

Personalized recommender system for e-Learning environment

Soulef Benhamdi^{1,2} · Abdesselam Babouri² · Raja Chiky³

Published online: 19 May 2016

© Springer Science+Business Media New York 2016

Abstract Traditional e-Learning environments are based on static contents considering that all learners are similar, so they are not able to respond to each learner's needs. These systems are less adaptive and once a system that supports a particular strategy has been designed and implemented, it is less likely to change according to student's interactions and preferences. New educational systems should appear to ensure the personalization of learning contents. This work aims to develop a new personalization approach that provides to students the best learning materials according to their preferences, interests, background knowledge, and their memory capacity to store information. A new recommendation approach based on collaborative and content-based filtering is presented: NPR_eL (New multi-Personalized Recommender for e Learning). This approach was integrated in a learning environment in order to deliver personalized learning material. We demonstrate the effectiveness of our approach through the design, implementation, analysis and evaluation of a personal learning environment.

Keywords Personalized recommender system · Personalization · e-Learning · Personal learning environments · IMSLD

1 Introduction

Learning is considered as a process of acquiring knowledge, values, and skills, through study, experience or teaching, (Tuomi 2005), whilst e-Learning could be

✉ Abdesselam Babouri
abdesselam.babouri@gmail.com

¹ Computer Science Department, University of Badji Mokhtar, 12, Annaba 23000, Algeria

² LGEG Laboratory, University of 8 Mai 1945, 401, Guelma 24000, Algeria

³ ISEP, Paris, France

defined as the use of web technologies in learning process, (Koshmann 1996). Learners constantly seek information to resolve a problem, or just to satisfy a curiosity. Due to the large amount of information available on the Net, learners spend more time on browsing and filtering information that suits their needs better, in term of preferences or domain of interests, instead of spending time on learning the materials. In addition, e-Learning environments consider that all learners are similar in their preferences and abilities. Recent researches have taken into account these problems and proposed the integration of personalization tools, in e-Learning environments. As a result, a new kind of e-Learning environments has emerged: Personal Learning Environment (PLE). A PLE is “a mix of applications (possible widgets) that are arranged according to the learner’s demands”, (Kirshenman et al. 2010).

This work aims to develop an enhanced PLE focusing on the integration of new recommendation approach (NPR-eL) in learning scenarios. To achieve this objective, we have based our work on the CSHTR (Cold Start Hybrid Taxonomy Recommender) technique developed by Li-Tang Weng, (Weng et al. 2008). CSHTR presents the best solution to overcome the cold-start problem that recommender systems suffer from. This happens in situations when new users and/or materials are involved and their historical data is missing. We propose in this paper an enhancement to CSHTR by taking into account information about the learner’s background knowledge and his memory capacity that reflects his aptitude to store information. This enables to handle the difficult cold-start situation where new learner has no time to read and rate documents which are irrelevant to his course. We call our method: NPR_eL for New multi-Personalized Recommender system for e Learning.

In order to evaluate its performance, we experimented the proposed approach in two situations, using two datasets (Book-Crossing and our University’s dataset). Due to the lack of taxonomies appropriate to e-Learning, we created a new one based on domains and sub-domains learned in our department: Graphic User Interface (GUI), Interaction Styles (sub-domain from GUI domain), Menu (sub-domain from Interaction styles domain), Knowledge Management (KM), ... In the first situation, we take into account two learner’s properties: its preferences and level. In the second one, we added another property: learner’s memory capacity. The learners should pass a post-test in the two cases (after reading the recommended items) in order to measure the impact of the added property on students’ Learning performances.

The first section presents related work, while section two details the proposed approach. Section three gives out the description of a running example of how the system makes recommendations to learners as well as the interpretation of the evaluation results. Section four concludes this paper.

2 Related work

The quality of recommendations is important for the success and acceptance of a PLE. In this section, we give an overview of basic recommendation methods, followed by the implementation of recommender systems in the field of PLE.

2.1 Recommender systems

Extracting valuable information from the huge quantity available on the web becomes a big challenge, for this reason recommender systems have been created. Recommender systems are specially developed for e-commerce, they recommend to users suitable products considering their personal needs. Three main approaches are used by this kind of systems:

- Content-based filtering (Van Meteren and Van Someren 2000), suggests to user items that are similar with those he preferred in the past. In this case, the creation of item's profile is a necessity.
- Collaborative filtering system, Kim and Li (2004), recommends items that are preferred by similar users. Thus, the exploration of new items is assured by the fact that other similar user profiles are also considered. User profiles can either be done explicitly by asking the users about their interests or implicitly by a user's given ratings.
- Hybrid recommender systems (Burke 2002) combine the two previous methods, exploiting the benefits of each one.

The cold start is one of the most experienced problems in recommendation systems. Several studies have suggested overcoming this problem by the combination of two approaches: collaborative filtering and content-based filtering (Burke 2002). The disadvantage of this solution is the lack of novelty as a part of this approach is content-based method (Ziegler et al. 2004). CSHTR, (Weng et al. 2008), is one of the existing approaches that use taxonomy to alleviate cold start problem related to new user and/or new item. It suggests the investigation of users' taxonomic preferences to treat the case where users have no common evaluations. Taxonomies are a set of keywords used to describe contents. Because of their hierarchical structure, taxonomies are more suitable for educational contents representation.

2.2 Recommender systems for PLEs

Personal Learning Environments (PLEs) become more and more popular in the field of technology enhanced learning (TEL). Personalized Learning Environment (PLE) governs the approach of an autonomous learner taking control of and managing his own learning process (Van Harmelen 2001). It assumes a selection and integration of different tools and services comprising a learning environment fully customized to the needs of an individual learner. This approach is opposed to the older paradigm of learning management systems (LMS) which have been introduced to institutions to address the needs of organizations, lecturer and of students, Taraghi et al. (2009).

Recently, recommender systems have been efficiently applied in e-commerce. Due to the success of this kind of technology, several works on Technology-Enhanced Learning (TEL) has used recommender strategies for learning, as documented by workshop proceedings, Manouselis et al. (2010), and special issues in journals, Santos and Boticario (2011). Addressing more learner-centric TEL streams, recommendations seem to be a powerful tool for personal learning environment (PLE) solutions (Mödritscher 2010). In PLEs, information are filtered based on significant context

limits thanks to personalized recommendations (Wilson et al. 2007; Salehi et al. 2014; Anandakumar et al. 2014), giving learners the opportunity to take the best of an environment where shared content differed in quality, target audience, subject matter, and is constantly expanded, annotated, and repurposed (Downes 2010). Recommendations could be valuable for various aspects of PLE-based learning activities, e.g. for formulating concrete learning goals or needs, retrieving relevant *artefacts*, finding relevant peers or tools, getting suggestions for learner interactions in a specific situation, etc. The quality of recommendations is integral for the success and acceptance of a PLE. “Good” recommendations ensure the user’s trust in the system, “bad” recommendations lead to the opposite. We propose to base not only on students’ domain of interest and background knowledge, but also on their memory capacity to provide good recommendations. Many research works in psycho pedagogy domain talk about the role of learning strategies in enhancing learners to remember what they read (Oxford 2001). However, despite its importance, at the best of our knowledge, there is no work that involves memory capacity in the recommendation process. Thus we propose to recommend to students, learning materials preferred by students who have the same memory capacity. When the learner has good memory, he could easily store what he learns, and consequently acquires more knowledge. In order to ascertain the students’ memory capacity, we used the Recall Serial Information method, Botvinick et al. (2003) (for more detail about this method please sees Section 3.1.1)

