User-centric Evaluation of Recommender Systems in Social Learning Platforms
Accuracy is Just the Tip of the Iceberg

Fazeli, Soude; Drachsler, Hendrik; Bitter-Rijpkema, Marlies; Brouns, Francis; van der Vegt, Wim; Sloep, Peter B.

DOI
10.1109/TLT.2017.2732349

Publication date
2018

Document Version
Final published version

Published in
IEEE Transactions on Learning Technologies

Citation (APA)
Fazeli, S., Drachsler, H., Bitter-Rijpkema, M., Brouns, F., van der Vegt, W., & Sloep, P. B. (2018). User-centric Evaluation of Recommender Systems in Social Learning Platforms: Accuracy is Just the Tip of the Iceberg. *IEEE Transactions on Learning Technologies*, *11*(3), 294 - 306. https://doi.org/10.1109/TLT.2017.2732349

User-Centric Evaluation of Recommender Systems in Social Learning Platforms: Accuracy is Just the Tip of the Iceberg

Soude Fazeli, Hendrik Drachsler, Marlies Bitter-Rijpkema, Francis Brouns, Wim van der Vegt, and Peter B. Sloep

Abstract—Recommender systems provide users with content they might be interested in. Conventionally, recommender systems are evaluated mostly by using prediction accuracy metrics only. But, the ultimate goal of a recommender system is to increase user satisfaction. Therefore, evaluations that measure user satisfaction should also be performed before deploying a recommender system in a real target environment. Such evaluations are laborious and complicated compared to the traditional, data-centric evaluations, though. In this study, we carried out a user-centric evaluation of state-of-the-art recommender systems as well as a graph-based approach in the ecologically valid setting of an authentic social learning platform. We also conducted a data-centric evaluation on the same data to investigate the added value of user-centric evaluations and how user satisfaction of a recommender system is related to its performance in terms of accuracy metrics. Our findings suggest that user-centric evaluation results are not necessarily in line with data-centric evaluation results. We conclude that the traditional evaluation of recommender systems in terms of prediction accuracy only does not suffice to judge performance of recommender systems on the user side. Moreover, the user-centric evaluation provides valuable insights in how candidate algorithms perform on each of the five quality metrics for recommendations: usefulness, accuracy, novelty, diversity, and serendipity.

Index Terms—Recommender systems, evaluation, social, learning, accuracy, performance

1 Introduction

Recommender systems provide a user with the content she or he might be interested in. They have become increasingly popular because of their successful applications in the e-commerce field, such as with Amazon and eBay. Recommender systems have been introduced in the educational domain as a practical solution to help users find suitable content that can support their learning process [1], [2], [3]. Traditionally, recommender systems have been evaluated according to accuracy metrics in the Information Retrieval area. However, such evaluations do not answer the question whether the users are actually satisfied with the recommendations as indicated by the accuracy metrics. Recently, researchers have realized that the goal of a recommender system goes beyond the accuracy metrics [4], [5]. This has prompted two major changes in the field of recommender systems. The first change, indicated by McNee et al. [5], is that "being accurate is not enough". These authors also emphasized that researchers should "study recommenders from a user-centric perspective to make them not only accurate and helpful, but also a pleasure to use" [5]. The second change has been introduced as "a broadening of the scope of research regarding the system aspects to investigate beyond just the algorithm of the recommender" [4], [6]. Following this, McNee et al. suggest researchers to also study the aspects of "Human-Recommender Interaction" [7]. Martin [8] claimed in his keynote to the ACM RecSys 2009 conference that around 50 percent of a recommender's commercial success goes to the aspects of "Human-Recommender Interaction" while the algorithm matters for 5 percent only [8].

The importance of the user perspective has been realized even more in the educational domain [1], [9], [10]. Indeed, the main goal of the educational recommender systems extends well beyond accurate predictions and should also take into account quality metrics such as usefulness, novelty, or diversity of the recommendations [10].

Although the importance of user-centric evaluations has become quite clear and vital, the majority of recommender system studies still solely report the traditional, data-centric evaluation results. Many of them are based on some implicit feedback such as Click Through Rate (CTR) [11], [12], which hardly reflect users' satisfaction and the perceived usefulness of the recommendations made for them. However, traditional offline user-centric evaluations, such as those based on CTR,

- *S. Fazeli is with the Delft University of Technology and the Open University of the Netherlands, Mekelweg 4, 2628 CD, Delft, The Netherlands. E-mail: s.fazeli@gmail.com.*
- *H. Drachsler is with the Goethe University Frankfurt, the German Institute of International Educational Research (DIPF), and the Open University of the Netherlands, Heerlen 6419 AT, The Netherlands. E-mail: contact@drachsler.de.*
- *M. Bitter-Rijpkema, F. Brouns, W. van der Vegt, and P.B. Sloep are with the Welten Institute, Open University of the Netherlands, Heerlen 6419 AT, The Netherlands. E-mail: {marlies.bitter, francis.brouns, wim.vandervegt, peter.sloep}@ou.nl.*

Manuscript received 7 Mar. 2016; revised 29 May 2017; accepted 10 July 2017. Date of publication 27 July 2017; date of current version 20 Sept. 2018. (Corresponding author: Soude Fazeli.)
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TLT.2017.2732349

are more straightforward to conduct than user-centric evaluations based on explicit questionnaires. There are several reasons that make user-centric evaluations complicated to carry out:

1) They can easily fail due to the lack of a sufficient numbers of participants
2) It is also quite tricky to design an experimental protocol such that it attracts users instead of detracting them. The users' task should be defined clearly and simply, helping users to spend a fair amount of time on the task and also making sure not to be misunderstood.
3) Setting up a test bed as an experimental environment is a time-consuming and delicate job.
4) User-centric evaluations can take up to several months and they are quite vulnerable to the availability and loading speed of the experimental environment (often a social platform as in this study); continuous availability of the participants is also a concern.

Moreover, many user-centric evaluations are conducted using crowdsourcing. Although that is a valid approach, it has its limitations [10], [13]. In crowdsourcing, tasks, reliability and accuracy of the collected feedback data is sometimes questionable since there are of course differences between "cheap labor" workers and expensive experts [14].

In this study, we want to investigate what is the added value of user-centric evaluations precisely because of the complexity of carrying them out. So our main research question is:

RQ: How is user satisfaction with recommender systems in social learning platforms related to the performance of such systems measured in terms of their accuracy?

We conduct both a traditional, data-centric evaluation and a user-centric evaluation. Such an evaluation aims to answer our research question by using a proposed graph-based approach and also state-of-the-art recommender algorithms within an authentic social learning platform developed by the eContentPlus Open Discovery Space (ODS) project (http://opendiscoveryspace.eu). By the term 'social learning platform', we refer to those platforms that combine traditional learning management systems (LMS) with commercial social networks such as Facebook to provide easy access, sharing, bookmarking, content creation, etc. Beside the forums and chat communities often provided in standardLMSs, social learning platforms let users establish better connections and thus improve their networks of peers. The main contribution of this study is evaluating state-of-the-art recommender systems in the ecologically valid setting of a real learning platform, following a standard evaluation methodology for recommender systems in educational domain [1], [3]. To best of our knowledge, there has been no work in the educational domain that studied all the five metrics we evaluated in our user study (usefulness, accuracy, diversity, novelty, serendipity).

The rest of the paper is structured as follows: In Section 2, we describe the experimental method used including the algorithms, the data, and the evaluation settings. Section 3 presents the experimental results including results of both traditional evaluation and user-centric evaluation. Section 4 discusses the extent to which the results answer the research question defined in this study, and finally, draws conclusions.

2 Experimental Method

We ran two sets of evaluations: 1. A conventional data-centric evaluation for comparing the performance of recommender systems based on traditional accuracy metrics, and 2. a user-centric evaluation as an online study to ask actual users for their feedback on the recommendations made for them.

Fig. 1 presents an overview of our experimental study portraying the two sets of evaluations. The figure shows how the evaluation methods work independently of each other and yet are related. Moreover, Fig. 1 shows the input, the procedures, and finally, the expected outcomes for each of the evaluation methods. We will provide detailed descriptions for each of these evaluation methods in the following sections.

Here, we first provide a description of the data used. Second, we give an overview of the recommender algorithms chosen for this study. Finally, we explain the settings of both evaluation methods (data-centric and user-centric).

2.1 Data

The data used in this study comes from the Open Discovery Space (ODS) platform. According to the official website, "[The] Open Discovery Space [project] addresses the challenge of modernizing school education by engaging teachers, students, parents and policymakers in a first of its kind effort to create a pan-European eLearning environment to promote more flexible and creative ways of learning by improving the way educational content is produced, accessed and used" (http://opendiscoveryspace.eu). In practice, the ODS platform is a large-scale Open Educational Resources (OER) platform, where mainly teachers can look for OERs and upload their own resources. Within the platform the teachers but also other ODS stakeholders (e.g., educational designers, educational advisors, and content experts, etc.) can meet, create groups, communicate, and work online with social media functions such as downloading, commenting, rating, tagging, and discussing OERs.

The ODS data therefore, collected through the platform, contains social data of users such as ratings, tags, reviews, etc. on learning resources, communities, groups, etc. The ODS data complies with the Context Automated Metadata (CAM) format [15], which provides a standard metadata specification for collecting and storing social data. A CAM schema aims to store whatever has attracted users' attention while the users are working with the platform. It also stores users' interaction with the platform such as rating, tagging, etc. A CAM schema records an event and its details whenever a user performs an action within a platform. The metadata stored in the CAM format describe all types of users' feedback and, therefore, can be further converted to the input data required for making recommendations for the users.

