Personalized Learning Path Delivery in Web based Educational Systems using a Graph Theory based Approach

C. Beulah Christalin Latha[1], Dr. E. Kirubakaran[2]

[1]Assistant Professor, Department of Computer Applications, Karunya University, Coimbatore, Tamilnadu, India.
[2]Additional General Manager, Bharat Heavy Electricals Ltd., Trichy, Tamilnadu, India
beulsonline@gmail.com

Abstract. Learning is the process of acquiring knowledge or skill. It is the process of filtering, storing and organizing information in our brains. E-learning is defined as an innovative approach for delivering well designed, learner centered, interactive and facilitated learning environment to anyone, at any place and at any time by exploiting the potential of digital learning resources and technologies well suited for open, flexible and pervasive learning environments. The efficiency of learning depends upon various factors including the way in which the knowledge is dissipated to the learner. Each learner learns in a unique way. Learning proves to be more effective if the teaching process becomes compatible with the characteristics of the learner such as his learning style, his learning goal and his learning need. In classroom teaching, the teacher takes care of the characteristics of the learner and can train the learner in the appropriate manner. In web based educational systems, the absence of a teacher or trainer becomes a bottleneck in delivering contents in an appropriate manner to the learner. Intelligent tutoring systems provide solution to this problem by tailoring the delivery of contents that suits the characteristics of the learner. These systems treat learners as unique individuals and deliver the contents in a variety of ways. Intelligent tutoring systems thus provide various means of providing an efficient learning including adaptation or personalization. Personalization is the process of making a generalized content specific to the needs and traits of the user. Personalization in e-learning is tailoring learning materials or contents according the learning style of the learner, learner's profile, learner's interests, previous knowledge level, learner's goal and pedagogical method. It provides a way to alter the one size fits all approach that is generally followed in traditional web based education systems. This paper proposes a novel way of recommending a personalized learning path to a user using a graph theory based approach in web-based learning systems in order to make the learning process effective.

[C. Beulah Christalin Latha, E. Kirubakaran. **Personalized Learning Path Delivery in Web based Educational Systems using a Graph Theory based Approach.** *J Am Sci* 2013; 9(12s):57-67]. (ISSN: 1545-1003). http://www.jofamericanscience.org. 7

Keywords: Intelligent Tutoring Systems, Personalization, Pedagogical Issues, Learning Path, Graph Theory

1. Introduction

1.1. Web based Educational Systems

Web-based educational systems can be categorized under two main categories. The first category is the category of learning management systems like Moodle, Docebo that provide a repository of learning content and the ways to manage and monitor the learning process. The other category of educational systems is called as Intelligent Tutoring systems or Adaptive Educational Hypermedia (AEH). Intelligent tutoring systems provide various means of providing an efficient learning including adaptation or personalization. Learning management systems provide very limited personalization capabilities whereas intelligent tutoring systems strive to overcome this hurdle and play the role of a human tutor for assisting the learner in the learning process. Web based learning environments provide classroom independence, platform independence and also an interactive learning environment. Automatic tailoring of the courses according to the needs of the learner plays a crucial role in intelligent tutoring systems. The variations in the characteristics of learners make the conventional e-learning course that offers a one size fits all approach fail to meet the expectations of all learners. Numerous research works emphasize that personalization is a critical issue in e-learning in order to provide effective and efficient learning.

1.2. Personalization

Personalization is the process of making a generalized content specific to the needs and traits of the user. Personalization or adaptation increases the effectiveness of web based applications. Commercial web applications such as Amazon.com provide personalization in the form of recommendation. Data mining and business intelligence are used for the recommendation process in most of the cases. E-learning platforms can also provide personalization concurring to the needs and styles of the learner which can prove to be an effective step in providing an efficient training in the absence of a human trainer. Educational systems that adapt themselves to the needs of the learner are commonly known as Adaptive Educational Hypermedia Systems (AEHS). Adaptive educational hypermedia systems have been the origin for personalized learning systems which are used for providing personalized learning experiences to an individual learner.

Personalization in e-learning is tailoring learning materials or contents according the learning style of the learner, learner's profile, learner's interests, previous knowledge level, learner's goal and pedagogical method. Personalization plays a crucial role in e-learning because in systems with no personalization follow a one size fits all approach. They provide a static learning experience regardless of the characteristics of the learner. Such systems cannot meet the expected goals of the learner. Personalization is crucial in web-based learning systems because the learner's population is characterized by considerable heterogeneity with respect to age, background knowledge, experiences, motivations and goals and the learners should take the main responsibility of self-learning. Moreover, web-based educational systems provide immense amount of information that leads to information overload. They deliver contents in the same pace for all the learners which may not suit the learning speed of every learner. On the other hand, personalized learning systems take into account the characteristics of the learner, the learning speed and their goals. The current research focuses on the learner's goal, background knowledge and the learning style.

