Journal of Computers in Education
https://doi.org/10.1007/s40692-022-00250-y

An improved adaptive learning path recommendation model driven by real-time learning analytics

Nisha S. Raj[1] · **V. G. Renumol**[1]

Received: 3 February 2022 / Revised: 22 September 2022 / Accepted: 20 October 2022

Abstract
The advancements in the education sector made e-learning more popular in recent years. The velocity of learning content creation and its availability is also growing exponentially day after day. It is challenging for a learner to find a learning path for a course with a vast content repository. So, recommending a learning path helps the learners streamline the learning materials systematically and achieve their goals. This article proposes a learning path recommendation approach focused on knowledge building and learning performance analysis. The model considers both static and dynamic learner parameters for learning path generation. The difficulty level of the learning resources is tuned based on the real-time performance analysis of the students. The learning resources are recommended based on learning preferences and the ability of a learner to learn the specific learning resource. The model also predicts the learning time and the expected score for each learner. Root Mean Square Deviation and Coefficient of Determination (R-Squared error) measures are used to find the correctness of the prediction. The model is also checked for its adaptivity to the learners' changing behavior and diversity of the LOs recommended for different learners. Ninety-six undergraduate learners participated in the study. The experimentations are conducted with 530 LOs from selected courses. The comparison results with three existing models show a better performance from the proposed approach with an average accuracy rise of 30% in learning path prediction based on the expected duration of learning 27.8% in expected score prediction with the second-best performing model. It is observed that the real-time learning analytics using the implicit learner log data benefits the recommendation process. LO rating strongly indicated the enhancement of learner satisfaction and experience with a rise of 25.5% when comparing the rating share with the second-best model.

Keywords Recommender systems · Personalized learning · Learning paths · Learning environments · E-learning

✉ Nisha S. Raj
nishasraj@cusat.ac.in

Extended author information available on the last page of the article

Published online: 14 December 2022

Introduction

Technological and pedagogical advances are redefining education. E-learning is at the center of this conjunction. Along with technology advancements, scalability and reduced costs also made e-learning attractive. Many learning materials, text resources such as basic web pages, and multimedia resources as videos have been uploaded in recent years due to this fast growth of information and communication technology usage in education. The substantial amount of information in the learning systems creates cognitive overload and disorientation. Also, the learner population is highly dynamic and heterogeneous, with differences in their learning preferences, basic knowledge, learning style, and interaction with the learning environment (Chen et al., 2014, Ciloglugil & Inceoglu, 2018, Christudas et al., 2018). Hence, the knowledge delivered in fixed sequences or patterns will create dissatisfaction and disinterest in learners. The "one-size-fits-all" approach may not be satisfactory for the learners (Essalmi et al., 2010). Learners demand personalized knowledge delivery which adapts to their changing needs (Deng et al., 2017; Hwang et al., 2020a). In general, Learning Management Systems, LMS, do not meet the requirements of individual learners depending on their profile. However, taking learners' profiles into account can improve the learning experiences and course success of students (Imran et al., 2016). In facilitating personalization in LMS, recommender systems can be used to suggest appropriate learning objects to learners in order to enhance their learning. Thus, generating adaptive, customized learning paths is an important research topic in the design of learning environments (George & Lal, 2019; Hwang et al., 2020b; Raj & Renumol, 2021). In education technology, it is advantageous to extract new hidden patterns in learner data for online learning systems. Personalized learning full-path recommendation research is particularly significant for the advancement of E-learning systems (Zhou et al., 2018). Online learners develop massive data with big-data features about their learning habits, which helps discover individual learning patterns (Chen & Zhang, 2014). The data generated from the learning environments can be fed back to the system, contributing to learning evaluation and monitoring (Sachan & Saroha, 2022). Thus, the learning material recommenders can be improved to monitor the student performance and adapt to the changes in the performance and their learning preferences.

According to Chen et al. (2014), the critical challenges to be focused on in the design of the learning resource recommenders are to provide convenient and effective access to the learning resources and boost the learner's learning experience and satisfaction. So, the recommender system needs to adapt to the learner's changing performance and preferences. The recommender system should map adaptive and personalized resources reducing the knowledge gap with the learner (Shi et al., 2020). This research work aims to generate personalized learning paths adapted to the changes in learning preferences and performance in real-time. Here, the learner log is constantly monitored, and getting feedback from this log, adjust the values of dynamic factors contributing to personalized content recommendations. A learning object (LO) is needed to learn a topic, and the sequences of the topics form the learning path. Learning objects are internet deliverable and reusable, instructional components that support learning (Wiely, 2002). We plan to find suitable learning objects for each topic and order them based on the knowledge dependencies between the topics.

To reduce the knowledge gap between the learner and the learning resources, the proposed model dynamically calculates the difficulty level of an LO and selects the LO with high similarity with the learner's ability. The system consists of the learner model, the LO model, learner logs, and the recommender engine (Raj & Renumol, 2018). The learner model represents the learner's behavioral information (learning preference, learning style), status information (learned LO, basic knowledge), and dynamic information (time for LO completion, score of LOs, number of attempts). The current study uses the Felder-Silverman learning style to analyze the learner's learning style (Felder & Silverman, 1988). The LO is modeled using IEEE LOM (Risk, 2002), and the fields are used as described in Raj and Renumol (2019). The IEEE LOM fields are Structure, Format, Learning Resource Type, Interactivity Type, Interactivity Level, and Difficulty. An additional field is used to hold the average rating for each LO. To model the learner, learning material, and learner log this study uses the ontology-based method, and the design of the model is explained in Sect. 4. The earlier works show that the ontology-based LO recommender system can perform well in personalized learning environments. The significance of ontology-based models is that they better handle the cold-start issues and data sparsity problems in recommendations. The SPARQL queries extract similar learning objects based on similar learner grouping (Joy et al., 2021). Thus, the proposal works in the following steps:

- Find all possible sequences of topics forming paths between the starting and final topics.
- For each topic
    - o Find the Top N LOs based on the learning log of similar learners
    - o Refine the Top N LO list so that the ability of the learner suitably maps with the difficulty of the LO.
- Form the LO sequences based on the topic sequences
- Suggest the sequence with the best time duration suitable for the learner's available time
- Log the interactions of learner such as score, time taken to learn, number of attempts, the rating is given for each learned material
- Update the difficulty level of the learning material and ability of the learner according to the logged information

For every course, the instructor annotates the learning resources and develops the knowledge link between the topics. The relationship between the topics is stored as a graph structure called a concept graph or knowledge graph. This work uses a two-layered knowledge graph, with lessons forming the upper layer and LOs in the lower layer. The knowledge graph is traversed to find the concept sequence. The instructor assigns an initial difficulty level for each LOs. Also, the instructor gives an approximate time expected to learn the LO. These values are used to cold start the model. Each time a learner interacts with the LO, the time and score associated with LO are updated. The suggested learning paths are optimized for learner attributes and aim to maximize learner satisfaction and performance.

The major implications of our work are:

(1) Design of a learner model and LO model with static and dynamic parameters, where the length of the parameters is comparatively larger than most of the existing works. Most of the recommendation system studies focus on learning object ratings as feedback from the learner. The model in this study considers the dynamic parameters as the time taken for learning, score obtained from learning, number of attempts taken for learning a concept along with the ratings provided by the learners. The model adjusts the difficulty of the learning objects and the ability of a learner to learn that object in real-time so that the adaptivity of the recommendations is enhanced.
(2) Real-time learner data analytics is incorporated to improve the accuracy of the next recommendations. Evaluation of the model using real learner data.
(3) Implementation of an adaptive learning path recommendation model which works in two steps. Initially, a concept path is constructed by arranging the concept in order following a knowledge graph. Secondly, personalized learning objects are selected for each of the concepts in the path, forming a learning path.
(4) The model addresses data sparsity and cold-start issues of recommender systems.
(5) A summary of recent studies on the learning path adaptation.

The rest of the paper is organized as follows. The problem statement and research questions are introduced in Sect. 2. Section 3 briefly explains the related works on the current domain from 2018 to 2021. Section 4 elaborates on the design of the knowledge and domain models. Section 5 describes the learning path recommendation model that is proposed in this paper. The subsections of Sect. 5 elaborate on the algorithm used for learning path generation. The experimentation procedure and results are presented in Sect. 6, and Sect. 7 discusses the developments in correlation with the research questions and states the limitations of the work. Section 8 concludes the current work with a discussion on future work.

Problem statement and research questions

The adaptive learning path recommendation model aims to provide learners with the most suitable personalized sequences of learning activities to follow (de Marcos et al., 2008). So, the objective of this work is to find a flexible model for personalized and adaptive recommendations than the "one-size-fits-all" approach. It was stated by Chen et al. (2014) about the importance of providing better access to learning resources to enhance the learner's performance and satisfaction. Also, suppose the learning model is generating the learning path by optimizing learner performance in terms of learning time and scores obtained. In that case, the paths are more effective (Chen, 2011). So, we decided to focus more on the learning duration, expected score, adaptivity, and acceptance of recommended learning resources. So, the research questions addressed are stated as follows:

RQ1: Can we accurately predict learning duration and expected score during the learning path recommendation process?