For the data-centric evaluation part of this study, we used the ODS dataset containing interaction data (9,117 events) of 2,567 users with 3,392 objects. Since the data-centric evaluation is carried out as an offline study, we used a portion of the interactions data for training the model and the remainder for validation. For the user-centric evaluation, we used the complete dataset for training as this scenario includes no offline evaluation. It should be noted that the data is too sparse in terms of user transactions (degree of sparsity=99.86%) to make recommendations with

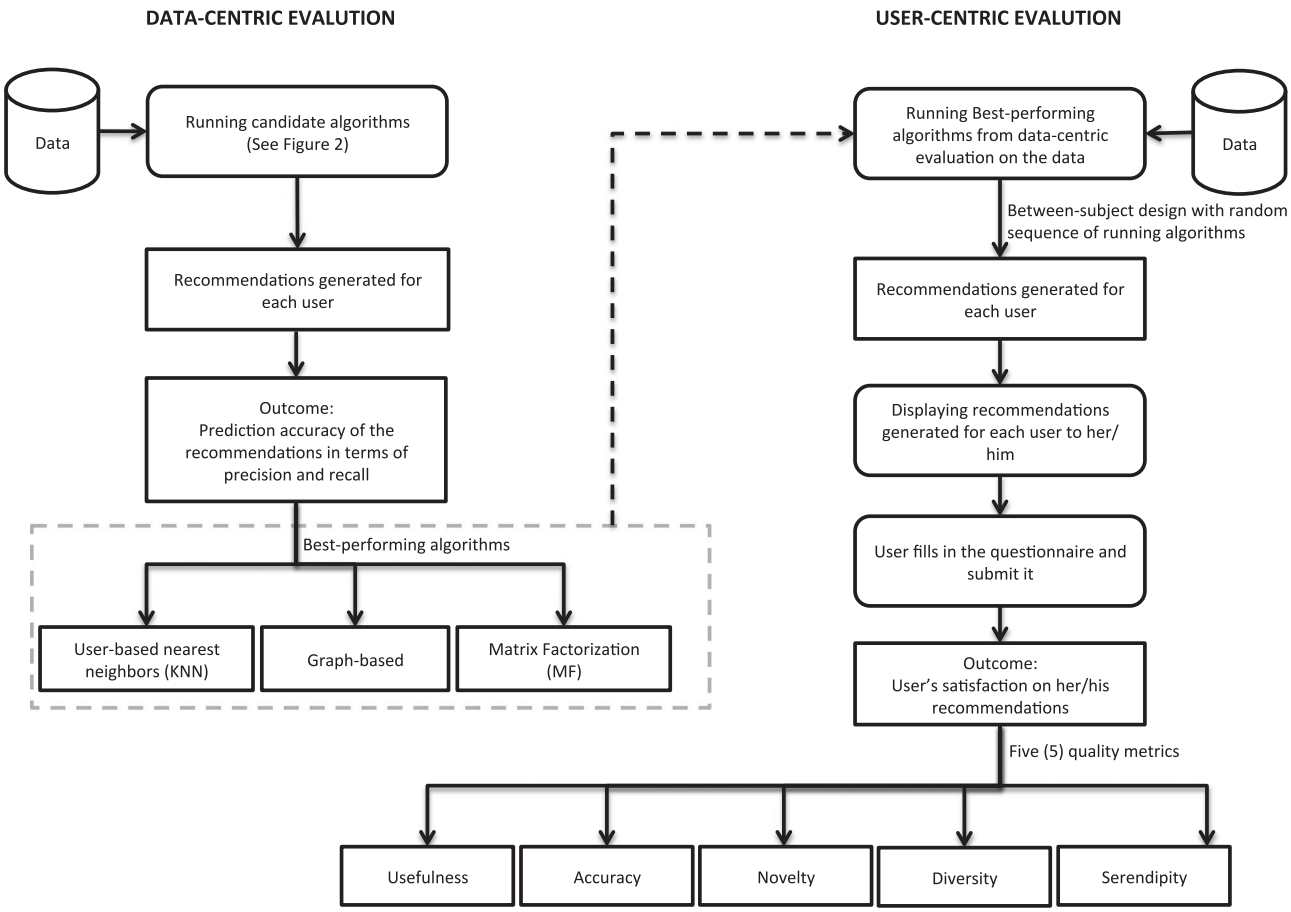


Fig. 1. Overview of the experimental study presented in this paper.

classical recommender systems. Although the dataset is rather small, it realistically represents the current ODS platform. Sparsity often occurs in educational settings and requires specific adjustments to the recommendation approach as shown by [16]. We therefore took sparsity into account as one of the data properties when trying to select the most appropriate algorithms for ODS. The data span the time period from May 2013 until October 2015.

2.2 Algorithms

The first step in developing a recommender system is to find out with what kind of input data to fuel the recommender engine. As already mentioned, the items in the ODS platform are learning resources, communities, groups, and discussion posts. The user activities in the ODS platform mainly consist of implicit user feedback coming from tracking data, such as viewing, bookmarking, downloading a resource or joining a community. Therefore, Collaborative Filtering (CF) recommenders can be applied. CF methods make recommendations for a target user based on other users' opinions and interests [17], [18]. Content-based methods should be used when there is no user rating information (5-star, binary, unary) available. However, as is also indicated in recommender systems studies [19], "even if very few ratings are available, simple rating-based predictors outperform purely metadata-based ones". This is likely due to the large difference between the item descriptions and the items themselves. Note that users rate items, not their descriptions. In general, the CF algorithms are categorized according to their *type* and *technique*: *Type* refers to model-based and memory-based algorithms and *technique* refers to user-based and item-based algorithms.

In the rest of this section, we are going to describe how we select a set of candidate algorithms for our experimental study according to the different *types* and *techniques* of the CF recommender systems. In this study, we try to make use of the algorithms from all the CF's categories as well as a graph-based method we proposed in our previous work [16]. Fig. 2 presents an overview of the candidate algorithms for this study.

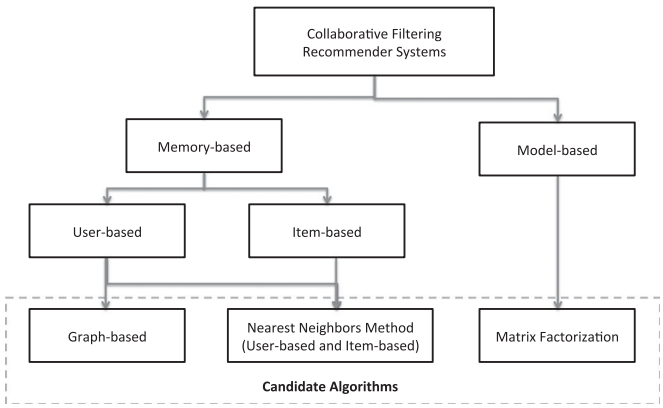


Fig. 2. Candidate algorithms for data-centric evaluation.

2.2.1 Memory-Based Recommender Systems

Most of the CF algorithms are based on k-nearest-neighbor (kNN) methods (k being the size of the neighborhood). They have proven to be quite successful[20]. kNN tries to find like-minded users and introduces them as the nearest neighbors of a target user for whom recommendations are generated. The kNN algorithms create a graph of users in which nodes are users and the edges are similarity relations between them. Depending on whether the data includes explicit user feedback (e.g., 5-star ratings) or implicit user feedback (e.g., views, downloads, clicks, etc.), different similarity measures are appropriate. The data used in this study provides implicit user feedback in the form of (userID,itemID) tuples, with, "item" referring to learning objects, communities, groups, etc. in the ODS platform. This kind of data is also known as "positive feedback only" since they present only users' interests in items where there is no negative feedback expressed by users on items [21]. Therefore, some of the similarity measures such as Pearson correlation are not suitable because they require explicit user feedback, i.e., in forms of 5-star ratings explicitly expressed by users. As one of the popular similarity measures, we used the Jaccard coefficient since the data includes implicit user feedback in binary format [22]. In this study, we use both user-based and item-based CF algorithms since we make use of users' interactions and activities. User-based algorithms try to find patterns of similarity between users in order to make recommendations; item-based algorithms follow the same process but are based on similarity between items.

