Using Simulations to Evaluate the Effects of Recommender Systems for Learners in Informal Learning Networks

Hendrik Drachsler, Hans Hummel and Rob Koper

Educational Technology Expertise Centre, Open University of the Netherlands,
PO-Box 2960, 6401 DL Heerlen, hendrik.drachsler@ou.nl

Abstract. Learning Networks consist of learners who are able to create, share and study learning activities. Through the emerging behaviour of such a network it may consist of a large amount of learning activities. Thus, the learners face the problem to select the most suitable learning activity regarding their learning goals in order to study the most efficient and effective learning path. This simulation study explores the use of recommender system technology like collaborative filtering to solve this problem. Learning activities that have been rated by comparable learners are recommended to the learners as navigational support. The simulation tool models a Learning Network in which learners search for, enrol in, study and rate learning activities. This article introduces our theoretical background for recommender systems in informal Learning Networks. It presents a model and flow chart of the simulation. It explains which collaborative filtering techniques we want to investigate and finally presents the experimental design for testing recommender systems in informal Learning Networks.

Keywords: SIRTEL, learning networks, recommender systems, collaborative filtering, simulation, informal learning

1.   Introduction

Informal learning describes the learning phase of so called lifelong learners that are not participating in any formal learning context like universities or schools. Lifelong learners are acting much more self-directed and they are responsible for their own learning pace and path [1]. In addition, the resources for their learning might come from many different sources: expert communities, work context, training or even friends might offer an opportunity for an informal competence development. The learning process is also not designed by an institution or responsible teachers like in formal learning but it depends to a very large extent on individual preferences learners have or choices that learners take. In general, when taking up on this responsibility, lifelong learners need to become self-directed [2], and perform in different Learning Activities (LAs) in various contexts at the same time. The learners are free to decide what, when, where and how they want to learn.

The design of a Learning Network (LN) addresses lifelong learning issues like self-responsibility. In LNs, the lifelong learners are able to publish their own LAs, or share, rate, and adjust LAs from other learners. Therefore, LNs are learner-centred and their development evolves bottom-up through the participation of the lifelong learners. The LN approach focuses on the support of the neglected informal learning part that is becoming more important through the Web 2.0 development nowadays. Therefore, it is in contrast to other learning environments, which are designed only top-down, because their structure, LAs, and learning plans are predefined by an educational institution or domain professionals (e.g., teachers).

The enormous amount of information that is published by any active learner (considering the *wisdom of the crowds* theory and Web 2.0 development) [3] makes it hard to get an overview of available LAs and to identify the most appropriate in a LN. The learners need support to manage this information overload [4]. Thus, filtering, clustering and recommendation technologies are promising to handle the information overload.

One possibility to address the information overload problem is the use of recommender systems. Recommender systems suggest information to users based on their personal preferences or a profile. They can be based on various technologies. Most famous recommender system technologies are collaborative filtering algorithms. Successful examples from the consumer world are the recommender systems from *amazon.com*, *ebay.com* or *netflix.com*. We are inspired by these solutions and want to develop recommender systems that support lifelong learners in informal LNs.

Therefore, we have to take into account the specific conditions of LNs. Informal learning offers are emerging from the bottom upwards through their communities. Thus, there is an absence of maintenance and structure in informal learning that is also called the 'open corpus problem' [5]. 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. The LAs in LNs are mainly structured through tags and ratings given by the lifelong learners. Therefore, bottom-up recommendation techniques like collaborative filtering (CF) are more appropriate because they require nearly no maintenance and improve through the emerging behavior of the community. A recommender system for informal learning has to behave as independent as possible without maintenance by an institution and rely on the data that is given in informal LNs.

In this paper, we present a model of the simulation for the exploration of CF for the navigation support in LNs. We want to analyse the relationship between the micro (learner) and macro level (LN) of recommender systems in LNs. Therefore, we address questions like: How does a lifelong learner benefit from recommender systems in a LN? But also, how does the LN as infrastructure benefit from the contributions of its member?

A simulation tool can be supportive to define requirements for different kinds of recommender system technologies for LNs before actually starting the costly process of development, implementation, testing and revision in real field experiment. Field experiments with real learners need careful preparation as they cannot be easily repeated or adjusted within a specific timeframe. Another advantage of simulations is that they avoid some ethical and practical constraints of field experiments. Differently to real world experiments, we do not have to take care of real participants and

therefore are able to setup a rigorous experimental design. For instance we do not have to cover the 'cold-start' problem of recommender system [6] which happen when no behavioural data is saved in the recommender system in the beginning. Simulations enable us to use a 'warm-up period' where the simulation computes the emerging behavior of learners over years as a synthetic data set for the recommender system. After this warm up period, we start the measurement of the experimental variables for the applied recommender system.

