AdaLearn: An Adaptive E-Learning Environment

Marwah Alian
Isra University
Amman, 11622, Jordan
+962 47 11710 ext 2523
marwa@ipu.edu.jo

Mousa AL-Akhras
The University of Jordan
Amman, 11942, Jordan
+962 53 55000 ext 22602
mousa.akhras@ju.edu.jo

ABSTRACT
This paper describes the theoretical and technical aspects that were taken into consideration in the design process of a web-based adaptive e-learning environment we called (AdaLearn). AdaLearn system saves learner's responses into learner's profile then they will be used in future guidance. This paper presents an adaptation scenario in order to give recommendations about contents to individual learners taking into consideration learner's behavior.

General Terms
Elearning in semantic web.

Keywords
E-Learning, Adaptive learning, learning object, learning profile.

1. INTRODUCTION
The traditional learning approach was based on face-to-face learning, this form has evolved into other forms of learning such as distance learning and self-learning from off-line or online materials. Distance learning can take many shapes and it has evolved from distance learning (D-Learning) to electronic learning (E-Learning) and more recently to mobile learning (M-Learning). D-Learning is a form of teaching/learning in which the learners are separated by physical distance; time and/or resources. E-Learning is a learning approach based on the utilisation of technology, E-Learning as a concept covers wide set of applications ranging from computer-based desktop training to web-based learning **[1]**.
E-Learning is an education paradigm that is based on electronic delivery of electronic learning materials over electronic media, including Internet, intranets, extranets, satellite broadcast, audio/video tape, interactive TV, and CD-ROM. The used electronic media are computing devices and the electronic materials are delivered using computer networks. E-Learning as a learning paradigm is also based on interaction with the learner **[1]**.
One of the major goals of E-Learning is to allow learners to access learning materials and information ubiquitously from anywhere and at anytime. Therefore, learners have control of when they want to learn and from which location they want to learn. Also, all humans have the right to access learning materials and information to improve their quality of life regardless of where they live, their status, and their culture.

ISWSA'10, June 14–16, 2010, Amman, Jordan.

When an E-Learning system to be delivered contains learning materials covering different levels of learning, the level of learner is taken into consideration to provide the learner with the learning materials that suit his/her level and his/her fields of interest. The concept is like running a level test of a student applying for a course and the level test is performed to decide the student's entry level. Adaptiveness is needed as some learning resources may not be in a format that is acceptable for different learners' needs and that fit the capabilities of different mobile devices, additionally content adaptation is needed to provide learners with appropriate courses view. To this end, E-Learning systems should employ some sort of adaptiveness.
Adaptiveness in the context of this work means that the same learning materials are represented differently to individual learners based on their interest which is determined based on their previous learning behavior.
There are two ways for the automatic use of course sequencing: adaptive courseware generation, and dynamic courseware generation. The idea of adaptive courseware generation is to generate a course suited to the needs of the learners. It can deliver adaptivity for small group of students, and it allows learners to communicate through the shared context and learn from each other. Also the static course that it generates can be delivered by a regular course management system **[3]**. While the goal of dynamic courseware generation is to generate a personalised (individualised) course taking into account a learning goal and the initial knowledge level of a learner. If the learner does not meet expectation, the course is dynamically re-planned.
In order to generate an individualised course, this course should take into consideration the learner's knowledge, goals, and timeframe, and to generate adaptive course, its difficulty and rate of progress should be considered **[3]**.
E-Learning systems give alternative learning styles through the use of Learning Objects (LOs) such as examples, case studies, and procedural information, in order to provide personalised learning experience. These options give learners the flexibility to choose a suitable learning path instead of a rigid one.
Different LOs have different navigation alternatives, depending on their type, role, content and structure. For example, a learner starting a problem solving is recommended to go through all problem solving steps.
The proposed e-learning system (AdaLearn) in this paper responds to different learners differently by adapting the presentation of learning content to meet the varying needs and learning preferences of different individual learners. It enables learners to select their modular components to customise their learning environments and it enables them to get flexible solutions that dynamically adapt contents to fit individual learning needs.