In the context of e-Learning, we cite some works that adopt recommender systems for personalization of pedagogical content. ReMashed, Drachsler et al. (2009) is a hybrid recommender system that takes advantage of the tag and rating data of the combined Web 2.0 sources. The users of ReMashed are able to rate the emerging data of all users in the system. (Bobadilla et al. 2009) observed that learners’ prior knowledge have a considerable effect on recommendation quality, for this reason he suggested to students a set of tests and introduced the results in recommendation calculation. Durao and Dolog (2010) used similarity between tags defined by learners to provide personalized recommendations. Anjorin et al. (2011) designed a conceptual architecture of a personalized recommender system considering the CROKODIL e-Learning scenario, and incorporating collaborative semantic tagging to rank e-Learning resources. A hybrid architecture that combines enhanced case-based recommending with (collaborative) feedback from users to recommend open courseware and educational resources presented in, Vladoiu et al. (2013). Lichtnow et al. (2011) created an approach for collaborative recommendation of learning materials to students in an e-Learning environment considering learning materials properties, students’ profile and the context.

By studying the state of the art, we noticed that all these existing works didn’t treat the cold start problem, so we have based our approach on CSHTR and proposed an extension in order to face this problem, and consequently, improve the quality of recommendations.

3 The proposed approach

In this section, we present the global framework in which the proposed solution will be used for better understanding. Our approach claims to help the teacher to propose

papers to students with regards to their preferences, level and memory capacity within a learning environment. Effectively, we integrated the proposed recommendation approach in a new approach that aims to create learning scenarios, using the modelling language IMS Learning Design (IMSLD) (Ferraris et al. 2005). We used IMSLD because it is an Educational Modelling Language (EML) that allows the modelling of any type of pedagogical content. An EML could be defined as: “a semantic information model and binding, describing the content and process within a “unit of learning “from a pedagogical perspective in order to support reuse and interoperability” (Rawlings et al. 2002). The main approach includes two phases (see Fig. 1): the first phase consists of the modelling of an individual learning situation, in which some course’s elements will be determined (objectives, activities ...). In the second one, the scenario created in the previous phase will be instantiated, i.e. the learning phase. Next, we detail the proposed approach.

3.1 Individual learning situation modelling

The proposed approach begins with the creation of the course a.k.a the learning situation. To create a course, two steps are required. The first step is the scenario’s particularization (manifest particularization). A manifest (presented as an xml file) is used to describe various courses’ elements. These elements are the concepts defined in the conceptual model of the IMSLD specification. In fact, only the concepts defined in the level ‘A’ are taken into account in the present work. The main manifest’s elements are the roles, activities, and method. A method identifies the activities that should be performed by each actor. We call the last element: resource. In addition to these elements, the manifest contains two other elements: title and learning objective. Among manifest’s elements, some of them are fixed from the beginning in the file, for example:

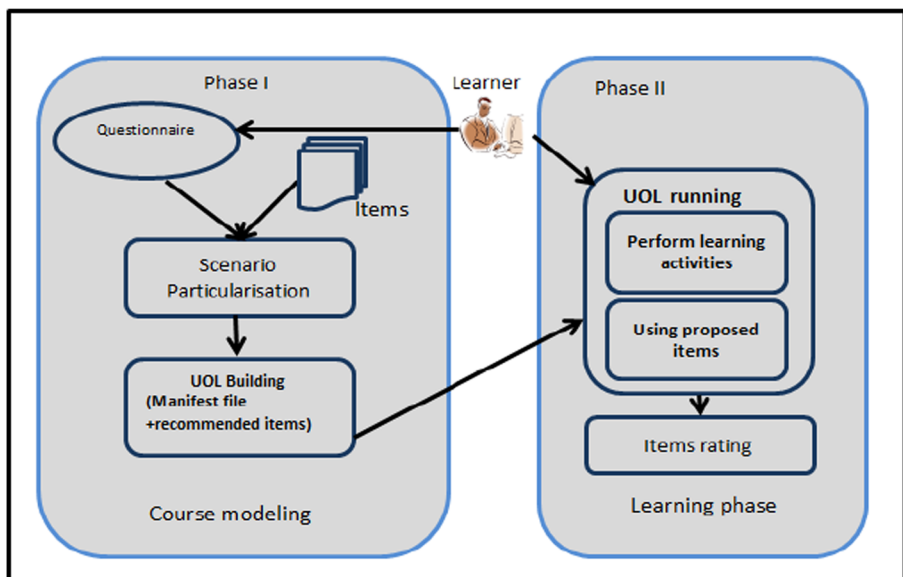


Fig. 1 The proposed approach

learning activities and actors' roles: teacher or learner. Other elements should be defined during the scenario particularization, such as: the learning objective, the description of different activities or what learners must do in detail, and the required items (papers or learning materials), which are recommended by the novel recommender system: NPR-eL (only the items' paths will be added in the manifest).

After the recommendation of appropriate items, and grouping them into a package with the particularized manifest, a Unit of Learning (UoL) is created. This is the second step.

3.1.1 Scenario particularization

Starting from a collection of information extracted from a questionnaire (the learner choose the domain he wants to read about and answer on the pre-tests questions), the educational system establishes the learner profile, which is composed of terms and tests scores. Next, the system removes the manifest (course template) that describes the selected content type from courses' templates database. These content types were extracted from pedagogical unit model described by Giacomini, Giacomini et al. (2006). This author has proposed 16 pedagogical unit models and scenarios associated with them, taking into account three types of learning units determined according to certain criteria such as: the existence of theoretical parts in a training module, various types of exercises (multiple choice, problems to solve, etc.), projects, evaluation types and / or self-assessment ... These three types are: theoretical concepts presentation, exercises as application of the course and projects (work practices). (Giacomini et al. 2006) see that pedagogical unit consists of one or more pedagogical scenarios. Each model defines not only the type of learning objects, but also the activities that must be performed by each actor. We are interested only in individual activities. Now, we pass to the next step which consists of the recommendation generation. After that, the system adds the recommended papers' paths (URLs or items location) in the manifest file (in the resource element).

In this section we present the proposed recommendation approach: NPR-eL (New Personalized Recommender for e-Learning), which aims to recommend suitable information to learners. This approach includes three steps: profiling, clustering, and prediction rating.

(1) System Model:

We used the following system model:

- a set of learners $A = \{a1, a2... ac\}$.
- a set of items $T = \{t1, t2, ... tk\}$.
- each item t is represented by a set of descriptors $D(t) = \{d1, d2, ..., dl\}$. A descriptor is a sequence of ordered taxonomic topics, denoted by $d = \{p0, p1, ..., pq\}$, $d \in D(t)$, $t \in T$.
- we created taxonomic information suitable for e-Learning domain. The taxonomic information describing five items are represented in the following listing:
 - $t1$: GUI < General
 GUI < Interaction styles < Menu
 GUI < Interaction styles < Command languages

- t2: KM < General
 - KM < Knowledge types < Explicit
 - KM < Knowledge types < Implicit
- t3: KM < Knowledge creation process < General
 - KM < Knowledge creation process < Nonaka’s model
- t4: GUI < General
 - GUI < Interface types < Command Interfaces
 - GUI < Interface types < Graphique Interfaces
- t5: KM < KM Models < measurement’s Models
 - KM < KM Models < evaluation’s Models
- The Book-crossing dataset uses the Amazon’s site taxonomic information (www.amazon.com).

