International Journal of Computer Science and Software Engineering (IJCSSE), Volume 7, Issue 6, June 2018
ISSN (Online): 2409-4285 www.IJCSSE.org Page: 130-137

Intelligent Recommendations for e-Learning Personalization Based on Learner's Learning Activities and Performances

Daminda Herath[1] and Lashman Jayarathne[2]

[1, 2] University of Colombo School of Computing, Colombo, Sri Lanka

[1]dherath10@gmail.com, [2]klj@ucsc.cmb.ac.lk

ABSTRACT

In this paper, we present the intelligent recommendations for e-Learning personalization approach which uses recommendation techniques for educational data mining specifically for identifying e-Learners' learning activities, monitoring, predicting performance. Recommended learning resources are computer-based on the current learner's navigational patterns, exploiting similarities and dissimilarities among learners' preferences, educational contents, results obtained in various practical, exercises and interactions with different activities. The proposed framework for intelligent recommendations for e-Learning environment is composed of three modules, a learner module which uses to identify learners' learning activities and preferences, a domain module which contains all the knowledge for a particular discipline and a recommendation module which pre-processes data to build a relevant recommendation list and predicting performances. Recommended resources are obtained by using level of knowledge of learners in different stages and the range of recommendation techniques based on content-based filtering and collaborative approaches. Several techniques such as classification, clustering, predictions and association rules are used to enhance personalization with filtering techniques to provide a recommendation and encourage learners to improve their performance.

Keywords: *E-Learning, Learning Activities, Educational Data Mining, Content-Based Filtering, Collaborative Filtering.*

1. INTRODUCTION

With the rapid growth of the Internet, e-Learning systems are efficiently used for education and training in academic and non-academic contexts. However, most of the e-Leaning systems have not been personalized, several works have addressed the need for personalization in the e-Learning domain. Most of the current e-Learning systems are still delivering the same educational resources in the same way to learners with different profiles [1].

In general, to enable personalization, existing systems use one or more type of knowledge (learners' knowledge, learning materials knowledge, learning process knowledge, etc.) and personalization in e-Learning systems concern adaptive interaction, adaptive course delivery, content discovery and assembly, and adaptive collaboration support. The category of adaptive course delivery presents the most common and widely used collection of adaptation techniques applied in e-Learning systems today [1].

Therefore, personalization plays a significant role in an adaptive e-Learning system. This needs learner profile due to different preferences, learning activities among learners. Due to the huge amount of learning resources on the web, it is hard to find learning resources related to learner request [2].

The recommendation system is an application capable of presenting a user a suggestion for an object, obtained on the basis of his previous preferences and the preferences of a community which has likings and opinions similar his/her. Therefore, recommendation systems help learners to reduce the overload of information that they encounter nowadays, providing, at the same time, customized access to information for a specific domain [3]. In that case, recommendation systems motivate learners to improve their performance as well.

Current e-Learning systems are not providing a better facility to track the learner's progress. It leads learners to interact less with the e-Learning system or keep out from e-Learning. This paper proposes a system with intelligent recommendations for e-Leaning personalization based on learners learning activities and performance. It means personalization approach for providing learning resources for active learners in the e-Learning system. This system recommends some learning resources (learning objects, articles, videos, event details etc.) based on learner's level of knowledge, learner's profile and some other learner's activities). Also, the system provides a facility to track learner progress based on practical tests and exercises (assignments) and monitor the learner's performance in order to guide and support the learners.

2. BACKGROUND OF THE RESEARCH

Current e-Learning systems have several problems. The first problem is the amount of time spent on searching for right content [2]. Learning can take place at any time and place. However, with the expansion of learning resources, it is a time-wasting effort for learners to access desired and suitable resources.

In addition, there is an inadequate search technique for searching the learning resources [2]. In order to solve this, educational data mining techniques are used to enable an implementation that is open, scalable and fast to deploy [1].

Another problem is the absence of personalization in current e-Learning system [2]. Learners with similar learning activities have the similar learning resources, even though, they have dissimilarities in learning activities. Learning profiling for all learners can solve this problem since learner model is created for every learner. The system will recommend learning contents based on the model [2].