In the following sections, we first discuss related work from the recommender system and the LN research field (section two). Further, we present the simulation model and flow chart of one simulation run (section three). After doing so, we explain the CF techniques and the synthesized data set that will be applied in the simulation tool (section four). Finally, we present the experimental design for the planned simulations (section five).

2.   Related work

Research results about the conditions and performance of different CF algorithms are well known in the recommender system field [6].

Traditionally, user-based CF works by searching a large group of people and finding a smaller set with tastes similar to yours. It looks for other things you like and combines them to create a ranked list of recommendations. The decisions to define if people are similar to each other are most of the time context related. The similar technique is known as item-based CF. Item-based CF is working similar to user-based CF. It allows many of the calculations to be performed in advance so that users can get the recommendations more quickly.

As a contribution to the SIRTEL discussion [7] we want to evaluate the effects of user- and item-based CF for informal LNs in different sizes. Therefore, we focus especially on the emerging effects of personalised recommendations in LNs to support the learning outcomes of lifelong learners.

Regarding Gilbert & Troitzsch [8] simulation studies can be designed through abstracting a model from a research target and further develop a simulation for that model. An advanced step in simulation design is the comparison of the simulation results with data collected in field studies of the research target. According to this method we based the parameters and conditions of our simulation on findings of previous studies. We designed a research circle that combines findings from field test studies with conclusions of simulation studies in order to guarantee the validity of assertions for informal LNs. This research circle started with a simulation study by Koper [9] to test the theory behind the informal LN approach. In a second step, a first field test experiment was conducted by Janssen et al. [10] to gather experience based on real data. In a third step, an additional field experiment was carried by Drachsler et al. [11] to gather additional real data for upcoming simulation studies. The latest simulation study that builds on the earlier field studies was designed by Nadolski et al. [12]. We continue the research with this sophisticated simulation to test additional recommendation techniques for informal LNs. Nadolski et al. combined an ontology and stereotype filtering recommendation techniques with an indirect rating

mechanism for one size of LNs. Therefore, they created treatment groups for the simulation through combining the recommendation techniques in various ways. Nadolski et al. tested which combination for recommendation techniques in a recommendation strategy had a higher effect on the learning outcomes of the learners in a LN. Their study confirms that providing recommendations leads towards more effective, more satisfied, and faster goal achievement. Furthermore, their study reveals that a bottom-up CF recommendation technique including a rating mechanism is a good alternative to maintain intensive top-down ontology recommendation techniques.

Our approach wants to extend the Nadolski et al. study through evaluating additional recommendation techniques for different sizes of LNs. Therefore, we want to apply the same learner and LA models and further design three different LNs with different dense data sets regarding the amount of learners, available LAs, and transaction in the system. We want to test user- and item-based CF techniques in a single setting without combining them in a recommendation strategy directly. Similar to Nadolski et al. we also want to assess the algorithms for their usability for recommendation strategies for hybrid recommender systems in LNs. Hybrid techniques combine recommendation techniques in order to provide more accurate recommendations. Several studies have already demonstrated the superiority of hybrid techniques when compared to single techniques for recommender system [13-18]. Since, LNs can exist in various conditions it is expected that a hybrid recommender system (a combination of recommendation techniques) is most suitable for LNs. Our research on simulation wants to identify promising recommendation techniques for different conditions of LNs to finally combine them in a hybrid recommender system that fits to different LNs characteristics. Most important for all recommendation techniques is their suitability to the needs of lifelong learning in informal LNs [7].

Sarwar et al. [19] has proven that item-based CF can give more accurate results than user-based CF for very large datasets (larger than movielens.org). Sarwar et al. also measured a higher performance of item-based CF versus user-based CF for the used data sets. We are interested if these differences also affect our research on learner support in LNs. It is known in the recommender system field that different algorithms perform better or worse on different data sets [20]. A mayor difference between data sets is their size regarding users, items and transactions. For instance, the well known *Movielens* data set consists out of 6040 users and 3900 movies with 1 million ratings. From the LN perspective a data set like the *Movielens* data set is a rather huge one, thus maybe the conclusions regarding the differences between user- and item-based CF from Sarwar et al. do not apply for recommender systems in LNs. We expect LN sizes between 100 and 1200 LAs and 250 to 1500 learners per LN. We align these assumptions with usage statistics of communities which act similar to LNs like the OpenLearn project[1] and the earlier simulation studies by Nadolski et al. and Koper.