2.2.2 A Graph-Based Recommender System

Although kNN methods are quite popular in the recommender systems area, they have two shortcomings. First, they usually do not work well when the user feedback data is sparse, which is often the case in the educational domain [1]. Second, they are only limited to k neighbors for each user. Thus two users who have not shown an interest in a common set of items cannot be connected, even though they might be a good source of information for each other. This affects the process of knowledge sharing and peer collaborations in online learning platforms. However, that platforms such as ODS have been set up exactly to foster peer collaboration, learning from each other, and other activities that promote the shared construction of knowledge. To address the sparsity issue and the restriction to k neighbors only, we propose to use a graph-based approach [16], [23]. Such an approach extends and improves the kNN's process of finding neighbors, by invoking graph search algorithms. The graph-based approach first forms a graph in which nodes are users and edges are similarity relations between users. Then, it collects recommendations for a target user by 'walking' through the target user's neighbors. We assign a Social Index (S-index) to each user, which is inspired by the H-index that is an indicator of publications of an author. The H-index combines information on the number of publications of some author with the number of citations [24]. Similarly, as we defined in[16], the S-index of a user u shows not only how many times user u has been selected as a neighbor, but also how much the user u contributed to interactions on items in common with her neighbors. The S-index is calculated using the algorithm 1 presented in our previous work [16]. In this study, the S-index ranges from 1 to 100 and the similarityScore between two users ranges from 0 to 1. The similarityScore between two users shows how similar the two users rated the same items. It is computed using Jaccard coefficient since the users data includes implicit user feedback in binary format [22]. We use the S-index to extend and improve the process of finding like-minded users (neighborhoods). It is used for sorting the list of raters of a particular item. Such items' raters list can help us to discover new neighbors for a user, who can be a good source of information for a target user but have been excluded when applying the traditional nearest neighbors method (due to the constraint of sticking to the k neighbors only). We formalized this procedure as follows [16]:

Algorithm 1. Computing S-Index for User u

```
upon event (COMPUTE S-INDEX| u, NeighborsList)
  SortedNeighborsList ← SortDescendingBySimilarityScore(NeighborsList);
  FinalNeighborsList ← Normalize(SortedNeighborsList,MaximumSindex);
  Sindex ← 0;
  for ( similarityScore(u,n); n in FinalNeighborsList) do
    if Sindex  <= similairtyScore then
      Sindex= Sindex+1;
    else
      Break;
    end if
end for
updateSindex(Sindex);
end event
```

```
G(V,E) = CreateSocialGraph(); // V contains users
                              // E contains similarity relations between users
for all u∈V do
  ComputeSindex(u, N); // N contains users who have user u as their neighbor
  G(V,E') ← BFS(u, G(V,E));     // E ⊂ E' where E' contains:
                                // 1. explicit similarity relations (u,n) ∈ E and
                                // 2. new inferred relations (u, n')
  TopItems ← CollectRecommendations(u, G(V, E'));
  UpdateSindex(u,N);       // N' contains new neighbors found
  UpdateSocialGraph();
end for
```

In the graph-based approach, we provide dynamic neighborhoods beyond k for each target user depending on the new neighbors the graph-based approach helps us to infer [16]. We make use of a modified Breadth First Search (BFS) graph search algorithm [25] to traverse the created user graphs using S-index and items raters lists. We chose BFS among the well-known walking algorithms to first consider the direct neighbors when collecting recommendations in the created user graph. While walking through the neighbors, there are still two conditions required to be met: first, the similarity score between the two neighbors should not fall under a certain threshold (0.1 in this study), and second, we limit the number of edges (traversal length) between the two neighbors to satisfy the bandwidth and performance issues (the traversal length is set to three (3) in this study, which indicates that we are allowed to walk through three (3) neighbors in-between). For more details, please refer to our previous work[16]. The graph-based approach is

memory-based and user-based. Approaches to improve performance of recommenders by using graph-walking algorithms do exist already and report positive effects in different domains [23], [26], [27], [28]. However, almost all these approaches use the data regarding either social relations between users or inter-user trust relations; these are not available for the datasets used in this study. Indeed, we use the graph-based approach with the aim of supporting the target users of social learning platforms to identify their potentially interesting and novel neighbors.

2.2.3 Model-Based Recommender Systems

In our experimental study, beside the memory-based approaches, we also want to evaluate model-based methods as they also represent one of the two main types of CF algorithm. Model-based methods create models of users' preferences using probabilistic approaches such as neural networks, Bayesian networks, and algebraic approaches such as those using eigenvectors. They are known for their fast performance as they create users' preferences models offline but they need a full set of users' preferences to develop a user model. Moreover, model-based methods often prove to be costly in terms of required resources and maintenance efforts. In this study, we need to choose model-based CFs that can deal with implicit feedback. Rendle et al. [29] applied their Bayesian Personalized Ranking (BPR) to the state-of-the-art matrix factorization models to improve the learning process in the Bayesian model used (BPRMF). Our data is also implicit feedback so the BPRMF seems to be an appropriate model-based candidate for our experimental study since it can work well with this kind of data.

2.3 Data-Centric Evaluation

We ran a data-centric evaluation to assess performance of the candidate recommender algorithms in terms of accuracy metrics in the Information Retrieval area. Within this conventional type of studies, there is no direct interaction with the actual users. Fig. 1 describes the data-centric evaluation. Furthermore, Fig. 2 complements this description by presenting an overview of the candidate algorithms involved in the data-centric evaluation. The algorithms are measured according to, precision and recall to measure the accuracy of the recommendations generated [30]. Precision is defined as the percentage of recommended items that are relevant to the user (i.e., ratio of the number of items recommended that were relevant to the total number of recommended items). Recall shows the probability that a relevant item is recommended (i.e., the number of items recommended that were relevant divided by the total number of relevant items in the entire test set). An item is considered relevant if a target user already accessed it. Both precision and recall range from 0 to 1. In this study, 80 percent of the data was randomly selected and assigned to the training set and 20 percent was used as the test set. These metrics and settings are commonly used for empirical studies on recommender systems [30].

2.4 User-Centric Evaluation

We conducted a user-centric evaluation to measure the perceived quality of the recommendations made for ODS users.

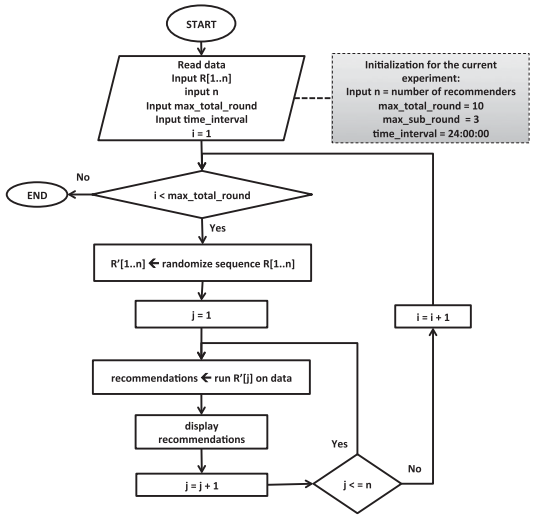


Fig. 3. Design of the user-centric evaluation with random sequence of running algorithms.

Fig. 1 presents an overview of the user-centric evaluation. It also shows that the best-performing algorithms from the data-centric evaluation are used in the user-centric evaluation for making recommendations. The recommendations have been made for each user based on her/his interactions data within the platform. The link to the questionnaire was only enabled when a user had already received recommendations. If a user had not received any recommendations yet, we showed a message "There is no recommendations for you today". This way, the users were able to explore their recommendations first and later to respond to the questionnaire based on their experience with their recommendations.

In the rest of this section, we first describe the design of the user-centric evaluation and then we present the questionnaire we used in our user study.

2.4.1 Design

In principle, two types of research designs are possible, 1. a repeated measures design in which all users are tested repeatedly, once for each recommender; and 2. a design in which each user is exposed to one recommender only, once and only once. In the first case, users are tested repeatedly by asking how they like the current recommendations from one of the recommender algorithms (within-subjects design); in the second case users act as each others' replications (between-subjects design) and rate only the recommendations of one recommender algorithm. Since it is impossible to guarantee that all users are indeed exposed to all recommenders and a repeated measures design with missing values is hard to analyze, only the second option is feasible. Besides, there is little a priori reason to expect that users have inherently different levels of responding (if that were the case, a repeated measures design would have been preferable as it removes variation due to those differences). Fig. 3 shows the method used in the user-centric evaluation. We have a set of candidate recommender systems R1...Rn where n is number of candidate recommender algorithms. In this study, n equals three (3) since we have three recommender algorithms from the main categorization of the CF recommender algorithms: 1. a memory-based CF (user-

based nearest neighbors method), 2. a graph-based CF, and 3. a model-based CF (matrix factorization).

These algorithms are the ones which best performed in the data-centric evaluation, as shown in Section 3.1, and thus, we run them on the data to generate recommendations for each user. Users will typically enter the ODS platform, be confronted with a recommendation list (size=5) made either by R1, R2, or R3. They then are requested to answer the questionnaire; the questionnaire becomes available by clicking on a link provided to them. This means that there may be sequence effects, since participants enter in the experiment one after the other. To avoid such effects, treatments (types of recommendations) were assigned in a random order over time.

Since it is technically not feasible to administer a randomly drawn treatment per user recommendation event, treatments were administered in blocks of fixed time periods (randomized block design): R1-R2-R3, then R3-R1-R2, then R2-R1-R3. If there is a sequential effect, it will thus be balanced out over time. Since any one of the recommenders was active for all ODS users during the fixed time period it was tested, including those users who had already participated in the experiment, the questionnaire link was hidden from the latter to prevent them from participating more than once. In summary, each ODS user has been randomly allocated to one of the three recommender algorithms and evaluated the recommendations produced for him/her in the satisfaction questionnaire.

2.4.2 Questionnaire

The questionnaire was designed to reflect how actual users perceive and appreciate the recommendations they receive, taking into account important aspects in user perception when running recommender systems' user studies [6], [31]. We asked the participants to answer six short questions by expressing their level of agreement with each of the questions. Agreement ranges from completely disagree (1) to completely agree (5). The questionnaire contains six statements: five questions regarding quality of the recommendations and one regarding the language of the recommendations. This is a rather low number, but we feared that the response rate would drop dramatically if we added more items: a recommendation is something one naturally inspects immediately not after answering a lengthy questionnaire first. The description of the quality metrics were embedded in each question itself. Question 6 is an open question we added at the end of the questionnaire. Through it a user can provide general comments. The statements were:

1. The recommendations are relevant to my activities (Accuracy).
2. The recommendations provide me with novel information (Novelty).
3. The recommendations differ significantly from each other (Diversity).
4. The recommendations are useful for me (Usefulness).
5. The recommendations are surprising to me (Serendipity).
6. I am satisfied with the language of the recommendations.