Another problem is to offer appropriate learning resources to the right learner in a correct way [2]. The proposed system can solve this problem by implementing the content-based and collaborative filtering approaches. In content-based filtering, the e-Learners are recommended relevant web contents that are similar to the one they preferred or accessed or liked in past. Collaborative filtering, the e-Learners are recommended relevant web contents that are similar to the other e-Learners' preferred or accessed or liked in past [5][6]. These approaches are named information filtering which uses resources to learners. Both of these approaches are based on "rating" or "preference" system [2].

However, both content-based and collaborative filtering suffer the cold-start problem. This problem happens in cases where there is a lack of information about e-Learners and their preferences in the past which makes it impossible to provide relevant recommendations [6][7]. Another limitation of collaborative filtering that it needs a community of learners who know each other. Thus, collaborative filtering is unable to recommend anything. Content-based filtering considers one learner, so the results are not shareable [2]. To overcome the problems with information filtering, the proposed system is aimed at introducing different levels of questionnaires to identify the initial level of knowledge of the learners.

Predicting learner performance and tracking the progress are other challenges in current e-Learning system [8]. Learners take different types of practical, initial, final and assignment tests to continue their learning process in order to gain knowledge. But current e-Learning systems are lacking predictive facility, even learner cannot check the progress as well. In order to address these issues, regression method is used to predict results and subsystem is introduced to check the progress of each e-Learner based on the above mentioned method.

The proposed system helps learners to choose and find learning resources they want to learn. Learner's learning activities such as leaner's history, navigational pattern, preferences, knowledge level, results of practical, initial level, final level, assignment tests, and various activities are stored in learner profile. Learner profile will be updated by the system dynamically based on relevant interactions on relevant interactions by the learner through the system. Then the system will evaluate learner's preference. Finally, learners have interacted with the user-friendly system.

3. LITERATURE REVIEW

Bourkoukou et al. [7] proposed a personalized E-learning system, which takes the learner's personality into account and uses collaborative filtering method for the recommender system. In this model, some modules for personality recognition and selecting an appropriate learning scenario for learner's personality are presented.

In another research, Essaid et al. [9] proposed a personalized e-learning system LearnFit which can which takes the dynamic learner's personality into account. In this system, some modules for personality recognition and selecting appropriate teaching strategy were used to achieve the learning. The results indicate that placing the learner beside an appropriate teaching style matching with learner's preference led to improvement and made the virtual learning environment more enjoyable.

Thai-Nghe [8] proposed a novel approach which uses recommender system techniques for educational data mining, especially in predicting student performance. They also proposed how to map the educational data to user/item in recommender systems. To validate this approach, they compared recommender system techniques with traditional regression methods such as logistic regression by using educational data. Experimental results showed that the proposed approach can improve the prediction results.

In this research, KHRIBI et al. [4] described a fully automatic learner modeling approach in learning management systems, taking into account the learners' educational preferences including their learning styles. They proposed a composite learner model made of three components: the learner's profile, learner's knowledge, and learner's educational preferences. The learner's profile represents the learner's general information such as identification data, the learner's knowledge captures the learner's interests on visited learning objects, and the learner's educational preferences are composed of the learner's preferences (in terms of the specific attributes of the visited learning objects) and his/her learning style. In the proposed approach, all the learner model components are automatically detected, without requiring any explicit feedback. All the basic learners' information is inferred from the learners' online activities and usage data, based on web usage mining techniques and a literature-based approach for the automatic detection of learning styles in learning management systems. Once learner models are built, the proposed system applied a hierarchical multi-level model-based collaborative filtering approach, in order to gather learners with similar preferences and interests in the same groups.

4. RESEARCH METHODOLOGY

Generally, intelligent recommendations for e-Learning personalization system consists of three main components, Domain Model, Learner Model, and Recommender Model. Fig. 4.1 shows the overall architecture of the proposed system.

4.1 Domain Model

A domain model contains the knowledge about the curriculum structure. This model is split into three layers, the first represents the course and each course is divided into several concepts, and each concept is presented by a set of learning objects.