(2) Profiling

Students should answer to a questionnaire to identify their preferences (domain of interest and educational content type). The first part of this questionnaire presents the taxonomy that we have created. Since we haven’t found an appropriate taxonomy for e-Learning, we chose the most important words representing some domains programmed to be learned in our department our university’s taxonomy, for example: Graphic User Interface “GUI”, Knowledge Management “KM”..., and construct a novel taxonomy. The second one presents different pedagogical content types.

This questionnaire contains also five tests or exams with different levels: very low, low, middle, high, and very high level. In addition, the students should pass the RSI’s test, Botvinick et al. (2003), in order to measure their memory capacity (memory span). The learner’s profile will be described by the following vector:

$$\vec{v}_{ai} = (S_1, \dots, S_m, .S_{t1}, \dots, S_{t5}, t_{s1}, \dots, t_{s5}, mc).$$

$$s_i = \sum_{j=1}^q p_i(t_j)/|nb| \quad (1)$$

Where m is the number of all topics that describe the learner’s domain of interest. S_x represents the score of test x , t_{sx} is time response corresponding to the test x , and mc represents the learner memory capacity. Formula (1) calculates the score of topic i . Recall that in our system, a new learner has no evaluation; in this case, the term weights (terms’ appearance frequency) will be extracted from taxonomy descriptors ($D(t)$: taxonomic information that describes items’ content) of all the items (papers or learning materials) including these terms. The score of term i , is the average of all its weights. $P_i(t_j)$: weight of term i in the

item t_j . $|nb|$: number of items containing the term i . the learner's profile vector is normalized based on L1-Norm technique such that:

$$\vec{v}_{ai} = (S_1, \dots, S_m, .S_{t1}, \dots, S_{t5}, t_{s1}, \dots, t_{s5}, mc)$$

$$\forall a_i \in A, \sum_{k=1}^{|j|} v_{ik} = 1$$

We present in the following the questionnaire that uses our university's taxonomic information:

What's your domain of interest?

GUI

Interaction styles

Menu

Command languages

KM

Knowledge types

Explicit

Implicit

KM

Knowledge creation process

Nonaka's model

GUI

Interface types

Command Interfaces

Graphique Interfaces

What educational content type you prefer?

Theoretical concepts.

Theoretical concepts and exercises.

Projects.

In the experiment, we choose the learner a_1 as a target learner in order to test our recommender system. The learner a_1 choose to read about GUI/ Interaction styles/Menu (an example of questionnaire (Learning domains and pre-test level 1) is shown in Fig. 5 Section 4). The learner a_1 's five tests' scores and memory capacity are presented in Table 2. The followings vectors (the second vector is normalized) represent a_1 's profile:

$$\vec{v}_{a1} = (S_{GUI}, S_{Interaction\ Styles}, S_{Menu}, .S_{t1}, \dots, S_{t5}, t_{s1}, \dots, t_{s5}, mc)$$

$$\vec{v}_{a1} = (0.03, 0.01, 0.0, 0.04, 0.02, 0.0, 0.04, 0.01, 0.13, 0.08, 0.21, 0.25, 0.17, 0.03)$$

Where $S_{GUI}=0.03$ presents the score of topic 'GUI'. $S_{t1}=0.04$ is the score of the first test. $t_{s1}=0.13$ is the time answer of test 1. $mc=0.03$ is the a_1 's memory capacity

(3) Clustering

In this step, the system will look for learners that are similar to the target learner. The idea that we would like to formalize is the weighting of recommendations, which stem from learner details, taking into account that the

recommendations of the learners with better scores have a greater weight than the recommendations of the learners with lower scores. We use to evaluate a multi-dimensional similarity between users not only their domain of interest and prior knowledge, but also the time they spend in different tests and their memory capacity. The new measurement for similarity between the users x and y , can be established as defined in the Eq. (2). We added other terms to the correlation metric in order to take into account the learners' properties: background knowledge (tests' scores and the time spent by the learner to answer on each pre-test) and memory capacity. The first term of the equation measures background knowledge similarity between two students x, y with knowledge (C_x) and (C_y) respectively, while the second term measures the answer time similarity between them. The answer time similarity is calculated by using the Eq. (3); the third term measures the item preference similarity between x, y . This similarity is calculated by a correlation metric (Eq. (4)), and the fourth term calculates the memory capacity similarity between x, y (mc_x, mc_y). RSI takes only the longest list remembered by the learner (l) in consideration. In order to calculate the memory capacity with more veracity, we proposed the new formula (5).

$$\begin{aligned} new_sim(x, y) = & \sum_{n=1}^L n\alpha(Cx_n - Cy_n) \\ & + 1/\phi(tx, ty) + sim(x, y) + mc_x - mc_y, \quad \alpha \in [0-1] \end{aligned} \quad (2)$$

$$\phi(tx, ty) = \sum_{z=1}^L |tx_z - ty_z| / L \quad (3)$$

$$sim(a, i) = \sum (r_{a,j} - \bar{r}_a)(r_{i,j} - \bar{r}_i) / \sqrt{\sum_{j \in T} (r_{a,j} - \bar{r}_a)^2 (r_{i,j} - \bar{r}_i)^2} \quad (4)$$

$$mc_x = l/g \quad (5)$$

Where tx_z presents the time that learner x needs to pass test z . L represents the number of tests ($L=5$). After recommendation, the target learner should evaluate explicitly the recommended items. The real ratios will replace the terms scores (S_i) in the learner's profile, so this ratios will be taken into account in calculation of similarity between that learner and the future target learner. $r_{a,j}$ is the learner a 's explicit evaluation (ratings) to item j , \bar{r}_a denotes the average rating made by $a \in A$. mc_x is the memory capacity of the learner x . g is the number of the longest list's

characters presented to the learner. l is the number of the longest list’s characters remembered by the learner after one presentation.

Based on (2), we can find the target learner’s group (uc) that contains the most similar learners. The similarity values are ordered, and we take the n first values, i.e. n is the number of the most similar learners ($|uc|$). The number n is defined by hand, in our experiments n is set to 4.

$$t_cluster(a) = argmax a \in A new_sim(a, i) \tag{6}$$

(4) Prediction Rating

After allocating the learner to the adequate group, he will normally receive recommendations based on his group’s preferred items. According to (Weng et al. 2008), the preferred items can be identified by the number of users who rated the item (popularity) and the average rating. However, the value of rating for an item does not represent its popularity as it may receive low values. Thus, we suggest to calculate items popularity using the Eq. (7), (Weng et al. 2008), on the basis of assessments that exceed (or equal) the half of the maximum rating value (i.e. 0.5 on the scale of 0–1 (only ratings ≥ 0.5 are taken into account)).

$$\sigma(uc, t_j) = \sum_{u \in uc} r(a, t_j) / |uc|, r(a, t_j) [0.5-1] \tag{7}$$

$$\psi(uc, t) = \sum_{a \in uc} r(a, t_j) / |uc(t_j)| \tag{8}$$

$r(a, t_j)$ represents the learner $a \in uc$ ’s explicit rating to item t_j . uc is the similar learners group. $|uc|$ refers to the number of learners in group uc . $\psi(uc, t_j)$ refer to the average explicit ratings to item t_j . $|uc(t_j)|$ denotes the number of students in uc who rated t_j explicitly. Formally, we calculate uc ’s general preference to item $t_j \in T$ in a group uc as follows:

$$cpref(uc, t_j) = \beta \times \psi(uc, t_j) + (1-\beta) \times \sigma(uc, t_j) \tag{9}$$

$0 \leq \beta \leq 1$ is a student controlled variable for adjusting the weights between the average item preference and item popularity. According to Weng’s model (Weng et al. 2008), β is set to 0.7.