For selecting the five quality metrics, we followed the ResQue framework presented by Pu et al. [32] that has been also introduced and used specifically in learning domain [10]. The framework provides a unified method for user-centric evaluations. However, making use of the whole framework can be very time consuming for participants since it includes many metrics. Therefore, we only focus on five important metrics that have been identified in the literature on recommender systems user studies as indicators of users satisfaction on the recommendations made for them [4], [10], [32], [33]. By tending towards simplicity we seek to guarantee responsiveness. Put differently, we struck a balance between getting reliable data and making sure the users would fill out the questionnaire at all. In a field study that champions ecological validity, this is always a major concern; more so than in controlled lab situations.

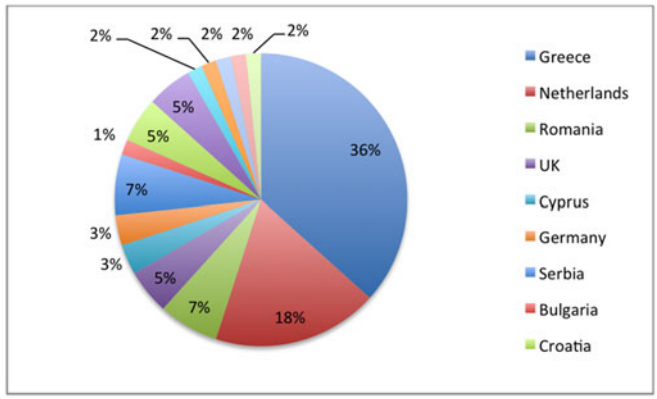


Fig. 4. Distribution of the participants over countries.

For translating these five metrics to the learning domain, we can take advantage of Vygotsky's zone of proximal development [34] and also a comprehensive survey on evaluating recommender systems in learning by Erdt et al. [10]: The 'accuracy' metric shows how much a recommended resource for a learner is relevant to the learners activities and history [35]. For 'accuracy', one could demand that the recommended resource be within the zone of proximal development for the learners in a way that the distance in knowledge can be bridgeable following Vygotzkys idea [34], [36]. However, Vygotzkys model also follows the pedagogical rule that 'recommended learning resource should have a level that is a little bit above a learner's current competence level' [34], [35]. In other word, it is important to expose learners to new and different viewpoints and resources to avoid getting them isolated in their own filter bubble [37]. This is why, recommender systems in learning should also provide a learner with novel resources (Novelty), or with different types of resources (Diversity) [10]. Furthermore, recommendations from a totally unexpected domain, discipline, etc. can surprise a learner in a positive way (Serendipity) and thus, can challenge the learner to think out her/his box (current knowledge) [10]. In addition to these four metrics, learners' perceived 'usefulness' of the recommendations has proved to be an important metric and has been measured in quite few works on recommender systems in learning [10], [38], [39].

In this study, we had sixty participants in total from fifteen European countries: Greece, the Netherlands, Romania, the UK, Cyprus, Germany, Serbia, Bulgaria, Croatia, Estonia, Ireland, Lithuania, Poland, Portugal, and Spain. Fig. 4 shows the distribution of the participants over these

countries. In total, 48 percent of the participants were female and 52 percent were male. The participants were both primary and secondary school teachers, educational designers, educational advisors and content experts. The participants were randomly provided with recommendations based on three candidate algorithms: 1. a memory-based CF (user-based nearest neighbors method), 2. a graph-based CF, and 3. a model-based CF (matrix factorization). We managed to obtain the same number of participants for the three algorithms, twenty for each. For user-centric evaluations in the recommender systems area, it has been claimed that "at least twenty (20) users" per condition is adequate to make a user-centric evaluation statistically sound [6]. The user-centric evaluation ran for five months (June 2015 to October 2015). The ODS users were carrying out the activities that users of social learning platforms typically do. To maintain ecological validity our experiment was not flagged as such. We only activated an option for the participants to join the questionnaire once our experiment started and we deactivated it when we closed the study. It took about five months to collect a sufficient number of responses to the questionnaire.

3 Results

We first provide results of a traditional data-centric evaluation on the ODS data and then we present the user-centric evaluation results.

3.1 Data-Centric Evaluation Results

The results of the offline data-centric evaluation on ODS data provide insights into the prediction accuracy of the recommendations made for ODS users. We conducted the offline evaluation in two steps, according to the types of the CFs (memory-based or model-based) (see Fig. 2):

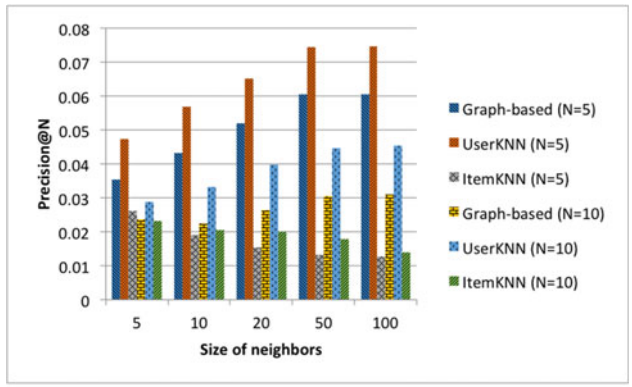
Step 1: Evaluating three candidate memory-based CFs: the user-based and item-based k-Nearest Neighbors methods (UserKNN and ItemKNN, respectively), and the user-based graph-based approach.

Step 2: Comparing performance of the candidate model-based CF that is a matrix factorization method (BPRMF) with the best-performing memory-based CFs from step 1. For more details regarding these candidate algorithms and the reasons behind choosing them, we refer to Section 2.2 (Algorithms).

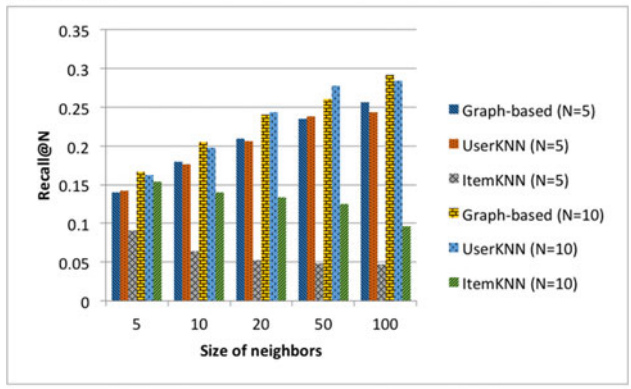
Fig. 5 shows results of step 1 that present precision and recall of memory-based CFs. For each memory-based CF algorithm, we evaluated five different sizes of neighborhoods (k=5,10,20,50,100). The horizontal axis (x) of both Figs. 5a and 5b indicate different sizes of neighborhood (k). The vertical axis (y) in Figs. 5a and 5b represent the values of precision and recall, respectively, at different cut-off values N (@N).

As Fig. 5b shows, while the precision values of user-based CFs (UserKNN and Graph-based) improve by increasing size of neighborhood (k), precision of item-based KNN (ItemKNN) declines by increasing the size of k. However, increasing the cut-off value (N) of precision from N=5 to N=10 improves the precision of ItemKNN, whereas precision of the user-based CFs (UserKNN and graph-based) decreases while N increases.

In general, UserKNN's precision@5 provides the highest values for precision from 0.047 (k=5) to 0.074 (k=100). The graph-based CF comes second with precision@5 values increasing from 0.035 (k=5) to 0.060 (k=100). The highest value of precision for ItemKNN's is 0.026 (precision@5; k=5), which declines to 0.013 (precision@5; k=10).



(a) Precision@N of memory-based CFs for different sizes of neighbourhoods



(b) Recall@N of memory-based CFs for different sizes of neighbourhoods

Fig. 5. Comparison of memory-based CFs. Precision and recall scores (range: 0-1) for different sizes of neighborhoods and for two cut-off points, N=5 and N=10.

Similar to precision results, Fig. 4b shows for recall that both user-based CFs (Graph-based and UserKNN) perform better than the item-based one (ItemKNN). In general, recall values for all algorithms increase when N increases from N=5 to N=10, which is expected in offline recommender system studies [30]. The recall of the UserKNN and the graph-based CF changes for different neighborhood sizes: for N=10, UserKNN's recall increases from 0.162 (k=5) to 0.283 (k=100) and the graph-based CF's recall increases from 0.166 (k=5) to 0.291 (k=100). The recall@10 for the ItemKNN goes from 0.1533 (k=5) to 0.0963 (k=10).

For the memory-based CFs, we set the size of neighborhood (k) to 10. Although performance of the algorithms improves by increasing k in terms of accuracy metrics, we had to keep the neighborhood size fairly small for reducing memory usage and also for making the recommendations generation task sufficiently fast for the user online evaluation. In a summary from step 1, we choose the graph-based and UserKNN CFs as the memory-based candidate CFs to be compared to the model-based matrix factorization method in the second step of the data-centric evaluation.