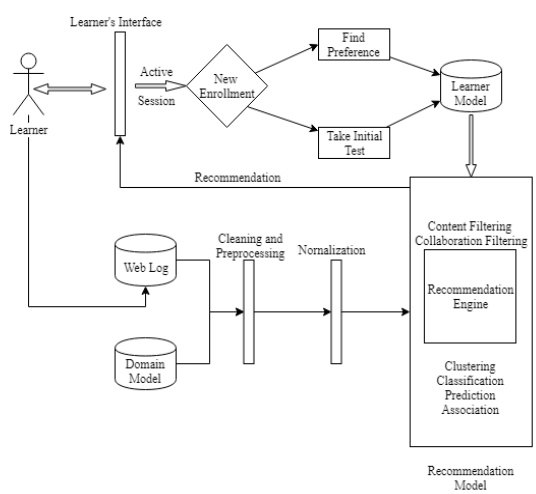

Fig. 4.1. The Overall Architecture of the of the Proposed System

A learning object holds one unit of knowledge and presents different aspects such as lecture notes, presentations, questions, activities, examples, exercises etc. [9]. In this research, "Introduction to Python Programming" course has been selected to explain course structure as below table 4.1.

Table 4.1: Introduction to Python Programming Course structure

Course : Introduction to Python Programming	
Concept 1: Introduction to Python	
Learning Object 1	Overview of Python
Learning Object 2	Basic Syntax
Learning Object 3	Variables and Data Types
Concept 2: Operators and Expressions in Python	
Learning Object 1	Types of operators
Learning Object 2	Operators precedence
Concept 3: Selection Control Structure in Python	
Learning Object 1	if, if else and Multi-way selection
Learning Object 2	Nested selection
Learning Object 3	Exception Handling
Concept 4: Selection Control Structure in Python	
Learning Object 1	The for loop
Learning Object 2	The while loop
Learning Object 3	Nested loop
Learning Object 4	break and continue
Concept 5: List, Tuple, Dictionary, Number and String	
Learning Object 1	Lists
Learning Object 2	Tuples
Learning Object 3	Dictionaries
Learning Object 4	Number
Learning Object 5	String
Concept 6: Functions	
Learning Object 1	Defining and Calling a function
Learning Object 2	Function Arguments
Learning Object 3	Return Statement
Learning Object 4	Scope of variables and The anonymous functions

Concept 7: Handling Files I/O	
Learning Object 1	Reading keyboard Input
Learning Object 2	Opening and Closing files
Learning Object 3	Reading and Writing files
Learning Object 4	File and Directory Handling

Each course includes the different level of the tests to identify the learner level of knowledge. Each level of test has a set of questions. Those questions have different types of proficiency levels. Table 4.2 describes the test structure and proficiency with allocated 5-energy points (from 30=very hard to 10=very easy).

Table 4.2: Test Structure

Test Name	***No of Test per Course***	***No of Questions per Test***	***No of Attempts***
Initial Level	No of concepts per Course (nCn)	nCn * 5	1
Practical	No of Learning Objects per Course (nLo)	nLo * 5	any
Final Level	No of concepts per Course (nCn)	nCn * 10	Maximum 2
Assignment	1	nCn * nLo * 5	1

4.2 Leaner Model

The learner model represents the various characteristics of the learner such as personal information, preferences, navigational patterns, accessed contents, level of knowledge, etc. which can be used to generate an individualized learning experience [7]. In our research, the learner who enrolls in a particular course is going to take questions (Initial level test) to determine the initial level of knowledge and build the learner profile. Apart from that, learner preferences are used to present the learner profile as well [9].

4.3 Recommendation Model

The proposed recommendation model has two modules, respectively 1) intelligent recommender module and 2) prediction module. In intelligent recommender module, if it is a new learner, the proposed system invites the learner to take the initial level test in order to build learner profile based on learning activities. Once the learner completes the initial test, the result is stored in learner model and then the system generates the recommendation list for specific learner based on the result. Then, the learning process can be started. We can overcome the cold-start problem in recommendation system. A common problem in recommender systems is the cold start problem. It occurs when the new user is logged into the system. Due to lack of ratings of the new user, it is impossible to calculate the similarity between her/him and other users and thus the system cannot make accurate recommendations [10].