1.3. Learning Path

A learning path is a path that starts from the initial knowledge of the learner and ends at the target knowledge. It fills the gap between the initial knowledge and the target knowledge by sequencing of the learning objects. The learning path can be tailored in a way that suits each learner based on his learning styles, learning goals and the needs of the learner. Such a path is known as a personalized learning path for a specific learner. A personalized learning path presents contents in a way that is better comprehensible by the learner, and it might motivate the learner in the learning process and provides better learning efficiency and learner satisfaction. Personalized learning paths are constructed by sequencing the contents or learning objects in a way that suits the learning styles, learning goals and the needs of the learner thereby facilitating the learner to gain the target knowledge in a more efficient way. In the proposed work, the learning objects are selected according to the learning styles of the learner based on the Felder-Silverman learning style model and, are sequenced. The difficulty level and interactivity level of the learner are tuned according to the current learner's feedback and similar learners' feedbacks to adjust to the concept drifts. Concept drift refers to the changes in the learners' state of knowledge during the course of learning.

2. Literature Survey

M.-J. Huang et al. in their paper *Constructing a Personalized e-Learning System based on Genetic Algorithm and Case-based Reasoning Approach* propose a personalization approach based on the evolvement technique through computerized adaptive testing (CAT). Then the genetic algorithm (GA) and case-based reasoning (CBR) are used to construct an optimal path for each learner. The personalized e-learning system based on mastery learning (PLS-ML) is the implemented version of the general framework of the genetic-based and case-based reasoning system. Based on the results of the assessment, if the learner needs to improve in the learned concept, the system will recommend an appropriate personalized curriculum sequencing suggestion using genetic algorithm. If the results are satisfactory, the learner can continue with the additional interesting topics or extension materials in enrichment activities. This provides a way for the learners to relearn the same concepts through different curriculum sequencing and corrective activities. The estimation of difficulty parameters for the curriculum design procedure is based on a statistics based method derived from Computer Adaptive Testing theory through a painstaking test process to determine the difficulty parameters. The curriculum modeling and test items designed by experts were analyzed by the item response theory in CAT using the statistics based BILOG program to obtain the appropriate difficulty parameters [15].

The relation degree between curriculums is calculated using the Vector Space Model (VSM). Each curriculum is represented as a vector and their relevance to the queries submitted by the user is measured through appropriate matching functions. The information retrieval is performed through the extraction of the term by the document matrix through pre-processing, normalizing and indexing. The relevance of a term for the representation of content in a curriculum is weighed by the term frequency and inverse document frequency. The concept relation degree between two learning objects is calculated using the cosine measure. The fitness function is determined by considering the difficulty parameter of the curriculum and the concept relation degrees. Genetic algorithm constructs the learning path by considering the curriculum with a score greater than a certain threshold value for a particular learner. The stop criterion for genetic algorithm is set to be 100 generations for generating a satisfied learning path for a learner. The results indicate that the generated learning path recommends an appropriate learning path to the learner while simultaneously considering the difficult parameters and the relation degree of the curriculum [15].

Marwah Alian and Riad Jabri in their paper *A Shortest Adaptive Learning Path in eLearning Systems: Mathematical View* develops an algorithm for improving the ability to select the dynamically an appropriate learning object (LO) for a specific learner using Eliminating and Optimized Solution (EOS) and selecting a learning path that costs least time and effort. The e-learning system has been modeled as a directed graph where each node represents a learning object. Each learning object may contain one object, one concept, one image or an audio session. Two nodes are connected if there exist a dependency relation, such that one node is a prerequisite to the other. The algorithm aims at designing adaptivity and creating a learning path that is tailored for the learner's needs and combining both of them [14].

In EOS approach, the suitability of a learning object is evaluated by using historical information about the learner, the learning object and the learning context. The attributes of a learning object can be classified into two groups: eliminating attributes and selecting attributes. The eliminating attributes are used in the filtering phase to eliminate the learning objects that do not match the learner's needs. Pedagogical objective (keyword), language, environment condition and financial cost are considered to be the eliminating attributes. The selecting attributes are used in the selection phase where each learning object is assigned a value based on the comparison of the selecting attributes with the learner's characteristics. A suitability function for each learning object is formed for selecting the candidate learning objects. An importance analysis of the features surrounding each LO or context is performed. It is reflected by assigning weight (W) for each attribute (feature) of the learning objects in a given context. A degree of match between these attributes and the requirement is performed which is represented by a value between 0 and 1. Time, presentation type and reading level have been chosen as selecting attributes for the LO [14].

The resulting set of learning objects enter the optimization phase in which the learning objects are assigned a value computed based on the history of using learning objects by previous learners. The following factors are considered for optimization: learner style, learner level and learner academic achievement. The knowledge space (ontology plane) and the media space (content space) are merged to form a directed acyclic graph (DAG) of LOs inheriting relations from both spaces. This graph contains all possible paths. From the DAG of all possible paths, a subgraph that is relevant to the learner is constructed. The sub graph is augmented with weights that represent the suitability of learning objects for the learner. An adaptive path that is as suitable and as shortest as possible is suggested for the learner using a shortest path algorithm. The results of the proposed approach are found to be competent with the results obtained by the experts. The overall performance and adaptation is found to be improved [14].

In Kazuya Seki, Tatsunori Matsui, and Toshio Okamoto's paper *An Adaptive Sequencing Method of the Learning Objects for the e-Learning Environment*, a learning ecological model called the Learning Environment Model (LEM) that represents a comprehensive learning environment is proposed in which a learning object is selected by each learner based on the learning that is most suitable to the individual learner. The learning scenario is constructed by sequencing the learning objects based on the learning necessity, the learning history information, and the curriculum information of the object of learning, according to the characteristics of the learning [13].

