Advances in Computer Science and Information Technology (ACSIT)
Print ISSN: 2393-9907; Online ISSN: 2393-9915; Volume 2, Number 1; January-March, 2015 pp. 7-12

http://www.krishisanskriti.org/acsit.html

# Recommendation Techniques for Adaptive E-learning

Devanshu Jain[1], Ashish Kedia [2], Rakshit Singla[3] and Sameer Sonawane[4]

*[1]Dept. of ICT DA-IICT Gandhinagar, Gujarat, India*
*[2]Dept. of IT NIT-Karnataka Surathkal, Karnataka, India*
*[3]Dept. of CSE IIT-Hyderabad Hyderabad, Andhra Pradesh, India*
*[4]Dept. of CSE VNIT-Nagpur Nagpur, Maharashtra, India*
*E-mail: [1]devanshu.jain919@gmail.com, [2]ashish1294@gmail.com, [3]rkshtsingla@gmail.com, [4]sameer9311@gmail.com*

**Abstract—***Personalization of learning is the need of the hour. Technology can play an important role in achieving this personalization of learning. While today's Learning Management Systems (LMSs) do facilitate the instructor to make the content available on Internet, yet they don't have any functionality to personalize the learning of the user. The adaptive e-learning technology extends this traditional classroom environment to make the guidance, a one to one mechanism, i.e. single machine guiding a single user through the course material. This paper proposes recommendation techniques to offer courses to the user.*

## 1. INTRODUCTION

Traditional classroom training method is no longer viable as it requires large budgets, extensive planning and logistics. That's why many are shifting their attention to e-learning as a technological solution to this problem. 98% of companies, nowadays, use technological infrastructure (on line learning) to control the delivery and management of training to its employees.[1] Using technology in the learning assists in changing the process from one based on rote to one based on comprehension. [2]

The true power of this educational technology is not just to deliver content. Adaptive e-learning intends to improve the user experience by capturing details about the user like his learning style, his cognitive abilities, knowledge level, interests, personal traits, etc. and provides the user a personalized learning path based on the information captured. As opposed to traditional classroom ideology of *one size fits all*, adaptive e-learning makes learning personal so that the user can trace the best learning curve. The system identifies user characteristics and provides him instructions accordingly. In other words, the goal of the system is *to provide the right content to the right person at the right time*.

The major part of our work is to come up with recommendation techniques to provide the next best favourable content to the user.

## 2. RELATED WORK DONE

There are three major components of an adaptive e-learning system, namely,*Content Modelling, User Modelling and Adaptive Engine*.

**Content Model** is used for domain level representation of the knowledge structure. Chrysafiadi and Virvou[3] suggests an approach for representing the domain knowledge by using Fuzzy Cognitive Maps. The domain knowledge is divided into concepts and there are interdependencies between these concepts. The structure takes the form of a directed graph, where each node represents a concept and arcs between these nodes represent the level of interdependencies among concepts.

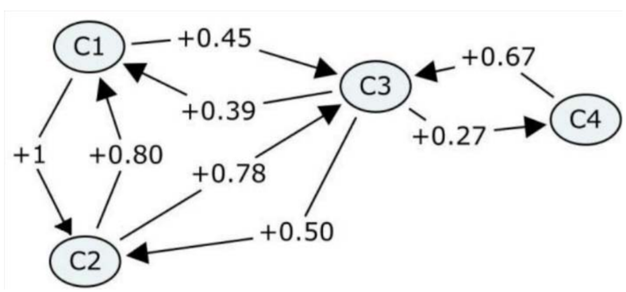


Fig. 1: Fuzzy Cognitive Maps[3]

Content model also describes the forms in which the content is available for its users, for example: e-book, slide shows, videos, animations, etc. This helps in providing the right type of content to the user i.e. the content which is suitable to his cognitive needs and personal preferences.Concept map (FCM) plays an important role as it helps in student assessment, recommendation and remediation. A major challenge in constructing a concept map is to find the relationship between concepts, automatically. It is tedious for an instructor to provide all the relationships, manually which may also be inaccurate. Shih-Ming Bai and Shyi-Ming Chen[4] provides a method to semi-automatize this construction of concept mapping which has further been improved by Shyi-Ming Chen and Po-Jui Sue[5]. The construction requires two types of information namely, how much grade does a student score in every question, denoted by Grade - matrix and how much does a question test the user on a particular concept, denoted by Question Concept - matrix.