Once the learner interacts with the system, data mining techniques such as clustering, classification, regression, association use to collect information about learner's learning activities such as navigational patterns, preferences, accessed contents, bookmarks etc. to build learner profile and to generate an intelligent recommendation. In this module, there are four steps to follow such as cleaning and preprocessed, Normalization, Similarity Computation and Recommendation [7].

The intelligent recommender module helps to generate suitable recommendations to learners based on learning activities. This module uses content-based filtering and collaborative filtering to do that. First, we apply the content-based filtering approach, the term vector is submitted in order to compute recommendation list. Results are ranked according to the cosine similarity of their content (vector of TF-IDF weighted terms) with submitted term vector. Second, we apply the collaborative approach in order to classify the active learner in one of the learner's groups [1].

4.4 Recommendation Process

Learners' initial preferences tend to be noisy. Hence relevant courses/Articles/Videos should be extracted from them. In order to do this, it is important to apply mathematical functions to them so that items can be selected based on some criteria, such as similarity, hence we use vector space model to represent learner's initial courses/articles/videos preferences.

Vector Space Model (VSM) is used to represent documents in a multi-dimensional algebraic manner to apply mathematical functions to the document. It represents a document as a vector. The vector is capable of containing sub-vectors within it. Each attribute of the document is considered as an individual vector. In the context of the given problem of the research, an item (course/article/video) is considered as a vector, and its attributes such as keywords/learning objects will be sub-vectors. Each item is considered as a point in the vector space and it assumes that the most relevant or similar items are the nearest ones. To compare the course/article/video, their relevant sub-vectors are compared with each other and similarity measured using Cosine Similarity and TF-IDF weights. Therefore, we adopt the same Vector Space Model into our research [11].

Time-Frequency (TF) can be represented as $tf_{t,d}$ is the frequency of a particular term t within a given document d. The equation for TF weight is as follows.

$$w_{t,d} = 1 + \log_{10} tf_{t,d} \;\; if \; tf_{t,d} > 0 \; otherwise \; 0 \quad (1)$$

Document Frequency (DF) gives the number of documents that contain a particular term t and is represented as df_t. Inverse Document Frequency (IDF) on the other hand reduce the prominence of highly used terms and gives an important to less frequently used items as well.

$$idf_t = \log_{10}\left(\frac{N}{df_t}\right) \quad (2)$$

TF-IDF weight is the product of both TF and IDF weights and provides the term specific weight of the system and this value is used in obtaining the Cosine similarity.

$$\mathrm{w}_{t,d} = (1 + \log tf_{t,d}) \times \log_{10}\left(\frac{N}{df_t}\right) \quad (3)$$

Cosine similarity provides means of calculating the similarity between two vectors. We can leverage it to compute the similarity between two courses based on a particular feature p. As we use TF-IDF weights in calculating the Cosine similarity, the learner profile will contain a reasonable amount of similarity as well as diversity and thereby by solving the overspecialization issue.

$$sim^p(\vec{m}_i, \vec{m}_j) = \frac{\vec{m}_i \bullet \vec{m}_j}{|\vec{m}_i||\vec{m}_j|} = \frac{\vec{m}_i}{|\vec{m}_i|} \bullet \frac{\vec{m}_j}{|\vec{m}_j|} = \frac{\sum_{n=1}^{t} w_{n,i,p} \cdot w_{n,j,p}}{\sqrt{\sum_{n=1}^{t} w_{n,i,p}^2} \cdot \sqrt{\sum_{n=1}^{t} w_{n,j,p}^2}} \quad (4)$$

$\vec{m}_i$- item i,
$W_{n,i,p}$- tf-idf weight of item i based on attribute p
$\vec{m}_j$ - item j,
$W_{n,j,p}$- tf-idf weight of item j based on attribute p

By comparing the similarity of every course in the learner's the initial preference list with the rest of the courses in the same list, we can find out what are the courses that give the highest similarity value with the TF-IDF weighting scheme. Out of the items in the learner's initial preference list, top 10 items will be added to the learner profile. Therefore the courses in the learner profile contain a reasonable amount of similarity [11].

