

QORECT – a Case-Based Framework for Quality-based Recommending Open Courseware and Open Educational Resources

Monica Vladoiu, Zoran Constantinescu, Gabriela Moise

UPG University of Ploiesti, Romania
{monica,zoran}@unde.ro, gmoise@upg-ploiesti.ro

Abstract. More than a decade has passed since the start of the MIT OCW initiative, which, along with other similar projects, has been expected to change dramatically the educational paradigms worldwide. However, better findability is still expected for open educational resources and open courseware, so online guidance and services that support users to locate the appropriate such resources are most welcome. Recommender systems have a very valuable role in this direction. We propose here a hybrid architecture that combines enhanced case-based recommending (driven by a quality model tenet) with (collaborative) feedback from users to recommend open courseware and educational resources.

Keywords: open courseware (OCW), open educational resources (OERs), quality model, case-based reasoning, recommendation system

1 Introduction

More than a decade has passed since the start of the MIT OCW initiative, which, along with other similar projects, has been expected to change dramatically the educational paradigms worldwide. However, despite the huge opportunities offered by open education, traditional textbooks and readings, and intranet educational resources are still here, dominating the majority of teaching and learning venues of Higher Education institutions even though all students are effectively online. Greater adoption of OERs both within formal and informal education seems to be impeded by four issues: *discoverability, quality assurance, bridging the last mile, and acquisition* [1]. Modern search engines generally do an ill job when searching for educational content because they are not tailored with this purpose, focusing mainly on content and metadata, and, moreover, they lack what it takes to locate the proper educational resource that is suited for a specific user's goal, that builds up on her prerequisites (for example, learner's previous knowledge), and that provide for making the next step towards her goal (e. g. mastering of a certain concept). For the time being, there is no quality assurance mechanism that could provide support for (1) *learners and instructors* in their quest for reaching the most appropriate educational resources for their specific educational needs in any particular context, neither for (2) *faculty or institutions* that

are or want to become involved in this movement, and they may be concerned about the challenges or interested in the gains of this process, nor for (3) *developers* who need guidelines for designing and building such educational resources, nor for (4) educational resources' *evaluators* [2, 3, 4]. In many OCW/OER repositories educational content exists only immersed in context and without a significant effort this content cannot be both sorted out from its initial environment (becoming truly reusable and remixable) and entangled within a new educational context, bridging the last mile. Acquisition is also difficult, taking into account all the fears of OCW/OERs providers (faculty, teachers, educational resources designers etc.): lack of credit, of copyright control over derivative works, and so on. Therefore, better findability is expected for open courseware and OERs, so online guidance and services that support users to locate the appropriate ones is beneficial as related work shows [5, 6, 7, 8, 9].

Recommender Systems (RSs) are a sort of information filtering systems that either try to predict whether a particular user would like a given item (*prediction problem*), or try to identify a set of N items that will be of interest to a certain user (*top-N recommendations*). Various kinds of recommendation approaches that rely on various paradigms are available: content-based (item features, user ratings), collaborative (similar ratings from similar users), case-based (content-based case based reasoning), demographic user profiles, knowledge based, and hybrid [10-15]. When using recommender systems in e-educational contexts (some authors call that Technology Enhanced Learning - TEL), the object of recommendation may be a learning resource, a learning activity, a peer learner, a mentor, and so on [7, 8, 9, 16]. Moreover, the recommendation goal is usually complex, e.g. the RS may suggest a set of alternative learning paths throughout a mixture of educational resources, in various forms (learning sequences, hierarchies of interacting learning resources), and the recommendation must be done within a meaningful pedagogical paradigm that reflects user instructional goal, specific interests, the context of use etc., and that helps him accomplish his instructional goal and objectives [16, 17].

In this paper we propose a hybrid approach that combines enhanced case based recommending (driven by a quality model tenet) with (collaborative) feedback from users to recommend OCW and OERs within a unified framework. The structure of the paper is as follows: the next section presents our case-based architecture for recommending OCW and OERs, detailing the quality model and the case-based reasoning process, the third section includes the related work pointing on its main unsolved issues, and the last section shows some conclusions and future work ideas.