RQ2: What learner-centric personalization parameters generate adaptive and diverse learning paths in an e-learning environment?

Related works

Many studies address the problem of learning path recommendations. From the literature, we can observe that this area's works fall under domain-based and heuristic models (Raj & Renumol, 2021). Both studies use a knowledge structure for recommending, considering the similarity between the learning objects. The researchers use various algorithms/methods/tools based on the method chosen for guiding the learning path. They have evaluated the models in online or offline mode, considering how much the students learn through the system. Some studies adopted the system performance evaluation methods, and few studies conducted a user satisfaction study.

Table 1 summarizes the related works study done as a preamble for the current research.

The studies above show that the learning paths are sequences of LOs using different machine learning methods or algorithms. The major personalization parameters used are the learning style, learning time, the score obtained, prior knowledge. The LOs with more similarity with the learner characteristics are recommended. But none of the studies explore the likelihoods that the 'difficulty' parameter of LO can be different for different students. Also, the difficulty of an LO differs for the learner to varying points in the timeline as learning progresses. This gap is addressed in the current research paper. It is observed that most studies concentrate on selecting LOs according to learner needs or preferences, only very few works give attention to sequencing the topics. So, here we are using a two-layer model for sequencing the selected LOs based on knowledge relations.

A learning path is the linear list of LOs, organized based on their knowledge relation. So, the problem of recommending a learning path can be reduced to two issues, (1) To know the knowledge relations between topics (2) To suggest appropriate LO for a topic. The knowledge relations are obtained from the knowledge graph of the course. Following the relationships, the knowledge units are organized. And the LOs are selected based on the mapping between the LO and learner characteristics. The model predicts the learning time and score for every LO recommendation. The learner's performance can be calculated in terms of learning time, scores obtained and the number of attempts needed, and satisfaction can be calculated based on the ratings given for each LO (Nabizadeh et al., 2017; Tarus et al., 2017).

Table 1 Summary of related works from 2018 to 2021

Citation	Recommendation Approach	Method/Algorithm/Tools	Evaluation approaches	Personalization parameters
Nabizadeh et al. (2018)	Considering the time limit of students to learn suggests learning path	Mean, Median, Item Response Theory (IRT), DFS, Probability of Error for learning time and score	Offline/system performance	Learning style/knowledge level/time taken
Segal et al. (2019)	Endorsing a set of questions to users in ascending order of difficulty based on their responses	EduRank, Voting method, Collaborative filtering methods	System performance/online	Performance based on score
Cun-Ling et al. (2019)	They suggest a path that improves a student's learning results while considering their learning style, learning need, and prior knowledge	Graph Theory, Improved Immune Algorithm, Felder-Silverman Learning Style Index	System Performance/Online/UserStudy	Learning style/knowledge level
Vanitha et al. (2019)	Recommending a path based on a user's emotion and cognitive capacity	Ant Colony Optimization, GeneticAlgorithm	System performance/online	Performance based on score
Li and Zhang (2019)	Repeatedly recommending the unattempted courses with the best score to a user based on user similarities and the learning effect of prior users	Network embedding, Learning effects, Breadth first search, Depth-first search, Random traverse	Offline	Learning style
Cai et al. (2019)	During the learning process, suggesting the best path specific to the needs of each knowledge unit	Knowledge tracing model, reinforcement learning, neural network, Markov decision process	Offline/system performance	Performance based on score
Nabizadeh et al. (2020)	Recommending a path that improves a user's score in the least amount of Time	Item Response Theory (IRT), Two-Layer course graph, Probability of error for learning Time and score, Depth First Search	Offline/system performance/online/user	Time taken/performance based on score/knowledge level

Table 1 (continued)

Citation	Recommendation Approach	Method/Algorithm/Tools	Evaluation approaches	Personalization parameters
Niknam and Thulasiraman (2020)	Learners are divided into different groups, and a path is chosen for them dependent on their prior knowledge	Fuzzy C-Mean, Clustering methods, Ant colony optimization algorithm	Online	Learning style/knowledge level
Shi et al. (2020)	Creating all possible paths while considering the students' learning objectives and needs, and recommending the one with the greatest score	Graph traversal algorithm, Knowledge graph, Cohen kappa coefficient for finding the quality of learning materials	Online/UserStudy	Time Taken/Performance based on score
Benmesbah et al. (2021a)	Generate sequences of LO considering the learner preference, course relations and LO features	Modified Genetic Algorithm	Offline/simulated data	Performance
Benmesbah et al. (2021b)	Learning Path Adaptation using concept graphs	Modified Genetic Algorithm	Offline/simulated data	Performance
Ramos et al. (2021)	Generating visual representation of learning path and suggestion are made to group of collaborative learners	Clustering algorithms K-means and clustering done based on learning path	Online/UserStudy	Performance based on score
Son et al. (2021)	Generate learning paths suitable for specific learning skills based on MOOCs	Metaheuristic algorithms, Genetic Algorithm (GA) and Ant Colony Optimization Algorithm (ACO)	Offline	Time taken/Performance based on score/Learning Style
Wang et al. (2021)	Recommending a learning path and evaluating learner satisfaction. Based on that selecting an alternate path	Differential evolution (DE) algorithm and knowledge graph	Offline	Performance based on score

Context: learning material and learner modeling

The learner model represents the learner's behavioral information (learning preference, learning style), status information (learned LO, basic knowledge), and achievement (time for LO completion, score of LOs, number of attempts).

The learning path adaptation models deal with extracting relationships between the learning materials and the learner to select an appropriate learning path. As it is observed from the earlier studies that modeling the learning materials and learner are very significant (Tarus et al., 2017, Dorça et al., 2016). This section elaborates on the learner and learning material modeling used in this work. Ontologies are used as storage units for saving the learner-related parameters, learning object metadata, and learner activity log (Joy et al., 2021).

Learner model

To achieve learning adaptation, the personalized preferences and differences between the learners should be considered. In this study, we are focused on static and dynamic parameters for modeling the learners. The ontology's main notion is a learner class, which is represented by various object properties or parameters for each student.

The learner is modeled using static and dynamic information, which are listed as follows:

Static Parameters: *Learner identifier (LID), age, gender, stream of study, basic qualification, basic knowledge* and *learning style* < *active/reflective, visual/verbal, intuitive/sensitive, global/sequential* >. These factors are explicitly fed into the system.

Dynamic parameters which are extracted from the learner activity log: *ID of learning materials visited, learning time in minutes, score, rating, count of repeated attempts of the same material*. These parameters act as implicit feedback to the recommendation process.

Also, the work considers additional information that is read from the learners as *search topic information* and the *time availability* to learn the topic. These values are not stored in the ontology but are stored in variables. The parameters are represented as the type values of the learner ontology (static) and learning log (dynamic) ontology classes.

Learning material model

The learning material metadata forms a significant part in the content recommendation. The learning content forms a hierarchical order (Nabizadeh et al., 2017). The four levels are course, lesson, concept, and learning object. The courses, represented as concept maps/graphs, form the topmost level and are often called subjects. A course can be covered using more than one lesson. The next level of abstraction is concept/topic. They are units of knowledge and learned by a learning object. One

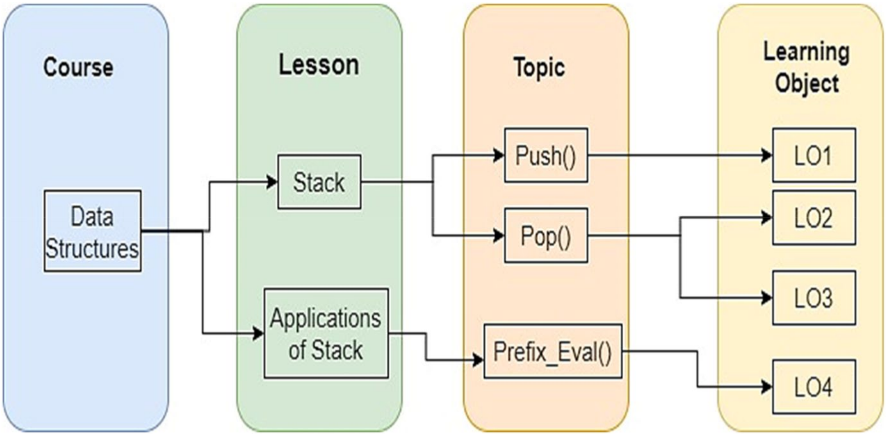

Fig. 1 Levels of abstraction of the learning content with example

concept is mapped to one or more learning objects. These LO selection and recommendations are the major task in an adaptive learning environment (Belacel et al, 2014; Dharani and Geetha, 2013). A sequence LOs forms a learning path. Sequencing the LOs according to the order of the related concepts makes the sequencing more meaningful (Shi et al., 2020). Figure 1 represents the level of learning content abstraction assumed in this work.