Based on our earlier experience [21], we believe that a recommender system has to take pedagogy rules and learning characteristics into account to support learners on their learning process. Therefore, a recommender system for learners requires deeper

---

[1] http://www.open.ac.uk/openlearn

reasoning than other domains. Simple semantics like "People who liked X also liked Y" might be misleading for learning recommender systems. For recommender systems in LNs we might need semantics like "People who studied X, Y, and Z on competence level 3 and prior knowledge level 2 seem to have the same learning goal, thus we recommend studying W". Thus, in our simulation study we introduce pedagogy research results like Vygotsky's "zone of proximal development" that follows the pedagogical rule 'recommended LAs should have a knowledge level that is a bit above learners current competence level' [22].

Additionally, a recommender system that is heading for learner support in LNs also should be evaluated on educational and network measures besides recommender system field measures [7]. Therefore, we have to combine recommender system algorithms measures like *accuracy* with learner performance measures like *effectiveness*, *efficiency* and *drop out rate* (e.g. *Do the learners perform more efficient or effective regarding their learning goal with technique A or B?*). Regarding the emerging behavior of LNs we also have to assess the benefit of the contributions of the learners for the LN as a whole. Social network analysis measures like *variety* are most suitable to estimate that (e.g. *How does the network benefit from the contribution of their members?*).

The results of this simulation study should clarify when a specific recommendation technique is more appropriate for specific sizes of a LN. Further, it shows whether the differences of user- and item-based CF also apply to our research field on LNs.

In the following section we present the adapted simulation model for our simulation tool.

3.   The Learning Network Simulation

As mentioned earlier we extended the previous research on simulation through defining two new foci for the evaluation of recommender system in LNs. First, we want to apply the so far unused user- and item-based CF techniques for LNs. Secondly, we want to test these algorithms in three different LNs with different dense data sets regarding the amount of learners, available LAs, and transaction in the system. In the following section we present our simulation model that is based on previous work by Koper (2005), and Nadolski et al..

3.1. The Simulation Model

Regarding the evaluation of purely bottom-up techniques (item- and user-based CF) for the navigation support of learners in LNs, we excluded preferences that were related to ontology based recommendations from the initial Learner model designed by Nadolski et al.. The remaining *Learner Model* and *LA Model* are in line with the previous research. Both models present our approach to simulated learners acting with LAs in a LN. In order to clarify the relations between the different simulation objects we divided the simulation model into a *Learning Network Interaction Model* and a *Recommender System Interaction Model*. In both models unused attributes are darker

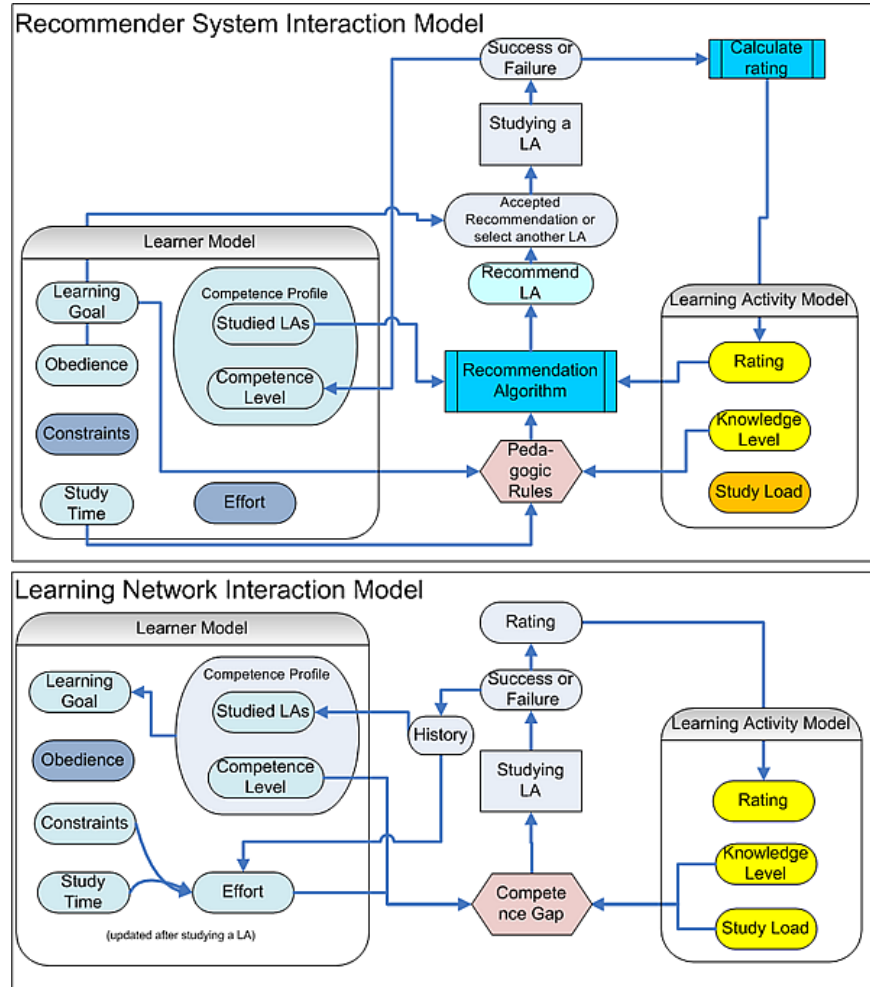than used attributes. Further, used attributes have a connection to another entity in the model.