As learners navigate in AdaLearn, the system will adapt the content based on the learner information profile. For example, learners might be sent to different LOs based on user-initiated request for clarification of prerequisite knowledge, or user requests for preferred knowledge presentation [2].
The rest of this paper is structured as follows. In section 2, previous attempts of adapting course contents in the literature are highlighted. In section 3, some previously developed relevance measures are presented. Section 4 gives detail description of the proposed adaptive e-learning system (AdaLearn). Section 5 illsutrates the proposed system with a working example that elaborates an intra and inter course adaptation scenario for giving recommendations to individual learners according to their motivation history. Section 6 gives recommendations and presents for possibilities of future work.

2. RELATED WORK

Many researchers have been working on the problem of adapting course contents to fit the learners need. An adaptive system for E-learning is proposed by Carchiolo et al. [4]. This system provides students with all paths from an initial knowledge to a desired one. These paths are retrieved and optimised based on the profile of both the student and the teacher. The candidate paths are presented to the students to select one of them and learn its course units.
An interactive web-based adaptive learning environment called *iWeaver* is presented by Wolf [13]. *iWeaver* aims to create an individualised learning environment that accommodates specific learning styles and it offers the learner different representations of learning material and gives recommendations. On the other hand, *iWeaver* does not support adaptive navigation; Building on the Dunn & Dunn learning styles model, Wolf work describes which media representation is allocated to each learning style and the underlying rationale for this allocation. The core concept of *iWeaver* is that this media-style allocation is flexible. It can change dynamically according to learner manners.
Learning Object selection problem in intelligent learning systems was addressed by Karampiperis and Sampson [5] and a decision model was proposed with a function that measures the suitability of a LO for a specific learner. The same methodology is proposed in educational hypermedia systems with some changes suggested on the constructed function by taking two assumptions into consideration [6]. First, the elements of the user model defined from the beginning by the designer and remain the same during the life cycle of the system. Second, Learner's characteristics and preferences stored in user model and the structure of the educational resource description model have been defined by the instructional designer. Then this suitability function is used for weighing the graph of learning paths in adaptive educational hypermedia systems (AEHS).
A framework for individualised LO selection was introduced by Liu and Greer [8]. This framework gives a suggestion to select a group of suitable LOs for the learner; also it calculates the suitability of a LO using a formula that depends on information about the LO, information about the learner, and historical information about the learner and the learning context. The framework has 3 main steps: eliminating irrelevant LOs depending on their characteristics, selecting LO depending mainly on educational information and pedagogical principles, and optimising the selected LOs.

An Adaptive Course Generation (ACG) system is presented by Viet and Si [12] to create adaptive courses for each learner based on evaluating demand, ability, background and learning style of learners. There is a test in each section in the course content; also an algorithm is proposed to select the LOs from the Learning Object graph, that are suitable for the requirements of a learner.

3. RELEVANCE MEASURMENT

To measure the relevance of a Learning Object to a learner, some measurement of relevance should be applied. Several relevance measures have been proposed in the literature.
The heuristic Relevant Knowledge First (RKF) was proposed by Kreuz and Roller [7] for making decisions in configuration processes based on the relevance of objects in a knowledge base. They suggested their own definition of relevance as:

"*The relevance at the time of t of an object o in the context of a task class c is calculated as a function of time since a last access (forget if o is not part of the solution) and the rewards given by a user (train, if o is part of the solution)*" [7].

