Je-LKS
Journal of e-Learning and Knowledge Society
The Italian e-Learning Association Journal
PEER REVIEWED PAPERS
Vol. 11, n.1, 2015
ISSN: 1826-6223 | eISSN: 1971-8829

ON DEMAND ANALYSIS OF LEARNING EXPERIENCES FOR ADAPTIVE CONTENT RETRIEVAL IN AN E-LEARNING ENVIRONMENT

Raghuveer V.R
B.K. Tripathy

School of Computing Science and Engineering, VIT University, Vellore, Tamilnadu, India
vrraghuveer@vit.ac.in - tripathybk@vit.ac.in

Keywords: xAPI, learning experience, learning objects, adaptive content retrieval

Understanding the learning experiences plays a vital role in identifying the suitable learning content for the learners. In this regard, the standards like the experience Application Programming Interface (xAPI) are of great help as they have the potential to record and represent the learning experiences over the e-learning environment. As the learner requirements vary with their understanding of the topics over the learning cycle, there is an inherent need for dynamic derivation of the learner's requirement at each learning instance. However, the limitation with experience statements generated through the xAPIs is that they fail to convey the detailed information about the Learning Object (LO) or the learner who used it. This paper addresses the issues with the representation of experience statements by proposing a multidimensional view of learning experiences such that they could be analyzed effectively. The Cross Dimensional Slicing (CDS) algorithm proposed in this paper has proved that the multidimensional representation of learning

for citations:
Raghuveer V.R, B.K. Tripathy, (2015), *On demand analysis of learning experiences for adaptive content retrieval in an e-learning environment,* Journal of e-Learning and Knowledge Society, v.11, n.1, 139-156. ISSN: 1826-6223, e-ISSN:1971-8829

experiences greatly improves the effectiveness of analyzing them and thereby improving the precision of LOs being recommended. Also, the steep increase in the accuracy of recommendation of LOs over the different batches of learners considered for the study has reduced the number of slow learners of the learning environment altogether.[1]

1 Introduction

With the advancements in World Wide Web (WWW) and mobile technologies, e-learning environments these days are capable of serving a vast majority of learners around the world (Alexander, 2006). These environments allow their learners to register online and utilize the digital LOs available in different formats viz. text, video, audio, etc. (Wiley, 2000) The LOs are stored and indexed inside a centralized storage called the Learning Object Repository (LOR) (Richards *et al.*, 2002). Each LO has the additional information associated with it called the Learning Object Metadata (LOM) that describes the properties of an object and enables effective search and discovery of objects from across the repositories (Learning Technology Standards Committee, 2002). The LOs delivered through the LMSs of these learning environments are intended to serve the specific learning objectives of their learners.

In any e-learning environment, the learner profile plays a major role as it has the potential to reflect the learner's requirement precisely to the Learning Management System (LMS) (Bergeron, 2014). The learner information recorded under the learner profile is mainly categorized under three major categories viz. knowledge, skills and preferences. The attributes of these categories are in turn used to filter the LOs retrieved for a specific learner query.

In most of the courses offered online, the LOs are delivered to the learners based on a predefined learning path with a set of topics under it (Warren *et al.*, 2014). The learning path and the contents for each topic are decided by the service providers and presented to the learners as a learning module. This learning module delivers the LOs in an orchestrated fashion to all the learners based on certain constraints set for the course.

With the learning activities taking place beyond the LMSs (like youtube, coursera, etc.) these days, there arise a demand to record such activities of the learner in order to get a complete picture of their learning profile (Dalsgaard, 2006; Raghuveer & Tripathy, 2012). This is achieved with the help of xAPI specification which are like pieces of code embedded into applications like web browser, web reader tools, etc. that records the learning experiences in the form of statements like <subject> <verb> <object> (e.g. John experienced flowchart) (Experience API Working Group, 2013). These statements are called experience statements and are stored inside the Learner Records Store (LRS).

[1] A preliminary version of this paper was presented in the International Conference on Technology for Education, December 18-21, 2014, Amrita University, Kerala, India.

The verbs in these experience statements are used to differentiate the experience of the learners with respect to that of the LOs they have used. The nature of these statements makes them easily portable across the LMS platforms and also enables the LMS to statistically analyze them to derive conclusions on the utilization of LOs (Gibson *et al.*, 2014). The vast amount of LOs available across the online repositories like MERLOT, Wikipedia, etc. has exposed the learners to a pile of learning content. In such a scenario, unless the experience statements are embedded inside each of these objects, their utilization becomes unknown to the LMSs. Also, the LOs presented through the online repositories are mostly in the large granular form like documents, videos, web-pages, presentation slides, etc (Raghuveer & Tripathy, 2012). Embedding the experience statements inside such large granular content requires a lot of effort as many statements have to be placed across the content to convey the LMS about the extent of coverage on a topic by the learner.