To predict the target user’s rating on a given item (rank a, t :see line 4 of the algorithm), NPR-eL controls if the item is commonly preferred by similar users (9) and calculates the taxonomic similarity between the target user and the item (Eq. 10). According to Weng’s model, ϕ is set to 0.5 (for more detail please refers to (Weng et al. 2008)).

$$t_sim(\vec{v}_a, \vec{v}_t) = \vec{v}_a \times \vec{v}_t \tag{10}$$

$t\text{-sim}(\vec{v}_a, \vec{v}_t)$ computes the similarity between learner a 's taxonomy vector and item t 's taxonomy vector. for example the t 's taxonomy vector:

$$\vec{v}_t = (\text{XGUI}, \text{X interaction style}, \text{X menu}, \text{X command language})$$

We calculate the score of term i (X_{ti}) in an item t using the term i 's frequency within the descriptor of this item and the number of all terms that compose this descriptor:

$$X_{ti} = f_i / \text{nb_topics}(t) \quad (11)$$

f_i is the term i 's frequency within the descriptor of the item t . The number of all terms that compose this descriptor is denoted by $\text{nb_topic}(t)$.

The following algorithm summarizes recommendation steps defined by the New multi-Personalized Recommender system for e-Learning.

Algorithm NPR-eL recommender (a, k)

Where $a \in A$ is a given target student

K is the number of items to be recommended

- 1) SET T , the candidate items list
- 2) FOR EACH $t \in T$
- 3) SET $uc = t_cluster(a)$
- 4) SET $rank_{a,t} = \varphi \times cpref(uc, t) + (1-\varphi) \times t\text{-sim}(\vec{v}_a, \vec{v}_t)$
- 5) END FOR
- 6) Return the top k items with highest $rank_{a,t}$ score to a .

$0 < \varphi < 1$ is a learner controlled variable for adjusting the weights between the predicted item preference ($cpref(uc, t)$) and predicted taxonomic preferences ($t\text{-sim}(\vec{v}_a, \vec{v}_t)$)

3.1.2 Unit of learning building

We first start by defining the Unit of Learning (UoL) a.k.a the unit of study. A unit of learning is used to identify a course or a module (Ferraris et al. 2005). It is composed of two parts. The first part is the content organization, which describes learning elements (learning objective, activities, roles of learning's actors, resources...) in hierarchical way in the manifest file. The second part concerns the resources. Learners need to use some learning materials during the learning process. The recommendation system selects among the papers set, those satisfy learners' needs (see Fig. 2).

3.2 Learning phase

This is the phase where the learning process actually takes place, in other words, the execution of the UoL created previously. At this time, the NPR-eL consults learner's

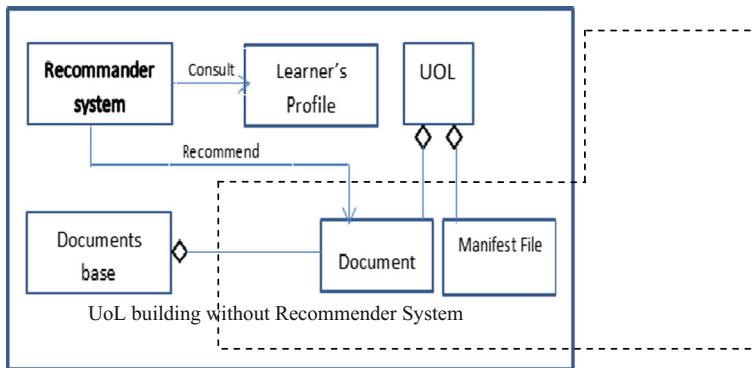


Fig. 2 UoL building

profile and offers to him a set of papers. Then, the learner begins to perform the various tasks listed in the manifest. This phase ends with the learner's explicit evaluation on the recommended items.

4 Experiment and evaluation

We conducted the experiment at the computer science department of our university, during delivering the course. The actors of these experiments are 10 master students from our university, Algeria. We have tested the proposed approach using two different datasets: The Book-Crossing (<http://www.informatik.uni-reiburg.de/~ziegler/BX/>), and the University's dataset. We used the Book-Crossing Dataset to compare between our recommender system and CSHTR's performance. To evaluate the new approach performance, two cases have been studied. In the first case, we used the new recommender system without integrating memory capacity factor. In the second case, recommendations were produced introducing the memory capacity factor. Here, the evaluation is performed on the subject 'GUIs' as in the previous example. The measurement for students' performances is done by calculating the deviation between the pre-tests score (the average of the five pre-tests marks) and post-test score for each student, in each case. We present a prototype of an ongoing work called NPLe (New Personalized Learning Environment) in which we integrate the proposed recommendation approach: NPR-eL. The NPLe offers several features to both learners and teachers. In this environment, learners can receive recommendations and consult the recommended papers by accessing to their space see (Fig. 3a). The recommended papers are chosen from papers created by the teachers (see Fig. 3b).

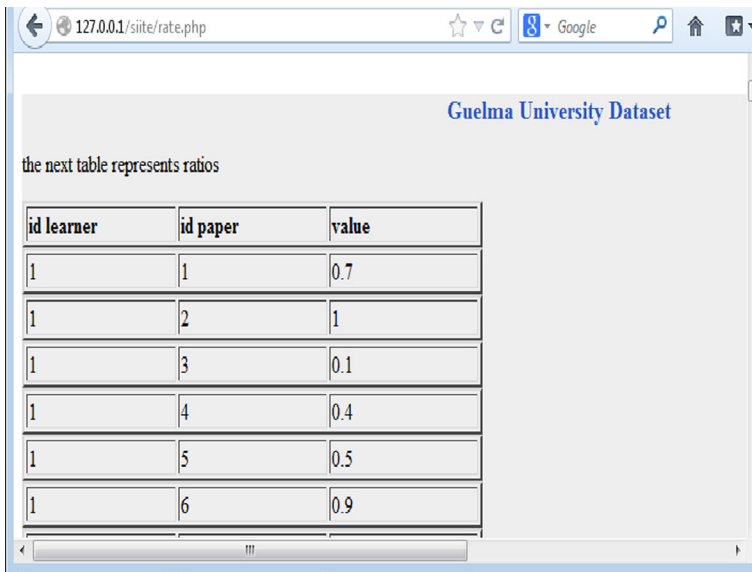
4.1 Evaluation metrics

Our objective in this experiment is to test the accuracy of the proposed recommendation technique that is to evaluate its capacity to recommend interest items to students. For the recommendation quality evaluation, we used



Fig. 3 a The learner’s space b teacher space (adding new item)

Precision, Recall and F1 metrics (Bobadilla et al. 2009). The first one defines the ratio between the number of interest selected items and the number of all



Guelma University Dataset

the next table represents ratios

id learner	id paper	value
1	1	0.7
1	2	1
1	3	0.1
1	4	0.4
1	5	0.5
1	6	0.9

Fig. 4 The university's dataset

items. It represents the possibility that the selected item is of interest, while the second metric defines the ratio between interest selected items and the available interest items. Recall metric represents the probability that an interest item will be selected. The last metric is the combination of the two previous one.