2 Case-based Architecture for Recommending OCW and OERs

In this section we present our approach of a case based recommendation system for OCW/OERs. Case Based Reasoning (CBR) is a very well-known artificial intelligence technique, which has already proven its effectiveness in numerous domains. The fundamental concept in CBR is that similar problems have similar solutions, and, therefore, solving a new problem is done by analyzing the solutions to previous, similar problems [18]. The solutions offered as an outcome of the CBR cycle rely on previous cases stored in the case base, and the system is able to learn continuously by adding new cases to the case base. In its more general form, CBR relies on the

k-Nearest Neighbors (kNN) algorithm, which core is a similarity function that will be used to find *k* previous cases similar to the new (target) case. Assessing similarity at the case level (or between the target query and candidate case) is based on combining the individual feature level similarities for the relevant features. Usually, a weighted sum metric such as that shown in Eq. 1 is used, in which the similarity between some target query, *t* and some candidate case *c*, is the weighted sum of the individual similarities between the corresponding features of *t* and *c*, namely t_i and c_i . Each weight encodes the relative importance of a particular feature in the similarity evaluation and each individual feature similarity is calculated according to a similarity function that is defined for that feature, $sim_i(t_i, c_i)$ (shown in Eq. 2 in our case, where $d_i = |c_i - t_i|$). The value of the similarity score is between 0 and 1, and the more the two cases *t* and *c* are similar, the more the similarity score gets closed to 1 [19, 20, 21].

$$similarity(t, c) = \frac{\sum_{i=1}^n w_i * sim_i(t_i, c_i)}{\sum_{i=1}^n w_i} \quad (1) \quad sim_i(d_i) = \begin{cases} 1, & d_i \leq 1 \\ 2 - d_i, & 1 \leq d_i \leq 2 \\ 0, & d_i \geq 2 \end{cases} \quad (2)$$

2.1 The quality model

We present briefly here the quality criteria for quality assurance of OCW/OERs, which have been introduced and presented in detail in [2], and put to work and refined further elsewhere [3-4]. These criteria can be applied for assessing quality of both small learning units and entire courseware. They fall within four categories concerned with the quality of the content, of the instructional design, of the technology-related aspects, and with the assessment of the courseware, as a whole. These criteria correspond to the quality characteristics of quality in use, internal and external product quality according to ISO/IEC 25000 SQuaRE standard, and they cover the next user needs: effectiveness, efficiency, satisfaction, reliability, security, context coverage, learnability, and accessibility [2-4]. A very concise presentation of these quality criteria is included in Table 1, which works as a rubric for our quality model (where the scoring meaning is as follows: 0=absence, 1=poor, 2=satisfactory, 3=good, 4=very good and 5=excellent). For the time being the evaluation of OCW/OERs is subjective, being based on many decades of evaluators' experience in Higher Education. However, this seems to be the tendency in other works in this area [4, 22-25].

Table 1. Quality Rubric for Quality Assurance of OCW and OER

| | <i>To what degree an educational resource allows learners to have engaging learning experiences that provide for mastery of the content.</i> | |
|---------------------------------|--|-----|
| Content related criteria | CR1: readability | 0-5 |
| | CR2: uniformity of language, terminology, and notations | 0-5 |
| | CR3: availability of the course syllabus | 0-5 |
| | CR4: comprehensiveness of the lecture notes | 0-5 |
| | CR5: modularity of the course content | 0-5 |
| | CR6: possibility to select the most suitable learning unit | 0-5 |
| | CR7: opportunity to choose the most appropriate learning path | 0-5 |
| | CR8: top-down, bottom-up or combined approach | 0-5 |