With the increasing volume of LOs available across the repositories, content recommendation has become popular in the field of e-learning as it reduces the time required to search and retrieve the appropriate LOs for the learners (Pazzani & Billsus, 2007). The content recommender works by identifying the suitable LOs based on the past actions of the learners. For example, if the learner has utilized a content of type animation, then the next search would bring more contents of animation type. Similarly, over the period of time the learner interests are determined based on the parameters of the LOs utilized. However, if the LOs have to be recommended on a new topic which the learner has not been exposed to, then the content based recommendation is of no help.

The other type of recommender that is popular is the collaborative recommender, which recommends the products based on the usage pattern of the people who belong to the same group or the same location (Qiao, 2014). Collaborative learning content recommenders analyze the contents that have been used by learners with similar profiles and recommend the contents for a new learner based on that. These types of recommenders are capable of recommending the right content for even the learners who are new to the learning domain as it does not need any information on the type of LOs used by the learner.

With the xAPIs capable of recording the learning experiences even beyond the LMS, there should be a proper method for analyzing the statements generated over the period of the learning cycle in order to determine the learning pattern of learners and recommend the appropriate LOs based on that (Siemens & Long, 2011). Moreover, as the LMSs these days are offering large number of LOs for its learners under different subject domains, the learner profile should record the changing needs of the learners with respect to each domain separately. This would enable the LMS to retrieve the contents according to the needs and preferences of the learner on a specific domain. But, presently the LMSs maintain single common learner profile for all the subject domains with the values of its attributes updated by the learner at the time of profile creation. Also, in majority of the cases, the learner profiles primarily record the content preferences of the learner and retrieve the LOs only based on that. The changes in the content preferences over the learning cycle are not reflected properly in the profile (Raghuveer & Tripathy, in press). Such a learner profile with inadequate learner information is the cause of concern for the modern day e-learning environments.

In order to improve the effectiveness of retrieving the most suitable content for the learners, the LMS must be aware of the dynamically changing requirements of the learner and also should know about the utilization of LOs by learners with similar profiles. For that, the learner profile should record the learner requirements in a subjective manner and update it over the learning cycle. Finally, the learning experiences thus recorded with the help of domain specific learner profiles must be analyzed periodically (at every learning instance) to determine the learner's requirements at every stage of learning.

2 Existing System

Some of the existing LMSs and organizations have adopted the Tin Can API (which is an implementation of xAPI) architecture in different ways for getting a clear picture of the learning history and interests of their employees/learners. A few of the Tin Can implementations for organizations and LMSs have been surveyed to understand the ways in which the experience statements have been utilized to improve the performance of the learners.

Watershed LRS

Watershed LRS is a Learning Record Store that was created with the intention of storing the learning experiences of the employees of an organization. It records different activities that the people of an organization have performed beyond their LMSs. The Watershed LRS is capable of tying together the different systems like the Customer Relationship Management, training platforms, Human Resources, and in turn allows the data from these systems to articulate the learning experiences in the form of statements. Such experiences of the learners generated from these systems are in turn stored inside the Watershed LRS. This allows the organizations to generate correlations between the learning data and real-world performance of employees. Also, the provision for sharing data across the LRSs helps to port the information about the employees who move from one organization to another.

BookOnPublish

A specific implementation of Tin Can architecture is the BookOnPublish, where the printed copies of the books are created in digital form with the support for interactive content. This helps the book author to provide the interactive quiz or multimedia content which otherwise is not possible through printed versions. The books created on BookOnPublish can record the learning experience through interaction and can forward it to the author for a quantitative feedback on its usage. The feedback on the content and its presentation along with the extent to which they are useful to the readers are instantly sent back to the content authors. Such information in the form of experience statements enables the author to alter the form of content or use of appropriate examples to make the book more interesting.

Tappestry

Tappestry is a mobile based social network for learning built on the top of Tin Can architecture. This application helps the employees of an organization to socialize and share their skills and knowledge they have gained over the period of time with the other members of the organization. The information regarding the skills and potential of the learners obtained through informal learning are recorded in the form of experience statements and then communicated to the organization's LRS. Also, the information expressed in the form of Tin Can statements can be easily organized and analyzed by the organizations based on their requirement.

All these existing systems support recording of learning experiences by embedding appropriate statements into the learning path such that they get triggered automatically when the learner utilizes the LO. The experience statements generated are of the form <Learner X experienced topic Y> and they are stored inside the LRS. The learning experiences thus generated can be analyzed to determine the utilization of LOs by the learners. The results of the analysis were mainly used by the LMSs to understand the collective behavior of learners in the e-learning environment to take the appropriate course of action towards presenting the learning content.

