

Ontology-Based Recommender System for a Learning Sequence in Programming Languages

<https://doi.org/10.3991/ijet.v16i12.21451>

Saman Shishehchi (✉)
Imam Khomeini International University, Iran
shishechi@bzte.ac.ir

Nor Azan Mat Zin
Universiti Kebangsaan Malaysia, Bangi Selangor, Malaysia

Esmadi Abu Abu Seman
Universiti Malaysia Sabah-Labuan International Campus,
Jalan Sungai Pagar, Malaysia

Abstract—The same learning process in educational systems could be boring and time consuming for some learners. This problem arises from the lack of personalized learning sequence for learners with different knowledge level. Recommender systems play an important role in assisting the learners to find suitable learning materials and personalized learning sequence. Use of ontology for knowledge representation in knowledge-based recommender systems would facilitate sharing, reuse and common terminology. Since programming concepts have logical relationships among together so, traditional education systems are more stressful and very time-consuming. This paper aims to propose an ontology-based recommender system to present a Personalized Learning Sequence in Programming (PLSP) domain which is depended to learner's knowledge level. A recommender module and, the knowledge base module are integrated together in the proposed framework. The recommender module as the main module in the framework, has three stages which is working based on semantic rules and ontology representation. Evaluation of the system was carried out by comparing the non-recommender system (web-based search) using 32 ICT respondents. Results demonstrate that the participants who used the proposed system spent 1119 seconds to find the suitable learning path in comparison to those who used a non-recommender system (3480 seconds) in the same learning material. It means that learners who follow learning path with PLSP, are more suitable for them. Furthermore, the average mean value of usability test is 4.47, (5 maximum scale) which indicates that the system proved to be useful, was easy to use, and satisfied the users.

Keywords—Learning sequence, Ontology, Programming language, Recommender system

1 Introduction

Programming language is one of the fundamental courses in computer science filed. Learners believe that traditional ways for programming learning add more stress during learning process [1]. Furthermore, for a few learners, this teaching model cannot match their needs. In traditional model, learners need to pass step by step programming stages to learn how to coding. Almost, fundamental programming concepts have logical relationships among together. In contrast to modern learning methods such as e-learning and distance learning, traditional education systems are very time-consuming. The learner spends too much time search for the desired topic when the probability of reaching the goal and finding the appropriate material is not guaranteed. The most beneficial feature of the e-learning system is its independence of the physical classroom. In recent years, e-learning environment is an additional way to replace traditional learning. However, the biggest problem in e-learning systems is the finding suitable learning sequence for every learner with the same request [2]. In the other word, lack of personalized learning sequence is annoying learners. E-learning systems are expected to provide suitable learning materials for different learners. However, this expectation cannot be met due to the similarity of existing learning material. Learning process via e-learning systems is the same for all learners and offers same learning sequences to learners whose background and knowledge may be different. The main principle of personalized learning environment is that no personalized sequence learning material for all learners [3]. Studies on learning sequence discovery have done to recommend suitable learning sequence to make a personalized learning environment. Furthermore, it mostly focuses on providing of suitable learning sequence to facilitate learning process [4]. Recommender systems play an important role in assisting the learners to find suitable learning materials and personalized learning sequence. Use of ontology for knowledge representation in knowledge-based recommender systems has become an interesting research area since would facilitate sharing, reuse and common terminology. The personalized learning sequence increases learning effectiveness and motivate learners to study enthusiastically. We aim to propose a semantic recommendation system to assist learners to learn main concepts of programming domain in the best way. The valuable characteristics of the proposed system are using ontology to represent the knowledge and give personalized learning path based on knowledge level to learners.

The draft of this paper is as follows. Section II gives the related works. Section III shows the proposed method. Section IV shows the results and analysis for the above system. Finally, Section V concludes along with the work which can be done afterward in future.

2 Related Works

Several researchers are proposed semantic recommender system based on different techniques. The personalized learning sequence, the learner's knowledge level,

knowledge representation and the learning process technology should be consideration in the previous works.

Vesin and his colleagues [5] emphasize to the principles of learner style identification and content recommendation for course personalization. They tried to recommend materials based on learners style not learner's knowledge level. The research work (Pata & Moura, 2018) finds the learning materials for learners based on learners' needs in terms of their learning styles. Another research [6] recommends a personalized learning sequence to the learner using the genetic-based algorithm. Xu and his colleagues [7] developed a system to show a suitable learning sequence via tag-based user profile. They didn't mention about how knowledge level used in this system. Al-Muhaideb [8] showed the personalized suitable learning sequence using genetic and memetic algorithms. It didn't use ontology representation while ontology would facilitate sharing, reuse and common terminology [5]. Another tutoring system (Sutskever, Vinyals, & Le, 2014) supports a web-based course delivery platform with a set of "intelligent" functions. It can provide both learner modeling and automatic learning sequence generation. Different learners have different learning sequence to learn the same learning materials. The generation of learning sequence in this system has 3 stages based on learning goals. Mr Hooshyar and his colleagues [9] developed intelligent tutoring system based on Bayesian network. Based on their system, different learners with different knowledge levels have different solutions to catch learning goals. A novel framework for curriculum generation (Singh, Ahuja, & Kumar, 2018) was done based on mathematical model and then implemented on field of geo-physics. INTUITEL approach [10] includes an intelligent tutoring system that gives learners recommendations and feedback about what the best learning sequence is for them according to their profile, learning progress, context and environmental influences. Results of INTUITEL show that developed learning sequence for different learners are more suitable for them. Developed tutoring system [11] proposed pedagogical module for making decisions in curriculum sequencing and evaluating the learner model. Indeed, curriculum sequencing is based on execution of instructional plan generated based on learner knowledge of domain matter. Chen [12] developed genetic-based personalized e-learning system, which can generate appropriate learning sequence. He was evaluated learners using a pretest. Based on the results of pre-test, the proposed genetic-based personalized e-learning system shows personalized curriculum sequencing through simultaneously considering courseware difficulty level and the concept continuity of learning sequences to support web-based learning. One more research related to learning sequence is about a technique for developing an individualized learning sequence that meets learner need [13] but they didn't follow learner's knowledge level. This proposed technique used ontology to generate learning sequence and then used genetic algorithm to choose the suitable one of them. Idris [14] used neural network to select suitable learning material based on learner's goal and then generate learning sequence graph through the sequencing of suitable learning materials. Furthermore, he tried to generate the optimized learning sequence for different learners.

