ATHABASCA UNIVERSITY

IMPROVING RECOMMENDER SYSTEMS FOR LEARNING: A DEEP DIVE INTO DESIGNING AND EVALUATING EDUCATIONAL RECOMMENDER SYSTEMS

BY

MOHAMMAD BELGHIS-ZADEH

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN INFORMATION SYSTEMS

FACULTY OF SCIENCE AND TECHNOLOGY

SCHOOL OF COMPUTING AND INFORMATION SYSTEMS

ATHABASCA, ALBERTA

MAY, 2020

Approval of Thesis

The undersigned certify that they have read the thesis entitled

IMPROVING RECOMMENDER SYSTEMS FOR LEARNING: A DEEP DIVE INTO DESIGNING AND EVALUATING EDUCATIONAL RECOMMENDER SYSTEMS

Submitted by

Mohammad Belghis-Zadeh

In partial fulfillment of the requirements for the degree of

Master of Science in Information Systems

The thesis examination committee certifies that the thesis
and the oral examination is approved

Supervisor:
Dr. Sabine Graf
Athabasca University

Committee Members:
Dr. Maiga Chang
Athabasca University

Dr. Marko Tkalcic
Free University of Bolzano

External Examiner:
Dr. Charalampos Karagiannidis
University of Thessaly

June 8, 2020

Dedication

This work is dedicated to my loving parents, Mahmoud Belghiszadeh and Showkat Sayyad for their love, endless support, and encouragement. This work is also dedicated to my wife, Elham Iravani for supporting me each step of the way.

Acknowledgments

My sincere gratitude and special deepest thanks to my supervisor, Dr. Sabine Graf, for her unwavering support of my research, for her patience, motivation, enthusiasm, and immense knowledge. Her guidance helped me in all the time of research and writing several papers and this thesis. I could not have imagined having a better advisor and mentor for my master's study.

Besides my advisor, I would like to thank the rest of my thesis committee: Dr. Maiga Chang and Dr. Marko Tkalcic, for their insightful comments and encouragement.

Moreover, I would like to thank Dr. Hazra Imran for giving me the opportunity to work with her on two research projects during my study.

I would also like to acknowledge the following organizations for their financial support through scholarships, awards, and grants: Athabasca University, Alberta Innovates – Technology Futures (AITF), Ministry of Advanced Education of Canada, the National Science and Engineering Research Council of Canada (NSERC) [funding reference number: 402053-2012-RGPIN], and Mitacs.

In addition, I would like to thank my colleagues and friends for their encouragement and support during my studies: Tom Everette, Stephen Benteau, Monica Blake, Peihao Jin, Rich Yoon, Leonard Wei, Temer Elgamal, Derek D'Amelio, Mehdi Nourizad, Navid Mellatshahi, Jaiganesh Gunasekaran, Maral Zafar Allahyari and all my colleagues at Critical Mass, CP, Vistaprint and Precima.

Finally, I must express my very profound gratitude to my parents, my brothers, my sister, and to my wife for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Abstract

Learning management systems (LMSs) are popular tools that are used in e-learning, however, these systems are still suffering from the lack of personalization. This thesis focuses on designing, developing and evaluating educational recommender systems as one of the tools that can be utilized to enhance the functionality of LMSs with personalization. In this thesis, first, the evaluation of two previously built recommender systems (RUBARS and PLORS) in the areas of learner-centered education and learning object recommendation was conducted. The outcomes of the evaluations showed very promising results and indicated that these systems potentially fill a gap in their respected areas. Next, as the main focus of this thesis, a new recommender system called WEBLORS was designed, developed and evaluated. WEBLORS is an adaptive web based recommender system that aims at providing learners with additional recommended, personalized and relevant learning objects from the web. The evaluation of WEBLORS showed very encouraging results. Based on the results of the evaluation, WEBLORS has a very high potential to help learners by recommending extra personalized recommendations from the web and helping them with information overload by only recommending learning objects relevant to the topic that is being studied and which fits students' profiles.

Keywords: Learning management systems, Recommender systems, Personalization, Information overload

Table of Contents

List of Tables

List of Figures

Chapter 1. Introduction

In this chapter the problem statement and the objectives of this thesis are discussed and the structure of the thesis is described.

1.1 Problem Statement

Enhancing and growing of the internet in recent years has changed the nature of the learning and teaching processes. Today's teaching and learning approaches are more collaborative and are transforming from being teacher-centered to learner-centered (Ibrahim & Fadzil, 2013; Ramu, Taib, Aziz, & Ismail, 2018). As a result of this shift and due to the increase in popularity of e-learning, learning management systems (LMS) have become more popular and are used by schools and universities to deliver learning materials to learners (Rhode, Richter, Gowen, Miller, & Wills, 2017). However, the lack of personalization and adaptivity is one of the main issues with LMSs (Arens-Volland et al., 2019; Saleh, Salama, & Bokhary, 2018). Due to this issue, integrating personalization features into commonly used LMSs has got lots of attention from the scientific community (Saleh et al., 2018) and there is a vast demand for tools and add-on modules that can enhance the functionality of the LMSs with personalization (Arens-Volland et al., 2019). The research in this thesis focuses on designing and evaluating educational recommender systems, which are one of the tools that can be utilized to add personalized features into LMSs.

As mentioned above, one of the main issues with LMSs is the lack of personalization. Although different students have different profiles, preferences and personal characteristics, LMSs usually contain pre-created learning materials that are presented to all students in the exact same way (Bradáč, Šimík, Kotyrba, & Volná, 2017; Heng, Sangodiah, Muniandy, & Yuen, 2018). Considering the differences in learner's background, goals, strengths, personalities, interests, and preferences, adaptivity and personalization have become important factors and concerns in learning management systems (Klašnja-Milićević, Ivanović, & Nanopoulos, 2015; Shobana & Kumar, 2018).

Adaptivity and personalization concepts in the education domain have been researched and improved by different researchers for many years. Personalization in learning is defined as approaches that are used to provide learners with the materials and content that best match the learners' profile (Khribi, Jemni, & Nasraoui, 2009). These approaches consider students' profiles to decide which content should be presented to each learner (Khribi et al., 2009). According to Kinshuk and colleagues, personalization can be attained through adaptive approaches (Kinshuk, Chang, Graf, & Yang, 2010). In the literature, adaptivity in the context of learning systems is defined as the ability of the system to adjust itself to different situations and conditions in order to deliver a better learning experience to the learners while increasing the success rate of the learners in absorbing and understanding of the delivered concepts and materials (Cahyani, Basuki, Sari, & Kustiyahningsih, 2015). In other words, an adaptive learning system has the ability to change its behavior based on different criteria to provide the students with personalized content to address their needs (Vandewaetere, Vandercruysse, & Clarebout, 2012).

Research has shown that adaptivity can be applied based on different attributes of the learners such as prior knowledge, learning styles, and learners' behavior while interacting with the system (i.e. the number of attempts on different learning objects, time spent on activities, etc.) (Vandewaetere et al., 2012). In the context of learning, there are three adaptive approaches that have been introduced: (1) adaptive content selection, (2) adaptive navigation support and (3) adaptive presentation (Manouselis, Drachsler, Verbert, & Duval, 2012).

- Adaptive Content Selection: This approach searches for the items that are requested by the user and adaptively finds the items that are related to previously requested items and are most interesting to the user (Brusilovsky, 2003).

- Adaptive Navigation Support: In this approach, links are adaptively hidden or shown to the user to reduce the number of required clicks in order to find the required information (Brusilovsky, 1996).

- Adaptive Presentation: In this approach, the focus is on content presentation rather than searching information. In other words, in adaptive presentation, the content is adaptively presented to the learner in a way that is most interesting to that user (Paris, 1988).

Recommender systems are one of the tools that can be used to incorporate adaptivity into LMSs. In general, recommender systems are defined as tools that help users to make decisions by making suggestions or recommending contents or services to be used by the user (Burke & Ramezani, 2011; Zhong, Xie, & Wang, 2019). In the context of learning in particular, recommender systems are defined as techniques that help students to reach their learning goals faster by finding and providing learning materials and resources that satisfy their needs (Fraihat & Shambour, 2015). The idea of recommender systems in the learning domain has been around for a long time (Manouselis et al., 2012) and different recommender systems have been built using different approaches and algorithms to serve different purposes (Bobadilla, Ortega, Hernando, & Gutiérrez, 2013; Burke, 2007; Fraihat & Shambour, 2015; Manouselis et al., 2012; Wen, Fang, & Guan, 2012). Based on the citation analysis report of the Web of Science database, in recent years, providing recommendations in e-learning has become a hot research topic and more and more researchers have become attracted to conduct researches in this area (Zhong et al., 2019). Despite all the work that has been done in this area, due to differences in learners' profiles, interests, learning styles, and learning capabilities, delivering personalized content to learners is still a great challenge (Shobana & Kumar, 2018).

1.2 Objectives

The research in this thesis is divided into two parts. The first part aims at evaluating two previously built recommender systems, namely RUBARS and PLORS. The second part which is the main focus of this thesis is focused on designing, developing and evaluating an adaptive web-based learning object recommender system called WEBLORS.

As mentioned above, the first part of the research in this thesis is focused on evaluating two recommender systems that have been previously built to address different issues in LMSs. During my masters' study, I was part of a team that created these two recommender systems for learning to support learners in their learning process (Imran, Belghis-Zadeh, Chang, Kinshuk, & Graf, 2014a, 2016).

The first recommender system is RUBARS that was built to support learners in learner-centered education where learners are responsible for setting their own learning goals (Jonassen, 2000) and choosing the learning materials and activities that are beneficial to them and help them achieve their goals (Pedersen & Liu, 2003). For example, in learner-centered education, there might be some assignments that contain many learning tasks and activities and learners would be responsible for choosing the tasks that they like to complete in order to successfully finish the assignment. However, due to lack of knowledge, some learners might have difficulties to choose the activities that fit best for them. RUBARS is a recommender system that supports learners in such situation by providing recommendations of learning tasks within assignments to learners and helps them choose the tasks that are beneficial for them among many available tasks in an assignment (Imran, Belghis-Zadeh, Chang, & Graf, 2015; Imran et al., 2014a).

The second recommender system is PLORS which has been created in the area of learning object recommendation. PLORS was built to help learners by recommending the available learning objects within the course in a sequence that fits learners' profiles (Imran et al., 2016; Imran, Hoang, Chang, & Graf, 2014b).

The aim of the first part of this thesis is to evaluate RUBARS and PLORS using user studies to capture users' feedback regarding these systems. In this thesis, both systems are evaluated based on their (1) recommender system acceptance, (2) ease of use and user friendliness, and (3) user satisfaction using qualitative and quantitative methods.

The second part of the thesis is focused on one of the new research trends for learning object recommendations that is extending the search and recommend personalized learning objects from the web (Al Abri & Dabbagh, 2018) while helping learners to overcome the information overload. This part is the main part of the research in this thesis and aims at creating an adaptive recommender system called WEBLORS that discovers relevant learning materials from the web and delivers them to LMS users to help them in their learning process. To date, many learning object recommender systems such as PLORS (that is evaluated in the first part of the thesis) are limited to recommend the available content and learning objects that either have been created in the course or have been collected in learning object repositories. Thus, these systems have access to the limited number of learning objects to recommend to learners. WEBLORS tends to address this limitation. The goal of this recommender system is to provide LMS users with personalized recommendation of learning materials from the web that are relevant to the topic that the student is currently learning. In addition, in this part of the research, WEBLORS is evaluated based on its

(1) keyword accuracy, (2) recommendation accuracy, (3) recommender system acceptance, (4) ease of use and user friendliness, and (5) user satisfaction using two simulations and a user study.

One thing to note about the two parts of this thesis is that although PLORS and WEBLORS are both learning object recommender systems, they have different focuses and different aims and there is no similarity between them in terms of the approach and algorithms that are used in these systems.

This thesis focuses on answering four research questions:

Q1: Can a recommender system help learners to choose learning tasks that match their profiles within assignments?

Q2: Can a recommender system be utilized to recommend learning objects in the course to learners in a personalized sequence?

Q3: How can web mining techniques be utilized to discover relevant learning objects on the web?

Q4: How can a recommender system be designed and used to deliver the best matching learning objects discovered from the web to students based on their profiles, learning styles and the ratings given by other learners?

1.3 Structure of the thesis

This thesis is organized into eight chapters. The first chapter is the introduction to the research topic. It discusses the research motivation, summarizes the problem, and announces the research questions. Chapter 2 explains the format of the thesis and discusses the relevance of the manuscripts that have been published or submitted for publication. Chapters 3 to 6 consist of the manuscripts. Chapter 7 provides a discussion about the results and, finally chapter 8 concludes the thesis by summarizing the main contributions of the thesis and discusses future research directions.

Chapter 2. Thesis Format

This manuscript-based thesis describes the results of original research completed during my graduate studies and fulfills the requirements for the Master of Science in Information Systems at Athabasca University. For the evaluation studies conducted within this thesis, the application for ethics approval were submitted and approved by Athabasca University (file number: 21778). The thesis is comprised of four manuscripts.

The first manuscript is titled “Evaluation of RUBARS – A Rule-Based Recommender System for Learner-Centered Education” and focuses on the evaluation of a rule-based learning task recommender system called RUBARS. This paper has been submitted for publication in the IEEE Transactions on Learning Technologies and is currently under review. As the primary author of this work, I performed the data collection, conducted the data analysis, developed the conclusions and wrote the manuscript under the guidance of my thesis supervisor, Dr. Graf. I wrote the first draft of the manuscript and revised the manuscript multiple times with input from Dr. Graf. Also, Dr. Imran who is the initial designer of RUBARS collaborated with me and prepared the third section of the manuscript which describes the architecture of RUBARS. She also contributed to this manuscript by designing the questionnaire that was used in the evaluation. Writing this manuscript helped me gain knowledge about rule-based recommender systems, data collection methods, participants selection and the evaluation methods that are used to evaluate recommender systems.

The second manuscript is titled “Evaluation of PLORS – a Personalized Learning Object Recommender System for Learning Management Systems” and describes the results of the evaluation of a personalized learning object recommender system for learning management systems called PLORS. This paper has been submitted for publication in ACM Transactions on

Web journal and is currently under review. As the primary author of this work, I performed the data collection, conducted the data analysis, developed the conclusions and wrote the manuscript under the guidance of my thesis supervisor, Dr. Graf. I wrote the first draft of the manuscript and revised the manuscript multiple times with input from Dr. Graf. Dr. Imran who is the initial designer of PLORS collaborated with me and prepared the third section of the manuscript which describes the architecture of PLORS. She also contributed to this manuscript by designing the questionnaire that was used in the evaluation. Writing this manuscript helped me to expand my knowledge on the field of recommender systems for learning and assess the research and evaluation design, participant selection, and data collection methods further. Also, the feedbacks collected from these two evaluations (i.e., PLORS and RUBARS) were used to make my proposed system (WEBLORS) more robust and user-friendly.

The third manuscript is titled “WEBLORS – a Personalized Web-Based Recommender System” and focuses on introducing my proposed system, WEBLORS, and initial results of the evaluation regarding recommender system acceptance. This paper has been published in the Proceeding of the International Conference on Web-based Learning (ICWL 2019) (Belghis-Zadeh, Imran, Chang, & Graf, 2019). As the primary author of this work, I designed and developed the system, designed the evaluation, performed the data collection, conducted the data analysis, developed the conclusions and wrote the manuscript under the guidance of my thesis supervisor, Dr. Graf. Like the other two manuscripts, I wrote the first draft of the manuscript and revised the manuscript multiple times with input from Dr. Graf. Also, Dr. Imran and Dr. Chang contributed by reviewing the initial design of the system and providing me with constructive feedback and helped me to improve the system. Dr. Imran also contributed to this manuscript by designing the questionnaire that was used in the evaluation.

The fourth manuscript is titled "Recommendations of Personalized Learning Objects from the Web Based on Users' Profiles Utilizing Google Custom Search Engine and RAKE Text Mining Algorithm" and discusses WEBLORS' architecture in more detail and presents the evaluation of the system in terms of (1) keyword accuracy, (2) recommendation accuracy, (3) recommender system acceptance, (4) ease of use and user friendliness, and (5) user satisfaction. This paper has been submitted for publication in User Modeling and User-Adapted Interaction Journal and is currently under review. As the primary author of this work, I designed and developed the system, designed the evaluation, created the pre and post tests, performed the data collection, conducted the data analysis, developed the conclusions and wrote the manuscript under the guidance of my thesis supervisor, Dr. Graf. I wrote the first draft of the manuscript and revised the manuscript multiple times with input from Dr. Graf. Dr. Imran and Dr. Chang contributed by reviewing the initial design of the evaluation and providing me with constructive feedback and helped me to improve the evaluation process. Dr. Imran also contributed to this manuscript by designing one of the questionnaires that was used to capture the participants' feedback regarding WEBLORS.

Chapter 3. Manuscript 1. Evaluation of RUBARS – A Rule-Based Recommender System for Learner-Centered Education

Abstract

As learner-centered learning has become more common, supporting learners to successfully set their own learning goals became a main research issue. One way that learner-centered education can be offered is by allowing learners to select tasks for assignments from a pool of learning tasks with different difficulty levels instead of forcing learners to do a set of pre-defined tasks. The challenge that learners can face in this approach is to choose the tasks that are most appropriate for them and can help them in their learning process. RUBARS is a rule-based recommender system that recommends the best possible learning tasks among available tasks for an assignment in a course to learners, considering a learner's learning style, prior knowledge, expertise level, performance of the learner on previously completed tasks, performance of similar learners, the tasks that the learner has initially selected, and the difficulty level of the available tasks. The system aims at selecting tasks that are most appropriate for learners in order to increase their learning and help them achieve good grades. This paper presents the evaluation of RUBARS using quantitative and qualitative methods and data from 51 participants. In this research, RUBARS was evaluated based on its (1) recommender system acceptance, (2) ease of use and user friendliness and (3) user satisfaction, and the evaluation showed very promising results. As a result, we found that users had a positive experience interacting with RUBARS and high scores were given to all three categories on which the system was evaluated.

3.1 Introduction

Enhancing and growing of the internet in recent years has changed the relationship between service providers and consumers drastically in every industry and the educational domain is not an exception. Nowadays, online courses have become more and more popular, where learners can learn new materials and skills in their own place and at their own pace. As a result of this change, learning management systems (LMS) are commonly used as content delivery tools and repositories of educational information materials and resources (Holmes & Prieto-Rodriguez, 2018; Rhode, Richter, Gowen, Miller, & Wills, 2017). Like any other online system, delivering personalized learning materials can enhance the functionality of LMSs and help the learners and teachers in their learning and teaching processes (Luna-Urquizo, 2019; Perišić, Milovanović, & Kazi, 2018).

According to Khribi and colleagues (2009), personalization in LMSs is defined as techniques that are used to choose the materials and content that best matches the learners' profile (e.g., demographic, geographic, learners' skills, knowledge, background, personality and interests). Such techniques consider students' profiles to decide which content should be presented to each learner. Recommender systems are one of the tools that can be used to deliver personalized content to users.

In the context of learning in particular, recommender systems are defined as techniques that help students to reach their learning goals faster by finding and providing learning materials and resources that satisfy their needs (Fraihat & Shambour, 2015; Nafea, Siewe, & He, 2019). In the past thirty years, different recommender systems have been designed and developed using different approaches that recommend different types of learning materials such as courses, learning objects, assignments, mentors, funding and scholarship opportunities, research and academic papers, etc. Recommender systems in the learning domain are used for many different purposes and help learners in different ways. In this paper, we focus on recommender systems that are supporting learner-centered learning. In learner-centered learning, learners are responsible for setting their own learning goals (Akdemir & Özçelik, 2019; Jonassen, 2000). Learners are also in charge of choosing the learning materials and activities that are beneficial to them and help them achieve their learning goals (Pedersen & Liu, 2003). One of the challenges that learners can face in learner-centered education is that some learners do not have the required knowledge to make proper choices and choose the activities that fit best for them. In this case, recommender systems can help to address this issue. For example, in learner-centered education, there could be assignments that contain many learning tasks with different difficulty levels, and students may choose which tasks they prefer to complete in order to fulfill the requirements of the assignment (e.g., many easy tasks, a few difficult tasks, etc.). However, in such a situation, some students might not have enough knowledge to select the tasks that are most beneficial for them. In order to support students in such situation and help them to make the best selections, a rule-based recommender system called RUBARS has been built by Imran and colleagues that provides recommendations of learning tasks to learners and helps them to choose the learning tasks that are most beneficial to them (Imran et al., 2014). RUBARS recommends a set of learning tasks to learners in a course based on the learner's characteristics, his/her performance in the previous tasks, the performance of similar learners, the tasks that the learner has initially selected and the difficulty level of the available tasks (Imran et al., 2014). In this paper, an evaluation of RUBARS is introduced that was performed to investigate whether the system is beneficial to learners. More concretely, RUBARS was evaluated with 51 participants based on its (1) recommender system acceptance, (2) ease of use and user friendliness and (3) user satisfaction.

The remainder of this paper is structured as follows: Section 3.2 presents related work. Section 3.3 discusses the background and describes RUBARS' architecture and approach. Section 3.4 explains the evaluation methodology and approach that was taken to evaluate the system. Section 3.5 presents the result of the evaluation. Section 3.6 discusses the result in more detail and finally, section 3.7 concludes the paper by summarizing the main contributions of the paper and discusses future research directions.

3.2 Related Work

The idea of recommender systems in the educational domain has been around for decades, however, the definition and implementation techniques for these systems evolved over time (Manouselis, 2013). By reviewing the previous literature, it was observed that most researchers categorized recommender systems based on their underlying techniques into three main categories: (1) collaborative filtering, (2) content-based filtering, and (3) hybrid filtering (Bobadilla, Ortega, Hernando, & Gutiérrez, 2013; Nilashi, Ibrahim, & Bagherifard, 2018; Sarwar, Karypis, Konstan, & Riedl, 2001).

- Collaborative filtering (CF): These recommender systems provide recommendations based on opinions or ratings that were provided by other similar users (Sarwar et al., 2001). These systems work the same way that humans make decisions in real life which is often based on help from other people's opinion (Bobadilla et al., 2013).
- Content-based filtering (CBF): These systems generate recommendations based on users' preferences in the past (or present). In other words, these systems select and recommend the items by comparing the user profiles with the available recommendable items and recommend items that are similar to items the user liked in the past (Van Meteren & Van Someren, 2000). For example, if a learner looked at computer science articles in the past, the system recommends more new papers and articles in computer science to that user (Fraihat & Shambour, 2015).
- Hybrid filtering (HF): These systems combine and utilize the strength of multiple recommendation techniques to suggest items or actions to the user (Burke, 2007; Tarus, Niu, & Kalui, 2018). These systems work based on combining user modeling and item

classification techniques to improve the accuracy of recommendations (Wen, Fang, & Guan, 2012).

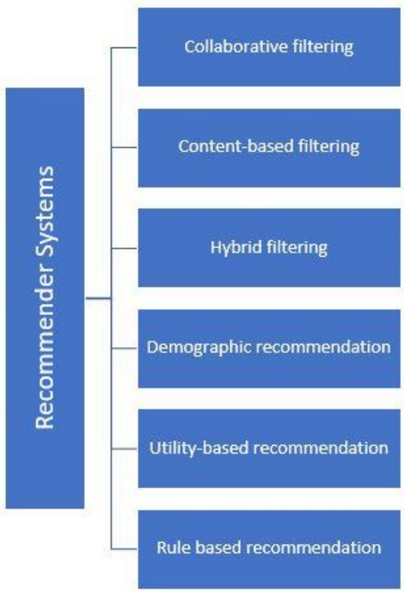
In addition to the above main categories, there are some other categories that have been proposed in the literature (Burke, 2002, 2007; Martinez, Barranco, Pérez, & Espinilla, 2008; Zapata, Menéndez, Prieto, & Romero, 2013):

- Demographic recommender systems (DE): These systems generate recommendations based on users' demographic attributes. These systems create a demographic profile for each user and provide recommendations based on ratings of other users with a similar demographic profile.
- Utility-based recommender systems (UB): These systems generate recommendations by creating a utility function for each recommendable item to determine which items are beneficial and interesting to a given user.
- Knowledge-based or rule-based recommender systems (RB): These systems recommend items to users by considering users' profiles and applying some pre-defined rules to decide which items should be recommended to each user.

Figure 3-1 shows those different types of recommender systems.

Figure 3-1

Different Types of Recommender Systems



As mentioned above, the idea of recommender systems in the educational domain has been around for many years and different systems using different techniques have been developed. Next, some of these recommender systems are described.

Rodríguez and colleagues (2013) created a hybrid recommender system that uses students' learning styles, educational level, language preference and users' opinions to search and discover relevant learning objects from repositories. This system uses a two-stage process in order to generate the recommendations. It first clusters the users based on learning styles, educational level, and language preference. Next, it finds the learning objects from repositories that match the user's query and were also found interesting by similar users. Another similar hybrid recommender system was built by Dwivedi and Bharadwaj (2015). This system categorizes users based on their similarity and recommends learning resources for groups of users instead of individuals. This system uses learners' past activities, learners' learning styles and their knowledge level to create students' profiles. Next, it clusters the users using the nearest neighbor algorithm, merges the profiles of the users within each group and provides recommendations based on the group's profile to students belonging to the respective group.

Another group of recommender systems generates recommendations based on similarities between user and items rather than similarities between users. For example, Salehi and colleagues (2013) built a content-based recommender system that considers the users' past activities extracted from server logs as well as different attributes of the items and users to build a prediction model and predicts the users' interests to unseen learning objects. This system uses the historical rating data to determine what attributes of the learning objects are more attractive to the user. Next, it uses a genetic algorithm and nearest neighborhood algorithm to find the relationship between users' preferences and the available items and decides which items should be recommended to the user.

Some other systems use different rules to match and recommend the best possible available recommendable items to the users while generating recommendations. For instance, Chen, Lee and Chen (2005) built a rule-based recommender system that uses item response theory and applies some pre-defined rules to recommend course materials to learners based on learners' learning abilities. This system determines the learners' abilities by asking users to complete a questionnaire. Also, this system categorizes the course materials based on their difficulty level. The difficulty level of the materials is determined based on the feedback that is provided by users. In this system, all materials are marked as moderate difficulty level at the beginning by default. Also, for new users, the system recommends moderate course materials. Every time that the system presents a recommendation to a learner, that learner is asked to submit their feedback regarding the difficulty level of the recommended material and based on the feedbacks, the difficulty level of the material gets updated. Another rule-based system was proposed by DorÇA and colleagues (2016) which defines a set of rules that are used to categorize the learning objects based on their teaching style. Next, it identifies the students' learning style and recommends learning objects with a teaching style that support a students' learning style.

Table 3-1 provides an overview and comparison of the main characteristics of the before-mentioned research works versus the RUBARS system proposed in this paper which will be explained in more detail in the following section. In this table, these systems are compared with respect to object of recommendation, attributes and adaptivity, models and algorithms, and recommendation techniques that are used in each of these systems.

Table 3-1

Comparison of related educational recommender systems

		(Salehi, Pourzaferani,	eL_GRS (Dwivedi & Bharadwaj,	(DorÇA, AraÚJo, de Carvalho,	BROA (Rodríguez, Tabares, Mendez, Carranza, & Vicari. 2013)	PEL-IRT (Chen, Lee, & Chen, 2005)	RUBARS (Imran, Belghis-Zadeh,
Type of RS		CBF	CF	RB	HF	RB	RB
Object of Recommendation	Learning Object	Y	Y	Y	Y	Y	N
	Learning Tasks in Assignments	N	N	N	N	N	Y
Attributes and Adaptivity	Learning Style	N	Y	Y	Y	N	Y
	Expertise Level	N	Y	N	Y	Y	Y
	Prior Knowledge	N	Y	N	N	Y	Y
	Performance	N	N	N	N	Y	Y
	Past Activities	Y	Y	N	N	Y	Y
	Learning Ability	N	N	N	N	Y	N
	Language preference	N	N	N	Y	N	N
	Teaching Style	N	N	Y	N	N	N
Models and Algorithms	Neighborhood	Y	Y	N	Y	N	Y
	Real-time Neighborhood	N	N	N	Y	N	Y
	User Ratings	Y	Y	N	Y	Y	N
Recommendation Techniques	Object Similarity	N	N	N	N	Y	N
	User Similarity	N	Y	N	Y	N	Y
	Similarities between User and Object	Y	N	Y	N	N	N

As shown in Table 3-1, RUBARS is different compared to other recommender systems in terms of the type of objects that are recommended to learners. Although the presented systems are closely related to RUBARS, to the best of our knowledge, there is no other recommender system that recommends learning tasks within assignments. In addition, RUBARS focuses on providing highly personalized recommendations and therefore uses four different user attributes (i.e., learners' learning styles, prior knowledge, expertise level, and performance on previously completed tasks) to build each user's neighborhoods, which is more than most other systems. Another important feature of RUBARS is that it generates the neighborhood in real-time, again allowing for more precise and up-to-date recommendations. Only a few recommender systems (e.g., Rodríguez and colleagues (2013)) in the educational domain are using this technique. In the next section, an overview of RUBARS is provided.

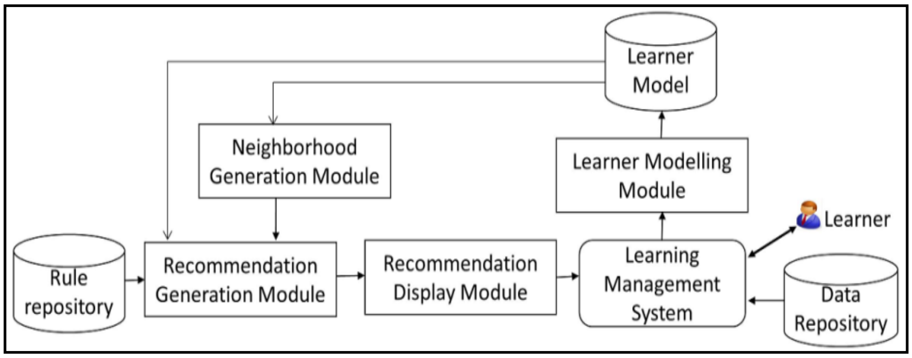
3.3 RUBARS

In this section, a description of RUBARS' architecture as well as how each module contributes to providing recommendations to learners is provided. While this section only provides an overview of RUBARS, a detailed explanation of the system and its components can be found in Imran et al.'s article (2014).

RUBARS is comprised of four modules, and each module is explained briefly in the following paragraphs. Figure 3-2 illustrates the architectural diagram of RUBARS.

Figure 3-3

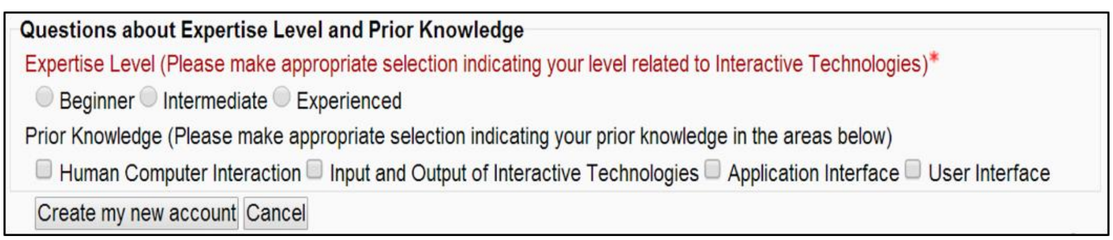
Architecture of RUBARS



The goal of the **Learner Modeling Module** is to collect information about the learners and storing this information in the Learner Model. The system considers four types of information, namely, learning styles, prior knowledge, expertise level, and performance. Figure 3-3 shows an interface that allows the user to provide information about their expertise level and prior knowledge.

Figure 3-2

Interface for gathering information about a learner



Along with the above information, the user is also asked to fill a learning style questionnaire (Felder & Soloman, 1997) which was developed by Felder and Soloman and was found to be valid, reliable and suitable for identifying learning styles (Felder & Spurlin, 2005). The performance of learners is gathered automatically through the marks that learners receive on their assignments and in particular on individual tasks. In addition, each learner is requested to submit an initial plan, outlining the tasks he/she is planning to do in each assignment.

The aim of the **Neighborhood Generation Module** is to find the neighbors of a target learner based on the information from the Learner Model. In order to do so, the Euclidean distance (see formula 1) is used to calculate how similar a learner *j* is to the target learner *i* based on their characteristics (i.e., learning styles, expertise level, prior knowledge, and performance).

$$Euclidean\_distance\ (L_{i}, L_{j}) = \sqrt{\sum_{k=1}^{n}(L_{i_k} - L_{jk})^2}\ , \qquad (1)$$

where L_{ik} signifies the characteristic k of learner i and n represent the number of characteristics measured. A threshold t=0.66 has been set through experimentation, determining whether the similarity of characteristics of a learner is close enough for the learner to be considered to be a neighbor of the target learner. Threshold t was set to 0.66 based on the assumption that if the difference between each attribute of two learners is on average less than or equals to 0.25 (on a scale of 0 to 1), then two learners can be considered similar. In such situation, the Euclidean distance between two learners would be less than or equals to 0.66.

The **Recommendation Generation Module's** goal is to make suitable recommendations to the target learner. Each learning task is categorized as one of the following: Easy (E), Moderate (M) and Challenging (C) by instructors. The rule generation process has three inputs: *the previous performance of the target learner on each difficulty level, neighbor's average performance and the target learner's initial plan.* Recommendation generation has two steps: (1) ranking of difficulty levels and (2) selection of learning tasks based on the ranking of difficulty levels.

In the first step, the system identifies which difficulty levels are best suited for the target learner and created respective ranks for the three difficulty levels (i.e., easy, moderate and challenging). To do so, each difficulty level is mapped to one of the three *levels of priority*, representing the ranks: highest priority level (HPL), medium priority level (MPL) and low priority level (LPL). To identify HPL, the system uses nine different rules as shown in Table 3-2, considering the performance of the target learner on easy, moderate and challenging tasks on previous assignments as well as the performance of neighbor learners on easy, moderate and challenging tasks of the current assignment. After identifying HPL, the system determines which difficulty levels should be assigned to MPL and LPL. In order to do so, the system takes into account the average performance of the target learner on previously conducted learning tasks of the two remaining difficulty levels (e.g., average performance on challenging tasks) and the average performance of neighbor learners on tasks of the current assignment of the two remaining difficulty levels. Based on those two average values, a combined average is built, and the difficulty level with the higher result is assigned to MPL and the other one is assigned to LPL.