1) First, we calculate the similarity between questions' responses by the students, i.e. the counter values on the basis of G (Grade) matrix.Only the pairs of questions, for which the count value is greater than threshold value: n * 40%, are considered for next step.
   Here n is the number of students.
   Consider for example the following grade matrix:

$$G = \begin{array}{c} \\ Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \\ Q_5 \end{array} \begin{array}{c} \begin{array}{ccccc} S_1 & S_2 & S_3 & S_4 & S_5 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

So similarity between Q1 and Q2 is 0 $\oplus$ 0+ 0 $\oplus$ 0+ 0 $\oplus$ 0+ 1 $\oplus$ 0+ 1 $\oplus$ 1=1+ 1+ 1+ 0+ 1=4, which is greater than cut-off, 5 * 0.4=2. Hence this pair, Q1 and Q2 moves to second step.

2) Then the item-set support relationship is calculated. Item set is of four types: 1-item-set for right and wrong support and 2-item-set for right and wrong support.

Table 1: 1-question item set support table

| 1-Question Item Set | Right Support |
|---|---|
| Q1 | 2 |
| Q2 | 1 |
| Q3 | 0 |
| Q4 | 2 |
| Q5 | 3 |

It represents that 2 people have got Q1 right 1 has got Q2 right and so on. It denotes the support for right attempts of each question. Similarly, the support for wrong attempts are also found out.

Then, a two item set support table is constructed, as follows:

Table 2: 2-question item set support table

| 2-Question Item Set | Right Support |
|---|---|
| Q1 & Q2 | 1 |
| Q2 & Q3 | 0 |
| Q3 & Q4 | 0 |
| Q4 & Q5 | 2 |
| Q5 & Q1 | 0 |

It represents that for Q1 and Q2, only 1 has got both of the questions right. It denotes the 2-item set support for right attempts. Similarly, the 2-itemset support for wrong attempts are also found out. Now, we use the following formula to calculate the confidence between questions.

$$Confidence(Qx \to Qy) = \frac{Support(Qx, Qy)}{Support(Qx)}$$

Through this, the confidence level between the questions is established. Confidence levels for two kinds of association rules are found out: one for the correctly attempted and second for the wrongly attempted, as mentioned above. In layman words, the confidence(Q1 → Q2)$_{\text{right}}$ represents that if the student attempts Q1, correctly then what is the probability by which he attempts Q2, correctly. Association rules with confidence level greater than 75% are considered in future steps.

For example,

$$Confidence(Q1 \to Q2)_{\text{right}} = \frac{Support(Q1, Q2)}{Support(Q1)} = \frac{1}{2}$$

3) Now a new Question-Concept matrix is created (QC'),based on below two rules:
   a. If there are two or more nonzero values in column $C_t$ of the questions-concepts matrix QC, then the degree of relevance of question Qx with respect to concept $C_t$ in the constructed questions-concepts matrix QC' is calculated as follows:
   $$qt'_{ut} = \frac{qc_{xt}}{\sum_{u=1}^{m} qc_{ut}}$$

Where m is the number of questions

   b. If there is only one nonzero value in column $C_t$ of the questions-concepts matrix QC, then the degree of relevance of question Qx with respect to concept $C_t$ in the constructed questions-concepts matrix QC' is calculated as follows:
   $$qc'_{xt} = qc_{xt}$$