Fig. 6 presents the results of step 2 as a final comparison of different best-performing memory-based CFs (graph-based and UserKNN) with the candidate model-based

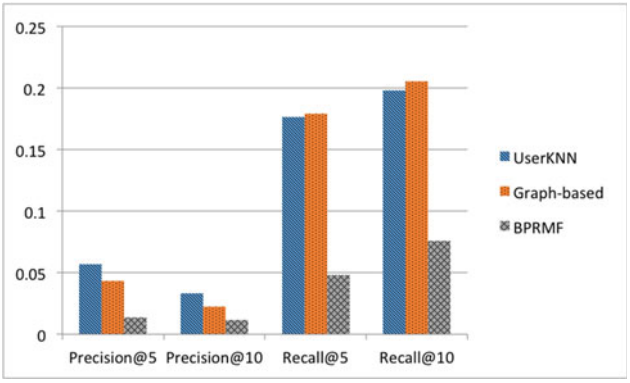


Fig. 6. Final comparison of the candidate CFs. Precision and recall scores (range: 0-1) for two cut-off points, N=5 and N=10.

matrix factorization method (BPRMF). For the model-based algorithm, we tried three different numbers of latent factors (3, 5, and 8). Among these three latent factors, BPRMF with f=8 achieved the best values for both precision and recall; consequently, we chose a number of latent factors equal to 8 for this final comparison. We set the learning rate (alpha) at 0.05 and the regularization parameter for user factors at 0.0025. The parameters have been tuned by using a validation set. The horizontal axis (x) in Fig. 5 indicates the performance metrics in terms of precision and recall at two different cut-offs (N=5 and N=10). The vertical axis (y) shows values of precision@5, precision@10, recall@5, and recall@10 for different algorithms.

As Fig. 6 shows, the user-based CFs (UserKNN and graph-based) outperform the matrix factorization method (BPRMF). The highest precision of BPRMF is precision@5=0.0135 whereas the lowest precision value for the user-based CFs is 0.0331 for the UserKNN's precision@10. For recall, the highest value for BPRMF (recall@10=0.0754) is still much smaller than the lowest recall@10 value for the memory-based CFs (UserKNN' s recall@5=0.1762).

In summary, the data-centric evaluation used in this study shows that the user-based CFs outperform the model-based CFs. According to conventional recommender systems evaluations, for this reason data scientists would use the user-based CFs algorithms in the live system. Since we want to investigate whether the user satisfaction results are in line with the data-centric evaluation results, we apply the three candidate algorithms that were tested in the data-centric evaluation; from both categories of model-based and memory-based CFs: the UserKNN, graph-based CF and BPRMF in the user-centric evaluation part of the study (refer back to Fig. 1).

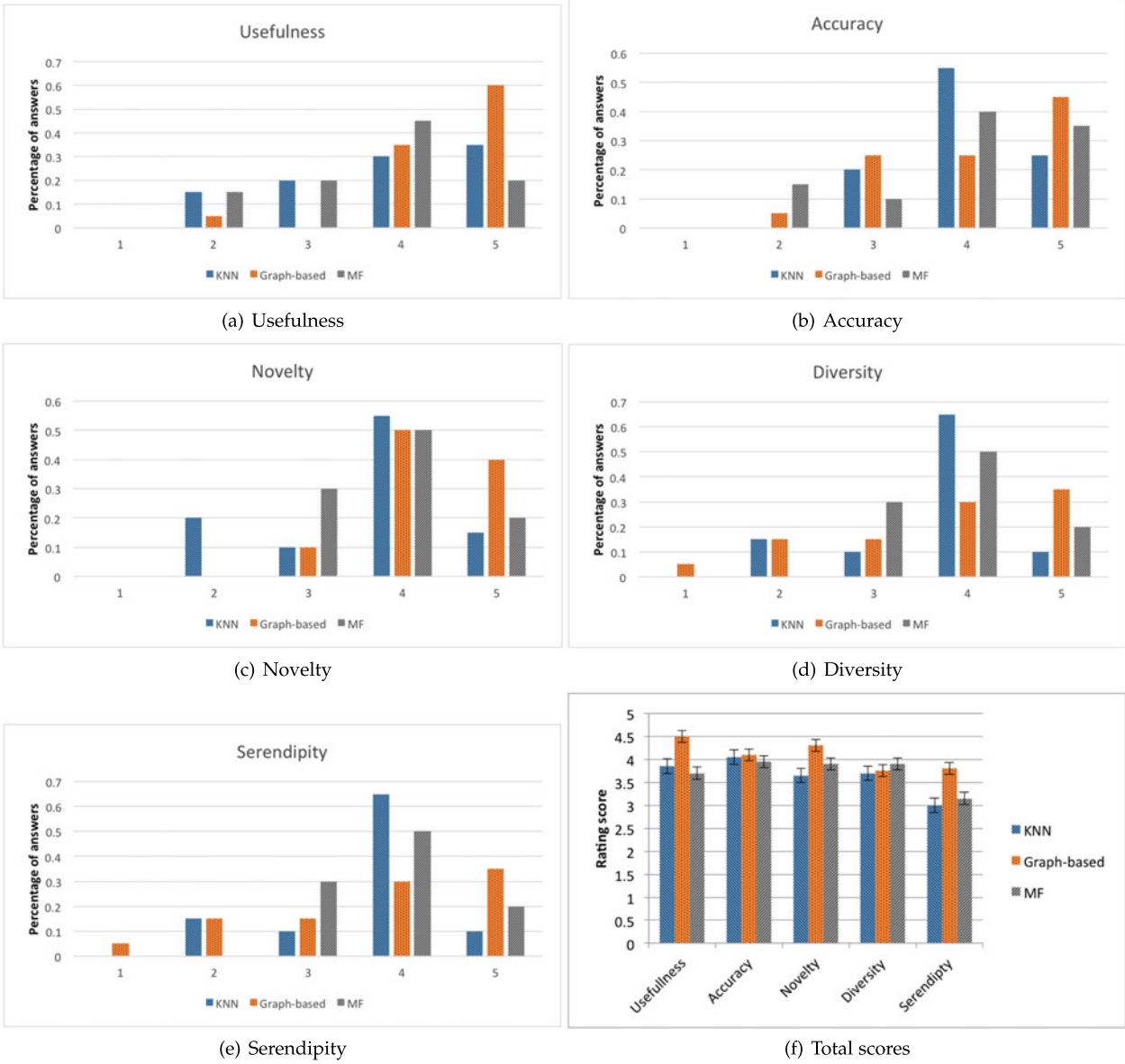
3.2 User-Centric Evaluation Results

Fig. 7 shows, in terms of percentages of answers, the level of agreement given by users on each of the five statements asked. The level of agreement ranges from 1 (completely disagree) to 5 (completely agree). Moreover, in contrast with the others, Fig. 7f presents the average rating scores of each of the recommender algorithms for each of the five statements.

We analyzed each of the five quality metrics (five statements in the questionnaire). Each of the statements is mapped onto a quality metric, which each represents a dependent variable. The dependent variables are: 1. Usefulness, 2. Accuracy, 3. Novelty, 4. Diversity, and 5. Serendipity. We have one independent variable at three levels, corresponding to the three groups that are the recommender algorithms we used: 1. a memory-based CF (user-based nearest neighbors method) that was referred to as UserKNN in the data-centric evaluation, 2. a graph-based CF, and 3. a model-based CF (matrix factorization) that was referred to as BPRMF in the data-centric evaluation. These three recommender algorithms have been selected on the basis of the data-centric evaluation presented in previous section. For the sake of simplicity, from now on, we refer to "UserKNN" as "KNN" and to "BPRMF" as "MF"; thus, we have three experimental groups: 1. KNN, 2. Graph-based, and 3. MF.

We chose to carry out a non-parametric test since we found moderate deviations from normality after running normality tests on the data. For details of normality test result, please refer to the appendix. Therefore, we carried out five non-parametric univariate tests, one for each dependent variable (metric). We used Kruskal and Wallis (K-W). Note that in the literature the power of a K-W test is found not to be much less than that of a parametric ANalysis Of VAriance (ANOVA) (assuming the use of the latter is warranted, which in our case it isn't for the lack of normality) [40]. Since our new procedure now amounts to making multiple comparisons by repeatedly testing the same subjects—once for each metric—it is necessary to correct for the so-called family-wise error rate. Therefore, we used a Bonferroni-Holm (B-H) correction. Furthermore, to be able to generalize over metrics and to compare the algorithms, we carried out a posteriori comparisons of medians and average ranks, using adjusted values of alpha (Fig. 8 and Tables 2 and 3). Table 1 provides the results of the K-W test in increasing order of magnitude of the p-values obtained for the five dependent variables. The results seemingly show that the algorithms are different in terms of usefulness, according to the K-W test for the variable usefulness (p-value= 0.17 which is smaller than alpha at 0.5). Fig. 7f also shows that the graph-based recommender receives a higher average rating score (4.5 out of 5) than the scores for both KNN (3.85) and MF (3.70), regarding the level of agreement on usefulness (Fig. 7a). However, after correcting for the family-wise error using B-H, there turns out to be no significant difference (the B-H correction demands that alpha be divided by the number of hypotheses to be tested, ordered from the smallest p-value to the highest; here, $0.017 > 0.010$). Nevertheless, the pairwise comparison of the algorithms shows that the graph-based method is different from the MF in terms of usefulness with adjusted p-values of 0.029 and 0.019 (both < 0.05) in Table 2 and also Table 3, respectively. According to the pairwise comparison of the algorithms (Tables 2 and 3) KNN and MF cannot be shown to be different in terms of usefulness since the p-values are too high (> 0.05). So we can conclude that our data do not allow us to differentiate KNN from MF from the user's perspective in terms of users' perceived usefulness of the recommendations.