Table 1 The learners' explicit evaluations

	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10
t1	0.7	0.8	0.9	0.0	1.0	0.2	0.0	0.9	0.2	0.9
t2	1.0	0.2	0.1	0.5	0.0	0.1	0.5	0.9	0.1	0.1
t3	0.1	0.1	0.1	0.1	0.5	0.5	0.2	0.1	0.2	0.9
t4	0.4	0.5	0.7	0.3	0.2	0.8	0.3	0.1	0.1	0.7
t5	0.5	0.8	0.6	0.1	0.3	0.2	0.9	0.1	0.4	1.0
t6	0.9	0.2	0.1	0.5	0.9	0.4	0.9	0.4	0.1	0.5
t7	0.0	0.4	0.8	0.8	0.9	0.1	0.1	0.0	0.3	0.8
t8	0.4	0.1	0.2	0.2	0.1	0.3	1.0	0.9	0.4	0.9
t9	0.8	0.3	0.1	0.1	0.1	0.4	0.0	0.1	0.2	0.8
t10	0.9	0.4	0.5	0.5	0.1	0.5	0.5	0.1	0.3	0.9
t11	0.7	0.5	0.8	0.8	0.4	0.8	0.2	0.4	0.0	0.7
t12	1.0	0.9	0.2	0.2	0.0	0.2	0.3	0.0	0.5	1.0
t13	0.0	0.8	0.4	0.4	0.9	0.4	0.9	0.9	0.2	0.6
t14	0.5	0.5	0.1	0.1	0.7	0.1	0.3	0.7	0.3	0.4
t15	0.8	0.4	0.3	0.3	1.0	0.3	0.9	1.0	0.0	0.5

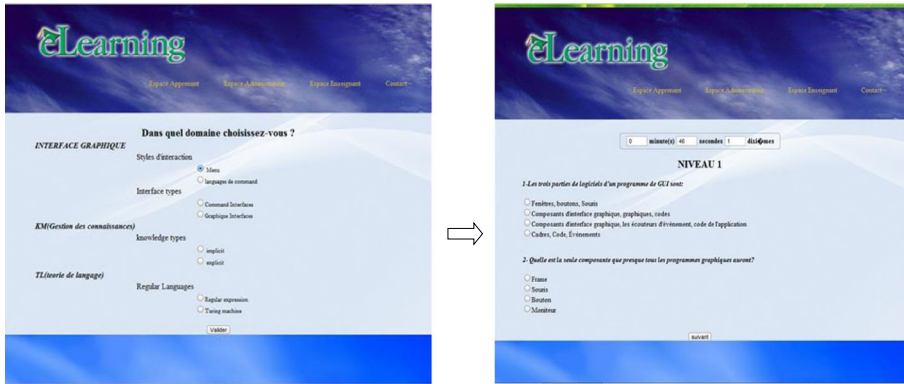


Fig. 5 The questionnaire

4.2 Experiment with the university dataset

We create a new dataset based on the novel taxonomy. Indeed, we propose fifteen papers to ten learners in order to rate them. Figure 4 shows the explicit ratios of the learner a1. Table 1 presents the learners' explicit ratings (evaluations).

As we mentioned in the previous section, the learner should answer to the proposed questionnaire, and pass the five pre-tests (Fig. 5) and the RSI test (see Fig. 6). The motive for the pre-tests is to assess the pre-knowledge of the students before they start their learning process.

We note that the target learner's has chosen: GUI/ Interaction styles/Menu. The pre-tests will propose questions about Graphic User Interface and especially about menu. Table 2 shows tests scores (scores and time response of each test). Next, the system creates learner's profile based on his answers. The recommender engine recommends a list of K (5 to 15) items to the target

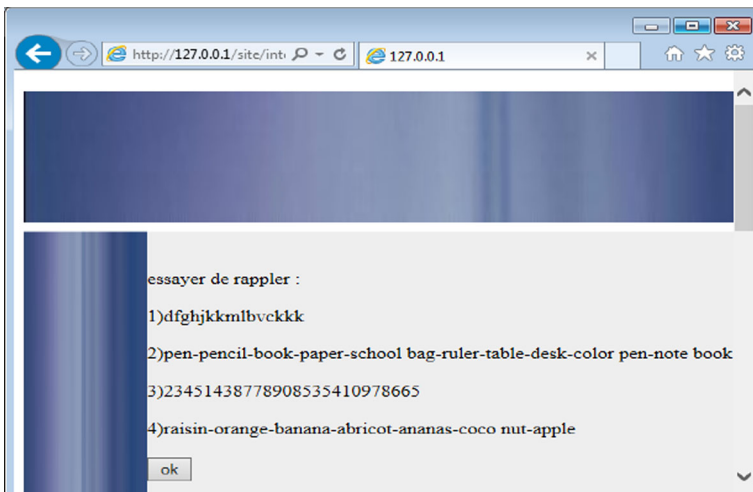


Fig. 6 The RSI test

Table 2 Pre-tests' scores (S_{ti} : score of test t_i , t_i : time response of test i)

	S_{t1}	S_{t2}	S_{t3}	S_{t4}	S_{t5}	t_1	t_2	t_3	t_4	t_5
a1	0.9	0.4	0.1	0.9	0.2	3	2	5	6	4
a2	0.5	0.4	0.1	0.7	0.3	5	7	6	8	5
a3	0.2	0.7	0.1	0.3	0.1	2	5	7	9	6
a4	0.6	0.3	0.5	0.2	0.9	6	3	8	9	7
a5	0.1	0.4	0.2	0.0	0.0	1	2	9	9	8
a6	0.9	0.9	0.1	0.1	0.1	2	5	6	3	7
a7	0.2	0.1	0.5	0.3	0.2	2	5	7	1	2
a8	0.7	1.0	0.2	0.5	0.3	2	9	9	5	7
a9	0.9	0.1	0.1	0.1	0.4	2	5	6	2	5
a10	1.0	0.5	0.8	0.9	0.8	1	2	1	3	3

learner a1. The recommended items ordered by their predicted ratings (from the highest value to the lowest one), are listed in Table 3.

Then comes the learning process (see Section 3.2) in which the learner realizes the proposed activities. Figure 7 shows the execution of an example of UoL using CopperCore, which is integrated in NPLe. The learner was given the necessary time to read the recommended papers before he was examined in a post-test. The post-test questions cover a deep knowledge about menu (Fig. 8). Figures 9 and 10 shows the post-test and pre-tests scores (case 1: SR without memory capacity, case 2: SR with memory capacity).