The learning unit, composing of the learning content to achieve the learning goal and the learning configuration suited to the achievement (learning procedure, method, and tool), is considered as the learning object. The learning objects are managed using the learning object metadata, called *cell*. The learning objects are developed in various representation formats. The adaptive sequencing of the LOs is performed using the LO metadata, the learning requirement information, the learning history information and curriculum information. Multidimensional information and multiple evaluation viewpoints are used for the adaptive sequencing. The evaluation viewpoint for sequencing has been formulated as a multi-objective optimization problem and has been solved using genetic algorithm [13].

Four kinds of evaluation points are used for the sequencing of learning objects and the evaluation points are initially assigned with equal weights and the weights are modified during the course of learning based on the real-time intention of the learner. The relationship among unit items, the presupposition of the unit item, the difficulty of the unit item and the adequateness of learning action are considered to be the four evaluation points. Learning history information is used to determine the overlap of learning materials. The overlap calculated in this way is multiplied by the adaptability function as the weight. The weighting avoids a situation in which a learning object with learning similar to the already understood unit item or learning action is assigned to an individual. The experimental results indicate that sequencing technique considered in this study can generate a learning object sequence that is comparable to the result by the experts [13].

Andrea Sterbini, Marco Temperini in their paper *Adaptive Construction and Delivery of Web based Learning Paths* present a web-based learning environment, *LeComps*, that automates the construction of adaptive personalized learning paths tailored over learning goals, learner's knowledge and individual learning styles. A constraint logic based engine is used to perform the course construction and to support the teacher in the definition of the learning objects to be used. Learning objective templates (LOs) are used to support the management of knowledge throughout the framework. The system builds a personalized course as a selection of Learning Components (LCs). The selection is based on the Target Knowledge (TK) and on the individual Starting Knowledge (SK). The set of LCs (a course configuration) is made linear in lessons, according to appropriate constraints on lesson length. Then, the learning path is administered, by presenting the material in different ways, depending on the personal learner's learning styles based on the Felder-Silverman learning style model. The personalized course can be rebuilt on the fly to adapt to changes in the learner's state of knowledge [1].

The basic elements in the framework are the learning components (LCs), that convey and specify learning content in such a way to make it available for automated treatment. The specification part of LCs is obtained through instantiation of *Learning Objective* (LO) templates that describes a certain skill with respect to a certain concept in such a way to be testable after learning processes. A learning object (LO) is expressed as a predicate ranging over the concepts involved, the cognitive level and the learning context. The learner's knowledge state (LKS) is represented as a set of pairs of the LO presently owned and the level of its certainty. The learning styles based on Felder-Silverman model has been represented as a 4-tuple of couples. Once the learner gets enrolled, based on his input values, a course is configured through a graphical interface. The pool of LCs are modeled as a directed acyclic layered graph, where both LC and LOs are vertices and a LC is connected to an LO if the LO appears either in its Acquired Knowledge (AK) or in its Required Knowledge (RK). The recommended learning components are highlighted by active nodes in a graph and the learner can learn those LCs. Virtual components or VLCs are used to capture the knowledge gap among similar learning objects and to make explicit the additional effort that the learner should take to bridge the corresponding gap without help. An evaluation at the end of each lesson helps the system to evaluate the degree of certainty of possession of each LO by the learner. The Learner's Learning Style evaluation can be updated by taking into consideration the material to which the learner was exposed and the answers to tests [1].

The graph can be mapped to an AND-OR graph, where LCs are AND vertices and LOs are OR-vertices. A branch-and-bound optimization algorithm is applied to the set of constraints describing the AND-OR graph, while minimizing the course evaluation function. Different optimization functions are used to obtain a course that could be either compact (minimizing the total effort of the selected LCs), or smooth (minimizing the effort's variance) or easy (minimizing the max effort present). The problem is mapped to the construction of a topologic ordering of the learning paths produced which makes the learning process easier for the learner. The feedback obtained by the participation of the learner is used to update the model. The system reduces the load on the teacher also by automatically sequencing the contents [1].

Nguyen Viet Anh, Nguyen Viet Ha, Ho Si Dam in their paper *Constructing a Bayesian Belief Network to Generate Learning Path in Adaptive Hypermedia System*, propose an approach to generate a learning path adaptive to each learner using Bayesian belief networks. A shortest path search algorithm is used to evaluate learning objects based on their attributes. The LO attributes are used to build the course structure or knowledge maps for each learner. The content developers can develop their course knowledge maps using a tool that is a directed acyclic graph which includes vertices and direct edges. The vertices represent knowledge units which are constructed by one or more learning objects and the edges represent knowledge unit relationships. The knowledge unit is modeled into a semantic web. It is constructed by one or more learning objects, the learning objects are also characterized by attributes such as pre-requisites, master level, difficulty level, required time, relation, interactive style and skill. Some attributes are supplemented for assets. Content developers help to assign weights for assets when they create the course. The system will automatically generate the best learning for learner which is based on learner's profile as well as knowledge map that had been designed by the teacher or the content developer for learning the course. A shortest path algorithm is used for searching to select learning path based on the attributes of learning objects and a Bayesian belief network has been used to construct a learning [16].