Despite to the importance of ontology, most of the related works didn't use ontology to present the knowledge while ontology would facilitate sharing, reuse and

common terminology. Some related works used other techniques to present their knowledge while developed knowledge are usable for own systems and are not reusable for other one. Besides, some the other works used ontology while didn't take into account the learners' knowledge level to provide learning path.

Considering the importance of **knowledge level** in learning systems [15, 16], and the effective performance of **ontology**, we have applied both of them to propose a semantic recommender system to present personalized learning sequence. Furthermore, most previous related works[17] recommend learning materials and not learning sequence to learners. Since the programming course is the main course for ICT learners, it is selected as a domain for testing.

3 Methods

3.1 Proposed framework

The proposed framework of Personalized Learning Sequence in Programming domain (PLSP) is shown in Figure 1. This framework is consisting of two main modules; Recommender and Knowledge based modules. This paper emphasizes to role of recommender module.

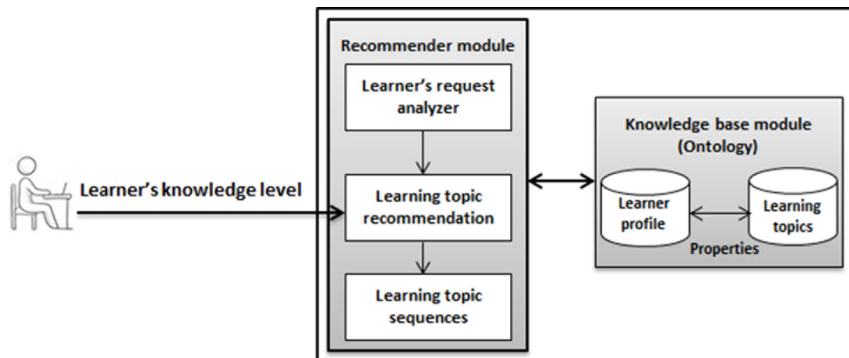


Fig. 1. PLSP framework

As be seen in Figure 1, the learner's knowledge level is the input for the proposed system. The knowledge level evaluation is the separate module which was developed by our team in previous research. The learner's knowledge level evaluation done according to the grading scale (Table 1).

Table 1. Grading scale [18, 19]

Grade	Percentage of correct answers	Score	Knowledge Level
Excellent (A)	90% - 100%	5	Advance
Very Good(B)	80% - 89%	4	Advance
Good (C)	70% - 79%	3	Medium
Average (D)	50% - 69%	2	Medium
Poor (F)	Less than 50%	< 2	Poor

The knowledge-based module developed a domain ontology to represent knowledge related to learners and programming learning materials. Developed ontology is a backbone of the knowledge-based module for PLSP which is created using protégé 4. Figure 2 depicts an overview of main classes and subclasses in ontology. Recommender module as a main module of PLSP system consists of three stages; learner’s request analyser, learning material recommendation and learning material sequences. All details related to these three stages will describe in next section.

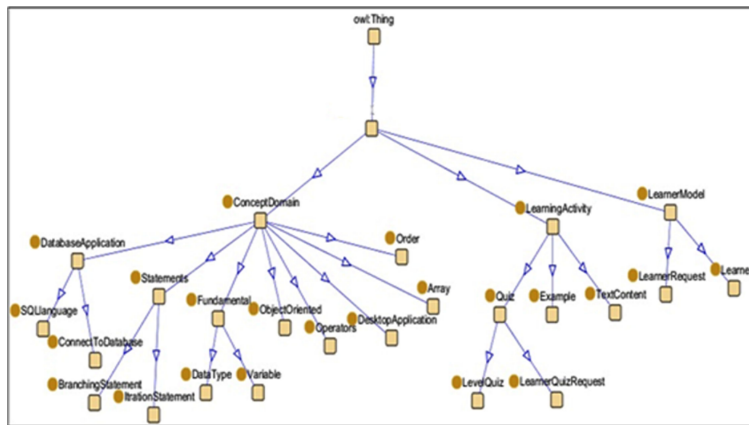


Fig. 2. Overview of Ontology

Figure 3, expresses the relationship among modules of proposed framework and shows the sequence of recommendation process flow.