Figure 1 shows the sample set of experience statements made available through SCORM public cloud in the Java Script Object Notation (JSON) format. These experience statements are recorded with additional information like the type of activity, progress of the learner, etc. in the form of <name, value> pairs. The advantage of JSON over the Extensible Markup Language (XML) format of content representation is that the JSON format matches with the data model of most of the programming languages available in the market thereby making these statements portable across the products that were developed using these technologies.


```
{"statements":[{"id":"bb196fad-0cdc-46b3-8bc5-
def9f4c646f8","actor":{"name":["Chris
hooks"],"mbox":["mailto:chris.hooks@scorm.com"],"objectType":"Person"},"verb"
:"experienced","inProgress":false,"timestamp":"2014-07-
07T21:12:38.906Z","stored":"2014-07-
07T21:12:38.906Z","authority":{"account":[{"accountServiceHomePage":"http://c
loud.scorm.com/","accountName":"anonymous"}],"objectType":"Agent"},"object":{
"id":"http://scorm.com/pong/beatJake noble","definition":{"name":{"en-
US":"Beating Jake noble at Rustici
PingPong"}},"objectType":"Activity"}},{"id":"5d1b18e1-953e-4d13-af80-
ec2840a6e76c","actor":{"name":["Chris
hooks"],"mbox":["mailto:chris.hooks@scorm.com"],"objectType":"Person"},"verb"
:"experienced","inProgress":false,"timestamp":"2014-07-
07T20:28:14.335Z","stored":"2014-07-
07T20:28:14.335Z","authority":{"account":[{"accountServiceHomePage":"http://c
loud.scorm.com/","accountName":"anonymous"}],"objectType":"Agent"},"object":{
"id":"http://scorm.com/pong/beatJim ingram","definition":{"name":{"en-
US":"Beating Jim ingram at Rustici
PingPong"}},"objectType":"Activity"}},{"id":"7632a386-e13b-4470-ab0b-
a36960070dc7","actor":{"name":["Chris
hooks"],"mbox":["mailto:chris.hooks@scorm.com"],"objectType":"Person"},"verb"
```


Fig. 1 - Sample Tin Can statements available at SCORM public cloud

The drawback with the existing form of experience statements is that they were pre-defined by the content creators and embedded into the appropriate topic levels inside the learning path. These statements only convey the information about the nature of objects experienced by the learner without getting much into the background details of the learner and the type of the LO utilized. This prevents the LMS from knowing more about the type of objects that can cater a particular category of learners.

However, in a typical classroom based environment, the tutor is aware of the type of contents that can cater different categories of learners by having interaction with them. This helps the tutor to take corrective action by providing the alternative content to those learners for whom the given content has not catered. In order to emulate the real classroom based learning over the e-learning environment, the learning experience should be modeled appropriately so that they can convey the changing needs of the learners over their learning cycle (Dolog *et al.*, 2004). Also, the system must be capable of analyzing these learning experiences in real time in order to recommend the most appropriate learning content for its learners. In a typical learning environment as the one considered in this study, where the learners take online learning in addition to face to face classes, the collaborative recommendation based on the learning experiences is considered to be the best option for recommendation.

3 Proposed System

In this paper, we propose Learning Experience Modeling System (LEMS) that represents the learning experiences in a multidimensional form using the snowflake schema for its data representation. The learning experiences are recorded under the facts table, whereas, the dimension table maintains the information about the various dimensions on which the facts are generated. The idea behind isolating the learning experience facts from the other dimensions about the learner / LO is that the facts could be used along with only the key dimensions of interest at a specific search scenario. This prevents the need for processing huge amount of data each time, thereby reducing the time and memory required to perform on demand analysis.

The advantages of snowflake schema for representing the learning experiences data are as follows,

1. Simple facts on the learner's learning experiences are stored under the facts table.
2. Less repetition of data as the duplicates are removed at the dimension tables itself.
3. Multi dimensional view of facts that helps to perform real time analysis
4. Easy integration with the available OnLine Analytical Processing (OLAP) tools for effective analysis.
5. Separation of facts from dimensions in order to make the independent of each other.
6. Support for dimensionality reduction in cases where minimal information is enough (Berson & Smith, 1997).

The snowflake schema of LEMS in figure 2 highlights the learning experiences recorded based on the dimensions of the learner profile (learner background, skills, preferences, knowledge) and the LOM. The learning experience facts table records the experiences by mapping the LOM attributes of the LO utilized with that of the profile attributes of the learner who used it.

Since the experience of a learner with respect to a LO is purely based on the learner's background and the type of LO that has been used, the mapping between the LOM and the learner profile attributes would greatly help to make precise statements about likes and dislikes of a learner. However, when such facts are generated by considering the all the possibilities of LOM attribute with the LP attributes, then multiple statements that conveys the detailed information about the LO utilization can be generated.