Roberto Pirrone et. al. in their paper *Learning Path Generation by Domain Ontology Transformation* presents an approach to automated learning path generation from a domain ontology via a transformation of the ontology itself in a suitable weighted graph using A* algorithm. The Knowledge Space Theory (KST) has been used to obtain a transformation from the original ontology defined in the OpenCyc knowledge base to a weighted graph where the A* algorithm was applied to determine learning paths. A suitable heuristics has been employed to obtain the arc's weights from the semantics of each relation in the ontology. The arc's weights describe the subjective relevance of each concept with respect to the learner's goals. Finally, a concept map that represents and organizes a set of concepts in a spatial way is used to visualize the path of the learning materials. A three level schema is used for modeling the course topics. At the lowest level, information is aggregated as a set of HTML documents which can represent single learning objects or a composition of them. The intermediate representation is achieved by a concept map that is implemented as a SOM (self organizing map- a type of artificial neural network using unsupervised learning) used to cluster documents in a vector space representation using a measure of the similarity between the documents. Every concept is related to one or many others via a few relations and therefore, the ontology is strongly connected [17].

The ontology-to-graph transformation is performed according to the Knowledge Space Theory formulation of learning path: the learner is characterized by a knowledge state defined as the set of concepts he knows at a certain time and he can move to another state by evaluating what he is ready to learn. The current knowledge state of the user is determined by a test containing a set of questions on the topics. The ontology structure allows to define knowledge states in the same manner as in KST, and transitions between states takes place moving across the navigation relations. The natural language dialog interaction is used to determine the initial state and the goal. The transformation proceeds by mapping the original ontology to a graph and the extraction of the path from the graph [17].

G L A Alvarez & J L Montesinos, in their paper *An Authoring Tool for Learning Object Sequencing* propose the design of a learning object sequencing tool based on SCORM 2004 that can be used with open source learning management systems. It supports sequencing and navigation [9]. Anh Nguyen Viet, Dam Ho Si's paper *Applying Weighted Learning Object to build Adaptive Course in E-learning* generates an adaptive learning path for each learner based on learner's profile. A set of weights has been supplemented for each learning object which is basic for selecting learning path process. It also suggests an algorithm for shortest path selection based on time to finish the course. The learner model is built based on the learner's profile and a suitable learning path is generated. The learner model is dynamically updated throughout the course. The content of the course is modeled as a graph with nodes corresponding to a set of learning objects. The learning objects are classified into two categories: mandatory learning objects (MLOs) and secondary learning objects (SLOs). The learning objects are attached with attributes. To select a learning path, a function to evaluate learning object based on value of its attribute is used. The selection of LO is made based on a threshold value for the function. Bayesian belief networks were also used to decide the selection of LO [2].

3. Proposed Work

3.1. Knowledge Representation

The initial knowledge of the learner is represented by an initial knowledge state set K_i. K_i contains the list of the topics known to the learner. The knowledge state of the learner is dynamic and varies while he continues with the course. The topics learnt by the learner are added to the knowledge state of the learner. This representation is based on Knowledge Space Theory (KST) which is a theoretical framework proposed by Doignon and Falmagne in the field of learner modeling [6, 7, 3, 5]. According to the Knowledge Space Theory, a knowledge domain can be decomposed into small learning quanta or items that are called learning objects. A learning object may be one of the learning resources or skills or tasks that are meant to be learnt by the learner in order to learn a concept. Thus, a knowledge domain is a set of learning objects.

Each learner is represented by his knowledge state set K_l in a particular domain. The initial knowledge state K_i contains the list of topics in a particular knowledge domain that the learner is familiar with. Knowledge states are characteristic to a specific learner and a specific knowledge domain. A learner's knowledge state set K_l is dynamic during a course and at the end of the course it contains all the topics required for learning the concept that is the goal of the learner. The final goal state of the learner is represented by K_g. Learning objects are related using pre-requisite relationships and therefore, all possible subsets of learning objects are not knowledge states. The set of feasible knowledge states for a particular domain is known as a Knowledge Space K_s. For any two knowledge states K and K' that is a subset of K_s, $K \cup K' \subset K_s$.

$$\forall\ (K,\ K^{'} \in K_s), K \cup K^{'} \in K_s$$

The empty set $\emptyset$ and complete learning object set $\mathcal{L}$ are also elements of K_s.

$$\emptyset \in K_s, \mathcal{L} \in K_s$$

The application of knowledge space framework for web based learning systems facilitates the definition of personalized learning path to the learners leading the learner from his initial state K_i to the goal state K_g.

In knowledge space theory, pre-requisite relationships are formalized by surmise relations i.e. quasi-orders on a set $\mathcal{L}$ of learning objects. Such a surmise relation $\sqsubseteq$ may be interpreted as $\ell \sqsubseteq \ell'$ if and only if from a correct response to problem domain ℓ' we can surmise a correct response to problem domain ℓ. It can also be defined as mastering a learning object ℓ is a pre-requisite for mastering the learning object ℓ [4].