| | | |
|---------------------------------------|---|----------------|
| | CR9: availability of assignments (with or without solutions) | 0-5 |
| | CR10: <i>resource related</i> : accuracy ¹ , reasonableness ² , self-containedness ³ , context ⁴ , relevance ⁵ , multimedia inserts ⁶ , interactive elements ⁷ , correlation with the entire course ⁸ , links to related readings ⁹ , links to other resources (audio, video etc.) ¹⁰ | 0-5 x 10 |
| Instructional design criteria | <i>Address instructional design and other resource's pedagogical aspects</i> | |
| | ID1: goal and learning objectives (<u>outline</u> the material) | 0-5 |
| | ID2: learning outcomes (<u>knowledge, skills, abilities, attitudes</u>) | 0-5 |
| | ID3: appropriate instructional activities | 0-5 |
| | ID4: availability of the evaluation and auto-evaluation means | 0-5 |
| | ID5: learning theory | 0-5 |
| | ID6: instructional design model | 0-5 |
| | ID7: <i>reflective learning opportunities</i> : desired outcome of education becomes the construction of coherent functional knowledge structures adaptable to further lifelong learning | 0-5 |
| Technology related criteria | <i>OCW/OERs are expected to benefit fully from ICT technologies, to have user-friendly interfaces, and to comply with various standards.</i> | |
| | TR1: conformity with standards for interoperability | 0-5 |
| | TR2: compliance with standards for accessibility | 0-5 |
| | TR3: extensibility wrt to adding content, activities, and assessments, from a technological viewpoint(developers and learners) | 0-5 |
| | TR4: user interface's basic technological aspects | 0-5 |
| | TR5: supporting technology requirements at user's end | 0-5 |
| | TR6: prerequisite skills to use the supporting technology | 0-5 |
| | TR7: multi-platform capability | 0-5 |
| | TR8: supporting tools | 0-5 |
| Courseware evaluation criteria | <i>All major OCW initiatives have become lately more involved with their learners, and therefore regular assessment of OCW effectiveness and using the results for further improvements is essential.</i> | |
| | CW1: <i>courseware overview</i> : content scope ¹ and sequence ² , intended audience ³ , grade level ⁴ , periodicity ⁵ of content updating, author's credentials ⁶ , source credibility ⁷ , multiple-languages ⁸ , instructor facilitation ⁹ or semi-automated support ¹⁰ , suitability for self-study ¹¹ , classroom-based ¹² study, and/or peer collaborative ¹³ study, time requirements ¹⁴ , grading policy ¹⁵ , instructions on using ¹⁶ the courseware, reliability ¹⁷ , links to other ¹⁸ educational resources (readings, OCW, OERs etc.) | 0-5 x 18 |
| | CW2: availability of prerequisite knowledge | 0-5 |
| | CW3: availability of required competencies | 0-5 |
| | CW4: matching the course schedule with learner's own pace | 0-5 |
| | CW5: <i>terms of use/service</i> : availability of repository policies wrt copyright&licensing issues, security for primary, secondary and indirect users, anonymity, updating and deleting personally identifiable information, age restrictions, netiquette, etc. | 0-5 |
| | CW6: freeness of bias and advertising | 0-5 |

| | | |
|--|---|-----------|
| | CW7: suitable design and presentation of educational content | 0-5 |
| | CW8: <i>user interface richness (style)</i> : navigational consistency ¹ , friendliness ² , multimedia ³ , interactivity ⁴ , adaptability ⁵ (both to user's needs and context) etc. | 0-5 x5 |
| | CW9: providing a formal degree or a certificate of completion | 0-5 |
| | CW10: <i>participatory culture and Web 2.0 facets</i> : contribution to the content ¹ , collection of users' feedback ² , collaboration with fellows ³ , sharing the development ⁴ /using experience ⁵ | 0-5 x5 |

2.2 The QORECT Architecture

The architecture we propose for recommending open courseware and OERs based on a quality model is three layered, as it can be seen in Fig. 1, and it is called QORECT (Quality driven Open educational resource/courseware case-based RECommending Tenet). First layer is *User and Context Layer*, which is dedicated to user's request and to collecting information about the user and about his context. Thus, a user may address a request for specific OCW/OERs that includes her instructional goal, the subject, the material level (graduate, undergraduate, K12 etc.), an indication about her preference for OCW or OERs, an expectation with respect to the resource quality (i. e. more than good), and so on. For the time being the resources are manually collected and inserted within a local pool of resources, but for future versions of the system we intend to include an automatic OCW/OER federated search engine, based on the taxonomy introduced in a previous work [26]. Additionally, she is expected to provide information about the context in which the resource will be used (for example, within a classroom setting, for self-study, either independent or in a learning network etc.) to be processed by the Context Manager. We also foresee for future versions the capability of automatic capturing of context information within a context-aware architecture. Finally, this layer is responsible with processing information about the user (input and capture) for (case-based) profiling, personalization, and for creating opportunities for learning about the user. A conversation module is included here. Automatic capturing of information about the user behavior is also envisaged for future versions.