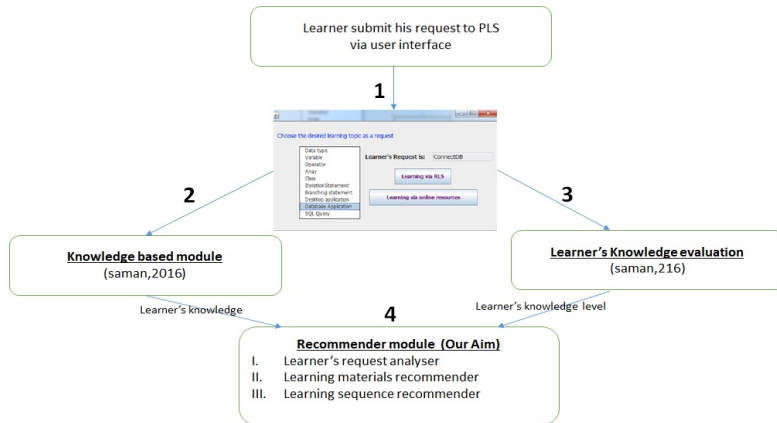


Fig. 3. Process flow among modules of framework

According to Figure 3, each learner has a learning material as a learner's request in his mind. Firstly learner chooses and submits his learner's request via user interface to PLSP to learn. Then, knowledge-based module (domain ontology) is created and next, level of learner's knowledge is evaluated. In last and main stage, the recommender module will give recommended learning materials and the personalized sequence of learning materials to learners. Complete details related to recommendation process are as a follow:

1. **Learner's request analyser:** Graphical user interface was developed using Java (Netbeans 7.1) Jena[20] is a semantic framework that was used to connect developed ontology to graphical user interface. Learners can see all programming learning materials via user interface and able to choose one of them as a learner's request to learn according to Figure 4.

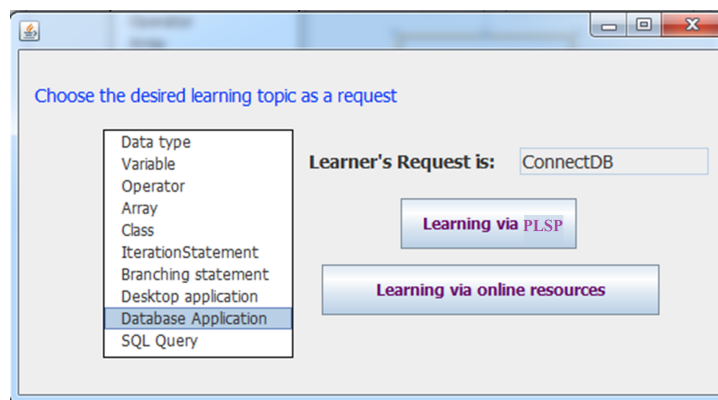


Fig. 4. Learning topic selection as a learner's request

After selecting a learning material as a learner’s request, the learning process method can be chosen via two learning ways. The first learning way is through the PLSP and the second way is using non-recommender learning systems. Non-recommender system means that the system without recommender module like searching via web. Result of search will be developed based on online programming resources. To make learning process easier for learner via search on the web, some URLs related to programming learning contents are suggested to learners and also they can use other online resources to achieve their learning request. The request analyser stage has two parts. The first part is defining some semantic rule (SWRL)[21] to find out all the prerequisites for the learner’s request and the second part is creating the query (SPARQL)[22] to extract all needed prerequisites for the desired learner’s request. All predefined object properties of the ontology are shown on Table 2.

Table 2. Defined Object Properties

Object Properties	Explanation
HasAverage	To show the average knowledge of learners in the learning material. Subject is from “Learners”class and Object from “ConceptDomain”class.
HasVeryGood	To show Very Good knowledge of learners in the learning material. Subject is from “Learners”class and Object from “ConceptDomain”class.
HasExcellent	To show Excellent knowledge of learners in the learning material. Subject is from “Learners”class and Object from “ConceptDomain”class.
HasGood	To show Good knowledge of learners in the learning material. Subject is from “Learners”class and Object from “ConceptDomain”class.
HasPoor	To show Poor knowledge of learners in the learning material. Subject is from “Learners”class and Object from “ConceptDomain”class.
HasNotKnowledge	To show zero knowledge of learners in the learning material. Subject is from “Learners”class and Object from “ConceptDomain”class.
HasRequest	To show the particular learning material as a learner’s request. Subject is from “Learners”class and Object from “ConceptDomain”class.
HasKnowledge Medium	To show Medium level of knowledge as a current learner’s knowledge level in learning materials. Subject is from “Learners”class and Object from “ConceptDomain”class.
HasKnowledge Poor	To show Poor level of knowledge as a current learner’s knowledge level in learning materials. Subject is from “Learners”class and Object from “ConceptDomain”class.
HasPrerequisite	To show the prerequisite associations for each learning material. Subject is from “ConceptDomain”class and the Object is from “ConceptDomain”class.
HasRecomemndation	To show the recommended learning material for particular learner’s request. Subject is from “LearnerRequest”class and the Object is from “ConceptDomain”class.
HasOrder	To show the priority sequence’ priority of learning materials. Subject is from “ConceptDomain”class and the Object is from “Order”class.

Logical relationships among concepts of programming topics are important during the learning process. So, learners need to have an enough knowledge on prerequisite concepts for their learning request. For example, “Variables and Data Types” concepts should be learned before “Array” concept. Rule1, is created to extract all the

prerequisites for the desired learner’s request via a recursive structure. Rule 1, follows the transitive format in SWRL rules format.

