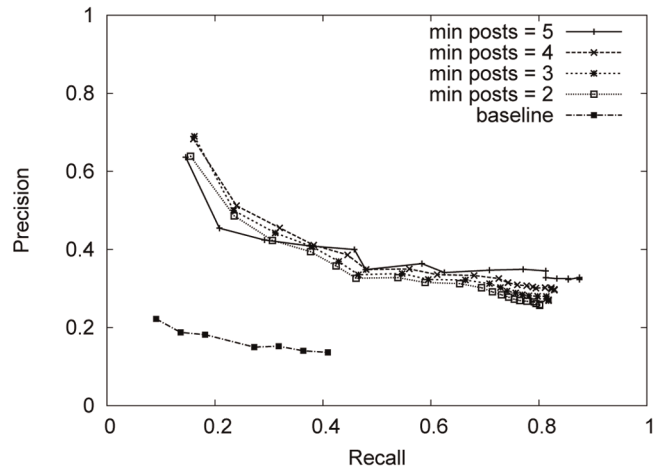
Fig. 8. The precision-recall diagram based on implicit user feedback for users who have posted at least 2, 3, 4, or 5 times in the training set week.

The precision-recall distribution is built by iterating over all users in the class and calculating the top-k recommendations for these users. k is chosen from 1 to the number of all posts. For every k, the precision and recall is calculated as the average mean of all precision and recall values of all users in the class. Therefore, the recommendation system is invoked as follows: First, the posts generated in the training set are passed to the recommender system to determine the similarity between the users. Afterwards, the recommendations are calculated by passing all posts to the recommender system, which were created in the test set.

Fig. 8 displays the precision-recall distribution for the nonpersonalized baseline algorithm and the personalized recommendations based on users who have contributed at least 2, 3, 4, or 5 posts in the training set. While for $k \leq 3$, the classes 3 to 5 perform better than class 2, class 2 performs better for $k > 4$. However, none of the different classes results in significantly better results than the other classes. Furthermore, all approaches are able to retrieve not more than 80 percent of the threads the users have contributed to. This can be explained by the characteristics of the recommendation process: When a thread is recommended, a user who is similar to the current user must have contributed to this thread. Hence, threads, which are discussed by a limited number of users, are recommended rarely. This issue is known as *new item* problem in collaborative recommender systems [19].

Overall, the results imply that 1) the nonpersonalized baseline algorithm is outperformed by the personalized algorithm, and that 2) two posts in a week are sufficient to generate precise personalized recommendations while more posts do not improve the results significantly.

4.4 Implicit versus Explicit User Feedback

In the second step, we tried to deduce what kind of input data (explicit or implicit) gains the best quality regarding the recommendation of threads (second question). By explicit data we mean energy assignments, whereas implicit data is based on the posting behavior of a user. Based on the
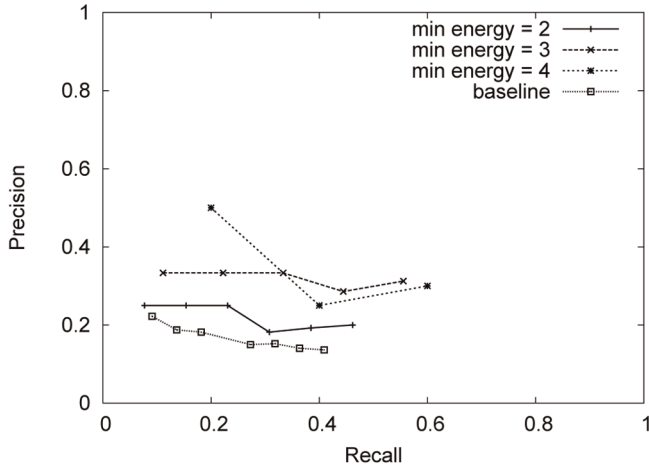
Fig. 9. The precision-recall diagram based on explicit user feedback for users who have rated at least 2, 3, or 4 posts of other users in the training set week.