So, if QC matrix was:

$$QC = \begin{bmatrix} & C1 & C2 & C3 & C4 & C5 \\ Q1 & 1 & 0 & 0 & 0 & 0 \\ Q2 & 0 & 1 & 0.5 & 0 & 0 \\ Q3 & 0.5 & 0 & 0.5 & 0 & 0 \\ Q4 & 0.3 & 0.4 & 0 & 0.3 & 0 \\ Q5 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Thenew matrix will be:

$$QC' = \begin{bmatrix} & C1 & C2 & C3 & C4 & C5 \\ Q1 & 0.555 & 0 & 0 & 0 & 0 \\ Q2 & 0 & 0.714 & 0.5 & 0 & 0 \\ Q3 & 0.278 & 0 & 0.5 & 0 & 0 \\ Q4 & 0.167 & 0.286 & 0 & 0.3 & 0 \\ Q5 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

4) Based on the associative rule Qx → Qy, the relevance between concept $C_i$→ $C_j$, is calculated:

$$rev(c_i, c_j)_{Qx \to Qy} = qc_{xi} * qc'_{yj} * Confidence(Qx \to Qy)$$

Here, $C_i$ denotes a concept in question Qx and $C_j$ denotes a concept in question Qy, $qc_{xi}$ denotes the degree of relevance of question Qx with respect to concept $C_i$ in the questions-concepts matrix QC, $qc_{yj}$ denotes the degree of relevance of question Qy with respect to concept $C_j$ in the questions-concepts matrix QC. Confidence represents the confidence of the association rule Qx→Qy.

So, relevance between concept $C_1$ and $C_2$, on the basis of association rule Q1→Q2 will be:

$$rev(c_1, c_2)_{Q1 \to Q2} = qc_{11} * qc'_{22} * Confidence(Q1 \to Q2)_{right}$$

= 1*0.714*0.5=0.357

5) Calculate a threshold value of the relevance degree $\mu$=MIN($qc_{xt}$), where $1 \le x \le m$ and $1 \le t \le p$, m is the number of questions and p is the number of concepts.
6) If rev$(C_i,C_j)_{Qx \to Qy} < \mu$, then calculate $€_{ij} = N_i + N_j$, where $N_i$ is the number of questions related to concept i. If $€_{ij}$> m*50%, then the relevance relation is retained.
7) Now, in some cases, there are two relevance degree between same pair of concepts - one for the associative rule (correctly learned to correctly learned) and second for incorrectly learned to incorrectly learned. The one with maximum value is chosen.

**Student Model** refers to the method of representing a user in the virtual world. Student Model is used to collect and store user's information like knowledge, misconception, goals, emotional state, etc. This information is then used by the system to determine user's need and adapt itself accordingly. There are two types of information collected [2] - Domain related (related to the context of the course like knowledge about different concepts, misconceptions, etc.) and Domain unrelated (personal traits of the user, i.e. cognitive abilities, learning style, age, sex, etc.). Much of the information stored in a student model is *static* in nature i.e., it remains constant throughout the learning phase such as age, sex, mother tongue etc. Such information is usually collected via questionnaires. All other information is *dynamic* in nature i.e., it changes during the learning phase like knowledge level, performance etc. Such information is available directly via the student's interaction with the system and is constantly updated. Chrysafiadi and Virvou have presented a nice literature of the popular student modelling techniques used in the past decade [6].

We suggest representation of domain related information to be done using an overlay model [7], i.e. the user's knowledge is expressed as a subset of the knowledge domain, which represents the expert knowledge in that domain.