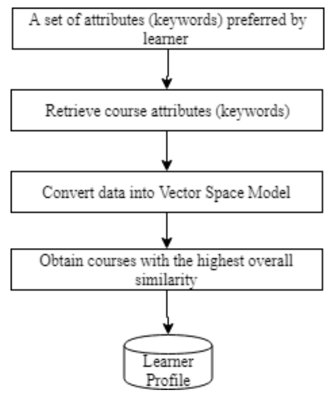

Fig. 4.2. Creating a Learner Profile

4.5 The Utility Matrix

Recommendation systems deal with users and items. A utility matrix offers known information about the degree to which a user likes an item. Normally, most entries are unknown, and the essential problem of recommending items to users is predicting the values of the unknown entries based on the values of the known entries [12].

Example 1. After weighing courses with keywords, we obtained a Course-Keyword Matrix with n rows in which n denotes the number of courses $C=\{c_1,c_2,c_3,\ldots.c_n\}$, and m columns denote the number of keywords $K=\{k_1,k_2,k_3,\ldots. k_m\}$.

Table 4.3: Course-Keyword Matrix

	k_1	k_2	k_3	k_4	k_5
c_1	0	0	0	1	1
c_2	1	1	0	0	0
c_3	1	0	1	0	0
c_4	1	1	1	0	0

The number of times an attribute value occurs within a single item can only occur once (1) or not occurs at all (0).
Example 2. After weighting learning resources, we obtained a preference model for each learner defined as a Learner-Learning Object Rating Matrix with n rows in which n denotes the number of learners $L=\{l_1,l_2.....l_n\}$, and m columns denote the number of learning objects $J=\{j_1, j_2, ..., j_m\}$.

Table 4.4: Learner-Learning Object Rating Matrix

	j_1	j_2	j_3	j_4	j_5
l_1	4	0	1	4	3
l_2	2	1	0	0	0
l_3	4	0	2	2	3
l_4	5	4	4	0	0

This matrix uses a 0-to-5 rating scale where: 5 means that the learner is strongly satisfied with the selected learning object, 1 indicates that the learner is not at all satisfied with the learner object, and finally the score 0 indicates that the learning object is not yet explicitly rated or used at all [7].
Finally, prediction module in recommendation model predicts future results for the different tests. Learners can check progress individually and compare with other learners. To implement prediction module, the system uses linear regression algorithm.

5. EXPERIMENT RESULTS

The In order to implement and evaluate the proposed personalization approach, we used "Introduction to Python Programming" course as a prototype. We selected a set of students who are following Bachelor of Degree in Information Technology at University of Colombo School of Computing to determine learning activities and performances. Those students had a little or lack of knowledge about python programming.
The results show that performance of students improves significantly as progress in the above mentioned course and also their learning activities were high during the studies. The following figures show the relevant evidence such as learner's individual performance, recommendations list based on the learner's interactions, result prediction of the learner, learner's activities, learner's login frequency etc. Finally, it shows the comparison with other learners as well.

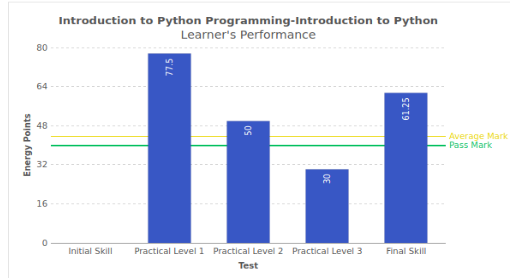

Fig. 5.1. Learner's individual performance based on Introduction to Python Concept

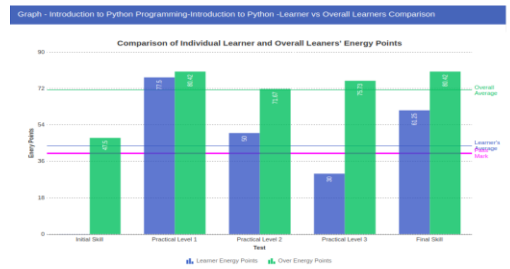

Fig. 5.2. Comparison of Individual Learner and Overall Learners' Energy Points based on Introduction to Python Concept