As for accuracy (Fig. 7b), the algorithms cannot be shown to perform differently since the differences between their total scores are not significant even before applying the B-H correction (> 0.05) according to Table 1. The pairwise



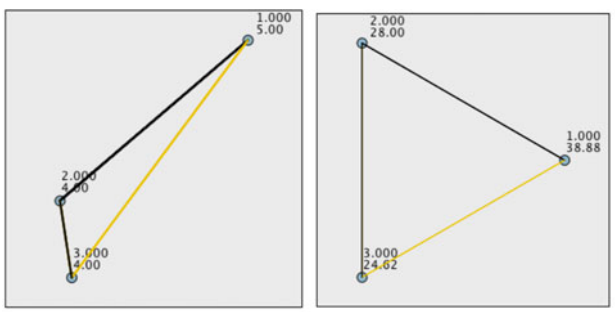
(a) Usefulness (b) Accuracy

(c) Novelty (d) Diversity

(e) Serendipity (f) Total scores

Fig. 7. Percentage of answers of the online user-centric evaluation for the five independent variables (based on) (Figs. 6a, 6b, 6c, 6d, and 6e); Total average ratings (range: 1-5) (Fig. 6f); N=60.

comparison of the algorithms confirms this (see Tables 2 and 3). This at best suggests that users perceive accuracy of the recommendations similarly for the KNN, the graph-based and the MF.



(a) Each node shows sample median of Algorithm. (b) Each node shows sample average rank of Algorithm.

Fig. 8. Pair-wise comparison of algorithms.

The results for novelty (Fig. 7c) suggest that the graph-based recommender received a better average rating score (4.30 out of 5) compared to MF (3.90 out of 5) and KNN (3.70 out of 5) but, again, the differences are not significant (> 0.05) (see Table 1). The difference between KNN and MF is not significant either, even though the results show MF to have a slightly better

TABLE 1
Kruskal-Wallis Test for Five Quality Metrics; Significant P-Values Are Marked with a Star (*)

Variable	p-value
Usefulness	0.017*
Serendipity	0.079
Novelty	0.105
Diversity	0.852
Accuracy	0.917

The significance level (alpha) is 0.05.

TABLE 2
Each Row Tests the Hypothesis that Samples 1 and 2 Distributions Are Not the Same

Algorithms (Sample1 - Sample2)	p-value (sig.)	p-value (adj. sig.)
Graph-based - MF	0.010*	0.029*
Graph-based - KNN	0.113	0.340
KNN - MF	0.288	0.864

Asymptotic significances (2-sided tests) are displayed. Significant p-values are marked with a star (*). The significance level (alpha) is 0.05.

TABLE 3
Each Row Tests the Hypothesis that Samples 1 and 2 Distributions Are Not the Same

Algorithms (Sample1 - Sample2)	p-value (sig.)	p-value (adj. sig.)
Graph-based - MF	0.006*	0.019*
Graph-based - KNN	0.037*	0.111
KNN - MF	0.517	1.000

Asymptotic significances (2-sided tests) are displayed. Significant p-values are marked with a star (*). The significance level (alpha) is 0.05.

average rating score (3.9 out of 5) than KNN (3.65 out of 5) according to Fig. 7f.

The case for diversity (Fig. 7d) is almost the same as how accuracy of the recommendations is perceived by users. So the perceived diversity of the recommendations does not seem to be not different (p-values $> 0.0.5$) for all three algorithms: KNN, the graph-based, and MF. However, Fig. 7f shows that MF received the highest average rating score (3.9 out of 5) compared to the graph-based (3.75 out of 5) and KNN methods (3.7 out of 5).

Serendipity of the recommendations (Fig. 7e) received the lowest rating scores from users but the average ratings values for all algorithms are still greater than the average (3 out of 5) based on the results in Fig. 7f. In retrospect, it seems that serendipity is a complicated term for users quickly to grasp. When looking for a proper statement that can reflect serendipity, we decided to map serendipity to a statement about 'surprising' recommendations. However, the term 'surprising' has both a positive and negative connotation, this might have made it hard for users to interpret the results properly. According to Table 1, there is no significant difference between the algorithms in terms of users' perceived serendipity of the recommendations.

4 Discussion and Conclusion

The main research question in this study is:

RQ: How is user satisfaction with recommender systems in social learning platforms related to the performance of such systems as measured in terms of their accuracy?

Our traditional, data-centric evaluation results (Figs. 5 and 6) show that the user-based nearest neighbors method outperforms other algorithms in terms of precision. As for recall, the nearest neighbors method and the graph-based method perform similarly and they both perform better than the matrix factorization method. This amounts to a clear-cut conclusion. However, the user-centric evaluation results show a quite different image (Fig. 7). All three algorithms are not significantly different from a user's perspectives in terms of accuracy of the recommendations. In fact, users provide rather high average rating scores to all the algorithms (KNN: 4.05; graph-based: 4.1; MF: 3.95; all out of 5). Since according to the relevant literature our sample size was sufficiently large at 20 users per algorithm [6], [33], we suggest this to show that the users were satisfied with the accuracy of the recommendations, regardless of the type of algorithm that generated them. Of course, our inability to establish the existence of differences does not amount to proving that there is none. However, we do conclude that user tests at least put in doubt the wisdom of choosing algorithms on the basis of their data-centric evaluation alone. Where algorithms may differ in the kind of data needed or computational overhead to produce recommendations, this is an important result.

Ignoring for the moment the lack of significance, the user-centric evaluation results (Fig. 7) seem to show that the graph-based recommender received a somewhat larger average rating score for perceived usefulness, novelty and serendipity of the recommendations by users than did the other two algorithms. If indeed there is such a difference, it is probably due to the fact that the graph-based recommender uses graph-walking methods to discover novel neighbors. These novel neighbors might be useful sources of information for a target user but they have no direct relations yet since they had no items rated in common. The further-away neighbors discovered by the graph-based method can provide useful, novel or serendipitous recommendations for a target user since they share less similarity with the target user and even they might be somehow dissimilar to some extent. Similarly, the average rating scores of the matrix factorization method for novelty, diversity, and serendipity of the recommendations perceived by users seem to be greater than the ones for the nearest neighbors method, although, as already indicated, the difference is not significant (> 0.05). For diversity of the recommendations, matrix factorization has got a greater average rating score (3.9 out of 5) than the one for the graph-based method (3.75 out of 5). Fig. 7 suggests that, unlike the traditional evaluation results, the algorithms ranking order changes depending on the quality metrics (five statements) perceived by users. We provided a detailed description of these five quality metrics in Section 2.4.2.

In general, our results show that the user-centric evaluation results do not confirm results of the traditional data-centric evaluation. This conclusion puts our study on a par with studies that argue for the necessity to study recommender systems also from user-centric perspective [9], [32], [33], even though user-centric evaluations are complicated and costly in terms of time and resources. Indeed, our study shows that recommender systems steered only by data-driven success indicators might guide data scientist to a less effective road in terms of users satisfaction. Furthermore, the results of the study make the effort to find the most accurate algorithm according to data-centric measures questionable. Considering the results of this study, one could argue that having a recommender algorithm is beneficial for the users but investing a lot in finding the most accurate recommender algorithm for the users may not be quite worth