Fig. 10. The precision-recall diagram shows prediction quality for $x + 1, x + 2, \ldots, x + 7$ weeks ahead generated based on the training data of week x.

classes defined in the previous section that used implicit user feedback by engaging the posts a user created, we define equivalent classes of explicit user feedback. These classes contain users who have at least added or removed x energy points to posts from other users in the training set week and have at least posted once in the test set week.

To recommend posts by utilizing user ratings, we modified the similarity function of the recommender system. Instead of comparing the similarity of user vectors containing threads a user has posted in, we use vectors containing the energy distribution. Two users are considered as similar when they both assigned energy to the same post, hence expressing interest in the same post. We did not take into account if users added or removed energy as we interpreted every form of energy assignment as interest in a post. The recommender algorithm itself was not modified.

Fig. 9 gives an overview of the precision-recall ratio of recommendations based on explicit feedback for the classes of users having rated at least 2, or 3 other users in the training set period. The class with five energy assignments was omitted as it did not contain enough users to deliver reliable results. The graph outlines that—like in the previous section—a comparable small amount of input data, namely two energy assignments, are sufficient to create appropriate recommendations and that increasing the amount of input data does not increase the precision or recall of the recommendations significantly. Compared to the precision-recall distribution generated by implicit user feedback, the quality of the results generated by explicit feedback, in respect of both, precision and recall, are lower.

We also tried to combine explicit feedback and implicit feedback as we expected that input from different sources could improve the overall performance. We used the average mean to combine the weighted result sets of the recommendations based on explicit feedback and implicit feedback. We examined that the more we increased the weight of the explicit user feedback, the worse our recommender system performed.

Our conclusion for the given setting is that explicit feedback (energy assignments) performs always worse than
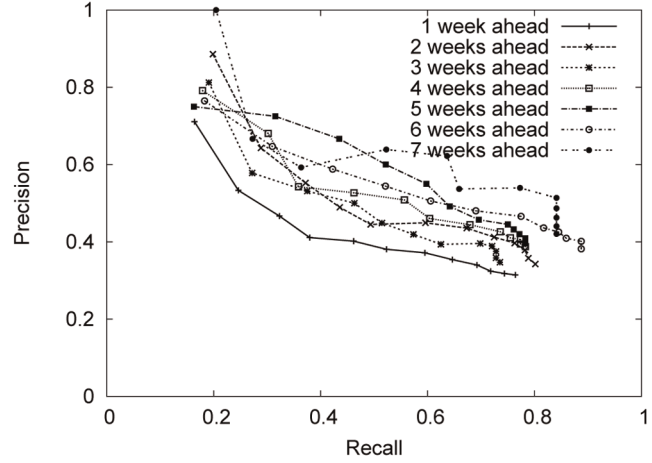
implicit feedback (posting behavior) and cannot be used to improve recommendation based on implicit feedback. However, if no implicit feedback is given for a specific user, explicit feedback performs better than the nonpersonalized baseline algorithm. Hence, explicit feedback based recommendations can be used as a fallback if no implicit feedback is available.

Based on these results, we used implicit user feedback as source for the recommendations applied in the following evaluations.

4.5 User Behavior

The Comtella-D system was strongly coupled with the timeline of the lectures. This means that the users discussed every week a new topic. We assume that the behavior of users changes over time (and over different topics) which means that the more weeks ahead recommendations are created, the more imprecise they are. Furthermore, as topics discussed in a given week should still be somewhat fresher in the memory of the students, we assume that the forecast for the next week would be more precise than forecasts for two or more weeks ahead.

To verify our assumptions, we iterated over all weeks and used them as training data. We calculate the recommendations for n weeks ahead, where $n = 1, 2, \ldots, 7$ and compared them with the test data. Afterwards, we created the precision-recall diagram displayed in Fig. 10.