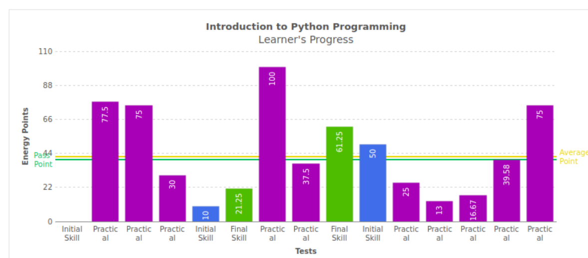

Fig. 5.3.Learner's Progress in Introduction to Python Programming Course

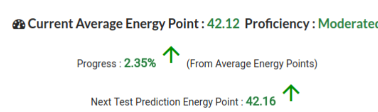

Fig. 5.4. Learner's Next Result Prediction

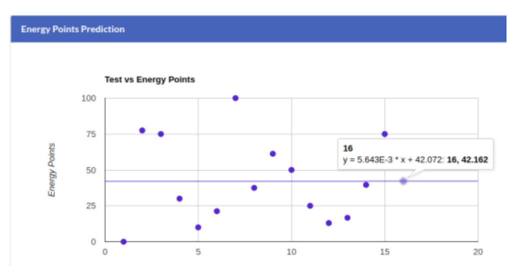

Fig. 5.5. Learner's Energy Points Scatted plot in Introduction to Python Programming Course

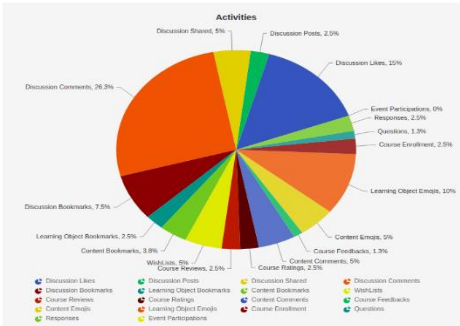

Fig. 5.6. Learner's Activities

Level 3

Star Points : 502

Contribution	Points earned
Course Enrollment	5
Course Rating	1
Course Review	5
Course Feedback	5
Learning Object Access	16
Learning Object Expression	8
Post a Discussion	10
Content Expression	3
Response to Q&A	4
Complete Initial Skill Test	120
Complete Practical Test	125
Complete Final Skill Test	80

Fig.5.7. Learner's Earned Points

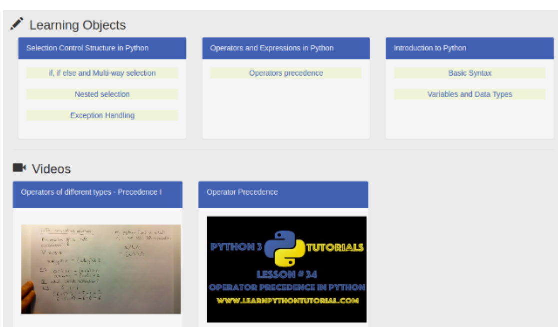

Fig.5.8. Given Recommendations to Learner

The analysis reveals that the average energy points in initial level attempt in each quiz of the concept are lower than the average energy points in other tests. It is due to the reason that the initial level attempts of quiz started with zero or beginning level of knowledge of the learners. Then the system recommended suitable learning objects, articles, videos to repeat the lesson and improve their knowledge based on their individual performance and other learners' performances. Practical tests are provided to fine-tune learner's knowledge about referred learning resources in a systematic way. The process continued till learner achieved acceptable energy points in the final level attempts. Finally, learner had to take assignment quiz to evaluate the final result.

During the e-Learning process, the system encouraged the learners to participate different activities such as taking various quizzes, discussing topics, rating, reviewing, adding bookmarks etc. to and earn points. Fig 5.6 showed the earned points individually by the learner. Therefore, this point indicator motivated the learner to interact with the system more and more.

6. FUTURE WORK

In future work, we will develop an academic repository of different learning resources with question databank to create more adaptive and adaptable e-Learning environment.