Also, the authors considered two factors: First: Objects are considered relevant if they have already been useful for similar tasks, and objects that did not help to find solutions will probably not help in the future. Second: New objects should be taken into consideration in order to avoid conservatism, because objects can be forgotten.
The proposed formula for relevance of an object is defined as follows:

$$\mathrm{rel}(o, t, c) = \begin{cases} \mathrm{train}(o, t, c) & \text{if } o \text{ is part of the solution} \\ \mathrm{forget}(o, t, c) & \text{if } o \text{ is not part of the solution} \end{cases} \qquad \textbf{(1)}$$

Using this formula an object is considered to be important if it is used frequently as its relevance is increased. Relevant Knowledge First aims to speed up the configuration process and to improve the quality of the solutions relative to the value that users given when using the object.
Another formula was proposed by Peñas et al. in an application of corpus-based terminology extraction in interactive information retrieval for term weighing [10]. In this work a relevant value is given to every detected term, in order to select the most relevant terms in the domain. The used relevance formula is defined as:

$$\mathrm{Relevance}(t, sc, gc) = 1 - \mathrm{Log}_2\left[2 + \frac{F_{t,sc} \cdot D_{t,sc}}{F_{t,gc}}\right] \qquad \textbf{(2)}$$

Where

$F_{t,\,sc}$: relative frequency of term *t* in specific corpus *sc*.

$D_{t,\,sc}$: relative number of documents in *sc* where term *t* appears.

$F_{t,\,gc}$: relative frequency of term *t* in generic corpus *gc*.

The above formula satisfies two constraints; First: Less frequent terms in the domain corpus should be less relevant. Second: Highly frequent terms in the domain corpus should have higher relevance, unless they are also very frequent in the comparison corpus or they appear in a very small fraction of the documents in the domain corpus. It can be noticed that most of the previous formulas depend on several factors to determine relevance; nevertheless, they share a common concept which is the use of the most frequent term/object as the most relevant one.

We are going to define relevant Learning Object depending on weights given for that object using time factor and the most frequent Learning Object will be the most relevant one. Discussion for weight measurement is given in details in the next section.

4. ADAPTIVE E-LEARNING

To be able to adapt an e-learning system to fit the learner's profile and needs, adaptation should made to give more emphasis to the contents the user is more interested in. This concept is similar to learning by example. In learning by example or what is known as instance-based-learning (IBL) the system learns what to do by comparing the symptoms of the current case with previously seen similar cases **[9]**. A doctor does this all the time by comparing the symptoms of the current patient with similar patients he/she treated before. Once the doctor decides which previous patient is similar to the current one, then he/she diagnoses the current patient analogous to the way he/she diagnosed previous patient(s). The same (similar) medicine would be given in this case to the current patient.

The system proposed in this paper could be considered to be inspired from the above learning experience the doctor uses. The aim of the proposed e-learning system is to provide some sort of adaptiveness to the learning contents based on memorising the learner's behavior during his/her previous learning sessions. That behavior would be saved in a learner's profile. The information saved in the learner's profile is going to be used to give recommendation to the learner.

The proposed system is a typical e-learning system that consists of a set of courses and each course contains different learning materials about that course, the material could be visual, audio or textual. Adaptation is done at different levels: First, inside the course (intra course) adaptation by giving emphasis for the materials that the learner previously navigated the more. Second, among the courses (inter courses) adaptation, by giving more emphasis for the most interesting course among the available courses.

Next the database structure of the proposed system is illustrated, default weights for different Learning Objects are assigned, and intra and inter course adaptation are illustrated.

4.1 Database Structure

In order to implement adaptation of the proposed adaptive e-learning system (AdaLearn), certain information needs to be stored about the courses, the course's materials (Learning Objects) and the learner's profile. Historical usage of Learning Objects can also help in future guidance. The above information is organised in the Entity Relationship Diagram (ERD) shown in Figure 1.

When the entities and relations shown in Figure 1 are mapped into equivalent tables according to standard database mapping techniques **[11]**, the required information will be organised in four tables as described in the following tables. Table 1 Courses, Table 2 Learning Objects, Table 3 Learners, and Table 4 Browsing History, along with attributes that link these tables together.

Table 1 contains information about available courses; the basic information includes an identifier for each course (Primary Key, PK) and a description for the course. Other information can be added if needed.