Table 3-2

Rules to identify HPL

	Target Learner' Performance (previous assignments)		Neighbors' Performance (current assignment)		HPL
If	E > M & C	AND	E > M & C	Then	Easy
If	E > M & C	AND	M > E & C	Then	Moderate
If	E > M & C	AND	C > E & M	Then	Easy
If	M > E & C	AND	E > M & C	Then	Easy
If	M > E & C	AND	M > E & C	Then	Moderate
If	M > E & C	AND	C > E & M	Then	Challenging
If	C > E & M	AND	E > M & C	Then	Challenging
If	C > E & M	AND	M > E & C	Then	Moderate
If	C > E & M	AND	C > E & M	Then	Challenging

In the second step, the most suitable learning tasks for the target learner are chosen based on three parameters: (1) the priority levels for the target learner as calculated in step 1, (2) the average performance of the neighbor learners on the respective tasks (Avg_N_Perform) and whether that performance is above a threshold T (which is set to 60% but can be adjusted by instructors), and (3) whether tasks were chosen by the target learner in the initial plan (SELECTED or NOT_SELECTED). The selection of learning tasks is based on twelve rules (shown in Table 3-3). These rules are applied in the same sequence until enough tasks are selected for the target learner to fulfill the requirement of the assignment.

Table 3-3

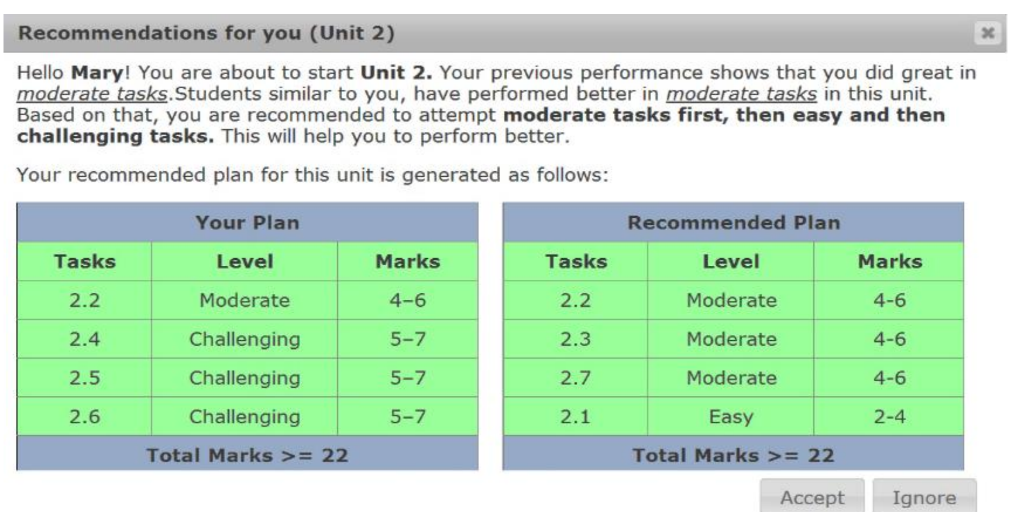
Rules For The Selection Of Learning Tasks

Rule #	Rule
1	Select HPL tasks where *Avg_N_Perform* > *T* and is SELECTED by target learner
2	Select HPL tasks where *Avg_N_Perform* > *T* and is NOT_SELECTED by target learner
3	Select HPL tasks where *Avg_N_Perform* is UNKNOWN and is SELECTED by target learner
4	Select HPL tasks where *Avg_N_Perform* is UNKNOWN and is NOT_SELECTED by target learner
5	Select MPL tasks where *Avg_N_Perform* > *T* and is SELECTED by target learner
6	Select MPL tasks where *Avg_N_Perform* > *T* and is NOT_SELECTED by target learner
7	Select MPL tasks where *Avg_N_Perform* is UNKNOWN and is SELECTED by target learner
8	Select MPL tasks where *Avg_N_Perform* is UNKNOWN and is NOT_SELECTED by target learner
9	Select LPL tasks where *Avg_N_Perform* > *T* and is SELECTED by target learner
10	Select LPL tasks where *Avg_N_Perform* > *T* and is NOT_SELECTED by target learner
11	Select LPL tasks where *Avg_N_Perform* is UNKNOWN and is SELECTED by target learner
12	Select LPL tasks where *Avg_N_Perform* is UNKNOWN and is NOT_SELECTED by target learner

The goal of the **Recommendation Display Module** is to display the recommendations to the target learner. The target learner can accept or ignore the recommendations. The system also stores the recommendations for future access. An example of such a recommendation is shown in Figure 3-4.

Figure 3-4

An example of a recommendation



3.4 Methodology

In this section, we introduce the methodology used to evaluate RUBARS in order to confirm that the approach and generated recommendations are truly practical and effective through qualitative and quantitative evaluation. The main metrics that were considered during the evaluation are (1) recommender system acceptance, (2) ease of use and user friendliness and (3) user satisfaction. The research design and participant selection are explained in the next two subsections.

3.4.1 Research Design

As mentioned in section 3.2, to the best of our knowledge, there is no other recommender system that recommends learning tasks within assignments. Therefore, an A/B test could not be performed to compare RUBARS with other systems to prove whether RUBARS outperforms the other systems or not. Another common way to prove the usefulness and effectiveness of recommender systems is to conduct a user study, which is a scientific method that is commonly used to evaluate recommender systems (Erdt, Fernández, & Rensing, 2015; Knijnenburg, 2012; Rahman & Abdullah, 2018; Shani & Gunawardana, 2011). To evaluate RUBARS through a user study, a group of 51 users were asked to use the system and then provide us with their feedback regarding their experience while interacting with the system. In order to perform the user study, a three-step process was designed and published on the web. Figure 3-5 shows the website that was used for this evaluation to guide participants through the evaluation process.

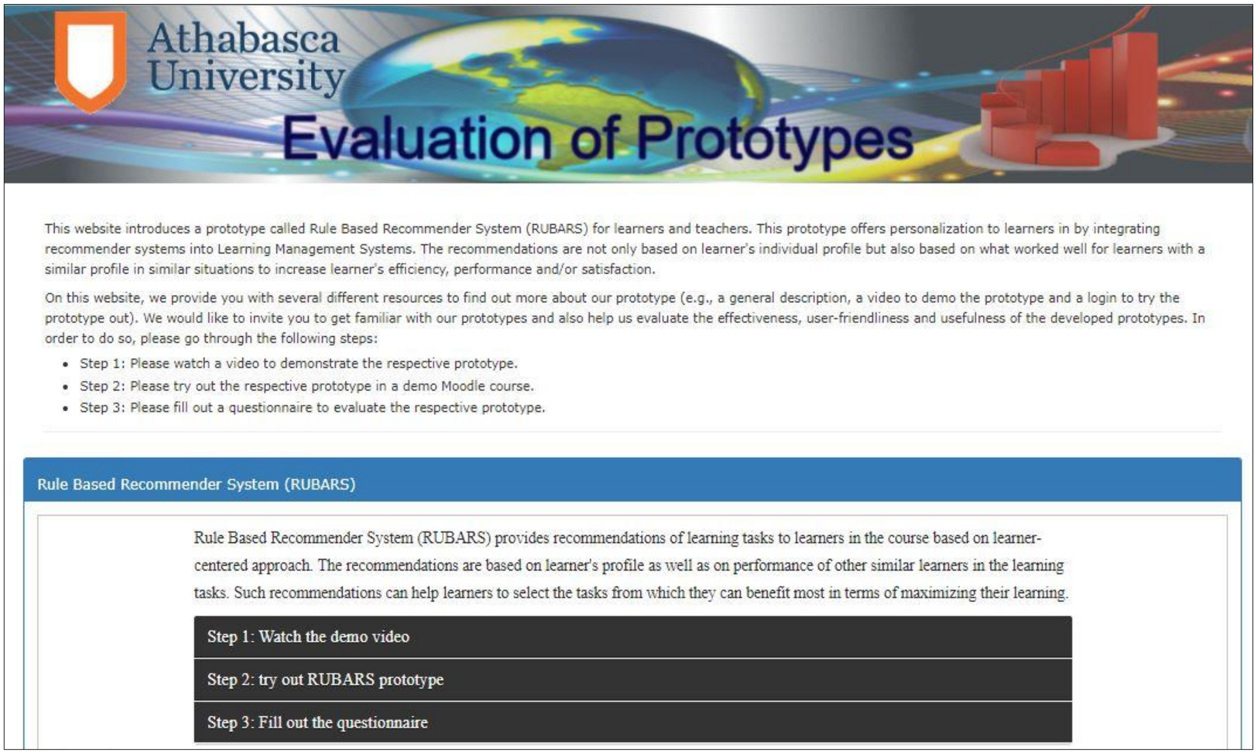
Those three steps of the evaluation are as follows:

1. Participants were asked to watch a video that contains a full demo of RUBARS and helps participants to get familiar with the system and evaluation process. In this video, the purpose and the functionalities of the system are explained, and users are walked

through how the system can be used.

Figure 3-5

RUBARS Evaluation



2. In the second step, users were asked to use the system. For this step, RUBARS was integrated into an instance of the LMS Moodle (Dougiamas, 2019) and a sample introductory course on the topic of Interactive Technologies containing 4 units was used. Each unit in the course contains some learning objects. In addition, each unit has an assignment and each assignment contains different tasks with different difficulty levels. Data from 10 manually simulated users was used as the base data.

    For this evaluation, each participant was provided with a login credential. Users were asked to log in to the system. Once logged in to the system, users were presented with the Index of Learning Styles questionnaire that they needed to fill out. Next, users were asked to answer the expertise level and background knowledge questions. After that, users were asked to navigate to the sample course and submit an initial plan that contained the list of tasks that they wanted to do for each assignment. After that, users were instructed to navigate to each unit of the course and visit different learning objects in each unit. Once users visited the first learning object of a unit, RUBARS provided them with a recommendation containing (1) the list of tasks that they selected in their initial plan and (2) the recommended tasks that would be best for them to complete. Each time such a recommended plan is presented to the users, they had the option to accept or ignore the recommended plan. Whether users accept or ignore the recommendation, the recommended plans are stored in the database and users are presented with an option in each unit to see their recommended plan for that unit at any time.

3. In the third step, users were provided with a questionnaire that contained 16 questions to allow participants to provide feedback regarding their experience while using RUBARS. This questionnaire contained 12 multiple-choice questions (Q1 to Q12) and four open-ended questions. For the multiple-choice questions, users could rate their experience on a scale of 1 to 5, where 1 indicated strong disagreement with the respective statement and 5 indicated strong agreement. These multiple-choice questions were created based on the SUS questionnaire (Brooke, 1996; Lewis & Sauro, 2009). According to the focus of the questions, they were categorized into three categories with particular emphasis on recommender systems. These three categories are (1) Recommender System Acceptance, (2) Ease of Use and User Friendliness, and (3) User Satisfaction. In order to prove the reliability of the questions in each category, Cronbach alpha reliability coefficient was calculated for each category (Cronbach &

Meehl, 1955) using the data from the 51 participants in the study. The alpha for each category is reported below:

- Recommender System Acceptance: 0.81
- Ease of Use and User Friendliness: 0.86
- User Friendliness: 0.78

The calculated alpha values above are greater than 0.7 indicating that the questions in each category are reliable (Nunnally, 1994; Santos, 1999; Tavakol & Dennick, 2011).

In addition to the 12 multiple-choice questions, four open-ended questions were designed where participants could write their responses and comments (Q13 to Q16). Moreover, participants had the option to provide us with their name, email address and occupation optionally. Questions 1 to 16 are listed below:

Recommender System Acceptance:

Q1: I would like to use RUBARS frequently

Q2: I would like to see such recommendations in other courses as well

Q3: I trusted the recommendations provided by RUBARS

Q4: I think recommendations provided by RUBARS will be helpful in increasing students' learning

Q5: The recommendations will help students in selecting the appropriate tasks in the course

Ease of Use / User Friendliness:

Q6: I liked the interface of RUBARS

Q7: I found RUBARS unnecessarily complex

Q8: It took me a long time before I could understand the recommendations

Q9: The information provided in the recommendation was clear and easy to understand

User Satisfaction:

Q10: I felt very confident in using RUBARS

Q11: I think such recommendations would be helpful for students in order to perform better in a course

Q12: As a whole, I am satisfied with the learning tasks recommended by RUBARS

Open-ended Questions:

Q13: List aspects you appreciate most in RUBARS

Q14: If you could change one thing in RUBARS, whether it is major or minor, what would be at the top of the list?

Q15: Was there something missing you were expecting to see in the recommendations provided by RUBARS?

Q16: Is there any other comment that you want to give for RUBARS?

3.4.2 Participants Selection

For this evaluation, a new task was created and posted on Amazon Mechanical Turk and 51 participants accepted this task. There was no qualification defined for the participants as RUBARS was designed to be used for any online course and by anyone who is attempting to learn a concept online. Also, prior experience using LMSs was not required for the participants, however, nowadays most people have used LMSs of some sort in the past as these systems are widely used for online and blended education by schools as well as for skill-based training in workplaces (Sabharwal, Hossain, Chugh, & Wells, 2018). Thus, different people with different occupations participated in this study. Among the 51 participants, there were graduate and undergraduate students, teachers and professors, software developers, doctors, retired employees, business owners and construction workers. The fact that different people with different levels of education were included in the sample group makes us believe that a broad target audience was covered in this evaluation and the result represents the opinions of many of the potential users of RUBARS. All participants went through the three-steps process described in Section 3.4.1 and completed the feedback questionnaire. Section 3.5 explains the result of the evaluation.

3.5 Results

In order to analyze the result and transform the users' answers to meaningful information, results were divided into quantitative and qualitative categories. In the following two subsections, quantitative and qualitative results are presented.

3.5.1 Quantitative Results

As the first analysis, the answers that were given to the twelve multiple-choice questions that were collected from 51 participants were aggregated. Each question has five possible answers with respective scores provided in brackets: Strongly Agree (5), Agree (4), Neither Agree nor Disagree (3), Disagree (2) and Strongly Disagree (1). Next, the weighted average score WAvg was calculated for each question *q* using formula 2.

$$\mathrm{WAvg}(q) = \frac{\sum_{w=1}^{5} w.a_w}{\sum_{w=1}^{5} a_w} \qquad (2)$$

In formula 2, *q* is the given question, WAvg*(q)* is the weighted average score for the given question, *w* is the score/weight of the possible answer and a_w(w=1..5) is the number of participants who selected the answer with the score/weight of *w* for the given question. For example, 51 participants answered Q1. Out of these 51 answers, 3 participants selected “Strongly Agree”, 35 participants selected “Agree”, 6 participants selected “Neither Agree or Disagree”, 6 participants selected “Disagree” and one participant selected “Strongly Disagree” as their answers. Based on these answers, the weighted average score for Q1 was calculated as follows:

$$\mathrm{WAvg(Q1)} = \frac{3\text{x}5+35\text{x}4+6\text{x}3+6\text{x}2+1\text{x}1}{3+35+6+6+1} = 3.65$$

Accordingly, the weighted average score for Q1 (WAvg*(Q1)*) is 3.65 out of 5.

Table 3-4 contains the responses that were received from participants and the calculated weighted average score (*Wavg*) for each question.

Table 3-4

Results of quantitative analysis

	Question		**Total**	**Strongly Agree (5)**	**Agree (4)**	**Neither Agree nor Disagree (3)**	**Disagree (2)**	**Strongly Disagree (1)**	**WAvg(q)**
Recommender System Acceptance	**Q1**	I would like to use RUBARS frequently	51	3	35	6	6	1	3.65
	Q2	I would like to see such recommendations in other courses as well	51	15	31	1	3	1	4.10
	Q3	I trusted the recommendations provided by RUBARS	51	5	35	8	3	0	3.82
	Q4	I think recommendations provided by RUBARS will be helpful in increasing students’ learning	51	11	30	7	2	1	3.94
	Q5	The recommendations will help students in selecting the appropriate tasks in the course	51	10	36	3	2	0	4.06
Ease of Use / User Friendliness	**Q6**	I liked the interface of RUBARS	51	10	20	7	11	3	3.45
	Q7	I found RUBARS unnecessarily complex	51	3	13	6	25	4	2.73
	Q8	It took me a long time before I could understand the recommendations	51	0	14	8	22	7	2.57
	Q9	The information provided in the recommendation was clear and easy to understand	51	6	25	13	5	2	3.55
Satisfaction	**Q10**	I felt very confident in using RUBARS	51	6	24	12	6	3	3.47
	Q11	I think such recommendations would be helpful for students in order to perform better in a course	51	11	31	6	2	1	3.96
	Q12	As a whole, I am satisfied with the learning tasks recommended by RUBARS	51	4	37	6	3	1	3.78

As shown in Table 3-4, the recommender system acceptance category contains five questions (Q1 to Q5). The weighted average score for Q1 is 3.65 which is between a neutral opinion and agreeing with a slight tendency towards agreeing that participants would like to use RUBARS frequently. Regarding the question whether participants would like to see such recommendations in other courses as well (Q2), the weighted average score is 4.10 which is between agreeing and strongly agreeing with a strong tendency towards agreeing. The weighted average score for Q3 is 3.82 which is between a neutral opinion and agreeing with a tendency towards agreeing that participants trusted the recommendations provided by RUBARS. In Q4, users are asked whether they think recommendations provided by RUBARS will be helpful in increasing students' learning. The weighted average score for this question is 3.94 which is between a neutral opinion and agreeing with a very strong tendency towards agreeing. The weighted average score for Q5 is 4.06 which is between agreeing and strongly agreeing with a very strong tendency towards agreeing that provided recommendations will help students in selecting the appropriate tasks in the course.

The ease of use and user friendliness category contains two sets of questions. Q6 and Q9 are positive questions. Regarding Q6 which asks users whether they like the interface of RUBARS, the weighted average score is 3.45 which is between a neutral opinion and agreeing. For Q9, the weighted average score is 3.55 which is between a neutral opinion and agreeing that information provided in the recommendations were clear and easy to understand. The other two questions in the ease of use and user friendliness category (Q7 and Q8) are negative questions. Q7 asks users whether they found RUBARS unnecessarily complex. The weighted average score for this question is 2.73 which is between a neutral opinion and disagreeing with a tendency towards neutral opinion. In Q8, participants are asked if it took them a long time before they could understand the recommendations. The weighted average score for this question is 2.57 which is between a neutral opinion and disagreeing.

Q10 to Q12 address user satisfaction with RUBARS. The weighted average score for Q10 is 3.47 which is between a neutral opinion and agreeing that users felt confident in using RUBARS. In Q11, users are asked if they think such recommendations would be helpful for students in order to perform better in a course. The weighted average score for this question is 3.96 which is between a neutral opinion and agreeing with a very strong tendency towards agreeing. In Q12, users are asked if they were satisfied with the learning tasks recommended by RUBARS. The weighted average score for this question is 3.78 which is between a neutral opinion and agreeing with a tendency towards agreeing.

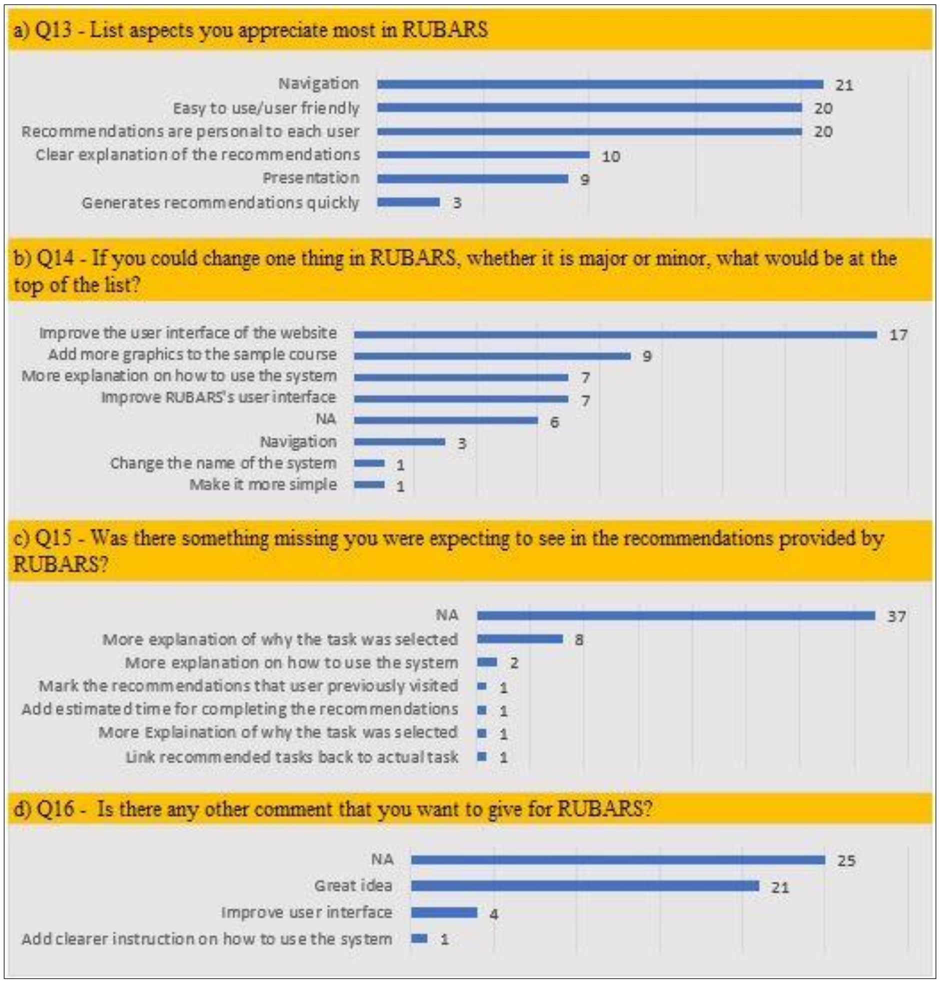
3.5.2. Qualitative Results

As mentioned in Section 3.4, four open-ended questions were included in the questionnaire (Q13-Q16), so users could explain their answers and provide additional feedback regarding their experience with RUBARS. As these questions did not have pre-defined answers, each response was classified into topics. In order to classify the responses, each response was manually reviewed. Next, responses were grouped together based on the similarity of the topics which were discussed in the answers. An answer could include more than one topic. For example, a response such as "I appreciate that recommendations are personal to learners. I also appreciate the simple and easy to use user interface of RUBARS." to Q13 was classified under the topics of "Recommendations are personal to each user" and "Easy to use / User Friendly". Next, the number of times that each topic occurred in the answers was counted. Figure 3-6. shows the result of this qualitative analysis.

In Q13, users were asked to list the aspects that they appreciate most in RUBARS. As shown in the chart (a) of Figure 3-6, easy navigation, ease of use and user friendliness, personalization of recommendations, clear explanation of recommended tasks, and presentation of recommendations are the top five topics that were appreciated by the users of RUBARS.

Figure 3-6

Results of qualitative analysis



In Q14, participants were asked to specify one thing that they would like to see changed in the system. Chart (b) of Figure 3-6 shows that 17 users mentioned that the user interface of the website (Moodle) needs to be improved and 9 participants stated that the sample course that was used for the evaluation seemed to be too plain and they like to see more graphics to be added to the course. In addition, 7 users mentioned that more instruction should be added to the system and

7 users stated that they like to see improvements in the user interface of the RUBARS. Also, 6 users specified that nothing needs to be changed in RUBARS. Moreover, a few users believed the navigation within the course needs to be improved. Also, one user mentioned that he/she would change the name of the system and another user believed that the system needs to be simplified.

In Q15, participants were requested to identify the gaps and list what they believe is missing in the recommendations provided by the system. As shown in the chart (c) of Figure 3-6, 37 users think that nothing was missing. Also, 8 participants believed that more explanation should be added to the recommendations to help users understand why a certain task was selected and recommended to the learners. In addition, 2 users mentioned that more explanation needs to be added to the system and one user stated that it would be helpful to add the estimated time of completion to each recommended task. Moreover, one user suggested that a checkmark to be added to the recommendations that users have visited previously. Also, one user suggested that the recommended tasks should be linked to the actual tasks in the course, so users can click on the recommendation and navigate to the recommended tasks directly.

In Q16, users were asked to add any additional comments that they might have regarding their experience while using RUBARS. As shown in the chart (d) of Figure 3-6, most participants either did not have any additional comments or stated that RUBARS is a great idea. A few users mentioned that they would like to see some improvements on the user interface of the website (Moodle) and one user asked for clearer instructions on how to use the system to be added to the evaluation instruction and the demo video.

3.6 Discussion

The results presented in section 3.5 show that most of the participants of this evaluation study believe that recommendations provided by RUBARS potentially solve an issue in learner-centered education by helping learners to choose the tasks that are most appropriate for them (per answers given to Q5). Based on feedback from most participants, such recommendations were found to have the potential to help learners in their learning process (per answers given to Q4).

In terms of recommender system acceptance, the quantitative results show that most participants (41 out of 51) agreed that the provided recommendations by RUBARS can help learners in selecting the appropriate tasks in a course (per answers given to Q5). Similarly, per answers given to Q2, most users (46 out of 51) agreed that they would like to have such recommendations available to them while studying other courses. In addition, most participants (40 out of 51) agreed with the statement in Q3 and trusted the recommendations provided by RUBARS. Moreover, most participants (41 out of 51) agreed that the provided recommendations by RUBARS can be helpful in increasing students’ learning (per answers given to Q4). In addition, most participants (40 out of 51) agreed that they would like to use RUBARS frequently (per answers given to Q1). Qualitative results also reflect the positive feedback from the quantitative data regarding the recommender system acceptance. Regarding the aspects that users appreciated most in RUBARS, as shown in chart (a) of Figure 3-6, 20 (out of 51) users appreciated that the recommendations are personal to each user and 10 participants appreciated the clear explanation of the provided recommendations. In addition, 21 (out of 51) participants provided additional feedback and stressed that they believe RUBARS is a great idea (as shown in chart (d) of Figure 3-6).

In terms of ease of use and user friendliness, the quantitative results show that the majority of users (29 out of 51) did not find RUBARS unnecessarily complex and believed that they could understand the recommendations in a reasonable amount of time (per answers given to Q7 and Q8). In addition, as per answers given to Q9, the majority of participants (31 out of 51) believed that the information provided in the recommendations were clear and easy to understand. Moreover, the majority of users (30 out of 51) agreed with the statement in Q6 and liked the interface of RUBARS. Qualitative results regarding ease of use and user friendliness also show that the “easy navigation”, and “ease of use and user friendliness” are the top two aspects that were appreciated most by the participants. As shown in chart(a) of Figure 3-6, out of 51 participants, 21 users considered easy navigation as an aspect that they appreciate most in RUBARS. 20 users (out of 51) also stated ease of use and user friendliness as an aspect that they most appreciated in RUBARS (as shown in chart (a) of Figure 3-6). In addition, as shown in chart (a) of Figure 3-6, 9 users appreciated the way that recommendations were presented to them in RUBARS, and 10 users appreciated the clear explanation of the recommendations. Although the majority of users liked the interface of RUBARS (as per answers given to Q6), 7 users listed the user interface of RUBARS as something that needs to be improved (as shown in chart (b) of Figure 3-6). Qualitative results regarding ease of use and user friendliness also show that some of the participants raised some concerns with regards to the interface of the website (Moodle) or the sample course that was used for the evaluation (as shown in chart (b) of Figure 3-6). As shown in chart (b) of Figure 3-6, 26 participants mentioned that improving the user interface of the website (Moodle) or adding more graphics to the sample course would be at the top of their list to improve.

In terms of user satisfaction, quantitative results show that most users (42 out of 51) agreed that the recommendations provided by RUBARS would be helpful for students in order to perform better in a course (as per answers given to Q11). In addition, per answers given to Q12, most participants (41 out of 51) agreed that they were satisfied with the learning tasks recommended by RUBARS. Moreover, the majority of users (30 out of 51) agreed with the statement in Q10 and felt confident in using RUBARS. Qualitative results regarding user satisfaction indicates that the majority of users (37 out of 51) stated that they were satisfied with the current state of RUBARS and did not feel like anything was missing in the system (as shown in the chart (c) of Figure 3-6). In addition, as shown in chart (d) of Figure 3-6, 20 users provided additional feedback and emphasis that they believe RUBARS is a great idea. Moreover, as shown in chart(a) of Figure 3-6, personalization of the recommendations is stated by 20 users as an aspect that they appreciated most in RUBARS. In addition, 9 participants appreciated the way that recommendations are presented to them.

3.7 Conclusion

This paper introduces RUBARS which is a rule-based recommender system that supports learner-centered education by helping students to choose the most beneficial learning tasks for them among many available tasks within assignments. RUBARS recommends a set of learning tasks to learners in a course based on the learner’s learning styles, prior knowledge, expertise level, performance on previously completed tasks, performance of similar learners, the tasks that the learner has initially selected, and the difficulty level of the available tasks. Reviewing the existing literature shows that, to the best of our knowledge, there is no other system that recommends learning tasks within assignments. Thus, RUBARS fills a gap in learner-centered education and can enhance the functionality of LMSs. The focus of this paper is on evaluating different aspects of RUBARS as a personal rule-based recommender system using a user study to capture users’ feedback regarding RUBARS. 51 participants participated in this research study where they were asked to use the system and provide feedback regarding their experience while interacting with RUBARS. In this research, RUBARS was evaluated based on its (1) recommender system acceptance, (2) ease of use and user friendliness and (3) user satisfaction. Quantitative and qualitative results show that overall, users provided positive feedback regarding RUBARS.

The results from the quantitative and qualitative data both show that participants provided positive feedback about recommender system acceptance. Most participants trusted the recommendations provided by RUBARS (Q3), would like to see such recommendations available in online courses that they attend (Q2), and believed that the recommendations that were provided by RUBARS will help students in selecting the appropriate tasks in a course (Q5) and will be helpful in increasing students' learning (Q4). In addition, the majority of users would like to use RUBARS frequently (Q1). Several users also pointed out that they appreciated the clear explanation of the recommendations and that the recommendations are personal (chart (a) of Figure 3-6). In addition, many users stated as additional comment that they believe RUBARS is a great idea (chart (d) of Figure 3-6).

In addition, quantitative and qualitative results both indicate that users provided positive feedback regarding the ease of use and user friendliness of RUBARS. The majority of participants liked the interface of RUBARS (Q6), believed that the information provided in the recommendations were clear and easy to understand (Q9), did not find RUBARS unnecessarily complex (Q7), and believed that it did not take them a long time before they could understand the recommendations (Q8). Many users also stated that they appreciated the "easy navigation" and "ease of use and user friendliness" of RUBARS (chart (a) of Figure 3-6). In addition, several users stated that they appreciated the clear explanation of the recommendations and the way that the recommendations were presented to them (chart (a) of Figure 3-6). Despite the positive feedback provided by many participants regarding the ease of use and user friendliness of RUBARS, some users pointed out that they would like to see the user interface of the website (Moodle) to be improved and more graphics to be added to the sample course that was used for the evaluation (chart (b) of Figure 3-6). A few users also stated that they would like to see improvements in the user interface of RUBARS (chart (b) of Figure 3-6).

Moreover, quantitative and qualitative results both show that users provided positive feedback regarding user satisfaction. Most participants were satisfied with the learning tasks recommended by RUBARS (Q12) and believed that such recommendations would be helpful for students in order to perform better in a course (Q11). In addition, the majority of participants felt confident in using RUBARS (Q10). Many users also pointed out that they were satisfied with the recommendations and did not find anything missing in the recommendations provided by RUBARS (chart (c) of Figure 3-6). In addition, several users mentioned as additional comment that RUBARS is a great idea (chart (d) of Figure 3-6). Moreover, several users appreciated the personalization and presentation of recommendations (chart (a) of Figure 3-6).

To conclude, the collected feedback from users show that most participants provided positive feedback about RUBARS with respect to its (1) recommender system acceptance, (2) ease of use and user friendliness and (3) user satisfaction. Furthermore, as RUBARS is a novel idea and to the best of our knowledge, there is no other recommender system that recommends learning tasks within assignments, it can add value to the LMS community and contribute to the enhancements of LMSs. Future work will deal with the broad use of the system in different courses.

References

Akdemir, E., & Özçelik, C., 2019. The investigation of the attitudes of teachers towards using student centered teaching methods and techniques. Universal Journal of Educational Research. 7(4), 1147-1153.

Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A., 2013. Recommender systems survey. Knowledge-Based Systems. 46, 109-132.

Brooke, J., 1996. SUS: A quick and dirty usability scale, in P. W. Jordan, B. Thomas, I. L. McClelland, & B. Weerdmeester (Eds.), Usability Evaluation in Industry. Taylor and Francis, London, pp. 189-194.

Burke, R., 2002. Hybrid recommender systems: Survey and experiments. User modeling and user-adapted interaction. 12(4), 331-370.

Burke, R., 2007. Hybrid Web Recommender Systems, in P. Brusilovsky, A. Kobsa, & W. Nejdl (Eds.), The adaptive web: Methods and strategies of web personalization. Springer, Berlin, Heidelberg, pp. 377-408.

Chen, C. M., Lee, H. M., & Chen, Y. H., 2005. Personalized e-learning system using item response theory. Computers & Education. 44(3), 237-255.

Cronbach, L. J., & Meehl, P. E., 1955. Construct validity in psychological tests. Psychological Bulletin. 52(4), 281-302.

DorÇA, F. A., AraÚJo, R. D., de Carvalho, V. C., Resende, D. T., & Cattelan, R. G., 2016. An automatic and dynamic approach for personalized recommendation of learning objects considering students learning styles: An experimental analysis. Informatics in Education. 15(1), 45-62.

Dougiamas, M. (2019). Moodle. http://moodle.org/. Retrieved from http://moodle.org/

Dwivedi, P., & Bharadwaj, K., 2015. e-Learning recommender system for a group of learners based on the unified learner profile approach. Expert Systems. 32(2), 264-276.

Erdt, M., Fernández, A., & Rensing, C., 2015. Evaluating recommender systems for technology enhanced learning: A quantitative survey. IEEE Transactions on Learning Technologies. 8(4), 326-344.

Felder, R., & Soloman, B. (1997). Index of learning style questionnaire. Retrieved from https://www.webtools.ncsu.edu/learningstyles/

Felder, R., & Spurlin, J., 2005. Applications, reliability and validity of the index of learning styles. International Journal of Engineering Education. 21, 103-112.