The figure displays a result which does not comply with our assumptions: The one week ahead precision-recall values for small top-k result sets are worse than all other forecasts. Furthermore, the forecasts for more weeks ahead do not comply to any rule or trend. This means that the behavior of the users indeed changes over time and topic (third question), but that the change of behavior is not monotonic and cannot be forecasted. However, we have to remark that our data sets covers only three months of data. Thus, we can only infer about the short time behavior of users but cannot conclude that there is not a long time trend in user behavior.
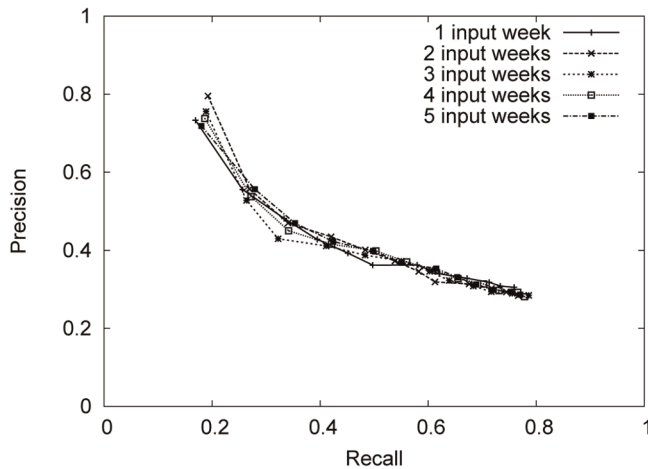
Fig. 11. The precision-recall diagram shows prediction quality of one week ahead recommendations based on the previous one to five weeks of training data.

4.6   Size of the Time Frame

In the previous section, we have shown that the user behavior changes over the weeks making a constantly high forecast for several weeks ahead impossible. To lower this effect, we increase the input data timeframe by aggregating several weeks as training set and creating recommendations for one week ahead. We expect that aggregating several weeks of input data normalizes the behavior of a user on the one hand and increases the amount of input data one the other. Both effects should result in an increased quality of the recommendations. Fig. 11 displays the measurement aggregating one to five weeks of input data and calculating the precision and recall of the recommendations for the following week.

All input periods result in similar results. Our expectation that more input weeks could improve the result could not be proven. This also underlines our previous observation that the changes of quality regarding precision and recall seem to follow no rule or trend. Hence, we can also answer our fourth question: Active users, i.e., users who have posted frequently in Comtella-D, are not more reliable as source for recommendations than users who posted less frequently.

4.7   Results

The results proof that a small amount of input data (two posts or two energy assignments) and a small number of users—which is a typical scenario within an e-learning application like Comtella-D—is enough to generate precise recommendations. Hence, collaborative recommender algorithms are appropriate to be used in the e-learning domain.

Furthermore, we have shown that the implicit user feedback, given by the posting behavior of users gives much better recommendations than explicit user feedback given by the energy assignment of the users while the user behavior tends to be stable over weeks. A further experiment has shown that more input data does not always generate better recommendations. Thus, a flexible method to combine different recommender algorithms based on the available input data is required.

According to these observations, an optimal personalization rule to recommend threads in the Comtella-D system is the following:

```
if at least two posts of the user exist:
then create recommendation based
     on implicit feedback
else
  if at least two energy
     assignments of the user exist:
  then create recommendation based
     on explicit feedback
  else use the nonpersonalized
     baseline algorithm
```

By enhancing already existing rules or adding this personalization rule to the e-learning environment, e-learning systems can easily recommend relevant information/discussions to a learner.

5   CONCLUSION AND FUTURE WORK

This paper identified the advantages of using a discussion forum within an e-learning system to foster communication between learners. We motivated the need for recommendations in discussion fora to maintain an efficient learning experience within large or weak structured discussion boards.

We identified the need to implement recommendation functionality into various discussion boards. Hence, we proposed a general-purpose Semantic Web Service based framework to encapsulate generic personalization algorithms. Therefore, applications shall define personalization rules to select, access, and optionally combine different personalization algorithms.

As hand on, we evaluated how to use different recommendation strategies to generate recommendations within the e-learning focused discussion forum Comtella-D. Our results show that for different user groups of the discussion forum, different recommendation strategies return most appropriate recommendations.

In the future, we plan to make the personalization rule user-adjustable. Therefore, we will introduce user-adjustable weights in the rule which enable the combination of different techniques according to a user's preferences. Thus, users do not have to cope with the (formal) rule itself, but can use an easy-to-use user interface.