3.2. Mathematical Model

Albert, Held and Hockemeyer have suggested a mathematical model for the hyperstructure using Dexter Hypertext reference model [8]. Dexter Hypertext Reference model has been suggested as suitable for e-learning by Hend Madhour & Maia Wentland Forte [10, 11]. According to the Dexter model, a hypertext can be considered to consist of a set of components and a set of links between these components. A component consists of a base component and source and destination anchors which are located on the base component. Links are specified by the anchors and the components on which the anchors are located. There exist three sets B, S and D of base components, source anchors and destination anchors. A component $c = \left(b, S_{c,} D_c\right) \in (B, 2^S, 2^D)$ is a triplet formed by a base component $b \in B$, a subset $S_c \sqsubseteq S$ of source anchors and a subset $D_c \sqsubseteq D$ of destination anchors. A set $L \subseteq (CXS)X(CXD)$ where each $l = \left((c,s),(c',d)\right) \in L$ fulfills the condition $s \in S_c$ and $d \in D_{c'}$ is called a set of links. A pair $H = (C,L)$ is called a hypertext. A link $l = \left((c,s_c),(c',d_{c'})\right)$ connects a source anchor s_c located on a component c with a destination anchor $d_{c'}$ on a component c'. A binary relation $\vdash$ represents the linkage between components. For a hypertext $H = (C,L)$, the link relation is defined as $\vdash \subseteq CXC$ on the set C of components. For any component $c, c' \in C$, $c \vdash c'$ if and only if there exists a link $l = \left((c,s),(c',d)\right) \in L$ [4, 12].

$$\forall\, c, c' \in C, c \vdash c' \Leftrightarrow l = \left((c,s),(c',d)\right) \in L$$

3.3 Hypertext Model for Knowledge Representation

The hypertext model can be combined with the knowledge space theory for building a framework for adaptive educational hypermedia systems by applying the concept of prerequisite relationships. Prerequisite relationships can be represented by means of a special link type known as the prerequisite link [4, 11].

A hypertext can be defined by the set $H = (C,L)$ where C is the set of components and L is the set of links. For an arbitrary subset $\subseteq S$, a link $l = \left((c,s),(c',d)\right)$ is called as a prerequisite link P if and only if $s \in P$.

$$L^P = \{l = ((c,s),(c',d)) \in L | s \in P\}$$

where L^P is the set of all links of type P. A binary relation $\vdash^P$ is defined as, $\vdash^P \subseteq \vdash \subseteq CXC$ and is known as a P-relation [4].

$$\forall\, c, c' \in C, c \vdash^P c' \Leftrightarrow l = \left((c,s),(c',d)\right) \in L$$

A prerequisite link from a component c to a component c' means that c' is a prerequisite for c. For educational adaptive hypermedia systems, the surmise relation can be used for representing the knowledge of the learner whereas the prerequisite relation can be used for referring the appropriate learning resources to be disseminated to the learner.

3.4 Proposed Framework

The current research is a smart curriculum design that disseminates knowledge to the learners in a personalized way by adapting the contents to the learner's learning style, goal and also the learner's background knowledge. It is an adaptive educational hypermedia system that provides adaptation in content presentation. The system provides a personalized learning path using the learning style of the learner and the goals of the learner. There is a user interface serves a means of collection of input information from the learner and also the trainer. The trainer uploads the learning objects through the user interface. The learning objects are stored in a learning object repository. The user enters the required information such as his profile, knowledge background, learning goal etc. through the interface. The learner details are stored in the learner profile repository. The learning style of the user is identified through the Felder-Solomon questionnaire. The path generator generates all sets of possible paths from the learning objects repository using the pre-requisite relationship. The knowledge of the learner is represented using the surmise relation $\sqsubseteq$. Then, the path that is most suitable for the learner's requirements is presented to him. Thus the path presented to the learner is a personalized path.

The proposed work provides personalization using a graph based approach. The topic graph is a directed acyclic graph *G (V, E)* with $V = \{v_0, v_1, \ldots, v_n\}$ as the set of vertices, v_i represents a topic or a concept, $E = \{e_0, e_1, \ldots, e_n\}$ is the set of edges, and e_i represents relationship among the various topics or concepts. Each e_i is assigned with a weight w_i whose value represents the attribute concerning the learning goal of the learner. The learner's personal details and preferences are collected using a user interface. The learner's style is identified in all the four dimensions as per the Felder-Silverman learning style model. The current research focuses on the learning style, the learning goals and background knowledge of the learner. The learner model is constructed using a graphical user interface which collects information from the learner. It is explicitly done via questionnaires and collection of user information. This way of acquiring the user model is called cooperative modeling [11]. The metadata of the learner are chosen as per the IEEE/PAPI standard developed by the IEEE Learning Technology Standardization Committee (LTSC). The initial learner model is dynamically updated during the course based on the feedbacks obtained from the learner during the course. This helps in managing the concept drift during the course of learning.

The learning objects will be stored in the learning object repository. Learning objects can be defined as any entity, digital or non-digital, which can be used, reused or referenced during technology supported learning. Trainers provide the metadata about the learning objects while uploading them. The following metadata are collected about the learning object: file or media type of the learning object, the type of the contents in the learning object, the depth of the material contained in the learning object, the title of the learning object, the author, the granularity, its pedagogical purpose, the learning objectives, the date of creation, the difficulty level of the learning object, average learning duration, conceptual association, application contexts, keywords, the pre-requisites for the learning object, the post order following the learning objects and the interactivity level of the learning object.