The rules for mapping the LOM and the learner profile attributes are explicitly specified in Resource Description Framework (RDF) format based on the requirements of the learning environment. For example, if the learner is a beginner and (s)he liked an animation on sorting, then the triples generated by LEMS are as follows, <learner X experienced animation>, <beginner likes sor-

ting animation>, <novice_learner utilized animation>, <slow_learner likes flash animation>, <Learner x understood bubble sort>, <native language learner used author Y's content> etc. These experience statements generated by the LEMS are stored inside the LRS which is modeled as a multidimensional data cube.

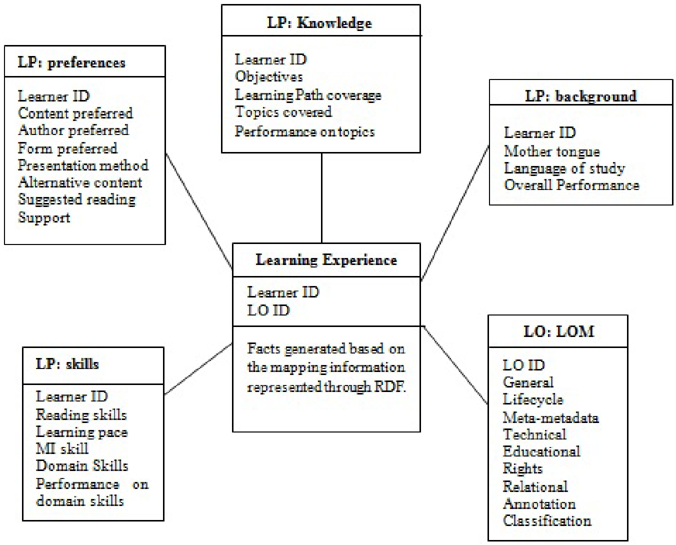

Fig. 2 - Snowflake schema for representing learning experiences

The architecture of LEMS given in figure 3 showcases the steps involved in generating the experience statements through LEMS. The backend of LEMS has a LOR that stores granular LOs created based on the LOCAI (Learning Object Content Assembly Interface) (Raghuveer & Tripathy, 2012). These granular objects have their own metadata that represents the different aspects of the object.

The interface of LEMS takes the learner query keyword and forwards it to the search subsystem which retrieves the LOs that matches with the query keyword. The retrieved LOs are then filtered based on the learner's requirements derived from real time analysis of the LRS. Finally, the LOs are reordered and presented to the learners in decreasing order of the extent to which they can cater the learner's needs and preferences. When the learner utilizes the object, the object filter (OF) gives the feedback to the mapping engine which in turn fetches the LOM of that object used and the attributes of the learner profile to generate experience the statements by mapping them.

The mapping engine maintains the criteria for mapping the LOM with the learner profile parameters in the form of RDF templates. This template stores the mapping definitions of the form <Learner Profile attribute, mapping verb, LOM attribute>. For example, if the system has to record information on the type of content preferred by the slow learners, then the RDF rule template has an entry like <learning pace, experienced, content type>. Similar RDF templates can be created for the learners with specific type of requirements so that the experience statements generated could be used in a better way of recommending LOs for them.

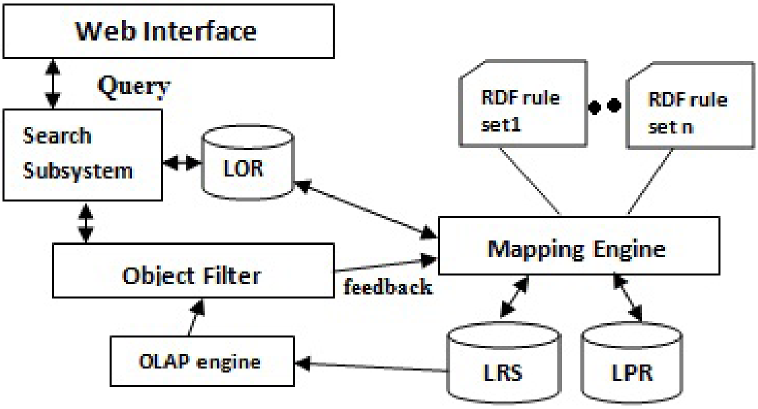

Fig. 3 - Architecture of LEMS

The RDF mapping information is created explicitly in order to support mapping of only the necessary attributes of learner profile and LOM based on the requirements of the e-learning environment. The experience statements generated based on the mapping criteria are stored in the LRS which is a multidimensional data cube as given in figure 4.

The cube in figure 4 has three major dimensions viz. the LOM of the LOs utilized by the learners in a specific subject domain, the learner profile categories of the learners who have utilized the LOs and the learning experience facts which is the mapping between the Learner profile categories (background, preferences, skills and knowledge) and the LOM.