Fraihat, S., & Shambour, Q., 2015. A framework of semantic recommender system for e-Learning. Journal of Software. 10(3), 317-330.

Holmes, K. A., & Prieto-Rodriguez, E., 2018. Student and staff perceptions of a learning management system for blended learning in teacher education. Australian Journal of Teacher Education. 43(3), 21-34.

Imran, H., Belghis-Zadeh, M., Chang, T.-W., Kinshuk, & Graf, S., 2014. A rule-based recommender system to suggest learning tasks. In Proceedings of the International Conference on Intelligent Tutoring Systems, Springer, Honolulu, 672-673.

Jonassen, D. H., 2000. Revisiting activity theory as a framework for designing student-centered learning environments. Theoretical foundations of learning environments, 89-121.

Khribi, M., Jemni, M., & Nasraoui, O., 2009. Automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval. Journal of Educational Technology & Society. 12(4), 30-42.

Knijnenburg, B. P., 2012. Conducting user experiments in recommender systems. In Proceedings of the sixth ACM conference on Recommender systems, ACM, 3-4.

Lewis, J. R., & Sauro, J., 2009. The factor structure of the system usability scale. In Proceedings of the International Conference on Human Centered Design, Springer, Berlin, Heidelberg, 94-103.

Luna-Urquizo, J., 2019. Learning management system personalization based on multi-attribute decision making techniques and intuitionistic fuzzy numbers. International Journal of Advanced Computer Science and Applications. 10(11), 669-676.

Manouselis, N. (2013). Recommender systems for learning. New York: Springer.

Martinez, L., Barranco, M. J., Pérez, L. G., & Espinilla, M., 2008. A knowledge based recommender system with multigranular linguistic information. International Journal of Computational Intelligence Systems. 1(3), 225-236.

Nafea, S. M., Siewe, F., & He, Y., 2019. On recommendation of learning objects using felder-silverman learning style model. IEEE Access. 7, 163034-163048.

Nilashi, M., Ibrahim, O., & Bagherifard, K., 2018. A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques. Expert Systems With Applications. 92, 507-520.

Nunnally, J. C. (1994). Psychometric theory 3E: Tata McGraw-Hill Education.

Pedersen, S., & Liu, M., 2003. Teachers' beliefs about issues in the implementation of a student-centered learning environment. Educational Technology Research and Development. 51(2), 57-76.

Perišić, J., Milovanović, M., & Kazi, Z., 2018. A semantic approach to enhance moodle with personalization. Computer Applications in Engineering Education. 26(4), 884-901.

Rahman, M. M., & Abdullah, N. A., 2018. A personalized group-based recommendation approach for web search in e-learning. IEEE Access. 6, 34166-34178.

Rhode, J., Richter, S., Gowen, P., Miller, T., & Wills, C., 2017. Understanding faculty use of the learning management system. Online Learning. 21(3), 68-86.

Rodríguez, P. A., Tabares, V., Mendez, N. D. D., Carranza, D. A. O., & Vicari, R. M., 2013. BROA: An agent-based model to recommend relevant learning objects from repository federations adapted to learner profile. International Journal of Interactive Multimedia and Artificial Intelligence. 2(1), 6-11.

Sabharwal, R., Hossain, M. R., Chugh, R., & Wells, M., 2018. Learning management systems in the workplace: A literature review. In Proceedings of the 2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering, IEEE, 387-393.

Salehi, M., Pourzaferani, M., & Razavi, S. A., 2013. Hybrid attribute-based recommender system for learning material using genetic algorithm and a multidimensional information model. Egyptian Informatics Journal. 14(1), 67-78.

Santos, J. R. A., 1999. Cronbach's alpha: A tool for assessing the reliability of scales. Journal of extension. 37(2), 1-5.

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J., 2001. Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th international conference on World Wide Web, ACM, Hong Kong, 285-295.

Shani, G., & Gunawardana, A., 2011. Evaluating recommendation systems, in F. Ricci, L. Rokach, B. Shapira, & P. B. Kantor (Eds.), Recommender systems handbook. Springer US, Boston, MA, pp. 257-297.

Tarus, J. K., Niu, Z., & Kalui, D., 2018. A hybrid recommender system for e-learning based on context awareness and sequential pattern mining. Soft Computing. 22(8), 2449-2461.

Tavakol, M., & Dennick, R., 2011. Making sense of Cronbach's alpha. International journal of medical education. 2, 53-55.

Van Meteren, R., & Van Someren, M., 2000. Using content-based filtering for recommendation. In Proceedings of the Machine Learning in the New Information Age, 47-56.

Wen, H., Fang, L., & Guan, L., 2012. A hybrid approach for personalized recommendation of news on the web. Expert Systems With Applications. 39(5), 5806-5814.

Zapata, A., Menéndez, V. H., Prieto, M. E., & Romero, C., 2013. A framework for recommendation in learning object repositories: An example of application in civil engineering. Advances in Engineering Software. 56, 1-14.

Chapter 4. Manuscript 2. Evaluation of PLORS – a Personalized Learning Object Recommender System for Learning Management Systems

Abstract

As e-learning has become more common in the modern world, delivering personalized learning materials to individual learners has become a main research issue. Nowadays, learning management systems (LMS) are often used to deliver the learning materials to learners. One of the challenges with LMSs is that they usually contain fixed content that has been created in the course based on the teaching style of the course creator and is presented to all learners without considering each learner's profile (e.g., learning styles, goals, prior knowledge, abilities, and interests). However, the teaching style of the course creator might not be the most beneficial style for all students with different profiles. Thus, students might not visit the materials in the same sequence that has been created in the course and as a result, they could miss some relevant learning objects or visit the learning objects in a sequence that is not the best learning path for them. This paper introduces PLORS which is a personalized recommender system that supports learners in such situation by providing recommendations of the learning objects in a sequence that fits students' profiles. These personalized recommendations can potentially help learners to visit more learning objects that are beneficial for them. This paper also presents the evaluation of PLORS investigating participants' acceptance of the system, perceived ease of use and user friendliness, and user satisfaction. Data from 50 participants were collected and quantitative and qualitative methods were used to analyse the data. The evaluation showed some promising results. As a result of the evaluation, we found that most users were satisfied with the recommendations that were provided by PLORS and believed that recommending learning objects in a personalized sequence would be helpful in increasing students' learning.

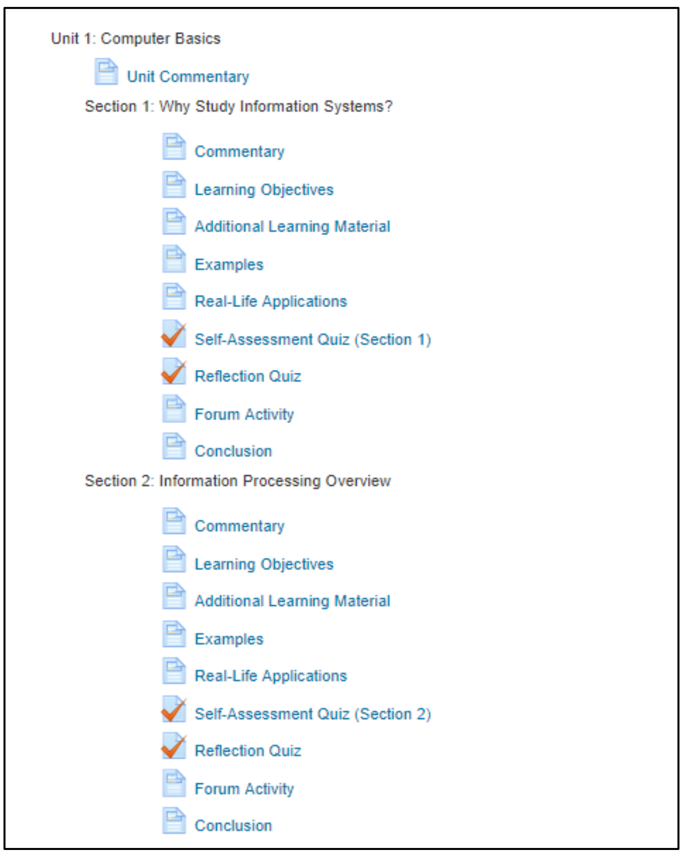
4.1 Introduction

Nowadays, learning management systems (LMS) have become a large space of educational materials and resources and are heavily used by schools and educational institutions to deliver learning materials to their students (Rhode, Richter, Gowen, Miller, & Wills, 2017). However, LMSs are still suffering from the lack of personalization. These systems usually have content that is created by instructors/teachers and the same content (with the same sequence as it was created) is presented to all learners regardless of the differences in their learning styles and personal characteristics (Bradáč, Šimík, Kotyrba, & Volná, 2017; Heng, Sangodiah, Muniandy, & Yuen, 2018). However, the large number of users and differences in their background, goals, strengths, personalities, interests and preferences makes personalization an important factor and concern in LMSs (Klašnja-Milićević, Ivanović, & Nanopoulos, 2015). According to Kinshuk and colleagues (2010), personalization can be attained through adaptive approaches. In LMSs, such approaches can consider students' profiles which may include the learning styles, experience level, prior knowledge, performance, geographical location, personality and interests of learners to decide which learning materials should be presented to each learner (Khribi, Jemni, & Nasraoui, 2009). Recommender systems are one of the tools that can be used to deliver such personalized content to learners.

In general, recommender systems are defined as tools that help users to make decisions by making suggestions or recommending contents or services to be used by the user (Burke & Ramezani, 2011). Recommender systems for learning are defined as tools that can help learners to achieve their learning goals faster by recommending learning materials or other related contents or resources to learners that fit their needs (Fraihat & Shambour, 2015). Such materials, content and resources can be an assignment, a mentor, a course, a learning object, a funding opportunity or scholarship, an academic research paper, etc. However, most literature in this area has been about learning object recommendations. The Institute of Electrical and Electronics Engineers (IEEE) defines a learning object as “any entity, digital or non-digital, which can be used, re-used or referenced during technology supported learning.” (IEEE Learning Technology Standardization Committee, 2007). As stated by Imran and colleagues (2016), learning objects can be seen as the most granular components of each course in a LMS. Different types of learning objects in a course can be exercises, animations, examples, text documents, quizzes, videos and audio files. Figure 4-1 shows a sample course within an LMS.

Figure 4-1

Sample course in LMS



As shown in Figure 4-1, learning objects in a course can be optionally grouped under different sections and sections can be grouped under different units. The sequence of the learning objects in each section and each unit is set by the teacher based on his/her teaching style (Abu Rasheed, Weber, Harrison, Zenkert, & Fathi, 2018; Bradáč et al., 2017). However, a teacher's teaching style might not be the best fit for all students in the course due to the differences in students' learning styles, experience level, prior knowledge, performance and other attributes and characteristics (Abu Rasheed et al., 2018; El-Bishouty et al., 2019). For example, Felder and Silverman (1988) proposed a teaching styles model that suggests what teaching styles are more beneficial for each learning style. Based on Felder and Silverman's teaching styles model, learners with different learning styles will benefit more from different teaching styles and as a result, different types of learning objects are more beneficial for different learners (El-Bishouty, Saito, Chang, Kinshuk, & Graf, 2013). Table 4-1 shows the relation between the learning object types and the learning styles that is proposed by El-Bishouty and colleagues (2013).

Table 4-1

The relation between the learning object types and the learning styles (El-Bishouty et al. 2013)

Learning Style	**Learning Object Types**
Active	Self-Assessment Tests, Discussion Forum Activities, Animations, Exercises
Reflective	Reflection Quizzes, Additional Reading Materials, Examples
Sensing	Self-Assessment Tests, Animations, Exercises, Examples
Intuitive	Reflection Quizzes, Additional Reading Materials
Visual	Animations
Verbal	Discussion Forum Activities, Additional Reading Materials
Sequential	No particular learning object (providing guidance is more important than the type of learning object)
Global	Real-Life Applications

In addition, different research works have shown that students with different profiles have different navigational behaviours within the course and might not visit the learning objects in the same sequence as was set by the teacher or spend less time on the learning objects that do not fit their profile (El-Bishouty et al., 2019; Graf & Kinshuk, 2008; Graf, Liu, & Kinshuk, 2010; Karagiannis & Satratzemi, 2018). Due to differences in students' navigational behaviour, they could miss some relevant learning objects or visit the learning objects in a sequence that is not the best learning path for them.

In order to support students in such situation where they do not follow the learning objects with the pre-defined sequence in the course, Imran and colleagues built an adaptive recommender system called PLORS that can be integrated into LMSs to recommend learning objects with personalized sequence to learners in a course based on learners' profile, the current learning object that is visited by the learner, and the history of visited learning objects by similar learners (Imran et al., 2016). The recommendations provided by PLORS tend to help learners to visit the learning objects with a sequence that best fits their learning style, prior knowledge, expertise level, and performance and also ensure that the student will not miss or ignore the learning objects that are beneficial for him/her. In order to generate a personalized list of recommendations of learning objects to be presented to an individual learner, PLORS forms a neighbourhood of learners based on their profiles, including their learning styles, prior knowledge, expertise level and performance within the course. Then it uses association rule mining to discover associations among learning objects to identify the useful learning objects visited by other similar learners. In this paper, an evaluation of PLORS is introduced that was performed using a user study to capture users' feedback regarding PLORS. 50 participants participated in this user study and evaluated PLORS based on its (1) recommender system acceptance, (2) ease of use and user friendliness and (3) user satisfaction.

The remainder of this paper is structured as follows: Section 4.2 presents related work. Section 4.3 introduces PLORS' architecture and approach. Section 4.4 explains the evaluation methodology and approach that was taken to evaluate the system. Section 4.5 presents the result of the evaluation. Section 4.6 provides a discussion about the results and finally, section 4.7

concludes the paper by summarizing the main contributions of the paper and discusses future research directions.

4.2 Related Work

The idea of learning object recommender systems in the educational domain has been around for years and different systems using different techniques have been developed to serve this purpose. In this section, some of these recommender systems are described based on their underlying techniques. First, we discuss recommender systems that generate recommendations for the individual learner based on similarities among the users' past activities in the course. Second, we describe the systems that group learners together based on their profiles and use similar learners' opinion (ratings or tags) to generate recommendations for the individual learner.

The first group of research works consists of recommender systems that use the associations among the activities done by learners and generate recommendations for the active learner based on the history of learner's activities within a course (Bourkoukou, El Bachari, & El Adnani, 2017; Dahdouh, Oughdir, Dakkak, & Ibriz, 2019; Salehi, Pourzaferani, & Razavi, 2013; Xiaowei & Shanshan, 2018; Zaiane, 2002). For example, Zaíane (2002) built a recommender system that recommends a set of learning activities to the learner based on the access history and past activities of the learner. Salehi and colleagues (2013) also built a recommender system that considers the users' past activities extracted from server logs as well as different attributes of the items to build a prediction model that predicts the users' interests to unseen learning objects. Bourkoukou and colleagues (2017) also proposed a recommender model for e-learning systems that considers users past activities extracted from logs and uses an association rule mining algorithm to select and recommend the appropriate learning objects to each learner. Another system was proposed by Dahdouh and colleagues (2019) that generates the list of recommended resources for each learner by finding similarities between learners past activities collected from system logs. Xiaowei and colleagues (2018) also proposed a recommender system that uses the browsing history of the users and utilizes the weighted association rule algorithm to recommend a set of personalized learning materials to each user.

The second group consists of recommender systems that use clustering techniques to group users together and use the opinions, provided ratings or generated tags by similar users to determine which learning objects are most likely interesting and beneficial for the active learner (Dwivedi & Bharadwaj, 2015; Klašnja-Milićević, Ivanović, Vesin, & Budimac, 2018; Rodríguez, Tabares, Mendez, Carranza, & Vicari, 2013; U, Chai, & Chen, 2018; Zapata, Menéndez, Prieto, & Romero, 2013). For example, Dwivedi and Bharadwaj (2015) built a recommender system that categorizes users based on their similarities and provides recommendations of learning objects to each user based on the ratings that were given to each object by other similar users. Rodríguez and colleagues (2013) also created a learning object recommender system that uses students' learning styles, educational level and language preference to build student profiles. Next, it clusters the users based on the similarities in their profiles and uses similar users' ratings to generate recommendations for individual learners. Zapata and colleagues (2013) also created a recommender system that considers users' profiles, ratings that were given to learning objects by other learners and metadata of the learning objects to find relevant learning objects and recommend them to learners. U and colleagues (2018) also proposed a recommender system that clusters the users based on their similar interests and considers the users' ratings to generate a list of recommended materials for each user. Klašnja-Milićević and colleagues (2018) proposed a system that clusters the users based on their learning styles and recommends a list of materials to each learner based on the tags that have been created by similar learners.

PLORS does not belong to any of the groups mentioned above. By using both association rules mining and clustering as well as learners' profile information and their navigational history (i.e., history of visited learning objects), PLORS combines the strengths of both groups. As mentioned above, the first group of recommender systems uses users' past activities to group learners. PLORS groups similar learners based on their profiles (e.g. learning style, expertise level, prior knowledge and performance) instead of their past activities. This approach can potentially lead to more accurate grouping because in our approach the users' profiles which are the motivators behind users' actions are considered to group users rather than the actions themselves. The second group of the above-cited recommender systems uses ratings or tags created by similar learners to decide which learning objects are useful for the active learner, while PLORS generates recommendations without requiring any such additional information from learners. Instead of using ratings or tags, PLORS uses associations among the learning objects visited by the learner and other similar learners to determine the usefulness of learning objects for the active learner.

In addition to the groups that we discussed above, we found one similar recommender system to PLORS that has been created by Khribi and colleagues (2009) which uses both association rules mining and clustering to generate recommendations for the learners. Although this system is similar to PLORS in terms of its underlying techniques, there are some differences between them. Khribi and colleagues' system groups learners together based on the similarities among the recently visited learning objects. Next, it uses associations among the learning objects visited by the active learner and other similar learners to generate recommendations for the active learner. This system and PLORS both use associations among the learning objects visited by the learner and other similar learners to provide recommendations of the learning objects. However, the two systems are different based on the way that they group learners together. As mentioned above, the system built by Khribi and colleagues (2009) groups learners based on the recently visited learning objects, however, PLORS groups similar learners based on their profiles (e.g. learning style, expertise level, prior knowledge and performance). As mentioned above, PLORS's approach can potentially lead to a more accurate grouping.

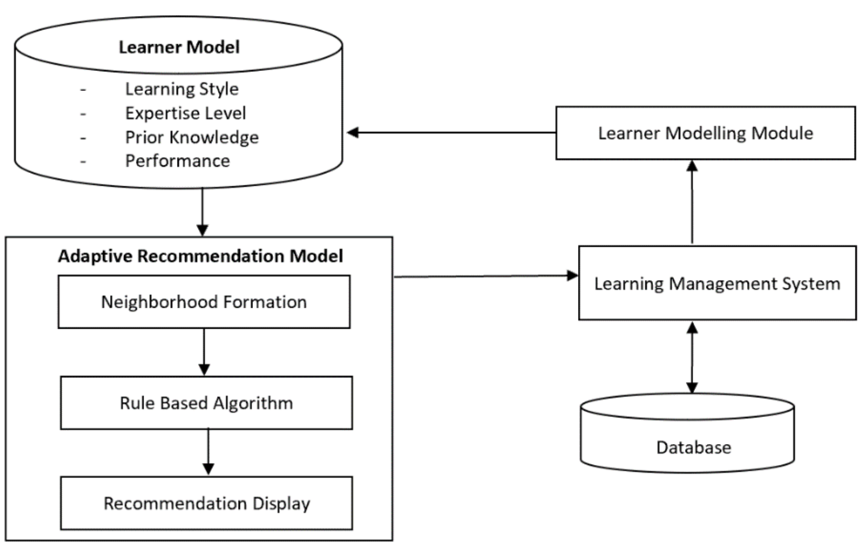
4.3 PLORS

In this section, a short overview of PLORS (Imran et al., 2016) is given, including the proposed framework for integrating PLORS into a LMS.

The main function of PLORS is to collect, analyze, and process data from learners to produce personalized recommendations to each learner, with the aim to help them improve their performance in the course. The personalized recommendations are generated based on the successful learning experience of other similar learners. PLORS consists of three modules, namely, Learner Modelling Module, Learner Model, and the Adaptivity Recommendation Module (ARM). The framework for integrating PLORS into an LMS is illustrated in Figure 4-2. In the following subsections, each of the modules is described in detail.

Figure 4-2

The architecture of PLORS



4.3.1 Learner Modelling Module

The Learning Modelling Module is at the core of PLORS. Its main function is to generate the Learner Model. The learner model stores information about a learner, including prior knowledge related to the course in which they are registered, expertise level related to the content covered in the course, learning styles to show how they prefer to learn and the ongoing performance in the course. All the information except the performance is gathered at the time of the registration. First, information about the expertise level and prior knowledge is collected through a form. Then, in order to collect information about students' learning styles, the Index of Learning Styles (ILS) questionnaire (Felder & Soloman, 1997), a commonly used questionnaire consisting of 44 questions, is shown to students. The ILS questionnaire was developed by Felder and Soloman (1997) and was found to be valid, reliable and suitable for identifying learning styles (Felder & Spurlin, 2005). The ILS identifies the learning styles of students based on the Felder Silverman learning style model (Felder & Silverman, 1988), which broadly categorizes learning styles on four dimensions: active/reflective, sensing/intuitive, visual/verbal and sequential/global. Depending on a learner's responses to the questionnaire, four values between +11 to -11 are calculated, depicting the learning style of the learner on each of the four dimensions.

The course is divided into several units and each unit has different components such as resources, assignments and quizzes. For the performance data, the scores of the learner in each gradable component is recorded.

4.3.2 Learner Model

After the Learner Modelling Module has collected the information on learners' expertise level, prior knowledge and learning styles, it passes this information to the Learner Model where it is stored. In addition, every time the grades of a learner change (e.g., when a component gets graded), the performance of the learner in the Learner Model is updated.

4.3.3. Adaptivity Recommendation Module (ARM)

The ARM is responsible for creating relevant recommendations based on the information from the Learner Model and displaying the recommendations to learners in order to help them properly select their next learning object. The ARM can consider and provide recommendations for 11 different types of learning objects namely:

1. Commentaries (give a brief overview of what the section is about)
2. Content Objects (are the learning material of the course and are rich in content)
3. Reflection Quizzes (contain open-ended questions about the topics in the section)
4. Self-Assessment Tests (include closed-ended questions about the topics in the section)
5. Discussion Forums (allow learners to ask a question and join/initiate discussions with their peers and instructor)
6. Additional Reading Materials (provide additional sources of reading materials about the topics in the section)
7. Animations (explain a concept of a section in an animated multimedia format)
8. Exercises (allow learners to practice their knowledge and skills)
9. Examples (illustrate the theoretical concepts in a more concrete way)
10. Real-Life Applications (demonstrate how the learned material can be applied in real-life situations) and
11. Conclusions (summarize the topics learned in a section).

The ARM uses two types of data typically logged by LMS: the number of clicks on a learning object (indicating the number of visits to a learning object) and the corresponding duration spent on the respective learning object (indicating engagement).

To provide recommendations, ARM identifies neighbours of the learner who have similar characteristics as the current learner. The underlying assumption is that success can be replicated by following earlier successful paths of similar learners. The idea is that if a path of particular learning objects works well for the neighbour(s) of a learner then there is a high probability that the same path would work well with the learner too.

The ARM uses performance-based comparisons and processes data algorithmically to recommend alternative/additional learning objects for faster and easier learning of the subject content. As such, the ARM achieves the purpose of identifying the most beneficial personalized learning objects for a more effective learning outcome. ARM uses three steps to achieve this goal, namely, neighbourhood formation, rule generation and recommendation display. The next subsections discuss each step in detail.

Neighbourhood Formation: The aim of this step is to find the neighbours of the current learner. Two learners are neighbours if they have similar characteristics (i.e., learning styles, prior knowledge, expertise level and performance). To calculate whether a learner is a neighbour of the current learner, Euclidean distance is used:

$$Euclidean\_distance\ (L_i, L_j) = \sqrt{\sum_{k=1}^{n} (L_{ik} - L_{jk})^2} \qquad (1)$$

L*i* and L*j* represent two learners and *k* represent the learners’ characteristics. The Euclidean distance measure shows the proximity of the two learners based on their characteristics. If this proximity/measure is smaller than a certain threshold, the two learners are considered neighbours. This threshold was set to 0.66 through experimentation, meaning that on average the difference between each characteristic of the two learners is equal or lower than 0.25 (on a scale from 0 to 1 for each characteristic).

Rule Generation: The recommendations generated for a learner are based on the observed mismatch between actions of learner L_i (current learner under observation) and his/her neighbourhood. Association rule mining is used and association rules are evaluated based on their support and confidence (Imran et al., 2016). As a simple example, one such rule could be:

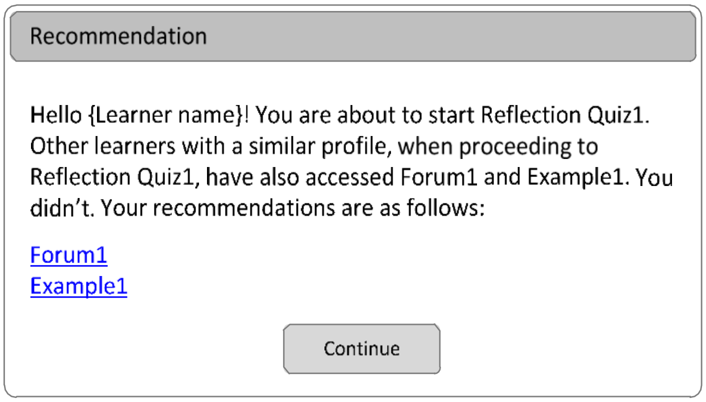
R1 : {Content Object1, Forum1}⇒ {Self-Assessment Test1}

In the above rule, the Self-Assessment Test1 corresponds to two Learning Objects {Content Object1, Forum1}. For example, if the learner L_i has not visited Content Object 1 yet and wants to attempt the Self-Assessment Test1, then the actions of the successful neighbourhood are compared to that of learner L_i through the association rule. Accordingly, the missing learning objects are suggested to learner L_i. Based on the above rule, a recommendation would be provided to learner L_i to visit Content Object1 before attempting the Self-Assessment Test1. Learner L_i can then decide whether or not he/she would like to follow this recommendation. All such behaviour data is logged by ARM and leads to an evolving population in new association rules depending on the successful completion of Self-Assessment Test1.

Recommendation Display: Once a learner visits a learning object, association rules are checked and if a mismatch between the current learner's navigational behaviour and his/her neighbourhood is detected, a recommendation is provided (see Figure 4-3). The recommendation includes a link to the corresponding learning object(s) for easier navigation.

Figure 4-3

Example of personalized recommendation



As shown above, two learning objects are offered to the learner with links to access each of them. The learner now has three options: visit each of the learning objects, visit either of the recommended objects or, ignore the recommendations and continue to go to the selected learning object.

4.4 Methodology

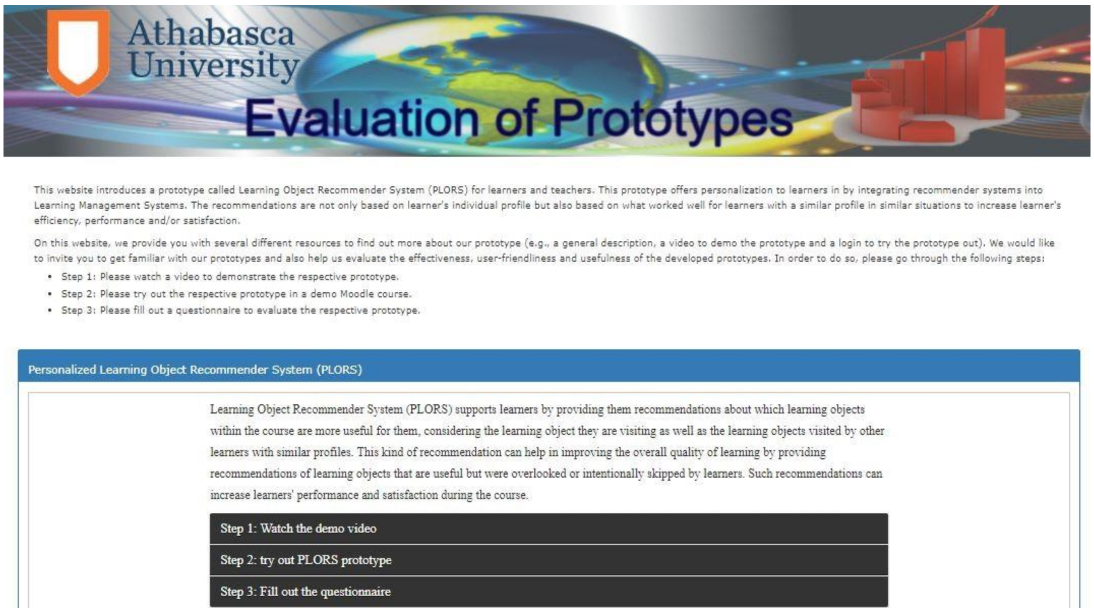
In this section, we introduce the methodology used to evaluate PLORS. The main metrics that were considered during the evaluation are (1) recommender system acceptance, (2) ease of use and user friendliness and (3) user satisfaction. The research design and participants selection are explained in the next two subsections.

4.4.1 Research Design

One common way to prove the usefulness and effectiveness of recommender systems is to conduct a user study, which is a scientific method that is commonly used to evaluate recommender systems (Erdt, Fernández, & Rensing, 2015; Knijnenburg, 2012; Shani & Gunawardana, 2011). To evaluate PLORS through a user study, a group of 50 users were asked to use the system and provide their feedback regarding their experience while interacting with the system. In order to perform the user study, a three-step process was designed and published on the web. Figure 4-4 shows the website that was designed for this evaluation.

Figure 4-4

PLORS evaluation website



Those three steps of the evaluation are as follows:

Step 1: The first step was for the participants to watch a video that contains a full demo of the system and helps participants to get familiar with PLORS and the evaluation process. In this video, the purpose and the functionalities of the system are explained, and users are walked through how the system can be used, all the way from logging-in to the system to following the recommendations that are generated by the system.

Step 2: In the second step, users were asked to use the system. For this step, PLORS was integrated into an instance of the LMS Moodle (Dougiamas, 2019) and a sample introductory course on the topic of computing and information systems containing 11 units and 174 learning objects was used. Data from 10 manually simulated users was used as the base data.

For this evaluation, each participant was provided with a login credential. Users were asked to log in to the system. Once logged in to the system, users were presented with the Index of Learning Styles questionnaire that they needed to fill out. Next, users were asked to answer the expertise level and background knowledge questions. After that, users were asked to navigate to each unit and visit the learning objects within each unit in any sequence that they like. As users navigated through the units and visited each learning object, PLORS provided them with a set of recommended learning objects to visit next. The learning objects were recommended in a sequence that best fit users' learning styles and profiles. The users could click on the recommendations to navigate directly to the recommended learning objects or they could ignore the recommendations and continue navigating through the course.

Step 3: For the third step, users were provided with a questionnaire to allow them to provide feedback regarding their experience while using PLORS. This questionnaire contained 13 multiple-choice questions (Q1 to Q13) and four open-ended questions (Q14-Q17). For the multiple-choice questions, users could rate their experience on a scale of 1 to 5, where 1 indicated strong disagreement with the respective statement and 5 indicated strong agreement. These multiple-choice questions were created based on the SUS questionnaire (Brooke, 1996; Lewis & Sauro, 2009). According to the focus of the questions, they were categorized into three categories with particular emphasis on recommender systems. These three categories are (1) Recommender System Acceptance, (2) Ease of Use and User Friendliness, and (3) User Satisfaction. In order to prove the reliability of the questions in each category, Cronbach alpha reliability coefficient was calculated for each category (Cronbach & Meehl, 1955) using the data from the 50 participants in the study. The alpha for each category is reported below:

- Recommender System Acceptance:0.82
- Ease of Use and User Friendliness:0.80
- User Friendliness:0.86

The calculated alpha values above are greater than 0.7 indicating that the questions in each category are reliable (Nunnally, 1994; Santos, 1999; Tavakol & Dennick, 2011).

In addition to the 13 multiple-choice questions, four open-ended questions were designed where participants could write their response and comments (Q14 to Q17). Moreover, participants had the option to provide us with their name, email address and occupation optionally. Questions 1 to 17 are listed below:

Recommender System Acceptance:

Q1: I would like to use PLORS frequently

Q2: I would like to see such recommendations in other courses as well

Q3: I trusted the recommendations provided by PLORS

Q4: I think recommendations provided by PLORS will be helpful in increasing students' learning

Ease of use / User Friendliness:

Q5: I liked the interface of the PLORS

Q6: I found PLORS unnecessarily complex

Q7: It took me a long time before I could understand the recommendations

Q8: The information provided in the recommendation was clear and easy to understand

User Satisfaction:

Q9: I think such recommendations would help students to visit more learning objects in the course

Q10: I felt very confident in using PLORS

Q11: I think such recommendations would help students to visit useful learning objects that were ignored by them

Q12: I think such recommendations would be helpful for students in order to perform better in a course

Q13: As a whole, I am satisfied with the recommendations provided by PLORS

Open-Ended Questions:

Q14: List aspects you appreciate most in PLORS

Q15: If you could change one thing in PLORS, whether it is major or minor, what would be at the top of the list?

Q16: Was there something missing you were expecting to see in the recommendations provided by PLORS?

Q17: Is there any other comment that you want to give for PLORS?

4.4.2 Participants Selection

For this evaluation, a new task was created and posted on Amazon Mechanical Turk (Amazon, 2019) and 50 participants accepted this task. There was no qualification defined for the participants as PLORS was designed to be used for any online course and by anyone who wants to learn a concept online. Also, prior experience using LMSs was not required for the participants, however, nowadays most people have used LMSs of some sort in the past as these systems are widely used for online and blended education by schools as well as for skill-based training in workplaces (Sabharwal, Hossain, Chugh, & Wells, 2018). Thus, different people with different occupations participated in this study. Among the 50 participants, there were graduate and undergraduate students, professors and teachers, web and application developers, accountants, nurses and marketing professionals. The fact that different people with different levels of education were included in the sample group makes us believe that a broad target audience was covered in this evaluation and the results represent the opinions of many of the potential users of PLORS. All participants went through the three-steps process described in Section 4.4.1 and completed the feedback questionnaire. Section 4.5 explains the result of the evaluation.

4.5 Results

In order to analyze the results and transform the users' answers to meaningful information, results were divided into quantitative and qualitative categories. In the following two subsections, quantitative and qualitative results are presented.