The learning paths form two-layered topics/concepts as the outer layer and the associated LOs in the second layer. The cognitive linkage between the concepts is visualized using a directed graph named a concept graph or knowledge graph. The graph's vertices form the concepts/topics, and the edges form their relationship (Benmesbah et al., 2021a; Zhu et al., 2018). Sequencing the concepts to develop a concept graph forms one part of the learning path adaptation problem.

The second part of the problem is selecting the appropriate LO for each topic. In Fig. 1, we can see that the Pop() topic is associated with two LOs. So, more than one LOs associated with any topic, and the adaptation of LO selection is significant in personalized learning path recommendation. As stated in the introduction, we have used the IEEE LOM schema for representing the LO metadata.

Out of the nine categories of metadata description, the study adopts the general field and education field as both have significantly contributed to the personalized LO recommendations in our previous works (Joy et al., 2021; Raj & Renumol, 2019). The title is a unique name given to identify the LO. Duration in minutes, the organization of the structure, type or format of the LO, the level of its interactivity and the type (active, expositive, mixed) and learning resource type of the LO forms the static information about the LO. The difficulty level is considered as an integer value in this work, which is dynamically computed for each LO depending upon the learning log. The values of all static metadata and the initial difficulty level of the LO are provided by the instructor. The *LearningObject* class of the ontology stores LO metadata. This study considers two types of LO: Expository LO, LOex, and Evaluative LOs, LOev (Nabizadeh et al., 2017). For each concept, the model tries to recommend one LOex and one LOev. LOex helps the learner learn the concept, and LOev evaluates the knowledge.

The graphical representation the ontology classes and their relationships are depicted in the Fig. 2. The dotted lines show the common data object shared by two types. The ontology is created completely in Java using a set of JENA APIs. The data is defined using RDF tools. Jena is a Java framework for creating Semantic Web apps. It includes rich Java libraries for writing code that works with different versions of RDF and SPARQL following W3C standards. Jena has a rule-based inference feature, an ontology reasoning engine based on OWL and RDFS ontologies.

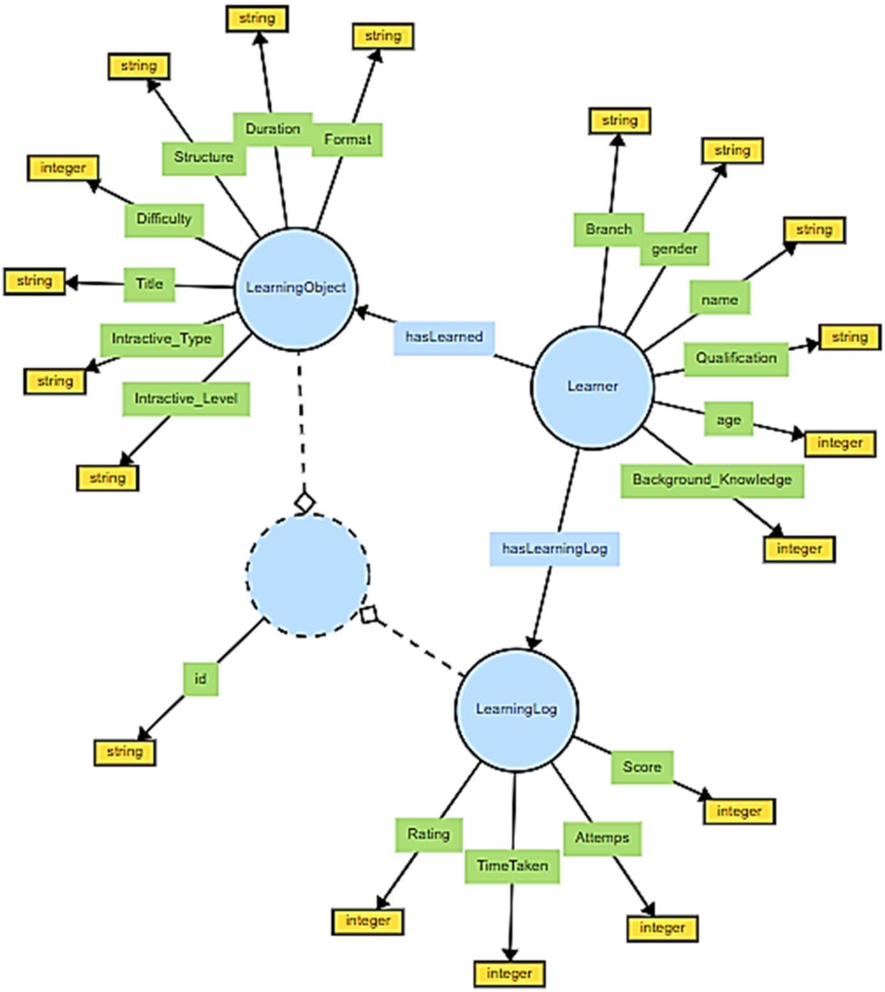

Fig. 2 Learner and learning object model

Learning path recommendation model

This section discusses the procedure to generate the learning paths as sequences of LO. This study uses a two-layered context/knowledge graph, KG, to represent the relation between the topics/KUs and the LOs, as shown in Fig. 3. We maintain a separate KG with specific starting and ending units for each lesson. The learner is asked to enter both the starting (SU) and target (TU) topics in a particular lesson they want to learn. The model creates a learning path connecting different KUs based on the first layer of KG. Also, from the second layer of the KG, an appropriate LO is chosen. The LOs selected are connected as a graph and are suggested to the students as a learning path (Algorithm 1, Fig. 4). Based on this observation, the learner's interaction is recorded and later recommends LOs.

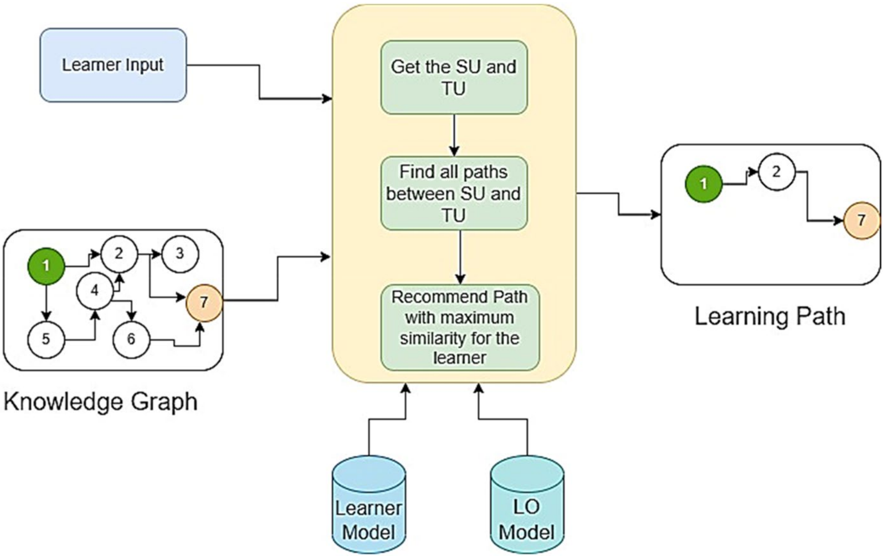

Fig. 3 An abstract model representing the workflow

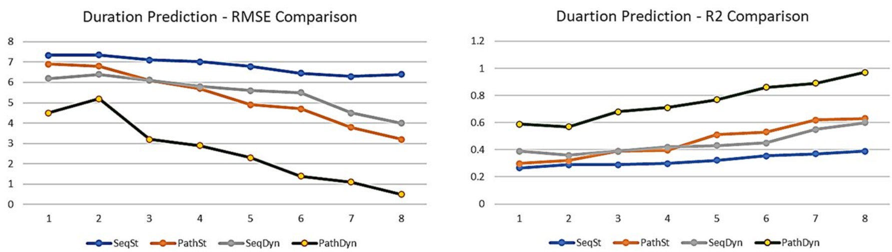

Fig. 4 Comparison of predicted time for completing LOs and actual time taken

The Depth First Search (DFS) is used to find the sequence of KUs (Algorithm 2) and algorithm 3 proposes the method to rank the LOs associated with each KUs.

Path generation

To generate the learning path, learners choose the starting and target knowledge units/topics SU and TU, respectively. From the graph, the sequence of topics is obtained and stored as a DAG, S = (LT, RE), a subset of KG. S's starting and ending topics are explicitly obtained from learners' input. We assume that the learners are familiar with all of the predecessor KUs of the SU. Initially, SU is set as the first and only node of the learning path. A suitable LO is selected for SU, estimate the expected time and score, and attach the LO with SU. Algorithm 3 will help to choose apt LO for the learner. Algorithm 2 is invoked recursively to get all the possible paths from SU to TU and stores the result in the variable PathList[]. The model suggests the shortest path from this set of possible LO paths.