TABLE 4
Results of Normality Test

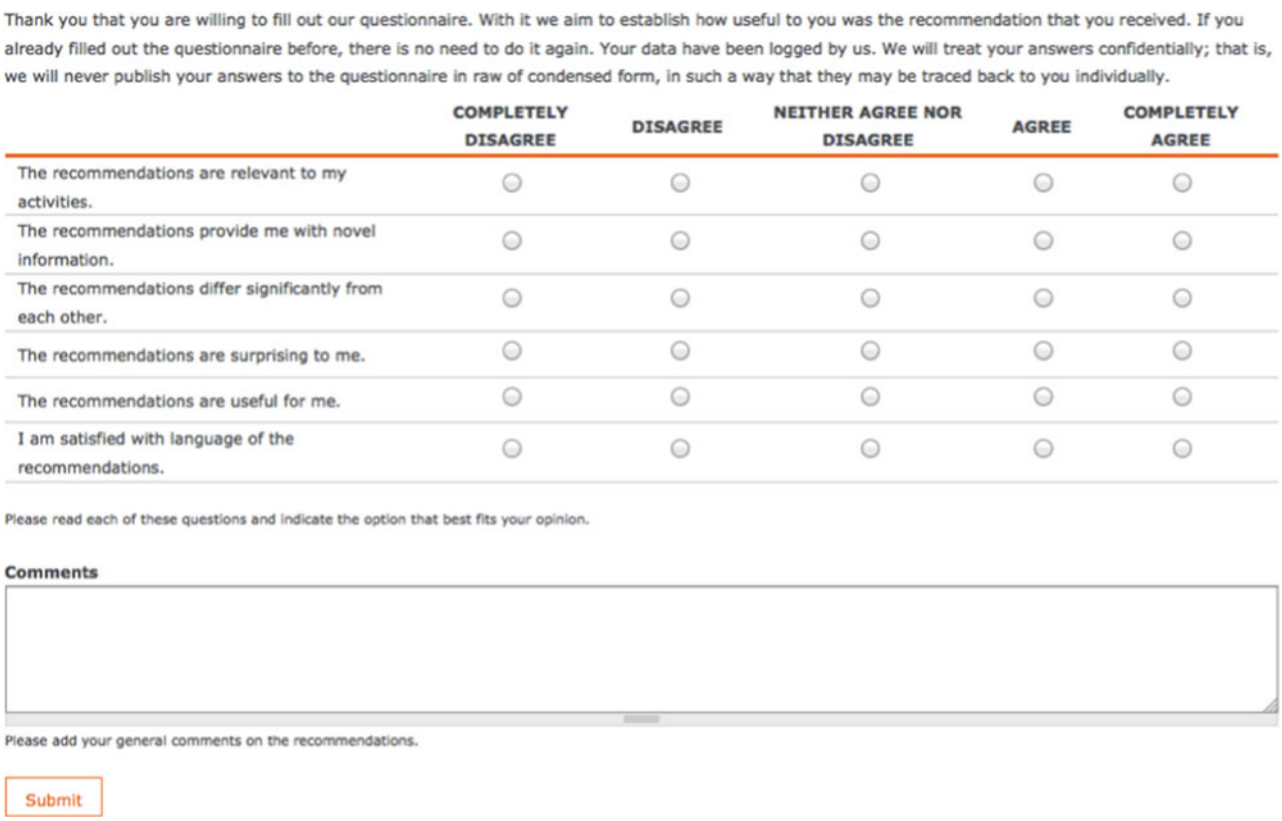
		Kolmogorov-Smirnov[a]			Shapiro-Wilk		
Algorithm		Statistic	df	Sig.	Statistic	df	Sig.
Usefulness	1	.344	20	.000	.653	20	.000
	2	.205	20	.028	.849	20	.005
	3	.270	20	.000	.867	20	.010
Accuracy	1	.247	20	.000	.816	20	.002
	2	.255	20	.001	.812	20	.001
	3	.269	20	.001	.819	20	.002
Novelty	1	.276	20	.000	.780	20	.000
	2	.314	20	.000	.824	20	.002
	3	.255	20	.001	.812	20	.001
Diversity	1	.229	20	.007	.860	20	.008
	2	.386	20	.000	.755	20	.000
	3	.255	20	.001	.812	20	.001
Serendipity	1	.201	20	.033	.864	20	.009
	2	.187	20	.064	.923	20	.112
	3	.216	20	.016	.855	20	.006

a. Lilliefors significance correction.

the efforts. According to our results, the users do not seem to recognize nor value the differences between the recommender systems in terms of the accuracy of their recommendations. We still believe that having recommender systems serve a user's specific goals is beneficial. We do not believe, though, that when having a recommender that does a reasonable job on a user goal, warrants much more research to improve it. Particularly for a social learning platform, we believe that more may be gained from investing in aspects of learning platforms other than increasing the accuracy of the recommender systems.

This conclusion lays out an interesting research line and asks for studies that complement the results found in this article. In this study, we traded loss of experimental control (which would have been obtained by working with fake users and fake problems) for increased ecological validity (which is obtained by working with real users, real problems, and real resources). Admittedly, this decision may well account for our inability to find significant results. The results of this study, therefore, need to be confirmed within a longitudinal study that tracks user satisfaction in the longer run (note that our experiment already ran for five months). It should preferably also take into account more users than the 60 people we questioned. But note that even in a large set up as provided by ODS, it was already difficult to recruit enough participants. As we already noted in the introduction, it is not simple to obtain larger numbers of users and still work in an ecologically valid setting. Our experiment testifies to that finding. Nevertheless, such settings are necessary in order to make user-based evaluations more meaningful. Preferably and if at all feasible, more questions should also be used to delineate the five user satisfaction variables (usefulness, accuracy, novelty, diversity, serendipity).

Furthermore, there exist other measures for a user-centric evaluation in learning platforms [10]. Learning effectiveness is also an important measure for user evaluations in learning platforms. However, choosing what to measure very much depends on the goal of a study. In this study, we aimed at measuring user satisfaction on the recommendations we make for them. For this, we chose to ask users explicitly whether they are satisfied with the recommendations in terms of the five metrics: usefulness, accuracy, novelty, diversity, and serendipity. However, we suggest in future studies to complement this kind of evaluation with other metrics that in our work are only implicitly evaluated,


Thank you that you are willing to fill out our questionnaire. With it we aim to establish how useful to you was the recommendation that you received. If you already filled out the questionnaire before, there is no need to do it again. Your data have been logged by us. We will treat your answers confidentially; that is, we will never publish your answers to the questionnaire in raw of condensed form, in such a way that they may be traced back to you individually.

	COMPLETELY DISAGREE	DISAGREE	NEITHER AGREE NOR DISAGREE	AGREE	COMPLETELY AGREE
The recommendations are relevant to my activities.	○	○	○	○	○
The recommendations provide me with novel information.	○	○	○	○	○
The recommendations differ significantly from each other.	○	○	○	○	○
The recommendations are surprising to me.	○	○	○	○	○
The recommendations are useful for me.	○	○	○	○	○
I am satisfied with language of the recommendations.	○	○	○	○	○

Please read each of these questions and indicate the option that best fits your opinion.

Comments

Please add your general comments on the recommendations.

Submit

Fig. 9. The questionnaire as it was administered to each participant.

such as learning effectiveness or learning performance. Erdt et al. describe learning effectiveness as 'the aim is to measure the number of visited, studied or completed items during a learning phase' [10]. Transferring this to the context of recommender systems, one perhaps could investigate whether the number of visited items by users changes when applying a specific recommender algorithms (one of the candidate recommender algorithms of our study). As for performance, one could use the results of assessments (e.g., exams) as a means of comparing learning platforms with (a variety of) recommender systems and without them. However, this was not applicable in the context of the ODS social learning platform since, in keeping with its intention merely to offer a meeting place, it lacks tests or exams. We hope that our experimental design may serve as a stepping stone for the recommender system community to gain more insights into the differences between data-centric and user-centric evaluations measures for recommender systems.

APPENDIX

This section presents results of the normality test we ran on the data. Table 4 shows the results of Kolmogorov-Smirnov and Shapiro-Wilk tests. The results reject all null hypotheses that the data is normally distributed; for the five metrics over the three algorithms (p-values are significant < 0.05). So we can conclude that the data is not normally distributed. This section also provides an overview of the questionnaire we used in the user-centric evaluation (Fig. 9).

ACKNOWLEDGMENTS

This paper is part of a doctoral study funded by the Netherlands Laboratory for Lifelong Learning (NELLL) and the Open Discovery Space project. Open Discovery Space is funded by the European Union under the Information and Communication Technologies (ICT) theme of the 7th Framework Programme for R&D. The work of Hendrik Drachsler has been supported by the EU project LACE (FP7 Program). This document does not represent the opinion of the European Union, and the European Union is not responsible for any use that might be made of its content.

