We are IntechOpen, the world's leading publisher of Open Access books
Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Application of Data-Mining Technology on E-Learning Material Recommendation

Feng-Jung Liu and Bai-Jiun Shih
Tajen University
Ping Tung, Taiwan

1. Introduction

"Information overload" problem is even more emphasized with the growing amount of text data in electronic form and the availability of the information on the constantly growing World Wide Web (Mladenic et. al., 2003). When a user enters a keyword, such as "pencil" into a search engine, the result returned is often a long list of web pages, many of which are irrelevant, moved, or abandoned (Smith et. al., 2003). It is virtually impossible for any single user to filter out quality information under such overloading situation (Shih et. al., 2007).

Designing appropriate tools for teaching and learning is a feasible approach to reduce the barriers teachers might encounter when adopting technology in their teaching (Marx et. al., 1998; Putnam & Borko, 2000). With potentially hundreds of attributes to review for a course, it is hard for the instructor to have a comprehensive view of the information embedded in the transcript (Dringus & Ellis, 2005). Computer-based systems have great potential for delivering learning material (Masiello, Ramberg, & Lonka, 2005), which frees teachers from handling mechanical matters so they can practice far more humanized pedagogical thinking. However, information comes from different sources embedded with diverse formats in the form of metadata, making it troublesome for the computerized programming to create professional materials. (Shih et. al., 2007).The major problems are:

(1)Difficulty of learning resource sharing.

Even if all E-earning systems follow the common standard, users still have to visit individual platforms to gain appropriate course materials contents. It is comparatively inconvenient.

 (2) High redundancy of learning material.

Due to difficulty of resource-sharing, it is hard for teachers to figure out the redundancy of course materials and therefore results in the waste of resources, physically and virtually. Even worse, the consistency of course content is endangered which might eventually slow down the innovation momentum of course materials.

(3) Deficiency of the course brief.

It is hard to abstract course summary or brief automatically in efficient way. So, most courseware systems only list the course names or the unit titles. Information is insufficient for learners to judge quality of course content before they enroll certain courses.

To solve the problems mentioned above, we propose an automatic inquiring system for learning materials which utilize the data-sharing and fast searching properties of LDAP. Our system not only emphasizes friendly search interfaces, but also excavates the association rule from log data of learning activities. Meanwhile, collaborative filtration is employed to improve the reliability of the searching results.

However, the result lacked meaningful associations or visual accessibility in current approach. We then intend to employ some AI techniques to improve the results. A topic map seems to be a good solution. Fisher, Wandersee and Moody (2000) said that a topic map presents a group of assets in a network structure; the knowledge network is completed by interlinked conceptual nodes, and spread out in a framework. It not only involves sets of concepts but also the organization of concepts in terms of their inter-relationships.

In this Chapter, we will give a brief review of what has been done on addressing the information overload problem on e-Learning, and explain how related techniques were applied. After then, a detailed description of the system implementation will be provided, followed by pedagogical application suggestions to possible alternatives for the integration of the technology into the learning field. Both the technical and educational evaluations of the system were discussed. At last, our conclusion brings some ideas and suggestions to the alternative possible integration of technology into learning in all fields.

2. Literature Review

A key factor of hypermedia-based learning is customizable cognitive style as it suffices users' information processing habits, representing individual user's typical modes of perceiving, thinking, remembering and problem solving. Cognitive psychologists recognize that knowledge has a basic structure which presents the inter-relationships between concepts. Anderson (1980) distinguished knowledge into declarative and procedural types to identify their characteristics as abstract or practical functions (Anderson, 1980). In understanding knowledge, Ausubel (1968) provided two strategies including progressive differentiation and integrative reconciliation for making meaningful learning focusing on the systematic methods to learning (Ausubel, 1968). The implication of these theories is two-fold; one, knowledge has an internal structure to sustain itself; two, systematic retrieval and understanding of knowledge is an effective method for learning (Shih et. al., 2007).

In order to effectively use technology to assist the education process, helping learners to collect, process, digest, and analyze the information, we introduce data mining, association and collaborative filtering technologies, describing how the technology facilitates the processing of data, and how the dynamic map can achieve the pedagogical goal.