Fig. 1. Simulation Model for the Simulation to tool

3.2. The Learning Network Interaction Model

The Learner Model
The *Learner Model* consists of variables we explain now in detail.

The *Learning Goal* is a randomly distributed variable that defines the goal or interest of a learner.

The *Competence Profile* is restricted to one competence which can include up to three *Competence Levels*. It is assumed that a learner will only start studying LAs that

can contribute to reach the *Learning Goal*. Successfully completed LAs contribute to their associated *Competence Level*. Each *Competence Level* included in the *Learning Goal* has its own amount of LAs that have to be successfully completed for its mastery.

The *Competence Level* of the learner indicates the learner's achievement with respect to the *Learning Goal* and the influences by the results of *Success / Failure* value after the study period, thus it is a dynamic variable.

The learner *Effort* is at the start of the simulation normally distributed amongst learners, but it changes dynamically during the learners study. The *Effort* value determines if a learner will drop out or not [23]. If the *Effort* gets below zero, a learner will drop out and will not graduate. *Effort* depends on *previous Effort*, *Competence GAP* between learners and LAs, *Constraints,* and the *History* of *Success / Failures* values. Several successes in a row are expected to increase *Effort* (more motivated), whereas failure will have negative influences on the motivation of a learner, ultimately a learner could drop out of the LN.

Constraints are related to the research by Koper (2004). Koper mainly modeled negative constrains so called disturbance factors. Nadolski et al. added also positive factors and called these *Constraints*. *Constraints* are related to a learning flow, a noisy or quiet environment, stress, etc. They influence the amount of *Effort* learners want to invest for studying. Constraints are a randomized factor for each studied LA. For calculation purposes, we define constraints as '1' in case of positive effects, '-1' in case of negative effects, and '0' in case of a neutral effect.

Obedience differs between learners but remains constant for each learner in the simulation. *Obedience* represents whether or not following a recommendation [24]. In one of the previous studies we identified an obedience level of 60% [11] which is similar to other studies [25]. Thus, we aligned the Obedience p*arameter* in the simulation with the result from the real world.

The *Study Time* has the same scale as the simulation frequency (1 run = 1 week). It is also randomly distributed among the learners. It has an influence in case of a competence gap between a learner and a LA. A high *Study Time* can bridge the *Competence Gap* through investing more *Effort*.

The Learning Activity Model
Rating of a LA is based on the behavior of the learners and computed as an indirect measure. Ratings are influenced by whether or not the learner successfully completes a LA, and the *Effort* the learner spends. Except for *Rating*, all characteristics in the *Learning Activity Model* remain unchanged.

The *Knowledge Level* is randomly distributed variable among the LAs. It is a constant that represents the complexity of the LA.

The Study Load is the time a learner has to invest before doing an LA examination.

The Actions in between the Learner and Learning Activity Model

The *Competence Gap* measures alignment between the *Competence Level* of the Learner and the *Knowledge Level* of the LA. A pedagogy reasonable match occurs if the *Knowledge Level* is one level above the C*ompetence Level of a Learner* [22]. Mismatches for competences will have a negative influence on learner's *Effort*, whereas good matches will increase *Effort*. Consequently, for LAs that are a bit beyond learners' *Competence Level* more *Effort* can lead to their successful completion.