REFERENCES

[1] N. Manouselis, H. Drachsler, K. Verbert, and E. Duval, *Recommender Systems for Learning*. Berlin, Germany: Springer, 2012.
[2] J. Vassileva, "Toward social learning environments," *IEEE Trans. Learn. Technol.*, vol. 1, no. 4, pp. 199–214, Oct.–Dec. 2008.
[3] H. Drachsler, K. Verbert, O. Santos, and N. Manouselis, *Panorama of Recommender Systems to Support Learning*. Berlin, Germany: Springer, 2015, pp. 421–451.
[4] B. P. Knijnenburg, M. C. Willemsen, Z. Gantner, H. Soncu, and C. Newell, "Explaining the user experience of recommender systems," *User Model. User-Adapted Interaction*, vol. 22, no. 4/5, pp. 441–504, 2012.
[5] S. M. McNee, J. Riedl, and J. A. Konstan, "Being accurate is not enough: How accuracy metrics have hurt recommender systems," in *Proc. Extended Abstracts Human Factors Comput. Syst.*, 2006, pp. 1097–1101.
[6] B. P. Knijnenburg, M. C. Willemsen, and A. Kobsa, "A pragmatic procedure to support the user-centric evaluation of recommender systems," in *Proc. 5th ACM Conf. Recommender Syst.*, 2011, pp. 321–324.
[7] S. M. McNee, J. Riedl, and J. A. Konstan, "Making recommendations better: An analytic model for human-recommender interaction," in *Proc. Extended Abstracts Human Factors Comput. Syst.*, 2006, pp. 1103–1108.
[8] F. Martin, "Industry keynote: Top 10 lessons learned developing, deploying, and operating real-world recommender systems," in *Proc. 3rd ACM Conf. Recommender Syst.*, 2009, pp. 1–2.
[9] M. A. Chatti, S. Dakova, H. Thus, and U. Schroeder, "Tag-based collaborative filtering recommendation in personal learning environments," *IEEE Trans. Learn. Technol.*, vol. 6, no. 4, pp. 337–349, Oct.–Dec. 2013.
[10] M. Erdt, A. Fernandez, and C. Rensing, "Evaluating recommender systems for technology enhanced learning: A quantitative survey," *IEEE Trans. Learn. Technol.*, vol. 8, no. 4, pp. 326–344, Oct.–Dec. 2015.
[11] J. Liu, P. Dolan, and E. R. Pedersen, "Personalized news recommendation based on click behavior," in *Proc. 15th Int. Conf. Intell. User Interfaces*, 2010, pp. 31–40.
[12] M. Richardson, E. Dominowska, and R. Ragno, "Predicting clicks: Estimating the click-through rate for new ads," in *Proc. 16th Int. Conf. World Wide Web*, 2007, pp. 521–530.
[13] S. Schnitzer, C. Rensing, S. Schmidt, K. Borchert, M. Hirth, and P. Tran-Gia, "Demands on task recommendation in crowdsourcing platforms-the worker's perspective," in *Proc. CrowdRec Workshop*, ACM RecSys, Vienna, 2015.
[14] M. Allahbakhsh, B. Benatallah, A. Ignjatovic, H. R. Motahari-Nezhad, E. Bertino, and S. Dustdar, "Quality control in crowdsourcing systems: Issues and directions," *IEEE Internet Comput.*, vol. 17, no. 2, pp. 76–81, Mar./Apr. 2013.
[15] H.-C. Schmitz, M. Scheffel, M. Friedrich, M. Jahn, K. Niemann, and M. Wolpers, "CAMera for PLE," in *Learning in the Synergy of Multiple Disciplines*. Berlin, Germany: Springer, 2009, pp. 507–520.
[16] S. Fazeli, B. Loni, H. Drachsler, and P. Sloep, "Which recommender system can best fit social learning platforms?" in *Open Learning and Teaching in Educational Communities*. Berlin, Germany: Springer, 2014, pp. 84–97.
[17] J. L. Herlocker, J. A. Konstan, and J. Riedl, "Explaining collaborative filtering recommendations," in *Proc. ACM Conf. Comput. Supported Cooperative Work*, 2000, pp. 241–250.
[18] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web*. Berlin, Germany: Springer, 2007, pp. 291–324.
[19] I. Pilászy and D. Tikk, "Recommending new movies: Even a few ratings are more valuable than metadata," in *Proc. 3rd ACM Conf. Recommender Syst.*, 2009, pp. 93–100.
[20] A. Bellogin, P. Castells, and I. Cantador, "Neighbor selection and weighting in user-based collaborative filtering?: A performance prediction approach," *ACM Trans. Web*, vol. 1, pp. 76–81, 2014.
[21] F. Ricci, L. Rokach, B. Shapira, and P. Kantor, *Recommender Systems Handbook*. Berlin, Germany: Springer, 2011, 2011.
[22] K. Verbert, H. Drachsler, N. Manouselis, M. Wolpers, R. Vuorikari, and E. Duval, "Dataset-driven research for improving recommender systems for learning," in *Proc. 1st Int. Conf. Learn. Analytics Knowl.*, 2011, pp. 44–53.
[23] S. Fazeli, A. Zarghami, N. Dokoohaki, and M. Matskin, "Mechanizing social trust-aware recommenders with T-index augmented trustworthiness," in *Trust, Privacy and Security in Digital Business*. Berlin, Germany: Springer, 2010, pp. 202–213.
[24] J. E. Hirsch, "An index to quantify an individual's scientific research output," *Proc. Nat. Academy Sci. United States America*, vol. 102, no. 46, pp. 16569–16572, 2005.
[25] C. Y. Lee, "An algorithm for path connections and its applications," *IRE Trans. Electron. Comput.*, vol. EC-10, vol. 3, pp. 346–365, Sep. 1961.
[26] J. A. Golbeck, "Computing and applying trust in Web-based social networks," Ph.D. Dissertation, University of Maryland at College Park, College Park, MD, USA. AAI3178583, 2005.
[27] H. Ma, H. Yang, M. R. Lyu, and I. King, "SoRec: Social recommendation using probabilistic matrix factorization," in *Proc. 17th ACM Conf. Inf. Knowl. Manage.*, 2008, pp. 931–940.
[28] P. Massa and P. Avesani, "Trust-aware recommender systems," in *Proc. ACM Conf. Recommender Syst.*, 2007, pp. 17–24.
[29] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in *Proc. 25th Conf. Uncertainty Artif. Intell.*, 2009, pp. 452–461.
[30] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 5–53, 2004.
[31] B. P. Knijnenburg and M. C. Willemsen, "Evaluating recommender systems with user experiments," in *Recommender Systems Handbook*. Berlin, Germany: Springer, 2015, pp. 309–352.

[32] P. Pu, L. Chen, and R. Hu, "A user-centric evaluation framework for recommender systems," in *Proc. 5th ACM Conf. Recommender Syst.*, 2011, pp. 157–164.
[33] A. Said, B. Fields, B. J. Jain, and S. Albayrak, "User-centric evaluation of a k-furthest neighbor collaborative filtering recommender algorithm," in *Proc. Conf. Comput. Supported Cooperative Work*, 2013, pp. 1399–1408.
[34] L. S. Vygotsky, *Mind in Society: The Development of Higher Psychological Processes*. Cambridge, MA, USA: Harvard Univ. Press, 1980.
[35] H. Drachsler, H. G. K. Hummel, and R. Koper, "Personal recommender systems for learners in lifelong learning networks, the requirements, techniques and model," *Int. J. Learn. Technol.*, vol. 3, no. 4, pp. 404–423, Jul. 2008. [Online]. Available: http://dx.doi.org/10.1504/IJLT.2008.019376
[36] H. Spoelstra, "Collaborations in open learning environments," Ph.D. dissertation, Welten Institute, Open University of the Netherlands, Heerlen, Netherlands, 2015.
[37] E. Pariser, *The Filter Bubble: What the Internet Is Hiding from You*. London, U.K.: Penguin Group, 2011.
[38] E. Vasilyeva, P. M. E. De Bra, and M. Pechenizkiy, "Immediate elaborated feedback personalization in online assessment," in *Proc. 3rd Eur. Conf. Technol. Enhanced Learn.*, 2008, pp. 449–460.
[39] M. Manouselis, R. Vuorikari, and F. V. Assche, "Collaborative recommendation of e-learning resources: An experimental investigation," *J. Comput. Assisted Learn.*, vol. 26, no. 4, pp. 227–242, 2010.
[40] A. Field, J. Miles, and Z. Field, "Discovering statistics using R.," London: SAGE Publications, 2012.

Soude Fazeli received the master's degree in software engineering of distributed systems from the Royal Institute of Technology (KTH), Sweden, in 2009. She is a postdoctoral researcher with the Delft University of Technology, the Netherlands, and is working toward the PhD degree at the Open University of the Netherlands. Since her master's degree, she has been working on recommender systems' implementation, integration, and evaluation. She reviews papers for several journals and conferences in the field of computer science and also technology enhanced learning. She's currently working within the EU FP7 CrowdRec project (http://crowdrec.eu). The main focus of CrowdRec project is on next generation recommendations, which are real-time, large-scale, socially informed, interactive, and context aware.

Hendrik Drachsler is professor of educational technologies and learning analytics at Goethe University Frankfurt, the German Institute of International Educational Research (DIPF), and the Welten Institute of the Open University of the Netherlands. His research interests include learning analytics, personalisation technologies, recommender systems, educational data, and mobile devices. He is elected member of the Society of Learning Analytics Research (SoLAR). In the past he has been principal investigator and scientific coordinator of various national and EU projects (e.g., laceproject.eu, patient-project.eu, LinkedUp-project.eu). He regularly chairs international scientific events and is an associate editor of *IEEE Transactions on Learning Technologies*, and special issue editor of the *Journal of Computer Assisted Learning (JCAL)*.

Marlies Bitter-Rijpkema received the PhD degree in educational technology dealing with virtual multidisciplinary teamwork. She works as an assistant professor in the Welten Institute, Open University of the Netherlands. She is a fellow of ICO and SIKS and expert of the EADTU Empower network. Over the years, she worked in various roles as a researcher, project manager, and developer across national projects and was the liaison officer of the OU to the Dutch Digital University Consortium. Her current research focusing on co-creativity, organizational contexts, professional learning in organizational contexts, social (open) innovation networks and business modeling materializes in the development of networks for innovation and learning on f.e.media literacy for social change in Asia (Medlit), supportive learning networks, and training for female entrepreneurship (Digifem) and new librarianship (LibrarySchool, Biebkracht) as well as new learning support for beta sciences (NILMRT on Microreactortechnology, gamified (m-)learning for chemistry and civil engineering) and open education (recommenders in ODS, Open Scout).

Francis Brouns has been employed at the Welten Institute since 1999 and has been working at the design, development, and implementation of learning specifications and innovative learning environments in support of lifelong learning. In her position of assistant professor, her research has evolved into supporting social and networked learning through technology enhanced learning, embracing current developments in MOOCs, and learning analytics. She takes part in various EU funded projects such as TENCompetence, LTfLL, and ODS. Currently, she is involved in the EU CIP projects ECO and EMMA, both projects considering MOOCs, developing pedagogical framework, and learning analytics approaches.

Wim van der Vegt is a senior IT developer specialized in game technologies. He has a strong background in application design, game technologies, game and simulation development, and portability issues. He participated in various European projects, e.g., Sharetec, Cooper, and Ten Competence. As a participant in RAGE, he has been the originator of the RAGE client component architecture allowing third parties to develop game technology components that can be easily ported and integrated in diverse game engines.

Peter B. Sloep is an emeritus professor of technology enhanced learning at the Welten Institute of the Open University of the Netherlands. His research encompasses such topics as networked learning (specifically but not exclusively for professionals), learning design, open educational resources, learning objects, standards for learning technologies, as well as knowledge sharing and creative collaboration in communities and networks. He co-authored more than 200 peer-reviewed publications in scholarly journals and conference proceedings, and has co-authored or edited three books. Dr. Sloep is a frequent speaker at national and international conferences. He frequently reviews papers for various journals and conferences in the field of technology enhanced learning.