The second layer, called *OCW-OER Case-Based Recommender Engine*, is dedicated to the recommendation of OCW/OERs, which results from a CBR process. First, the user request is re-constructed from its parts in form of an *input case FV (Feature Vector)*, by including user's descriptive information and user's context, and by retaining user requirements. Then, the first *Retrieve* step in the CBR cycle is activated, and the best k1-NN resources are selected (k1 FVs that describe those resources) based on a simple similarity measure (Eq. 1 and Eq. 2). Further on, the quality of these k1 resources is assessed based on our quality model, and k1 quality-enhanced FVs are obtained. Based on the user's initial quality expectation, k2-NN ($k2 \leq k1$) resources are retained, in the second *Retrieve* step, for further processing within the CBR cycle (described by their *Quality Feature Vectors - QFV*). Next, the *Reuse* phase is on, and these resources are presented to the user. If she is happy with the results, then we have

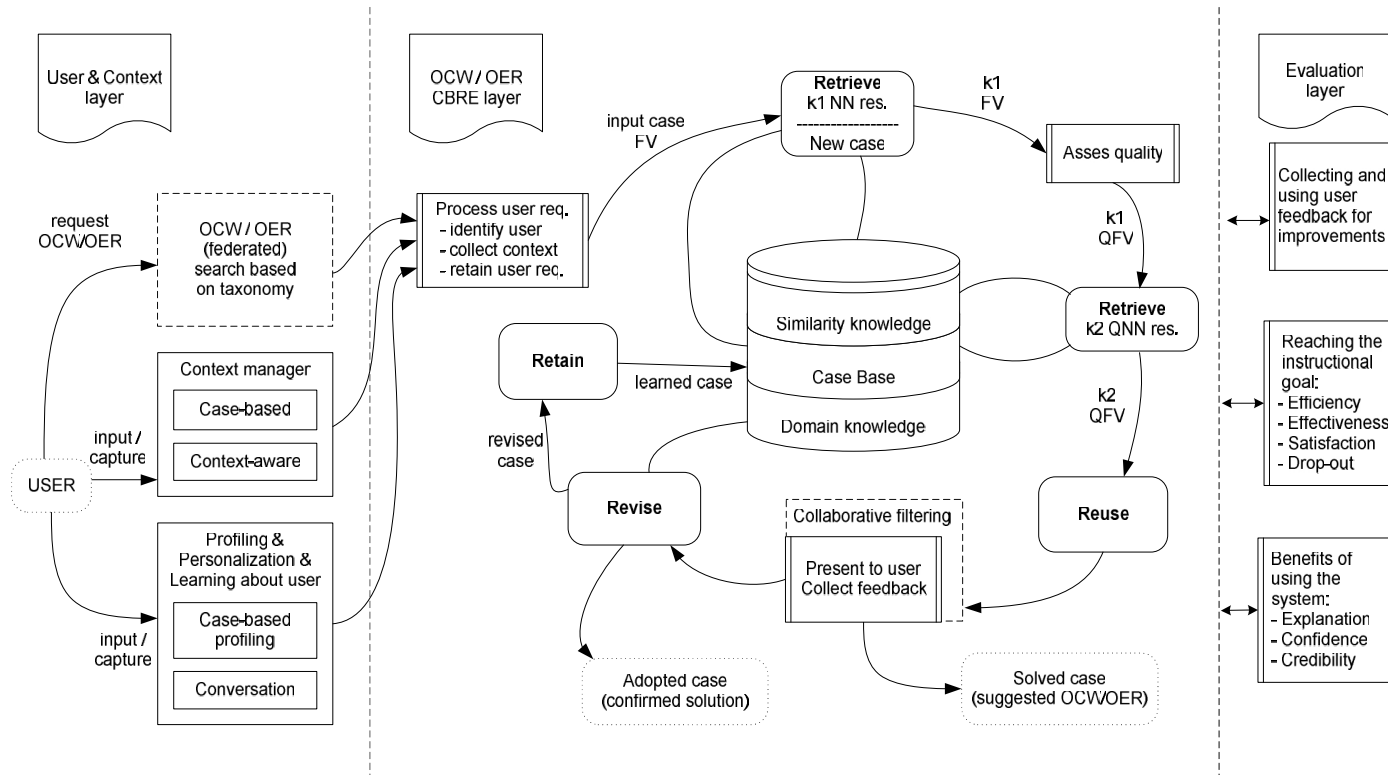


Fig. 1. QORECT – a Case-Based Architecture for Collaborative Recommending OCW and OERs based on a Quality Model

“solved the case” for her. Still, we need her feedback for future collaborative resource filtering based on her appreciations of both quality and usefulness of the resource for her instructional goal in a particular context. Her feedback may be collected also in the case she is not content with the recommendation, and further *Revise*-ing is needed, resulting an adapted case that is again presented to her. Finally, the system is able to *Retain* as learned cases the new cases within the revised ones. Within our framework, a case consists of information (and learned knowledge) about the user, about the desired resource (quality included), about the context of use, and on user’s feedback.