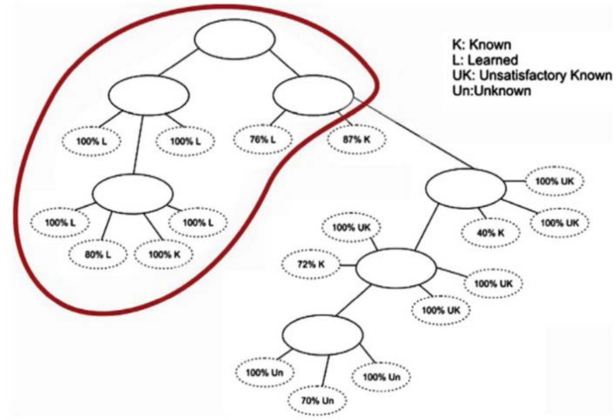


Fig. 2: Overlay Model [8]

User's knowledge, instead of being represented in concrete terms, is represented in an abstract (fuzzy) way, which is more close to human understanding and results in better interpretation. The knowledge is categorized in four fuzzy sets: Unknown (Un), Unsatisfactorily Known (UK), Known (K) and Learned(L). Membership function of each set is described using simple equations as mentioned in [8].

Domain unrelated information can be modelled using Felder-Silverman Learning Style Model (FSLSM)[9]. It distinguishes the user's preferences on four dimensions.

1) Way of Learning - Active learners are the ones who like to apply the learned material and work in groups, communicating their ideas. Reflective workers try to work alone, think about what they have learned.
2) Intuitive and Sensory preferences - Sensing learning style like concrete learning material and like to solve problems using standard approaches. Intuitive learners like totally rely on abstract theories and their underlying meanings.
3) Visual and Verbal preferences - Visual learners are the ones, who prefer learning from what they have seen. They have less memory retaining capacity. Verbal learners are the ones who prefer textual representation (written/spoken).
4) Process of Understanding - Sequential learners learn in small steps and their learning graph is linear. They are more interested in details. Global learners, on the other hand, are more interested in overviews and a broad knowledge.

On the basis of these four dimensions, the user is characterized and an appropriate kind of learning object, which suits his learning style is presented to him. There are two kinds of recommendations, one to offer the way the study material is presented to the user and other, to offer the next concepts to the user. Next section describes four recommendation techniques to offer next concepts to the user.

## 3. OUR CONTRIBUTION

We propose new techniques to offer next concepts to the user once user has completed learning the current concept:

1) Path that observed highest gain in knowledge level
2) Path that students with similar history has taken
3) Concepts in which the student needs revision

## 4. MAXIMUM SUCCESS PATH

Here, we recommend the next concept to the user based on the path from current concept that received maximum success in the past. Because one concept is related to another, hence change in knowledge level of one concept affects user's knowledge level of other concept too. This algorithm recommends the concept which will provide the highest overall average increase in knowledge level across all concepts.

Whenever, a student traverses the edge $C_i \rightarrow C_j$, i.e. he takes the quiz of concept $C_j$ when the last concept done by him is $C_i$, his knowledge level for various concepts is changed based on the quiz result. The average change in knowledge level across all the concepts for the student is recorded.

$$AC_{for\ current\ user} = \frac{\sum_{c_m \in Concepts} KL_m(t+1) - KL_m(t)}{|Concepts|}$$

Here, $KL_m$(t+1)is the knowledge level in concept m after the quiz and $KL_m$(t+1) is the knowledge level in concept m before the quiz.Now the new average change for $C_i$ to $C_j$ is calculated as:

$$AC_{C_i \to C_j} = \frac{AC_{C_i \to C_j} * \left|Students_{c_i \to c_j}\right| + AC_{for\ current\ user}}{\left|Students_{c_i \to c_j}\right| + 1}$$

Here AC stands for average change. Thus, if a user has completed concept $C_i$, all concepts $C_j$ which have not been completed are recommended in the order of decreasing value of average increase of $C_i \rightarrow C_j$ across all users.

## 5. STUDENT SIMILARITY BASED RECOMMENDATION

User-user collaborative filtering has been widely used in e-commerce systems but e-learning is a new platform for it. User based collaborative filtering works around finding similarity between users based on how they rate certain items in the domain. Then it predicts ratings for the current user on unrated items based on how similar users rated those items.

This method can be very effectively used in e-learning systems as similarity between users can be used in recommending courses and concepts.

