Chapter 7
Evaluating the Effectiveness of Personalised Recommender Systems in Learning Networks

Hendrik Drachsler, Hans Hummel, Bert van den Berg, Jannes Eshuis, Wim Waterink, Rob Nadolski, Adriana Berlanga, Nanda Boers, and Rob Koper

7.1 Introduction

In view of the professional development concept, learning can no longer be considered to be part of childhood and youth alone, but is becoming a lifelong achievement. Professional development no longer remains limited to the context of a regular school or university campus, but is becoming integrated into workplace learning and personal development, where formal and informal learning activities are intertwined. Professionals find themselves placed at centre-stage, which means that no longer a teacher or teaching institute is responsible for the learning process but that they now are responsible for their own learning processes (Longworth 2003; Shuell 1992). Taking up on this responsibility, professionals need to become self-directed (Brockett and Hiemstra 1991), and might be performing different learning activities in different contexts at the same time. On the one hand learners are becoming free to decide what, when, where and how they want to learn, and on the other hand they are forced to be responsible for their own professional competence development.

H. Drachsler (✉)
Centre for Learning Sciences and Technologies, Open University of the Netherlands,
Valkenburgerweg 177, 6419 AT Heerlen, The Netherlands
e-mail: hendrik.drachsler@ou.nl

R. Koper (ed.), *Learning Network Services for Professional Development*,
DOI 10.1007/978-3-642-00978-5_7,

In this chapter we will describe the decisions developers have to make if they want to set up an experimental study to evaluate the effects of recommender systems for Learning Networks. Common tools for these kinds of feedback are recommender systems that support users in finding their way through the possibilities on the WWW. Many online companies like *amazon.com, netflix.com, drugstore.com,* or *ebay.com*(Linden et al. 2003; Schafer et al. 1999) are using a recommender system to direct the attention of their costumers to other products in their collection. The general purpose of recommender systems is to pre-select information a user might be interested in Adomavicius and Tuzhilin (2005).

Recommender systems can exist of various combinations of differently designed algorithms. A good overview about current recommender system technologies can be found in Burke (2002) and Herlocker et al. (2000). A detailed overview about especially recommender systems in the learning domain can be found (Drachsler et al. 2009b Nadolski et al. 2009).

This chapter offers guidelines to set up experiments for the evaluation of recommender systems in Learning Networks. It is based on our experiences with the ISIS experimentation that we conducted together with the Psychology department of the Open University of the Netherlands. The experiment focused on supporting learners in their course selection by providing personalised recommendations. In this chapter we focus on the methodology and technical decision that have to be taken; detailed experimental results and further information of the ISIS experiment can be found in Drachsler et al. (2009b).

In order to design an experiment for recommender systems in Learning Networks several things have to be considered. Firstly, the experimental designers have to be aware of the underlying concept of professional development that inspires the whole experiment. Secondly, they have to be aware of the Learning Networks concepts. Thirdly, the designers are expected to have at least basic knowledge about recommender system technologies. Finally, standardised methods of experimental design are required in order to run a valid experiment. In the following section of the chapter we shortly introduce the reader to those requirements.

First, for a proper experimental design researchers have to decide which hypotheses should be tested and which variables support those hypotheses. Most of the times in Technology Enhanced Learning (TEL) research, we want to observe if learners perform more efficient, more effective, are more satisfied, or if the instrument used decreased the drop-out rate of learners. In the special case of Learning Networks we also have to consider aspects form Social Networks Analysis in order to analysis how the network benefits from the contributions of their learners.

Second, running real life experiments with recommender systems requires a specific kind of statistic analysis. This analysis is based on measurements on a regular basis over a fixed time period. It enables the researches to monitor the effects of their instrument (recommender system) during the runtime and at the end of the experimentation.

Third, experimental designers have to make a decision which techniques should be used to present the learning activities to the participants of the experiment. Most of the time, a common virtual learning environment (VLE) will be selected. There are many Open Source solutions available like *Drupal* or *Moodle* to set up a Learning Network. The experiment can also rely on an alternative in-house solution that is already successfully applied in an institution. Especially for recommender systems, researchers have to make a decision if they want to build their own recommender system or apply already existing recommender system plug-ins or frameworks in a VLE.

In the second section of this chapter, we will now describe an experimental design for a recommender system in a Learning Network. Section three explains details about the statistical analysis of this ISIS experiment. Section four will discuss the selection of suitable techniques. Finally, the last section offers ideas for future research regarding recommender systems in Learning Networks.

7.2 Experimental Design

In the recommender system research, most of the time offline experiments are done with several data sets with specific characteristics (the MovieLens dataset, the Book-Crossing data sets, or the EachMovie dataset) before preparing an experiment with real users (Goldberg et al. 2001; O'Sullivan et al. 2002; Sarwar et al. 2002). This is also because classic recommender system research has its focus on the optimisation or invention of more efficient algorithms for certain recommendation problems. These data sets are used as a common standard or benchmark to evaluate new kinds of recommendation algorithms. Furthermore, machine-learning research only evaluates recommendation algorithms based on common technical measures like accuracy, coverage, and performance in terms of execution time (Adomavicius and Tuzhilin 2005; Burke 2002; Herlocker et al. 2004). Accuracy measures how close the predicted ranking of items for a user differs from the user's true ranking of preference. Coverage measures the percentage of items for which a recommender system is capable of making predictions. Performance observes if a recommender system is able to provide a recommendation in a reasonable time frame.

Research on recommender systems in Learning Networks is also in need of these technical measures, but in the first place we have to improve the learning process with the selected technology. We have to deal with information about learners and learning activities and combine different levels of complexity for the different learning situations the learner may be involved in. The main recommendation goal for recommender system in Learning Networks is to provide learners with suitable learning activities in order to support their professional competence development. Therefore, recommender systems in Learning Networks have to consider relevant pedagogical rules describing pedagogy-oriented relations between learners' characteristics and LA-characteristics. For example: from Vygotsky's 'zone of proximal development' follows the pedagogical rule 'recommended learning activities should have a level a little bit above learners' current competence level' (Vygotsky 1978). Thus, recommender systems in Learning Networks have to take into account competence levels in order to suggest an appropriate learning activity. Further differences between recommendation in the e-commerce domain and the learning domain can be found in Drachsler et al. (2009a).