Generally speaking, there are two approaches to organize and present knowledge: top-down and bottom-up. Top-down approaches consist of supervised-learning. They require human-experts to define knowledge ontology (Noy & McGuinness, 2001) and taxonomy (Bruno & Richmond, 2003). Accordingly, the machine classifies the knowledge by those predefined rules, based on co-occurrence and ANN (Artificial Neural Network). Inversely, bottom-up approaches are fully-automatic and represent unsupervised-learning, including SOM (Kohonen, 2001) and knowledge maps. However, most applications took hybrid approaches, that is, a human expert (SME: subject material expert) to predefine ontology and taxonomy and then clustering and organizing the knowledge by using ANN or SOM. This is sometimes called reinforcement learning. (Bruno & Richmond, 2003)

2.1 Data Searching, Storage and Retrieval

The Lightweight Directory Access Protocol, LDAP, is an application protocol for querying and modifying directory services running over TCP/IP. LDAP was originally intended to be a "lightweight" alternative protocol for accessing X.500 directory services through the simpler (and now widespread) TCP/IP protocol stack. The advantages are:

(1) fast searching: LDAP utilizes the properties of data hierarchy, caching technology and the innovative index technology to offer fast inquiry service;

(2) Extendable Data schemas: The data schema mainly describes and defines the attribute of entries in the directory tree. LDAP allows users to define data schemas by themselves and let schema specification more flexible;

(3) Multiple access permissions and data encoding: Except that be able to establish the access permissions according to users' specifications individually, LDAP also supports some security mechanisms, such as SSL (Secure Socket Layers), TLS (Transport Layer Security) and SASL (Simple Authentication & Security Layer);

(4) Suitable for the inquiry of a large amount of data: The directory database is designed under the assumption that the frequency of reading is greater than frequency of writing. It can improve the usability and dependability of data by duplicating data extensively.

XML and Java technology are recognized as ideal building blocks for developing Web services and applications that access Web services. JAXB (Java Architecture for XML Binding) is an XML binding model that defines the way of automatic mapping XML documents into objects in a programming language. Two major processes, marshalling and unmarshalling, take care of the mapping between Java objects and XML documents, which makes JAXB surpass traditional SAX and DOM approaches. Its advantages are:

(1)Simplicity: It is Java procedure too to derive the classification (Schema-Derived Classes & Interfaces) through the outline that Binding Compiler compiles out, so does not need to deal with XML file by oneself, and does not deposit and withdraw the content tree without according to the order;

(2)Extensibility: The programmer can revise the schemas and derive the classification independently, and let the procedure accord with systematic requirements even more. Additionally, when XML Schema is changed to some extents, it just needs to recompile Schema, and increase some more procedures newly, instead of needing to revise the original procedures,

(3)Efficiency: Because all of data of the content tree are produced of JAXB according with the definition of XML Schema, there not exist any invalid methods or objects. Even it could use the Unmarshaller class to verify whether XML file is effective.

2.2 Data Association

The Apriori algorithm(Agrawal, 1993) is one of the most representative algorithms of association rule. Its key steps are: (1) produce the candidate set from the database, and then find out the largest item-set according with the minimum support degree; (2) find out the items from the large item-set derived in previous step, in accord with the minimum confidence degree.

However this approach is time-consuming because the database are usually scanned too many times. Therefore, many algorithms, like the DIC algorithm (Brin et. al., 1997), DHP

algorithm(Park. et. al., 1995) , are proposed successively to improve the performance. We suggest using association rule mining techniques to build an e-learning material recommendation system that intelligently recommend on-line learning activities or shortcuts in the course web site to learners based on the actions of previous learners to improve course content.

2.3 Collaborative Filtering

Collaborative filtering is one of recommendation mechanism for personage, also called "people to people correlation". Currently, it is applied to all kinds of e-commerce extensively to infer user's interest in other products or the service through the analysis of user's materials or the behavior. Currently, the filtration technologies are classified into the following three methods:

(1)Rule-based filtration:
By the questionnaire or other ways to obtain the data, such as user's preference, or interest, etc. which will be made use of the reference as recommending.
(2)Content-based filtration:
The system recommends other relevant contents according to the user's selected content. And, the recommended content mostly were used and hived off or consulted by other user's experience before.
(3)Activity-based filtration:
By collecting user's activity information, infer the relation of some contents. Such mechanism is usually applied on these websites without member system.Prior works in collaborative filtering have dealt with improving the accuracy of the predictions. In the proposed system, we adopted the collaborative filtering instead of the complicated content processing, and provided recommendation on possible keywords of contents with the accuracy.