For calculating similarity between students we use a modified cosine similarity metric:

$$sim(i,j) = \frac{\sum_{c \in c_i \cap c_j} s_{ic} * s_{jc}}{\sqrt{\sum_{c \in C_i} s_{ic}^2}\sqrt{\sum_{c \in C_j} s_{jc}^2}}$$

Where $s_{ic}$ represents the score of $i^{th}$ student in concept cand $C_i$ represents the list of concepts whose test, student i has given.

If we replace both c € $C_i$ and c € $C_j$ in the denominator with c € $C_i \cap C_j$, it is essentially cosine similarity but using it in the given form has an added benefit. It acts as an automatic damping factor and also takes into consideration the cases when two students have a large difference in the total number as well as list of concepts they have each taken. Any concept which is not common will contribute to the denominator but not to the numerator thus reducing the similarity value, which is intuitively correct.

After calculating the similarity, the prediction value for each concept (which is not attempted by the current user) is calculated for the current user.

$$P_{ic} = \frac{\sum_{j \in S} sim(i,j) * s_{jc}}{\sum_{j \in S} sim(i,j)}$$

Where $P_{ic}$ represents Prediction value for Student i in Concept c and the set S represents the set of students who have attempted Concept c. After calculating the prediction values, the concepts whose prediction value is greater than a threshold(can be the passing marks) are recommended in decreasing order of prediction values.

## 6. COLLABORATIVE FILTERING (BASED ON RATINGS)

Collaborative filtering is one of the widely used techniques for recommendation. Collaborative Filtering is an approach to determine the similarity between two items based on ratings provided by other users. It uses the known preferences of a group of users to make recommendations or predictions of the unknown preferences for other users [10]. This is one of the most successful technology for building recommendation systems till date and is widely used. In the proposed recommendation model items are learning objects or material like tutorials or lectures from which a student learns about a concept. This method attempts to predict the utility/suitability of a learning objects to a particular user based on the ratings provided by other users. Once we have predicted the utility of various learning objects, we propose to recommend the top k learning objects to the user [11]. The two key steps involved are as follows:

1) Computing similarity between two items. The most popular techniques used for this step is the Pearson's correlation coefficient [12] and cosine based approach. The simple well-known formula used is:

$$sim(i,j) = \frac{\sum(R_{u,i}-\overline{R_u})(R_{u,j}-\overline{R_u})}{\sqrt{\sum(R_{u,i}-\overline{R_u})^2}\sqrt{\sum(R_{u,j}-\overline{R_u})^2}}$$

Where $R_{u,i}$ is the rating given to $I_i$ by user u, R is the mean rating of all the ratings provided by u. An item-item similarity matrix is created and top k itemssimilar to the last learning object used by the user is chosen.

2) The prediction for each user u in the user-set U correlated with each item i in the item-set I is calculated as follows:

$$P_{u,i} = \frac{\sum_{t\in N}(sim\,(i,t)*R_{u,t})}{\sum_{t\in N}(|sim\,(i,t)|)}$$

Where N represents the item i's similar item set, and $R_{u,t}$ is the rating given to item t by user u.

## 7. RECOMMENDING CONCEPTS FOR REMEDIATION

Recommendations are not only designed to suggest best new concept, but also to suggest concepts, which the user is attempting incorrectly frequently. For this, we suggest the following recommendation technique suggesting the concepts to the user, which he has forgot. The basis of this technique is that any question does not test the user just on one concept. There is a certain degree to which a question judges the student on one concept, as denoted in the Question-Concept (QC) matrix, in the concept-mapping section before.

For every concept for a particular student, we retrieve two parameters:

1) Number of times, $N_i$, concept $C_i$'s questions have been attempted wrongly consecutively
2) The total dependency, $D_i$ among the questions, attempted wrongly for the concept $C_i$

A concept is considered as forgotten if and only if the following condition holds true:

$$N_i \geq M * 30\%$$

Here M is the total number of questions contributing to that concept, $N_i$ is the total number of consecutive wrong attempts in concept $C_i$'s question, $D_i$ is the total dependencies of wrongly attempted questions. We have assigned a revision importance ($R_i$) to every concept that signifies the priority with which the student should revise the concepts in which he has misconception. This parameter is calculated by giving equal weight-age to both the parameters namely $N_i$ and $D_i$, as follows:

$$R_i = 0.5 * N_i + 0.5 * D_i$$

Now the user is recommended concepts in order of decreasing importance ($R_i$) of the concepts.