4.5.1 Quantitative Result

As the first analysis, the answers that were given to the thirteen multiple-choice questions that were collected from 50 participants were aggregated. Each question has five possible answers with respective scores provided in brackets: Strongly Agree (5), Agree (4), Neither Agree nor Disagree (3), Disagree (2) and Strongly Disagree (1). Next, the weighted average score *Wavg* was calculated for each question *q* using formula 2.

$$\mathrm{Wavg}(q) = \frac{\sum_{w=1}^{5} w.a_w}{\sum_{w=1}^{5} a_w} \tag{2}$$

In formula 2, *q* is the given question, *Wavg(q)* is the weighted average score for the given question, *w* is the score/weight of the possible answer and a_w(w=1..5) is the number of participants who selected the answer with the weight of *w* for the given question. Table 4-2 contains the response that we received from participants and the calculated weighted average score (*Wavg*) for each question.

Table 4-2

Results of quantitative analysis

	Question		Total	Strongly Agree (5)	Agree (4)	Neither Agree nor Disagree (3)	Disagree (2)	Strongly Disagree (1)	Wavg(*q*)
Recommender System Acceptance	**Q1**	I would like to use PLORS frequently	50	6	30	11	2	1	3.76
	Q2	I would like to see such recommendations in other courses as well	50	13	31	3	2	1	4.06
	Q3	I trusted the recommendations provided by PLORS	50	12	27	8	3	0	3.96
	Q4	I think recommendations provided by PLORS will be helpful in increasing students' learning	50	11	31	6	2	0	4.02
Ease of Use / User Friendliness	**Q5**	I liked the interface of the PLORS	50	5	24	9	7	5	3.34
	Q6	I found PLORS unnecessarily complex	50	2	18	3	21	6	2.78
	Q7	It took me a long time before I could understand the recommendations	50	4	7	7	21	11	2.44
	Q8	The information provided in the recommendation was clear and easy to understand	50	15	24	4	5	2	3.90
Satisfaction	**Q9**	I think such recommendations would help students to visit more learning objects in the course	50	9	33	4	4	0	3.94
	Q10	I felt very confident in using PLORS	50	14	22	9	3	2	3.86
	Q11	I think such recommendations would help students to visit useful learning objects that were ignored by them	50	6	29	8	6	1	3.66
	Q12	I think such recommendations would be helpful for students in order to perform better in a course	50	10	31	5	4	0	3.94
	Q13	As a whole, I am satisfied with the recommendations provided by PLORS	50	8	35	4	2	1	3.94

As shown in Table 4-2, the recommender system acceptance category contains four questions (Q1 to Q4). The weighted average score for Q1 is 3.76 which is between a neutral opinion and agreeing with a tendency towards agreeing that participants would like to use PLORS frequently. In Q2, users were asked whether they would like to see such recommendations in other courses as well. The weighted average score for this question is 4.06 which is between agreeing and strongly agreeing with a very strong tendency towards agreeing. The weighted average score for Q3 is 3.96 which is between a neutral opinion and agreeing with a very strong tendency towards agreeing that participants trusted the recommendations provided by PLORS. In Q4, users are asked whether they think recommendations provided by PLORS will be helpful in increasing students' learning. The weighted average score for this question is 4.02 which is between agreeing and strongly agreeing with a very strong tendency towards agreeing.

The ease of use and user friendliness category contains two sets of questions. Q5 and Q8 are positive questions. In Q5, users were asked if they liked the interface of PLORS. The weighted average score for this question is 3.34 which is between a neutral opinion and agreeing with a slight tendency towards a neutral opinion. Regarding Q8, the weighted average score is 3.90 which is between a neutral opinion and agreeing with a strong tendency towards agreeing that the information provided in the recommendations were clear and easy to understand. The other two questions in this category (Q6 and Q7) are negative questions. Q6 asks users whether they found PLORS unnecessarily complex. The weighted average score for this question is 2.78 which is between disagreeing and a neutral opinion with a tendency towards a neutral opinion. In Q7, participants were asked if it took them a long time before they could understand the recommendations. The weighted average score for this question is 2.44 which is between disagreeing and a neutral opinion.

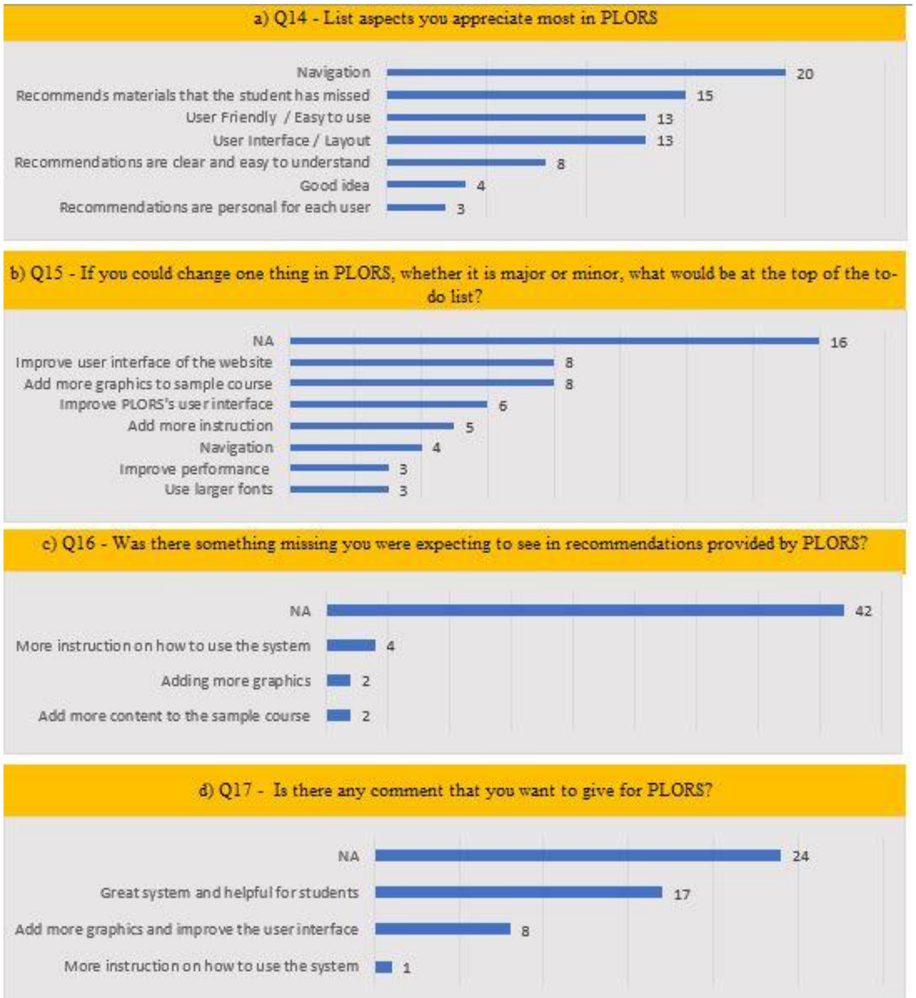
Q9 to Q13 address user satisfaction with PLORS. The weighted average score for Q9 is 3.94 which is between a neutral opinion and agreeing with a very strong tendency towards agreeing that the provided recommendations by PLORS would help students to visit more learning objects in the course. Q10 asks users whether they felt confident in using PLORS. The weighted average score for this question is 3.86 which is between a neutral opinion and agreeing with a tendency towards agreeing. The weighted average score for Q11 is 3.66 which is between a neutral opinion and agreeing with a slight tendency towards agreeing that the provided recommendations by PLORS would help students to visit useful learning objects that were ignored by them. Q12 asks users whether they think that such recommendations would be helpful for students in order to perform better in a course. The weighted average score for this question is 3.94 which is between a neutral opinion and agreeing with a very strong tendency towards agreeing. In Q13, users were asked whether they were satisfied with the recommendations provided by PLORS. The weighted average score for this question is 3.94 which is between a neutral opinion and agreeing with a very strong tendency towards agreeing.

4.5.2 Qualitative Result

As mentioned in Section 4.4, four open-ended questions were included in the questionnaire (Q14-Q17), so that users could explain their answers and provide additional feedback regarding their experience with PLORS. As these questions did not have pre-defined answers, each response was classified into topics. In order to classify the responses, each response was manually reviewed. Next, responses were grouped together based on the similarity of the topics which were discussed in the answers. For example, a comment such as “There are times I keep questioning if I have missed out anything when learning. PLORS shows me exactly what I may be missing.” Was classified under the topic of “Recommends materials that the student has missed”. An answer could include more than one topic. Next, the number of times that each topic occurred in the answers was counted. Figure 4-5 shows the result of this qualitative analysis.

Figure 4-5

Results of qualitative analysis



In Q14, users were asked to list the aspects that they appreciate most in PLORS. As shown in the chart (a) of Figure 4-5, 20 users specified that they appreciate the easy navigation between contents and recommendations. In addition, 15 participants mentioned that they appreciate the fact that the system recommends the materials that learners have missed. Ease of use, user friendliness, user interface and layout are other aspects that users liked about PLORS. Moreover, 8 users mentioned that they liked the fact that recommendations are clear and easy to understand. Also, 4 users mentioned that PLORS is a good idea and 3 users liked the fact that the generated recommendations were personal for each user.

In Q15, participants were asked to specify one thing that they would like to see changed in the system. Although the question asked for one thing to be specified, participants could optionally list more than one thing that they would like to see changed. Chart (b) of Figure 4-5 shows that 16 users specified that nothing needs to be changed in PLORS. In addition, 8 users mentioned that the interface of the website (Moodle) should be improved and 8 participants stated that the sample course that was used for the evaluation seemed to be too plain and they like to see more graphics to be added to the course. Moreover, 6 users mentioned that they like to see improvements in the user interface of PLORS. In addition, 5 users requested more instructions to be provided to the learners to help them better understand how to use the system. Improvements in navigation, system's performance and size of fonts are other aspects that were mentioned by a few participants.

In Q16, participants were requested to identify the gaps and list what they believe is missing in the system. As shown in the chart I of Figure 4-5, 42 users stated that nothing was missing in the system. Also, 4 participants believed that more explanation should be added to the demo video and the evaluation instructions. In addition, 2 users stated that it would make the user interface more attractive if more graphics are added to the course. Moreover, 2 users suggested that more content should be added to the sample course to increase the number of available learning objects that are recommended to learners.

In Q17, users were asked to add any additional comments that they might have regarding their experience while using PLORS. As shown in the chart (d) of Figure 4-5, most responders (i.e., 41 participants) either did not have any additional comments or stated that PLORS is a great system and is helpful for students. Also, 8 users mentioned that they would like to see more graphics in the course and improvements to the user interface of the website (Moodle), and one user asked for clearer instructions on how to use PLORS to be added to the evaluation instructions and the demo video.

4.6 Discussion

In this section the results of the evaluation of PLORS are discussed. In addition, the approach taken to evaluate PLORS in the light of related literature is reviewed and the differences between PLORS and similar systems in the literature are explained.

With regards to recommender system acceptance, the quantitative results presented in section 4.5 show that most participants of this evaluation study (42 out of 50) found the recommendations provided by PLORS useful and believed that such recommendations will be helpful in increasing students’ learning (as per answers given to Q4). The majority of participants (36 out of 50) also agreed that they like to use PLORS frequently (as per answers given to Q1). In addition, the majority of users (39 out of 50) agreed with the statement in Q3 and trusted the recommendations that were provided by PLORS. Moreover, most users (44 out of 50) agreed that they like to have such recommendations available to them while studying other courses (as per answers given to Q2). Qualitative results also show the positive feedback from participants regarding the recommender system acceptance. As shown in chart (c) of Figure 4-5, most participants (42 out 50) stated that nothing was missing in the recommendations provided by PLORS. In addition, 17 participants provided additional comments regarding PLORS and stressed that they believe that PLORS is a great system and is helpful for students (as shown in chart (d) of Figure 4-5). In addition, 15 participants pointed out that they appreciate that PLORS recommends the materials that the student has missed (as shown in chart (a) of Figure 4-5). Moreover, as shown in chart (a) of Figure 4-5, 7 users stated that PLORS is a good idea and appreciated the fact that the generated recommendations by PLORS are personal for each learner.

In terms of ease of use and user friendliness, the quantitative results show that the majority of participants (39 out of 50) agreed that the information provided in the recommendations were clear and easy to understand (as per answers to Q8). In addition, the majority of participants (27 out of 50) did not find PLORS unnecessarily complex (as per answers given to Q6). Similarly, the majority of users (32 out of 50) disagreed with the statement in Q7 and believed that they could understand the recommendations provided by PLORS in a reasonable amount of time. The majority of users (29 out of 50) also agreed that they liked the interface of PLORS (as per answers to Q5). Qualitative results regarding ease of use and user friendliness also show that the “easy navigation”, “user friendliness and ease of use”, and “user interface and layout” are among the top four aspects that were appreciated most by the participants. As shown in chart (a) of Figure 4-5, 20 users (out of 50) considered easy navigation as an aspect that they appreciated most in PLORS. In addition, “user friendliness and ease of use” and “user interface and layout” were each listed by 13 users (out of 50) as aspects that were most appreciated in PLORS. In response to Q14, only 6 users stated that the user interface of the PLORS needs to be improved (chart (b) of Figure 4-5), however, one observation from qualitative results regarding ease of use and user friendliness is that some of the participants raised some concerns with regards to the interface of the website (Moodle) or the sample course that was used for the evaluation. As shown in chart (b) of Figure 4-5, 19 participants mentioned that improving the user interface of the website (Moodle), adding more graphics to the sample course, and making the fonts larger would be at the top of their list to improve. Similar results are shown in charts (c) and (d) of Figure 4-5 where 2 participants stated that they expected to see more graphics and 8 users commented that more graphics should be added to improve the user interface of the course.

In terms of user satisfaction, quantitative results show that most users (43 out of 50) were satisfied with the recommendations provided by PLORS (as per answers to Q13). The majority of users (36 out of 50) also felt confident in using the system (as per answers to Q10). In addition, as per answers to Q9, most participants (42 of 50) agreed that recommendations provided by PLORS would help students to visit more learning objects in the course. Moreover, as per answers to Q11, the majority of users (35 of 50) agreed that recommendations provided by PLORS would help students to visit useful learning objects that were ignored by them. In addition, most users (41 of 50) agreed that such recommendations would be helpful for students in order to perform better in a course (as per answers to Q12). Qualitative results regarding user satisfaction also indicate that most users (i.e., 42 participants) did not see anything missing in the recommendations provided by PLORS (as shown in chart (c) of Figure 4-5). Also, as shown in chart (a) of Figure 4-5, 15 users appreciated the fact that PLORS recommends learning objects to learners that have been missed by them. Also, as shown in chart (b) of Figure 4-5, 16 participants believed that nothing needs to be changed in PLORS. In addition, as shown in chart (d) of Figure 4-5, most users (i.e., 41 participants) either did not have any concerns with PLORS or had a positive experience interacting with PLORS and believed that PLORS is a great system and will be helpful for students.

As mentioned above, the comparison of PLORS' evaluation with the evaluations of similar systems in the literature is presented in this section as well. Based on the literature review that is presented in section 4.2, it was observed that two main approaches have been used to evaluate similar recommender systems that were reviewed. The first approach focused on evaluating the accuracy of the generated recommendations and do not measure the perceived usefulness of the recommendations for the end users (Bourkoukou et al., 2017; Dahdouh et al., 2019; Dwivedi & Bharadwaj, 2015; Khribi et al., 2009; Klašnja-Milićević et al., 2018; Rodríguez et al., 2013; Salehi et al., 2013; U et al., 2018; Xiaowei & Shanshan, 2018). The second approach focused on evaluating the perceived usability and perceived usefulness of the generated recommendations for the end users by conducting user studies (Zaiane, 2002; Zapata et al., 2013). PLORS was also evaluated using the second approach by performing a user study to evaluate the perceived usability and perceived usefulness of the system for the end users. In the literature review that was presented in section 4.2, only two other evaluations of similar recommender systems using user studies were found (Zaiane, 2002; Zapata et al., 2013). Out of these two works, Zaiane (2002) did not publish the result of his user study in the paper that was reviewed. It is stated in the paper that the system was still being evaluated when the paper was published. The other user study was done by Zapata and colleagues (2013). The following differences between this user study and the user study performed to evaluate PLORS were found:

1. Zapata and colleagues (2013) used the SUS (Brooke, 1996) and the CSUQ (Lewis, 1995) questionnaires to perform their user study. In their study, participants were presented with two separate questionnaires. All questions from these questionnaires were used as one single category focusing on the usability of the system. PLORS uses a single questionnaire that was created based on the SUS questionnaire (Brooke, 1996; Lewis, 1995), however, the questions were categorized into three categories with particular emphasis on recommender systems. These three categories are (1) Recommender System Acceptance, (2) Ease of Use and User Friendliness, and (3) User Satisfaction.
2. Zapata and colleagues added an open-ended question to each questionnaire, so users could provide their comments regarding the system. The questionnaire used in the evaluation of

PLORS includes four open-ended questions to allow users to provide more feedback regarding the system.

3. Zapata and colleagues' user study was performed with 24 participants. PLORS was evaluated by 50 participants.

As mentioned above, questionnaires that are used by Zapata and colleagues and in the evaluation of PLORS are not identical. Thus, the overall results cannot be compared between these two systems, however, one of the questionnaires that Zapata and colleagues used was the SUS questionnaire. Similarly, PLORS' questionnaire has been created based on the SUS questionnaire. Thus, PLORS contains some questions that are similar to some of the questions in the questionnaire used by Zapata and colleagues.

Table 4-3 shows the comparison between similar questions in PLORS' and Zapata and colleagues' evaluation.

Table 4-3

Comparison of the results of Zapata and colleagues' user study with PLORS' results

PLORS (Questions)	(Zapata et al., 2013) (Questions)	PLORS(WAvg)	(Zapata et al., 2013) (WAvg)
I would like to use PLORS frequently	I think that I would like to use this system frequently	3.76	4.45
I liked the interface of the PLORS	I like using the interface of system the	3.34	2.48
I found PLORS unnecessarily complex	I found the system unnecessarily complex	2.78	2.37
The information provided in the recommendation was clear and easy to understand	The information provided for system is easy to understand	3.90	3.66
As a whole, I am satisfied with the recommendations provided by PLORS	Overall, I am satisfied with system	3.94	3.84

As shown in table 4-3, five similar questions from both evaluations are compared. Based on the average scores, PLORS outperforms Zapata and colleagues' system in three of five questions that focus on systems' interface, clarity of the provided information, and user satisfaction. The system created by Zapata and colleagues got higher average scores in two of the questions. One thing to note in this comparison is that PLORS was evaluated using 50 participants, but Zapata and colleagues' system was evaluated by 24 participants. In addition, the third question in table 4-3 is a negative question, so a lower score is better for this question.

4.7 Conclusion

This paper introduces PLORS which is a personalized learning object recommender system that aims to help learners improve their performance in the course by recommending learning objects in a sequence that best fits the learners' profile. Students with different profiles benefit more from different types of learning objects. They also have different navigational behaviour within the course. As a result, they might miss or intentionally ignore some learning objects in the course that would be beneficial to them. PLORS supports learners in such situations by recommending a set of learning objects to each learner in a course based on the learning object that the learner is visiting at the time of recommendation generation and the history of visiting learning objects by similar learners in the course. The focus of this paper is on evaluating different aspects of PLORS using a user study to capture users' feedback regarding the system. 50 participants participated in this research study where they were asked to use PLORS and provide feedback regarding their experience while interacting with the system. In this research, PLORS was evaluated based on its (1) recommender system acceptance, (2) ease of use and user friendliness and (3) user satisfaction. Quantitative and qualitative results show that participants provided positive feedback regarding PLORS.

The results from the quantitative and qualitative data both show that participants provided positive feedback about recommender system acceptance. Most participants believed that the recommendations provided by PLORS will be helpful in increasing students' learning (Q4) and agreed that they would like to see such recommendations available in online courses that they attend (Q2). Moreover, the majority of participants trusted the recommendations provided by PLORS (Q3) and agreed that they would like to use PLORS frequently (Q1). In addition, most participants were happy with the recommendations and did not find anything missing in the recommendations provided by PLORS (chart (c) of Figure 4-5). Several participants also stated that PLORS is a great system and is helpful for students (chart (d) of Figure 4-5). In addition, some users pointed out that PLORS is a good idea and they appreciated the fact that PLORS recommends the materials that learners have missed and that the recommendations generated by PLORS are personal for each user (chart (a) of Figure 4-5).

In addition, quantitative and qualitative results show that users provided positive feedback regarding the ease of use and user friendliness of PLORS. The majority of the participants found the information provided in the recommendations clear and easy to understand (Q8), liked the interface of PLORS (Q5), believed that it did not take them a long time before they could understand the recommendations (Q7), and did not believe that PLORS is unnecessarily complex (Q6). In addition, among the four top aspects that participants appreciated in PLORS, three were related to ease of use and user friendliness. Many users pointed out that they appreciated the easy navigation between contents and recommendations, ease of use, user friendliness, user interface and layout of PLORS (chart (c) of Figure 4-5). Beside the positive feedback from many users regarding the ease of use and user friendliness of PLORS, some users commented that they would like to see improvements in the user interface of the website (Moodle) and the sample course that was used for the evaluation. A few users also stated that they like to see improvements in the user interface of PLORS.

Moreover, quantitative and qualitative results both show that users provided positive feedback regarding user satisfaction. Most participants were satisfied with the recommendations provided by PLORS (Q13) and believed that such recommendations would be helpful for students in order to perform better in a course (Q12). In addition, most participants agreed that recommendations provided by PLORS would help students to visit more learning objects in the course (Q9). The majority of users also felt confident in using PLORS (Q10) and agreed that recommendations provided by PLORS would help students to visit more learning objects in the course that were ignored by them (Q11). In addition, most users stated that PLORS met their expectations and they did not see anything missing in the recommendations provided by PLORS (chart (c) of Figure 4-5). Several users also pointed out that they appreciated that PLORS recommends learning objects to learners that have been missed by them (chart (a) of Figure 4-5). In addition, several users stated that they were happy with the current state of the system and believed that nothing needs to be changed in PLORS (chart (b) of Figure 4-5). Many users also either did not have any concerns or additional comments regarding the system or had a positive experience interacting with PLORS and believed that PLORS is a great system and will be helpful for students (chart (d) of Figure 4-5).

Regarding PLORS' evaluation approach in the light of related literature, it was noted that only two of the reviewed systems used user studies to evaluate their systems (Zaiane, 2002; Zapata et al., 2013). One of these systems did not publish the results of their user study (Zaiane, 2002). Thus PLORS' evaluation was compared with the other user study that was done by Zapata and colleagues (2013). The questionnaire that was used by Zapata and colleagues is different from the one that was used to evaluate PLORS, however, these two questionnaires have five similar questions. Comparing the weighted average scores for these similar questions show that in three out of five questions, the weighted average score for PLORS's surpassed the weighted average score of Zapata and colleagues' system. These three questions focus on systems' interface, clarity of the provided information, and user satisfaction.

To conclude, the collected feedback from users show that most participants who participated in the evaluation study provided positive feedback about PLORS with respect to its (1) recommender system acceptance, (2) ease of use and user friendliness and (3) user satisfaction. Furthermore, based on the feedback from participants, recommendations provided by PLORS were found to have the potential to support learners in their learning process. Also, in situations that learners do not follow the learning objects in the default sequence, PLORS can potentially help learners to visit the useful learning objects in the course that have been visited by similar learners but might have been missed or ignored by the learner. Future work will deal with the broad use of the system in different courses.

References

Abu Rasheed, H., Weber, C., Harrison, S., Zenkert, J., & Fathi, M., 2018. What to learn next: Incorporating student, teacher and domain preferences for a comparative educational recommender system. In *Proceedings of the* 10th International Conference on Education and New Learning Technologies, IATED, Palma, Spain, 6790-6800.

Amazon. (2019). Amazon Mechanical Turk. Retrieved from https://www.mturk.com

Bourkoukou, O., El Bachari, E., & El Adnani, M., 2017. A recommender model in e-learning environment. Arabian Journal for Science and Engineering. 42(2), 607-617.

Bradáč, V., Šimík, M., Kotyrba, M., & Volná, E., 2017. Personalisation of a moodle course from student's perspective. In *Proceedings of the* European Conference on e-Learning, Retrieved from http://0-search.ebscohost.com.aupac.lib.athabascau.ca/login.aspx?direct=true&db=ehh&AN=126548003&site=eds-live, 70-77.

Brooke, J., 1996. SUS: A quick and dirty usability scale, in P. W. Jordan, B. Thomas, I. L. McClelland, & B. Weerdmeester (Eds.), Usability Evaluation in Industry. Taylor and Francis, London, pp. 189-194.

Burke, R., & Ramezani, M., 2011. Matching recommendation technologies and domains Recommender systems handbook. Springer, Boston, MA, pp. 367-386.

Cronbach, L. J., & Meehl, P. E., 1955. Construct validity in psychological tests. Psychological Bulletin. 52(4), 281-302.

Dahdouh, K., Oughdir, L., Dakkak, A., & Ibriz, A., 2019. Building an e-learning recommender system using association rules techniques and R environment. International Journal of Information Science & Technology. 3(2), 11-18.

Dougiamas, M. (2019). Moodle. *http://moodle.org/*. Retrieved from http://moodle.org/

Dwivedi, P., & Bharadwaj, K., 2015. e-Learning recommender system for a group of learners based on the unified learner profile approach. Expert Systems. 32(2), 264-276.

El-Bishouty, M., Aldraiweesh, A., Alturki, U., Tortorella, R., Yang, J., Chang, T.-W., . . . Kinshuk. 2019. Use of Felder and Silverman learning style model for online course design. Educational Technology Research and Development. 67(1), 161-177.

El-Bishouty, M., Saito, K., Chang, T.-W., Kinshuk, & Graf, S., 2013. An interactive course analyzer for improving learning styles support level. In *Proceedings of the* Human-Computer Interaction & Knowledge Discovery in Complex, Unstructured, Big Data, Springer, Berlin, Heidelberg, 136-147.

Erdt, M., Fernández, A., & Rensing, C., 2015. Evaluating recommender systems for technology enhanced learning: A quantitative survey. IEEE Transactions on Learning Technologies. 8(4), 326-344.

Felder, R., & Silverman, L., 1988. Learning and teaching styles in engineering education. Engineering Education. 78(7), 674-681.

Felder, R., & Soloman, B. (1997). Index of learning style questionnaire. Retrieved from https://www.webtools.ncsu.edu/learningstyles/

Felder, R., & Spurlin, J., 2005. Applications, reliability and validity of the index of learning styles. International Journal of Engineering Education. 21, 103-112.

Fraihat, S., & Shambour, Q., 2015. A framework of semantic recommender system for e-Learning. Journal of Software. 10(3), 317-330.

Graf, S., & Kinshuk. 2008. Analysing the behaviour of students in learning management systems with respect to learning styles. Advances in Semantic Media Adaptation and Personalization. 93, 53-73.

Graf, S., Liu, T.-C., & Kinshuk. 2010. Analysis of learners' navigational behaviour and their learning styles in an online course. Journal of Computer Assisted Learning. 26(2), 116-131.

Heng, L. E., Sangodiah, A., Muniandy, M., & Yuen, P. K., 2018. Integration of learner's model into learning management system environment. Journal of Fundamental & Applied Sciences. 10(1), 1771-1778.

IEEE Learning Technology Standardization Committee. (2007). The Learning Object Metadata standard. Retrieved from https://www.ieeeltsc.org/working-groups/wg12LOM/lomDescription/

Imran, H., Belghis-Zadeh, M., Chang, T.-W., Kinshuk, & Graf, S., 2016. PLORS: a personalized learning object recommender system. Vietnam Journal of Computer Science. 3(1), 3-13.

Karagiannis, I., & Satratzemi, M., 2018. Analysis of students' navigational behaviour in a moodle course. In *Proceedings of the* European Conference on e-Learning, Retrieved from http://0-search.ebscohost.com.aupac.lib.athabascau.ca/login.aspx?direct=true&db=ehh&AN=134951073&site=eds-live, 226-234.

Khribi, M., Jemni, M., & Nasraoui, O., 2009. Automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval. Journal of Educational Technology & Society. 12(4), 30-42.

Kinshuk, Chang, M., Graf, S., & Yang, G., 2010. Adaptivity and personalization in mobile learning. Technology, Instruction, Cognition & Learning. 8(2), 163-174.

Klašnja-Milićević, A., Ivanović, M., & Nanopoulos, A., 2015. Recommender systems in e-learning environments: a survey of the state-of-the-art and possible extensions. Artificial Intelligence Review. 44(4), 571-604.

Klašnja-Milićević, A., Ivanović, M., Vesin, B., & Budimac, Z., 2018. Enhancing e-learning systems with personalized recommendation based on collaborative tagging techniques. Applied Intelligence. 48(6), 1519-1535.

Knijnenburg, B. P., 2012. Conducting user experiments in recommender systems. In *Proceedings of the* sixth ACM conference on Recommender systems, ACM, 3-4.

Lewis, J. R., 1995. IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. International Journal of Human-Computer Interaction. 7(1), 57-78.

Lewis, J. R., & Sauro, J., 2009. The factor structure of the system usability scale. In *Proceedings of the* International Conference on Human Centered Design, Springer, Berlin, Heidelberg, 94-103.

Nunnally, J. C. (1994). *Psychometric theory 3E*: Tata McGraw-Hill Education.

Rhode, J., Richter, S., Gowen, P., Miller, T., & Wills, C., 2017. Understanding faculty use of the learning management system. Online Learning. 21(3), 68-86.

Rodríguez, P. A., Tabares, V., Mendez, N. D. D., Carranza, D. A. O., & Vicari, R. M., 2013. BROA: An agent-based model to recommend relevant learning objects from repository federations adapted to learner profile. International Journal of Interactive Multimedia and Artificial Intelligence. 2(1), 6-11.

Sabharwal, R., Hossain, M. R., Chugh, R., & Wells, M., 2018. Learning management systems in the workplace: A literature review. In *Proceedings of the* 2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering, IEEE, 387-393.

Salehi, M., Pourzaferani, M., & Razavi, S. A., 2013. Hybrid attribute-based recommender system for learning material using genetic algorithm and a multidimensional information model. Egyptian Informatics Journal. 14(1), 67-78.

Santos, J. R. A., 1999. Cronbach’s alpha: A tool for assessing the reliability of scales. Journal of extension. 37(2), 1-5.

Shani, G., & Gunawardana, A., 2011. Evaluating recommendation systems, in F. Ricci, L. Rokach, B. Shapira, & P. B. Kantor (Eds.), Recommender systems handbook. Springer US, Boston, MA, pp. 257-297.

Tavakol, M., & Dennick, R., 2011. Making sense of Cronbach's alpha. International journal of medical education. 2, 53-55.

U, L., Chai, Y., & Chen, J., 2018. Improved personalized recommendation based on user attributes clustering and score matrix filling. Computer Standards & Interfaces. 57, 59-67.

Xiaowei, H., & Shanshan, H., 2018. An algorithm for generating a recommended rule set based on learner's browse interest. International Journal of Emerging Technologies in Learning. 13(11), 102-116.

Zaiane, O., 2002. Building a recommender agent for e-learning systems. In *Proceedings of the* International Conference on Computers in Education, IEEE, Auckland, New Zealand, 55-59.

Zapata, A., Menéndez, V. H., Prieto, M. E., & Romero, C., 2013. A framework for recommendation in learning object repositories: An example of application in civil engineering. Advances in Engineering Software. 56, 1-14.

Chapter 5. Manuscript 3. WEBLORS – a Personalized Web-Based Recommender System

Abstract

Nowadays, personalization and adaptivity becomes more and more important in most systems. When it comes to education and learning, personalization can provide learners with better learning experiences by considering their needs and characteristics when presenting them with learning materials within courses in learning management systems. One way to provide students with more personal learning materials is to deliver personalized content from the web. However, due to information overload, finding relevant and personalized materials from the web remains a challenging task. This paper presents an adaptive recommender system called WEBLORS that aims at helping learners to overcome the information overload by providing them with additional personalized learning materials from the web to increase their learning and performance. This paper also presents the evaluation of WEBLORS based on its recommender system acceptance using data from 36 participants. The evaluation showed that overall, participants had a positive experience interacting with WEBLORS. They trusted the recommendations and found them helpful to improve learning and performance, and they agreed that they would like to use the system again.

5.1 Introduction

Although different students have different needs, learning management systems (LMSs) usually have fixed content that is presented to all students in the same way (Imran, Belghis-Zadeh, Chang, Kinshuk, & Graf, 2016). However, these systems can be enriched with personalization through recommender systems (RS). To date, many RSs are limited to recommend the available learning objects (LOs) that either have been created in the course, which greatly limits the variety of the recommendable objects, or have been collected in LO repositories (LOR) (Dwivedi &

Bharadwaj, 2015). Using LORs provides RSs with access to a larger pool of LOs, however, the quality of recommendations is highly impacted by the quality of the metadata that was provided by users who created the objects (Sabourin, Kosturko, & McQuiggan, 2015). Moreover, the available pool of LOs in a LOR could still be limited based on topics and types of LOs. However, there are more LOs and learning materials openly and freely available on the web that can be targeted by RSs (Al Abri & Dabbagh, 2018). However, due to the vast number of these objects on the web, different techniques need to be utilized to overcome the information overload and find relevant and personalized learning materials that fit students' needs (Akhtarzada, Calude, & Hosking, 2011).

In this paper, we introduce WEBLORS, a recommender system that aims at helping students by considering their individual needs and the ratings given by other learners to present the learner with additional learning materials from the web that are relevant to the learner and the topic he/she is currently learning. This paper also presents the evaluation of WEBLORS based on its recommender system acceptance.

The remainder of this paper is structured as follows: Section 5.2 presents related work. Section 5.3 discusses WEBLORS' architecture and approach. Section 5.4 explains the evaluation methodology and the results of the WEBLORS' evaluation and, section 5.5 concludes the paper.