Currently, we do not have any standardised data sets for offline experiments publicly available. Further, it is not appropriate to focus only on technical measures for recommender systems in Learning Networks without considering the actual needs and characteristics of professionals. Thus, further evaluation procedures that are complementary to technical evaluation approaches are needed.

In the following we split this section into two subsections. The first subsection (Sect. 7.2.1) explains general requirements to evaluate recommender system in Learning Networks. The second subsection (Sect. 7.2.2) describes the experimental setup of the ISIS experiment in detail.

7.2.1 An Evaluation Framework for Recommender Systems in Learning Networks

A pedagogy driven recommender system for Learning Networks that takes into account learner characteristics and specific learning demands also should be evaluated by multiple evaluation criteria. To evaluate the influence of the recommender system we need a mixture of educational, technical and network measures. We advise you to mix technical evaluation criteria with educational research measures and network measures (Drachsler et al. 2009a) in a recommendation framework. Therefore, we suggest the following for the analysis of the suitability of recommender system in Learning Networks.

Classic educational research is most of the time evaluated base on the outcomes of the learning process of the learner (Thorpe 1988). The aim is to develop the competences of the learner on cognitive or motor level. Therefore, commonly used measures for valid evaluations are effectiveness, efficiency, satisfaction, and the drop-out rate because of two reasons. First, these criteria are used to evaluate for instance universities regarding their outcomes, and second they can be efficiently operationalised. For example, effectiveness is a measure of the total amount of completed, visited, or studied learning activities during a learning phase. Efficiency indicates the time that learners needed to reach their learning goal. It is related to the effectiveness variable through counting the actually study time. Satisfaction reflects the individual satisfaction of the learners with the given recommendations. Satisfaction is close to the motivation of a learner and therefore a rather important measure for learning. Finally, the drop-out rate mirrors the numbers of learners that dropped out during the learning phase. In educational research the drop-out rate is a

Table 7.1 Suitable measurements and their corresponding parameters building up an evaluation framework for recommender systems in Learning Networks

Measurements	Parameters
Technical measures	Accuracy Coverage Performance
Educational measures	Effectiveness Efficiency Satisfaction Drop-out rate
Social network measures	Variety Centrality Closeness Cohesion

very important measure because one aim is to graduate as many learners as possible during a learning phase.

The Social Network Analysis (SNA) measures are needed to estimate the benefit coming from the contributions of the learners for the network as a whole (Wasserman and Faust 1999). These are more specific measures that are mainly related to informal Learning Networks. SNA give us various insights into the different roles learners own in a Learning Network. SNA measures like variety, centrality, closeness and cohesion. Variety measures the level of emergence in a Learning Network through the combination of individual learning paths to the most successful learning routes. Centrality is an indicator for the connectivity of a learner in a Learning Network. It counts the number of ties to other learners in the network. Closeness measures the degree a learner is close to all other learners in a network. It represents the ability to access information direct or indirect through the connection to other network members. Cohesion indicates how strong learners are directly connected to each other by cohesive bonds. Peer groups of learners can be identified if every learner is directly tied to every other learner in the Learning Network.

These evaluation criteria can be conflicting. For instance, learners with many rated learning activities get a central role in a Learning Network from the SNA perspective. They get many direct ties to other learners through the huge amount of rated learning activities. From an SNA perspective these learners are beneficial for the Learning Network because they contribute heavily to it. But from the educational research perspective the same group of learners may be less important because their educational measures are quite poor. It might be that they needed much more study time (efficiency) or complete less learning activities successfully (effectiveness) compared to others learners in a Learning Network (LN). Therefore, further research regarding the evaluation of recommender systems for their support for learners in LNs is needed.

7.2.2 An Exemplary Experimental Setup to Evaluate a Recommender System in a Learning Network

To evaluate a recommender system in a Learning Networks we conducted, together with the Psychology faculty of the Open University of the Netherlands, the ISIS experiment. In ISIS (Individualised Support In Sequencing) the learners were able to study learning activity in any order instead of following the learning activities in a fixed order. The experiment focused on supporting learners in their course selection through personalised recommendation by a recommender system. The recommender system supported them with recommendations based on their learner profile and the behaviour of learners that were similar to them. We called that approach personalised navigation support and were especially interested in the learning outcomes of the learners and less in measures like algorithm performance of the machine-learning field. Thus, we selected effectiveness, efficiency, variety and satisfaction as evaluation criteria from the evaluation framework.

The following hypotheses were tested in the ISIS experiment, where the control group was provided with the Moodle virtual learning environment and a text book; whereas the experimental group was additionally provided with a recommender system that recommended best next learning activity based on successful choices of other learners with similar profiles.

The experimental group will be able to complete more learning activities than the control group (effectiveness). The proportion of completed learning activities is bigger in the experimental group compared to the control group.

The experimental group will complete (the same amount of) learning activities in less time, because alignment of learner and learning activity characteristics will increase the efficiency of the learning process (efficiency).

The experimental group has a broader variety of learning paths than the control group because the recommender system supports more personalised navigation (variety).

The experimental group will be satisfied with the navigational support of the recommender system (satisfaction).

It is always challenging to design an experiment corresponding to real life conditions because conditions are never the same like in a laboratory. However, the experimental design has to be strict as possible. In our example we adapted a formal course of the Psychology faculty of the Open University of the Netherlands to certain characteristics of professionals in Learning Networks. Consequently, we used the learning activities designed by domain experts and integrated them into a condition which was comparable to a Learning Network.