## 8. EVALUATION MODEL

Evaluation of recommender systems has only lately started to become more important and systematic. In our system, we have implemented a layered evaluation model [13] which decomposes the recommendation model into several layers based on several criteria and then evaluates each layer individually. Since our learning model is based on programming concepts, the recommendation system is broken down into following 5 criteria as used by the PeRSIVA evaluation model [14] which forms the basic framework for our evaluation model - Effectiveness of System, Adaptability of the System, State on Computer Programming, Students' progress in Future, and Necessity of Revision.

The student is provided with a small set of feedback questions each time he/she interacts with the learning model. The responses of the student are collected for several questions over a period of time.The responses of the student are over a scale of range 1(not at all) to 5(very much).The feedback questions are on the above mentioned basic criteria. Based on these responses, the average response for each criteria is calculated and the then the system is judged based on these criteria.

Apart from evaluating the model based on feedback, we have also implemented an evaluation technique to judge the quality of the learning material and the quiz based upon the material.It is very important for students' learning process that the learning material and the quiz based upon that are very much related and the quiz is based on the material. This helps the student to correctly monitor his learning process as well as his knowledge levels in the concepts. Thus, to measure the relation between the material and the quiz, we have inculcated the *accuracy* factor. *Accuracy* can simply be defined as average score of all students in each concept in terms of percentage. The corresponding accuracy and relation table can be depicted as follows:

Table 3: Accuracy and Relation

| Accuracy % | Relation between material and quiz |
|---|---|
| ≥75 | Excellent |
| 50-75 | Good |
| 25-50 | Average |
| <25 | Poor |

Apart from the above two criteria, we have inculcated two very well-known parameters in the domain of evaluation systems - *precision* and *recall* [15]. Precision and recall, according to our model, can be defined as:

$$Precision = \frac{Numnerof\ good\ concepts\ recommended}{Total\ number\ of\ concepts\ recommended}$$

$$Recall = \frac{Numnerof\ good\ concepts\ recommended}{Total\ number\ of\ good\ concepts}$$

By the term- "Good Concept" we mean concepts which have average ratings above 4 in scale of 1-5. Also,"Number of concepts recommended" is the number of recommendations displayed to the learner. The values of precision and recall vary between 0 and 1 and it is often observed that increase in any one of the leads to decrease in the other. Hence, a new parameter which combines both of them is generated and popularly known as the

*F1 metric*. It can be stated as follows:

$$F1\ metric = \frac{2*Precision\ *Recall}{Precision\ +Recall}$$

The *F1 metric* in our model gives equal weight-age to precision and recall. Its value ranges from 0 to 1 and higher its value, better is the recommendation model.

## 9. CONCLUSION AND FUTURE WORKS

Adaptive E-learning is a powerful tool to challenge illiteracy. It removes the requirement for all third party activities like logistics, operational expenses, etc. which act as bottlenecks for efficient imparting of education. But it is unfortunate that the technology's state in the present time is just above that of an on-line lecture, where lecture videos and assignments are published on the Internet and the student can browse through it, without any recommendations.

The system can be further improved by engaging parameters based on context independent information like personal traits and cognitive abilities of the user. NiskosManouselis et al proposed such parameters [16]. The concept mapping can also be fully automated by mining the data from academic articles as proposed by Chen, et al [17]. The collaborative filtering algorithms can also be extended to account for multiple criteria as proposed by Nilashi et al [18].

## REFERENCES

[1] L. Freifeld, “Training magazine ranks 2013 top 125 organisations,” Training Magazine, 2013.