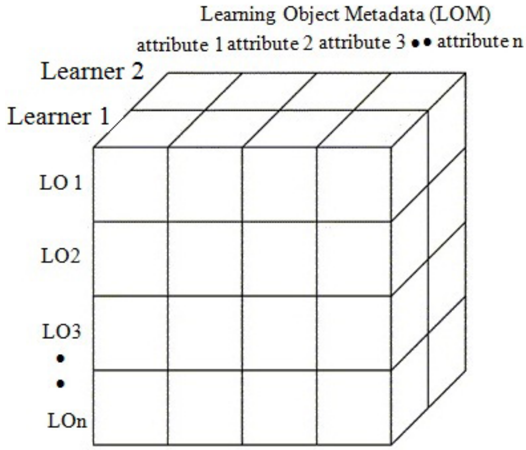

Fig. 4 - Representation of learning experience statements using a multi dimensional cube

3.1 On- Demand analysis of learning experiences

Whenever the learner queries for a particular topic, the learning experience statements inside the LRS are analyzed to obtain the information on the type of LOs that the similar learners have utilized. Such an analytical processing done in an on-demand basis helps to determine the learning pattern of the learners with respect to a particular topic of the subject domain (Negash, 2004). The learning pattern thus obtained paves the way for deriving the most appropriate LOM that can cater the learner's current learning requirement. The steps involved in on- demand analysis of the learning experiences are as follows. First the learner query keyword is obtained and the LOs are retrieved. The next step is to identify the learners who have a similar profile as that of the current learner who raised the query. The overall similarity ($Sim_{u,v}$) between the profile of the current learner 'u' who raised the query and that of a leaner 'v' is computed as in (1),

$$Sim_{u,v}\ overall = \{0.5 * sim_{u,v}\ knowledge + 0.3 * sim_{u,v}\ preferences + 0.2 * sim_{u,v}\ skills\} \tag{1}$$

where, 0.5, 0.3 and 0.2 are the weights given for the knowledge, skills and preference categories of the learner profile respectively. Finding this overall similarity among the profiles begins with a pre-processing stage where the system first identifies the set of learners who have undergone the topic which the current learner is about to take up. This stage retrieves only the profiles of the learners who have utilized the LOs on that particular topic. The retrieved profiles are then compared with the current learner's profile to find the extent of similarity amongst them. Since the extent of similarity can't be expressed as a binary value (0 or 1) (Deza & Deza, 2009), it is computed using the Cosine similarity function which calculates the extent of similarity between the attributes of the learner profile categories (Appendix) as a scalar value in the range of -1 to 1. Where -1 refers to the negative similarity which means the entities of comparison are quite opposite to each other and 1 represents that the entities are exactly similar to each other. Any other value between 0 and 1 conveys the partial similarity between the entities being compared.

$$Similarity_{u,v} = \cos(\theta) = \frac{\sum_{i=1}^{n} u_i \times v_i}{\sqrt{\sum_{i=1}^{n} {u_i}^2} \times \sqrt{\sum_{i=1}^{n} {v_i}^2}} \tag{2}$$

In (2), u and v represent the profile of the learners being compared. The vector 'u' represents the learner profile parameters of learner 'u' $\{u_1, u_2 \ldots u_n\}$ and the vector 'v' represents the learner profile parameters of 'v' $\{v_1, v_2 \ldots v_n\}$ where 'n' is the total number of parameters in the learner profile. These vectors are compared by finding the dot product of their attribute values. (The detailed list of attributes of learner profile of the learners is given in Appendix A.)

In some cases the profile parameter values were used straightaway as their values themselves represent their significance e.g. performance attributes. In some other cases the importance given to the attribute value is considered for computing their similarity. Eg. author preferred. (here, if the usage statistics of the LO highlights a particular author as the most preferred in the author queue, then that is being compared to the position of the preferred author of the other learners.)

Once the similarity is computed as a scalar value, the profiles of the top 10 similar learners determined based on this value are considered for the analysis. The learning experiences of these 10 learners are analyzed to derive the LOM requirement of the current learner. The derived LOM requirements are then used to re-rank the retrieved LOs before they are presented to the learners.

3.1.1 The Cross Dimensional Slicing (CDS) algorithm

The advantage of multi dimensional data representation is to have the overall information organized appropriately under different classes. This helps to create dimension based views on data according to the learner's requirement.

Another highlight of representing the learning experiences using the data cube is that the cube can be dynamically sliced through a specific dimension in order to determine the learning pattern of the learners with respect to that dimension. The cross dimensional slicing algorithm presented here retrieves the most appropriate LOs for the current learner based on the experiences of the similar learners at that learning instance.

The Algorithm

Begin

Step 1: Let 'w' be the query keyword issued by learner 'x'.

Step 2: Retrieve the LOs that matches with the keyword 'w'.

Step 3: Derive the similarity between the profiles of the current learner x and the other learners using the similarity function in (2).

Step 4: Filter the LRS to obtain the top 10 records based on the values of their similarity.

Step 5: For each LO utilized by the similar learners, increase the weight of LOM attributes by multiplying it with the similarity value.

Step 6: Finally aggregate all the metadata attributes and their values in the decreasing order of their weights.