The learning objects are grouped into six groups that will suit different types of learners possessing various learning styles. The grouping is based on the result of the association rules on Felder-Soloman questionnaire. Based on the above result, the learning objects are grouped as follows: the learning objects containing majority of diagrams, graphs, tables & charts form group 1, the learning objects with text, text combined with diagrams and graphs with explanation form group 2, the learning objects containing analytical problems, puzzles and source codes form group 3, learning objects which consist of examples, exercises and activities form group 4, learning objects which contain only hints form group 5, learning objects which consists of detailed notes form group 6.

From the results of the association rule mining, the first group of learning objects that consists of diagrams, graphs, tables and charts are identified to be suitable for visual learners and global learners. Similarly the second group consisting of text, diagrams and graphs with explanation are suitable for verbal and sequential learners. Learning objects consisting of analytical problems, puzzles and source codes are suitable for intuitive and reflective learners and intuitive learners, and those containing examples, exercises and activities are suitable for active and sensing learners. Learning objects which contain just hints, outlines and overview about the lessons to be learnt are identified to be suitable for global learners and learning objects whose contents form detailed notes are suitable for sequential learners. This information has been used to identify the learning object that suits the learner the most based on his learning style.

The metadata about the learning objects are used for identifying the relationship between the learning objects. The pre-requisite relationship $\vdash^P$ is used for representing the relationship between the learning objects as a topic graph. The graph is formed using the metadata pre-requisites and post-order of the learning objects. The pre-requisite relationships are fed into the system by the trainer using a graphical user interface. A sample topic graph for the subject 'Basics of Programming using C Language' that has been constructed using the pre-requisite relationship $\vdash^P$ is shown in Fig.1.

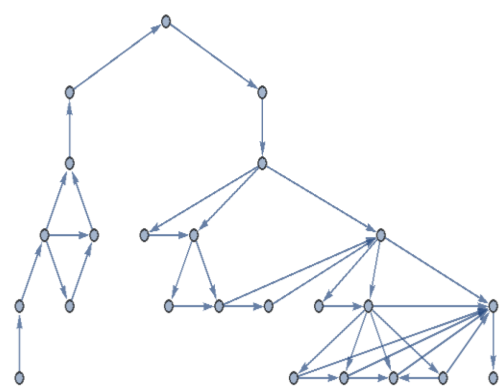

Fig. 1. Topic Graph for 'Basics of Programming using C Language'

This graph has been constructed for twenty four topics in programming basics. The topics considered are shown in Table 1.

Table 1: Topics in Basics of Programming using C Language

Topic Id	Topics	Topic Id	Topics
1	Tokens	13	Multidimensional Arrays
2	Data Types	14	Strings
3	Operators	15	String Functions
4	Input Output Statements	16	Pointers
5	Conditional Statements	17	Pass by Value and by Reference
6	Iterative Statements	18	Structures
7	Goto	19	Arrays of Structures
8	Continue	20	Unions
9	Break	21	Structures and Functions
10	Functions	22	Pointers to Structures
11	Recursion	23	Introduction to Files
12	Arrays	24	File Handling Functions

The pre-requisite relationships for these topics in the syllabus are defined by the trainer and the graph is constructed. The hierarchy of the topics is represented using a directed acyclic graph (DAG). The graph that shows the first five nodes in the hierarchy is shown in Fig. 2 The pre-requisite relationships are indicated by the arrows and the topic numbers.

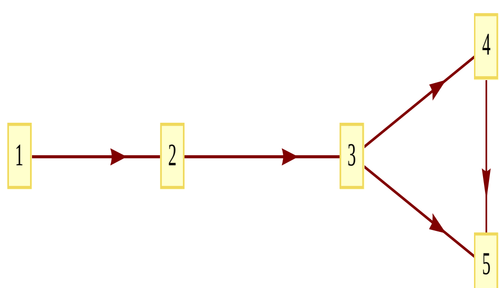

Fig. 2 Topic graph representing the first five nodes

The pre-requisite relationships among all the twenty four topics of the subject are shown in Fig. 3. Then, the topic graph is represented using an adjacency matrix. Each topic consists of r learning objects (LOs) where r is a variable with respect to the topic. The total number of learning objects available for the 'Basics of Programming using C Language' is 105. The learning objects are related using the relations between the topics and represented as a weighted graph that is represented as a matrix. The learning objects that do not suit the learning styles o the learner are filtered. The results of the association rule mining form the basis for the filtering. Then weights are assigned to the LOs based on the goals and styles of the learner.

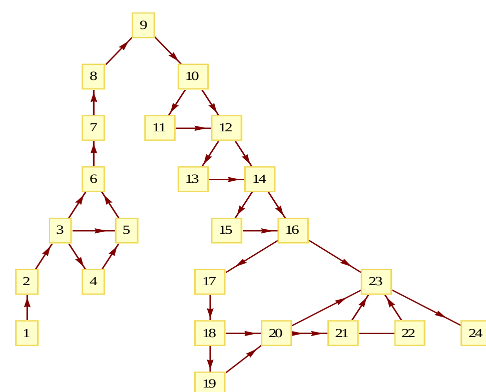

Fig. 3 Topic Graph representing the pre-requisite relationship among the topics of 'Basics of Programming using C Language'

A weighted graph in which the average learning duration is chosen as the weight is shown in Fig. 4. The nodes of the labels represent the weights of the nodes which in this case is the average learning duration.