Result discussion

The Comparing of real assessments with predicted ones by the computation of Precision, Recall and F1 metrics, Herlocker et al. (2004) (see Fig. 11), conducted us to a good judgment of our recommendation approach. We tested different values for k (number of recommended items) ranging from 5 to 15. We note that among the five recommended items there is only one uninteresting item (t_8 : evaluated with small value), while the four other items are well evaluated. In fact, this item hasn't been rated at all. In the second case ($k=10$), we find only three uninteresting items (t_8 : rated with small value and two items were not evaluated (t_7 , t_{13})). In the third case ($k=15$), there is five uninteresting items (t_8 , t_7 , t_{13} , t_4 , t_3). We note that NPR-eL recommender is better in recommending smaller item lists. From the pre-tests and post-tests scores, we see that there is a significant difference between the marks of pre-tests and post-tests in each case which means that learners acquired more knowledge when they used the

Table 3 Recommendation results

Number of items (k)	Recommended items
$k=5$	t_8 , t_5 , t_{10} , t_6 , t_{11}
$k=10$	t_8 , t_5 , t_{10} , t_6 , t_{11} , t_{15} , t_7 , t_{13} , t_1 , t_{12}
$k=15$	t_8 , t_5 , t_{10} , t_6 , t_{11} , t_{15} , t_7 , t_{13} , t_1 , t_{12} , t_3 , t_9 , t_4 , t_2 , t_{14}

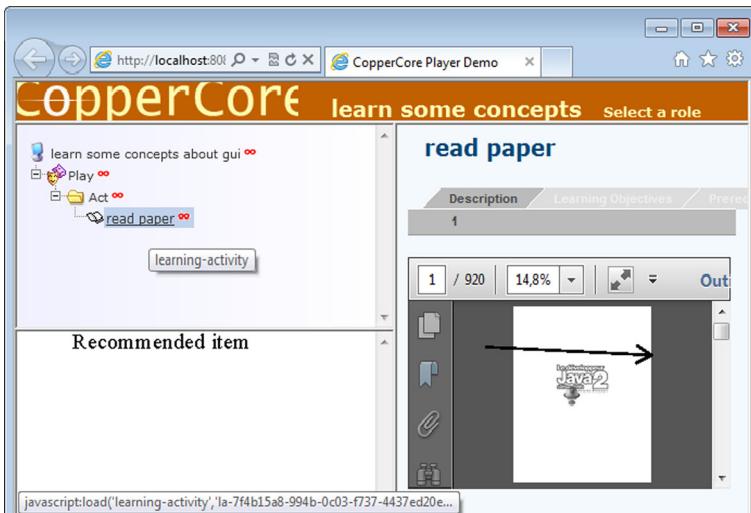


Fig. 7 Learning process

proposed recommender system. The gap between the pre-tests and post-test marks is larger in the second case (introducing memory capacity factor), that means that adding another learners' property could increase their learning performance.

4.3 Experiment with Book-Crossing dataset

In this section, we present the results obtained when we used the book-Crossing dataset. We select ten users and fifteen books from the dataset. The taxonomic information associated to these books was extracted from Amazon's site. Indeed, we

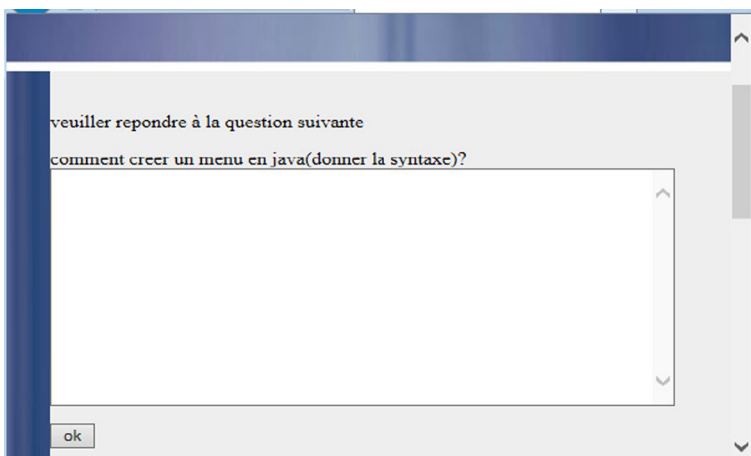


Fig. 8 An example of post-test

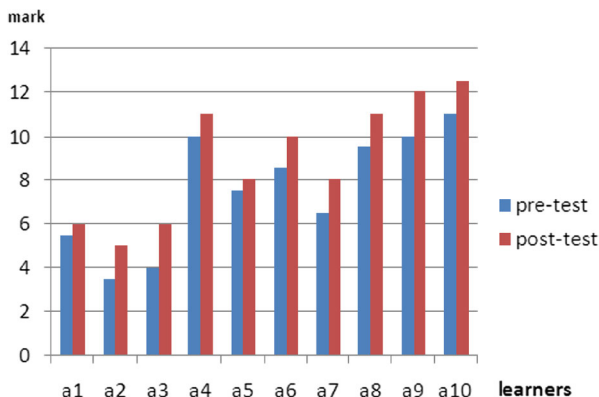


Fig. 9 Pre-tests' and post-tests' scores (SR without memory capacity factor, Benhamdi and Seridi (2011))

have obtained the same order of items as shown in Table 1 (see Fig. 12). Figures 13 and 14 summarizes the pre-test and post-test scores in both cases.

Result discussion

By comparing our results (see Fig. 15) with Weng's experiments results (Weng et al. 2008), which prove that CSHTR is better than TPR (Taxonomy-driven Product Recommendations), (Ziegler et al. 2004), we conclude that NPR-eL is more efficient than both of them. The main difference between CSHTR and NPR-eL is in learner's profile computation and clusters formation. Recall that Learner's profile is a set of topics' weights; NPR-eL calculates these weights with more veracity using items' taxonomic profiles. CSHTR calculates similarity between users taking into account only the item preferences, whereas NPR-eL introduces multiple factors: learner's item preferences, his level and his memory capacity. We can note that there is a difference between pre-tests and post-tests marks for each learner in both cases. It is also clear that the highest deviation between pre-tests and post-tests marks is obtained in the second case. These results confirm our hypothesis about the good impact of the memory capacity factor on learning quality.

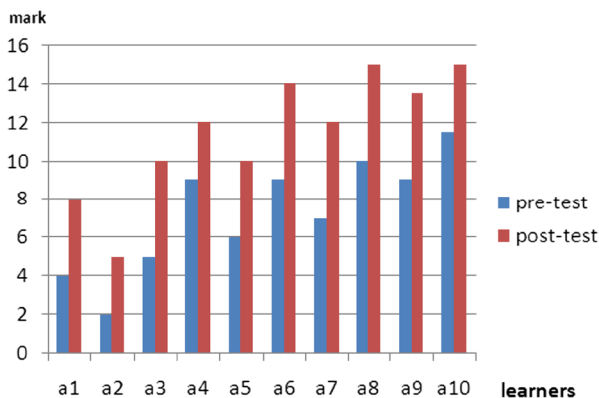


Fig. 10 Pre-tests' and post-test' scores (SR with memory capacity factor)