Table 2 contains information about Learning Objects that belong to each course; the basic information should be an identifier for each Learning Object (Primary Key, PK) and description for that Learning Object type. Information about to which course that Learning Object belongs to is essential and is added in a form of Foreign Key (FK) taken from the courses table. One very important piece of information is related to the time a learner is expected to spend on the Learning Object. This time is proportional to the duration and/or complexity of that Learning Object and it can differ from one learning object to another. The longer and/or the more complex the Learning Object is, the longer the required time is assigned to that Learning Object.

If the actual time the learner spends navigating that Learning Object exceeds the expected time, then more weight will be given to that object in future visits to the system. Similarly, if the learner spends less than the expected time or no time at all, less weight will be allocated to that object. The time the learner spends on the learning object is calculated by running a script that calculates the difference in timestamp between the time the user opens the page containing the object and the time the user closes that page. Such calculations of weight and relevance are going to be described in the next section.

Table 3 contains information about the learners; such information includes his/her id and password. If needed, more information (not shown here) like the address, telephone number, fax number could be added.

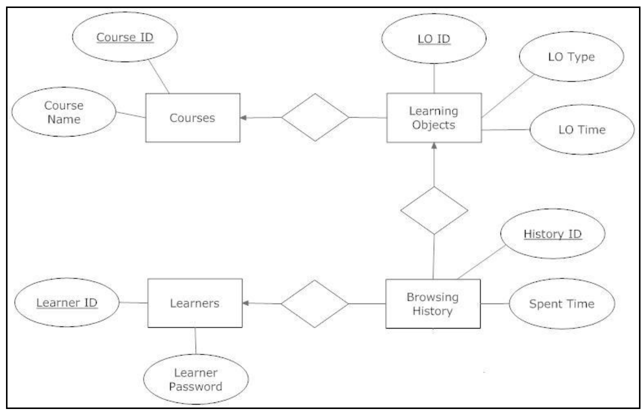

Figure 1: ER diagram for tables used in e-Learning system

Table 1: Courses Table

Attribute	Description
Course ID	Course unique identification number
Course Name	Course description

Table 2: Learning Objects Table

Attribute	Description
LO ID	Learning Object unique Identification number
LO Type	Learning Object Type (i.e. video, audio, text)
Course ID	Course unique Identification number (foreign key)
LO Time	Learning Object required time to learn.

Table 3: Learners Table

Attribute	Description
Learner ID	Learner unique Identification number
Learner Password	Password that the learner uses to access the e-Learning System

Only authorised learners can navigate through the system using learner's ID and password. Learner navigation information which will be saved during learner navigation through the e-Learning system is presented in Table 4 which contains information about the browsing history of the learners; such information acts as log information for each learner, i.e. a learner's profile. Each record in the browsing history contains an identifier for the transaction (History ID). Each record contains identifier about which learner is involved in the transaction, and which course and Learning Object(s) are used. The actual time spent on that Learning Object will also be stored and the weight of that transaction. If the time spent is more than the expected time (derived from the Learning Object table), a weight greater than one is assigned. If the time spent is less than the expected, a weight less than one is assigned. A weight of one is assigned if the expected and actual times are identical.

Table 4: Browsing History Table

Attribute	Description
History ID	History Identification number
LO ID	Learning Object unique Identification number foreign key (foreign key)
Course ID	Course unique Identification number (foreign key)
Learner ID	Learner unique Identification number (foreign key)
Spent Time	Time spent by learner in order to learn the Learning Object

4.2 Weight Calculations

The weight of Learning Object i (LO_i) is calculated as shown in Equation 3:

$$W_i = \frac{\sum t_s}{t_e * \text{Number of previous visits}} \qquad (3)$$

Where

W_i: the weight for LO_i.

Ts: the time the learner spent in learning LO_i in each of his/her previous visits.

t_e: the time expected for learning LO_i.

Calculated weights will be used as future guidance for the learner, i.e. to adapt the contents for that learner. These weights will be recalculated each time the learner logins to the system and navigates through its courses. As such, recommendations that given to the learner will subsequently change.