Rule 1: $\text{HasPrerequisite}(?x,?y) \wedge \text{HasPrerequisite}(?y,?z) \rightarrow \text{HasPrerequisite}(?x,?z)$

Rule 1, returns the prerequisites for the current learning material and also the prerequisites of extracted prerequisites based on the transitive structure of the rules. The rule engine (Jess) adds more topics as prerequisites for the learner’s request. Query 1 is created to extract all needed prerequisites for the desired learner’s request. The Query1 is as a follow:

Query 1:

```
PREFIX PROGRAMMING:<http://PLSP#>
Select DISTINCT? y where < http://PLSP# SpecificRequest
>PROGRAMMING:HasPrerequisite ? y.
```

2. Learning topics recommendation: Learning topic recommendation selects some appropriate learning materials as a recommended learning material for learners. Recommended learning materials are recommended based on the initial learner’s knowledge level. As an example, if the learner has “Advance” level on any learning material means that he/she has mastered on the current learning material and does not need to study it again. Thus, the system automatically removes the current learning material from the list of prerequisites for the learner’s request. Therefore, the list of recommended learning materials consists of prerequisites which the knowledge level of current learner is “Medium”, “Poor” or “NoKnowledge”. Through this filtering, the recommended learning materials are those learning materials that he/she needs to learn more about. This part of recommender stage also needs to have some rules and queries to return the suitable learning materials as the recommended topics to learners based on the current request. These rules use the properties explained in ontology. Rule2, developed to provide the recommended learning material for the current learner:

Rule 2:

- i. $\text{Learners}(?l) \wedge \text{hasLearner_LearnerRequest}(?l,?lr) \wedge \text{hasLearnerRequest_Request}(?lr,?r) \wedge \text{hasRequest}(?l,?r) \wedge \text{HasPrerequisite}(?r,?p) \wedge \text{HasNotKnowledge}(?l,?p) \rightarrow \text{hasRecommendation}(?lr,?p)$
- ii. $\text{Learners}(?l) \wedge \text{hasLearner_LearnerRequest}(?l,?lr) \wedge \text{hasLearnerRequest_Request}(?lr,?r) \wedge \text{hasRequest}(?l,?r) \wedge \text{HasPrerequisite}(?r,?p) \wedge \text{HasKnowledgePoor}(?l,?p) \rightarrow \text{hasRecommendation}(?lr,?p)$
- iii. $\text{Learners}(?l) \wedge \text{hasLearner_LearnerRequest}(?l,?lr) \wedge \text{hasLearnerRequest_Request}(?lr,?r) \wedge \text{hasRequest}(?l,?r) \wedge \text{HasPrerequisite}(?r,?p) \wedge \text{HasKnowledgeMedium}(?l,?p) \rightarrow \text{hasRecommendation}(?lr,?p)$

Based on the definition of object properties, the rule engine (Jess) infers these rules and adds some topics as recommended learning materials. These three rules assigned

the name of current learner to the learning materials with “NoKnowledge”, “Medium” or “Poor” knowledge levels in the ontology. These assignments are done using the “hasRecommendation” object property to update the ontology. Learners can request to see the list of recommended learning materials through user interface.

The result of Query2 shows the desired recommended list to learner.

Query 2:

```
PREFIX PROGRAMMING: http://PLSP#
Select DISTINCT? Y
Where {<http: //PLSP # User + SpecificRequest >
PROGRAMMING:hasRecommendationY.}
```

All recommended learning materials are assigned to the learners via “hasRecommendation” property. The learner can briefly see the recommended learning materials for the preferred request. As an example, Figure 5 shows the recommended learning materials for a selected learner named “Farhana”.

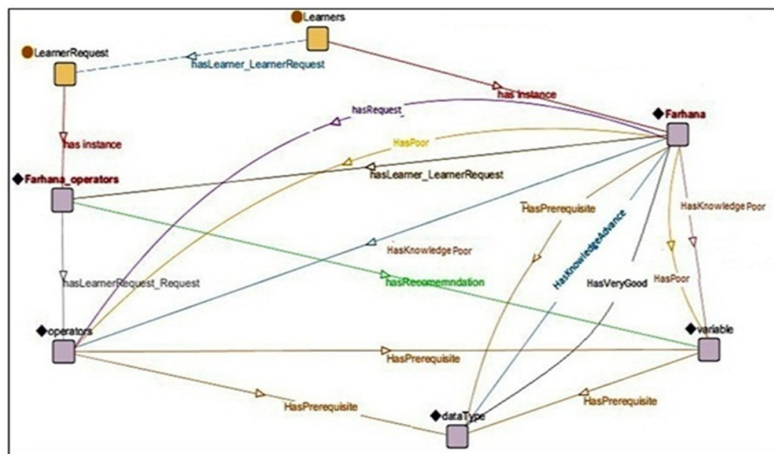


Fig. 5. Recommended learning materials for “Farhana”

As be seen in Figure 5, her request is “Operator” and the prerequisites for “Operator” are “Variable” and “Data type”. These prerequisites are assigned to the request by “HasPrerequisite”. The initial learner’s level for “Variable” is “Poor”, which is assigned to the learner by the “hasPoor” property. Since the initial learner’s level for “DataType” is “VeryGood”, according to the grading table (Table 1), the initial knowledge level for “DataType” is “Advance” and for “Variable” is “Poor”. These knowledge levels are indicated by “HasAdvanceKnowledge” and “HasPoorKnowledge”. According to the request analyser, the request has two prerequisites such as “Variable” and “DataType”. Since the new knowledge of the learner in “DataType” is “Advance” so the system removes this learning material

from the recommended list and recommends just “Variable” to the learner. The “Variable” is assigned to “Farhana” by “hasRecommendation”.