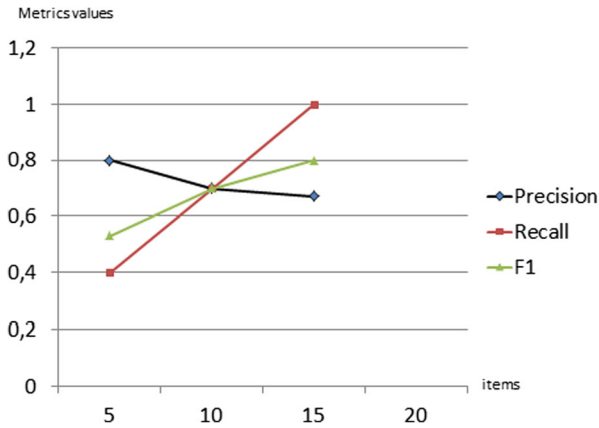


Fig. 11 NPR-eL evaluation (the university’s dataset)

4.4 Global discussion

The objective of this experiment is to evaluate the impact of learner’s properties on the quality of recommendations and learning. Corresponding to this objective, we tested our recommender in two situations. In the previous work, Benhamdi and Seridi (2011), the recommender was tested taking into account two properties: learner’s preferences and background knowledge. The results (see Fig. 16) prove that this recommender is more efficient than (Weng et al. 2008). The main difference between these two recommenders is in the number

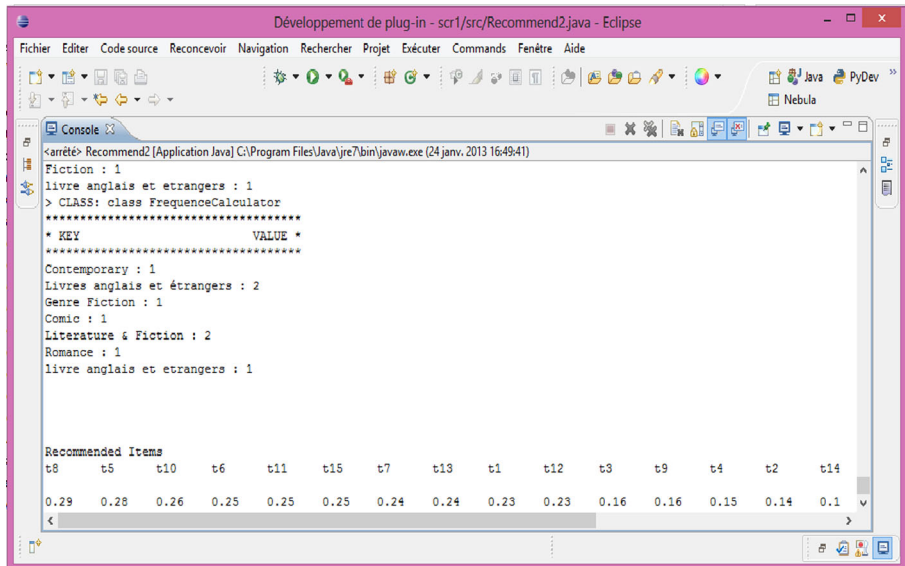


Fig. 12 Recommended items

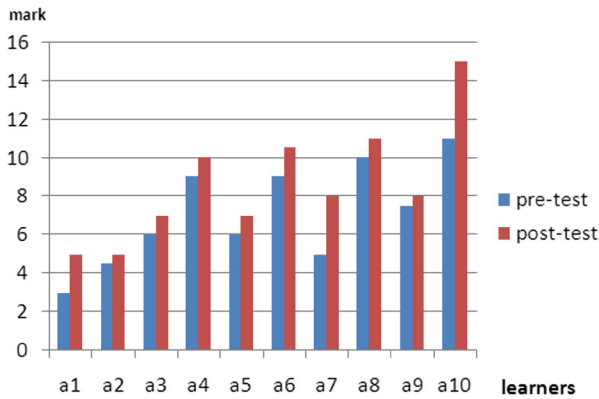


Fig. 13 Pre-tests' and post-test' scores (SR without memory capacity factor, Benhamdi and Seridi (2011))

of learners' properties introduced in the recommendation process. Weng's (2008) recommender uses only learner's preferences. This work presents the second situation, in which another property is added: learner's memory capacity. It can be observed from the Figs. 11, 15 and 16 that the NPR-eL recommender achieves the best results when we introduce this property. This comparative study shows that the proposed recommender system (NPR-eL) worked better when we gather all the previous factors in recommendation computation.

It can be seen from the results that NPR-eL approach generates more accurate recommendation than TPR and CSHTR by using two dataset. However, more experiments should be carried on larger datasets that can contribute positively in recommendation quality. In fact, the more learners and items are used the better recommendation is expected to be.

Since the personalization is applied in learning environments to motivate learners in their studies, good personalized recommendations lead necessarily to a better learning quality. The post-test's results confirm this assumption.

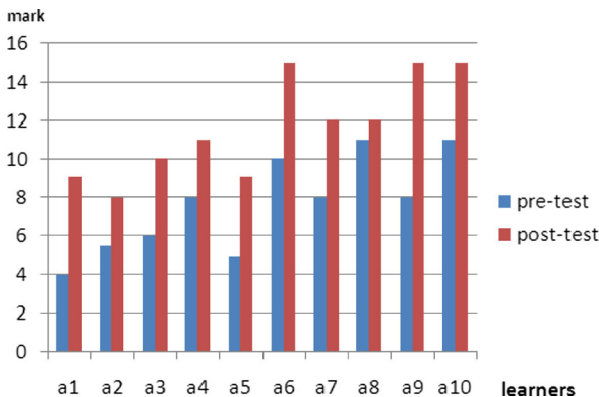


Fig. 14 Pre-tests' and post-test' scores (SR with memory capacity factor)

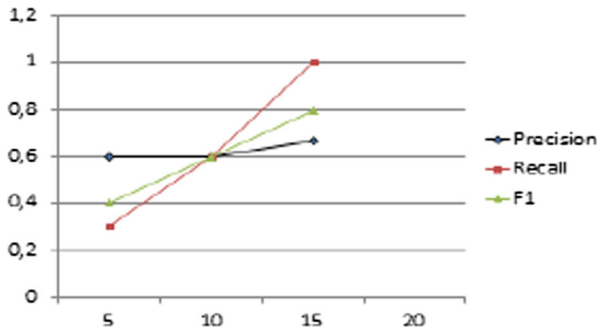


Fig. 15 NPR-eL evaluation (book-crossing dataset)

Finally, NPLE by integrating of recommender system offers potentials for enhancing the personalization and improving learning quality, while putting students in control of their learning process spanning across different tools and services:

- Resources reuse, thanks to the courses modelling using IMSLD.
- Student’s evaluation could help teacher in their task, for example they may have an idea about the content they should create (resources, learning activities ...).
- NPLE guides learners through their learning session by not only recommending learning materials, but also proposing the activities they should do.
- Student centric: the student is the controller of his/her learning process, he/she takes initiative in learning and responsibility for managing his/her knowledge and competences.
- The NPLE enhance personalization due to the integration of multiple properties related to the learner.

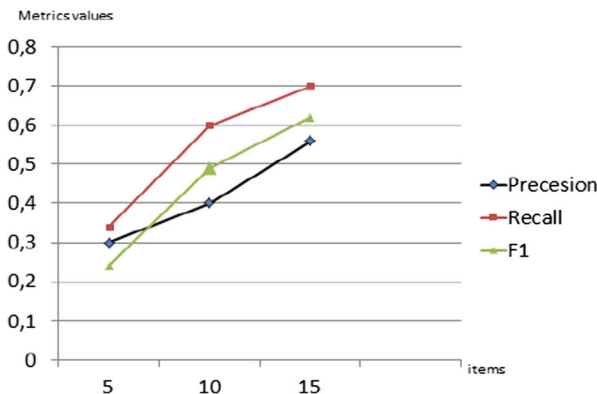


Fig. 16 The evaluation results, Benhamdi and Seridi (2011)

5 Conclusion

Previously, learning platforms offered static systems with pre-defined tools to a set of many learners (one-size-fits-all model). The emergence of personalization tools specially recommender systems has changed the learning domain. Learning platforms has become more flexible to learners; according to learners' preferences and abilities, the system has to find the best learning resources that better fit their needs.