The *Evaluation layer* is concerned with collecting user feedback, and with the evaluation of the system within educational settings from a dual point of view: reaching the instructional goal, and benefiting from using the system. With respect to the instructional aspects, there are several issues to be considered: effectiveness, efficiency, satisfaction, the drop-out rate etc. [16]. Effectiveness may be measured as the total number of completed, visited or studied OCW or OERs during a learning session, while efficiency may be quantified as the total amount of time needed for reaching the instructional goal. Satisfaction reflects the user’s contentment with regard to the accomplishment of his instructional goal. If this goal is learning, then the drop-out rate, which shows the number of learners that abandon during the learning phase, is an important indicator of learning process as it is shown in a literature survey [16]. Satisfaction with the recommendations made, explanations supporting these recommendations (quality rubrics to motivate the score, collaborative filtering results), confidence and credibility are specific benefits from using the recommendation system.

2.3 The Case-Based Recommending process

In this sub-section we present briefly the recommendation process:

Step0: process OCW/OER pool → Feature Vectors (FVs)
Step1: process user request:
 - identify (profile of) the user
 - collect the context data (input + automatic capture)
 - retain user requirements (goal, subject, quality expectation, some relevant quality criteria, and so on)
 - construct the input (target) case (as a FV)
Step2: retrieve the k1 most similar resources or create a new case (new case not similar to stored ones) and go to 8
Step3: assess quality for the k1 resources → k1QFV
Step4: retrieve the k2 most similar cases wrt quality expectation and relevant criteria and collaborative rating
Step5: present the k2 resources to the user (reuse); if s/he is content with the suggested OCW/OER → *solved case*
Step6: collect user feedback (reaching instructional goal, resource usefulness etc.) and collaborative rating
Step7: revise the case (*adapted case* - confirmed)
Step8: retain the new or adapted case as a *learned case*

3 Related Work

In the TEL domain there is a variety of RSs that propose learning resources to users using several recommending approaches, and the most related to our work –in some particular aspects - will be presented further here. A well-known system that has been proposed for the recommendation of learning resources is RACOFI (Rule-Aplying Collaborative Filtering), which combines recommendation based on users' ratings with results of an inference rules based engine that determines associations between the learning resources and their use. However, no assessment of the pedagogical value of the resources or of the recommending engine is available yet [7]. QSIA (Questions Sharing and Interactive Assignments) is a system for sharing, assessing, and recommending learning resources that is used by online communities, which has a rather atypical approach, different from mainstream RSs, by putting users in control of the recommendation process. Thus, s/he may choose on whom to advise or whether to use a collaborative filtering service or not [6, 27]. In [28], recommendations based on NN collaborative filtering of learning objects are performed, but the novel twist regards the multi-attribute algorithms that provide for multi-dimensional user ratings of the learning resources. It is interesting to notice that the same algorithms perform differently depending on the context where the testing takes place. RecoSearch, an engine that combines content, collaboration, collaborative filtering and searching techniques provides for a collaborative infrastructure for authoring, searching, recommending, and presenting Java source code learning objects [29]. A web-based learning system that can both find relevant content on the open Web and adapt it to learners and their situated learning characteristics, based on system's observation on learners and their ratings, is presented in [30]. Pro-active recommending of learning objects that combine content and social aspects, and that is able to adapt to learner's profile and his navigation history, relying on an ontology of topics from programming (as an index of the learning objects) is presented in [31]. An interesting idea is put to practice in [32], where authors propose that the users with greater knowledge have greater weight when computing recommendations than the ones with less knowledge. A totally different approach of obtaining the quality rating of learning objects is proposed in [33]. It contains a hybrid approach that includes content-based and collaborative filtering, and that implements a Markov model to verify the quality evaluation of the learning objects. Their system uses Bayesian Belief Networks to overcome the incompleteness and lack of learning object quality reviews, as well as the differences between evaluations of different reviewers. A work that uses CBR to make personalized recommendation in online learning object repositories is [34]. The authors present their combination of content-based filtering techniques with collaborative filtering mechanisms to be applied to a repository with more than 200 programming examples written in different programming languages. Students are expected to include ratings for the learning objects, both existing and new.