3. Learning topic sequence: Learning topic sequence or learning material sequence [23] is an important research issue for educational systems and is one of the key components in ITS since no one fixed learning sequence will be appropriate for all learners. Learning sequence is a set of planning techniques used in educational systems to provide the learner with the most appropriate sequence of elementary knowledge units to learn or problems to solve. The learning sequence construction is fundamental to the education process [24]. As learners may progress differently, to help a learner achieve the designated learning outcomes with minimal effort, approaches for generating adaptive learning sequences were represented [24]. The structure of the learning process follows the textbook. The logical relationships among learning materials are important so the logical sequence of the recommended materials should be observed in the recommender stage. The logical relationship among the learning materials which are located in the “ConceptDomain” class by the “hasOrder” property has been provided. Query 3 extracts all the recommended learning materials via “hasRecommendation” property, according to their assignment to individuals belonging to the “Order” class, which includes some individuals with the priorities indicated via numbers.

Query 3:

```
PREFIX PROGRAMMING:http://PLSP#
Select DISTINCT? y ?z where { < http://PLSP#
user_specificRequest >
PROGRAMMING:hasRecomemndation ?y.
?y: PROGRAMMING:HasOrder ?z.
ORDER BY ?z }
```

The “ORDER BY” causes the individuals to be sorted in ascending order and shows the logical orders among the recommended learning materials. Figure 6 shows an overview of recommender module process for a learner who named “Sahar” with “Database Application” or “ConnectDB” as a learner’s request.

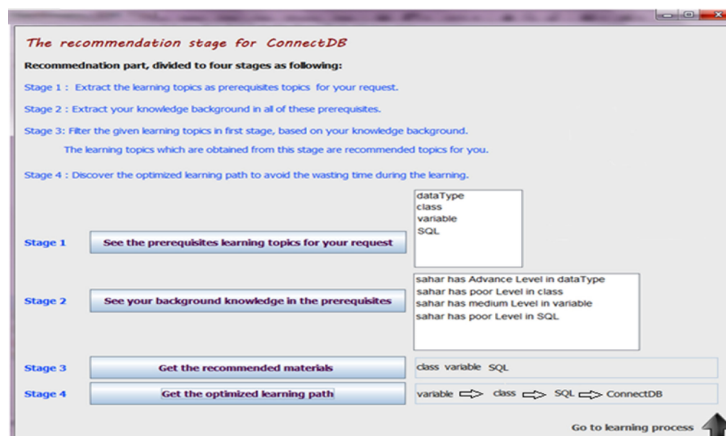


Fig. 6. Overview of recommender process for “Sahar”

As be seen in Figure 6, the learner can see the recommenders in 4 stages. The first stage is the needed prerequisites for the learner’s request (database application). This learner’s request has 4 learning materials as prerequisites. In the second stage, the system shows the last learner’s knowledge level in each pre-requisites. In the third stage, the system filters some learning materials based on the learner’s knowledge level and recommends a suitable one to learners as the recommended learning. According to the current learner’s knowledge in these learning materials, he/she is at the “Advance” level in the “data type” learning material, so the system removes this one from the list of recommended learning materials. The final recommended topics for the current learner named “Sahar” are: “Class”, “Variable” and “SQL”. After the last recommended topic, the learner’s request will be added as a final learning material in the learning sequence. Finally, a learning sequence is recommended to the learner. As an example, the desired learning sequence for her request is as follows:

“Variable” “Class” “SQL” “ConenctDB”.

Hence, the learner starts to learn “Variable” and then the after topics in the sequence. The learning process continues until learner’s request is achieved.

4 Evaluation and Results

Based on related works, there is no personalized learning sequence inclusive ontology representation and knowledge level involvement in programming domain. It is not fair to compare PLSP with the learning systems in other domains since the types of learning materials, difficulty of learning materials and also the knowledge level of learners are diverse. The purpose of evaluation is to demonstrate the capability of proposed system and show the efficiency of personalized learning sequence in learning process. To emphasize the importance of personalized sequences and also due to the lack of similar system in related works, this system is compared to a system without recommender module. The selected learning system without a recommender

module in the same domain could be web-based search using online programming resources. Some other research[5] tried to show the efficiency of its system using self-assessment and didn't compare to other one, so in addition to compare with non-recommender system, we measured the learner's satisfaction of our system using user acceptance test to show high degree of satisfaction.

4.1 Data collection

The evaluation process was evaluated the efficiency of PLSP comparing to web-based search. Respondents are learners who had to meet certain requirements and want to learn programming learning materials. They could be from different fields of ICT and different levels of study. To work with PLSP, the level of learner's knowledge does not matter, thus they could be at the novice, medium or advance level. To find some participants, an announcement to inform learners was done, but only 32 were selected based on similarity of their knowledge level. Even with the small number of participants, it was still enough for the evaluation part since the minimum number, based on similar literature works, should be around 35. As an example, in similar researches such as [25], 32 learners were employed for comparison between an intelligent tutoring system and a real human tutor. All the experimental data for this study is collected in the computer lab, where the 32 learners used the proposed system. The experimental data, evaluation and results will be explained in the following sections.