If *Success* is true, the learner passes the LA examination and achieves the *Knowledge Level* corresponding with the LA and the learning goal and *Competence Level* will improve. A *Failure* will be registered in the *History* of the model and can have an influence on the learner's *Effort* if the *Failures* occur more recently. A *Failure* will not decrease the *Competence Level* of a learner.

3.3. The Recommender System Interaction Model

The same models apply for the *Recommender System Interaction Model* but different attributes of the previous explained models are used for the computation of the LN.

For instance the *Obedience* parameter is now needed to calculate if a learner obeys a recommendation or not. Also the recommendation algorithms and the rating mechanism are shown as a process to indicate that they are computed in this model.

An additional difference is the use of *Pedagogy Rules* in the recommender system that aims on the recommendation of LAs to already mentioned rules like going from simple to more complex LAs. The *Pedagogy Rules* entity is corresponding to the *Competence Gap* by suggesting most suitable LAs to bridge the *Competence Gap* and to achieve the *Learning Goal* in an efficient manner.

3.4. Flow Chart of the Simulation

Having explained the underlying models we now want to present a flow chart diagram that makes clear how the simulation tool works for the computation of one study week (see Figure 2).

In the beginning all completed *LAs* are excluded from the *LAs* that can be selected. Based on the *Treatment Groups* of the *Learners* they decided either for a random *LA* or they got a recommendation for specific *LAs* based Item- or User-based CF. The recommended *LAs* follows certain implemented *Pedagogy Rules*. Based on the success the learners have with the selected LA they either *Graduated* (if the *Learning Goal* is reached), or they *Drop out* (if the *Effort* becomes smaller 0), or they just *Study further* (in this case they restart at the beginning of the flow chart).
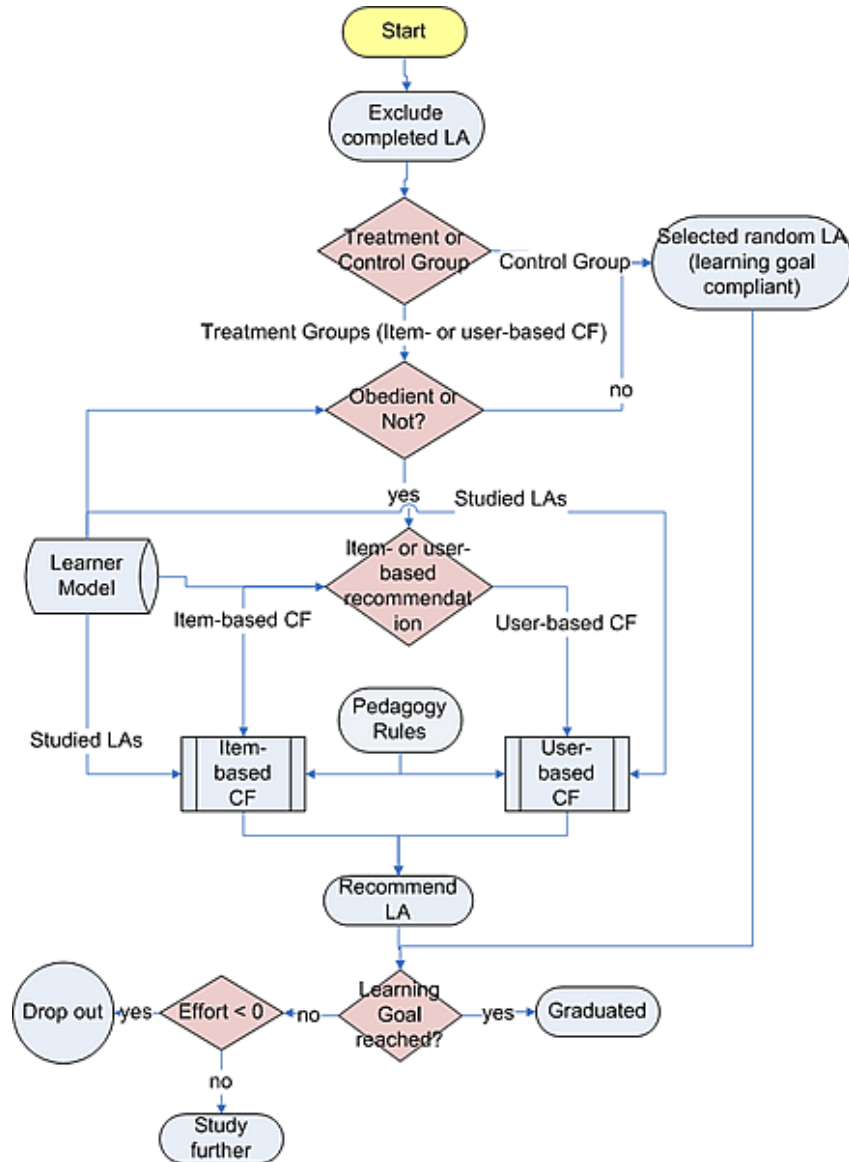
Fig. 2. Flow chart diagram of one simulation run.