Algorithm 1: Algorithm for path generation.

```
Input: ID, TU, SU, D, G.
Output: A path having the maximum score.
1.    currentnode ← SU;  SU: starting knowledge unit.
2.    P ← [SU]; . P is a list.
3.    Select LO for P; .                    LO selection Alg 3;
4.    i ← 1;
5.    TP ← EstimateTime(LO for SU); . T:time.
6.    SP ← EstimateScore(LO for SU);. S:score.
7.    PathList[] = AllPathsSelection(currentnode, KG, D, SU, TU , P, i) ; Algorithm 2.
8.    LearningPath ← Select the path with the best fit duration
9.    Return LearningPath;
```

Get all possible paths

The recursive function AllPathsSelection() will generate all possible paths from SU to TU (Algorithm 2). To nodes in the paths are LOs, and appropriate LOs are selected using Algorithm 3.

Algorithm 2: AllPathsSelection (currentnode,KG,D,P,TP,SP,i)

Input: currentnode, KG, D, SP, TP, P, i.
Output: Generate all paths from SU to TU
1. if (edgelist(currentnode) =∅) then ; currentnode is not leading to any other nodes
   a. PathList[i] ← (P, TP , SP); . PathList contains all possible paths
   b. i++ ;
2. else
   a. foreach (node ∈ edgelist(currentnode)) ; Recursively find the list of adjacent nodes
      i. LOlist ← Select all LOs ; Algorithm3
      ii. LOnode ←Select LOex and Loev from LOlist
      iii. Snode ← EstimateScore(LOnode) ; mean of previous scores obtained
      iv. Tnode← EstimateTime(LOnode) ; mean of previous time taken
      v. Node ← LOnode
      vi. TP+=Tnode
      vii. SP+=Snode
      viii. P←P.add(Node)
      ix. AllPathsSelection (Node, KG, D, P, TP, SP, i);
   b. PathList[i] ← (P, TP , SP); . PathList contains all possible paths
   c. i++ ;
3. Return PathList

Selecting suitable LOs

A set of LOs represents a topic or knowledge unit. Each LO is a self-sufficient module for learning a particular topic. In this study, we considered two types of LOs: explanatory LO (LOex) and evaluative LO (LOev). Explanatory LOs form the set of descriptive LOs in text, video, audio with different difficulty levels and interactivity. The Evaluative LOs forms the group of LOs that facilitates assessments of each knowledge unit. LOev is also of varying difficulty levels. The initial step in selecting a suitable LO for the required topic is to generate the top N recommendation list of LOs based on the learners' preferences.

In the case of a new learner, natural learner groups are generated by running SPARQL queries against Learner ontology. The learning history of the existing learners included in the learner group is extracted by running the SPARQL query from the LearnerLog and Learning Material sub-ontologies. Learner similarity with the multivariate clustering method is used in this step, as explained in our previous works (Joy et al., 2019, 2021). For each LO in the top N list, the LO with a difficulty level compatible with the learner's ability to learn is selected. The LO's difficulty and the learner's ability to understand the LO are computed dynamically using the Eqs. (1) and (2) respectively.

Table 2 Parameters and description

Parameters	Description
S	Maximum Score assigned for the LO
T	Minimum Time assigned for the LO
S_{ij}	The score obtained by learner *i* for the LO*j*
T_{ij}	Time taken by learner *i* for the LO*j*
$\widehat{s}_j$	Mean score obtained by learners who studied LO*j*; ratio of sum of scores and count of learners who attempted the LO
$\widehat{T}_j$	Mean time taken by learners who studied LO*j*; ratio of sum of Time taken and count of learners who attempted the LO
R_{ij}	Number of attempts made by learner *i* to study LO*j*

In an educational environment, the implicit feedbacks logged by the learners are the test scores, time taken to learn the LO, and the number of attempts made by the learner (Raj & Renumol, 2021). The ability of the learner to learn an LO and the difficulty of an LO is dynamically computed based on these logged parameters (Table 2).

$$A_{ij} = \alpha \times \left(\frac{S_{ij}}{S}\right) + \beta \times \left(\frac{T - T_{ij}}{T}\right) + \gamma \times \left(1 - R_{ij}\right), \tag{1}$$

Aij in Eq. 1 is the ability of the i^{th} learner to learn the j^{th} LO. Here the weighted sum of the score obtained, the time taken and the number of attempts made by the learner is used for updating the ability of the learner. S and T represent the maximum score obtained and the maximum time taken for the LO. R_{ij} is the number of attempts taken by the *i*th learner to learn the *j*th LO. We have assumed that when learner attempts an LO multiple times, they find it more difficult. Hence the value is considered to have an inverse effect in finding the ability of the learners. Hence the R_{ij} is subtracted from the desired maximum number of attempts for any LO, i.e.,1. Similarly, the higher the time taken for learning the LO, the learner is considered to be less able for learning the LO. Here also the maximum time taken recorded for that LO is subtracted from the T_{ij} and weight is applied. The higher the score, the better is the ability, so S_{ij} is taken as a positively correlated parameter. α, β, γ (Table 3) are the weights applied for the mean score, time and number of attempts respectively. The values are obtained by repeated trials for better prediction accuracy, using simulated and previously logged data.

Suppose the learner is not opting for the provided LO or is not completing the learning process. In that case, the score is adjusted to zero, and the time taken is admitted to the maximum assigned.

Table 3 Optimal parameter values obtained through trials using simulated and historical data

Parameters	α	β	γ	δ	ζ	θ
OptimalValue	0.4	0.3	0.3	0.5	0.3	0.2

$$D_i = \delta \times Dj + \xi \times 1 - \left(\frac{\hat{s}_j}{S}\right) + \theta \times \left(\frac{\hat{T}_j}{T}\right). \tag{2}$$

The Eq. 2 represents the computation of difficulty of the *j*th learning object, D_j. The previous value of D_j is one parameter in computation. Also, the score obtained and time taken by the learners who have learnt the LO_j is also considered for recomputing D_j. We have assumed that as the score obtained by learning the LO increases the difficulty decreases and when the time taken for learning the LO increases the difficulty also increases. The weights applied, δ, ζ, θ are obtained by conducting trials in the simulated and previously logged data.

Each LO is a combination of LOev and LOex. When the LO is initially stored in the LO repository, for each new LO, the instructor assigns a maximum score, maximum learning time, and difficulty level associated with the LO. When the *j*th LO is processed, each *i*th learner produces a new set of values as score S_{ij}, time T_{ij} and number of attempts R_{ij} committed to learning the concept. The expectation about efficient learning is a better score in a shorter and fewer attempt (Nabizadeh et al., 2020; Raj & Renumol, 2021). So, we took the weighted sum of the three parameters to quantify the learner's ability. A lower score and longer learning time make the LO more challenging to learn. Thus, the difficulty of LO is computed as the weighted sum of three parameters, the current difficulty, mean score obtained by all learners who learned the LO and mean of time taken by all learners who learned the LO.

The parameters named as α, β, γ, δ, ζ, θ are weights applied to the factors in the above equation where $\alpha+\beta+\gamma=1$ and $\delta+\zeta+\theta=1$. The optimal values that are observed using simulated data are shown in Table 3.

Consider the score obtained by *i*th learner for the *j*th LO is 5 where the maximum score is 10 in their initial attempt, by spending 2 min where the expected time is 3 min, then the ability is calculated as: $A_{ij}=0.4\times(5/10)+0.3\times(1- (2/3))+0.3\times(1–1)=0.3$ If the same happens in three attempts then the ability score will be − 0.3. Similar effects for score and time parameters too.

Suppose, the mean score obtained for the previously considered LO by all of its learners is 8, by spending an average of 3 min, and the LOs previously calculated difficulty is 0.6, then the new value will be: $D_j=0.5\times0.6+0.3\times(1 - (8/10))+0.2\times(3/3)=0.56$, since the score decreased the difficulty increased.

The highlight of Algorithm 3 for Selecting LO suitable for a learner is the dynamic computation of ability and difficulty factors. The similarity of *i*th learner and *j*th LO, S_{ij} is calculated using Euclidean distance, Eq. (3). The more similar LOs are selected from LOex and LOev lists for a KU, as explained in algorithm 3.

$$S_{ij} = \sqrt{\left(A_{i_j} - D_j\right)^2}. \tag{3}$$

So, if Aij is 0.3 and D_j is 0.56 the $S_{ij}=0.26$, which is the difference between the values; square is applied to make the values positive. Higher the S_{ij} more the *j*th LO compatible for the *i*th learner, assuming a match between the ability of the learner and difficulty of the LO.