However, in an online e-learning system this guidance for the learner is interpreted as enlargement for the picture that represents a recommended Learning Object because courses and Learning Object types are presented in the system as pictures of a specific size.

To compute the picture size for a recommended type of Learning Object (i.e. text, audio, or video) weights which are calculated as in Equation (3) will be multiplied by the actual size of LO picture that represents the Learning Object during the navigation process.

Next two algorithms are constructed in order to give recommendations for learners while they navigate through the materials of each course (intra course) adaptation and among the courses themselves (inter courses) adaptation.

4.3 Intra Course Adaptation

Once the learner started the navigation of AdaLearn, previous visits to each Learning Object that belongs to the course currently navigated are taken into account. All previous visits to each Learning Object are retrieved. Summation of time spent on that Learning Object is divided by the expected time to be spent on that object multiplied by the number of visits as illustrated in Equation 3.

Relative weights are then calculated as following: The total weight for all objects in that course is calculated as the summation of the weights for each Learning Object that belongs to that course. The size of the hyperlink (picture) to each Learning Object is resized (increased or decreased) as needed. The new size is calculated as shown in the following Equation:

$$N_{size} = \frac{W_{LOi}}{\frac{\sum W_{LO}}{NO}} \times D_{size} \qquad (4)$$

Where

N_{size}: the new picture size of Learning Object i (LO_i)

D_{size}: the Default size for all Learning Objects pictures.

W_{LOi} : Weight for Learning Object i.

$\sum W_{LO}$: Overall weight for all Learning Objects that belong to the current course.

NO: Number of objects that belong to the current course.

The denominator in Equation 4 represents the default weight when all Learning Objects have the same importance. Based on Equation 4 one or more of the Learning Objects inside the course are recommended for the learner. The above description is illustrated in Figure 2.

4.4 Inter Courses Adaptation

In order to lead the learner for a recommended course, all weights for all Learning Objects of that specific course are summed to calculate the weight for each course. Then summation of these weights is then made and divided by the number of courses to calculate the default weights for all courses in the system; the weight for each course is divided by the default weight and multiplied by the default size of the hyperlink (picture) that represents the course to enlarge, decrease, keep the same size the hyperlink for that course. This way one or more of the courses will be recommended to the learner based on previous activities (depending on courses weights that resulted from totaling weights for all Learning Objects representing the course). Inter courses adaptation is shown in Figure 3.

Future recommendations for the learner depends on information like the time that learner actually spent studying a Learning Object which is a main factor in calculating the weights of a

Learning Object and the weight of courses. Another factor in calculating the weight of the Learning Object is the expected required time which is an attribute in Learning Objects table (Table 2) and it is estimated when a Learning Object (LO) is constructed.

Initial value for W_i =1/Number of objects in that page

1. Calculate total time that learner spent in Learning Object i (LO_i) during previous visits
2. Calculate the weight W_i for each LO_i as in Equation 3
3. Calculate the overall weight for all objects.
4. Calculate the default weight (no preference) as overall weight/Number of objects in that page
5. Calculate new picture's size S*i* that represents LO_i in Course exploring page by:
   Actual size * object weight of LO_i/ default weight
6. The recommended LO_i has the largest S_i

Figure 2: **Algorithm 1** Learning Object type recommendation algorithm

1. Use steps 1,2 from Algorithm 1
2. Calculate ΣW_i for all LO_i for a Course *i* (C_i)
3. Calculate the default weight (no preference) as overall weight for all courses/Number of courses in that page
4. Calculate new picture's size S_{ci} that represents C_i in Learner's Learning exploring page by
   Actual size * weight of C_i/default weight
5. The recommended C_i has the largest S_{ci}

Figure 3: **Algorithm 2** Course recommendation

5. ILLUSTRATIVE EXAMPLE OF AdaLearn

In this section an example will be given about the calculations used to adapt the contents of the e-learning system to the learner. Also examples of part of the system before and after adaptation will be given.

Figure 4 shows example of 3 Learning Objects of a specific course (Chemistry) before any kind of adaptation. Figure 5 shows the higher level page which is the courses before any adaptation.