3. Implementation

We utilized the techniques of the LDAP directory server and JAXB to reduce the load of development of search engine and the complexity of parsing the contents (Li et al., 2005). The data-mining techniques will be applied to support the e-learning materials recommendation. Relevant techniques including LDAP, JAXB, association rule and collaborative filtering are illustrated below.
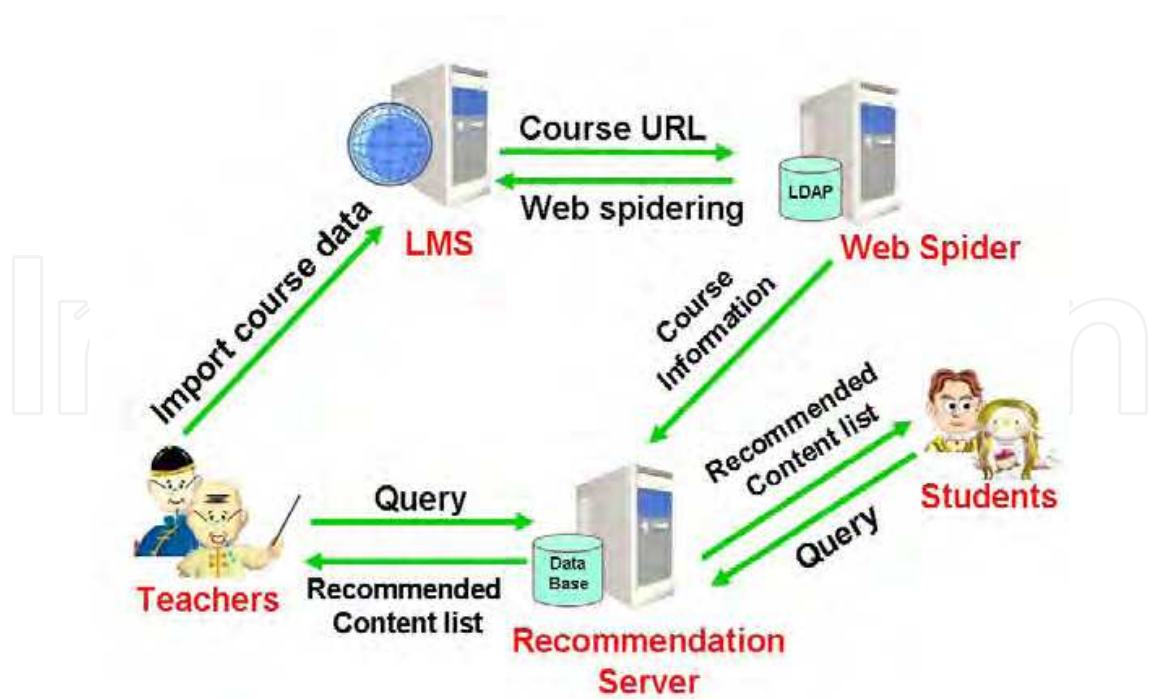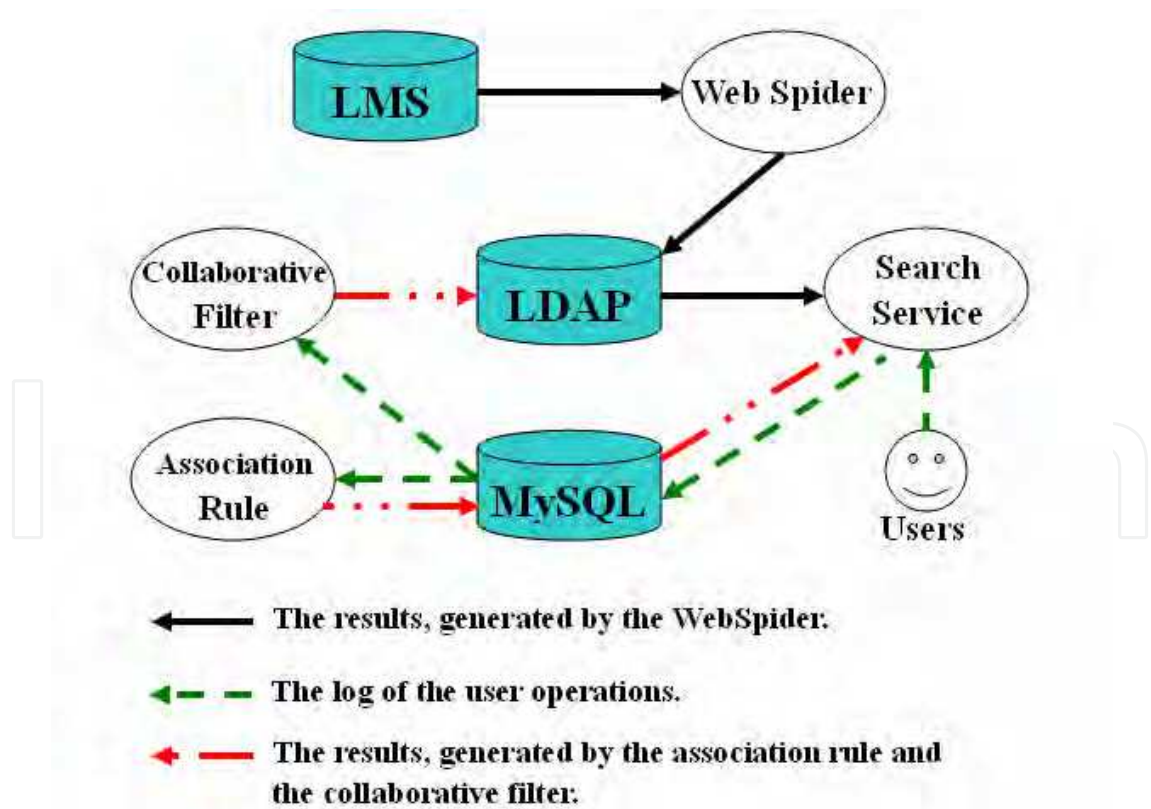
Fig. 1. System architecture.