One of these works takes into account very few quality aspects when make recommendations [33], while other considers briefly the pedagogical value and other educational aspects of the resources [30]. The majority of them approach closed repositories that contain very specific instructional resources. Therefore, we hope that our approach will contribute to better recommendations of diverse educational resources (particularly OCW and OERs) for a large variety of users.

4 Conclusions and Future Work

We introduced here our hybrid approach of recommending framework for open OCW and OERs that combines case based recommending based on a quality model with (collaborative) feedback from users. The current stage of the project is as follows: we are developing the first prototype, which collects in a common pool of resources several OCW and OERs (around 10 resources per subject) that are necessary to graduate majoring in Computer Science. Additionally, we are evaluating the quality of these resources using the rubrics presented briefly here. Our first goal is to use this prototype for our Computer Science students both in formal and informal environments and to evaluate and, hopefully, validate the viability of our approach. Further on, we consider automating some activities of our framework: the federated search of resources based on the taxonomy, some quality evaluations, capturing context and knowledge about user etc. to obtain a true adaptive recommender system.

References

1. Kortemeyer, G.: Ten Years Later: Why Open Educational Resources Have Not Noticeably Affected Higher Education, and Why We Should Care, *Educase Review* online, <http://www.educause.edu/ero/article/ten-years-later-why-open-educational-resources-have-not-noticeably-affected-higher-education-and-why-we-should-care>
2. Vladoiu, M.: Quality Criteria for Open Courseware and Open Educational Resources. In: 11th ICWL 2012 Workshops, LNCS, vol. 7697, Springer, Heidelberg (2012)
3. Vladoiu, M., Constantinescu, Z.: Evaluation and Comparison of Three Open Courseware Based on Quality Criteria. In: Grossniklaus, M., Wimmer, M. (eds.), 12th ICWE 2012 Workshops – QWE 2012 (3rd Workshop on Quality in Web Engineering), LNCS, vol. 7703, pp. 204-215, Springer, Heidelberg (2012)
4. Moise, G., Vladoiu, M., Constantinescu Z.: MASECO - Multi-Agent System for Evaluation and Classification of OERs and OCW based on Quality Criteria, in press (2013)
5. Nicoara E. S.: The Impact of Massive Online Open Courses in Academic Environments. In 9th Int. Conf. eLearning and Software for Education, Ed. Universitara, Bucharest (2013)
6. Manouselis, N., Drachsler, H., Vuorikari, R., Hummel, H. G. K., Koper, R.: Recommender Systems in Technology Enhanced Learning. In Kantor, P. B., Ricci, F., Rokach, L., & Shapira, B., (eds.), *Recommender System Handbook*, 387-415, Springer, Berlin (2011)
7. Lemire, D., Boley, H., McGrath, S., Ball, M.: Collaborative Filtering and Inference Rules for Context-Aware Learning Object Recommendation. *International Journal of Interactive Technology and Smart Education*, 2(3), 179-188 (2005)
8. Cechinel, C., Sicilia, M.-A., Sánchez Alonso, S., García Barriocanal, E.: Evaluating Collaborative Filtering Recommendations Inside Large Learning Object Repositories. *Information Processing and Management*, 49(1), 34-50 (2013)
9. Zapata, A., Menéndez, V. H., Prieto, M.E., Romero, C.: A Framework for Recommendation in Learning Object Repositories: An Example of Application in Civil Engineering. *Advances in Engineering Software*, 56, 1-14 (2013)
10. Resnick, P., Varian, H. R.: Recommender Systems. *Commun. ACM* 40(3), pp. 56-58 (1997)
11. Adomavicius, G., Tuzhilin, A.: Toward the Next Generation of Recommender Systems: a Survey of the State-of-the-art and Possible Extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6), 734-749 (2005)