Assume the required time for the Video (LO_1), Audio (LO_2) and Text (LO_3) contents are 90 minutes, 20 minutes and 10 minutes, respectively. If a learner u1 visited that course and spent 60 minutes on LO_1 of type video and spent 30 minutes on LO_2 of type audio, then he/she spent 5 minutes on LO_3 of type text. The weights for the three Learning Objects will be W_1 =0.67, W_2 = 1.5, and W_3 = 0.5 (relative weights). The total summation of weights for all the Learning Objects is 2.67 (overall weight). As we have 3 objects in the page, each object is expected to have almost the same size if the learner has no preference of one object over the other, i.e. each object is expected to have weight of 2.67/3=0.89 (default weight).

Assuming the original sizes of the 3 links to the LOs are 50×50, then the new sizes of LO_1, LO_2, and LO_3 are calculated as follows:

LO_1 (New Height or Width) = 50×0.67/0.89=37.64

LO_2 (New Height or Width) = 50×1.5/0.89=84.27

LO_3 (New Height or Width) = 50×0.5/0.89=28.1

Therefore, the sizes of three Learning Objects pictures become:

S_1 (picture size for LO_1) =37.64×37.64

S_2 (picture size for LO_2) =84.27×84.27

S_3 (picture size for LO_3) =28.1×28.1

Each time the learner visits the system, information about the time that he/she spent over each Learning Object will be saved in the Browsing History table to aid in recalculating the new weights. The more time the learner spends on an object, the higher the weight for that object will be and the more emphasis will be focused on that object in the form of larger links as shown in Figure 6 in the second visit for the system. It can be noticed that the picture hyper link for the audio Learning Objects is increased due to the more focus, therefore AdaLearn recommends going to this content again.

The same principle can be applied at the course level. Using a query against the browsing history table, total weights for all Learning Objects representing a specific course can be obtained. Figure 7 shows an example that the following weights after learner navigation: course1 total equals 1 (time spent/time expected), course2 equals 1.2, and course 3 equals 0.7. The overall weight is 2.9. The Default weight is 2.9/3=0.967.

Assuming the original sizes of the 3 links to the courses are 140×100, then the new sizes of courses are calculated as follows:

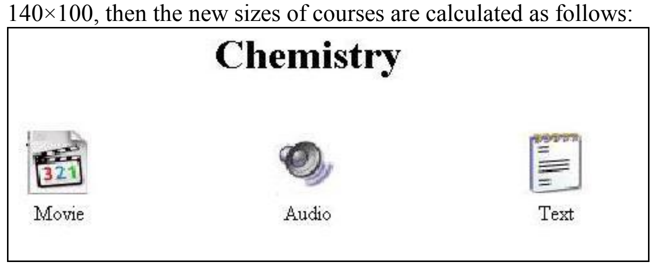

Figure 4: Learning Objects before adaptation

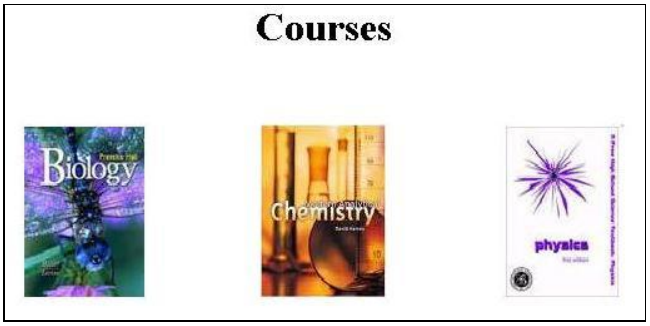

Figure 5: Course before adaptation

C_1 (New Height) = 140×1/0.967=144.77

C_1 (New Width) = 100×1/0.967=103.41

C_2 (New Height) = 140×1.2/0.967=173.73

C_2 (New Width) = 100×1.2/0.967=124.095

C_3 (New Height) = 140×0.7/0.967=101.34

C_3 (New Width) = 100×0.7/0.967=72.38

Therefore, the size of 3 Courses pictures become

S_{C1} (picture size for C_1) =144.77×103.41

S_{C2} (picture size for C_2) =173.73×124.095

S_{C3} (picture size for C_3) =101.34×72.38

AdaLearn changes the focus of the 3 illustrated courses based on the learner's preference as shown in Figure 7. More focus (relative size) will be given to the chemistry course as it has more weight over the other two courses. Of course this can change if the learner changes his/her navigation style as the learner can feel more changes every time he/she visits AdaLearn.