5.2 Related Work

The idea of RSs in the learning domain has been around for decades and different recommendable objects such as courses, learning materials and academic papers have been targeted (Manouselis, 2013). However, most literature in this area has been about LO recommendations, and one of the new research trends for LO recommendations is to broaden the search and recommend LOs from web-based LORs, social networks or even from the web. There are different ways how RSs decide which LOs to recommend. Many RSs recommend learning content based on users' past activities (Bourkoukou & Bachari, 2018; Dahdouh, Oughdir, Dakkak, & Ibriz, 2019). For example, Dahdouh and colleagues (2019) proposed a recommender system that generates recommendations by considering learners' historical data as a factor and finds similarities between learners past activities collected from system logs. Another example is the system built by Bourkoukou and colleagues (2018) that generates recommendations for learners based on the user's historical data collected from server logs and other attributes of learners. Some other systems generate recommendations based on the keywords that are passed by the users (Atkinson, Gonzalez, Munoz, & Astudillo, 2014; Rahman & Abdullah, 2018; Zapata, Menéndez, Prieto, & Romero, 2013). For example, the RS built by Zapata and colleagues (2013) considers the keywords that are specified by a user and finds relevant LOs from a LOR called AGORA. Similarly, Atkinson and colleagues (2014) proposed a system that accepts the queries as input from users and uses focused crawling and metadata extraction to find relevant web resources. Rahman and colleagues (2018) also proposed a group-based recommender system that accepts users' queries, considers users' profiles, and uses Google search engine to recommend learning materials to learners based on their profiles. After reviewing the existing literature, we identified some gaps for RSs in education that we addressed in our system. First, WEBLORS recommends LOs from the web and therefore, aims at advancing our knowledge in this new trending area. Second, many RSs consider past activities of learners as a major factor when generating recommendations. Therefore, cold start is a problem in these systems. To address this issue, WEBLORS does not rely on users' past activities and instead uses learners' learning styles, the opinions of other learners (if available) and the topic that is being studied. Third, many RSs with a broad search space often work similar to search engines and heavily rely on the search criteria that are passed by the users.

In WEBLORS, this issue is avoided by creating keywords automatically through extracting them from the content that a learner is learning.

5.3 Architecture of WEBLORS

WEBLORS consists of two main parts that are shown in Figure 5-1 and Figure 5-2 and are further described in the next two subsections.

FIGURE 5-1

Architecture of CLOA

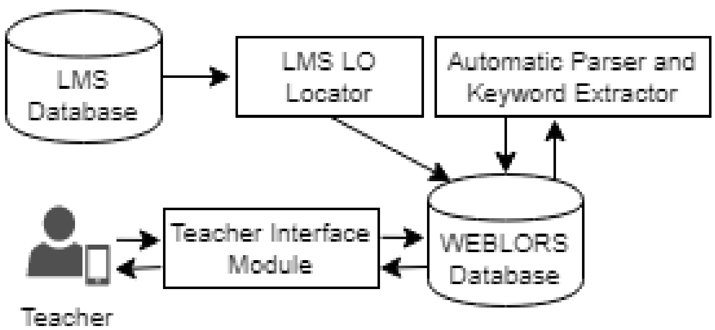
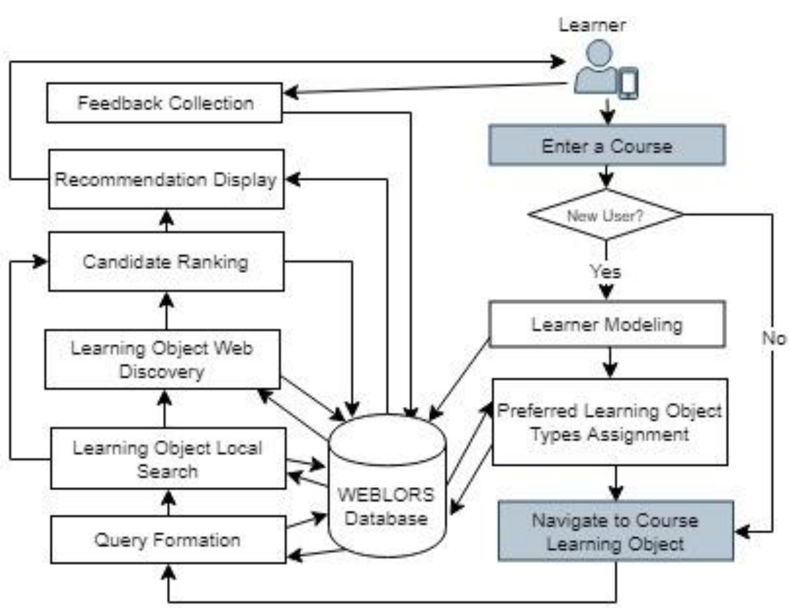


Figure 5-2

Architecture of ALORS



5.3.1 Course LOs Analyzer (CLOA)

As shown in Figure 5-1, CLOA contains a set of modules and components. The aim of the **LMS LO Locator module** is to locate all LOs within the LMS and extract their content. As part of the installation process of WEBLORS, this module searches through each course and LO in the LMS database and stores its content and the searchable criteria into the WEBLORS database (DB). Also, when a new LO is added to the LMS by a teacher, this module stores the content and the searchable criteria of the newly added LO into the DB. The aim of the **Automatic Parser and Keyword Extractor module** is to parse the content of each LO, extract a set of candidate keywords and store the keywords into the DB. This module uses the RAKE algorithm (Rose, Engel, Cramer, & Cowley, 2010) to discover the keywords and key phrases that best fit the LO. The aim of the **Teacher Interface** module is to display each LO and its extracted keywords to the instructor where he/she can confirm the accuracy, relevance and the importance of the keywords or overwrite them with a set of new keywords if required.

5.3.2 Adaptive LO Discovery and Recommender System (ALORS)

As shown in Figure 5-2, ALORS contains several modules and activities. **The aim of the Learner Modeling Module** is to capture learners' learning styles (LSs) based on Felder-Silverman Learning Style Model (Felder & Silverman, 1988), a widely known and commonly used LS model. Based on this model, learners are classified in four dimensions: (1) active/reflective (Act/Ref), (2) sensing/intuitive (Sen/Int), (3) visual/verbal (Vis/Ver) and (4) sequential/global (Seq/Glo). This module uses a questionnaire called Index of Learning Styles (ILS) (Felder & Soloman, 1997) that contains 44 questions. ILS was developed by Felder and Soloman and was found to be valid, reliable and suitable for identifying LSs (Felder & Spurlin, 2005). ILS is presented to each user when he/she enters his/her first course for the first time and based on the provided answers, his/her

LSs are calculated as four numeric values (each for one LS dimension). This module then builds a profile *(sp)* for each student *(s)* which is represented as a vector of 8 elements and is formed as *sp(s)*=(Act, Ref, Sen, Int, Vis, Ver, Seq, Glo). In *sp*, each LS dimension is represented with 2 elements where each element has a value between 0 and 2, representing the strength of the LS preference.

The aim of the Preferred Learning Object Types Assignment is to identify a set of preferred LO types (PLOTs) and their associated keywords for each learner based on their LSs. This module uses a mapping table (Table 5-1) that has been created based on the mapping proposed by El-Bishouty and colleagues (El-Bishouty, Saito, Chang, Kinshuk, & Graf, 2013) and has been extended with the LO type of videos that according to Felder and Silverman is suitable for visual and verbal learners (Felder & Silverman, 1988). In this module, each LO type (*lot*) is represented by a LO type profile (*lp*) which is a vector with the same 8 elements as the *sp*. Each element of *lp* is either 0 or 1 and is assigned per Table 5-1, indicating whether (1) or not (0) the LO is beneficial for that LS.

Table 5-1

Mapping Table (based on (El-Bishouty et al., 2013))

LO Type	LO Type Keyword	Act	Ref	Sen	Int	Vis	Ver	Seq	Glo
Exercises	exercise 1	1	0	1	0	0	0	0	0
Examples	example 1	0	1	1	0	0	0	0	1
Real Life Application	real world application	0	0	1	0	0	0	0	1
Video	video	0	0	0	0	1	1	0	0
Self-Assessment Test	questions and answers	1	0	1	0	0	0	0	0
Additional Reading Material	pdf	0	1	0	1	0	1	1	0

Next, this module calculates a numeric value for each LO type that is called **Relevance** value (Rel(*s*,*lot*)) which is the scalar product of *sp(s)* and *lp(lot),* and is used to determine the most preferred LO types for a given student with a certain LS. All LO types that have a positive Rel(*s*,*lot*) form the student's preferred LO types (PLOT).

The aim of the Query Formation Module (QFM) is to take the previously extracted keywords from the LO that the student is currently visiting and the LO type keywords associated with each PLOT of the student (per Table 5-1) as input and form one query per PLOT. WEBLORS considers three different categories of LOs when generating recommendations: (1) course LOs, (2) local LOs and (3) web LOs. Course LOs are objects that are created by the teacher and are part of the course. Local LOs are the objects that have been previously discovered from the web, recommended to learners and stored in the DB. Web LOs are the objects that are discovered from the web for the first time.

The aim of **Learning Object Local Search Module (LOLSM)** is to select a set of local LOs for each query that has been formed by the QFM and mark them as candidate local LOs and pass them to the Candidate Ranking module for further processing. Local LOs are considered to be a candidate local LO if they are of a LO type that the given query has been created for and satisfy one of the following conditions: (1) local LOs that have been previously rated (with values between 1 and 5) by five or more users and the weighted average rating for them (WAvg(*lo*)) is greater than or equals to 3.5 out of 5 (i.e., ≥ 70% of agreement) or (2) all local LOs that have been rated less than five times (to give enough chance to new local LOs to be recommended and rated by users).

The Learning Object Web Discovery Module (LOWDM) aims at using the Google API to execute the queries that are created by the QFM on the web and finding the candidate web LOs. To ensure that only educational materials are being targeted, a new Google Custom Search Engine (CSE) was created and configured to only target learning resources, scholarly articles and educational materials on the web. Also, to narrow down the search and control the number of results that are returned from the web, this module appends the index of the first result that should be returned (*start*) and the number of results that should be retrieved (*num*) to each query before running them. Both *num* and *start* parameters can be configured. The *num* parameter is set to 5 by default to enforce the query to return only 5 results at a time. In order to find at least one web LO that has not been recommended before, the *start* parameter is used in a way that if all 5 LOs that are returned by the query exist in the DB, then the system increases the *start* parameter by 5, reruns the query and returns the next 5 results until at least one new LO is found in those 5 results. At this point, the 5 results are checked and those that have not been previously recommended to any user (1 to 5 web LOs) are considered as candidate web LOs and are passed to the Candidate Ranking module. This process is repeated for each query so that there are 1-5 web LOs passed to the Candidate Ranking module for each query.

The aim of the **Candidate Ranking Module** is to accept the candidate local and web LOs from the LOLSM and LOWDM as input and decide which of them should be recommended to the learner. To generate the list of recommendations for a given student (*s*), this module calculates an **Importance** value (Imp*(lo)*) as the scalar product of the relevance value (Rel*(s,lot)*) and the weighted average rating for each candidate LO (WAvg*(lo)*). A default value of 2.5 (average rating) is used as WAvg(*lo*) for web LOs and the local LOs with less than five ratings. Next, all candidate LOs are ranked in ascending order in a way that the candidate LO with the lowest Imp(*lo*) gets the rank of 1. Subsequently, the Fitness Proportionate Selection algorithm (FPS) (Bäck, 1996) is used to select the recommendable objects in a way that the LOs with a higher Importance value have higher chance to be selected, but LOs with lower Importance value still have a small chance to be recommended. In order to select *N* candidate LOs where *N* is the number of LOs that should be recommended to the student, FPS is applied *N-1* times. Next, the list of already selected LOs is checked. If at least one LO from the web is already selected, FPS is applied one more time. Otherwise the web LO with the highest Importance value is selected as the *N*th LO.

The aim of the **Recommendation Display Module** is to accept the recommendation list from the Candidate Ranking module and display them to the learner. Also, a five-star rating system is presented for each recommended LO where the learner can rate the quality of the recommendation. The aim of the **Feedback Collection Module** is to collect the ratings that were provided by the users and store them in the DB.

5.4 Evaluation

In this section, the methodology used to evaluate the users' acceptance of the system is introduced. The research design, participants selection, and the results are explained in the next three subsections.

5.4.1 Research Design

For this evaluation, WEBLORS was integrated into an instance of Moodle (Dougiamas, 2019) and a sample course on the topic of Data Presentation in Computers was created that contained 5 LOs. Also, a four-step process was designed and published on the evaluation website where participants were asked to complete the following tasks: (1) watch a video that contains a demo of the system, (2) complete a pre-test that contains 9 questions about the course topic and one trick question, (3) login to the course, fill out the ILS, read and learn each of the LOs, and read, learn and rate the generated recommendations (5 recommendations are generated for each LO), (4) complete a post-test, which consisted of the same questions as the pre-test and can demonstrate a students' knowledge increase, (5) complete a feedback questionnaire that contains one trick question and 6 multiple-choice questions (created based on (Brooke, 1996) and (Lewis

& Sauro, 2009)) where users could rate their experience on a scale of 1 (strong disagreement) to 5 (strong agreement). Questions 1 to 6 are listed in table 5-2.

5.4.2 Participants Selection

For this evaluation, a new task was created on Amazon Mechanical Turk and 95 users accepted the task. To ensure that only valid data is included in the analysis, the following acceptance criteria were defined. Users should have completed all steps of the evaluation, answered all trick questions correctly, read at least 3 out of 5 LOs in the course, read and rated more than one third of the generated recommendations (9 or more out of 25), and spent at least 35 minutes on the sample course. Based on our assessment, at minimum, 35 minutes are required to complete the ILS, read at least 3 out of 5 LOs and 9 out of 25 recommendations and complete the post test. Although extracted times spent gathered from data logs might not be the exact time that users spent on the resources, it still provides valuable insights into the reliability of the collected data. After validating the collected data, responses from 36 participants (out of 95) met the acceptance criteria, and the rest were excluded from the evaluation.

5.4.3 Results

In order to analyze the data, the answers given to the 6 multiple-choice questions (Q1 -Q6) by the 36 accepted participants were aggregated. Each question has five possible answers that are shown in table 5-2 with respective scores provided in brackets. In addition, the weighted average score was calculated for each question. Based on the results shown in table 5-2, very high average scores have been given to Q1, Q2, Q3, Q4 and Q6 indicating that overall users agreed with the statements in these questions. These scores show that most users trusted the recommendations, found the system very useful, and believed that this system can increase learners' performance and help them in their learning process. In addition, users stated that they like to use WEBLORS frequently and have such system available to them while studying other courses. Q5 was a negative question and the low score that was given to this question shows that on average users disagreed with the statement in this question and believed that WEBLORS does not put much extra work on users to provide ratings.

Table 5-2

Results of quantitative analysis

Question	Total	Strongly Agree (5)	Agree (4)	Neither Agree nor Disagree (3)	Disagree (2)	Strongly Disagree (1)	Weighted Average Score
Q1- I would like to use WEBLORS frequently	36	16	16	1	2	1	4.22
Q2- I would like to see such recommendations in other courses as well	36	16	16	3	1	0	4.31
Q3- I trusted the recommendations provided by WEBLORS	36	21	13	1	1	0	4.50
Q4- I think recommendations provided by WEBLORS will be helpful in increasing students' performance	36	20	15	1	0	0	4.53
Q5- I think WEBLORS will put extra work on students for providing ratings	36	4	2	4	6	20	2.00
Q6- I think recommendations provided by WEBLORS will be helpful in increasing students' learning	36	17	17	1	1	0	4.39

5.5 Conclusion

The focus of this paper is on explaining the architecture of WEBLORS as well as the evaluation of the system in terms of recommender system acceptance. WEBLORS is a RS that considers the topic that the learner is learning as well as the ratings of LOs given by other learners and provides the learner with relevant learning materials from the web that are beneficial for him/her based on his/her LSs. Recommended materials are selected from a set of relevant LOs that are either discovered from the web for the first time or have been previously recommended to other learners and were given high ratings (or have been rated by less than 5 users), with the condition that at least one new LO from the web is recommended every time that WEBLORS generates recommendations. The results of the evaluation show that the 36 users provided promising feedback with respect to recommender system acceptance. Based on the result, users like to use

WEBLORS frequently and are interested to have such system available to them in other courses as well. Also, users trusted the generated recommendations and believed that the provided recommendations can help students in their learning process and will have a positive impact on students' performance. Also, the results show that most users believe that asking users to rate the recommendations does not add lots of overhead and does not put much extra work on students. To conclude, the results show that WEBLORS fills a gap in LMSs by recommending extra personalized learning materials from the web and helping with information overload by only recommending LOs relevant to the topic that is being studied and which fits students' LSs. Future work will deal with evaluating the system further based on other aspects such as ease of use, user friendliness, knowledge increase of users after using WEBLORS, and others. In addition, future work will deal with the broad use of the system in different courses.

References

Akhtarzada, A., Calude, C. S., & Hosking, J., 2011. A multi-criteria metric algorithm for recommender systems. Fundamenta Informaticae. 110, 1-11.

Al Abri, M., & Dabbagh, N., 2018. Open educational resources: A literature review. Journal of Mason Graduate Research. 6(1), 83-104.

Atkinson, J., Gonzalez, A., Munoz, M., & Astudillo, H., 2014. Web metadata extraction and semantic indexing for learning objects extraction. Applied Intelligence. 41(2), 649-664.

Bäck, T. (1996). *Evolutionary algorithms in theory and practice: Evolution strategies, evolutionary programming, genetic algorithms*. New York, NY: Oxford University Press.

Bourkoukou, O., & Bachari, E. E., 2018. Toward a hybrid recommender system for e-learning personnalization based on data mining techniques. International Journal On Informatics Visualization. 2(4), 271-278.

Brooke, J., 1996. SUS: A quick and dirty usability scale, in P. W. Jordan, B. Thomas, I. L. McClelland, & B. Weerdmeester (Eds.), Usability Evaluation in Industry. Taylor and Francis, London, pp. 189-194.

Dahdouh, K., Oughdir, L., Dakkak, A., & Ibriz, A., 2019. Building an e-learning recommender system using association rules techniques and R environment. International Journal of Information Science & Technology. 3(2), 11-18.

Dougiamas, M. (2019). Moodle. *http://moodle.org/*. Retrieved from http://moodle.org/

Dwivedi, P., & Bharadwaj, K., 2015. e-Learning recommender system for a group of learners based on the unified learner profile approach. Expert Systems. 32(2), 264-276.

El-Bishouty, M. M., Saito, K., Chang, T.-W., Kinshuk, & Graf, S., 2013. An interactive course analyzer for improving learning styles support level. In *Proceedings of the* Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data, Springer, Berlin, Heidelberg, 136-147.

Felder, R., & Silverman, L., 1988. Learning and teaching styles in engineering education. Engineering Education. 78(7), 674-681.

Felder, R., & Soloman, B. (1997). Index of learning style questionnaire. Retrieved from https://www.webtools.ncsu.edu/learningstyles/

Felder, R., & Spurlin, J., 2005. Applications, reliability and validity of the index of learning styles. International Journal of Engineering Education. 21, 103-112.

Imran, H., Belghis-Zadeh, M., Chang, T.-W., Kinshuk, & Graf, S., 2016. PLORS: a personalized learning object recommender system. Vietnam Journal of Computer Science. 3(1), 3-13.

Lewis, J. R., & Sauro, J., 2009. The factor structure of the system usability scale. In *Proceedings of the* International Conference on Human Centered Design, Springer, Berlin, Heidelberg, 94-103.

Manouselis, N. (2013). *Recommender systems for learning*. New York: Springer.

Rahman, M. M., & Abdullah, N. A., 2018. A personalized group-based recommendation approach for web search in e-learning. IEEE Access. 6, 34166-34178.

Rose, S., Engel, D., Cramer, N., & Cowley, W., 2010. Automatic keyword extraction from individual documents Text Mining. John Wiley & Sons Ltd, pp. 1-20.

Sabourin, J., Kosturko, L., & McQuiggan, S., 2015. Where to next? A comparison of recommendation strategies for navigating a learning object repository. In *Proceedings of the* International Conference on User Modeling, Adaptation, and Personalization, Springer, Cham, 208-215.

Zapata, A., Menéndez, V. H., Prieto, M. E., & Romero, C., 2013. A framework for recommendation in learning object repositories: An example of application in civil engineering. Advances in Engineering Software. 56, 1-14.

Chapter 6. Manuscript 4. Recommendations of Personalized Learning Objects from the Web Based on Users' Profiles Utilizing Google Custom Search Engine and RAKE Text Mining Algorithm

Abstract

Increasing numbers and popularity of online courses in recent years made learning management systems (LMSs) very common and frequently used. One of the main issues with LMSs is that the course content and learning materials are not tailored based on learners' needs. One way to provide students with more personalized learning materials is to deliver personalized content from the web. However, due to information overload, finding relevant and personalized materials from the web remains a challenging task. WEBLORS is an adaptive recommender system that aims at helping learners to overcome the information overload by providing them with additional personalized learning materials from the web to increase their learning and performance. This paper introduces WEBLORS and presents the evaluation of this system using system-centric and user-centric evaluations, and data from 36 participants and 30 simulated students. In this research, WEBLORS was evaluated based on its (1) keyword accuracy, (2) recommendation accuracy, (3) recommender system acceptance, (4) ease of use and user friendliness, and (5) user satisfaction. The evaluations showed very promising results. The system-centric evaluation showed high accuracy on extracted keywords as well as the generated recommendations. The user-centric evaluation showed that participants had a positive experience interacting with WEBLORS and high scores were given to all categories on which the system was evaluated.

6.1 Introduction

Nowadays, learning management systems (LMSs) are heavily used by schools and universities to deliver learning and training materials to their students (Rhode, Richter, Gowen, Miller, & Wills, 2017). However, these systems are still suffering from the lack of personalization

(Imran, Belghis-Zadeh, Chang, Kinshuk, & Graf, 2016). Although different students have different learning styles and personal characteristics, LMSs usually have fixed content that is presented to all students (Bradáč, Šimík, Kotyrba, & Volná, 2017; Graf, Kinshuk, & Ives, 2010; Heng, Sangodiah, Muniandy, & Yuen, 2018; Imran et al., 2016). One of the improvements that can be made to these systems to benefit learners is to enrich these systems with personalization through recommender systems.

One of the main challenges in providing students with personalized content is feeding the recommender system with relevant and accurate items that can be recommended to students based on their different characteristics such as learning styles and interests. To date, many recommender systems are limited to recommend the available content and learning objects (LO) that either have been created in the course or have been collected in LO repositories (Dwivedi & Bharadwaj, 2015; Fazazi, Qbadou, Salhi, & Mansouri, 2018; Imran et al., 2016; Nafea, Siewe, & He, 2019; Zaiane, 2002). Providing recommendations of LOs that exist in a course greatly limits the variety of the objects that can be recommended to the students. Using repositories provides recommender systems with access to a larger pool of LOs to choose the recommendations from, however, the quality of recommendations is highly impacted by the quality of the metadata that was provided by users who created the objects (Sabourin, Kosturko, & McQuiggan, 2015). Moreover, since rich personalization requires a large variety of different LOs, the available pool of LOs in a repository could still be limited based on topics and types of LOs.

However, there are more LOs and learning materials openly and freely available on the public domain that can be targeted by recommender systems (Al Abri & Dabbagh, 2018). Accordingly, learning materials from the web could be used to provide learners with materials that are personalized to their needs and preferences. However, due to the vast number of learning materials in different formats on the web, different techniques need to be utilized to overcome the information overload and find relevant and personalized learning materials that fit students' needs and preferences (Akhtarzada, Calude, & Hosking, 2011; Obeid, Lahoud, El Khoury, & Champin, 2018; Tarus, Niu, & Mustafa, 2018).

In this paper, we introduce an adaptive recommender system called WEBLORS that utilizes the power of the internet and the vast number of LOs on the web. WEBLORS discovers relevant learning materials from the web and delivers them to LMS users (i.e., learners) to help them in their learning process. The goal of the proposed system is to provide students with a personalized learning experience by considering their individual needs and the ratings given by other learners (if available) to present the learner with additional learning materials from the web that are relevant to the learner and the topic he/she is currently learning.

The remainder of this paper is structured as follows: Section 6.2 presents related work. Section 6.3 introduces WEBLORS' architecture and approach. Section 6.4 explains the evaluation methodology and the approach that was taken to evaluate the system. Section 6.5 presents the result of the evaluation. Section 6.6 provides a discussion about the results and finally, section 6.7 concludes the paper by summarizing the main contributions of the paper and discussing future research directions.

6.2 Related Work

In general, recommender systems are defined as tools that help users to make decisions by making suggestions or recommending contents or services to be used by the user (Burke & Ramezani, 2011). In the context of learning in particular, recommender systems are defined as techniques that help students to reach their learning goals faster by finding and providing learning materials and resources that satisfy their needs (Fraihat & Shambour, 2015). The idea of recommender systems in the learning domain has been around for decades, however, implementation techniques for these systems have improved over time and different recommendable objects and resources have been targeted (Adomavicius & Kwon, 2015; Manouselis, 2013). Such objects and resources include courses, LOs/materials, assignments, mentors, funding and scholarship opportunities, research and academic papers, and others (DorÇA, AraÚJo, de Carvalho, Resende, & Cattelan, 2016; Drachsler, Hummel, & Koper, 2008; Imran, Belghis-Zadeh, Chang, Kinshuk, & Graf, 2014). After reviewing the existing literature, it was observed that most researchers categorized recommender systems based on their underlying techniques into three main categories: (1) collaborative filtering, (2) content-based filtering, and (3) hybrid filtering (Bobadilla, Ortega, Hernando, & Gutiérrez, 2013; Sarwar, Karypis, Konstan, & Riedl, 2001; Van Meteren & Van Someren, 2000).

- Collaborative filtering: These recommender systems consider opinions or ratings that were provided by other similar users to generate recommendations (Sarwar et al., 2001). These systems decide on what should be recommended the same way that humans make decisions in real life that is typically based on help from other people's opinion (Bobadilla et al., 2013)

- Content-based filtering: These systems consider users' preferences in the past or present to generate recommendations. In other words, these systems generate recommendations by comparing the users' preferences with the available recommendable items and recommend items that are similar to items in which the user was interested in the past (Van Meteren & Van Someren, 2000).

- Hybrid filtering: These systems use multiple recommendation techniques to generate recommendations (Burke, 2007).

As mentioned above, the idea of recommender systems in the educational domain has been around for many years and different recommendable objects have been targeted, however, most literature in this area has been about LO recommendations. The Institute of Electrical and Electronics Engineers (IEEE) defines LO as "any entity, digital or non-digital, which can be used, re-used or referenced during technology supported learning." (IEEE Learning Technology Standardization Committee, 2007). In other words, LOs can be seen as the most granular components of each course in a LMS that can be used to achieve a learning objective (Fraihat & Shambour, 2015; Imran et al., 2016). Different types of LOs can be web pages, tutorials, text documents, quizzes, videos and audio files. One of the new research trends for LO recommendations is to broaden the search and recommend LOs from web-based repositories, social networks or even from the web.

There are different ways of how LO recommender systems decide which LOs to recommend. Many recommender systems recommend learning content based on users' past activities (Bourkoukou & Bachari, 2018; Dahdouh, Oughdir, Dakkak, & Ibriz, 2019; Khribi, Jemni, & Nasraoui, 2009; Sharif, Afzal, & Muhammad, 2015). For example, Sharif and colleagues (2015) designed a recommender system that generates recommendations by fetching tweets from twitter and recommending them to learners based on the learning materials that a user has viewed in the past. Another example is the system proposed by Khribi and colleagues (2009) which uses learners' navigation histories to automatically generate online recommendations. Dahdouh and colleagues (2019) also proposed a recommender system that generates recommendations by considering learners' historical data as a factor and finding similarities between learners' past activities collected from system logs. Another example is the system built by Bourkoukou and colleagues (2018) that generates recommendations for learners based on the users' historical data collected from server logs and other learners' attributes. Tarus and colleagues (2018) also proposed a recommendation approach that mines web logs to discover information such as knowledge level and learning goals about the user.

Some other systems generate recommendations based on the keywords that are passed by the users (Atkinson, Gonzalez, Munoz, & Astudillo, 2014; Rahman & Abdullah, 2018; Rodríguez, Tabares, Mendez, Carranza, & Vicari, 2013; Wang, Kun Hua, Ming Che, & Ti Kai, 2007; Zapata, Menéndez, Prieto, & Romero, 2013). For example, the recommender system built by Zapata and colleagues (2013) considers the keywords that are specified by a user, the user's profile, votes that were given to the objects by other users and metadata of the available LOs, and finds relevant LOs from a LO repository called AGORA. Similarly, Rodrigues and colleagues (2013) created a multi-agent recommender system that accepts a search string from the user and considers the user's learning style and other users' opinions to find and recommend learning materials from remote repositories. Another example is the system proposed by Wang and colleagues (2007) that considers users' keywords as a factor and recommends LOs from LO repositories on the web. Atkinson and colleagues (2014) also proposed a system that accepts the queries as input from teachers and uses focused crawling and metadata extraction to find relevant web resources. Rahman and colleagues (2018) also proposed a group-based recommender system that accepts users' queries, considers users' profiles, and uses Google search engine to recommend learning materials to learners based on their profiles.

After reviewing the existing literature, we identified some gaps for recommender systems in education that we addressed in our system. First, as mentioned above, a new trend in recommender systems in education is to broaden the search space and recommend LOs from web-based repositories, social networks or even from the web. WEBLORS recommends LOs from the web and therefore, aims at advancing our knowledge in this new trending area. Second, many recommender systems in the learning domain consider past activities of learners or existing information about the recommendable objects (such as ratings, tags or keywords) as a major factor when generating recommendations. Therefore, cold start could be a problem in these systems. To address the user cold-start issue, WEBLORS does not rely on users' past activities and instead considers the learners' learning styles, the opinions of other learners (if available) as well as the content and topic that is being studied in order to maximize the adaptivity and improve the accuracy of the recommendations. In addition, to address the item cold-start, WEBLORS does not rely on teachers to manually assign keywords to LOs that are added to the course and instead, WEBLORS automatically extracts the keywords for all LOs in the course at the time of installation as well as when a new LO is added to the course, WEBLORS automatically extracts the keywords for the newly added LO. Third, many recommender systems with a broad search space often work similar to search engines, by triggering their search based on keywords that are passed by the user (Atkinson et al., 2014; Rahman & Abdullah, 2018). This is a manual step and these recommender systems heavily rely on the search criteria that are passed by the users. In WEBLORS, this issue is avoided by creating keywords automatically through extracting them from the content that a learner is learning.

6.3 WEBLORS

In this section, we introduce WEBLORS which is an adaptive web-based recommender system that uses a combination of web mining, text mining and recommendation techniques to discover, validate, rate and categorize relevant LOs from the web and presents them within an

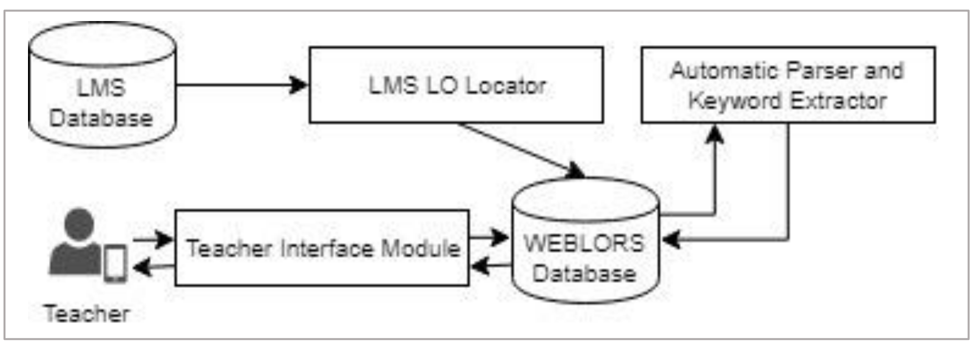
LMS course. WEBLORS applies adaptivity in both object and user dimensions and considers the similarities between objects, learners' attributes and learners' opinions about the recommended materials to maximize the personalization as well as the accuracy of the discovered objects. The system considers six types of LOs (i.e., self-assessment tests, exercises, additional reading material, real-life applications, videos, and examples) as recommendable objects. WEBLORS consists of two main parts: (1) Course LOs Analyzer (CLOA) and (2) Adaptive LO discovery and recommender system (ALORS). These parts are further described in the next two subsections.

6.3.1 Course LOs Analyzer (CLOA)

CLOA contains a set of components that are shown in Figure 6-1 and explained in the next three subsections.

Figure 6-1

Architecture of CLOA



6.3.1.1 LMS LO Locator. The aim of this module is to locate all LOs within the LMS and extract their content. As part of the installation process of WEBLORS, this module searches through each course and LO in the LMS database and stores its content, attributes and the searchable criteria into the WEBLORS database. Also, when a new LO is added to the LMS by a teacher, this module stores the content, attributes and the searchable criteria of the newly added LO into the WEBLORS database.

6.3.1.2 Automatic Parser and Keyword Extractor. The aim of this module is to parse the content of each LO, extract a set of candidate keywords for each LO and store the keywords into the WEBLORS database. In order to do so, this module performs the following tasks:

1. This module uses the RAKE algorithm (Rose, Engel, Cramer, & Cowley, 2010) which is a commonly used text mining algorithm. The RAKE algorithm splits the given text into potential keywords and key phrases by using a list of provided stop words and extracting the words between stop words and/or punctuations. The Automatic Parser and Keyword Extractor module includes a text file that contains the list of stop words. This file is utilized by the RAKE algorithm. After splitting the given text into potential keywords and key phrases, the RAKE algorithm calculates a score for each potential keyword or phrase based on the word co-occurrence graph frequency (Rose et al., 2010). The score for each potential keyword is calculated as the ratio of the word degree (the number of times it co-occurs with other potential keywords) to the word frequency. Thus, the RAKE scoring favors the words that mostly occur in longer potential key phrases and disfavors the words that appear too frequently and not in long potential key phrases (Rose et al., 2010). The minimum score for keywords is 1 and the maximum score varies based on the co-occurrence graph frequency. For potential key phrases which consist of the potential keywords that co-occur between the stop words and/or punctuations, the score is calculated as sum of the scores of all words in the phrase. Next, the RAKE algorithm generates the list of potential keywords (including key phrases) sorted based on their scores in descending order and returns the top one-third potential keywords from the list along with their scores (Li & Yan, 2019; Rose et al., 2010). The returned potential keywords with higher scores are more likely to be more important.