4.   The collaborative filtering algorithms

CF is one of the widely used recommendation approaches. It characterizes users and item implicitly by their previous interactions. The simplest example is to recommend the most used item to all users. Researchers in the machine-learning field are advancing CF algorithm to provide personalized recommendation to users. Thus, specific item- and user-based CF approaches are available. The main advantages of the techniques are the usage of information that is provided bottom-up by user ratings, that they are domain-independent and require no content analysis and that the quality of the recommendation increases over time [6].

As mentioned earlier, for the simulation we want to focus on the popular user-based and item-based CF algorithms and apply these for the support of learners in LNs. We use the following notation to describe the CF problem in LNs. To prevent confusions with the notation we call the LAs in the following 'learning resource' and use LA for their notation.

The problem input is an $M \times N$ transition matrix $A=(a_{ij})$ associated with M learners $L = (L_1, L_2, ..., L_M)$ and N learning resources $LA = (LA_1, LA_2, ..., LA_N)$. We focus on recommendations based on transactional data between learners and learning resources. That is a_{ij} can take the value of 0 or 1, with 0 representing the absence of any transaction and 1 representing a successfully completed LA between L_i and LA_j. We considered a CF algorithm output to be *likely values* for interesting learning resources for individual learners. The recommendation consists of a ranked list of K learning resources with the highest likely values for an individual learner.
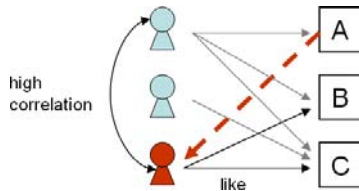
4.1. User-based CF

Fig. 3. Technical drawing of user-based collaborative filtering algorithm (Kim, 2006)

User-based CF correlates users by mining their (similar) ratings and then recommends new LAs that were preferred by similar users (see Figure 4). The algorithm first computes a learner similarity matrix $WL = (wl_{st})$, s, $t =1, 2, ..., M$. The similarity value wl_{st} is calculated based on the row vectors of A using for instance the *slope one* algorithm. A high similarity value wc_{st} indicates that learner s and t may have similar preferences since they have previously purchased a set of common LAs. WL·A gives potential values of the LA for each learner. The element at the lth row and lath column of the resulting matrix aggregates the value of the similarities between learner l and other learners who have purchased learning resource la previously. In words, the more similar other learners to the target learner are, the more likely the target learner

will also be interested in their learning resource because they seem to have the same background.
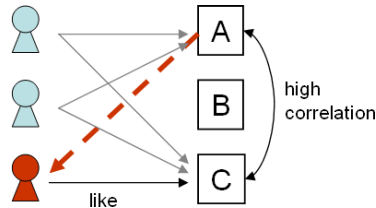
4.2. Item-based CF

Fig. 4. Technical drawing of item-based collaborative filtering algorithm [26]

Item-based techniques correlate the items by mining (similar) ratings and then recommend new, similar items (see Figure 4). The item-based algorithm is therefore different from the user-based algorithm only in that item similarities are computed instead of user similarities. In our case, this algorithm first computes a learning resource similarity matrix $WLA = (wla_{st})$, $s, t = 1, 2,…, N$. Here, the similarity value wp_{st} is calculated based on column vectors of A. A high similarity value wp_{st} indicates that learning resource s and t are similar in the sense that they have been studied by similar learners. $A·WLA$ offers the *likely value* of the learning resources for each learner. Here, the element at the lth row and lath column of the resulting matrix aggregates the values of the similarities between learning resource la and other learning resources previously purchased by learner l. The purpose behind this algorithm is similar: the more similar to the target learning resource the learning resources studied by the target learner are, the more likely the target learner will also be interested in that learning resource.