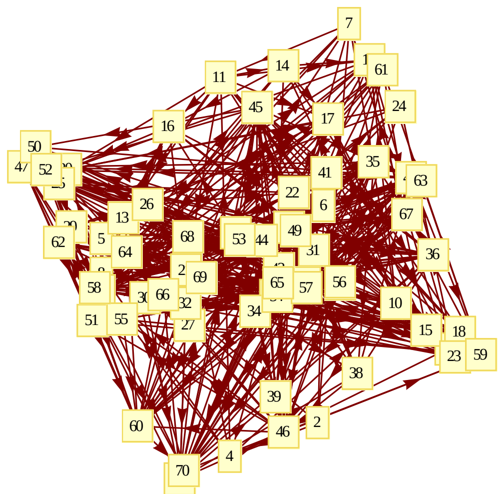

Fig. 4 A learning object graph with average learning duration as the weight

3.5 Construction of a Personalized Learning Path

The learning path is a set of vertices $V = \{v_i, v_{i+1}, \ldots, v_j, v_t\}$ in the learning object graph which sequences the learning objects that learners need to browse when they participate in a course. V_i is the starting point for the learner to reach V_t which is the target knowledge unit. The weights for the graph are chosen based on the style and goal or purpose of the learner. Based on the learner profile collected and the inputs from the learner, the most suitable learning objects from the available set are recommended. The paths for reaching the goals of the learner are determined from the learning object graph and the most effective path that suits the learning style of the learner and the goals of the learner is presented to the learner. The background knowledge of the learner is also taken into account and the concepts that are already known to the learner are removed from the learning path. The personalized learning path is a learning path that has $\sum w_i \rightarrow min$ or $\sum w_i \rightarrow max$ from s to e with min or max value with respect to a threshold value. The algorithm for selecting the personalized learning path based on the goal as minimum time duration is given below.

The topicCnt represents number of topics between the source topic and the target topic. A topic graph is constructed with number of nodes = topic Cnt where nodes represent the topic Ids and edges are connected based on the pre-requisite relationships. The graph is represented using an adjacency matrix. The suitable path for learning is identified by the A* algorithm. The number of learning objects corresponding to the topics is chosen by filtering the learning objects based on the learning style of the learner. A learning object graph is constructed with number of nodes = number of learning objects where Nodes represent the learning object ids and edges represent the connectivity among the topics. Edge weights represent the time duration to learn a learning object. The edge weights may be changed based on the learning goals and learning needs of the learner. An array of adjacency lists are used for finding the adjacent nodes of a given node. The minimum weight of the group of learning objects belonging to a topic that is less than or equal to the minimum threshold value is calculated. The corresponding learning object is stored in an array. The procedure is repeated till the target topic is reached.

4. Results and Discussion

The personalized path chosen for a learner from a source to target topic based on time of learning to be minimum as his learning goal is shown in Fig. 5.

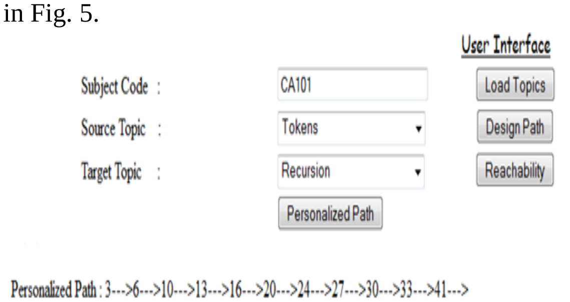

Fig. 5 The Personalized Path for a learner

The personalized learning path has been generated for different learners for their goal as minimum time duration of learning. The learning time taken for learning by choosing the learning objects manually is compared with the learning time taken for learning using the recommended path. The results are shown in Fig. 6.

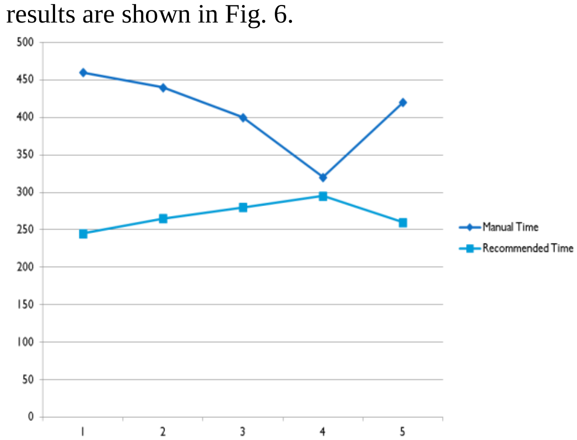

Fig. 6 Comparison of the recommended path with manual selection

The comparison of the scores of five different learners with the recommended personalized path and manual learning are shown in Fig. 7.