2. The Automatic Parser and Keyword Extractor module takes the returned list from RAKE and finds the potential keyword with the highest score (*m*) in the list. The module then takes all potential keywords from the list where their scores are greater or equal to *m*-1 and stores them in the WEBLORS database as candidate keywords. The *m*-1 threshold was set through experimentation. It was observed that selecting the top *x* potential keywords from the list is not the best approach. For example, selecting the top 1 or 2 potential keyword(s) can work well for LOs with less word length, however, 1 or 2 potential keyword(s) might not be enough to represent a LO with a large word length. As mentioned above, potential keywords with higher scores are more likely to be more important. Thus, by using a threshold of m-1, the quality (in terms of its score) of a keyword is used to decide how many potential keywords should be considered for each LO.

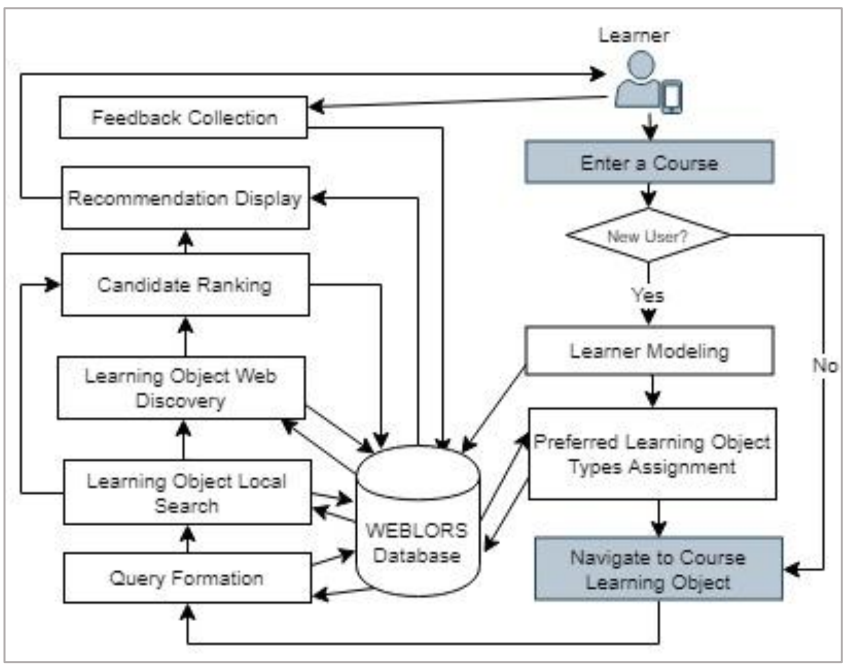
6.3.1.3 Teacher Interface Module. The aim of this module is to display each LO and its extracted keywords to the instructor where he/she can confirm the accuracy, relevance, and importance of the keywords or overwrite them with a set of new keywords if required. While this step is optional for the instructor, it can increase the accuracy of the recommendations.

6.3.2 Adaptive LO Discovery and Recommender System (ALORS)

ALORS contains several modules and activities that discover the relevant LOs from the web and from the WEBLORS database, and recommend them to learners. When a learner enters his/her first course for the first time, ALORS calculates the learning style of that student and stores his/her learning style in the WEBLORS database. Once the learner navigates to a course and visits a LO, ALORS retrieves the student’s learning style and the keywords for the LO that the learner is visiting from the WEBLORS database, identifies a set of LO types that are beneficial for that student and discovers a set of candidate LOs from the web and the WEBLORS database that are relevant to the topic the learner is learning and fit his/her learning style. Next, it decides which LOs among the discovered candidate LOs should be recommended to the learner. In addition, the integration logic with the LMS resides in ALORS where the generated recommendations are passed back and displayed inside the LMS. The different activities of ALORS are shown in Figure 6-2. The modules and components conducting those activities are further described in the next eight subsections.

Figure 6-2

Architecture of ALORS



6.3.2.1 Learner Modeling Module. The aim of this module is to calculate the learning styles of students. WEBLORS uses learners' learning styles to determine what LOs are most likely interesting and beneficial to each learner (Goldberg, Nichols, Oki, & Terry, 1992). WEBLORS uses the Felder-Silverman Learning Style Model (Felder & Silverman, 1988), a widely known and commonly used learning style model. Based on Felderand Silverman (1988), learners are classified in four dimensions: (1) active/reflective, (2) sensing/intuitive, (3) visual/verbal and (4) sequential/global. This module contains a questionnaire called Index of Learning Styles (Felder & Soloman, 1997) which was developed by Felder and Soloman and was found to be valid, reliable and suitable for identifying learning styles (Felder & Spurlin, 2005). This questionnaire contains 44 questions and is presented to each user when he/she enters his/her first course for the first time. After a student has answered the questions, the Learner Modeling module calculates his/her learning styles as four numeric values (each for one learning style dimension) between -11 and +11, and stores the calculated values in the WEBLORS database. The value for each dimension indicates the preference for that learning style dimension. Next, the calculated values are normalized by dividing them to five buckets and map each bucket to a single number between -2 and +2 (i.e., -11 to -9 are mapped to -2, -7 to -5 are mapped to -1, -3 to +3 are mapped to 0, etc.). In WEBLORS, each student is represented with a student profile (*sp*) which is a vector of 8 elements and for a given student (*s*), it is formed as *sp*(*s*)=(Active, Reflective, Sensing, Intuitive, Visual, Verbal, Sequential, Global). In *sp*, each learning style dimension is represented with 2 elements where each element has a value between 0 and 2, representing the strength of the learning style preference. For example, if the student has a value of -2 for Sensing/Intuitive dimension which means that the student has a strong preference for intuitive learning, a value of 2 is assigned to the Intuitive element and a value of 0 to the Sensing element in the student's profile.

6.3.2.2 Preferred Learning Object Types Assignment Module. The aim of this module is to assign a set of LO types to learners based on their learning styles. In order to do so, this module contains a mapping table that maps each learning style to a set of LO types that are beneficial for that learning style. This mapping table has been created based on the mapping proposed by El-Bishouty and colleagues (2013) and has been extended with the LO type of videos. El-Bishouty et al.'s mapping table as well as our mapping for videos has been created based on teaching guidelines from Felder and Silverman (1988) to consider learning styles of the Felder-Silverman learning style model (Felder & Silverman, 1988). In addition, to accurately retrieve LOs of the respective type, a set of keywords has been defined which describe each LO type. Table 6-1 illustrates the mapping table. The values in this table are either 1 or 0 where 1 means that the LO is beneficial for that learning style.

Table 6-1

LO Type to Learning Style Mapping Table (based on the work done by El-Bishouty et al. (2013))

LO Type	LO Type Keyword	Active	Reflect	Sensin	Intuitiv	Visual	Verbal	Sequen	Global
Exercises	exercise 1	1	0	1	0	0	0	0	0
Examples	example 1	0	1	1	0	0	0	0	1
Real Life Application	real world application	0	0	1	0	0	0	0	1
Video	video	0	0	0	0	1	1	0	0
Self-Assessment Test	questions and answers	1	0	1	0	0	0	0	0
Additional Reading Material	Pdf	0	1	0	1	0	1	1	0

This module contains a set of algorithms that accept the learning styles of the student as input and use the mapping table to calculate and return a set of preferred LO types (PLOT) that are associated to a learner's learning style. As explained, in the Learner Modeling module each student is represented with a vector with 8 elements. Similarly, each LO type (*lot*) is represented by a LO type profile (*lp*) which is a vector with the same 8 elements. Such profile is formed as follows: *lp*(*lot*)=(Active, Reflective, Sensing, Intuitive, Visual, Verbal, Sequential, Global). Each element of *lp* is either 0 or 1 and is assigned per Table 6-1 in a way that if the given LO type is suitable for the element, the value of 1 is assigned to that element, otherwise a value of 0 is assigned. For example, the *lp* for the Self-Assessment Test is formed as *lp*(Self-Assessment Test)=(1,0,1,0,0,0,0,0). Once the student profile (*sp*) and the LO type profile (*lp*) for each LO type are created, this module calculates a numeric value for each LO type that is called Relevance value.

The Relevance value is used to determine the most preferred LO types for a given student with a certain learning style and is calculated for each LO type using the formula 1.

$$\mathrm{Rel}(s, lot) = sp(s) * lp(lot) \qquad (1)$$

In formula 1, *s* represents a student, *lot* represents a LO type, *sp(s)* is the student profile for the give student, *lp(lot)* is the LO type profile of the given LO type and Rel(*s,lot*) is the Relevance value of the given LO type for the given student. By using this formula, the Relevance values for each LO type for a given learner are calculated. All LO types that have a Rel(*s,lot*) >0 form the student's preferred LO types (PLOT). If the Relevance values for all LO types for a given student are 0, then all LO types are considered as PLOT for that student.

6.3.2.3 Query Formation Module. The aim of this module is to form a set of queries (one query per PLOT) for each LO and each user in the course. These queries are used to find LOs from the web and the WEBLORS database that are relevant to the topic of the LO that is being viewed by a learner and are of a LO type that is preferred by him/her. This module takes the previously extracted keywords from the LO that the student is viewing and the LO type keywords associated with each PLOT (per Table 6-1) as input to form one query for each PLOT for the respective student. The generated queries and the LO types that the queries are created for are stored in the WEBLORS database. The queries that are created by this module are in the form of Keywords for LO + LO Type Keywords. As there are 6 different LO types considered by WEBLORS (see Table 6-1), this module creates up to 6 queries for each LO in the course that is being visited by a learner. The query that contains the PLOT with the highest relevance value is called *qu1*, query that contains the PLOT with the second highest relevance value is called *qu2*, etc.

6.3.2.4 Learning Object Local Search Module. WEBLORS considers three different categories of LOs when generating recommendations: (1) course LOs, (2) local LOs and (3) web

LOs. Course LOs are objects that are created by the teacher and are part of the course. Local LOs are the objects that have been previously discovered from the web, recommended to learners and stored in the WEBLORS database. Web LOs are the objects that are discovered from the web for the first time and have not been recommended to any learner before. The aim of this module is to identify the local LOs that are relevant to the course LO that the student is visiting, determine if any of them match the learners' PLOT and if so, mark them as candidate recommendable LOs. In order to select the candidate local LOs for a given course LO that is being visited by a student, this module evaluates each query that has been formed by the Query Formation module for this course LO and performs the following tasks for each query (*qu1*, *qu2*,..):

1. It first identifies all local LOs that match the PLOT of the given query.
2. Next, it loops through the local LOs that were found in step 1 and checks whether they have been previously rated by five or more users. This module then calculates the weighted average rating for the local LOs that have been previously rated by five or more users. WEBLORS uses a five-star rating system that is used by users to rate the recommendations and these ratings are used to calculate the weighted average rating in this module. Formula 2 is used for this calculation:

$$\mathrm{WAvg}(lo) = \frac{\sum_{w=1}^{5} w * t_w}{\sum_{w=1}^{5} t_w} \qquad (2)$$

In Formula 2, *lo* is the recommended LO, WAvg(*lo*) is the weighted average rating for the recommended LO, *w* is the number of stars that was given and t_w(w=1..5) is the number of the *w* star ratings that were given to the recommended LO. For example, the weighted average for a recommended LO with 6 ratings of 4 stars and 2 ratings of 3 stars is calculated as Wavg(*lo*)=(4x6 +3x2) / (6+2) = 3.75

3. Once the WAvg(*lo*) is calculated, the module removes all local LOs with WAvg(*lo*) < 3.5 as these local LOs are considered as not of high enough quality to be recommended. Then, up to five local LOs with the highest WAvg(*lo*) are considered as candidate LOs for the given query, ranked according to their WAvg(*lo*) value.
4. In addition, to give enough chance to new local LOs to be rated by users, this module considers all local LOs that have been found in step 1 and have been rated less than five times as candidate LOs for the given query.

Once the candidate local LOs are identified for each query, these candidate LOs are passed to the Candidate Ranking module for further processing.

6.3.2.5 Learning Object Web Discovery Module. While the Local Object Search Module is used to find the candidate local LOs, this module aims at finding the candidate web LOs. To do so, this module executes the queries that are created by the Query Formation module on the web and returns at least one web LO that has not been previously recommended to any learner from each query. This module accepts the queries from the Query Formation module as input and uses Google API to run these queries on the web. The returned results from each query are stored in the WEBLORS database. To ensure that only educational materials are being targeted by Google API, a new Google Custom Search Engine (CSE) was created and as part of the CSE configuration, schema.org types were defined to only target learning resources, scholarly articles, and educational materials on the web (Google, 2019). Once the CSE was created, a unique search engine ID (*cx*) was assigned to this CSE by Google. This ID is appended to each query to enforce the schema.org restrictions that have been defined in the CSE. Also, in order to narrow down the search and control the number of results that are returned from the web, this module appends some additional parameters that are provided by Google API to each query before running them. Each query that is formed by the Query Formation Module consists of the course LO keywords and the LO type keywords. This module also appends the index of the first result that should be returned (*start*) and the number of results that should be retrieved (*num*) to each query. Following is a sample call to Google API:

https://www.googleapis.com/customsearch/v1?*q=binary+base+number+systems+AND +exercise+AND+1&*cx=003343599881931238989:5kf8arilscs*&num=5&start=6*

The above sample query was formed for a student with a PLOT of exercises. In this example, "binary" and "base number systems" are the keywords for the course LO, "exercise 1" is the LO type keyword, and *cx* is the search engine ID. Also, *num*=5 indicates that 5 results should be returned from the web and *start*=6 means that the first 5 results should be ignored and the 6th result should be the first object that is being returned. The *num* parameter is set to 5 by default to enforce the query to return only 5 results at a time but this parameter can be configured by a system administrator. In order to find at least one web LO that has not been recommended before, the start parameter is used in a way that if all 5 LOs that are returned by the query exist in the WEBLORS database, then the system increases the start parameter by 5, reruns the query and returns the next 5 results until at least one new LO is found in those 5 results. At this point, the 5 results are checked and those that have not been previously recommended to any user (1 to 5 web LOs) are considered as candidate web LOs and are passed to the Candidate Ranking module. This process is repeated for each query so that there are 1-5 web LOs passed to the Candidate Ranking module for each query. By using the configurations and parameters mentioned above, on average, it takes 0.6 seconds to find 1-5 web LOs for each query.

6.3.2.6 Candidate Ranking Module. The aim of this module is to decide which candidate local LOs and candidate web LOs should be presented to the learner. This module contains a LO selection algorithm that accepts the candidate LOs that were returned from the Learning Object Local Search module and the Learning Object Web Discovery module as input and decides which LOs should be recommended to the learner. These candidate LOs are grouped into three groups: (1) LOs that were discovered from the web and have not been previously recommended to any learner (i.e., web LOs), (2) LOs that have been found by the Learning Object Local Search module and have already five or more ratings, and (3) LOs that have been returned by the Learning Object Local Search module and have either not been rated or rated by less than five learners.

The Candidate Ranking module considers the above three groups and uses the Fitness Proportionate Selection algorithm (Bäck, 1996) which is commonly used by recommender systems to select recommendable objects. To generate the list of recommendations for a given student (*s*), this module performs the following tasks:

1. First, formula 3 is used to calculate an Importance value for each candidate LO, returned from the Learning Object Web Discovery module and the Learning Object Local Search module.

$$\mathrm{Imp}(lo)= \mathrm{Rel}(s,lot) * \mathrm{WAvg}(lo) \qquad (3)$$

In formula 3, *lo* represents a given candidate LO, Imp(*lo*) is the Importance value of the given candidate LO, *s* is the given student, *lot* is the type of the candidate LO, and WAvg(*lo*) is weighted average rating for the given candidate LO. A default value of 2.5 (average rating) is used as *WAvg(lo)* for LOs that were returned from the web and the local LOs without ratings (or with less than five ratings).

2. Once the Importance values are calculated, the candidate LOs and their respective Importance values are stored in a table called Selection table. This Selection table is sorted

based on Imp(lo) in ascending order in a way that the LO with the lowest Imp(lo) is at the top of the table and the LO with the highest Imp(lo) is at the bottom of the table. In addition, a new column called Rank is added to the Selection table. The Rank column is populated in a way that the candidate LO with the lowest Imp(lo) gets the value of 1. The candidate LO with the second lowest Imp(lo) gets the rank of 2 and so on.

3. To make it possible that each candidate LOs in the Selection table can be selected, the Fitness Proportionate Selection algorithm (Bäck, 1996) is used. In this algorithm, a probability is assigned to each candidate LO so that the LOs with a higher Imp(lo) have a higher probability of being selected and recommended to learners. The probability is calculated for each LO by a function that is shown in formula 4 and is stored in the Selection table.

$$\text{If } r = 1 \text{ then } \mathrm{p}(lo_r) = \frac{\mathrm{Imp}(lo_r)}{\sum_{j=1}^{R} \mathrm{Imp}(lo_j)} \text{ else } \mathrm{p}(lo_r) = \frac{\mathrm{Imp}(lo_r)}{\sum_{j=1}^{R} \mathrm{Imp}(lo_j)} + \mathrm{p}(lo_{r-1}) \qquad (4)$$

In formula 4, r represents the associated Rank of a given candidate LO in the Selection table, $\mathrm{p}(lo_r)$ is the accumulated probability that is assigned to the given candidate LO and R is the number of candidate LOs in the Selection table. In this formula, $\mathrm{Imp}(lo_r)$ is the Importance value of the given candidate LO with rank r and $\mathrm{p}(lo_{r-1})$ is the accumulated probability of the candidate LO with rank r-1. In addition, $\sum_{j=1}^{R} \mathrm{Imp}(lo_j)$ is the summation of all Importance values that exist in the Selection table. As $\mathrm{p}(lo_r)$ is the accumulated probability value, it is always between 0 and 1.

4. Next, the Selection table is used to select N candidate LOs where N is the number of LOs that should be recommended to the student. This is done by repeating the following tasks N-1 times:

a. A random number (ran) between 0 and 1 is generated.

b. The unselected candidate LO with the lowest rank and p(lo_r) >= ran is selected and added to the list of recommendations. If all candidate LOs with p(lo_r) >= ran have been previously selected, then the unselected candidate LO with the highest p(lo_r) is selected.

5. At this point, one more candidate LO must be selected in order to have N LOs in the final list. Since WEBLORS should recommend at least one new LO from the web every time it generates recommendations, in this step, the list of already selected LOs is checked to ensure that at least one LO from the web is already selected. If this is the case, then steps 4a and 4b are repeated one more time. If all selected LOs are local LOs, then the web LO with the highest p(lo_r) is selected as the Nth LO.

6. All selected LOs are stored in the WEBLORS database.

6.3.2.7 Recommendation Display Module. The aim of this module is to accept the recommendation list from the Candidate Ranking module and display the recommendations to the learner. In addition, this module presents a five-star rating system for each recommended LO where the learner can rate their experience and choose whether they liked the recommendation. Figure 6-3 shows a sample of the presented recommendations.

Figure 6-3

Sample recommendation

As shown in Figure 6-3, accordion tabs are used to show and hide recommendations, so each recommendation expands when a user clicks on it. In addition, a search bar has been created at the bottom of the recommendation block where users can search for additional learning materials if they like. This search bar performs a regular web search through Google API and materials that are returned by performing this search are not personalized to the user.

6.3.2.8 Feedback Collection Module. The aim of this module is to collect the ratings that were provided by the users and store those ratings in the WEBLORS database. This module also logs the users' activities while visiting the recommended LOs. These activities include how many times the learner clicked on each recommendation and how much time they spent on each recommended LO.

6.4 Methodology

In this section, the methodology used to evaluate WEBLORS is introduced. The main metrics that were considered during the evaluation are (1) keyword accuracy, (2) recommendation accuracy, (3) recommender system acceptance, (4) ease of use and user friendliness, and (5) user satisfaction. The research design and participant selection are explained in the next two subsections.

6.4.1 Research Design

In order to perform the evaluation, WEBLORS was integrated into an instance of the LMS Moodle (Dougiamas, 2019) and two separate evaluations were performed. The first evaluation was a system-centric and the second one was a user-centric evaluation which are two common ways to evaluate recommender systems (Afridi, 2018; Cremonesi, Garzotto, & Turrin, 2013; Fazeli et al., 2017). System-centric evaluations focus on evaluating the accuracy of the generated recommendations and do not measure the perceived usefulness of the recommendations for the end users. User-centric evaluations focus on evaluating the perceived usability and perceived usefulness of the generated recommendations for the end users. The research design for these two evaluations are described in the following two subsections.

6.4.1.1 System-centric evaluation. In this evaluation, two simulations were performed to measure the keyword accuracy and recommendation accuracy.

The first simulation focused on evaluating the keyword accuracy. In order to perform this simulation, a sample introductory course on the topic of computing and information systems containing 5 units and 58 LOs was imported into the Moodle instance and the following steps were taken to perform this evaluation:

1. CLOA was run to automatically extract the keywords from the LOs of the newly imported course.

2. The extracted keywords from the LOs of the course (58 LOs) were manually reviewed by a domain expert who is a member of our research team. The review process was performed using the teacher interface. As mentioned in section 6.3.1.3, the extracted keywords can be reviewed on the teacher interface and each extracted keyword can be accepted or rejected. In this step, 284 keywords (extracted from 58 LOs) were reviewed and each keyword was marked as accepted if it best represented the content of the LO or rejected if it was not the best representation of the LO’s content.

3. Once all keywords were reviewed, keyword accuracy was calculated (using formula 5) as the ratio of the number of keywords that were accepted in step 2 to the total number of extracted keywords.

$$\text{Keyword Accuracy} = \frac{Number\ of\ accepted\ keywords}{Total\ number\ of\ extracted\ keywords} \tag{5}$$

The second simulation focused on evaluating the recommendation accuracy of WEBLORS. There are two measures that are commonly used to evaluate recommendations: (1) precision and (2) recall (Fazeli et al., 2017; Herlocker, Konstan, Terveen, & Riedl, 2004; Klašnja-Milićević, Vesin, & Ivanović, 2018). Precision is the measure of accuracy and recall is the measure of completeness (Krauss, Merceron, & Arbanowski, 2019). In the context of LO recommendation, precision shows what percentage of the recommended LOs are relevant and recall shows what percentage of the possible relevant LOs have been recommended to the users. As it is not possible to know the total number of the correct and relevant LOs on the web, different techniques are used by search engines and web based recommender systems to calculate the relative recall, however, even such calculation requires using different search engines and comparing the results between them (Prakash & Kumar, 2009). In WEBLORS, a single API (Google) is used to query the web, therefore relative recall was not calculated. Thus, only precision was used to measure the accuracy of the generated recommendations.

To evaluate the recommendation accuracy (i.e., calculate precision) of WEBLORS, a simulation using 30 simulated students was performed. For this evaluation, a sample course on the topic of “Data Presentation in Computers” was created in the Moodle instance, containing 5 LOs.

The following steps were taken to perform this evaluation:

1. The simulation automatically created 30 users in the Moodle instance and automatically enrolled them in the sample course.

2. Four random numbers between “-11” and “+11” (in steps of 2) were generated for each simulated student and stored in the WEBLORS database as the learning styles of that simulated student. The normalized values of the calculated learning styles (between -2 and +2) of the simulated students were calculated and also stored in the WEBLORS database.

3. Next, this simulation automatically generated recommendations for the simulated students for the LOs in the course. For each LO in the course, 5 LOs (either web LOs or local LOs, but at least one web LO) were recommended to each simulated student. As a result, 25 recommendations were generated for each simulated student. Given that we had 30 simulated students, this leads to 750 generated recommendations.

4. Next, each recommended LO (750 LOs) was reviewed manually by a domain expert who is part of our research team and was marked as correct if it satisfied the following two conditions: (1) it was of a type of PLOT of the simulated student that the recommendation was generated for (fit the learning style of the simulated student), and (2) it was relevant to the topic of the course LO that was being visited by the simulated student at the time of recommendation generation. As explained in section 6.3.2.3, each query that is created for each learner is associated with one PLOT of that learner and contains the keywords for the course LO that the learner was visiting at the time of recommendation generation. In this step, the content of each recommended LO as well as the query that was created to discover that LO were reviewed manually to check whether (1) the type of the recommended LO matched the PLOT of its associated query and (2) the content of the recommended LO was relevant to the keywords specified in the query. For example, if a recommended LO was discovered for a simulated student with the PLOT of “exercises” while visiting a course LO with “binary” and “base number systems” keywords, the recommended LO was marked as correct if it was an exercise about binary and base number systems.

5. Next, precision was calculated using the formula 6.

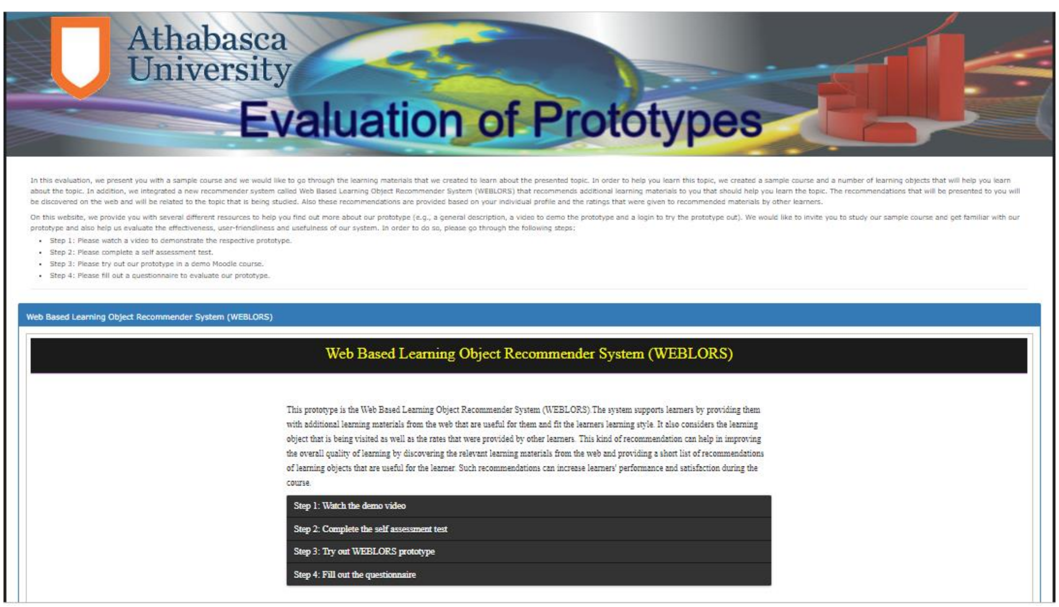
$$\text{Precision} = \frac{tp}{tp+fp} \qquad (6)$$

In formula 6, *tp* represents the number of recommendations that were marked as correct in step 4 and *fp* represents the number of recommendations that were marked as incorrect.

6.4.1.2 User-centric evaluation. For the user-centric evaluation, a user study was conducted to capture users' feedback regarding WEBLORS in terms of its (1) recommender system acceptance, (2) ease of use and user friendliness, and (3) user satisfaction. A user study is a scientific method and is one of the common ways that is used to evaluate the perceived usability and perceived usefulness of the generated recommendations for the end users (Erdt, Fernández, & Rensing, 2015; Knijnenburg, 2012; Shani & Gunawardana, 2011). To evaluate WEBLORS through a user study, a group of 36 users were asked to use the system and then provide their feedback regarding their experience while interacting with the system. For this evaluation, the same course (on the topic of "Data Presentation in Computers" containing 5 LOs) that was used for the recommendation accuracy evaluation (described in section 6.4.1.1) was used. In order to perform the user study, a four-step process was designed and published on the web. Figure 6-4 shows the website that was used for this evaluation to guide participants through the evaluation process.

Figure 6-4

WEBLORS Evaluation Website



Those four steps of the evaluation are as follows:

1. Participants were asked to watch a video that contains a full demo of the system as well as information about the evaluation.

2. Users were asked to complete a pre-test. This pre-test contains 9 questions about the course topic. The purpose of this test is to identify how much a user already knows about the topic. In addition to these 9 questions, one trick question was added as well to be able to identify participants who provided random answers. Upon completion of the pre-test, each participant was provided with a login credential.

3. Users were asked to login to the system and start studying in the sample course. Once logged in to the system, users were presented with the Index of Learning Styles questionnaire that they needed to fill out. Next, users were asked to navigate to each LO in the course (5 LOs). For each LO, the system provided learners with 5 recommendations. Users were instructed to read and learn the content of each LO, then read each recommendation that was generated for that LO, learn the content of that recommendation and rate each recommendation based on their experience. Users were instructed to study the course properly and focus on learning the content. Participants were asked that once they read and learned all LOs and recommendations, they should complete a post-test, which consisted of the same questions as the pre-test. A comparison of the results of the pre-test and post-test can be used to get information about students' knowledge increase after using WEBLORS.

4. In this step, users were provided with a feedback questionnaire that contained 16 questions (plus one trick question) and participants were asked to provide feedback regarding their experience while using WEBLORS. This questionnaire contained 12 multiple-choice questions (Q1 to Q12) and four open-ended questions (Q13 – Q16). For the multiple-choice

questions, users could rate their experience on a scale of 1 to 5, where 1 indicated strong disagreement with the respective statement and 5 indicated strong agreement. These multiple-choice questions were created based on the SUS questionnaire (Brooke, 1996; Lewis & Sauro, 2009). According to the focus of the questions, they were categorized into three categories with particular emphasis on recommender systems. These three categories are (1) Recommender System Acceptance, (2) Ease of Use and User Friendliness, and (3) User Satisfaction. In order to prove the reliability of the questions in each category, Cronbach alpha reliability coefficient was calculated for the first two categories (Cronbach & Meehl, 1955) using the data from the 36 participants in the study. The user satisfaction category contains only one question, so the Cronbach alpha was not calculated for this category. The calculated alpha values are reported below:

- Recommender System Acceptance:0.76
- Ease of Use and User Friendliness:0.85

The calculated alpha values are greater than 0.7, indicating that the questions in each category are reliable (Nunnally, 1994; Santos, 1999; Tavakol & Dennick, 2011).

In addition to the 12 multiple-choice questions, four open-ended questions were designed where participants could write their responses and comments (Q13 to Q16). Moreover, participants had the option to provide us with their name, email address and occupation optionally. Questions 1 to 16 are listed below:

Recommender System Acceptance:

Q1: I would like to use WEBLORS frequently

Q2: I would like to see such recommendations in other courses as well

Q3: I trusted the recommendations provided by WEBLORS

Q4: I think recommendations provided by WEBLORS will be helpful in increasing students' performance

Q5: I think WEBLORS will put extra work on students for providing ratings.

Q6: I think recommendations provided by WEBLORS will be helpful in increasing students' learning

Ease of Use and User Friendliness:

Q7: I liked the interface of WEBLORS

Q8: I found WEBLORS unnecessarily complex

Q9: It took me a long time before I could understand the recommendations

Q10: The information provided in the recommendation was clear and easy to understand.

Q11: I felt very confident in using WEBLORS

User Satisfaction:

Q12: As a whole, I am satisfied with the recommendations provided by WEBLORS.

Open-Ended Questions:

Q13: List aspects you appreciate most in WEBLORS

Q14: If you could change one thing in WEBLORS, whether it is major or minor, what would be at the top of the list?

Q15: Was there something missing you were expecting to see in the recommendations provided by WEBLORS?

Q16: Is there any other comment that you want to give for WEBLORS?

6.4.2 Participants Selection

As mentioned above, 30 users were simulated for the system-centric evaluation, so no actual participants were needed for this evaluation.

For the user-centric evaluation, a new task was created and posted on Amazon Mechanical Turk (Amazon, 2019) and 95 qualified workers (users with high profiles) accepted this task. The quality of the data was a major concern for this evaluation, so significant effort was put towards ensuring that the collected data is valid and reliable. The data quality was measured in different ways such as checking Moodle's activity log, using trick questions in the pre-test, post-tests, Index of Learning Styles and feedback questionnaire and using negative questions in the feedback questionnaire. In order to ensure that only valid data would be included in the analysis, the following acceptance criteria were defined:

1. Users must have completed all the steps of the evaluation and answered all questions in each questionnaire.
2. Users must have answered all trick questions correctly.
3. Users must have read at least 3 out of 5 LOs in the course. Whether or not a user has read a LO is determined by the amount of time that the user spent on the LO (extracted from Moodle's activity log) and the length (i.e., number of words) of the LO.
4. Users must have read and rated more than one-third of the generated recommendations (9 or more out of 25). If a user spent more than 1 minute on the recommendation, that recommendation is considered as read. In addition, if the user rated the recommendation as 1 (i.e., very poor), it is assumed that the content was not relevant and therefore, the recommendation is considered as read regardless of the amount of time spent on it.

5. Users must have spent at least 35 minutes on the sample course. Time spent on the course consists of the time users spent on the Index of Learning Styles, reading the LOs in the course, reading the recommendations and completing the post-test. Based on our assessment, 35 minutes is the minimum time that is required to complete the Index of Learning Styles, read at least 3 out of 5 LOs and 9 out of 25 recommendations and complete the post-test.

Although extracted times spent on questionnaires, LOs, recommendations, and the whole course gathered from data logs might not be the exact time that users spent on those resources, it still provides valuable insights into the reliability of the collected data.

For this evaluation, all participants were asked to go through the four-steps evaluation process described in section 6.4.1. Next, the collected data was validated using the acceptance criteria and data from participants with invalid responses were removed. Overall, data from 36 participants (out of 95) met the acceptance criteria, and the rest were excluded from the evaluation. On average, each of these 36 participants used the system for 182 minutes and read and rated 15 out of 25 recommendations.

6.5 Results

In this section, the result of the system-centric and user-centric evaluations are presented.

6.5.1 System-centric evaluation result

For keyword accuracy, as mentioned in section 6.4.1.1, 284 keywords that were automatically extracted by the simulation were reviewed manually. Out of the 284 extracted keywords, 203 keywords were accepted, and 81 keywords were rejected by the domain expert. Next, the keyword accuracy was calculated using formula 5. The result of this evaluation is shown in table 6-2.

Table 6-2

Results of keyword accuracy analysis

Total number of extracted keywords	Accepted	Rejected	Keyword Accuracy
284	203	81	0.7148

As shown in table 6-2, the keyword accuracy is 0.7148, indicating that 71.48% of the extracted keywords were extracted correctly and best represented the content of the LO. Among the 58 LOs that were used in this simulation, there were LOs with different word lengths. This gives us confidence that similar result would be achieved with other course materials as well.

For recommendation accuracy, as mentioned in section 6.4.1.1, 750 recommendations that were generated by the simulation were reviewed manually. Next, precision was calculated using formula 6. The result of this evaluation is shown in table 6-3.