4.2 Main parameter of PLSP evaluation

Studies on learning sequence discovery mostly focus on providing of suitable learning sequence to facilitate learning process and increases learning effectiveness [4]. Since quality enhancement is significant matter in a PLSP framework, in this section, time parameter to show the efficiency of PLSP via statistical analysis is emphasised.

Time spent [26-28], which is defined as the duration of finding the relevant learning materials as recommended learning materials as well as producing the personalized learning sequence. The time for both recommender system (PLSP) and web-based searching process are measured when learners are working with these systems. Since the comparison between web based searching and PLSP is the main analysis of the experimental study, the collected data was analysed via an independent T-test [29] using SPSS version 17 [30, 31]. In addition, a survey questionnaire was used for user acceptance test. The result of this questionnaire shows user satisfaction towards PLSP.

4.3 Experimental study

The efficiency of PLSP is analyzed to shows less time spent using the system in comparison to web-based search system. Respondents are divided to two independent groups; A and B (Figure 7).

Group A		Group B	
Key Name	Request	Key Name	Request
L1	Variable	L`1	Variable
L2	Array	L`2	Array
L3	Data type	L`3	Data type
L4	Iteration statement	L`4	Iteration statement
L5	Variable	L`5	Variable
L6	Array	L`6	Array
L7	SQL	L`7	SQL
L8	Database application	L`8	Database application
L9	Desktop application	L`9	Desktop application
L10	Operators	L`10	Operators
L11	Object oriented	L`11	Object oriented
L12	Iteration statement	L`12	Iteration statement
L13	Desktop application	L`13	Desktop application
L14	Desktop application	L`14	Desktop application
L15	Array	L`15	Array
L16	Operators	L`16	Operators
L17	Database application	L`17	Database application
L18	SQL	L`18	SQL
L19	Database application	L`19	Database application

Fig. 7. Respondents in group A and group B

As be seen in Figure 7, every learner has a key name and the specific request. The respondents in group A, use the proposed recommender system (PLSP) and the group B use web-based searching to achieve their goals. A respondent in group A will be compared with another respondent in group B, as corresponding respondents. The same learning request, is suggested to corresponding respondents. As an example, learner with key name “L1” from group A, has the same request (“Variable”) as the learner with key name “L`1” from group B.

The time spent in the second group (B) is measured when the learner starts to find related materials and is stopped when the learner finish the searching. Time spent for participants in group A and B is referring to find the relevant learning materials. Total time is measured automatically by the system without learner access. The comparison result emphasizes to less time spent using PLSP. As a sample, Table 3 shows the comparison between learners with key name “L8” and “L`8” and Table 4 shows comparison between learners with key name “L5” and “L`5”.

Table 3. Time spent comparison between L8 and L`8

Learners	Time spent using commendation system	Time spent using Online searching
L 8	47	214
L` 8	41	169

Table 4. The Time spent comparison between L5 and L'5

Learners	Time spent using commendation system	Time spent using Online searching
L 5	49	254
L'5	59	236

According to tables above, the respondents L8 and L5 who used PLSP spent less total time in comparison with their corresponding respondents L'8 and L'5 who used web-based learning system. This result is obtained according to the same learner's knowledge level and same learner's request of the corresponding respondents (L8, L'8) and (L5, L'5).

Statistical analysis is conducted based on time spent to emphasize the validity of empirical results.

4.4 Analysis of time spent

As mentioned in the previous section, the two groups of participants are independent groups so the independent T-test is chosen to analyse these groups based on the total time spent during the finding of their request. The T-test is done based on the total amount of time, which was measured in the experimental section. The results of the T-test for the total time spent to achieve the learner's requests are shown in Figure 8.

		Levene's Test for Equality of Variances		t-test for Equality of Means						
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
Difference time	Equal variances assumed	14.042	.001	-8.57	30	.000	-147.563	17.223	-182.74	-112.388
	Equal variances not assumed			-8.57	17.73	.000	-147.563	17.223	-183.79	-111.338

Fig. 8. T-test for the total time spent

The significant value in this result is .001, which is smaller than 0.05. This value shows that the equality between the variances is observed and since the value of (Sig.) or "p" is 0.000 and less than 0.05, it means that the significant difference between the two learning systems based on the time spent factor is evident. The desired score for this T-test is: $t(32) = 2.337$ ($p = -8.57 < 0.05$). Figure 9 represents the related diagram to show the comparison between measured time spent for all respondents in group A and B who used PLSP and web-based searching respectively. The majority of learners spent less time via PLSP compared with web-based learning.

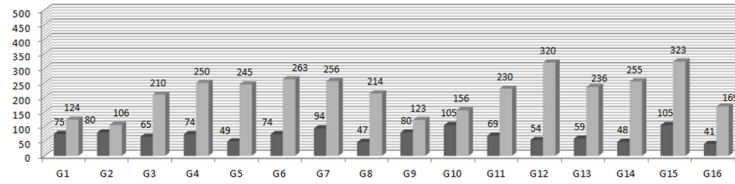


Fig. 9. Time spent comparison among learners

In order to see the total time spent via PLSP and web-based search, general diagram is shown in Figure 10. This diagram emphasizes that less time was spent in PLSP compared to web-based learning system.