Fig. 2. The data flow of the recommendation system.

In the recommendation system, it consists of three participants: learning management system, web spider and recommendation server as shown in Figure. 1. Learning management system, LMS, not only manages and provides the content services, but also registers the content URL address to web spider for collecting content index data. Through the URL address of content, the web spider will visit and parse all of the content recursively and store resulting contents into LDAP server. This scheme is applicable for all of e-learning content accessible by HTTP protocol. At last, a web-based querying system is presented to the users for friendly learning.

As shown in the Figure. 1, while teachers or content providers uploaded their learning materials to the LMS systems, the background process stored in the LMS will pass the pathname or URL address of the uploaded materials to the web spider-like process for collecting index data. The web spider parses and traces all linked documents of the course contents. As an example of course content made by StreamAuthor package (CyberLink. http://www.cyberlink.com/multi/products/main_7_ENU.html), the content mostly consisted of a lot of HTML document translated from MS Powerpoint formatted files. In implementation, we defined and realized the PptHtmlParser class derived from HtmlParser interface to parse the materials created by the StreamAuthor package. The web spider will recursively visit these links embedded in the parsed documents. In Figure. 2, it describes the data flow in the recommendation system including three data sources.

In implementation, we developed the backend program for collecting user behavior-related data in the JAVA programming language. In addition to the sake of the properties of cross-platform and easy usages, JAVA can directly utilize the abundant packages with powerful functions, such as HttpClient in Apache Jakarta project (http://jakarta.apache.org/commons/httpclient/), HtmlParser (http://htmlparser.sourceforge.net/) created by Derrick Oswald, etc.., is good for reducing the cost of system implementation. Additionally, we adopted the JAXB mechanism to transform the result, generated by HtmlParser process, to the XML documents, fit to the predefined schema as shown in Figure. 3. Because of the unmarshalling mechanism, it is easy to access data elements in XML documents with the minimum burden of programming.

```
<xs:element name="LearningResource" type="LRTy...
<xs:complexType name="LRType">
  <xs:sequence>
  <xs:element name="url" type="xs:anyURI"/>
  <xs:element name="title" type="xs:string"/>
  <xs:element name="topic" type="xs:string" minO...
  <xs:element name="content" type="xs:string" min..
  <xs:element name="oriContent" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

Fig. 3. The schema of JAXB.

The proposed recommendation system is achieved with the four jobs: collecting indexing data, inquiring services, association rule and collaborative filtering as shown in Figure. 2. We will have detailed descriptions as follows.

3.1 Index Data Collection
Collecting data index is similar to other web spider searching engines. The parser will automatically search the complete course content and distinguish the topic from the content body by HTML tags. Moreover, it transform the data into the XML documents compliant to the XML schema by utilizing the Marshaller mechanism of JAXB (Ed Ort et al., 2003) as showed in Figure. 4. Besides, it is easy to validate and access XML documents by JAXB unmarshalling functions by JNDI (java naming and directory Interface). The processed data are stored into the LDAP database at last.

```
                    public static void Html2Xml(HtmlParser hp,String ...