In the ISIS experiment we focused on the delivery of learning activities to professionals. We neglected the learning activity creation by learners and focused purely on learner support through recommender systems. In order to draw conclusions to professional development networks we especially addressed professional development characteristics like self-responsibility and its support through recommender systems. Therefore, we neglected the formal university conditions and constraints to design the experiment as similar as possible to the conditions of professionals in Learning Networks. Both groups got a maximum of freedom for their studies; in principle they were able to study the course over years. We informed both groups that they do not have to follow the learning activities in a certain order or pace. Further, the students could register for a final exam whenever they wanted, even without completing any of the online available multiple-choice tests for self-assessment.

Detailed results of the ISIS experiment that acts here as an example can be found in Drachsler et al. (2009b). The experiment examined the effects of the navigation support on the completion of learning activities measured (effectiveness), needed time to complete them (efficiency), satisfaction with the system (satisfaction), and the variety of learning paths (variety). The recommender system positively influenced all measures with having significant effects on efficiency, variety, and satisfaction on a four month run time.

Participants. In order to run experiments with recommender systems in Learning Networks the experimental designers should get as many participants as possible, because there is always a drop-out rate on various levels of participation. Thus, the group of participants that can be used for statistical analysis is getting smaller than the initial number of subscriptions.

In our example a total of 244 participants subscribed to the ISIS experiment. All participants were distance learners who studied the learning material on their own. Both the experimental and control group contained an equal amount of learners (122 learners per group) because the learners were randomly allocated, see Fig. 7.1. Twenty-four participants (19.7%) in the experimental group and 30 participants (24.5%) in the control group never logged into the Moodle environment. This group of non-starters was not included in our analyses. This leaves a group of 190 learners who did enter the Moodle environment; 98 in the experimental and 92 in the control group.

The group of actual starters had to be further differentiated into active and passive learners, because not all of the learners actually used or made progress in the Moodle environment. From the 98 participants in the experimental group 72 learners completed learning activities; from the control group 60 learners completed learning activities. Thus, in total a group of 132 were active learners during the experiment. We used this total amount of active learners to analyze hypothesis 1 (Effectiveness), hypothesis 2 (Efficiency), and hypothesis 3 (Variety).

The participants could voluntarily register for the new version of the course, and were informed that they were taking part in an experiment with a new learning environment. They were not informed that only half of the students would receive additional navigation support.

The conditions of the experiment allowed learners to start their studies whenever they want to. As a consequence not all students started at the same time; some of them started later and we got a *dynamic starting point* of students that have to be specially treated in the statistic analysis.

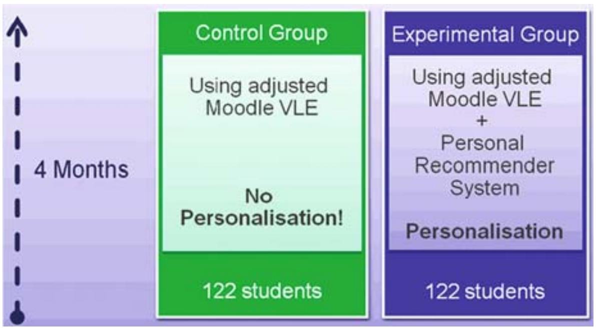

Fig. 7.1 Experimental design of the ISIS experiment

7.3 Statistical Analysis

To evaluate the effects for the experiment according to our hypotheses we applied a mix of different analysis procedures.

Effectives and *efficiency* measures where monitored every two weeks during the experimental runtime with a *repeated measurement design*. The repeated measurement design is part of the *generalised linear model* (GLM) a flexible generalisation of ordinary least squares regression. The GLM is commonly used in applied and social research. It is the foundation for the t-test and the Analysis of Variance (ANOVA).

For the evaluation of the *variety* of learning paths we developed a visualisation tool based on the multi-agent environment Netlogo. The tool shows an overlay of all learning paths within a group of learners. Thus, you can easily recognise their variance in the learning paths.

Satisfaction was measured through an online questionnaire and further analyzed with descriptive statistics. Therefore, we used an Open Source questionnaire tool called UCCASS (http://www.bigredspark.com/survey.html). In the following sections we introduce the different analysis techniques for the ISIS experiment.

7.3.1 Analysis of Effectiveness and Efficiency

In order to deal with a selection problem in our experiment we defined a goal attainment of 5 completed learning activities out of 17 in total. Our aim was to support as much learners as possible to complete these 5 learning activities as fast as possible. To measure the effectiveness and efficiency of the recommender system learners were taken into account that applied to the following condition; completed more than 5 learning activities, or successfully completed the final exam, or were still studying at the measure point. This condition leaves a number of 101 students at the end of the experiment ($n = 52$ in the experimental group and $n = 49$ in the control group). Regarding the individual dynamic starting points of the students the recorded measure in Table 7.1 contained 0 values in case students started later. In order to run a MANOVA analysis (Keselman et al. 1998) all individual starting points of the students were moved in one ‘starting’ column through deleting the 0 values. Therefore, Table 7.1 was transformed into a study progress table (see Table 7.2). Table 7.2 differs from Table 7.1 through moving the individual starting points into one ‘starting’ column (first column), and duplicating the study results towards the end of the table if the students complied to the above mentioned condition. To test hypothesis 1 and 2, we analysed the measures taken using SPSS version 12. To avoid inflated Type I error due to multiple tests, a priori tests of specific contrast scores were used.

The effectiveness and efficiency was analyzed by means of linear and quadratic trend analysis. To test hypothesis 1 and 2, we analysed the measures taken using SPSS version 12. To avoid inflated Type I error due to multiple tests, a priori tests of specific contrast scores were used. The effectiveness and efficiency was Averaged completion scores and averaged completion time during the two experi-

Table 7.2 Example table of biweekly recorded measures

Learner	Biweekly measure points						
	Oct	Oct 2	Nov	Nov 2	Dec	Dec 2	Jan
1	1	2	4	7	7	7	8
2	0	0	0	1	3	5	9
3	0	0	0	0	0	1	1
n	1	2	3	4	4	4	4

This table represents the not yet transformed recorded measures of the biweekly measure points. The 0 values are related to the individual starting point of the participants. The numbers show the amount of learning activities they completed successfully at the specific measure point.