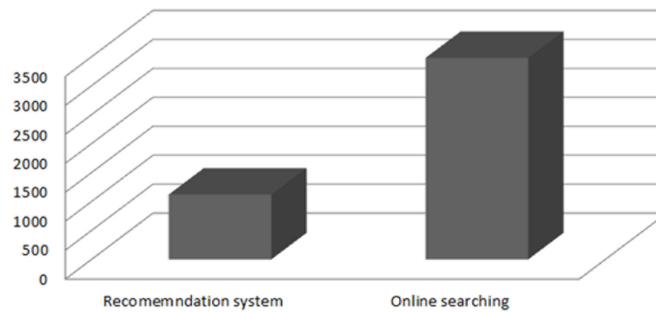


Fig. 10. Total time spent by all learners

4.5 User acceptance test

A user acceptance test (UAT) is provided to measure learner’s satisfaction when using PLSP. The survey questionnaire had to consist of well-prepare questions with a different view in order to find the PLSP degree of capability. All the questions are designed to produce an integrated questionnaire. The classifications of questions in the questionnaire are Usefulness, Ease of use, and Ease of learning and Satisfaction. This questionnaire is prepared just for the first group A, who used the recommender system because only the respondent’s satisfaction of PLSP is considered and not web-based learning. The validity and reliability of this questionnaire will be discussed in the following sections. As noted in Table 5, the value of Cronbach's Alpha for the 16 real samples is 0.918.

Table 5. The Cronbach’ alpha for reliability test

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
.918	.922	20

The mean of each question, each construct as well as total mean are calculated to show the acceptance degree of PLSP (Table 6). This Table illustrates that the mean values for all questions, are between 4 “Agree” and 5 “Strongly agree”. The mean values for each construct (Usefulness, Ease of use, Ease of learning and Satisfaction) are more than 4 and the total mean is 4.47 which are acceptable values to show the high degree of user acceptance of PLSP. This total mean (4.47) of questionnaire shows the acceptable level of learner’s satisfaction.

Table 6. Mean values for questions of questionnaire

Construct	Question Number	Mean	Mean of classification
Usefulness	Q1	4.44	4.54
	Q2	4.50	
	Q3	4.69	
	Q4	4.63	
	Q5	4.50	
	Q6	4.44	
	Q7	4.63	
Ease of use	Q8	4.31	4.30
	Q9	4.38	
	Q10	4.06	
	Q11	4.19	
	Q12	4.31	
Ease of learning	Q13	4.56	4.40
	Q14	4.69	
	Q15	4.13	
Satisfaction	Q16	4.38	4.65
	Q17	4.75	
	Q18	4.38	
	Q19	4.81	
	Q20	4.69	

5 Conclusion and Future Works

The overall theme of our research is to construct the semantic recommendation system for personalized learning sequence in programming language. This study tried to develop personalized learning sequence based on learner’s knowledge. To evaluate the performance of this developed recommender system, 32 learners were recruited to participate in this study. The empirical study was done and data analysis was conducted based on time spent parameter. The evaluation process of this research was evaluated the efficiency of PLSP comparing to web-based searching This means that PLSP learners simultaneously spend less time during the searching process. This is may be illustrated to the recommender system which have helped reduce time spent during the learning process via suitable recommended materials. Furthermore, a user acceptance test (UAT) result emphasize the high degree of satisfaction when using PLSP.

In the future work, we will modify PLSP based on the specifics object-oriented programming languages.

6 Acknowledgement

The authors would like to thank all the people who have contributed to this research, directly or indirectly. This work is supported by Universiti Kebangsaan Malaysia under the DCP-2017-007/2 research grant.

7 References

- [1] Al-Bastami, B.G. and S.S.A. (2017) Naser, Design and Development of an Intelligent Tutoring System for C# Language.
- [2] Tang, T. and G. McCalla, (2005) Smart recommendation for an evolving e-learning system: Architecture and experiment. *International Journal on E-learning*, 4(1): p. 105-129.
- [3] Kurilovas, E., I. Zilinskiene, and V. Dagiene. (2015). Recommending suitable learning paths according to learners' preferences: Experimental research results. *Computers in Human Behavior*. 51: p. 945-951. <https://doi.org/10.1016/j.chb.2014.10.027>
- [4] Xie, H., et al. (2017). Discover learning path for group users: A profile-based approach. *Neurocomputing*. 254: p. 59-70. <https://doi.org/10.1016/j.neucom.2016.08.133>
- [5] Vesin, B., et al. (2013) Ontology-based architecture with recommendation strategy in java tutoring system. *Computer Science and Information Systems*, 10(1): p. 237-261.
- [6] Chen, Y.-L., L.-C. Cheng, and C.-N. Chuang. (2008). A group recommendation system with consideration of interactions among group members. *Expert systems with applications*, 34(3): p. 2082-2090. <https://doi.org/10.1016/j.eswa.2007.02.008>
- [7] Xu, D., et al. (2012). Personalized learning path recommender based on user profile using social tags. in 2012 Fifth International Symposium on Computational Intelligence and Design. IEEE. <https://doi.org/10.1109/iscid.2012.133>
- [8] Al-Muhaideb, S. and M.E.B. Menai. (2011). Evolutionary computation approaches to the Curriculum Sequencing problem. *Natural Computing*. 10(2): p. 891-920. <https://doi.org/10.1007/s11047-010-9246-5>
- [9] Hooshyar, D., et al. (2015). A flowchart-based intelligent tutoring system for improving problem-solving skills of novice programmers. *Journal of Computer Assisted Learning*, 31(4): p. 345-361. <https://doi.org/10.1111/jcal.12099>
- [10] Verdú, E., et al. (2017). Integration of an intelligent tutoring system in a course of computer network design. *Educational Technology Research and Development*. 65(3): p. 653-677.
- [11] Jeremić, Z., V. Devedžić, and D. Gašević. (2004). An Intelligent Tutoring System for learning design patterns. Proc. of the ICWE2004 WSh. on Adaptive Hypermedia and Collaborative Web-based Sys., Munich, Germany.
- [12] Chen, C.-M. (2008). Intelligent web-based learning system with personalized learning path guidance. *Computers & Education*, 51(2): p. 787-814. <https://doi.org/10.1016/j.compedu.2007.08.004>
- [13] Shmelev, V., M. Karpova, and A. Dukhanov. (2015). An approach of learning path sequencing based on revised Bloom's taxonomy and domain ontologies with the use of genetic algorithms. *Procedia Computer Science*. 66: p. 711-719. <https://doi.org/10.1016/j.procs.2015.11.081>