Algorithm 3: Selecting LO

Recommended the learning path Pb with the highest score. Pb = GetLO(KU, LID, D).
Input:
The Knowledge Unit KU
The transaction data log D
The learner's ID
Output:
Recommend LO with maximum similarity for the user

1. Generate top N LO list based on the preference of the learner
2. for each LOi ∈ LOlist(KU); LOlist contains all LOs used for learning the KU
   a. Initialize Si =0
   b. Aij = calculateAbility(LID, D)
   c. Di = calculateDifficulty(LOi) ;
   d. Compute Si = Similarity (Lij, Aij, Dfi) and build (LOi,Si) pairs
3. Sort LOs in the ascending order of Si.
4. Select the top LOex / LOev
5. Return LOex / LOev

Experimentation and results

This section explains how the learning path recommendation model is evaluated. Also, it includes details of the experimentation process conducted and the results of the experimentations. We have created concept/knowledge graphs based on a C Programming and Data Structure course comprising 468 learning items and 1065 relationships. Another KG is made for Data Mining Course with 155 LOs and 254 connections. The distribution of LOs that are used in this study is shown in Table 4. Each LO is characterized by IEEE LOM parameters and the table shows the generic nature and count of the LOs used in the study. Both the KGs are fed into the system. The learning objects are crawled from various educational websites and extracted (Joy et al., 2019, 2021). The concepts are mapped based on the CUSAT syllabus for the courses CS201B Computer Programming, CS405 Data Structures and Algorithms, and CS604 Data Mining.

Table 4 Distribution of LOs

LO used	Count
Study materials in the form of PDFs	120
Study materials in the form of PPTs	71
Online Quizzes	69
Tutorials with description and practice question (interactive and non-interactive)	99
Study materials in the form of Videos	149
Others (Exercise, audio, diagrams, simulations etc.)	115

The relationships among the LOs are manually established by three instructors based on the course syllabuses and are verified by an expert. The LOs are fed into the system, and the initial fields are annotated by a group of ten instructors who teach the undergraduate courses in Computer Science and Engineering. Each learning object is annotated by at least three instructors. The majority decision is taken into consideration for fixing the initial values for discrete parameters of each LOs. The mean of three values is considered for fixing the initial values for continuous parameters. If there is no common agreement, the LO is passed to the subject expert for final decision. The initial values given to every field describing the characteristics of the LOs are anticipatory values given by the instructors. These values helped in the initiation of recommendation. As elaborated in the previous sections, when the recommendation process progresses, the values for each LO field is updated according to the learner feedback.

According to the learner's ability to learn and preferences, the current study suggests adaptive and personalized learning paths. The learners should analyze the quality and usability of the output. A total of 96 learners evaluated the model. The participants are enrolled for undergraduate Computer Science and Engineering programs in the two Indian state universities: APJ Abdul Kalam Technological University (KTU) and Cochin University of Science and Technology (CUSAT), Kerala, India. The experiments are conducted between February 2021 and September 2021. The experimentation was not a continuous process for all days. Each batch of experimentation was conducted for a batch of 10 students. Every participant is asked to join the learning process's three phases: (1) pre-test, (2) learning, and (3) practice. The pre-test and learning style identification is made at the entry into the procedure. The pre-test identifies the knowledge of the learner. In the learning phase they searched for a term, say "stack push", and the learning path is recommended to them. In the practice phase, they are asked to answer a maximum of three questions based on their study, which evaluates their learning performance. Three instructors monitored the student's activities and guided them throughout the procedure. All tests are conducted with questions of different difficulty levels based on Bloom's Taxonomy (Sosniak, 1994; Amstrong, 2016). In the experiment, it is verified that (i) the learning path generation algorithm proposed is better (minimizing the difference of actual and predicted score/time/rating values) than the baseline models (ii) more diverse paths are generated by the proposed approach than the LO-Learner mapping-based algorithm (iii) the learner's satisfaction is progressing with the recommended learning path. The proposed model is tested against three other models. We have named the existing models considered as SeqSt, SeqDyn, PathSt for better comprehension while discussing the experiments. PathDyn is the name that is given for the proposed model. The models are explained as:

- *SeqSt*: The model suggested a sequence of learning materials based on the static values of the parameters used (Tarus et al., 2017). Here, the ontology-based domain modeling method is explored. A personalized sequence of LOs is recommended based on historical library data. Learners previously rate the annotated LOs. This rating is used as the parameter to select LO and user preferences.

- *SeqDyn*: This model explores the dynamicity of the learner parameter. The changing values of learner satisfaction are considered here. But the sequencing of LOs is not done considering the knowledge relationship between the learning materials. We have implemented the SeqDyn based on previous works (Joy et al, 2021; Raj & Renumol, 2019). The change in student performance measures is incorporated with the basic model to evaluate the influence of dynamic parameters.
- *PathSt*: This model uses a knowledge graph as the structure used to sequence the LOs, thus providing a cognitive linkage between the learning objects (Nabizadeh et al., 2020). The base model tries to optimize suggestions based on the historical or available data of the learner. The difficulty parameter of the LO is considered static here.
- *PathDyn*: This proposed model considers a curated knowledge graph connecting the concepts/topics. The LOs are associated with these topics. LOs are selected based on the historical/available data and filtered further optimized by the ability of a learner to learn the LO of a particular difficulty level. Unlike other models here, the learner's ability and the LO's difficulty are adjusted based on their academic achievement, time invested to learn, the number of attempts they made, and ratings.

The primary aim of the experimentation is to compare the effectiveness of content sequence recommenders and path recommenders. The sequence recommenders SeqSt and SeqDyn recommend learning object sequences by analyzing the learner log. The learning path recommenders, PathSt and PathDyn suggest a learning path considering the learner log and a pre-designed knowledge graph. Here an inherited cognitive linkage is established between the suggested LO lists. The SeqSt and PathSt uses parameters with static values whereas the dynamicity of the parameters are used by the SeqDyn and PathDyn models. So two different aspects are experimented here, 1. The effectiveness of learning sequence and learning path 2. The effectiveness of static and dynamic parameters. The rest of the section elaborates on the experiment conducted and the result obtained.

A control experiment is conducted to find the model's accuracy to predict the learning time and score of the learners. The participants are divided into four groups, balancing their prior knowledge and learning style for control experiments. They were asked to conduct multiple topic searches on the available lists. For each iteration in the learning phase, the model predicted the time required to complete the learning path by the learner. Also, the actual time spent by the learner on each learning path is recorded. In the practice phase, both time and score are predicted and observed. Root Mean Square Error (RMSE) as in Eq. (4) and R-Squared (R2) Error as in Eq. (5) are used to derive interpretations from the predicted score and time against the observed values in the practice phase. The number of topic searches by the learners varies from 8 to 11 in the experimentation phase, so the first eight searches are considered here. The results are observed to be stabilizing by the first 8 iterations.

$$\mathrm{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(x_i - y_i\right)^2}, \tag{4}$$

$$R2 = 1 - \frac{\sum_{i-1}^{m}\left(x_i - y_i\right)^2}{\sum_{i=1}^{m}\left(\hat{y} - y_i\right)^2}, \tag{5}$$

where m is the number of observations in each iteration, x_i is the predicted value, y_i is the observed value and $\hat{y}$ is the mean of the observations.

RMSE gives the standard deviation of the residuals (prediction error) between the observed, x_i and predicted values, y_i. Here we have squared the residuals, took the mean and obtained the root of the value.

The results of experimentation are given below in Figs. 4 and 5. Each experimentation are considered for the first 8 iterations and *x*-axes of the figures represent the times of iteration.

Figures 4 and 5 shows that the proposed model is better in maintaining an accurate prediction of learning duration and expected score. The second-best model is observed as the sequence generating model which is modeled with dynamic parameters. The proposed path recommender shows an average of 30% more accuracy in predicting duration and 27.8% more accuracy in predicting the expected score. The time prediction is crucial as the learner wishes to complete the learning process in an available time depending on their learning goal (Zhu et al., 2018). The predicted score helps select the LO that can help the struggling learners perform better (Jdidou et al., 2021).

The Adaptivity, ADP, measures how much the recommendations suit a learner's preferences. In the current study, Euclidean distance is used for finding the similarity between a learner and recommended LO is used to measure the adaptivity (Meng et al., 2021). The lesser the value, the more chances for the learner to learn the topic (Plass & Pawar, 2020). The adaptivity is calculated as the mean of similarity measures of all recommendations in that iteration (Eq. 6).

$$ADP_t = \frac{\sum_{i,j=1}^{N} Sim_{ij}}{N}, \tag{6}$$

where N is the total number of learners, LO combinations in the iteration t. Figure 6 shows the adaptivity measure obtained for each model.