Table 7.3 Example table of prepared biweekly measures for MANOVA analysis

Learner	Study progress per learner per measure point						
	1	2	3	4	5	6	7
1	1	2	4	7	7	7	8
2	1	3	5	9	9	9	9
3	1	1					
n	1	2	3	4	4	4	4

This table shows the actual study progress of all active learners. Therefore, all 0 values from Table 7.1 are deleted and the individual starting points were moved into one 'starting' column (first column). The MANOVA analysis in SPSS requires equally distributed values for each participant. If the learners completed more than 5 learning activities or they completed the final exam and not for each column a value was available their final study result was duplicated towards the final measure point (e.g. Learner 2). Learners that completed less than 5 learning activities were only taken into account when they still studied at the final measure point (e.g. Learner 4). Learners like learner 3 were not taken into account because they did not complete more than 5 learning activities and were not studying at the final measure point.

mental periods were transformed into linear and quadratic trend contrast scores by means of computation of orthogonal polynomials. We applied multivariate analysis of variance (MANOVA) for repeated measures on these a priori chosen contrast scores with Group as between subjects factor and Time as within subjects factor. A significant interaction of contrast scores with Group was followed by testing of simple contrast effects. Due to the a priori character of these tests, they were performed with the conventional Type I error of 0.05 (Tabachnick and Fidell 2001).

7.3.2 *Analysis of Variety of Learning Paths*

To test hypothesis 3, the variety of learning paths, we analyzed the behaviour of the learners with a Graph Theory approach (Gross and Yellen 2006). Therefore, we modelled the Learning Network in Netlogo 4 (Tisue and Wilensky 2004), and observed the completion of learning activities by the learners. Analysis software and example data set can be downloaded (http://hdl.handle.net/1820/1493). If a learner completed for instance first learning activity 1 and second learning activity 7 it was counted as traffic between learning activity 1 and learning activity 7. A line was drawn between both learning activities in the graph when the traffic became larger than 3.

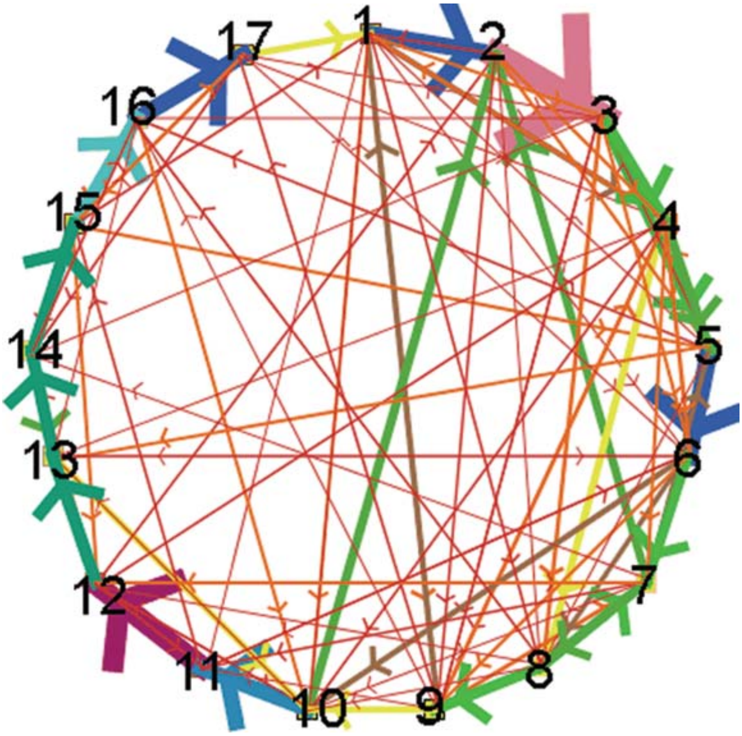

Fig. 7.2 Example picture of the variety of the learning paths. The standard curriculum order is indicated through numbers. *Arrows* show the learning paths of the learners in a group

If the learning path was used even more frequently, the traffic line got thicker and changed its colour. Consequently, the thickest path was used most often and the thinnest path was used only three times.

7.3.3 Analysis of Satisfaction with the Recommender System

To test hypothesis 4, the general satisfaction of the recommender system, we conducted an online recall questionnaire. The questionnaire was sent to all participants in both groups at the end of the experiment.

The Open Source UCCASS system makes online questionnaire an easy procedure. The system is also based on PHP and MySQL and therefore adjustable for

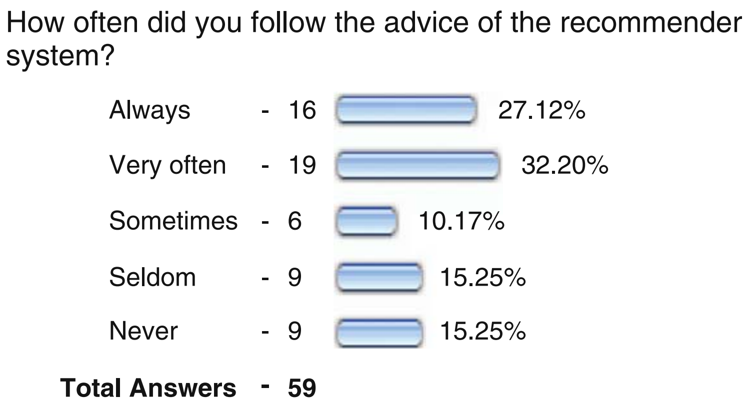

Fig. 7.3 Screenshot of the result view of the UCCAS online questionnaire system

certain wishes. It offers the possibility to load all participants into the MySOL database and to submit an invitation to every participant via e-mail. Further, any common question design is available and the results of the questionnaire can be filtered on different levels. The questionnaire results can easily exported from the database integrate in statistic programs like SPSS.