Additionally, as fora contain text messages, namely the posts, we will try to use content-based recommendations to improve the quality of recommendations. By analyzing the content of the posts it is also possible to derive specific interests of a user. These interests can be stored in an RDF-based user profile [51] and improve other applications that generate recommendations for the corresponding user.

In systems, where the number of user groups, personalization algorithms, or different kind of input data become too large to create personalization rules be hand, data mining tools like Weka[5] can be used to automatically identify the most appropriate strategies to personalize content according to a user's input data.

_____

5. http://www.cs.waikato.ac.nz/ml/weka/.

REFERENCES

[1] F. Bodendorf, "Controlling Communication Processes in E-Learning Scenarios," *Proc. Conf. Web-Based Education,* 2009.

[2] A. Sieg, B. Mobasher, and R. Burke, "Inferring Users Information Context: Integrating User Profiles and Concept Hierarchies," *Proc. Meeting of the Int'l Federation of Classification Societies,* 2004.

[3] A. Webster and J. Vassileva, "Visualizing Personal Relations in Online Communities," *Adaptive Hypermedia and Adaptive Web-Based Systems,* V.P. Wade, H. Ashman, and B. Smyth, eds., vol. 4018, pp. 223-233, Springer, 2006.

[4] S.U. Kelly, C. Sung, and S. Farnham, "Designing for Improved Social Responsibility, User Participation and Content in On-line Communities," *Proc. SIGCHI Conf. Human Factors in Computing Systems (CHI '02),* pp. 391-398, 2002.

[5] R.A. Schwier and S. Balbar, "The Interplay of Content and Community in Synchronous and Asynchronous Communication: Virtual Communication in a Graduate Seminar," *Canadian J. Learning and Technology,* vol. 28, no. 2, pp. 21-30, 2002.

[6] D. Helic, H. Maurer, and N. Scerbakov, "Discussion Forums as Learning Resources in Web-Based Education," *Advanced Technology for Learning,* vol. 1, no. 1, pp. 8-15, 2004.

[7] M. Thomas, "Learning Within Incoherent Structures: The Space of Online Discussion Forums," *J. Computer Assisted Learning,* vol. 18, pp. 351-366, 2002.

[8] E. Webb, A. Jones, P. Barker, and P. van Schaik, "Using E-Learning Dialogues in Higher Education," *Innovations in Education and Teaching Int'l,* vol. 41, no. 1, pp. 93-103, 2004.

[9] J. Bradshaw and L. Hinton, "Benefits of an Online Discussion List in a Traditional Distance Education Course," *Turkish Online J. Distance Education,* vol. 5, no. 3, 2004.

[10] K. Harman and A. Koohang, "Discussion Board: A Learning Object," *Interdisciplinary J. Knowledge and Learning Objects,* vol. 1, pp. 67-77, 2005.

[11] H. Drachsler, H. Hummel, B.V. den Berg, J. Eshuis, A. Berlanga, R. Nadolski, W. Waterink, N. Boers, and R. Koper, "Recommendation Strategies for E-Learning: Preliminary Effects of a Personal Recommender System for Lifelong Learners," technical report, Open Univ. of the Netherlands, 2007.

[12] M.A. Otair and A.Q.A. Hamad, "Expert Personalized E-Learning Recommender System," *Proc. Int'l Conf. e-Business and e-Learning,* 2005.

[13] O. Zaiane, "Building a Recommender Agent for e-Learning Systems," *Proc. Int'l Conf. Computers in Education,* pp. 55-56, 2002.

[14] T.Y. Tang and G. McCalla, "Smart Recommendation for an Evolving E-Learning System," *Proc. Workshop Technologies for Electronic Documents for Supporting Learning, Int'l Conf. Artificial Intelligence in Education (AIED '03),* 2003.

[15] N. Soontornphisaj, E. Rojsattarat, and S. Yim-ngam, "Smart E-Learning Using Recommender System," *Proc. Int'l Conf. Intelligent Computing,* 2006.