Due to the important impact of students' characteristics, we proposed a new recommender system: NPR-eL (New multi-Personalized Recommender system for e-Learning) considering learner's preferences, background knowledge and their capacity of storage, in order to make multi-personalized recommendation. The experiments that we have conducted on two different datasets prove that NPR-eL produces personalized recommendations of a good quality due to the integration of multiple factors related to learners specially their capacity to remember what they read.

In future work, we aim to introduce other learner's characteristics that may enhance both learning and recommendation quality, and carrying out experiments with a large number of users and items dataset in order to confirm our recommender technique's accuracy. We aim also to finalize NPLE (learning phase) and integrate new useful tools and services.

References

- Anandakumar, K., Rathipriya, K., & Dr Bharathi, A. (2014). A survey on methodologies for personalized e-Learning recommender systems. *International Journal of Innovative Research in Computer and Communication Engineering*, 2(6).
- Anjorin, M., Rensing, C., & Steinmetz, R. (2011). Towards ranking in folksonomies for personalized recommender systems in e-Learning. <http://www.crokodil.de>, <http://demo.crokodil.de>.
- Benhamdi, S., & Seridi, H. (2011). Pedagogical content personalization. In the Proceeding of ITHET'11 Conference (vol. 10), Turkey.
- Bobadilla, J., Seradella, F., & Hernando, A. (2009). Collaborative filtering adapted to recommender systems of e-Learning. *Knowledge-Based Systems*, 22, 261–265.
- Botvinick, M., et al. (2003). Constructive processes in immediate serial recall: A recurrent network model of the bigram frequency effect. In B. Kokinov & W. Hirst (Eds.), *Constructive memory* (pp. 129–137). Sofia: New Bulgarian University.
- Burke, R. (2002). Hybrid recommender systems: survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4), 331–370.
- Downes, S. (2010). New technology supporting informal learning. *Journal of Emerging Technologies in Web Intelligence*, 2(1), 27–33.
- Drachsler, R. et al. (2009). ReMashed - an usability study of a recommender system for mash-ups for learning. 1st Workshop on Mashups for Learning at the International Conference on Interactive Computer Aided Learning, Villach, Austria.
- Durao, F., & Dolog, P. (2010). Extending a hybrid tag-based recommender system with personalization. In Proceedings 2010 ACM Symposium on Applied Computing, SAC 2010, Sierre, Switzerland (pp. 1723–1727).
- Ferraris, C., Lejeune, A., Vignollet, L. & David, J.-P. (2005). Modélisation de scénarios d'apprentissage collaboratif. Université de savoie, EIAH'2005 (Environnements Informatiques pour l'Apprentissage Humain), Montpellier, 2005 (pp. 285–296).
- Giacomini, E. P., Trigano, P., & Alupoie, S. (2006). Knowledge base for automatic generation of online IMS LD compliant course structures. *Educational Technology & Society*, 9(1), 158–175.
- Herlocker, J. L., Konston, J. A., Terveen, L. G., & Riedl, J. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22, 5–53.

- Kim, B.M., & Li, Q. (2004). Probabilistic model estimation for collaborative filtering based on items attributes. In Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI'04), Beijing, China (pp. 185–191).
- Kirshenman, U. et al. (2010). The demands of modern PLEs and the ROLE approach. In M. Wolpers et al. (Eds.), *EC-TEL 2010, LNCS (6383)*, pp. 167–182.
- Koshmann, T. (1996). *CSCS: Theory and practice of an emerging paradigm*. Mahwah, NJ: Lawrence Erlbaum Associates.
- Lichtnow, D., et al. (2011). Recommendation of learning material through students' collaboration and user modeling in an adaptive e-Learning environment. *Technology enhanced systems and tools, SCI, 350*, 258–278.
- Manouselis, N., Drachler, H., Verbert, K., & Santos, O. C. (2010). Proceedings of the 1st workshop on recommender systems for technology enhanced learning (RecSysTEL). *Procedia Computer Science, 1(2)*, 2773–2998.
- Mödritscher, F. (2010). Towards a recommender strategy for personal learning environments. *Procedia Computer Science, 1(2)*, 2775–2782.
- Oxford, R. (2001). Language learning styles and strategies. In M. Celce- Murcia (Ed.), *Teaching English as a second or foreign language* (3rd ed.). Boston: Heinle & Heinle.
- Rawlings, A., van Rosmalen, P., Koper, R., Rodriguez-Artacho, M., & Lefrere, P. (2002). Survey of educational modelling languages (emls). Technical report.
- Salehi, M., Kamalabadi, I. N., & Ghouschi, M. B. G. (2014). Personalized recommendation of Learning material using sequential pattern mining and attribute based collaborative filtering. *Education and Information Technologies, 19*, 713–735.
- Santos, O. C., & Boticario, J. G. (2011). *Educational recommender systems and technologies: Practices and challenges*. Hershey: IGI Global.
- Taraghi, B., Ebner, M., Till, G., & M'uhlbürger, M. (2009). personal learning environment – a conceptual study. In International Conference on Interactive Computer Aided Learning (pp. 25–30).
- Tuomi, I. (2005). The future of learning in the knowledge society: Disruptive changes for Europe by 2020. In Y. Punie & M. Cabrera (Eds.), *The future of ICT and learning in the knowledge society: Report on a joint DG JRC–DG EAC workshop held in Seville, 20–21 October* (pp. 47–85). European Commission: Luxembourg.
- Van Harmelen, H. (2001). Design trajectories: four experiments in PLE implementation, *Interactive Learning Environments, 1744–5191, 16(1)*, 2008, 35–46.
- Van Meteren, R., & Van Someren, M. (2000). Using content-based filtering for recommendation. In: Proceedings of MLnet/ECML2000 Workshop, Barcelona, Spain.
- Vladoiu, M., Constantinescu, Z., & Moise G. (2013). QORECT – a case-based framework for quality-based recommending open courseware and open educational resources In C. Bădică, N.T. Nguyen, & M. Brezovan (Eds.), *ICCCI 2013, LNAI 8083*, 681–690.
- Weng, L. T., Xu, Y., Li, Y., & Nayak, R. (2008). Exploiting item taxonomy for solving cold-start problem in recommendation making. *ICTAI (2)*, 113–120.
- Wilson, S., Liber, P. O., Johnson, M., Beauvoir, P., & Sharples, P. (2007). Personal learning environments: challenging the dominant design of educational systems. *Journal of e-Learning and Knowledge Society, 3(2)*, 27–28.
- Ziegler, C.-N., Lausen, G., & Schmidt-Thieme, L. (2004). Taxonomy -driven computation of product recommendations. In Proceedings of the 2004 ACM CIKM Conference on Information and Knowledge Management Washington, D.C., USA, ACM Press, 406–415.