7.4 Suitable Recommendation Systems and Techniques

Depending on your resources and on the purpose of your experiment you have the choice between already existing recommender system plug-ins, programmable frameworks, or toolkits with additional functionality. There are also plenty of scientific publications regarding recommender system techniques which can be used to program own recommender systems (Adomavicius and Tuzhilin 2005; Burke 2002; Herlocker et al. 2004).

In the following section we will discuss various recommendation plug-ins, frameworks and a toolkit that can help to set up a recommender system environment for an experiment in Learning Networks research. Detailed information about recommender system techniques and how they can be adapted to the specific purposes for Learning Networks can be found in Drachsler et al. (2009b).

7.4.1 Available Recommender Systems

Currently, several recommender systems are available on various complexity levels. Some of them are available as plug-in for VLEs and websites and others are frameworks that have to be instantiated. Instantiations require programming effort but using a framework is still easier than creating an own recommender system from the beginning. A major advantage of the frameworks is that experimental researches

Table 7.4 Available recommender systems

Software type	Recommender systems
Plug-ins	Recommendation module for Drupal Vogoo
Frameworks	CoFE Taste Duine
Toolkit	Scout portal

can be sure to use the most efficient and effective recommendation algorithm from the machine learning field without being confronted with the mathematical calculations behind the algorithms. Instead of that the researchers have to feed the system with learning activities and learner profile information. The following systems are available.

On the plug-in side there are two suitable systems available, a *Content Recommendation Engine for Drupal* and the *Vogoo* recommender system. Both are based on PHP code and therefore easily to integrate into PHP based VLE like Moodle or Drupal.

The easiest way to integrate a recommender system into a VLE is the recommendation module for Drupal (http://drupal.org/node/920). It is limited to user-based collaborative filtering only. The module recommends interesting nodes, according to personal tastes of a user compared with other users in the system. Thus, users have to rate a couple of nodes (as 'Not Recommended', 'Recommended', or 'Highly Recommended') in order to get recommendations.

Another possibility is the Vogoo PHP Lib (http://www.vogoo-api.com/) a free PHP library licensed under the terms of the GNU GPL v2. The Vogoo PHP Lib has been designed with ease-of-use in mind. The team promises to add professional collaborative filtering functions to a website in minutes. Vogoo PHP includes two item-based and one user-based collaborative filtering technique and is therefore more flexible than the Drupal module. The Voogoo team also offers a commercial version called Vogoo PHP Pro as a proprietary version of Vogoo PHP Lib. This includes a highly optimised pre-computation engine for item-based collaborative filtering. Performance tests have shown an improvement of up to 20 times in execution speed for pre-computation scripts when compared to the GPL version.

On the framework side you can choose between three different recommender systems the *Taste*, the *CoFE*, and the Duine framework. An advantage of the recommender frameworks is the possibility to adapt the recommendation task to specific requirements of your experiment or your domain. This is not possible with the plug-ins, because they offer less flexibility for further development.

Taste (http://taste.sourceforge.net/) is a flexible collaborative filtering engine written in Java. It can be used as standalone application but it also can be used as external server, which exposes recommendation logic to your application via web services. The engine takes users preferences for items ('tastes') and returns estimated preferences for other items. Taste provides a rich set of components from which you can construct a customised recommender system from a selection of algorithms. It addresses important recommender system issue like performance, scalability and flexibility to provide fast recommendations also for huge data sets.

A similar project is the CoFE (http://eecs.oregonstate.edu/iis/CoFE/) project developed by the Intelligent Information Systems research group of Oregon State University. CoFE is a free, Open Source server for the Java platform that anyone can use to set up a recommendation system. Features include individual items recommendations, top-N recommendations across all items, top-N recommendations based on one type of item. Recommendations are computed using a popular, well-tested nearest-neighbour algorithm (Pearson's algorithm).

Finally, the Duine (http://sourceforge.net/projects/duine/) framework allows users to develop own prediction engines for recommender systems. Duine is also Open Source and available for free. Duine contains a set of recommendation techniques, ways to combine these techniques into recommendation strategies, a profile manager, and it allows users to add their own recommender algorithm to the system. Duine already includes some further functionality like a component for the management of user profiles. The result of a Duine prediction engine is the retrieved set of information with added data about how interesting each piece of information is for the user.

In the category toolkits, the Scout Portal Toolkit is available which makes it possible to set up a whole content management system. It can also be used to set up a VLE such as the learning languages project (http://www.learninglanguages.net/) that makes advantage of it. It is one of the easiest and fastest ways to setup an experiment for Learning Networks including a recommender system. The Scout Portal Toolkit provides a number of features beside a recommender system. It also enables cross-field searching, resource annotations by users, intelligent user agents, and resource quality ratings by users. The recommender system uses item-based filtering technique, based on community ratings.

Most of the time one of the presented recommender systems is more suitable then other ones for certain research conditions. If researchers want to make a case study within the concept of Learning Networks and no programming capacity is available we suggest using the Scout Portal Toolkit. In this case, the setup of the Learning Network is rapidly done and it already contains a recommender system. Similar applies for the Voogoo and the Drupal plug-in. In both case the experimental team has to add learning activities to a VLE and can additionally add a recommender system with minor programming knowledge. The Scout Portal Toolkit and the Drupal plug-in are both based on one recommendation technique only. The Voogoo plug-in offers already three different recommendation techniques but therefore it is also a bit more challenging regarding the implementation.

If the experimental designers have more specific research questions regarding recommender system in Learning Networks we suggest to use one of the recommender system frameworks. They allow much more adjustments of the systems to any experimental design and still hide complexity of recommender system algorithm. However, they definitely require more programming capacity and a deeper understanding of recommender system insights than the other solutions.