[16] J.A. Konstan, B.N. Miller, D. Maltz, J.L. Herlocker, L.R. Gordon, J. Riedl, and H. Volume, "GroupLens: Applying Collaborative Filtering to Usenet News," *Comm. ACM,* vol. 40, pp. 77-87, 1997.

[17] N.S. Glance, M. Hurst, and T. Tomokiyo, "BlogPulse: Automated Trend Discovery for Weblogs," *Proc. WWW Workshop Weblogging Ecosystem,* http://www.blogpulse.com/papers/www2004glance.pdf, May 2004.

[18] A.B. Brush, D. Bargeron, J. Grudin, A. Borning, and A. Gupta, "Supporting Interaction Outside of Class: Anchored Discussions Versus. Discussion Boards," *Proc. ACM Conf. Human Factors in Computing Systems (CHI '02),* 2002.

[19] R. Burke, "Hybrid Recommender Systems: Survey and Experiments," *User Modeling and User-Adapted Interaction,* vol. 12, no. 4, pp. 331-370, 2002.

[20] M. Khribi, M. Jemni, and O. Nasraoui, "Toward a Hybrid Recommender System for E-Learning Personalization Based on Web Usage Mining Techniques and Information Retrieval," *Proc. World Conf. E-Learning in Corporate, Govt., Healthcare, and Higher Education,* G. Richards, ed., pp. 6136-6145, 2007.

[21] M. Lee, P. Choi, and Y. Woo, "A Hybrid Recommender System Combining Collaborative Filtering with Neural Network," *Proc. Second Int'l Conf. Adaptive Hypermedia and Adaptive Web-Based Systems (AH '02),* pp. 531-534, 2002.

[22] N. Taghipour and A. Kardan, "A Hybrid Web Recommender System Based on Q-Learning," *Proc. ACM Symp. Applied Computing (SAC '08),* pp. 1164-1168, 2008.

[23] S. Puntheeranurak and H. Tsuji, "A Multi-Clustering Hybrid Recommender System," *Proc. Seventh IEEE Int'l Conf. Computer and Information Technology (CIT '07),* pp. 223-228, 2007.

[24] T. Tran and R. Cohen, "Hybrid Recommender Systems for Electronic Commerce," *Proc. Knowledge-Based Electronic Markets, Papers from the AAAI Workshop, AAAI Technical Report WS-00-04,* pp. 78-83, 2000.

[25] P. Brusilovsky and N. Henze, "Open Corpus Adaptive Educational Hypermedia," *The Adaptive Web,* P. Brusilovsky, A. Kobsa, and W. Nejdl, eds., pp. 671-696, Springer, http://dblp.uni-trier.de/db/conf/adaptive/adaptive2007.html#BrusilovskyH07, vol. 4321, 2007.

[26] N. Henze and D. Krause, "Personalized Access to Web Services in the Semantic Web," *Third Int'l Semantic Web User Interaction Workshop (SWUI '06),* Nov. 2006.

[27] F. Abel, R. Baumgartner, A. Brooks, C. Enzi, G. Gottlob, N. Henze, M. Herzog, M. Kriesell, W. Nejdl, and K. Tomaschewski, "The Personal Publication Reader, Semantic Web Challenge 2005," *Proc. Fourth Int'l Semantic Web Conf.,* Nov. 2005.

[28] P. Ralph and J. Parsons, "A Framework for Automatic Online Personalization," *Proc. 39th Ann. Hawaii Int'l Conf. System Sciences (HICSS '06),* p. 137b, 2006.

[29] F. Zhang and H.-Y. Chang, "On A Hybrid Rule Based Recommender System," *Proc. Fifth Int'l Conf. Computer and Information Technology (CIT '05),* pp. 194-198, 2005.

[30] G. Adomavicius and A. Tuzhilin, "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions," *IEEE Trans. Knowledge and Data Eng.,* vol. 17, no. 6, pp. 734-749, June 2005.

[31] P. Brusilovsky, "Adaptive Hypermedia," *User Modeling and User-Adapted Interaction,* vol. 11, pp. 87-110, 2001.