Table 6-3

Results of recommendation accuracy analysis

Total number of recommendations	Correct recommendations	Incorrect Recommendations	Precision
750	602	148	0.8026

As shown in table 6-3, 602 (out of 750) of the generated recommendations by WEBLORS are marked as correct and precision was calculated as 0.8026. The calculated precision indicates that 80.26% of the generated recommendations were (1) of a type of PLOT of the student that the recommendation was generated for and (2) were relevant to the topic of the course LO that was being visited by the simulated student at the time of recommendation generation.

6.5.2 User-centric evaluation result

As mentioned in section 6.4.1.2, for the user-centric evaluation, a user study was conducted and users' feedback regarding WEBLORS was collected. In order to analyze the results and transform the users' answers to meaningful information, results were divided into quantitative and qualitative categories. In the following two subsections, quantitative and qualitative results are presented.

6.5.2.1 Quantitative Results. For the quantitative analysis, the answers that were given to the twelve multiple-choice questions that were collected from 36 participants were aggregated. Each question has five possible answers with respective scores provided in brackets: Strongly Agree (5), Agree (4), Neither Agree nor Disagree (3), Disagree (2), and Strongly Disagree (1). Next, the weighted average score *WAvg* was calculated for each question *q* using formula 7.

$$\mathrm{WAvg}(q) = \frac{\sum_{w=1}^{5} w.a_w}{\sum_{w=1}^{5} a_w} \qquad (7)$$

In formula 7, *q* is the given question, WAvg*(q)* is the weighted average score for the given question, *w* is the score/weight of the possible answer and a_w(w=1..5) is the number of participants who selected the answer with the score/weight of *w* for the given question. For example, 36 participants answered Q1. Out of these 36 answers, 16 participants selected "Strongly Agree", 16 participants selected "Agree", 1 participant selected "Neither Agree or Disagree", 2 participants selected "Disagree" and 1 participant selected "Strongly Disagree" as their answers. Based on these answers, the weighted average score for Q1 was calculated as follows:

$$\mathrm{WAvg}(Q1) = \frac{16\text{x}5+16\text{x}4+1\text{x}3+2\text{x}2+1\text{x}1}{16+15+1+2+1} = 4.22$$

Accordingly, the weighted average score for Q1 (WAvg*(Q1)*) is 4.22 out of 5.

Among the twelve multiple choice questions, there are nine positive and three negative questions. Table 6-4 contains the responses that were received from participants and the calculated weighted average score (*WAvg*) for each positive question. Table 6-5 contains the same information for negative questions.

Table 6-4

Results of quantitative analysis for positive questions

	Question		Total	Strongly Agree (5)	Agree (4)	Neither Agree nor Disagree (3)	Disagree (2)	Strongly Disagree (1)	WAvg(q)
Recommender System Acceptance	**Q1**	I would like to use WEBLORS frequently	36	16	16	1	2	1	4.22
	Q2	I would like to see such recommendations in other courses as well	36	16	16	3	1	0	4.31
	Q3	I trusted the recommendations provided by WEBLORS	36	21	13	1	1	0	4.50
	Q4	I think recommendations provided by WEBLORS will be helpful in increasing students' performance	36	20	15	1	0	0	4.53
	Q6	I think recommendations provided by WEBLORS will be helpful in increasing students' learning	36	17	17	1	1	0	4.39
Ease of Use / User Friendliness	**Q7**	I liked the interface of WEBLORS	36	18	11	2	4	1	4.14
	Q10	The information provided in the recommendation was clear and easy to understand	36	18	11	4	2	1	4.19
	Q11	I felt very confident in using WEBLORS	36	19	13	2	1	1	4.33
Satisfaction	**Q12**	As a whole, I am satisfied with the recommendations provided by WEBLORS.	36	21	12	2	1	0	4.47

Table 6-5

Results of quantitative analysis for negative questions

	Question		Total	Strongly Agree (5)	Agree (4)	Neither Agree nor Disagree (3)	Disagree (2)	Strongly Disagree (1)	WAvg(q)
Recommender System Acceptance	**Q5**	I think WEBLORS will put extra work on students for providing ratings.	36	4	2	4	6	20	2.00
Ease of Use / User Friendlines	**Q8**	I found WEBLORS unnecessarily complex	36	2	4	3	13	14	2.08
	Q9	It took me a long time before I could understand the recommendations	36	1	6	5	9	15	2.14

The recommender system acceptance category contains five positive questions (Q1, Q2, Q3, Q4, and Q6) and one negative question (Q5). As shown in table 6-4, very high average scores have been given to the positive questions in this category. The weighted average score for Q1 is 4.22 which is between agreeing and strongly agreeing with a tendency towards agreeing that participants would like to use WEBLORS frequently. In Q2, users were asked whether they would like to see such recommendations in the other courses as well. The weighted average score for this question is 4.31 which is between agreeing and strongly agreeing with a slight tendency towards agreeing. Regarding the question whether participants trusted the recommendations provided by WEBLORS (Q3), the weighted average score is 4.50 which is between agreeing and strongly agreeing. In Q4, users were asked whether they think that the recommendations provided by WEBLORS will be helpful in increasing students’ performance. The weighted average score for this question is 4.53 which is between agreeing and strongly agreeing. Regarding the question whether participants think that the recommendations provided by WEBLORS will be helpful in increasing students' learning (Q6), the weighted average score is 4.39 which is between agreeing and strongly agreeing with a slight tendency towards agreeing. In Q5 which is a negative question in recommender system acceptance category, users are asked whether they think WEBLORS will put extra work on students for providing ratings. As shown in table 6-5, the weighted average score for this question is 2.00 which indicates users' disagreement with the statement in this question.

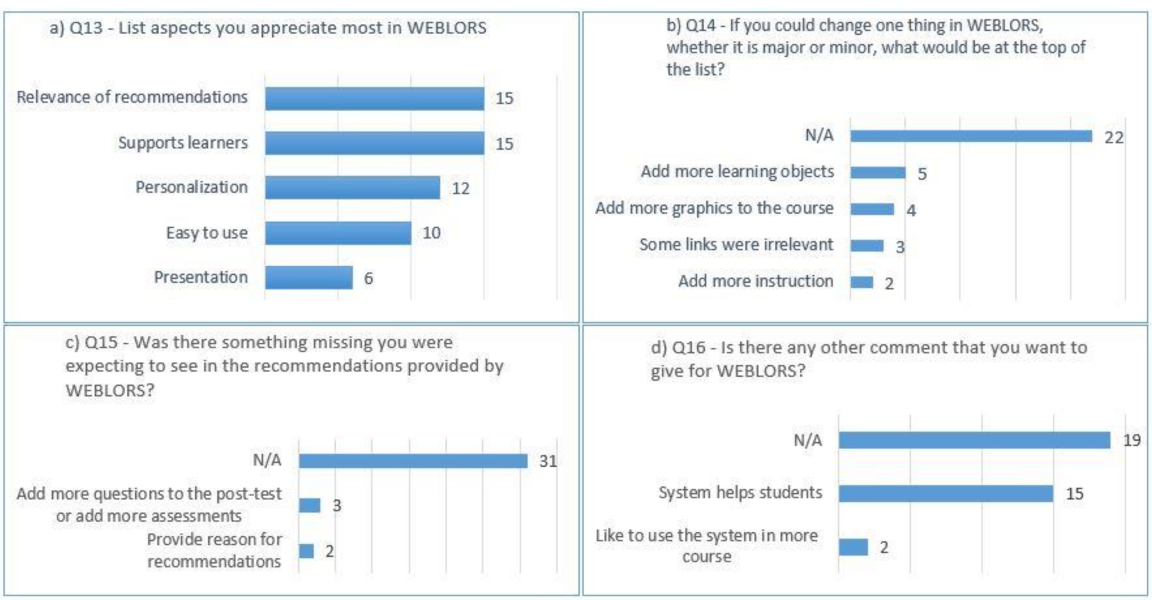
The ease of use and user friendliness category contains again two sets of questions. Q7, Q10, and Q11 are positive questions. As shown in table 6-4, in Q7, participants were asked whether they liked the interface of WEBLORS. The weighted average score for this question is 4.14 which is between agreeing and strongly agreeing with a strong tendency towards agreeing. The weighted average score for Q10 is 4.19 which is again between agreeing and strongly agreeing with a strong tendency towards agreeing. Regarding the question whether participants felt confident in using WEBLORS (Q11), the weighted average score is 4.33 which is between agreeing and strongly agreeing with a slight tendency towards agreeing. The other two questions in the ease of use and user friendliness category (Q8 and Q9) are negative questions. As shown in table 6-5, in Q8 users were asked whether they found WEBLORS unnecessarily complex. The weighted average score for this question is 2.08 which is between a neutral opinion and disagreeing with a very strong tendency towards disagreeing. In Q9, participants were asked if it took them a long time before they could understand the recommendations. The weighted average score for this question is 2.14 which is between a neutral opinion and disagreeing with a strong tendency towards disagreeing.

Q12 addresses user satisfaction with WEBLORS. As shown in table 6-4, in Q12 users were asked whether they were satisfied with the recommendations provided by WEBLORS. The weighted average score for this question is 4.47 which is between agreeing and strongly agreeing.

6.5.2.2 Qualitative Results. As mentioned in Section 6.4.1.2, four open-ended questions were included in the questionnaire (Q13-Q16), so users could explain their answers and provide additional feedback regarding their experience with WEBLORS. As these questions did not have pre-defined answers, each response was classified into topics. In order to classify the responses, each response was manually reviewed. Next, responses were grouped together based on the similarity of the topics which were discussed in the answers. An answer could include more than one topic. For example, a response such as "In my opinion, recommendations are according to the content and are really helpful for students" was classified under the topics of "Supports Learners" and "Relevance of Recommendations". Next, the number of times that each topic occurred in the answers was counted. Figure 6-5 shows the result of this qualitative analysis.

Figure 6-5

Results of qualitative analysis



In Q13, users were asked to list the aspects that they appreciate most in WEBLORS. As shown in the chart (a) of Figure 6-5, relevance of recommendations, providing supports to learners, personalization of recommendations, ease of use and good presentation of recommended materials are the aspects that were appreciated most by the participants.

In Q14, participants were asked to specify one thing that they would like to see changed in the system. Chart (b) of Figure 6-5 shows that the majority of participants (22 out of 36) specified that nothing needs to be changed in WEBLORS. In addition, a few users suggested to add more LOs and more graphics to the sample course. Moreover, 3 users mentioned that they got some irrelevant recommendations and 2 users believed that more instruction should be added to the evaluation instruction to help users to learn more regarding how to use the system.

In Q15, participants were requested to identify the gaps and list what they believe is missing in the recommendations provided by the system. As shown in the chart (c) of Figure 6-5, most users (31 out of 36) stated that in their opinion, nothing was missing in the recommendations provided by WEBLORS. In addition, 3 participants believed that more questions need to be added to the post-test or more assessments need to be added to the sample course. In addition, 2 users believed that explanation should be added to the recommendations to help users understand why a certain task was selected and recommended to the learners.

In Q16, users were asked to add any additional comments that they might have regarding their experience while using WEBLORS. As shown in the chart (d) of Figure 6-5, all participants (36 out of 36) were rather positive either stating that they think recommendations provided by WEBLORS help students in their learning process, expressing that they like to use WEBLORS in more courses or not having any additional comments.

6.6 Discussion

In this section the results of the evaluation of WEBLORS are discussed. In addition, the user study that was performed to evaluate WEBLORS is reviewed in the light of related literature and the differences between WEBLORS and similar systems in the literature which were evaluated using user studies are explained.

The results of the system-centric evaluation presented in section 6.5.1 show that 71.48% of the keywords that were automatically extracted by the system were accurate and best represented the content of the LO. In addition, as mentioned in section 6.3.1.3, teachers can use the teacher interface to review the list of keywords that were automatically extracted and if necessary, overwrite them with new keywords. This feature can lead to increase the accuracy of the keywords even further. In addition, regarding the recommendation accuracy, the result of the simulation shows that 80.26% of the generated recommendations generated by WEBLORS matched the learners' learning styles and were relevant to the topic of the course LO that was being visited by the simulated student at the time of recommendation generation. This simulation was performed on a fresh installation of WEBLORS, so there were no ratings in the system. As explained in section 6.3.2.6, ratings will give more chance to the high quality local LOs to be recommended to learners. Therefore, the consideration of ratings will increase the recommendation accuracy of WEBLORS even further.

Regarding user-centric evaluation, as shown in section 6.5.2, WEBLORS was evaluated in terms of its (1) recommender system acceptance, (2) ease of use and user friendliness, and (3) user satisfaction.

In terms of recommender system acceptance, the quantitative results show that most participants (32 out of 36) would like to use WEBLORS frequently and agreed that they would like to see such recommendation available in other online courses as well (per answers given to Q1 and Q2). Most participants (34 out of 36) also trusted the recommendations provided by WEBLORS and agreed that the recommendations provided by WEBLORS will be helpful in increasing students' learning (per answers given to Q3 and Q6). In addition, as per answers given to Q4, most users (35 out of 36) agreed that recommendations provided by WEBLORS will be helpful in increasing students' performance. Moreover, the majority of participants (26 out of 36) disagreed with the statement in Q5 and did not think that WEBLORS will put extra work on students for providing ratings. Qualitative results also reflect the positive feedback from the quantitative data regarding the recommender system acceptance. As shown in the chart (a) of Figure 6-5, relevance of recommendations, providing support to learners, and personalization of recommendations are the top three aspects that are appreciated by the participants. As shown in the chart (a) of Figure 6-5, 15 (out of 36) participants appreciated that the generated recommendations are personal for each user. 15 users also appreciated that provided recommendations are relevant to the subject of the LO that they visit. 10 users also appreciated that WEBLORS helps students in their learning process. Moreover, the majority of users (22 out of 36) stated that they do not believe anything needs to be changed in WEBLORS (as shown chart (b) of Figure 6-5). In addition, 17 participants (out of 36) provided additional feedback and stressed that they either believe that WEBLORS helps students in their learning process or stated that they like to use WEBLORS in more courses (as shown in chart (d) of Figure 6-5).

In terms of ease of use and user friendliness, the quantitative results show that most participants (29 out of 36) liked the interface of WEBLORS and believed that the information provided in the recommendations were clear and easy to understand (per answers given to Q7 and Q10). Most users (32 out of 36) also felt confident in using WEBLORS (per answers given to Q11). In addition, as per answers given to Q8, the majority of users (27 out of 36) did not find WEBLORS unnecessarily complex. In addition, the majority of users (24 out of 36) believed that they could understand the recommendations in a reasonable amount of time (as per answers given to Q9). Qualitative results regarding ease of use and user friendliness also show that the ease of use and the way that WEBLORS presents the recommendations are among the top five aspects that users appreciated most in WEBLORS (chart (a) of Figure 6-5). In addition, the majority of users (22 out of 36) believed that nothing needs to be changed in WEBLORS (chart (b) of Figure 6-5).

In terms of user satisfaction, quantitative results show that most users (33 out of 36) agreed that they were satisfied with the recommendations provided by WEBLORS (as per answers given to Q12). Qualitative results regarding user satisfaction also indicate that most users (31 out of 36) stated that they were satisfied with the current state of WEBLORS and did not feel like anything was missing in the recommendations provided by the system (chart (c) of Figure 6-5). In addition, as shown in the chart (d) of Figure 6-5, all participants (36 out of 36) either did not have any additional comments or concerns with WEBLORS, stated that the system is helpful for students or commented that they like to use WEBLORS in more courses.

As mentioned above, in addition to the results of the evaluation of WEBLORS, our evaluations are discussed in the light of related literature of similar evaluations and a comparison of WEBLORS' user-centric evaluation with the evaluations of similar systems (that were evaluated using user studies) is presented in this section. Based on the literature review that was presented in section 6.2, it was observed that different approaches have been used in the literature to evaluate educational recommender systems. As mentioned in section 6.4.1, these approaches can be grouped to two main groups. These groups are (1) system-centric evaluations, and (2) user-centric evaluations. System-centric evaluations can be further divided into three subgroups. There might be more system-centric approaches in the literature, however, the three subgroups were found based on similar recommender systems from our literature review in section 6.2.

1. Evaluations that use either a simulated dataset or a historical dataset (public or private) to evaluate the accuracy of the generated recommendations or compare the proposed recommendation algorithm with other approaches to see if the used approach outperforms the other approaches (Bourkoukou & Bachari, 2018; Dahdouh et al., 2019; Khribi et al., 2009).

2. Evaluations that measure the performance of the proposed systems by comparing a list of recommendations that are expected to be generated with the actual recommendations generated by the system (Sharif et al., 2015). In these evaluations, more matching between the expected recommendations and the generated recommendations reflects a better performance of the recommender system.

3. Evaluations that use a manual review process to determine the accuracy of the proposed system (Atkinson et al., 2014). In these evaluations, the generated recommendations by the system are reviewed and assessed by one or more domain experts. In these evaluations, more accurate recommendations identified by the expert(s) show a better performance of the recommender system.

User-centric evaluations typically consist of user studies using questionnaires to capture the users' feedback and measure the perceived usability and perceived usefulness of the systems for their end users (Rahman & Abdullah, 2018; Zapata et al., 2013).

WEBLORS was evaluated using the third approach of the system-centric evaluations (listed above). This approach was used to measure the keyword accuracy and recommendation accuracy of the system. In addition, WEBLORS was evaluated using a user-centric evaluation by performing a user study. Next, we focus on the user-centric evaluation of WEBLORS and compare its results with other user-centric evaluations of similar systems.

As mentioned above, in the literature review presented in section 6.2, only two other evaluations of similar recommender systems using user studies were found. These user studies were done by Zapata and colleagues (2013) and Rahman and Abdullah (2018). The following differences between these user studies and the user study performed to evaluate WEBLORS were found:

1. Zapata and colleagues' user study was performed using the SUS (Brooke, 1996) and the CSUQ (Lewis, 1995) questionnaires, therefore, participants were presented with two separate questionnaires. All questions from these questionnaires were used as one single category focusing on the usability of the system. Rahman and Abdullah performed their user study using the TAM questionnaire (Davis, 1989), where questions are categorized into two categories. These two categories are (1) Participants' perception of ease of use, and (2) Participants' perception of usefulness. WEBLORS uses one single questionnaire. The questionnaire was created based on the SUS questionnaire (Brooke, 1996; Lewis & Sauro, 2009), however, the questions were categorized into three categories with particular emphasis on recommender systems. These three categories are (1) Recommender System Acceptance, (2) Ease of Use and User Friendliness, and (3) User Satisfaction.
2. Both questionnaires used by Zapata and colleagues included a text field, so users could express their comments regarding the system. There was no additional text fields or questions added to the questionnaire used by Rahman and Abdullah. The questionnaire used in the evaluation of WEBLORS includes four open-ended questions to allow users to provide more feedback regarding the system.

3. Zapata and colleagues' user study was performed with 24 participants. 60 users participated in the user study that was performed by Rahman and Abdullah. WEBLORS was evaluated by 36 participants.

As mentioned above, the questionnaires that were used by Rahman and Abdullah, Zapata and colleagues, and in our evaluation are not identical. Thus, the overall results cannot be compared between these evaluations. However, one of the questionnaires that Zapata and colleagues used was the SUS questionnaire. Similarly, WEBLORS' questionnaire has been created based on the SUS questionnaire. Thus, WEBLORS contain some questions that are similar to some of the questions in the questionnaire used by Zapata and colleagues.

Table 6-6 shows the comparison between similar questions in WEBLORS' and Zapata and colleagues' evaluations.

Table 6-6

Comparison of the results of Zapata and colleagues' user study with WEBLORS' result

WEBLORS (Question)	(Zapata et al.) (Question)	WEBLORS(Average)	(Zapata et al.) (Average)
I would like to use WEBLORS frequently	I think that I would like to use this system frequently	4.22	4.45
I liked the interface of the WEBLORS	I like using the interface of system	4.14	2.48
I found WEBLORS unnecessarily complex	I found the system unnecessarily complex	2.08	2.37
The information provided in the recommendation was clear and easy to understand	The information provided for system is easy to understand	4.19	3.66
As a whole, I am satisfied with the recommendations provided by WEBLORS	Overall, I am satisfied with system	4.47	3.84

As shown in table 6-6, five similar questions from both evaluations are compared. Based on the average scores, WEBLORS outperforms Zapata and colleagues' system in four of five questions that focus on the system's interface, the complexity of the system, clarity of the provided information, and user satisfaction. The system created by Zapata and colleagues got a higher average score in one of the questions which focuses on whether users like to see the system frequently. WEBLORS was evaluated using 36 participants, and Zapata and colleagues' system was evaluated by 24 participants. In addition, the third question in table 6-6 is a negative question, so a lower score is better for this question.

As mentioned above, Rahman and Abdullah used the TAM questionnaire (Davis, 1989) which is different to the questionnaire that is used by WEBLORS, therefore the average scores of each question cannot be compared. However, the questions in the TAM questionnaire are categorized into two categories (participants' perception of ease of use, and participants' perception of usefulness). As mentioned above, the questions in the questionnaire that was used to evaluate WEBLORS were also categorized into three categories (recommender system acceptance, ease of use and user friendliness, and user satisfaction). Thus, there are two similar categories between these questionnaires that can be compared. In order to compare these categories, the average of the scores given to all questions in each category was calculated and reported in table 6-7. As shown in table 6-7, WEBLORS got a higher score in recommender system acceptance category. Regarding the ease of use and user friendliness, the scores are very close, however, Rahman and Abdullah's system got a slightly higher score (by 0.01 points).

Table 6-7

Comparison of the results of Rahman and Abdullah's user study with WEBLORS' result

WEBLORS Category	(Rahman and Abdullah) Category	WEBLORS (Average of average scores)	(Rahman and Abdullah) (Average of average scores)
Recommender System Acceptance	Participants' perception of usefulness	4.32	4.13
Ease of use and User Friendliness	Participants' perception of ease of use	4.09	4.10

6.7 Conclusion

This paper introduces WEBLORS which is an adaptive recommender system that can be integrated into LMSs to provide learners with additional learning materials from the web. The focus of this paper is on explaining the approach, algorithms and components of WEBLORS as well as the evaluation of the system based on its (1) keyword accuracy, (2) recommendation accuracy, (3) recommender system acceptance, (4) ease of use and user friendliness, and (5) user satisfaction. WEBLORS is a recommender system that considers the topic that the learner is learning and provides the learner with relevant learning materials that are beneficial for the learner based on his/her learning styles. Recommended materials are selected from a set of relevant LOs that are either discovered from the web for the first time or have been previously recommended to other learners and were given high ratings (or have been rated by less than 5 users) with the condition that at least one new LO from the web is recommended every time that WEBLORS generates recommendations.

The results of the system-centric evaluation show that a high percentage of the keywords automatically extracted from LOs by WEBLORS were accurate and best represented their respective LOs. The results also show that the recommendations generated by WEBLORS were highly personalized based on learners' learning styles and the topic that the learner was studying at the time of recommendation generation.

Regarding the user-centric evaluation, the results from the quantitative and qualitative data both show that participants provided positive feedback about recommender system acceptance. Most users trusted the recommendations provided by WEBLORS (Q3), would like to use WEBLORS frequently (Q1), like to use such recommendations in other courses (Q2), and believed that the recommendations that were provided by WEBLORS can help students in their learning process (Q6) and will have a positive impact on students' performance (Q4). The majority of users also believed that asking users to rate the recommendations does not add lots of overhead and does not put much extra work on students (Q5). Several users also pointed out that they appreciated that the generated recommendations are relevant to the topic of the course LO that they were visiting, are personal for each user, and are helpful for the students (chart (a) of Figure 6-5). The majority of users also believed that nothing needs to be changed in WEBLORS (chart (b) of Figure 6-5). In addition, several participants provided additional feedback and stated that they believe WEBLORS helps students in their learning process or expressed that they like to use WEBLORS in more courses (chart (d) of Figure 6-5).

In addition, quantitative and qualitative results both indicate that users provided positive feedback regarding the ease of use and user friendliness of WEBLORS. Most users felt confident in using WEBLORS (Q11), liked the interface of the system (Q7), and believed that the information provided in the recommendations are clear and easy to understand (Q10). In addition, the majority of participants did not find WEBLORS unnecessarily complex (Q8) and believed that it did not take them a long time before they could understand the recommendations (Q9). The majority of users also stated that they were happy with the system and do not see any change required to be made in WEBLORS (chart (b) of Figure 6-5). In addition, the ease of use and the way that WEBLORS presents the recommendations are appreciated by several users (chart (a) of Figure 6-5).

Moreover, quantitative and qualitative results both show that users provided positive feedback regarding user satisfaction. Most participants stated that they were satisfied with the recommendations provided by WEBLORS (Q12) and did not believe that anything was missing that they were expecting to see in the recommendations (chart (c) of Figure 6-5). In addition, the users either did not have any additional comments or concerns with WEBLORS, stated that the system is helpful for students or mentioned that they like to use WEBLORS in more courses (chart (d) of Figure 6-5).

To conclude, the system-centric and user-centric evaluations show positive results about WEBLORS in terms of its (1) keyword accuracy, (2) recommendation accuracy, (3) recommender system acceptance, (4) ease of use and user friendliness, and (5) user satisfaction. The results of the system-centric and the collected feedback from users show that WEBLORS potentially fills a gap in LMSs by recommending extra personalized recommendations from the web and helping with information overload by only recommending LOs relevant to the topic that is being studied and which fits students' profiles.

Future work will deal with trying different text mining algorithms and APIs such as Google Natural Processing Language API and compare the result with RAKE algorithm to determine if replacing RAKE with a different API can increase the accuracy of the keyword extraction process. In addition, future work will deal with integrating WEBLORS with different search engines such as Yahoo or Bing and measuring the accuracy of the recommendations. Moreover, future work will deal with the broad use of the system in different courses and evaluate it in real-life settings for a longer period of time. The results captured during that evaluation can be used to verify the findings of the user study and also can prove that the system is truly beneficial to learners.

References

Adomavicius, G., & Kwon, Y., 2015. Multi-criteria recommender systems, in F. Ricci, L. Rokach, & B. Shapira (Eds.), Recommender systems handbook. Springer, Boston, MA, pp. 847-880.

Afridi, A. H., 2018. Visualizing serendipitous recommendations in user controlled recommender system for learning. Procedia Computer Science. 141, 496-502.

Akhtarzada, A., Calude, C. S., & Hosking, J., 2011. A multi-criteria metric algorithm for recommender systems. Fundamenta Informaticae. 110, 1-11.

Al Abri, M., & Dabbagh, N., 2018. Open educational resources: A literature review. Journal of Mason Graduate Research. 6(1), 83-104.

Amazon. (2019). Amazon Mechanical Turk. Retrieved from https://www.mturk.com

Atkinson, J., Gonzalez, A., Munoz, M., & Astudillo, H., 2014. Web metadata extraction and semantic indexing for learning objects extraction. Applied Intelligence. 41(2), 649-664.

Bäck, T. (1996). *Evolutionary algorithms in theory and practice: Evolution strategies, evolutionary programming, genetic algorithms*. New York, NY: Oxford University Press.

Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A., 2013. Recommender systems survey. Knowledge-Based Systems. 46, 109-132.

Bourkoukou, O., & Bachari, E. E., 2018. Toward a hybrid recommender system for e-learning personnalization based on data mining techniques. International Journal On Informatics Visualization. 2(4), 271-278.

Bradáč, V., Šimík, M., Kotyrba, M., & Volná, E., 2017. Personalisation of a moodle course from student's perspective. In *Proceedings of the* European Conference on e-Learning, Retrieved from http://0-search.ebscohost.com.aupac.lib.athabascau.ca/login.aspx?direct=true&db=ehh&AN=126548003&site=eds-live, 70-77.

Brooke, J., 1996. SUS: A quick and dirty usability scale, in P. W. Jordan, B. Thomas, I. L. McClelland, & B. Weerdmeester (Eds.), Usability Evaluation in Industry. Taylor and Francis, London, pp. 189-194.

Burke, R., 2007. Hybrid web recommender systems, in P. Brusilovsky, A. Kobsa, & W. Nejdl (Eds.), The Adaptive Web. Springer, Berlin, Heidelberg, pp. 377-408.

Burke, R., & Ramezani, M., 2011. Matching recommendation technologies and domains Recommender systems handbook. Springer, Boston, MA, pp. 367-386.

Cremonesi, P., Garzotto, F., & Turrin, R., 2013. User-centric vs. System-centric evaluation of recommender systems. In *Proceedings of the* Human-Computer Interaction, Springer, Berlin, Heidelberg, 334-351.

Cronbach, L. J., & Meehl, P. E., 1955. Construct validity in psychological tests. Psychological Bulletin. 52(4), 281-302.

Dahdouh, K., Oughdir, L., Dakkak, A., & Ibriz, A., 2019. Building an e-learning recommender system using association rules techniques and R environment. International Journal of Information Science & Technology. 3(2), 11-18.

Davis, F. D., 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS quarterly, 319-340.

DorÇA, F. A., AraÚJo, R. D., de Carvalho, V. C., Resende, D. T., & Cattelan, R. G., 2016. An automatic and dynamic approach for personalized recommendation of learning objects considering students learning styles: An experimental analysis. Informatics in Education. 15(1), 45-62.

Dougiamas, M. (2019). Moodle. *http://moodle.org/.* Retrieved from http://moodle.org/

Drachsler, H., Hummel, G. K. H., & Koper, R., 2008. Personal recommender systems for learners in lifelong learning networks: The requirements, techniques and model. International Journal of Learning Technology. 3(4), 404-423.

Dwivedi, P., & Bharadwaj, K., 2015. e-Learning recommender system for a group of learners based on the unified learner profile approach. Expert Systems. 32(2), 264-276.

El-Bishouty, M., Saito, K., Chang, T.-W., Kinshuk, & Graf, S., 2013. An interactive course analyzer for improving learning styles support level. In *Proceedings of the* Human-Computer Interaction & Knowledge Discovery in Complex, Unstructured, Big Data, Springer, Berlin, Heidelberg, 136-147.

Erdt, M., Fernández, A., & Rensing, C., 2015. Evaluating recommender systems for technology enhanced learning: A quantitative survey. IEEE Transactions on Learning Technologies. 8(4), 326-344.

Fazazi, H., Qbadou, M., Salhi, I., & Mansouri, K., 2018. Personalized recommender system for e-Learning environment based on student's preferences. INTERNATIONAL JOURNAL OF COMPUTER SCIENCE AND NETWORK SECURITY. 18(10), 173-178.

Fazeli, S., Drachsler, H., Bitter-Rijpkema, M., Brouns, F., Van der Vegt, W., & Sloep, P. B., 2017. User-centric evaluation of recommender systems in social learning platforms: accuracy is just the tip of the iceberg. IEEE Transactions on Learning Technologies. 11(3), 294-306.

Felder, R., & Silverman, L., 1988. Learning and teaching styles in engineering education. Engineering Education. 78(7), 674-681.

Felder, R., & Soloman, B. (1997). Index of learning style questionnaire. Retrieved from https://www.webtools.ncsu.edu/learningstyles/

Felder, R., & Spurlin, J., 2005. Applications, reliability and validity of the index of learning styles. International Journal of Engineering Education. 21, 103-112.

Fraihat, S., & Shambour, Q., 2015. A framework of semantic recommender system for e-Learning. Journal of Software. 10(3), 317-330.

Goldberg, D., Nichols, D., Oki, B. M., & Terry, D., 1992. Using collaborative filtering to weave an information tapestry. Communications of the ACM. 35(12), 61-70.

Google. (2019). Creating a Custom Search Engine. Retrieved from https://developers.google.com/custom-search/docs/tutorial/creatingcse

Graf, S., Kinshuk, & Ives, C., 2010. A flexible mechanism for providing adaptivity based on learning styles in learning management systems. In *Proceedings of the* IEEE International Conference on Advanced Learning Technologies, IEEE, Sousse, Tunisia, 30-34.

Heng, L. E., Sangodiah, A., Muniandy, M., & Yuen, P. K., 2018. Integration of learner's model into learning management system environment. Journal of Fundamental & Applied Sciences. 10(1), 1771-1778.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T., 2004. Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems. 22(1), 5-53.

IEEE Learning Technology Standardization Committee. (2007). The Learning Object Metadata standard. Retrieved from https://www.ieeeltsc.org/working-groups/wg12LOM/lomDescription/

Imran, H., Belghis-Zadeh, M., Chang, T.-W., Kinshuk, & Graf, S., 2014. A rule-based recommender system to suggest learning tasks. In *Proceedings of the* International Conference on Intelligent Tutoring Systems, Springer, Honolulu, 672-673.

Imran, H., Belghis-Zadeh, M., Chang, T.-W., Kinshuk, & Graf, S., 2016. PLORS: a personalized learning object recommender system. Vietnam Journal of Computer Science. 3(1), 3-13.

Khribi, M., Jemni, M., & Nasraoui, O., 2009. Automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval. Journal of Educational Technology & Society. 12(4), 30-42.

Klašnja-Milićević, A., Vesin, B., & Ivanović, M., 2018. Social tagging strategy for enhancing e-learning experience. Computers & Education. 118, 166-181.

Knijnenburg, B. P., 2012. Conducting user experiments in recommender systems. In *Proceedings of the* sixth ACM conference on Recommender systems, ACM, 3-4.

Krauss, C., Merceron, A., & Arbanowski, S., 2019. The timeliness deviation: A novel approach to evaluate educational recommender systems for closed-courses. In *Proceedings of the* the 9th International Conference on Learning Analytics & Knowledge, 195-204.

Lewis, J. R., 1995. IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. International Journal of Human-Computer Interaction. 7(1), 57-78.

Lewis, J. R., & Sauro, J., 2009. The factor structure of the system usability scale. In *Proceedings of the* International Conference on Human Centered Design, Springer, Berlin, Heidelberg, 94-103.

Li, K., & Yan, E., 2019. Are NIH-funded publications fulfilling the proposed research? An examination of concept-matchedness between NIH research grants and their supported publications. Journal of Informetrics. 13(1), 226-237.

Manouselis, N. (2013). *Recommender systems for learning*. New York: Springer.

Nafea, S. M., Siewe, F., & He, Y., 2019. On recommendation of learning objects using felder-silverman learning style model. IEEE Access. 7, 163034-163048.

Nunnally, J. C. (1994). *Psychometric theory 3E*: Tata McGraw-Hill Education.

Obeid, C., Lahoud, I., El Khoury, H., & Champin, P.-A., 2018. Ontology-based recommender system in higher education. In *Proceedings of the* Companion Proceedings of the The Web Conference 2018, 1031-1034.