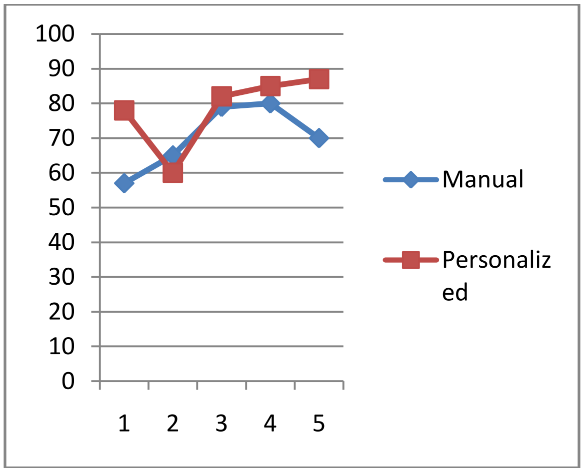

Fig. 7. Comparison of scores of the learners with and without personalization

The results show that the learning efficiency has been improved by recommending personalized paths that suit the goals and characteristics of the learner.

5. Conclusion

Personalization has been proved to be an effective mechanism in various fields. In the proposed work, personalization has been applied to the field of e-learning to improve the efficiency of learning in web based learning systems. Web based learning differs from traditional learning in various ways including the absence of a teacher or trainer. Therefore, they lack in the pedagogical methods that are in general, implemented by a teacher in a classroom to improve the efficiency of learning of each and every learner. Most of the web based educational systems do not take into account the individual learning traits of the learner and therefore, deliver the contents in a one size fits all approach. The modern advancements in web based educational systems in the form of intelligent tutoring systems, provide various means of solving the above problem. The proposed work provides one such solution using a graph theory based approach in which the learning contents are sequenced based on the characteristics of the learners such as the individual learning styles, the learning goal and the needs of the learner. The experimental results show an increase in the efficiency of learning.

References

1. Andrea Sterbini, Marco Temperini, (2009), Adaptive Construction and Delivery of Web-based Learning Paths, 39th ASEE/IEEE Frontiers in Education Conference, Oct 2009.
2. Anh Nguyen Viet, Dam Ho Si, (2006), Applying Weighted Learning Object to build Adaptive Course in E-learning, Proceedings of the 2006 Conference on Learning by Effective Utilization of Technologies, ISBN 1-58603-687-4, 2006.
3. Christina Stahl, Knowledge Space Theory (2011).
4. Dietrich Albert, Cord Hockemeyer, (1997), Adaptive and Dynamic Hypertext Tutoring Systems, Artificial Intelligence in Education: Knowledge and Media in Learning Systems, Vol 39, Frontiers in Artificial Intelligence and Applications, 1997.
5. Dietrich Albert, Cord Hockemeyer, Michael D. Kickmeier-Rust, Alexander Nussbaumer & Christina M. Steiner, (2012), E-learning based on Metadata, Ontologies and Competence-based Knowledge Space Theory, Springer Knowledge Technology, Communications in Computer and Information Science, Volume 295, 2012.
6. Falmagne, J., Doignon, J., Koppen, M., Vilano, M., Johannesen, L.,(1990), Introduction to knowledge spaces: How to build, test, and search them, Psychological Review 97 (1990) 201–224.
7. Falmagne, J.-C., & Doignon, J.-P. (2011). Learning Spaces . Berlin: Springer-Verlag, 2011.
8. Frank Halasz and Mayer Schwartz, (1994), The Dexter Hypertext Reference Model, Communications of the ACM, 1994.
9. G L A Alvarez, J L Montesinos, (2006), An Authoring Tool for Learning Object Sequencing, Latin American Conference on Learning Objects, 2006.
10. Hend Madhour & Maia Wentland Forte, (2008), Personalized Learning Path Delivery, Models and Examples of Application, Intelligent Tutoring Systems, Springer Lecture Notes in Computer Science, Vol 5091, 2008.
11. Hend Madhour & Maia Wentland Forte, (2010), Personalized Learning Path Delivery, Advances in Learning Processes, ISBN 978-953-7619-56-5, 2010.

12. Hongbiao Xu & Wenahou, (2010), Research on Tele-immersion: Collaborative-Learning Environment with Sharing Knowledge-based, Computer and Information Science, Vol 3, No. 4, 2010.
13. Kazuya Seki, Tatsunori Matsui and Toshio Okamoto, (2005), An Adaptive Sequencing Method of the Learning Objects for the e-Learning Environment, Electronics and Communications in Japan, Part 3, Vol 88, No. 3, 2005.
14. Marwah Alian and Riad Jabri, (2009), A Shortest Adaptive Learning Path in eLearning Systems: Mathematical View, Journal of American Science, 2009.
15. Mu-Jung Huang, Hwa-Shan Huang, Mu-Yen Chen, (2006), Constructing a Personalized e-learning System based on Genetic Algorithm and Case-based Reasoning Approach, Expert Systems with Applications, Elsevier, 2006.
16. Nguyen Viet Anh, Nguyen Viet Ha & Ho Si Dam, (2008), Constructing a Bayesian Belief Network to Generate Learning Path in Adaptive Hypermedia Systems, Journal of Computer Science and Cybernetics, Vol 24, No 1, 2008.
17. Roberto Pirrone, Giovanni Pilato, Ricardo Rizzo & Giuseppe Russo, (2005), Learning Path Generation by Domain Ontology Transformation, Advances in Artificial Intelligence, Springer Lecture Notes in Computer Science, Vol 3673, 2005.

11/29/2013