[32] F. Abel, I.I. Bittencourt, N. Henze, D. Krause, and J. Vassileva, "A Rule-Based Recommender System for Online Discussion Forums," *Proc. Fifth Int'l Conf. Adaptive Hypermedia and Adaptive Web-Based Systems,* pp. 12-21, 2008.

[33] F. Abel, N. Henze, D. Krause, and D. Plappert, "User Modeling and User Profile Exchange for Semantic Web Applications," *Proc. 16th Workshop Adaptivity and User Modeling in Interactive Systems (ABIS '08),* 2008.

[34] F. Abel, I. Brunkhorst, N. Henze, D. Krause, K. Mushtaq, P. Nasirifard, and K. Tomaschewski, "Personal Reader Agent: Personalized Access to Configurable Web Services." *Proc. 14th Workshop Adaptivity and User Modeling in Interactive Systems (ABIS '06),* pp. 12-13, 2006.

[35] N. Henze, "Personal Readers: Personalized Learning Object Readers for the Semantic Web," *Proc. 12th Int'l Conf. Artificial Intelligence in Education,* http://rewerse.net/publications/rewerse-publications.html#REWERSE-RP-2005-28, 2005.

[36] N. Henze and D. Krause, "User Profiling and Privacy Protection for a Web Service Oriented Semantic Web," *Proc. 14th Workshop Adaptivity and User Modeling in Interactive Systems,* pp. 42-46, 2006.

[37] F. Abel, I. Brunkhorst, N. Henze, D. Krause, K. Mushtaq, P. Nasirifard, and K. Tomaschweski, "Personal Reader Agent: Personalized Access to Configurable Web Services," technical report, L3S Research Center, July 2006.

[38] P. Dolog, N. Henze, W. Nejdl, and M. Sintek, "Personalization in Distributed E-Learning Environments," *Proc. 13th Int'l World Wide Web Conf. Alternate Track Papers & Posters (WWW Alt. '04),* pp. 170-179, 2004.

[39] S. Odeh and E. Ketaneh, "Collaborative Working E-Learning Environments Supported by Rule-Based E-Tutor," *Int'l J. Online Eng.,* vol. 3, no. 4, 2007.

[40] I. Bittencourt, E. Costa, E. Soares, and A. Pedro, "Towards a New Generation of Web-Based Educational Systems: The Convergence Between Artificial and Human Agents," *IEEE Multidisciplinary Eng. Education Magazine,* vol. 3, no. 1, pp. 17-24, 2008.

[41] T. Erl, *Service-Oriented Architecture: Concepts, Technology, and Design.* Prentice Hall PTR, 2005.

[42] I.I. Bittencourt, E. Costa, M. Silva, and E. Soares, "A Computational Model for Developing Semantic Web-Based Educational Systems," *Knowledge Based System J.,* special issue on AI and blended learning, vol. 22, no. 4, pp. 302-315, May 2009,

[43] I.I. Bittencourt, S. Isotani, E.B. Costa, and R. Mizoguchi, "Research Directions on Semantic Web and Education," *Proc. First Brazilian Workshop Semantic Web and Education, Brazilian Symp. Informatics and Education,* pp. 71-82, 2007.

[44] M. Klusch, B. Fries, and K. Sycara, "Automated Semantic Web Service Discovery with OWLS-MX," *Proc. Fifth Int'l Joint Conf. Autonomous Agents and Multiagent Systems (AAMAS '06),* http://portal.acm.org/citation.cfm?id=1160796, pp. 915-922, 2006.

[45] OWL-S/UDDIM, http://projects.semwebcentral.org/projects/owl-s-uddi-mm, 2005.

[46] I. Calado, H. Barros, I.I. Bittencourt, E. Costa, E. Soares, M. Silva, M. Arruda, and T. Toledo, "An Approach for Semantic Web Services Automatic Discovery and Composition with Similarity Metrics," *Proc. ACM Symp. Applied Computing (SAC) in Agent-Oriented Software Eng. Methodologies and Systems Track,* pp. 2210-2212, 2009.

[47] I. Horrocks, P.F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean, "SWRL: A Semantic Web Rule Language Combining OWL and RuleML," DARPA DAML Program, Draft Version, http://www.daml.org/rules/proposal, Dec. 2004.