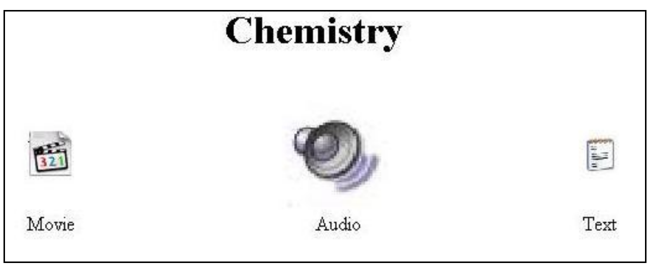

Figure 6: Learning Objects after adaptation

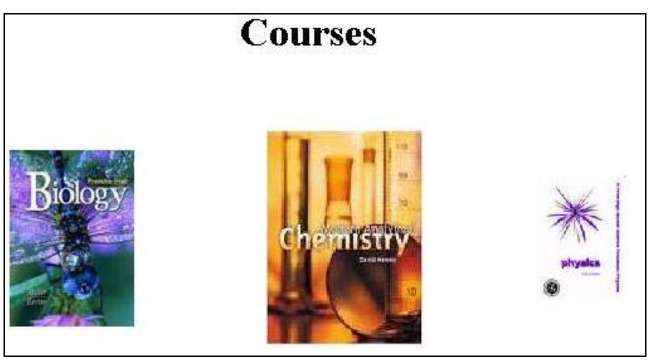

Figure 7: Courses after adaptation

6. CONCLUSIONS AND FUTURE WORK

In this paper the authors proposed an adaptive e-learning system (AdaLearn) that can be used for giving recommendations for individual learners about what is the best course that can fit their needs. The decision is based on previous navigation behavior of the learner. Similarly, inside each course, different Learning Objects are assigned different weights; hence, they will receive different focus based on the time spent by the learners.

An important aspect that still needs to be considered is the implementation of appropriate evaluation of the effect that AdaLearn in real learning environment. This implementation can be realised by testing AdaLearn in real class rooms and recording student's responses to see the level of satisfaction. Also as a future proposal the authors plan to improve AdaLearn by giving more focus to other possibilities of adaptiveness. Some of the possibilities include, adapting to courses that are more difficult to the learner to recommend him/her spending more time on more difficult subjects than easy subjects. The difficulty can be determined by an independent assessor, self-test or letting the learners themselves decide which subjects they feel are the most difficult.

7. REFERENCES

[1] AL-Akhras, M., Almobaideen, W., Hdairis, K., Al-Jaouni, A. and Al-Zeer, R. 2008. Remote conferencing using SIP protocol stack and Java APIs. 7th International Internet Education Conference & Exhibition (ICT-Learn 2008), 7-9 October, 2008, Cairo, Egypt.

[2] Atif. Y., Benlamri, R. and Berri J. 2003, Learning Objects Based Framework for Self-Adaptive Learning, Education and Information Technologies, IFIP Journal, Kluwer Academic Publishers, 8:4, pp 345-368.

[3] Brusilovsky P., Vassileva J. 2003. Course sequencing techniques for large-scale web-based education, International Journal of Continuing Engineering Education and Life-long Learning, Vol. 13. pp.75-94.

[4] Carchiolo V., Longheu A., and Malgeri M. 2002. Adaptive Formative Paths in a Web-based Learning Environment, Educational Technology & Society 5 (4), 2002.