Prakash, J. N., & Kumar, B. T. S., 2009. Precision and relative recall of search engines: A comparative study of google and yahoo. Singapore Journal of Library and Information Management. 38(1), 124-137.

Rahman, M. M., & Abdullah, N. A., 2018. A personalized group-based recommendation approach for web search in e-learning. IEEE Access. 6, 34166-34178.

Rhode, J., Richter, S., Gowen, P., Miller, T., & Wills, C., 2017. Understanding faculty use of the learning management system. Online Learning. 21(3), 68-86.

Rodríguez, P. A., Tabares, V., Mendez, N. D. D., Carranza, D. A. O., & Vicari, R. M., 2013. BROA: An agent-based model to recommend relevant learning objects from repository federations adapted to learner profile. International Journal of Interactive Multimedia and Artificial Intelligence. 2(1), 6-11.

Rose, S., Engel, D., Cramer, N., & Cowley, W., 2010. Automatic keyword extraction from individual documents Text Mining. John Wiley & Sons Ltd, pp. 1-20.

Sabourin, J., Kosturko, L., & McQuiggan, S., 2015. Where to next? A comparison of recommendation strategies for navigating a learning object repository. In *Proceedings of the* International Conference on User Modeling, Adaptation, and Personalization, Springer, Cham, 208-215.

Santos, J. R. A., 1999. Cronbach's alpha: A tool for assessing the reliability of scales. Journal of extension. 37(2), 1-5.

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J., 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the* 10th International Conference on World Wide Web, ACM, New York, NY, 285-295.

Shani, G., & Gunawardana, A., 2011. Evaluating recommendation systems, in F. Ricci, L. Rokach, B. Shapira, & P. B. Kantor (Eds.), Recommender systems handbook. Springer US, Boston, MA, pp. 257-297.

Sharif, N., Afzal, M. T., & Muhammad, A., 2015. Semantic based e-learning recommender system. European Scientific Journal. 1, 369-377.

Tarus, J. K., Niu, Z., & Kalui, D., 2018. A hybrid recommender system for e-learning based on context awareness and sequential pattern mining. Soft Computing. 22(8), 2449-2461.

Tarus, J. K., Niu, Z., & Mustafa, G., 2018. Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning. Artificial Intelligence Review. 50(1), 21-48.

Tavakol, M., & Dennick, R., 2011. Making sense of Cronbach's alpha. International journal of medical education. 2, 53-55.

Van Meteren, R., & Van Someren, M., 2000. Using content-based filtering for recommendation. In *Proceedings of the* ECML 2000 Workshop: Maching Learning in Information Age, 47-56.

Wang, T. I., Kun Hua, T., Ming Che, L., & Ti Kai, C., 2007. Personalized learning objects recommendation based on the semantic-aware discovery and the learner preference pattern. Journal of Educational Technology & Society. 10(3), 84-105.

Zaiane, O., 2002. Building a recommender agent for e-learning systems. In *Proceedings of the* International Conference on Computers in Education, IEEE, Auckland, New Zealand, 55-59.

Zapata, A., Menéndez, V. H., Prieto, M. E., & Romero, C., 2013. A framework for recommendation in learning object repositories: An example of application in civil engineering. Advances in Engineering Software. 56, 1-14.

Chapter 7. Discussion

This chapter discusses the approach and the results of the evaluation of RUBARS, PLORS, and WEBLORS (presented in chapters 3 to 6) in the light of related literature and explains the differences between these systems and similar systems in the literature.

Based on the literature review that was conducted in chapters 3 to 6, it was observed that different approaches have been used in the literature to evaluate educational recommender systems. Based on the focus of the evaluation, these approaches can be grouped into two common groups namely system-centric and user-centric evaluations (Cremonesi, Garzotto, & Turrin, 2013). These groups are further explained below:

1. System-centric evaluations: These evaluations focus on evaluating the accuracy of the generated recommendations and do not measure the perceived usefulness of the recommendations for the end users.
2. User-centric evaluations: These evaluations focus on evaluating the usability and perceived usefulness of the generated recommendations for the end users.

System-centric evaluations can be further divided into three subgroups. There might be more system-centric approaches in the literature, however, the three subgroups were found based on similar recommender systems from our literature reviews in chapters 4 to 6.

1a) Evaluations that use either a simulated dataset or a historical dataset (public or private) to evaluate the accuracy of the generated recommendations or compare the proposed recommendation algorithm with other approaches to see if the used approach outperforms the other approaches (Dahdouh, Oughdir, Dakkak, & Ibriz, 2019; Dwivedi & Bharadwaj, 2015; Khribi et al., 2009; Salehi, Pourzaferani, & Razavi, 2013; U, Chai, & Chen, 2018).

1b) Evaluations that measure the performance of the proposed systems by comparing a list of recommendations that are expected to be generated with the actual recommendations generated by the system (Sharif, Afzal, & Muhammad, 2015). In these evaluations, more matching between the expected recommendations and the generated recommendations reflects a better performance of the recommender system.

1c) Evaluations that use a manual review process to determine the accuracy of the proposed system (Atkinson, Gonzalez, Munoz, & Astudillo, 2014). In these evaluations, the generated recommendations by the system are reviewed and assessed by one or more domain experts. In these evaluations, more accurate recommendations identified by the expert(s) show a better performance of the recommender system.

User-centric evaluations, typically perform user studies using questionnaires to capture the users' feedback and measure the usability and perceived usefulness of the systems for their end users (Rahman & Abdullah, 2018; Zaiane, 2002; Zapata, Menéndez, Prieto, & Romero, 2013).

RUBARS, PLORS, and WEBLORS were evaluated using user-centric evaluations through user studies. For evaluation of each system, a new task was created and posted on Amazon Mechanical Turk and different groups of participants accepted each task and participated in each evaluation. These user studies intended to have a broad range of target audiences to participate in the evaluations as these systems were designed to be used for any online course and by anyone who is attempting to learn a concept online, thus it was important to recruit participants with different occupations and backgrounds to participate in the evaluations. Therefore, Amazon Mechanical Turk was used to recruit participants.

In addition to a user study, WEBLORS uses two simulations followed by manual review processes to measure the keyword generation and recommendation accuracy of the system as well. These additional evaluations used in WEBLORS fall under system-centric evaluation (1c).

Before discussing the evaluation results, the comparison between each of these three systems (RUBARS, PLORS, and WEBLORS) and other similar systems in the literature are discussed below.

Regarding RUBARS, as mentioned in chapter 3, in learner-centered learning, there might be some assignments that contain many learning tasks and students may choose which tasks they prefer to complete in order to satisfy the requirements of the assignment. RUBARS can support learners in such a situation by helping learners to choose the tasks that are best suited to them among the many tasks available in assignments (Imran et al., 2014a). As stated in chapter 3, RUBARS is different compared to other recommender systems in literature in terms of the type of objects that are recommended to learners. To the best of our knowledge, RUBARS is the only recommender system that recommends learning tasks within assignments. In addition, RUBARS focuses on providing highly personalized recommendations and therefore uses four different user attributes (i.e., learners' learning styles, prior knowledge, expertise level, and performance on previously completed tasks) to build each user's neighborhoods, which is more than most other systems. In addition, RUBARS generates the users' neighborhood in real-time which allows for more accurate and up-to-date recommendations. The results of the user study that was performed to evaluate RUBARS (presented in chapter 3) show that 51 participants who participated in this evaluation provided promising feedback about RUBARS and believed that RUBARS potentially solves an issue in learner-centered education by helping learners to choose the tasks that are best suited to them. As mentioned above, to the best of our knowledge there is no other system that recommends learning tasks within assignments. Therefore, it is not possible to compare the result of the evaluation of RUBARS to other similar systems.

Both PLORS and WEBLORS focus on learning object recommendation. In addition, both systems were evaluated using user studies (WEBLORS was evaluated using additional system-centric evaluations as well). Thus, we first explain the differences between these systems with other similar systems in the literature in the next two paragraphs and then discuss the evaluations of both systems with respect to other similar evaluations done in the literature.

Regarding PLORS, as presented in chapter 4, teachers usually create learning materials in the courses in LMSs. The types of learning objects as well as the sequence of these objects within the course are decided by the teacher based on his/her teaching strategy (Abu Rasheed, Weber, Harrison, Zenkert, & Fathi, 2018; Bradáč et al., 2017). However, to recall from chapter 4, students with different profiles benefit more from different types of learning objects (El-Bishouty, Saito, Chang, Kinshuk, & Graf, 2013). Learners also have different navigational behavior within the course. Therefore, they might not visit the learning materials in the same sequence as was set by the teacher (El-Bishouty et al., 2019; Graf & Kinshuk, 2008; Graf, Liu, & Kinshuk, 2010; Karagiannis & Satratzemi, 2018). Thus, learners might miss or intentionally skip some learning objects that would be beneficial to them. PLORS supports learners in such a situation by recommending a set of learning objects to each learner in a personalized sequence. As mentioned in section 4, PLORS is different compared to other recommender systems in literature as PLORS groups similar users together based on their profiles rather than their past activities. In addition, PLORS uses associations among the learning objects visited by the learner and other similar learners to determine the usefulness of learning objects for the active learner instead of using ratings or tags provided by other learners.

Regarding WEBLORS, as presented in chapters 5 and 6, there are many learning objects available on the web that can be targeted and recommended by recommender systems (Al Abri & Dabbagh, 2018). Accordingly, learning materials from the web could be utilized to provide learners with additional content that is personalized to their needs and preferences. However, there are an immense number of learning materials available on the web. Thus, different approaches need to be utilized to overcome the overload of the information and find relevant and personalized learning materials that meet the needs and preferences of learners (Akhtarzada, Calude, & Hosking, 2011). As mentioned in chapter 5 and 6, WEBLORS is an adaptive web-based recommender system that helps learners to overcome the information overload by recommending personalized learning objects from the web that are relevant to the topic that the learner is studying at the time of recommendation generation and fit students' learning styles as well. To recall from chapters 5 and 6, WEBLORS is different compared to other recommender systems in the literature. First, WEBLORS recommends learning objects from the public web rather than learning object repositories. Second, WEBLORS addresses the cold start problem by (1) using learners' learning styles, the opinions of other learners (if available) and the topic that is being studied instead of learner's past activities, and (2) automatically extracting the keywords for the learning objects that are added to the course. Third, despite many recommender systems that rely on the search criteria that are passed by the users, in WEBLORS, keywords are automatically created through extracting them from the content that a learner is learning.

PLORS and WEBLORS are both learning object recommender systems, however, these two systems have a different focus and different aims. Despite the differences between the two systems, one of the recommender systems that was found in the literature that is built by Zapata and colleagues (2013) has similarities to both systems. Thus, it was included in the literature review that was performed for both systems (chapters 4 to 6). Since the approach that was taken to evaluate the system created by Zapata and colleagues (2013) is similar to the approach that was used to evaluate PLORS and WEBLORS, and to avoid repetition of the same information, the discussion of the evaluations of PLORS and WEBLORS are grouped together and explained below.

Regarding the evaluations of PLORS and WEBLORS, as mentioned above, both systems were evaluated using user studies (with WEBLORS using system-centric evaluations in addition). In the literature review that was presented in chapters 4 to 6, only three other evaluations using user studies were found (Rahman & Abdullah, 2018; Zaiane, 2002; Zapata et al., 2013). Out of these three works, Zaiane did not publish the result of his user study in the paper that was reviewed (Zaiane, 2002). It is stated in the paper that the system was still being evaluated using questionnaires when the paper was published. The other two user studies were done by Zapata and colleagues (2013) and Rahman and Abdullah (2018). The following differences between these user studies and the user studies performed to evaluate PLORS and WEBLORS were found:

1. Rahman and Abdullah performed their user study using the TAM questionnaire (Davis, 1989) , where questions are categorized into two categories. These two categories are (1) Participants' perception of ease of use, and (2) Participants' perception of usefulness. Zapata and colleagues' user study was performed using SUS (Brooke, 1996) and CSUQ (Lewis, 1995) questionnaires. In this study, participants were presented with two separate questionnaires. All questions from these questionnaires were used as one single category focusing on the usability of the system. PLORS and WEBLORS each use one single questionnaire. The questionnaire was created based on SUS questionnaire (Brooke, 1996; Lewis &

Sauro, 2009), however, the questions were categorized into three categories with particular emphasis on recommender systems. These three categories are (1) Recommender System Acceptance, (2) Ease of Use and User Friendliness, and (3) User Satisfaction.

2. There was no additional text fields or questions added to the questionnaire used by Rahman and Abdullah. Both questionnaires used by Zapata and colleagues included a text field, so users could express their comments regarding the system. The questionnaire used in the evaluation of PLORS and WEBLORS each include 4 open-ended questions to allow users to provide more feedback regarding the systems.
3. Rahman and Abdullah's user study was performed with 60 users. 24 participants participated in the user study that was performed by Zapata and colleagues. PLORS was evaluated by 50 participants, and 36 participants participated in WEBLORS' evaluation.

As mentioned above, the questionnaires that were used by Rahman and Abdullah, Zapata and colleagues, and in our evaluation are not identical. Thus, the overall results cannot be compared between these evaluations. However, one of the questionnaires that Zapata and colleagues used was the SUS questionnaire. Similarly, PLORS and WEBLORS' questionnaires have been created based on the SUS questionnaire. Thus, both PLORS and WEBLORS contain some questions that are similar to some of the questions in the questionnaire used by Zapata and colleagues.

As mentioned above, the system built by Zapata and colleagues (2013) was reviewed in the literature review that was performed for both PLORS and WEBLORS (chapters 4 to 6). Thus, the results of the evaluation of this system are compared with the results from the evaluation of both PLORS and WEBLORS below.

Table 7-1 shows the comparison between similar questions in PLORS' and Zapata and colleagues' evaluation.

Table 7-1

Comparison of the result of Zapata and colleagues' user study with PLORS' result

PLORS (Question)	Zapata (Question)	PLORS(Average)	Zapata (Average)
I would like to use PLORS frequently	I think that I would like to use this system frequently	3.76	4.45
I liked the interface of the PLORS	I like using the interface of the system	3.34	2.48
I found PLORS unnecessarily complex	I found the system unnecessarily complex	2.78	2.37
The information provided in the recommendation was clear and easy to understand	The information provided for system is easy to understand	3.90	3.66
As a whole, I am satisfied with the recommendations provided by PLORS	Overall, I am satisfied with system	3.94	3.84

As shown in table 7-1, five similar questions from Zapata and colleagues' evaluation and PLORS's evaluation are compared. One thing to note in this comparison is that PLORS was evaluated using 50 participants, but the other system was evaluated by 24 participants. Moreover, the third question in the table is a negative question, so a lower score is better for this question. Based on the average scores shown in table 7-1, Zapata and colleagues' system got better scores on the first and third questions. The first question asks users if they like to use the system frequently and the third question asks whether they found the system unnecessarily complex. In the other three questions PLORS outperforms the other system. These three questions focus on systems' interface, clarity of the provided information, and user satisfaction.

Similar comparison was performed between similar questions in WEBLORS and Zapata and colleagues' evaluation. This comparison is shown in table 7-2.

Table 7-2

Comparison of the result of Zapata and colleagues' user study with WEBLORS' result

WEBLORS (Question)	Zapata (Question)	WEBLORS (Average)	Zapata (Average)
I would like to use WEBLORS frequently	I think that I would like to use this system frequently	4.22	4.45
I liked the interface of the WEBLORS	I like using the interface of system	4.14	2.48
I found WEBLORS unnecessarily complex	I found the system unnecessarily complex	2.08	2.37
The information provided in the recommendation was clear and easy to understand	The information provided for system is easy to understand	4.19	3.66
As a whole, I am satisfied with the recommendations provided by WEBLORS	Overall, I am satisfied with system	4.47	3.84

As shown in table 7-2, five similar questions from Zapata and colleagues' evaluation and WEBLORS's evaluation are compared. Based on the average scores in each system, WEBLORS outperforms the Zapata and colleagues' system in four of five questions that focus on systems' interface, the complexity of the system, clarity of the provided information, and user satisfaction. The system created by Zapata and colleagues got a higher average score in one of the questions. One thing to note in this comparison is that WEBLORS was evaluated using 36 participants and the other system was evaluated by 24 participants. In addition, similar to table 7-1, the third question in table 7-2 is a negative question as well, so a lower score is better for this question.

The system that was built by Rahman and Abdullah (2018) is only similar to WEBLORS. Therefore, it was only included in the literature review that was performed for WEBLORS (chapters 5 and 6). Thus, the result of the evaluation of this system is only compared with the results of the evaluation of WEBLORS. As mentioned above, Rahman and Abdullah used the TAM questionnaire (Davis, 1989) which is different to the questionnaire that is used by WEBLORS, therefore the average scores of each question cannot be compared. However, Rahman and Abdullah categorize their questions into two groups (participants' perception of ease of use, and participants' perception of usefulness). The questions in the questionnaire that was used to evaluate WEBLORS also were categorized into three groups (recommender system acceptance, ease of use and user friendliness, and user satisfaction). Thus, there are two similar groups between these questionnaires that can be compared. As these two systems are similar, it could be valuable to compare them with respect to their similar groups. In order to compare these groups, the average of the scores given to all questions in each category was calculated and reported in table 7-3. As shown in table 7-3, WEBLORS got higher score in recommender system acceptance category. Regarding the ease of use and user friendliness, the scores are very close, however, Rahman and Abdullah's system got a slight higher score (by 0.01 points).

Table 7-3

Comparison of the result of Rahman and Abdullah's user study with WEBLORS' result

WEBLORS Category	**Rahman and Abdullah Category**	**WEBLORS (Average of average scores)**	**Rahman and Abdullah (Average of average scores)**
Recommender System Acceptance	Participants' perception of usefulness	4.32	4.13
Ease of use and User Friendliness	Participants' perception of ease of use	4.09	4.10

As stated above, in addition to a user study, WEBLORS was evaluated using a system-centric evaluation as well. In order to perform the system-centric evaluation, first, two simulations were performed, and the results of these simulations were reviewed by the author of this thesis as a domain expert to measure the accuracy of the keyword extraction and recommendation generation of WEBLORS. As presented in chapter 6, based on the results of the simulations, 71.48% of the keywords that were automatically extracted by the system were accurate, relevant to the topic, and best represented the content of their respective learning object. Also, 80.26% of the generated recommendations generated by WEBLORS matched the learners' learning styles and were relevant to the topic that the simulated learner was visiting at the time of recommendation generation.

Chapter 8. Conclusion

This chapter provides a summary and discusses the contributions of the work conducted within this thesis and concludes the thesis with a discussion on future work.

8.1 Summary and Contributions

In this thesis, designing and evaluating of educational recommender systems was investigated. The first objective of this thesis was to evaluate two previously built recommender systems, namely RUBARS and PLORS, using a user study to capture users' feedback regarding these two systems based on their (1) recommender system acceptance, (2) ease of use and user friendliness, and (3) user satisfaction.

The second objective of this thesis was to design, develop and evaluate an adaptive web-based recommender system to help learners overcome the information overload and provide them with personalized learning objects from the web while studying a course in a LMS. This goal was achieved by introducing WEBLORS which is an adaptive recommender system that can be integrated into LMSs to provide learners with additional personalized learning materials from the web. Also, WEBLORS was evaluated based on its (1) keyword accuracy, (2) recommendation accuracy, (3) recommender system acceptance, (4) ease of use and user friendliness, and (5) user satisfaction.

In order to evaluate RUBARS, PLORS and WEBLORS, three variations of a new questionnaire were used that were created by Dr. Imran based on the SUS questionnaire (Brooke, 1996; Lewis & Sauro, 2009). In this thesis, the reliability of these questionnaires were investigated and based on the results, these questionnaires proved to be reliable and can be used by other researchers in the future to evaluate other recommender systems.

In addition, the result of the evaluation of RUBARS shows that the participants who participated in the evaluation of the system believe that RUBARS potentially fills a gap in learner-centered education and can help learners, who do not have enough knowledge to select the appropriate tasks from a pool of learning tasks for their assignments, to choose the best tasks that are suitable for them based on their profiles. Moreover, as mentioned before, to the best of our knowledge and after reviewing the existing literature, no other recommender system was found that recommends learning tasks within assignments. This means that RUBARS adds value to the LMS community and contributes to the enhancements of LMSs in a novel way.

Furthermore, the result of the evaluation of PLORS shows promising results about the possibility of utilizing a recommender system to recommend learning objects in a course to learners in a personalized sequence. Based on the feedback provided by participants, PLORS has the potential to support learners to improve their learning by helping them to visit the learning objects in the course in a sequence that best fits their profiles. Also, in situations that learners do not follow the learning objects in the default sequence, PLORS helps them to visit the useful learning objects in the course that have been visited by similar learners but might have been missed or ignored by the learner.

Moreover, the results of the system-centric evaluation of WEBLORS shows that by using RAKE algorithm, Google Custom Search API, and the recommendation algorithm introduced in WEBLORS, the generated recommendations from the web are highly personalized based on learners' learning styles and are relevant to the topic that is being studied by the learner (80.26%). This proves that web mining techniques can be utilized to discover relevant learning objects on the web.

Lastly, the results of system-centric and user-centric evaluations of WEBLORS prove that recommender systems can be designed and used to deliver the matching learning objects discovered from the web to students based on their profiles, learning styles and the ratings given by other learners (if available). Based on the feedback provided by participants who participated in the user study, WEBLORS potentially fills a gap in learning management systems by recommending extra personalized recommendations from the web and helping with information overload by only recommending learning objects relevant to the topic that is being studied and which fits students' profiles.

8.2 Future Work

Three main directions to future investigations can be considered:

- *Using RUBARS, PLORS, and WEBLORS in different courses*. One of the directions for future investigation would be to use the three systems introduced in the above-mentioned manuscripts in different courses and evaluate them in real-life settings for a longer period of time. The results captured during that evaluation can be used to verify the findings of the user studies and also can prove that the systems are truly beneficial to learners.
- *Evaluating different types of text mining algorithms in WEBLORS*. Another direction for future work would be to try different text mining algorithms and APIs such as Google Natural Processing Language API and compare the result with RAKE algorithm and determine if any of them can outperform the RAKE algorithm in terms of accuracy of the extracted keywords. If a more accurate API is found, RAKE algorithm can be simply replaced as WEBLORS is highly modular and each module can be simply replaced.
- *Evaluating different search engines in WEBLORS*. Another possible line of investigation is to integrate WEBLORS with different search engines such as Yahoo or Bing and

measure the accuracy of the recommendations. As mentioned above, WEBLORS is highly modular, so if another search engine outperforms the accuracy of Google Custom Search API, this API can be simply used instead.

References

Abu Rasheed, H., Weber, C., Harrison, S., Zenkert, J., & Fathi, M., 2018. What to learn next: Incorporating student, teacher and domain preferences for a comparative educational recommender system. In *Proceedings of the* 10th International Conference on Education and New Learning Technologies, IATED, Palma, Spain, 6790-6800.

Akhtarzada, A., Calude, C. S., & Hosking, J., 2011. A multi-criteria metric algorithm for recommender systems. Fundamenta Informaticae. 110, 1-11.

Al Abri, M., & Dabbagh, N., 2018. Open educational resources: A literature review. Journal of Mason Graduate Research. 6(1), 83-104.

Arens-Volland, A., Gratz, P., Baudet, A., Deladiennée, L., Gallais, M., & Naudet, Y., 2019. Personalized recommender system for improving gender-fairness in teaching. In *Proceedings of the* 14th International Workshop on Semantic and Social Media Adaptation and Personalization, IEEE, Larnaca, Cyprus, Cyprus, 1-5.

Atkinson, J., Gonzalez, A., Munoz, M., & Astudillo, H., 2014. Web metadata extraction and semantic indexing for learning objects extraction. Applied Intelligence. 41(2), 649-664.

Belghis-Zadeh, M., Imran, H., Chang, M., & Graf, S., 2019. WEBLORS–a personalized web-based recommender system. In *Proceedings of the* International Conference on Web-Based Learning, Springer, 258-266.

Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A., 2013. Recommender systems survey. Knowledge-Based Systems. 46, 109-132.

Bradáč, V., Šimík, M., Kotyrba, M., & Volná, E., 2017. Personalisation of a moodle course from student's perspective. In *Proceedings of the* European Conference on e-Learning, Retrieved from http://0-

search.ebscohost.com.aupac.lib.athabascau.ca/login.aspx?direct=true&db=ehh&AN=126548003&site=eds-live, 70-77.

Brooke, J., 1996. SUS: A quick and dirty usability scale, in P. W. Jordan, B. Thomas, I. L. McClelland, & B. Weerdmeester (Eds.), Usability Evaluation in Industry. Taylor and Francis, London, pp. 189-194.

Brusilovsky, P., 1996. Methods and techniques of adaptive hypermedia. User modeling and user-adapted interaction. 6(2-3), 87-129.

Brusilovsky, P., 2003. From Adaptive Hypermedia to the Adaptive Web, in G. Szwillus & J. Ziegler (Eds.), Mensch & Computer 2003: Interaktion in Bewegung. Vieweg+Teubner Verlag, Wiesbaden, pp. 21-24.

Burke, R., 2007. Hybrid web recommender systems, in P. Brusilovsky, A. Kobsa, & W. Nejdl (Eds.), The Adaptive Web. Springer, Berlin, Heidelberg, pp. 377-408.

Burke, R., & Ramezani, M., 2011. Matching Recommendation Technologies and Domains, in F. Ricci, L. Rokach, B. Shapira, & P. B. Kantor (Eds.), Recommender systems handbook. Springer US, Boston, MA, pp. 367-386.

Cahyani, A., Basuki, A., Sari, R., & Kustiyahningsih, Y., 2015. Design an adaptive e-learning application architecture based on IEEE LTSA reference model. Telecommunication Computing Electronics and Control. 13, 284-289.

Cremonesi, P., Garzotto, F., & Turrin, R., 2013. User-centric vs. System-centric evaluation of recommender systems. In *Proceedings of the* Human-Computer Interaction, Springer, Berlin, Heidelberg, 334-351.

Dahdouh, K., Oughdir, L., Dakkak, A., & Ibriz, A., 2019. Building an e-learning recommender system using association rules techniques and R environment. International Journal of Information Science & Technology. 3(2), 11-18.

Davis, F. D., 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS quarterly, 319-340.

Dwivedi, P., & Bharadwaj, K., 2015. e-Learning recommender system for a group of learners based on the unified learner profile approach. Expert Systems. 32(2), 264-276.

El-Bishouty, M., Aldraiweesh, A., Alturki, U., Tortorella, R., Yang, J., Chang, T.-W., . . . Kinshuk. 2019. Use of Felder and Silverman learning style model for online course design. Educational Technology Research and Development. 67(1), 161-177.

El-Bishouty, M., Saito, K., Chang, T.-W., Kinshuk, & Graf, S., 2013. An interactive course analyzer for improving learning styles support level. In *Proceedings of the* Human-Computer Interaction & Knowledge Discovery in Complex, Unstructured, Big Data, Springer, Berlin, Heidelberg, 136-147.

Fraihat, S., & Shambour, Q., 2015. A framework of semantic recommender system for e-Learning. Journal of Software. 10(3), 317-330.

Graf, S., & Kinshuk. 2008. Analysing the behaviour of students in learning management systems with respect to learning styles. Advances in Semantic Media Adaptation and Personalization. 93, 53-73.

Graf, S., Liu, T.-C., & Kinshuk. 2010. Analysis of learners' navigational behaviour and their learning styles in an online course. Journal of Computer Assisted Learning. 26(2), 116-131.

Heng, L. E., Sangodiah, A., Muniandy, M., & Yuen, P. K., 2018. Integration of learner's model into learning management system environment. Journal of Fundamental & Applied Sciences. 10(1), 1771-1778.

Ibrahim, N., & Fadzil, N. H., 2013. Informal setting for learning on campus: Usage and preference. Procedia - Social and Behavioral Sciences. 105, 344-351.

Imran, H., Belghis-Zadeh, M., Chang, T.-W., & Graf, S., 2015. Toward recommending learning tasks in a learner-centered approach. Emerging Issues in Smart Learning, 337-344.

Imran, H., Belghis-Zadeh, M., Chang, T.-W., Kinshuk, & Graf, S., 2014a. A rule-based recommender system to suggest learning tasks. In *Proceedings of the* International Conference on Intelligent Tutoring Systems, Springer, Honolulu, 672-673.

Imran, H., Belghis-Zadeh, M., Chang, T.-W., Kinshuk, & Graf, S., 2016. PLORS: a personalized learning object recommender system. Vietnam Journal of Computer Science. 3(1), 3-13.

Imran, H., Hoang, Q., Chang, T.-W., & Graf, S., 2014b. A framework to provide personalization in learning management systems through a recommender system approach. In *Proceedings of the* Asian Conference on Intelligent Information and Database Systems, Springer, 271-280.

Jonassen, D. H., 2000. Revisiting activity theory as a framework for designing student-centered learning environments, in D. H. Jonassen & S. M. Land (Eds.), Theoretical foundations of learning environments. Lawrence Erlbaum Associates Publishers, Mahwah, NJ, pp. 89-121.

Karagiannis, I., & Satratzemi, M., 2018. Analysis of students' navigational behaviour in a moodle course. In *Proceedings of the* European Conference on e-Learning, Retrieved from http://0-

search.ebscohost.com.aupac.lib.athabascau.ca/login.aspx?direct=true&db=ehh&AN=134951073&site=eds-live, 226-234.

Khribi, M., Jemni, M., & Nasraoui, O., 2009. Automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval. Journal of Educational Technology & Society. 12(4), 30-42.

Kinshuk, Chang, M., Graf, S., & Yang, G., 2010. Adaptivity and personalization in mobile learning. Technology, Instruction, Cognition & Learning. 8(2), 163-174.

Klašnja-Milićević, A., Ivanović, M., & Nanopoulos, A., 2015. Recommender systems in e-learning environments: a survey of the state-of-the-art and possible extensions. Artificial Intelligence Review. 44(4), 571-604.

Lewis, J. R., 1995. IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. International Journal of Human-Computer Interaction. 7(1), 57-78.

Lewis, J. R., & Sauro, J., 2009. The factor structure of the system usability scale. In *Proceedings of the* International Conference on Human Centered Design, Springer, Berlin, Heidelberg, 94-103.

Manouselis, N., Drachsler, H., Verbert, K., & Duval, E. (2012). *Recommender systems for learning*.

Paris, C. L., 1988. Tailoring object descriptions to a user's level of expertise. Computational Linguistics. 14(3), 64-78.

Pedersen, S., & Liu, M., 2003. Teachers' beliefs about issues in the implementation of a student-centered learning environment. Educational Technology Research and Development. 51(2), 57-76.

Rahman, M. M., & Abdullah, N. A., 2018. A personalized group-based recommendation approach for web search in e-learning. IEEE Access. 6, 34166-34178.

Ramu, V., Taib, N., Aziz, N. F., & Ismail, M., 2018. Transitional spaces in higher education as efficient informal academic learning spaces. Jurnal Kejuruteraan, Teknologi dan Sains Sosial. 1(1), 1-11.

Rhode, J., Richter, S., Gowen, P., Miller, T., & Wills, C., 2017. Understanding faculty use of the learning management system. Online Learning. 21(3), 68-86.

Saleh, M., Salama, R., & Bokhary, S., 2018. Augmenting moodle with adaptability environment based on learning styles. The International Journal of Engineering and Science. 7(6), 17-23.

Salehi, M., Pourzaferani, M., & Razavi, S. A., 2013. Hybrid attribute-based recommender system for learning material using genetic algorithm and a multidimensional information model. Egyptian Informatics Journal. 14(1), 67-78.

Sharif, N., Afzal, M. T., & Muhammad, A., 2015. Semantic based e-learning recommender system. European Scientific Journal. 1, 369-377.

Shobana, B., & Kumar, G. S., 2018. A personalised e-learning system using reinforcement learning technique. International Journal of Pure and Applied Mathematics. 118(18), 4803-4813.

U, L., Chai, Y., & Chen, J., 2018. Improved personalized recommendation based on user attributes clustering and score matrix filling. Computer Standards & Interfaces. 57, 59-67.

Vandewaetere, M., Vandercruysse, S., & Clarebout, G., 2012. Learners' perceptions and illusions of adaptivity in computer-based learning environments. Educational Technology Research and Development. 60(2), 307-324.

Wen, H., Fang, L., & Guan, L., 2012. A hybrid approach for personalized recommendation of news on the web. Expert Systems With Applications. 39(5), 5806-5814.

Zaiane, O., 2002. Building a recommender agent for e-learning systems. In *Proceedings of the* International Conference on Computers in Education, IEEE, Auckland, New Zealand, 55-59.

Zapata, A., Menéndez, V. H., Prieto, M. E., & Romero, C., 2013. A framework for recommendation in learning object repositories: An example of application in civil engineering. Advances in Engineering Software. 56, 1-14.

Zhong, J., Xie, H., & Wang, F. L., 2019. The research trends in recommender systems for e-learning. Asian Association of Open Universities Journal. 14(1), 12-27.

Appendix A: Certification of Ethical Approval

Athabasca University
RESEARCH CENTRE

CERTIFICATION OF ETHICAL APPROVAL

The Athabasca University Research Ethics Board (AUREB) has reviewed and approved the research project noted below. The AUREB is constituted and operates in accordance with the current version of the Tri-Council Policy Statement: Ethical Conduct for Research Involving Humans (TCPS) and Athabasca University Policy and Procedures.

Ethics File No.: **21778**
Principal Investigator:
Dr. Sabine Graf, Associate Professor
Faculty of Science & Technology\School of Computing & Information Systems

Project Team:
Dr. Hazra Imran (Co-Investigator)
Mr. Mohammad Belghis-Zadeh (Research Assistant)

Project Title:
Evaluation of Recommender System Prototypes

Effective Date: November 14, 2018 **Expiry Date**: November 13, 2019

Restrictions:

Any modification or amendment to the approved research must be submitted to the AUREB for approval.

Ethical approval is valid *for a period of one year*. An annual request for renewal must be submitted and approved by the above expiry date if a project is ongoing beyond one year.

A Project Completion (Final) Report must be submitted when the research is complete *(i.e. all participant contact and data collection is concluded, no follow-up with participants is anticipated and findings have been made available/provided to participants (if applicable))* or the research is terminated.

Approved by: **Date:** November 14, 2018

Carolyn Greene, Chair
Athabasca University Research Ethics Board

Athabasca University Research Ethics Board
University Research Services, Research Centre
1 University Drive, Athabasca AB Canada T9S 3A3
E-mail rebsec@athabascau.ca
Telephone: 780.675.6718