In case experimental designers decide to design a recommender system from the bottom onwards, they have the most freedom and possibilities for the development of a specific recommender system for a certain recommendation task. There are three overview articles available that are supportive for a selection of the most suitable recommendation technique (Adomavicius and Tuzhilin 2005; Burke 2002; Herlocker et al. 2004). For the ISIS experiment we decided to develop our own recommender system with particular aspects regarding professional development in distributed Learning Networks. We did so because we collaborated with the Psychology faculty at our institute that wanted to evaluate the *Moodle LMS* for their distance courses. In the joined ISIS project we supported them to set up and gain experience with Moodle. This way, we could rely on the learning material and the students as participants for our experiment. At the end of the ISIS project Psychology was satisfied with the research results and decided to use Moodle as LMS for all courses as well as to use the recommender system. Currently, they further develop the experimental prototype of the recommender system for support in other courses as well.

7.4.2 The Techniques We Used in the ISIS Experiment

For the ISIS experiment we decided to combine a domain ontology with a stereotype filtering technique . Recommender systems with a combined recommendation strategy provide more accurate recommendations when compared to single techniques recommender systems (Melville et al. 2002; Pazzani 1999; Soboro and Nicholas 2000). The ontology used personal information of the learner (e.g., interest) and compared that with the domain knowledge to recommend the most suitable learning activity. Stereotype filtering used profile attributes of the learners (e.g., interest, motivation, study time) to create learner groups and recommend learning activities preferred by similar learners.

The recommender system advices the next best learning activity to follow based on the interest of learners (ontology-based recommendation), and on the behaviour of the peers (stereotype filtering). If only information about the interest of a learner was available, then ontology-based recommendation technique was used, else the stereotype filtering technique was applied. The underlying recommendation strategy is presented in Fig. 7.4.

The use of the stereotype filtering was prioritised and the ontology approach was used mainly to cover the 'cold-start problem' (Herlocker et al. 2000) of the stereotype filtering technique. The stereotype filtering technique was personalised through attributes of the personal profile of the learners. If it was not possible to give any advice it disabled one of the personal attributes and tried to make a recommendation based on larger peer group with less common attributes.

Only in the case that the stereotype filtering was not able to provide any recommendation, the recommender system created ontology-based recommendations. The ontology visualised in Fig. 7.5 consists of two top domains (e.g., 'Environmen-

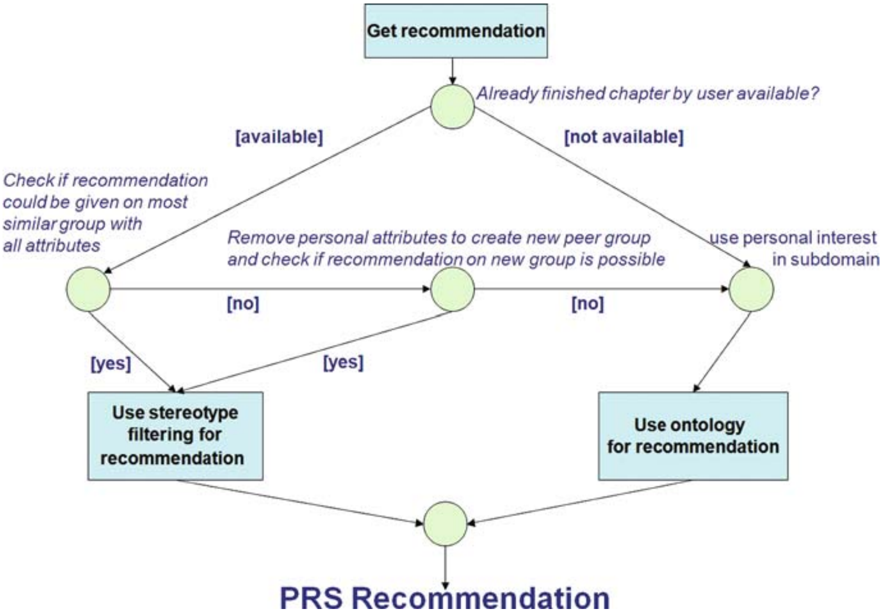

Fig. 7.4 Recommendation strategy for the implemented recommender system

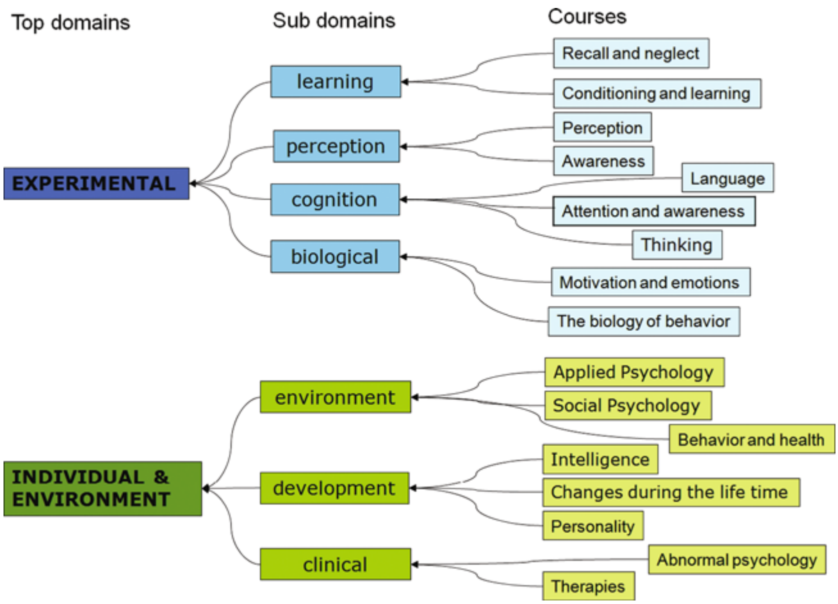

Fig. 7.5 Structure for ontology based recommendations

tal Psychology') that contain several sub domains (e.g., 'learning'), each containing two or three courses (or learning activity) (e.g., 'recall and neglect'). The learners had to select a special interest (one of the sub domains of the ontology) in their profile. If the learners had chosen a sub domain (e.g., 'clinical'), they received recommendations on courses located in that particular sub domain. If none of these courses had been completed by others so far, the recommender system randomly recommended one of them. If one course had already been completed by the learner the other course(s) was/were recommended. If all courses of the sub domain (e.g., 'clinical') were completed the ontology recommended a course that was part of the top domain 'Environmental Psychology'.