- [14] Idris, N., N. Yusof, and P. Saad. (2009). Adaptive course sequencing for personalization of learning path using neural network. *Int. J. Advance. Soft Comput. Appl.* 1(1): p. 49-61.
- [15] Prentzas, J., I. Hatzilygeroudis, and J. Garofalakis. (2002). A web-based intelligent tutoring system using hybrid rules as its representational basis. in *International Conference on Intelligent Tutoring Systems*. Springer. https://doi.org/10.1007/3-540-47987-2_16
- [16] Ferguson, K., et al. (2006). Improving intelligent tutoring systems: Using expectation maximization to learn student skill levels. in *International Conference on Intelligent Tutoring Systems*. Springer. https://doi.org/10.1007/11774303_45
- [17] Tarus, J.K., Z. Niu, and G. Mustafa, (2018). Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning. *Artificial intelligence review.* 50(1): p. 21-48. <https://doi.org/10.1007/s10462-017-9539-5>
- [18] Stankov, S., et al. (2008). TEx-Sys model for building intelligent tutoring systems. *Computers & Education.* 51(3): p. 1017-1036. <https://doi.org/10.1016/j.compedu.2007.10.002>
- [19] Stankov, S., M. Rosić, and V. Glavinić (2001). Using Quizzes in an Intelligent Tutoring System. *Proceedings of International Summer School of Automation*, p. 87-91.
- [20] McBride, B. (2002). Jena: A semantic web toolkit. *IEEE Internet computing.* 6(6): p. 55-59. <https://doi.org/10.1109/mic.2002.1067737>
- [21] Horrocks, I., et al. (2004). SWRL: A semantic web rule language combining OWL and RuleML. *W3C Member submission.* 21(79): p. 1-31.
- [22] Bailey, J., et al., *Web and semantic web query languages: A survey*, in *Reasoning Web. 2005*, Springer. p. 35-133. https://doi.org/10.1007/11526988_3
- [23] Karampiperis, P. and D. Sampson. (2005). Adaptive learning resources sequencing in educational hypermedia systems. *Journal of Educational Technology & Society*,8(4).
- [24] Yang, F., F.W. Li, and R.W. Lau. (2010). An open model for learning path construction. in *International Conference on Web-Based Learning*. Springer.
- [25] Beal, C.R., E. Shaw, and M. Birch. (2007). *Intelligent tutoring and human tutoring in small groups: An empirical comparison*. IOS Press.
- [26] Shishehchi, S., et al. (2011). Review of personalized recommendation techniques for learners in e-learning systems. *IEEE*.
- [27] Aleven, V., et al. (2009). A new paradigm for intelligent tutoring systems: Example-tracing tutors. *International Journal of Artificial Intelligence in Education.* 19(2): p. 105-154
- [28] Jeremić, Z., J. Jovanović, and D. Gašević, (2009). Evaluating an intelligent tutoring system for design patterns: The DEPTHs experience. *Educational Technology & Society.* 12(2): p. 111-130.
- [29] Sedgwick, P. (2010). Independent samples t test. *BMJ.* 340
- [30] Morgan, G.A., et al. (2010) *IBM SPSS for Introductory Statistics: Use and interpretation*: Taylor & Francis.
- [31] Bryman, A. and D. Cramer. (2012). *Quantitative data analysis with IBM SPSS 17, 18 & 19: A guide for social scientists*. Routledge. <https://doi.org/10.4324/9780203180990>

8 Authors

Saman Shishehchi received her Ph.D. in Information Technology in 2016 and currently is an Assistant Professor in Department Computer and Electrical Engineering, Buin Zahra Higher Education Center of Engineering and Technology,

Iran. Her research areas is Ontology Based Application, Semantic Recommender Systems and E-learning Technology. Email: shishechi@bzte.ac.ir

Nor Azan Mat Zin received her Ph.D. in Information Science in 2005 and currently is a Professor (Interactive computing) in Research Centre for Software Technology and Management (SOFTAM), Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia. She is the Head of Games lab and Multimedia Software and Usability Research Group. Her areas of specialisation include serious games, HCI (Accessibility) and E-learning Technology.

Esmadi Abu Abu Seman is with the Faculty of Computing and Informatics, Universiti Malaysia Sabah-Labuan International Campus, Jalan Sungai Pagar, 87000 W.P. Labuan, Malaysia.

Article submitted 2021-01-24. Resubmitted 2021-02-28. Final acceptance 2021-03-01. Final version published as submitted by the authors.