[2] V.Shute and B.Towle, “Adaptive e-learning”, Educational Psychologist, vol. 38, no. 2, pp. 105-114, 2003.

[3] K. Chrysfiadi and M. Virvou, “A knowledge representation approach using fuzzy cognitive maps for better navigation support in an adaptive learning system”, SpringerPlus, vol. 2, no. 1, 2013.

[4] S.-M. Bai and S.-M. Chen, “Automatically constructing grade membership functions of fuzzy rules for students’ evaluation”, Expert Systems with Applications, vol. 35, no. 3, pp. 1408-1414, 2008.

[5] S.-M. Chen and P.-J. Sue, “Constructing concept maps for adaptive learning systems based on data mining techniques,” Expert Systems with Applications”, vol 40, no. 7, pp. 2746-2755, 2013.

[6] K. Chrisfiadi and M.Virvou, “Student modelling approaches: A literature review for the last decade”, Expert Systems with Applications, vol. 40, no. 11, pp. 4715-4279, 2013.

[7] A.C. Martins, L. Faria, C. V. de Carvalho, and E. Carrapatoso, “User modelling in adaptive hypermedia educational systems”, Educational Technology and Society, vol. 11, no. 1, pp. 194-207, 2008.

[8] K. Chrisfiadi and M. Virvou “Evaluating the integration of fuzzy logic into student model of a web-based learning environment”, Expert Systems with Applications, vol. 39, no. 18, pp. 13127-13134, 2012.

[9] S. Graf, S. R. Viola, and T. Leo, “In-depth analysis of the felder-silvermanlearningstyledimensions,”JournalofResearchonTechnology in Education, pp. 79–93, 2007.

[10] [10] T. M. K. Xiaoyuan Su, “A survey of collaborative filtering techniques,” Advances in Artificial Intelligence, vol. 2009, 2009.

[11] [11] M. Deshpande and G. Karypis, “Item-based top-n recommendation algorithms,” ACM Trans. Inf. Syst., vol. 22, pp. 143–177, Jan. 2004.

[12] [12] Y. Li, Z. Niu, W. Chen, and W. Zhang, “Combining collaborative filtering and sequential pattern mining for recommendation in e-learning environment,” in Advances in Web-Based Learning - ICWL 2011 (H. Leung, E. Popescu, Y. Cao, R. Lau, and W. Nejdl, eds.), vol. 7048 of Lecture Notes in Computer Science, pp. 305–313, Springer Berlin Heidelberg, 2011.

[13] [13] S. D. G. Manouselis N., Karagiannidis C., “Layered evaluation in recommender systems: A retrospective assesment,” Journal of e-Learning and Knowledge Society, vol. 10, no. 1, pp. 11–31, 2014

[14] [14] M. V. KonstantinaChrysafiadi*, “Persiva: An empirical evaluation method of a student model of an intelligent e-learning environment for computer programming,” Computers & Education, vol. 68, pp. 322–333, 2013

[15] [15] G. S. AselaGunawardana, “A survey of accuracy evaluation metrics of recommendation tasks,” Journal of Machine Learning Research, vol. 10, pp. 2935–2962, 2009.

[16] [16] N. Manouselis and D. Sampson, “Dynamic knowledge route selection for personalised learning environments using multiple criteria,” in APPLIED INFORMATICS-PROCEEDINGS-, no. 1, pp. 448–453, UNKNOWN, 2002.

[17] [17]N.-S. Chen, Kinshuk, C.-W. Wei, and H.-J. Chen, “Mining e-learning domainconceptmapfromacademicarticles,”ComputersandEducation, vol. 50, no. 3, pp. 1009–1021, 2008.

[18] [18] M. Nilashi, O. bin Ibrahim, and N. Ithnin, “Hybrid recommendation approaches for multi-criteria collaborativefiltering,” Expert Systems with Applications, vol. 41, no. 8, pp. 3879–3900, 2014.