7.4.3 The Virtual Learning Environment

We selected Moodle as VLE (Dougiamas 2007), because it is an Open Source solution written in the PHP programming language and therefore easily adaptable to our experimental needs. The learning activities and the recommender system were implemented into Moodle. Moodle was adjusted to the experimental setup, thus some functionality of Moodle was blurred out and other functionalities like a multiple-choice tool where additionally added. Figure 7.6 shows the overview screen of learning activities for a learner in the experimental group. The overview is divided into three columns. The right column shows the learning activities the learner still has to study. The middle column presents the courses the learner is already enrolled for. Finally, in the left column all completed courses are listed.

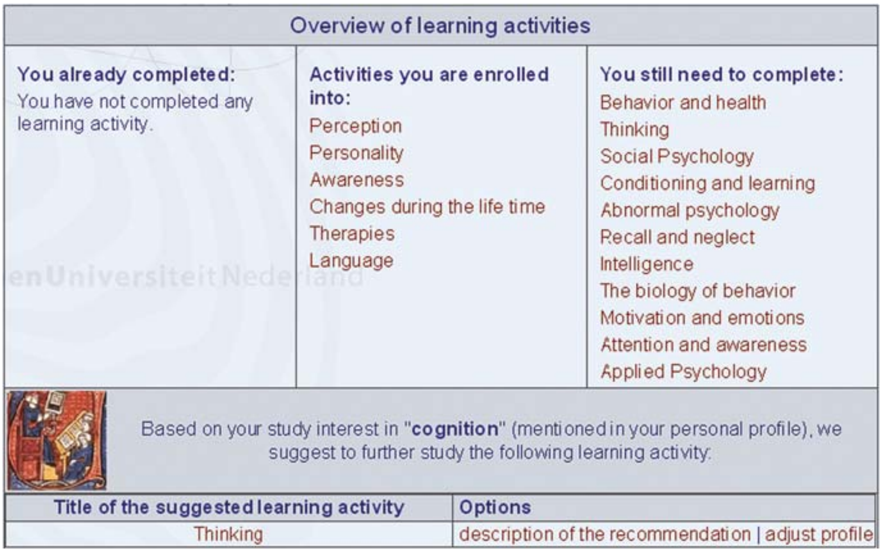

Fig. 7.6 Overview page of the experimental group with a recommendation

Below an explanation of the recommendation is given. In this screen, the recommender system has recommended 'Thinking' as next best course. Next to the recommendation there are additional options to get further information about the recommendation and to adjust the preferences set in the learner profile.

The Learning Network that was based on a Moodle adaptation contained 17 learning activities with an average study load of 12 hours. Completion of each learning activity was assessed by multiple-choice tests consisting of seven equally weighted questions. A score of 60% or more was considered as a successful completion of the learning activity. With the Moodle environment the learners received an Introduction to Psychology handbook that contained additional information to the 17 learning activities. All learning activities were separate entities in Moodle, setup according to the same didactical structure. The Moodle environment contained all further learning materials, including support and guidance, task assignments, progress tests, additional pictures and links, summaries, and other attractive learning tasks.

7.5 Conclusion

We have presented all the required tools and concepts that are needed to set up an experiment with recommender systems in Learning Networks for professional development. We have given an overview about a suitable experimental design and offered an example for that. Further, we introduced statistic methods and procedures to test hypotheses that can be based on a selection of variables from an evaluation framework. Finally, we discussed various available recommender system and suitable virtual learning environments to create a Learning Network. In this final section we want to give incentives for future research on the navigation support through recommender systems in Learning Networks.

Following experiments in this field can vary on four key elements: Changing the underlying recommendation algorithms, Adjusting the pedagogic context, Addressing a specific user group (older people, more technologically literate, higher educational achievement), and Using a different VLE or other educational services for the experiment.

These four key elements can be combined in various experimental settings. Based on the ISIS experiences we suggest to continue with variations on the second and fourth elements. We aim to apply the use of informal learning activities created by the professional to address the navigation problem in Learning Network on a higher level. Research in this area should make advantage of learning activities available in Web 2.0 services like *wikipedia.com, youtube.com* or *slideshare.com*. Future experiments in this area should use a mixture of formal and informal learning activities to simulate a Learning Network. In this case, it is hardly possible to apply a domain ontology because of the 'open corpus problem' (Brusilovsky and Henze 2007). The open corpus problem applies when an unlimited set of documents are given that can not be manually structured and indexed with domain concepts and metadata from a community. Thus, to prepare recommendations for informal learning activities different recommendation strategies have to be invented. Therefore, Open Educational Resources (OER) (Hylén 2006) are also a very interesting source for the data base of future experiments in Learning Networks. Experimental designers should consider mixing different kinds of these OER repositories and maybe additionally combining them with learning activities created by learners.

An unsolved issue is the measurement of accepted recommendations by the learner. The problem is the definition of an 'accepted recommendation'. Did learners appreciate a recommendation when they navigated to a recommended learning activity? Or did learners accept a recommendation when they used the recommended learning activity more than 5 min? Anyway an objective measure is needed to indicate a successful recommendation for a learning activity. In e-commerce recommender system a recommendation was successful if a consumer finally bought a recommended product. In the case of professional development we have to measure at least that a learner is busy with a learning activity. This could be done with various indicators like 'time spend on learning activity', 'click rate', 'repeated use of the learning activity', and 'added content to learning activity' in an interaction model.

Finally, in the ISIS experiment we limited ourselves to show only the 'best next learning activity', based on our recommendation strategy to the learners. We did that for experimental reasons. It is also thinkable to select a different experimental design and offer sorted lists of recommendations. In the real life of professionals a list or a sequence with suitable recommendations might be more valuable than a single recommendation.