4.3. Data set

Regarding the gap of available data set for the evaluation of recommender systems for learning and especially for LNs, we decided to use synthesized data sets [27] in the simulation rather than applying a data set that imperfectly matches the properties of a LN. Therefore, we modeled the LAs in the simulation with a fixed number of characteristics and learners which having preferences through their learning goal, study time and competence level for those LAs. For the design of a simulation tool that acts as a first evaluation phase for recommender algorithms in LNs we decided to use synthesized data sets than imperfectly adapted data sets. Furthermore, with the ongoing research in this field we expect that in the future data sets will be available to improve our simulation tool.

5.   Experimental design

For the test of recommender systems in LNs [7] proposed an evaluation framework that combines measures from the learning domain, recommender system field, and the social network analysis to describe the multidimensional effects of such a recommender system. Based on this framework we decided to use *Effectiveness*, *Efficiency*, and *Drop out* as key variables for the learning domain. Further, we selected *Accuracy, Precision, and Recall* as measure for the recommendation algorithms and *Variety* as measure for the connectivity of learners in the LN.

   We are planning to test the following four hypotheses in three consecutive simulation studies with LNs in different sizes, where the control group gets no recommendations; whereas *treatment group A* gets navigation support provided with an item-based CF algorithm, and *treatment group B* gets recommendation support based on a user-based CF algorithm (see Figure 5).

1. The treatment groups will be able to complete more learning activities than the control group (Effectiveness).

2. The treatment groups will complete learning activities in less time, because alignment of learners and learning activities increase the efficiency of the learning process (Efficiency).

3. The treatment groups have a broader variety of learning paths than the control group because the recommender system supports more personalised navigation (Variety).

4. There will be no significant difference between treatment group A and B regarding Effectiveness, Efficiency, Dropout rate, Variety.
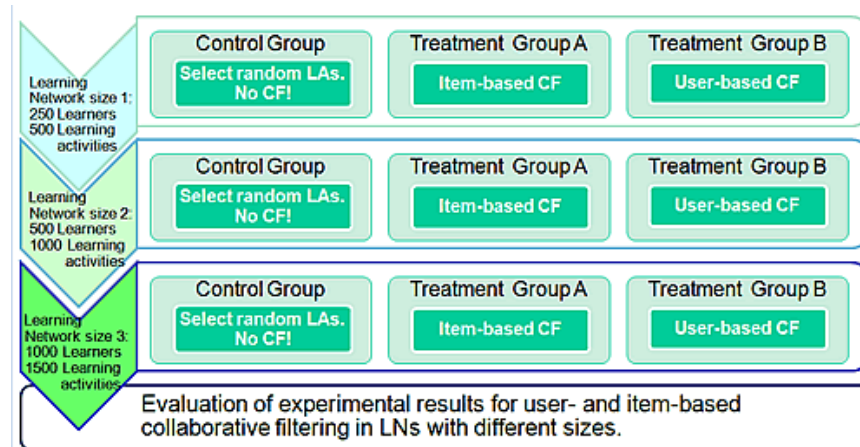

Fig. 5. Experimental design for three consecutive simulation studies.

6.   Conclusion

We shortly presented a theoretical background for research on recommender systems for lifelong learners in informal LNs. Furthermore, we presented a model and flow diagram of the LN simulation tool. Finally, we presented our experimental design the evaluation for recommender systems in informal LNs of different sizes.

   Currently, we are in the phase of developing the simulation tool. After implementing the simulation model we have to approve and validate the simulation to make sure that it is actually doing what we expect it to do. We will validate the simulation tool using extreme situations of LNs where the outcomes are easily predictable. After these steps we can start the proposed experimental study.

   We believe that these kinds of simulation studies can offer insides into the supportive effects of collaborative filtering techniques for LNs. If the results are satisfying we want to test additional algorithms in our simulation tool. To further generalize the results of simulation studies we have to design following up real world experiments.

Acknowledgement

References

1. Longworth, N.: Lifelong learning in action - Transforming education in the 21st century. Kogan Page, London (2003)
2. Brockett, R.G., Hiemstra, R.: Self-direction in adult learning: perspectives on theory, research and practice. Routledge, London (1991)
3. Surowiecki, J.: The wisdom of crowds: why the many are smarter than the few and how collective wisdom shapes business, economies, societies, and nations. Anchor New York (2005)
4. Koper, R., Tattersall, C.: New directions for lifelong learning using network technologies. British Journal of Educational Technology 35 (2004) 689-700
5. Brusilovsky, P., Henze, N.: Open Corpus Adaptive Educational Hypermedia. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.): The Adaptive Web: Methods and Strategies of Web Personalization. , Vol. 4321. Springer, Berlin Heidelberg New York (2007) 671-696
6. Herlocker, J.L., Konstan, J.A., Riedl, J.: Explaining collaborative filtering recommendations. Proceedings of the 2000 ACM conference on Computer supported cooperative work (2000) 241-250
7. Drachsler, H., Hummel, H., Koper, E.J.R.: Applying recommender systems to lifelong learning networks: requirements, suitable techniques and their evaluation. Journal of Digital Information (submitted)