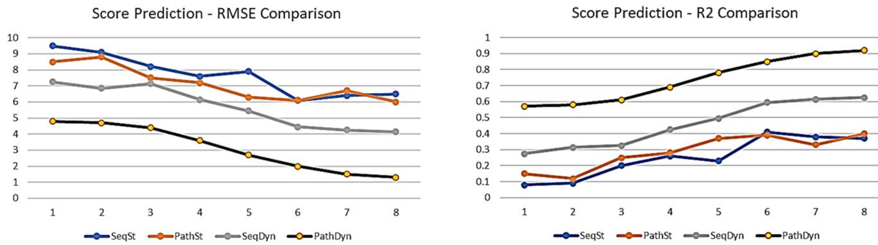

Fig. 5 Comparison of the predicted score on completing LOs and actual score obtained

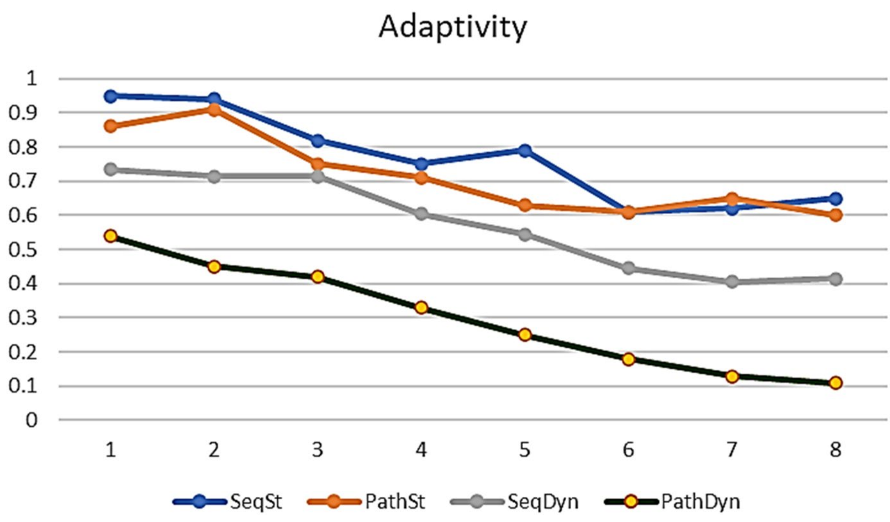

Fig. 6 Similarity between the recommended LO and learner characteristics

Diversity, DIV of the learning path: Ensures that the learning path suits the learner's needs is least likely to repeat (Liu et al., 2018, Meng et al., 2021). More diverse paths need to be generated by the model, and different classes of learners should get different learning paths also.

$$\mathrm{DIV} = \frac{\sum_{i \neq j}^{N} 1 - \left(\frac{p_i \cap p_j}{L}\right)}{N - 1}, \tag{7}$$

where DIV represents the diversity of the learning paths recommended and P represents each path. L is the length of a path, and the model suggests n number of total paths. Figure 7 shows the path diversity obtained for each model.

Participants are asked to rank the recommendations on a scale of 1 to 5 (1—poor, 2—average, 3—good, 4—very good, 5—excellent). This feedback from the learners is used to calculate the learner's satisfaction. The total recommendations made in the entire learning process comprises to 4285 LO ratings. From Fig. 8, the graph's X-axis denotes the rating share in percentage and the Y axis represents the rating level 1–5. We can observe the rating share of each model. For example, the 27% of LOs rated 1 are recommended by SeqSt, 29.8% by SeqDyn, 27% by PathSt, and 16.2% by PathDyn. Similarly, considering the 5 rated LOs, the rating share is observed as SeqSt -17.39% SeqDyn-24.63% PathSt—26.08% PathDyn-31.88%.

Again, suppose the rating levels are greater than two are considered. In that case, the total ratings obtained by each model are shown in Fig. 9. The values are normalized on a scale of 0–100.

Discussion

This section discusses the results obtained from systematic experimentation to answer RQ1 and RQ2. The RQ1 is a decision problem: *Can we accurately predict learning duration and expected score during the learning path recommendation process?* From the controlled experimentation performed, we are getting a positive answer for the RQ1. The learners are asked to search and learn various topics available in the repository in the process. RMSE and R2 measures are used to evaluate the regression performance of the model. Figure 4 shows the comparison between the learner's predicted and actual time taken through 8 different iterations.

Here we can observe that the proposed model, PathDyn, steadily improves prediction accuracy. The average RMSE values are almost decreasing in subsequent iterations for the proposed model. But the baseline models where the static characteristics are alone used for predicting LOs (SeqSt and PathSt) have comparatively higher error rates in predicting the duration of the learning process. The heterogeneous set of learners cannot follow the predictions by the system that uses static learner parameters (Tseng et al., 2008). In contrast, the proposed model uses the time taken log of the learners as one parameter to compute their ability. The time factor is also considered while updating the difficulty of each LO (Meng et al., 2021). So, the model can learn better with an increasing number of iterations and interactions, reducing the error and improving the predictions.

Similarly, a score log is also used while computing the learner's ability. The score is also considered as a parameter in the computation of the difficulty of each LOs. From Fig. 5, it is evident that the difference in the predicted and observed values of the score is decreasing with an increase in iterations. Also, the R2 measure is showing promising results with the proposed model. The baseline model does not offer any pattern in the relation between the learners' observed and predicted score measures. We have compared the two models which function on dynamic parameters (SeqDyn and PathDyn), analyzing the real-time implicit learner data. From the result analysis, PathDyn better predicts the score and learning duration. Here, the advantage of the knowledge-based sequencing of LOs forming path helped reduce the error rates (Tarus et al., 2018; Wu et al., 2020).

The RQ2 tries to evaluate the adaptive and dynamic nature of the proposed model, and the question is: *What are the factors contributing towards generating adaptive and diverse learning paths in an e-learning environment?* The answer to this question is the dynamic parameters. From Fig. 6, it is understood that the adaptivity of the model increases as the interactions increase. With more iterations, the system can log more information about the learners, and based on that, more adaptive LOs can be recommended. The diversity factor is also used to measure recommendations' adaptive and dynamic behavior. From Figs. 6 and 7, we can observe that the adaptivity and diversity of the recommended learning materials are better with the models designed with the addition of dynamic parameters too. The static parameters help find the learning object that suits the learner's preferences. But, naturally, the learner's performance varies as the learning process progresses. The dynamic parameters are needed to be analyzed to recommend LO adaptively to this change in performance. The dynamic parameters

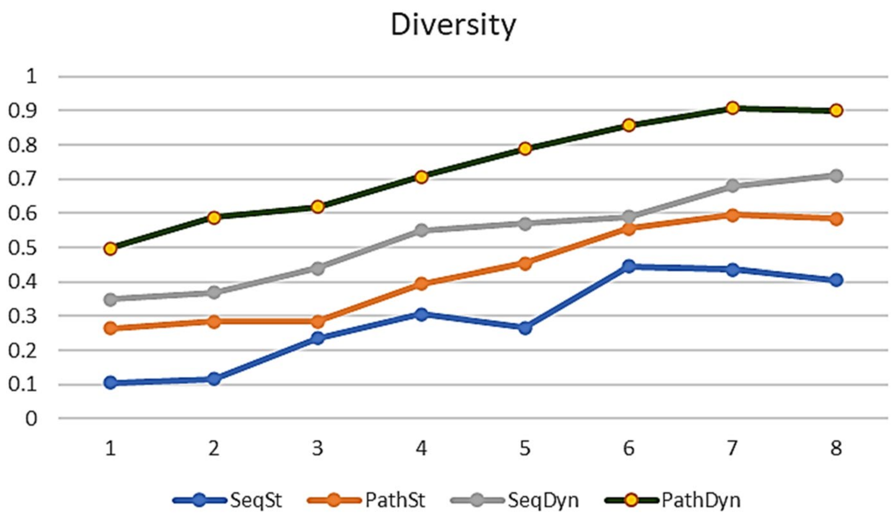

Fig. 7 Diversity in recommended LO for different learners

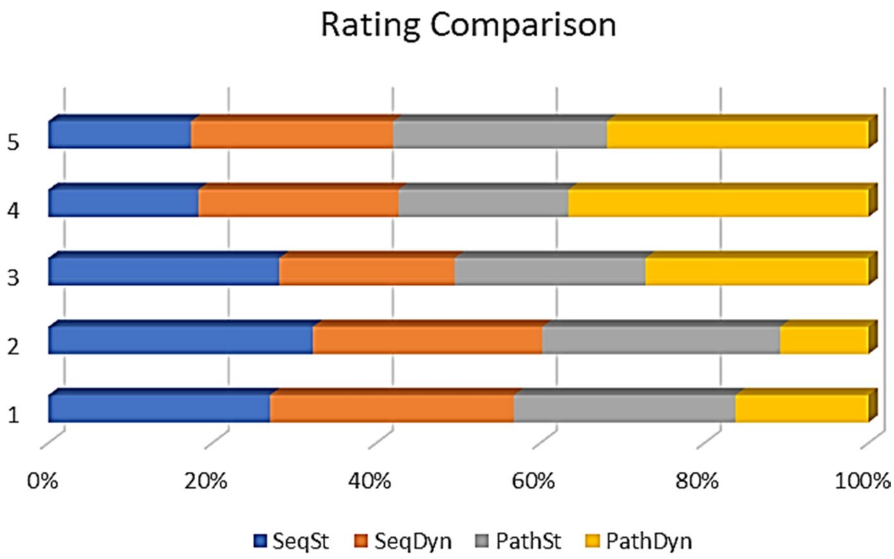

Fig. 8 Rating share of each model

considered in this study are the real-time learning duration, score obtained, number of attempts taken to study an LO, and the list of learned materials. These parameters are implicitly logged and analyzed in real-time scenarios. The implicitly collected learner log helps in adapting to the learner's changing needs, and the integration of implicit and explicit parameters makes the recommendations more adaptive (Gomede et al.,