12. Burke, R.: Hybrid Web Recommender Systems. In: Brusilovsky, P., Kobsa, A., and Nejdl, W. (eds.) LNCS, Vol. 4321, pp. 377-408. Springer-Verlag, Berlin, Heidelberg (2007)
13. Burke, R.: Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, 12(4), pp. 331-370 (2002)
14. Burke, R.: Knowledge-based Recommender Systems. In A. Kent (ed.), *Encyclopedia of Library and Information Systems*. 69(32), New York: Marcel Dekker (2000)
15. Adomavicius, G., Sankaranarayanan, R., Sen, S., Tuzhilin, A.: Incorporating Contextual Information in Recommender Systems using a Multidimensional Approach. *ACM Trans. Inf. Syst.* 23(1), 103-145 (2005)
16. Manouselis, N., Drachsler, H., Verbert, K.: TEL as a Recommendation Context, *Recommender Systems for Learning*, pp. 21-36, Springer, New York (2013)
17. Buder, J., Schwind, C.: Learning with Personalized Recommender Systems: A Psychological View. *Computers in Human Behavior*, 28, 207-216 (2012)
18. Aamodt, A., Plaza, E.: Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *AI Communications*, 7(1), 39-59 (1994)
19. Smyth B., Case Based Recommendation. In *The adaptive web*, pp. 342-376, Springer-Verlag Berlin, Heidelberg (2007)
20. Lee J. S., Lee, J. C.: Context Awareness by Case-Based Reasoning in a Music Recommendation System, LNCS 4836, pp. 45-58, Springer Berlin Heidelberg (2007)
21. Văduva, I., Albeanu, G., Introduction to fuzzy modelling, Ed. of Univ. of Bucharest (2004)
22. Nesbit, J.C., Li, J.Z. and Leacock, T.L.: Web-Based Tools for Collaborative Evaluation of Learning Resources, *J. of Systemics, Cybernetics and Informatics*, 3(5), 102-112 (2005)
23. Burgos Aguilar, J. V.: Rubrics to evaluate OERs (2011), www.temoa.info/sites/default/files/OER_Rubrics_0.pdf
24. ACHIEVE, <http://www.achieve.org>
25. OER Commons, <http://www.oercommons.org>
26. Vladoiu, M., Constantinescu, Z.: A Taxonomy of Opportunities for Searching, Browsing, and Retrieving OCW and OERs, submitted for publication (2013)
27. Rafaeli, S., Barak, M., Dan-Gur, Y., Toch, E.: QSIA: a Web-based Environment for Learning, Assessing and Knowledge Sharing in Communities. *Computers & Education*, 43(3), 273-289 (2004)
28. Manouselis N., Vuorikari R., Van Assche F.: Simulated Analysis of MAUT Collaborative Filtering for Learning Object Recommendation, in *Proc. of the Workshop on Social Information Retrieval in Technology Enhanced Learning (SIRTEL 2007)*, pp. 17-20 (2007)
29. Fiaidhi J.: RecoSearch: a Model for Collaboratively Filtering Java Learning Objects. *Int. J. Instruct. Technol. Distance Learning* 1(7), 35-50 (2004)
30. Tang T. Y., McCalla G. I.: Smart Recommendation for an Evolving e-Learning System: Architecture and Experiment. *Int. J. E-Learning*, 4, 105-129, (2005)
31. Ruiz-Iniesta, A., Jimenez-Diaz, G., Gómez-Albarrán, M.: Recommendation in Repositories of Learning Objects. In: *The 9th IEEE International Conference on Advanced Learning Technologies (ICALT 2009)*, pp. 543-545 (2009)
32. Bobadilla, J., Serradilla, F., Hernando, A.: Collaborative Filtering Adapted to Recommender Systems of E-learning. *Knowl.-Based Syst.*, 22(4), 261-265 (2009)
33. Kumar V., Nesbit J., Winne P., Hadwin A., Jamieson-Noel D., Han K.: Quality Rating and Recommendation of Learning Objects. In Pierre S, (ed.) *E-learning Networked Environments and Architectures*, Springer London, pp. 337-373 (2007)
34. Gomez-Albarran M., Jimenez-Diaz G.: Recommendation and Students' Authoring in Repositories of Learning Objects: A Case-Based Reasoning Approach, *International Journal of Emerging Technologies in Learning (IJET)*, 4(1), 35-40, (2009)