7. CONCLUSION

E-Learning environment plays an important role in today's education. As the number of learning resources becomes very large, providing personalized resource recommendation is a significant functionality for today's e-Learning systems. Therefore, the recommendation systems are one of the best tools to deal with the problem of overload information which will help users to find optimal interested items.

We proposed the intelligent recommendations for e-Learning personalization system, which takes the learner's learning activities into account and uses content-based filtering, collaborative filtering and educational data mining methods for recommendations and predictions. Here, we try to overcome the cold-start problem by introducing the initial level test to determine the initial profile of new learner.

In this research, the system evaluates learner's level of knowledge, learner's learning activities, and leaner's performances. Then, the system presents recommendation list according to the results of learner's evaluation and profile.

In order to evaluate the proposed system, we implemented a prototype among selected learners. Results showed that using the proposed approach could improve the performance of learners significantly.

ACKNOWLEDGMENT

We wish to convey our thanks to the academic and non-academic staff at University of Colombo School of computing, Colombo, Sri Lanka for the support they have given throughout the research. We also like to show our gratitude to all the participants who were involved in our research work.

REFERENCES

[1] M. K. Khribi, M. Jemni and O. Nasraoui, "Automatic Recommendationsfor E-Learning Personalization Based on Web Usage Mining Techniques and Information Retrieval," 2008 Eighth IEEE International Conference on Advanced Learning Technologies, Santander, Cantabria, 2008, pp. 241-245.

[2] S. Shishehchi, S. Y. Banihashem and N. A. M. Zin, "A proposed semantic recommendation system for e-learning: A rule and ontology based e-learning recommendation system," 2010 International Symposium on Information Technology, Kuala Lumpur, 2010, pp. 1-5.

[3] S. Martinez, P. Bustelo, G. Crespo and T. Franco, Using Recommendation System for E-learning Environments at degree level. International Journal of Interactive Multimedia and Artificial Intelligence, Volume 1, Issue 2, 2016, pp.67-70.

[4] M. K. Khribi, M. Jemni, O. Nasraoui, S. Graf and Kinshuk, "Toward a Fully Automatic Learner Modeling Based on Web Usage Mining with Respect to Educational Preferences and Learning Styles," 2013 IEEE 13th International Conference on Advanced Learning Technologies, Beijing, 2013, pp. 403-407.

[5] W. Chughtai, S. Ali and G. Imran, Short Systematic Review on E-Learning Recommender Systems. Journal of Theoretical & Applied Information Technology, Volume 57, Issue 2, 2017, pp.139-148.

[6] C. Yang, H. Hsu and J. Hung, “A Web Content Suggestion System for Distance Learning,” Tamkang Journal of Science and Engineering, Volume 9, Issue 6, 2006, pp.243-254.

[7] O. Bourkoukou, E. Bachari and M. Adnani, “A Personalized E-Learning Based on Recommender System. International Journal of Learning and Teaching,” Volume 2, Issue 2, 2016.

[8] Thai-Nghe N., Drumond L., Krohn-Grimberghe A and Schmidt-Thieme L., “Recommender system for predicting student performance,” In Procedia Computer Science, Volume 1, Issue 2, 2010, pp 2811-2819.

[9] E. Bachari, E. Abelwahed and M Adnani, “E-Learning personalization based on Dynamic learners' preference,” International Journal of Computer Science and Information Technology, Volume 3, Issue 3, 2011, pp.200-216.

[10] M. Bahadorpour, B. Soleimani Neysiani and M. Nadimi Shahraki, Determining Optimal Number of Neighbors in Item-based kNN Collaborative Filtering Algorithm for Learning Preferences of New Users. Journal of Telecommunication, Electronic and Computer Engineering (JTEC), Volume 9, Issue 3, 2017.

[11] L. Jayaratne, “Content Based Cross-Domain Recommendation Using Linked Open Data,” GSTF Journal on Computing (JoC), Volume 5, Issue 3, 2017, pp.7-15.

[12] J. Dean, “Big Data, Data Mining, and Machine Learning,” John Wiley & Sons, Inc., 2014, pp. 163-173.