Step 7:Re-rank the LOs retrieved based on the extent to which they can cater the learner's requirements derived through the aggregated metadata.

Step 8: Obtain the information regarding the usage of the recommended LOs by the current learner 'x' and update the domain specific learner profile in order to reflect the changing requirements of the learner.

Step 9: Generate the learning experience statements based on the mapping information available in RDF files and store them inside the LRS data cube.

End

The advantages of CDS algorithm are threefold; firstly, the algorithm formulates the LO requirement for a specific learner dynamically. Second, the algorithm increases the utilization of newly added LOs over the period of time as it retrieves the suitable LOs based on $LOM_{aggregate}$ rather than only retrieving the exact LOs that the similar learners have utilized. This prevents sidelining of newly added LOs due to the fact that they were underutilized by the learners. Finally, the fuzzy based similarity detection in (4) gives importance to the profiles based on their extent of similarity with the current learner's profile rather than treating them equally. This benefits the learners as it helps to arrive at more specific, custom made LO requirement that can cater the current learner precisely.

4 Experimental Results

An experiment was conducted on a sample set of 182 learners at UG level (doing second semester of B.Tech. in Electronics and Communication Engineering), divided into three batches viz batch1, batch2, and batch3 with a strength of 60, 63, 59 respectively. A total of 285 LOs stored inside the LOR under 73 topics of the "Data structures and algorithms" subject domain was used for the study. The batch1and batch2 learners took the course on day 1 and day 2 respectively on the first 25 topics and their learning experiences were recorded inside the LRS. Similarly, the batch3 learners have utilized the LOs under the first 25 topics on day 3. The learners of batch1 were initially classified as slow learners and average learners based on their profile parameters under the categories like skills and background. Also, the batch1 learners were all provided the LOs retrieved only based on the information available on their profiles. The quantum of data involved in the study is given in table 1. The initial number of slow learners in batch1 was 21 and in batch2 and bacth3 are 23 and 18 respectively. As the learners took up the LOs, the performance of the learner on each LO was used as a measure to update their learning pace. (The learners with an average score less than 2.5/5.0 on the LOs utilized was considered as slow learners).

When the batch2 learners took the course, they were recommended the LOs based on the experiences of similar learners in batch1. Similarly for batch3 learners the experiences of batch1 and batch2 learners were used together to recommend the LOs. The utilization of LOs retrieved at top k positions (Manning *et al.*, 2008) (where k=5 or 3 or 1) based on the recommendation of LEMS is calculated as follows,

Utilization at k = Number of top k objects utilized / total number of objects utilized

Table 1
QUANTUM OF DATA INVOLVED IN THE STUDY

Property	Batch1	Batch2	Batch3	Average
Total no. of learners	60	63	59	60.6
Initial no. of slow learners	21	23	18	20.6
Fraction of slow learners (initial) (row2/row1)	0.35	0.36	0.30	0.34
Final no. of slow learners	18	12	7	12.3
Fraction of slow learners (final) (row4/row1)	0.3	0.19	0.11	0.20

Table 1 highlights the information about the slow learners in each batch at the beginning as well as at the end of the study. (LEMS does not recommend the LOs for batch1 learners as it does for batch2 and batch3 learners.) Table 2 gives a complete picture on the utilization of LOs retrieved at top k positions by the slow learners of batch2 and batch3 in comparison with the learners of batch1.

5 Discussions

The results obtained from this study have highlighted two important aspects. First, the fraction of LOs utilized from top k results (particularly at k=1) has improved in batch2 and batch 3 when compared to batch 1. This shows that the recommendation based on the learning experiences found to be effective as the learners have utilized more LOs that were utilized by similar learners. Also, the utilization of LOs at k=3 and 5 has decreased in batch2 and batch3 which in turn highlights that the learners were mostly satisfied with the best result recommended by the LEMS.

Table 2
UTILIZATION OF LOS BY SLOW LEARNERS BASED ON THE RECOMMENDATION OF LEMS

k value	Number of times the LOs in top k utilized			No. of times the LOs at k utilized / total no. of LOs utilized all together		
	Batch1	Batch2	Batch3	Batch1	Batch2	Batch3
k = 5	190	123	78	0.431818	0.293556	0.236364
k = 3	138	144	109	0.313636	0.343675	0.330303
k = 1	112	152	143	0.254545	0.362768	0.433333

Figure 5 shows the fraction of LOs utilized at different k values (1, 3 and 5). The number of LOs utilized at k=1 was less in batch 1 as these learners were not recommended the LOs and the selection of content was left to the choice of the learners. Whereas, the increase in the fraction of LOs utilized at k=1 in batches 2 and 3 shows that the recommendation based on the experiences of the peer learners has greatly helped in identifying the right content for the learners in first place.