[5] Karampiperis P., Sampson D., (2004), "Adaptive Learning Object Selection in intelligent learning systems", Journal of Interactive Learning Research, vol. 15(4), pp. 389-407. November 2004.

[6] Karampiperis P., and Sampson D. 2005. Adaptive Learning Resources Sequencing in Educational Hypermedia Systems, IFETS: International Forum of Educational Technology and Society, pp 128-147.

[7] Kreuz I., and Roller D. 2001. Relevant Knowledge First, Reinforcement Learning and Forgetting in Knowledge Based Configuration, paperID: cs.AI/0109034, arXiv 19-Sep-2001.

[8] Liu J., and Greer J. 2004. Individualized Selection of Learning Object, Workshop on Applications of Semantic Web Technologies for e-Learning (SW-EL@ITS'04), 30 August-03 September, 2004, Maceió, Brazil.

[9] Mitchell, T. M. 1999. Machine learning and data mining. Communication of the ACM 42, 11 (Nov. 1999), 30-36. DOI= http://doi.acm.org/10.1145/319382.319388.

[10] Peñas A., Verdejo F. and Gonzalo J. 2001. Corpus-based terminology extraction applied to information access, Proceedings of the Corpus Linguistics, UNED, Spain.

[11] Silberschatz, A., Korth, H., and Sudarshan, S. 2006 *Database Systems Concepts*. 5th edition. McGraw-Hill, Inc.

[12] Viet A. N., Si D. H. 2006. ACGs: Adaptive Course Generation System - An Efficient Approach to Build e-Learning Course, Proceeding of The Sixth IEEE International Confernce on Computer and Information Technology (CIT'06), pp. 259- 265. IEEE Computer Society.

[13] Wolf C. 2002. *iWeaver:* Towards an Interactive Web-Based Adaptive Learning Environment to Address Individual Learning Styles, Australia, EURODL 2002.

Ms. Marwah Alian received her B.Sc degree in Computer Science from Hashemite University in 1999. After graduation, she worked as programmer then as a teacher in a number of high schools in Jordan then she received her M.Sc. degree in Computer Science from the University of Jordan in 2007 with a thesis titled "The Shortest Adaptive Learning Path in eLearning Systems". After graduation she became a member of a technical team for a funded project called Science Education Enhancement and Development (SEED) supported by Japan International Cooperation Agency (JICA). Since 2008 she is working with the Computer Science Department in the Science and Information Technology Faculty at Isra University, Jordan. Marwah's research's interests include e-learning Systems, Adaptive Learning, Mobile Learning, and Mobile Banking. She has a number of publications in the area of e-learning, adaptive e-learning, and data mining.

Mousa AL-Akhras received his B.Sc. and M.Sc. degrees in Computer Science from the University of Jordan, Jordan, in 2000 and 2003, respectively. He received his Ph.D. degree from De Montfort University, UK, in 2007. His M.Sc. and Ph.D. theses were titled: "Instance Reduction Using Techniques for Building Ensembles of Classifiers" and "Quality of Media Traffic over Lossy Internet Protocol Networks: Measurement and Improvement", respectively. He is currently working as an assistant professor in the Computer Information Systems Department in King Abdullah II School for Information Technology (KASIT) at the University of Jordan, Amman, Jordan. In addition to his teaching responsibilities he is working as an assistant dean for computer and development affairs in the faculty of graduate studies at the University of Jordan. His main research interest is the area of Artificial Intelligence and mainly artificial neural networks. His research interests include Robotics, Genetic Algorithm, Fuzzy Logic, statistics, and multimedia communications. He is also interested in the area of electronic learning (e-learning) and mobile learning (m-learning). Mousa is a member of IEEE and he was elected as a secretary for general activities in the IEEE executive committee in Jordan section, region 8 for the years 2010-2011. He is also a member of IEEE CIS & RAS Societies. Mousa is in the organizing and technical committees for a number of local and international conferences. Also he serves as a reviewer and a member of the editorial board in a number of local and International Journals.