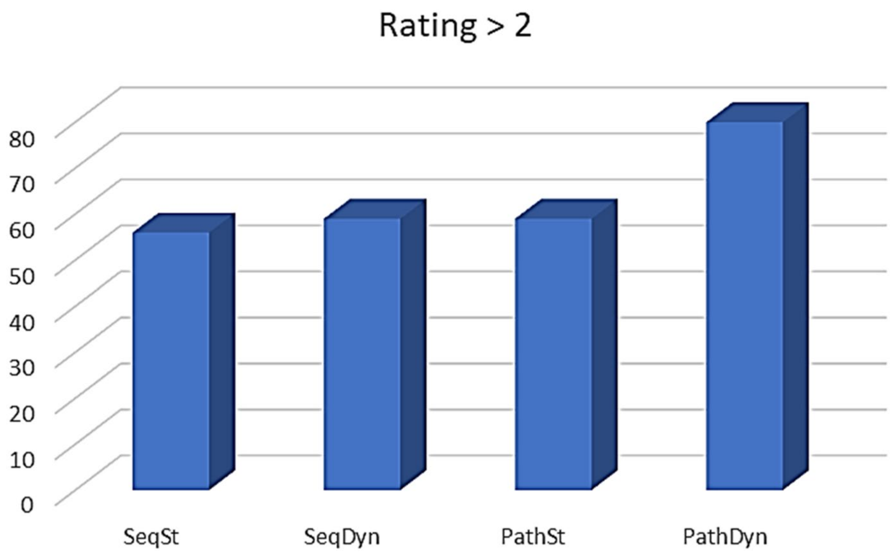

Fig. 9 Number of Ratings obtained for the satisfactory level > 2, normalized to 0–100 scale

2021; Xie et al., 2019). Thus, we can see that from the results of the experimentations, RQ1 and RQ2 are answered.

The learner satisfaction analysis is also done as discussed in Sect. 6. From the results, it is evident that the learners show more inclination toward the PathSyn generated LOs. The count of LOs recommended by the PathSyn model, which is rated with 3,4 and 5 levels, is more than all other models. Since the ontology model is used for representing the domain knowledge and involved in selecting LOs the cold-start issue is handled by the model (Joy et al., 2019, 2021).

Limitations

Even though the model generates a learning path suitable for the learner's style and ability, there are few limitations also. The limitations observed in the proposed model for learning path generation are as follows:

- The linkage between the LOs is made based on the proficiency of the subject experts only. The initial difficulty level and time expected to learn an LO are marked by the experts, so in the cold-start phase, these values are not personalized.
- The performance of learning path models cannot be compared as the behavior of learners changes dynamically. The data cannot be reused either.
- We have used simulated data for early training of the system to find the constraint parameters.
- The number of topics and LOs are limited in this study.
- The participants are assumed to have learned all the previous topics.

- Few learners made errors in data entry, and we needed to discard their records completely

Conclusion and future work

This research article presents a model for generating learning paths suitable for learners' preferences and abilities. The present study tries to solve the learning path adaptation problem by exploring the knowledge relationship between the learning resources. The learner, learning materials, and learner log data are represented as classes in the ontology model. The data objects of these classes are comprised of static and dynamic parameters defining each class. The study focuses on analyzing the dynamic parameters such as the time taken for learning, the obtained score, the number of repeated attempts, and the learning resource rating. Based on the analysis, the ability of the learner to learn a particular learning material is computed in real-time for every recommendation. Also, the difficulty level of the learning material is adjusted based on the learner's performance and LO rating. The LO is selected in two steps 1. Generate a list of LO using Collaborative filtering exploiting the similarity of learners 2. From this list, select the LOs that best match the learner's current ability. The LOs are sequenced using a concept graph for the generating the learning path. As the model trust more on the implicit feedback represented as the dynamic parameters, the ability to predict learning duration and expected score is progressing as with the learning process. Thus, recommending more adaptive and diverse learning paths. The comparison results with three existing models show a better performance from the proposed approach with an average accuracy rise of 30% in learning path prediction based on the expected duration of learning 27.8% in expected score prediction with the second-best performing model. The rating levels indicated the enhancement of learner satisfaction and experience with a rise of 25.5% when comparing the rating share with the second-best model. Ninety-six undergraduate Computer Science and Engineering students participated in the study which involved 623 learning materials from C Programming, Data Structures, and Data Mining courses.

Even the proposed model shows progressive results, the progression is slower. More experiments are planned with parameters such as students' cognitive ability, engagement, and procrastination to recommend learning paths to learners (Agnihotri et al., 2020; Farrell et al, 2019; Raj et al., 2021; Shimada et al., 2018). Also, we have plans to incorporate the model with the existing learning management system for a better e-learning experience. In the current situation of sudden shifting between on-campus and online modes of education caused by COVID-19 pandemic, a more personalized LMS will benefit toward the student engagement, performance and satisfaction (Clark et al., 2021; Patil & Naqvi, 2020).

Data availability The datasets generated during and/or analysed during the current study are not publicly available due to privacy reasons but are available from the corresponding author on reasonable request.

Declarations

Ethical statements

I hereby declare that this manuscript is the result of my independent creation under the reviewers' comments. There is no conflict of interests. Except for the quoted contents, this manuscript does not contain any research achievements that have been published or written by other individuals or groups. The legal responsibility of this statement shall be borne by the authors of this manuscript.

References

Agnihotri, L., Baker, R., & Stalzer, S. (2020). A Procrastination Index for Online Learning Based on Assignment Start Time. In Educational Data Mining.

Armstrong, P. (2016). Bloom's taxonomy. Vanderbilt University Center for Teaching.

Belacel, N., Durand, G., & Laplante, F. (2014). A Binary Integer Programming Model for Global Optimization of Learning Path Discovery. In Educational Data Mining (Workshops).

Benmesbah, O., Lamia, M., & Hafidi, M. (2021a). An improved constrained learning path adaptation problem based on genetic algorithm. *Interactive Learning Environments*. https://doi.org/10.1080/10494820.2021.1937659

Benmesbah, O., Lamia, M., & Hafidi, M. (2021b). An enhanced genetic algorithm for solving learning path adaptation problem. *Education and Information Technologies*. https://doi.org/10.1080/10494820.2021.1937659

Cai, D., Zhang, Y., & Dai, B. (2019). Learning path recommendation based on knowledge tracing model and reinforcement learning. In 2019 IEEE 5th international conference on computer and communications (ICCC) (pp. 1881–1885). IEEE.

Chen, L. H. (2011). Enhancement of student learning performance using personalized diagnosis and remedial learning system. *Computers & Education, 56*(1), 289–299.

Chen, C. P., & Zhang, C. Y. (2014). Data-intensive applications, challenges, techniques, and technologies: A survey on Big Data. *Information Sciences, 275*, 314–347.

Chen, W., Niu, Z., Zhao, X., & Li, Y. (2014). A hybrid recommendation algorithm adapted in e-learning environments. *World Wide Web, 17*(2), 271–284.

Christudas, B. C. L., Kirubakaran, E., & Thangaiah, P. R. J. (2018). An evolutionary approach for personalization of content delivery in e-learning systems based on learner behavior forcing compatibility of learning materials. *Telematics and Informatics, 35*, 520–533.

Ciloglugil, B., & Inceoglu, M. M. (2018). A learner ontology based on learning style models for adaptive e-learning. *International conference on computational science and its applications* (pp. 199–212). Springer.

Clark, R. M., Kaw, A. K., & Braga Gomes, R. (2021). Adaptive learning: Helpful to the flipped classroom in the online environment of COVID? *Computer Applications in Engineering Education*. https://doi.org/10.1002/cae.22470

Cun-Ling, B., De-Liang, W., Shi-Yu, L., Wei-Gang, L., & Jun-Yu, D. (2019). Adaptive learning path recommendation based on graph theory and an improved immune algorithm. KSII Transactions on Internet & Information Systems.

Deng, Y., Huang, D., & Chung, C.-J. (2017). Thoth lab: A personalized learning framework for cs hands-on projects. In Proceedings of the 2017 ACM SIGCSE technical symposium on computer science education (pp. 706). ACM.

Dharani, B., & Geetha, T. V. (2013, July). Adaptive learning path generation using colored Petri nets based on behavioral aspects. In *2013 International conference on recent trends in information technology (ICRTIT)* (pp. 459–465). IEEE.

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.

Essalmi, F., Ayed, L. J. B., Jemni, M., & Graf, S. (2010). A fully personalization strategy of E-learning scenarios. *Computers in Human Behavior*, *26*(4), 581–591.

Farrell, C. C., Markham, C., & Deegan, C. (2019). Real time detection and analysis of facial features to measure student engagement with learning objects. IMVIP 2019: Irish Machine Vision & Image Processing

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

George, G., & Lal, A. M. (2019). Review of ontology-based recommender systems in e-learning. *Computers & Education, 142*, 103642.