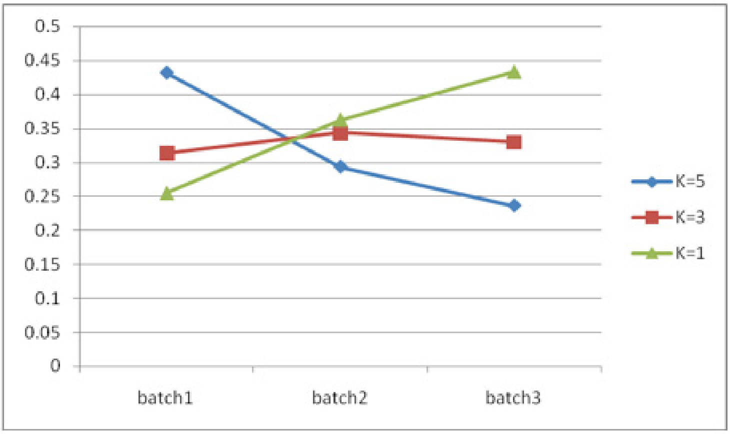

Fig. 5 - The number of LOs utilized based on recommendation at different k values

Second, the total number of slow learners in batch2 and batch3 got reduced considerably when compared to that of batch1 (figure 6). This is due to the fact that the slow learners of batches 2 and 3 were recommended the LOs that worked well for the slow learners of their earlier batches.

However, the reason for the fraction of LOs utilized at positions k=3 and k=5 can't be reduced greatly because even the learners with similar profiles at times may not prefer a similar object as there may exist some diversity amongst their learning methods.

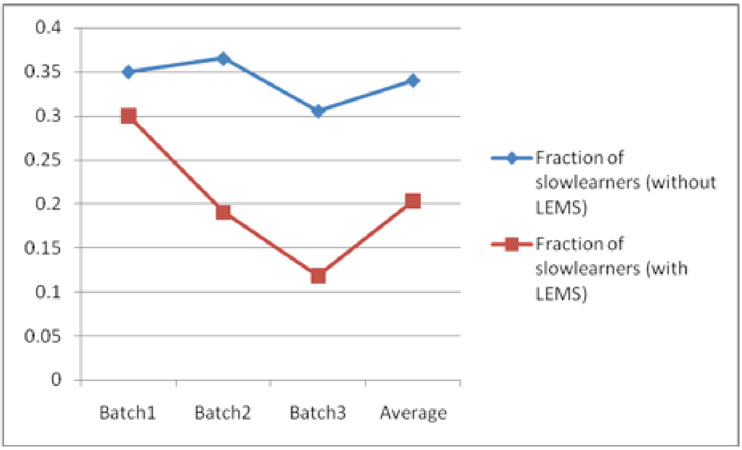

Fig. 6 - The fraction of slow learners across the batches before and after using LEMS

Altogether, the learning experience based LO recommendation by LEMS has increased the utilization of appropriate learning content as well as reduced the number of slow learners of an e-learning environment. However, the system picked up slowly in improving the accuracy of recommendation (from batch2 to batcht3) as the total number of learning experiences under the LRS was less in this pilot study. In real-time e-learning environments like Massively Open Online Courses (MOOC), the large volume of learners undertaking the courses would help in generating huge amount of learning experiences which in turn improves the accuracy of recommendation.

Conclusions and Future Research

The modeling of learning experiences has greatly helped in understanding the learner's needs and capabilities more precisely at the micro level. The experience statements generated through the LEMS has given multidimensionality appeal to the learning experiences so that they can be analyzed dimension wise. The need for retrieving the LOs based on the learner's requirement at different instances of learning cycle was addressed by deriving the LOM from the experiences of similar learners.

The cosine similarity function used was helpful to determine the proximity among the learners. The cross dimensional analysis of the LRS has given the insight into the learning patterns of the learners which made the task of recommendation easier. The recommendation of LOs based on the peer learners' experiences has improved the scope of LOs towards reaching the appropriate section of learners for whom it is intended. Altogether, the experiences based LO recommendation has addressed the issues related to the lack of assistance in e-learning environments by making the environment a self sustained one in handling its learners. However, if the chances of having learners with similar profiles are less in an e-learning environment, due to the fact that the environment is open for various categories of people like employees, students, homemakers, elders, etc. there has to be a different method for understanding the requirements of individual learners and their learning patterns for effective retrieval of LOs.

The future work is aimed at analyzing the cases where the recommendation based on the peer learners' experiences has failed, in order to determine unique recommendation policies to address the learner's requirements. Also, the recommendation of LOs is to be extended such that predictions could be made on the future scope for the learners on a particular subject domain based on their skills and performances. The focus of our study is also on identifying the ways of generalizing the learner profiles to form dynamic learning communities with common learning interests as this would enhance the leaner collaboration over the e-learning environment.

REFERENCES

Alexander, B. (2006), *Web 2.0: A new wave of innovation for teaching and learning?*. Educause review, 41(2), 32.