[48] C. Bizer, J. Volz, G. Kobilarov, and M. Gaedke, "Silk—A Link Discovery Framework for the Web of Data," *Proc. World Wide Web Conf.,* 2009.

[49] U. Shardanand and P. Maes, "Social Information Filtering: Algorithms for Automating 'Word of Mouth,'" *Proc. ACM Conf. Human Factors in Computing Systems (CHI '95),* vol. 1, citeseer.ist.psu.edu/shardanand95social.html, pp. 210-217, 1995.

[50] J.L. Herlocker, J.A. Konstan, A. Borchers, and J. Riedl, "An Algorithmic Framework for Performing Collaborative Filtering," *Proc. 22nd Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* http://www.acm.org/pubs/articles/proceedings/ir/312624/p230-herlocker/p230-herlocker.pdf, pp. 230-237, 1999.

[51] F. Abel, N. Henze, D. Krause, and D. Plapper, "User Modeling and User Profile Exchange for Semantic Web Applications," *Proc. 16th Workshop Adaptivity and User Modeling in Interactive Systems,* 2008.

Fabian Abel started working toward the PhD degree in computer science in March 2007 at the L3S Research Center (L3S). Among other activities, he participated in the development of the Personal Publication Reader which was awarded at the International Semantic Web Challenge 2005. In the context of the PhD, he is concerned with social semantic Web and investigates problems regarding ontology learning, search, user modeling, and personalization techniques.

Ig Ibert Bittencourt received the master's degree in computational modeling of knowledge from the Federal University of Alagoas and the PhD degree in computer science from Federal University of Campina Grande, in 2006 and 2009, respectively. He is a professor at the Federal University of Alagoas, Brazil, and a member of the Brazilian Computer Society. His research interests include Semantic Web and education, social Semantic Web, and adaptation and personalization. He has been a program committee member of international workshops, cochaired the Brazilian workshops on Semantic Web and Education, and presented several talks in the context of the Semantic Web and Education.

Evandro Costa received the PhD degree in electronic engineering from the Federal University of Campina Grande, Paraba, Brazil. He is currently a professor in the Computer Science Institute at the Federal University of Alagoas, Brazil and director of the same institute. He is also a collaborator professor in the graduate program at the Federal University of Campina Grande, Paraiba, Brazil. His research interests include multiagent systems, knowledge representation and Semantic Web, and artificial intelligence in education, where he has published more than 100 peer-reviewed papers. He teaches several issues, such as artificial intelligence, distributed artificial intelligence, and data mining.

Nicola Henze received the master's degree in mathematics in 1995 and the PhD degree in computer science in 2000 from the Leibniz University Hannover. She is a professor of computer science at the Leibniz Universität Hannover, Germany and an associated member of the L3S Research Center. She currently holds a junior professorship for Semantic Web at Leibniz Universität Hannover. Her research interests include user modeling and adaptation reasoning and personalization in the Semantic Web, and Web engineering. She has been a program committee member of various international conferences, has cochaired numerous international workshops, has been an invited speaker at international conferences, and has published more than 60 papers in international journals and conferences.

Daniel Krause is working toward the PhD degree at the L3S Research Center after he received the master's degree in computer science in 2006 from the Leibniz University of Hannover. In the Personal Reader project, Daniel focuses on rule-based approaches to create generic personalization components and a framework which enables the efficient creation of Semantic Web applications. Since 2007, he has been a steering committee member of the SIG ABIS of the German Computer Science Society.

Julita Vassileva is a professor of computer science at the University of Saskatchewan, Canada, where she currently teaches a popular class on social computing. Her interests are very broad and fall in the general areas of user modeling and user-adaptive systems, intelligent learning environments, multiagent systems, and social computing. She has developed a domain-independent approach instructional planning (or sequencing) for intelligent tutoring systems and learning; practical user modeling approaches in task-based information retrieval, new approaches for bilateral negotiation in multiagent systems, trust and reputation mechanisms in multiagent/service-oriented domains, as well as incentive mechanisms, and various social visualization designs to encourage participation in online communities.