Gomede, E., de Barros, R. M., & de Souza Mendes, L. (2021). Deep auto encoders to adaptive E-learning recommender system. *Computers and Education: Artificial Intelligence, 2*, 100009.

Hwang, G. J., Sung, H. Y., Chang, S. C., & Huang, X. C. (2020a). A fuzzy expert system-based adaptive learning approach to improving students' learning performances by considering affective and cognitive factors. *Computers and Education: Artificial Intelligence, 1*, 100003.

Hwang, G. J., Xie, H., Wah, B. W., & Gašević, D. (2020b). Vision, challenges, roles and research issues of artificial intelligence in education. *Computers & Education: Artificial Intelligence, 1*, 100001.

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*, 3–13. https://doi.org/10.1007/s40595-015-0049-6

Jdidou, Y., Aammou, S., & Khaldi, M. (2021). Adapt learning path by recommending problems to struggling learners. *International Journal of Emerging Technologies in Learning, 16*(20), 163.

Joy, J., Raj, N. S. & Renumol V. G. (2019). An ontology model for content recommendation in personalized learning environment. In Proceedings of the second international conference on data science, e-learning and information systems (pp. 1–6).

Joy, J., Raj, N. S., & Renumol, V. G. (2021). Ontology-based E-learning content recommender system for addressing the pure cold-start problem. *ACM Journal of Data and Information Quality, 13*(3), 1–27.

Li, W., & Zhang, L. (2019). Personalized learning path generation based on network embedding and learning effects. In 2019 IEEE 10th international conference on software engineering and service science (ICSESS) (pp. 316–319). IEEE.

Liu, Z., Li, H., Song, W., Kong, X., Li, H., & Zhang, J. (2018). Research on mixed recommendation method of learning resources based on bipartite network. *e-Educ. Res.*, *39*(8), 85–90.

de Marcos, L., Martínez, J. J., & Gutiérrez, J. A. (2008). Swarm intelligence in e-learning: a learning object sequencing agent based on competencies. In Proceedings of the 10th annual conference on Genetic and evolutionary computation (pp. 17–24).

Meng, L., Zhang, W., Chu, Y., & Zhang, M. (2021). LD–LP generation of personalized learning path based on learning diagnosis. *IEEE Transactions on Learning Technologies, 14*(1), 122–128.

Nabizadeh, A. H., Gonçalves, D., Gama, S., Jorge, J., & Rafsanjani, H. N. (2020). Adaptive learning path recommender approach using auxiliary learning objects. *Computers & Education, 147*, 103777.

Nabizadeh, A. H., Jorge, A. M., & Leal, J. P. (2018). Estimating time and score uncertainty in generating successful learning paths under time constraints. *Expert Systems*. https://doi.org/10.1111/exsy.12351

Nabizadeh, A. H., Mário Jorge, A., & Paulo Leal, J. (2017). Rutico: Recommending successful learning paths under time constraints. *Adjunct Publication of the 25th Conference on User Modeling Adaptation and Personalization*. https://doi.org/10.1145/3099023.3099035

Niknam, M., & Thulasiraman, P. (2020). Lpr: A bio-inspired intelligent learning path recommendation system based on meaningful learning theory. *Education and Information Technologies*. https://doi.org/10.1007/s10639-020-10133-3

Patil, D., & Naqvi, W. M. (2020). COVID-19 and education system: Impact of current pandemic on adaptive learning strategies in medical education system. *International Journal of Research in Pharmaceutical Sciences*. https://doi.org/10.26452/ijrps.v11iSPL1.2736

Plass, J. L., & Pawar, S. (2020). Toward a taxonomy of adaptivity for learning. *Journal of Research on Technology in Education, 52*(3), 275–300.

Raj, N. S., & Renumol, V. G. (2018). Architecture of an adaptive personalized learning environment (aple) for content recommendation. In Proceedings of the 2nd International Conference on Digital Technology in Education (pp. 17–22).

Raj, N. S., & Renumol, V. G. (2019). A rule-based approach for adaptive content recommendation in a personalized learning environment: An experimental analysis. In 2019 IEEE Tenth International Conference on Technology for Education (T4E) (pp. 138–141). IEEE.

Raj, N. S., Prasad, S., Harish, P., Boban, M., & Cheriyedath, N. (2021). Early prediction of at-risk students in a virtual learning environment using deep learning techniques. *International Conference on Human-Computer Interaction* (pp. 110–120). Springer.

Raj, N. S., & Renumol, V. G. (2021). A systematic literature review on adaptive content recommenders in personalized learning environments from 2015 to 2020. *Journal of Computers in Education*. https://doi.org/10.1007/s40692-021-00199-4

Ramos, D. B., Ramos, I. M. M., Gasparini, I., & de Oliveira, E. H. T. (2021). A new learning path model for E-learning systems. *International Journal of Distance Education Technologies (IJDET), 19*(2), 20–40.

Risk, U. (2002). Draft standard for learning object metadata. *IEEE Standard*. https://doi.org/10.13140/RG.2.2.26170.52166

Sachan, D., & Saroha, K. (2022). A review of adaptive and intelligent online learning systems. *ICT Analysis and Applications*. https://doi.org/10.1007/978-981-16-5655-2_24

Segal, A., Gal, K., Shani, G., & Shapira, B. (2019). A difficulty ranking approach to personalization in e-learning. *International Journal of Human-Computer Studies, 130*, 261–272.

Shi, D., Wang, T., Xing, H., & Xu, H. (2020). A learning path recommendation model based on a multidimensional knowledge graph framework for e-learning. *Knowledge-Based Systems, 195*, 105618.

Shimada, A., Konomi, S. I., & Ogata, H. (2018). Real-time learning analytics system for improvement of on-site lectures. *Interactive Technology and Smart Education*. https://doi.org/10.1108/ITSE-05-2018-0026

Son, N. T., Jaafar, J., Aziz, I. A., & Anh, B. N. (2021). Meta-heuristic algorithms for learning path recommender at MOOC. *IEEE Access, 9*, 59093–59107.

Sosniak, L. A. (1994). Bloom's taxonomy. In L. W. Anderson (Ed.), *Vanderbilt center for teaching*. University Chicago Press.

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.

Tarus, J. K., Niu, Z., & Yousif, A. (2017). A hybrid knowledge-based recommender system for e-learning based on ontology and sequential pattern mining. *Future Generation Computer Systems, 72*, 37–48.

Tseng, J. C., Chu, H. C., Hwang, G. J., & Tsai, C. C. (2008). Development of an adaptive learning system with two sources of personalization information. *Computers & Education, 51*(2), 776–786.

Vanitha, V., Krishnan, P., & Elakkiya, R. (2019). Collaborative optimization algorithm for learning path construction in e-learning. *Computers & Electrical Engineering, 77*, 325–338.

Wang, F., Zhang, L., Chen, X., Wang, Z., & Xu, X. (2021). A personalized self-learning system based on knowledge graph and differential evolution algorithm. *Concurrency and Computation: Practice and Experience, 2021*, e6190.

Wiley, D. A. (2002). *The instructional use of learning objects* (Vol. 1). Agency for instructional technology.

Wu, L., Liu, Q., Zhou, W., Mao, G., Huang, J., & Huang, H. (2020). A semantic web-based recommendation framework of educational resources in E-learning. *Technology, Knowledge and Learning, 25*(4), 811–833.

Xie, H., Zou, D., Zhang, R., Wang, M., & Kwan, R. (2019). Personalized word learning for university students: A profile-based method for e-learning systems. *Journal of Computing in Higher Education, 31*(2), 273–289.

Zhou, Y., Huang, C., Hu, Q., Zhu, J., & Tang, Y. (2018). Personalized learning full-path recommendation model based on LSTM neural networks. *Information Sciences, 444*, 135–152.

Zhu, H., Tian, F., Wu, K., Shah, N., Chen, Y., Ni, Y., Zhang, X., Chao, K., & Zheng, Q. (2018). A multi-constraint learning path recommendation algorithm based on knowledge map. *Knowledge-Based Systems, 143*, 102–114.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Nisha S. Raj is a Ph.D. Scholar in the Division of Information Technology, School of Engineering, Cochin University of Science and Technology, India. Her research interests include applications of EDM/LA in MOOCs, designing Personalized Learning Environments, and Recommender Systems. She is an activist in the field of novel practices in e-learning. Towards the same, she has delivered workshops and talks in Innovative Tools and Practises in elearning ecologies.

V. G. Renumol is a Professor in Information Technology, School of Engineering, Cochin University of Science and Technology, India. She secured her doctoral degree from IIT, Madras, India, and Post-Doctoral degree from IIT, Bombay, India. Her research interests include Computing Education, Cognitive Psychology, Personalised Learning, Educational Technology, Special Education. She possesses several national and international publications as conference proceedings and journals.

Authors and Affiliations

Nisha S. Raj[1] · **V. G. Renumol**[1]

[1] Division of Information Technology, School of Engineering, Cochin University of Science and Technology, Kochi, Kerala, India