8. Gilbert, N., Troitzsch, G.: Simulation for the Social Scientist, Vol. Second Edition. Open University Press, Buckingham (2005)
9. Koper, R.: Increasing Learner Retention in a Simulated Learning Network using Indirect Social Interaction. Journal of Artificial Societies and Social Simulation 8 (2005) 18
10. Janssen, J., Tattersall, C., Waterink, W., Van den Berg, B., Van Es, R., Bolman, C., Koper, E.J.R.: Self-organising navigational support in lifelong learning: how predecessors can lead the way. Computers & Education 49 (2005) 781-793
11. Drachsler, H., Hummel, H., van den Berg, B., Eshuis, J., Berlanga, A., Nadolski, R., Waterink, W., Boers, N., Koper, R.: Effects of the ISIS Recommender System for navigation support in self-organised Learning Networks. In: Kalz, M., Koper, R., Hornung-Prähauser , V., Luckmann, M. (eds.): 1st Workshop on Technology Support for Self-Organized Learners (TSSOL08) in conjunction with 4th Edumedia Conference 2008 Self-organised learning in the interactive Web – Changing learning culture?. CEUR Workshop Proceedings, Salzburg, Austria (2008) 106-124
12. 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) (accepted) 26
13. Balabanovic, M., Shoham, Y.: Fab: content-based, collaborative recommendation. Communications of the ACM 40 (1997) 66-72
14. Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D., Sartin, M.: Combining content-based and collaborative filters in an online newspaper. ACM SIGIR Workshop on Recommender Systems: Algorithms and Evaluation., Berkeley, CA. (1999)
15. Good, N., Schafer, J.B., Konstan, J.A., Borchers, A., Sarwar, B., Herlocker, J., Riedl, J.: Combining collaborative filtering with personal agents for better recommendations. Proceedings of AAAI 99 (1999) 439-446
16. Melville, P., Mooney, R.J., Nagarajan, R.: Content-boosted collaborative filtering for improved recommendations. 18th National Conference on Artificial Intelligence, Edmonton, Alberta, Canada (2002) 187-192
17. Pazzani, M.J.: A framework for collaborative, content-based and demographic filtering. Artificial Intelligence Review 13 (1999) 393-408
18. Soboro, I.M., Nicholas, C.K.: Combining content and collaboration in text filtering. In: Joachims, T. (ed.): the IJCAI Workshop on Machine Learning in Information Filtering, Vol. 99, Stockholm (2000) 86-91
19. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Analysis of recommendation algorithms for e-commerce. Proceedings of the 2nd ACM conference on Electronic commerce (2000) 158-167
20. Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.: Evaluating Collaborative Filtering Recommender Systems. ACM Transactions on Information Systems **22** (2004) 5-53
21. Drachsler, H., Hummel, H., Koper, R.: Personal recommender systems for learners in lifelong learning: requirements, techniques and model. International Journal of Learning Technology 3 (2008) 404 - 423

22. Vygotsky, L.S.: Mind in Society: The Development of Higher Psychological Processes. Harvard University Press (1978)
23. Ryan, R.M., Deci, E.L.: Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. Vol. 55 (2000) 68-78
24. Walker, A., Recker, M.M., Lawless, K., Wiley, D.: Collaborative Information Filtering: A Review and an Educational Application. International Journal of Artificial Intelligence in Education 14 (2004) 3-28
25. Bolman, C., Tattersall, C., Waterink, W., Janssen, J., van den Berg, B., van Es, R., Koper, R.: Learners'evaluation of a navigation support tool in distance education. Journal of Computer Assisted Learning 23 (2007) 384-392
26. Kim, J.: What is a recommender system? In: Kim, J. (ed.): Recommenders06.com. mystrands.com, Bilbao (2006) 1-21
27. Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R., Riedl, J.: GroupLens: applying collaborative filtering to Usenet news. Vol. 40. ACM New York, NY, USA (1997) 77-87