Bergeron, B. (2014), *Application of Learning Record Stores and Other Forms of Electronic Competency Records in Modeling Competency Degradation*. Medicine Meets Virtual Reality 21: NextMed/MMVR21, 196, 23.

Berson, A., & Smith, S. J. (1997), *data warehousing, data mining, and OLAP*. McGraw-Hill, Inc..

Dalsgaard, C. (2006), *Social software: E-learning beyond learning management systems*, European Journal of Open, Distance and E- Learning, 2006(2).

Deza, M. M., & Deza, E. (2009), *Encyclopedia of distances*. Springer Berlin Heidelberg.

Dolog, P., Henze, N., Nejdl, W., & Sintek, M. (2004), *Personalization in distributed e-learning environments*, proceedings of the 13th ACM international World Wide Web conference on Alternate track papers & posters, 170-179.

Experience API Working Group (2013), *Experience API*, the Advanced Distributed Learning (ADL) Initiative, Apr.2013.

Gibson, A., Kitto, K., & Willis, J. (2014), *A cognitive processing framework for learning analytics*, proceedings of the fourth ACM International Conference on Learning Analytics And Knowledge, 212-216.

IEEE 1484.2.1, *Standard for Learning Technology—Public and Private Information (PAPI) for Learners (PAPI Learner)—Core Features*. Available online at http://metadata-standards.org/Document-library/Meeting-reports/SC32WG2/2002-05-Seoul/WG2-SEL-042_SC36N0175_papi_learner_core_features.pdf

IEEE Learning Technology Standards Committee (2002), *Draft standard for learning object metadata*. Accessed July, 14, 2002.

LIP, 2001, IMS LIP (2001), *IMS Learner Information Package Specification*, the Global Learning Consortium.

Manning, C. D., Raghavan, P., & Schütze, H. (2008), *Introduction to information retrieval*, Cambridge university press.

Negash, S. (2004), *Business intelligence*, The communications of the Association for Information Systems, 13(1), 54.

Pazzani, M. J., & Billsus, D. (2007), *Content-based recommendation systems*. In The adaptive web. Springer Berlin Heidelberg.

Poltrack, J., Hruska, N., Johnson, A., & Haag, J. (2012), *The next generation of scorm: Innovation for the global force*. The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC), National Training Systems Association, 1(1).

Qiao, Z., Zhang, P., Cao, Y., Zhou, C., & Guo, L. (2014), *Improving Collaborative Recommendation via Location-based User-item Subgroup*. Procedia Computer Science, 29, 400-409.

Raghuveer, V.R, & Tripathy, B.K (2012), *An Object Oriented Approach to Improve the Precision of Learning Object Retrieval* in a Self Learning Environment. Interdisciplinary Journal of E-Learning and Learning Objects, 8(1), 193-214.

Raghuveer, V.R., Tripathy, B.K. (in press), *Affinity based Learning Object (LO) retrieval in an e-Learning environment using Evolutionary Learner Profiles*. Knowledge Management & E-Learning: An International Journal (KM&EL).

Richards, G., McGreal, R., Hatala, M., & Friesen, N. (2002), *The evolution of learning object repository technologies: Portals for on-line objects for learning*. International Journal of E-Learning & Distance Education, 17(3), 67-79.

Siemens, G., & Long, P. (2011), *Penetrating the fog: Analytics in learning and education*. Educause Review, 46(5), 30-32.

Warren, S. J., Lee, J., & Najmi, A. (2014), *The impact of technology and theory on instructional design since 2000*, Handbook of research on educational communications and technology, Springer, New York.

Wiley, D. (2000), *Getting axiomatic about learning objects*.

World Wide Web Consortium. (2013), *Resource Description Framework Specification*.

http://site.watershedlrs.com/

http://tincanapi.com/

http://www.bookonpublish.com/

https://www.tappestryapp.com/

Appendix

GLOBAL LEARNER PROFILE CATEGORIES AND ITS ATTRIBUTES

GLP Category	Attributes
Background	Name, gender, date of birth, nationality, address, phone number, email, medium of study
Skills	multiple intelligence skill, reading skills, use of technology, learning pace
Stated Preferences	language, content type, presentation mode, format preference, connection type, device
Domain knowledge	domains exposed, exposure level, domains of interest, suggested domains, overall performance on each domain, scope for further study

GLOBAL LEARNER PROFILE CATEGORIES AND ITS ATTRIBUTES

LLP Category	Attributes
Objectives	Specific objectives predefined (based on course outcomes)
Learning Path coverage	Learning path hierarchy, List of topics covered, percentage of completion, topic wise performance
Domain skills obtained	Total no. of skills attained, Skill name, topic, LO used, performance on skill.
Evolving Preferences	Author, content type, language, format
Explicit feedback	The feedback given by the learner on a specific LO