References

Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering **17**(6), 734–749 (2005)

Brockett, R.G., Hiemstra, R.: *Self-direction in Adult Learning: Perspectives on Theory, Research and Practice* (London: Routledge 1991)

Brusilovsky, P., Henze, N.: Open corpus adaptive educational hypermedia. In P. Brusilovsky, A. Kobsa, W. Nejdl (Eds.): *The Adaptive Web: Methods and Strategies of Web Personalization* Lecture Notes in Computer Science ed., Vol. 4321, pp. 671–696 (Berlin, Heidelberg, New York: Springer 2007)

Burke, R.: Hybrid recommender systems: survey and experiments. User Modeling and User-Adapted Interaction **12**(4), 331–370 (2002)

Dougiamas, M.: *Moodle* [Retrieved 12 June 2007, from http://moodle.org/] (2007)

Drachsler, H., Hummel, H., Koper, R.: Personal recommender systems for learners in lifelong learning: requirements, techniques and model. International Journal of Learning Technology, **3**(4), 404–423 (2008)

Drachsler, H., Hummel, H., Koper, R.: Identifying the goal, user model and conditions of recommender systems for formal and informal learning. Journal of Digital Information, **10**(2), 4–24 (2009)

Drachsler, H., Hummel, H., van den Berg, B., Eshuis, J., Waterink, W., Nadolski, R.J., Berlanga, A.J., Boers, N., Koper, E.J.R.: Effects of the ISIS Recommender System for navigation support in self-organised Learning Networks. Journal of Educational Technology and Society, **12**(3), 122–135 (2009b)

Goldberg, K., Roeder, T., Gupta, D., Perkins, C., Eigentaste, A.: *Constant Time Collaborative Filtering Algorithm*, Vol. 4, pp. 133–151 (New York: Springer 2001)

Gross, J.L., Yellen, J.: *Graph Theory and Its Applications* (Boca Raton, FL: Chapman & Hall/CRC 2006)

Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.: Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems **22**(1), 5–53 (2004)

Herlocker, J.L., Konstan, J.A., Riedl, J.: Explaining collaborative filtering recommendations. *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, pp. 241–250, 2000, Philadelphia, PA

Hylén, J.: Open educational resources: opportunities and challenges.*Proceedings of Open Education*, pp. 49–63, September 27–29, 2006, Utah State University, Logan, UT [Retrieved October 5, 2008 from http://cosl.usu.edu/conferences/opened2006/docs/opened2006-proceedings.pdf/]

Keselman, H.J., Huberty, C.J., Lix, L.M., Olejnik, S., Cribbie, R.A., Donahue, B., Kowalchuk, R.K., Lowman, L.L., Petoskey, M.D., Keselman, J.C.: Statistical practices of educational researchers: an analysis of their ANOVA, MANOVA, and ANCOVA analyses. Review of Educational Research **68**(3), 350–386 (1998)

Linden, G., Smith, B., York, J.: Amazon.com recommendations: item-to-item collaborative filtering. IEEE Internet Computing **7**(1), 76–80 (2003)

Longworth, N.: *Lifelong Learning in Action – Transforming Education in the 21st Century* (London: Kogan Page 2003)

Melville, P., Mooney, R.J., Nagarajan, R.: Content-boosted collaborative filtering for improved recommendations. *Proceedings of 18th National Conference on Artificial Intelligence*, pp. 187–192, July 28–August 1, 2002, Edmonton, Alberta, Canada

Nadolski, R., Van den Berg, B., Berlanga, A., Drachsler, H., Hummel, H., Koper, R., Sloep, P.: Simulating light-weight personalised recommender systems in learning networks: A case for pedagogy-oriented and rating-based hybrid recommendation strategies. Journal of Artificial Societies and Social Simulation (JASSS), **12**(1), 4 (2009)

O'Sullivan, D., Wilson, D., Smyth, B.: Improving case-based recommendation a collaborative filtering approach. *Proceedings of 6th European Conference on Case-Based Reasoning, ECCBR,* Aberdeen, Scotland, UK: LNAI Springer, 2002

Pazzani, M.J.: A framework for collaborative, content-based and demographic filtering. Artificial Intelligence Review **13**(5), 393–408 (1999)

Sarwar, B.M., Karypis, G., Konstan, J., Riedl, J.: Recommender systems for large-scale e-commerce: scalable neighborhood formation using clustering.*Proceedings of Fifth International Conference on Computer and Information Technology,* 2002 [Retrieved October 5, 2008 from internal-pdf://2002_RS_for_ large_scale_e-commerce_sarwar_cluster-1331033344/2002_RS_for_large_scale_e-commerce_sarwar_cluster.pdf]

Schafer, J.B., Konstan, J., Riedl, J.: Recommender systems in e-commerce. Paper presented at the *1st ACM Conference on Electronic Commerce*, Denver, Colorado, 1999

Shuell, T.J.: Designing instructional computing systems for meaningful learning. In M. Jones, P.H. Winne (Eds.): *Adaptive Learning Environments*, pp. 19–54 (Berlin: Springer-Verlag 1992)

Soboro, I.M., Nicholas, C.K.: Combining content and collaboration in text filtering. *Proceedings of the IJCAI Workshop on Machine Learning in Information Filtering*, pp. 86–91, August 1999, Stockholm, Sweden, 2002

Tabachnick, B.G., Fidell, L.S.: *Using Multivariate Statistics*, 4th ed. (Boston, MA: Allyn and Bacon 2001)

Thorpe, M.: *Evaluating Open and Distance Learning* (Essex, UK: Longman Harlow 1988)

Tisue, S., Wilensky, U.: NetLogo: a simple environment for modeling complexity. *Proceedings of International Conference on Complex Systems*, Boston, MA, USA, 2004

Vygotsky, L.S.: *Mind in Society: The Development of Higher Psychological Processes* (Cambridge: Harvard University Press 1978)

Wasserman, S., Faust, K.: *Social Network Analysis: Methods and Applications* (Cambridge: Cambridge University Press 1999)