//Create LRType  Object, from the derived class from JAXB
    LRType lrt = new LRType();
    lrt.setUrl(hp.getURL().toString());
    ......
    JAXBContext jc = JAXBContext.newInstance...
    JAXBElement<LRType> lrElement =
    (new ObjectFactory()).createLearningResource(lrt);

//Create the Marshaller Oobject and store the LRType in a file
    Marshaller mar = jc.createMarshaller();
    mar.setProperty(Marshaller.JAXB_FORMATT...
    mar.marshal(lrElement,new FileOutputStream...
                                            ....
}
```

Fig. 4. Parts of the program for JAXB Marshaller mechanism.

3.2 Inquiring services
Usually, learners in the e-learning system concern about learning resources related to the topics. Through the data sharing mechanism of the LDAP, the metadata of the learning materials distributed in several platforms are stored into the same directory database. With such a deployment, teachers can easily inquire the related learning resource and enrich them. The learners can now search through the common interfac for materials orginally distributed among different LMS sites.

Collecting the users' operating behavior, including searching strings, entry links, etc., is another task. We treated each searching string (keywords) issued as a transaction. Then, these transaction logs are used to mine the association rule with user behaviors. Additionally, the usage logs created by students following the resource links will be stored for collaborative filtering.

3.3 Association rule mining

Both the Apriori Algorithm(Agrawal et. al., 1994) and the Tree-based algorithm are employed to develop the association rule for the learning material recommendation system. An itemset can also be seen as a conjunction of a sequence of searching keywords by a learner in the same session. If an itemset satisfied minimum support, then it is a frequent itemset. It means that most of learners used those keywords in the frequent itemset to look up their interested contents. The minimum support threshold and minimum confidence threshold are influent on the reliability of filtered results. If set to the less, the reliability of conducted results will be reduced; otherwise, it will cause the number of the resulted data become too small and fail to infer to the accuracy prediction.

We used the SQL language to filter and sort the large one-items according to minimum support. And then, to make the combination of these filtered items and to calculate the number of transaction records in database which have the same elements as ones of the composed subset. The pseudo code for association rule mining is shown in Figure. 5.

In the association rule, an item-set is called a frequent itemset if it satisfies the minimum support threshold. And more, the reasonable items will be derived if these items satisfy the minimum confidence threshold. As to the settings of the minimum support threshold and the minimum confidence threshold, it is worth mentioning that the settings of the minimum support threshold and the minimum confidence threshold have much influence on the reliability of the filtered results. If these thresholds are set to a smaller value, the reliability of conducted results will be reduced; otherwise, it will cause the amount of the resulted data become too small and cannot infer to the accuracy prediction.

```
d=new array
t="select distinct session as s"
foreach(t as row){
  w="select word where session=row.s"
  foreach(w as item){
    arr=combination(item)
    foreach(arr as i){
      if(i contains d)
        i.count++
      else
        d.add(i)
    }
  }
}
```

Fig. 5. Pseudo code for association rule mining.

3.4 Collaborative filtering

The general collaborative filtration recommends the related information to the users. We used such a mechanism to discovering out and providing the corresponding keywords to these courses. Text mining techniques could also be used to find out the keywords in a context. However, related algorithms are sometimes too complicated to implement. Thus, we inferred the possible keywords of the course contents by analyzing the transaction logs of learners' searching strings. The steps of the proposed mechanism are shown as following.

```
u=new array
t=select distinct url
foreach(t as row){
  u.add(row.url)
  u[row].k=new array
  w=select word where url=row.url
  foreach(w as key){
    if(key contains u[row].k)
      u[row].k.count++
    else
      u[row].k.add(key)
  }
}
```

Fig. 6. Pseudo code for collaborative filtering.

Step 1: Calculate the frequency number of keywords used to inquire the preferred subjects by learners.
Step 2: By utilizing the result of the step above, filter the items by minimum threshold, and to store into LDAP server.

In the system implementation, the association rule is mainly adopted to find out the relations between these keywords learners used for searching the content. And, the collaborative filtration is applied to automatically filter the correct keywords of each course. The psuedo code for collaboratve filtering is shown in Figure. 6. Such mechanisms will greatly benefit learners on searching out their interested materials quickly and correctly. We believe that as the amount of material grows, the better the performance of the content the navigation recommendation system will provide to the learners will become.

4. Results

In our system, main page shows a simplified table of contents with multiple levels of embedded categories. Learners can search contents by using multiple keywords concurrently like in regular search engine. Meanwhile, queries based on content creator, topic, content body are also allowed. Users start from the first-level keyword search, the related material will be tabulated just like the regular search engine. Below each topic, the collaborative filtered keywords will be presented. Users can follow the link to reach

associated topics (course material). In this way, users can recursively trace down the topic tree with virtually infinite search levels.
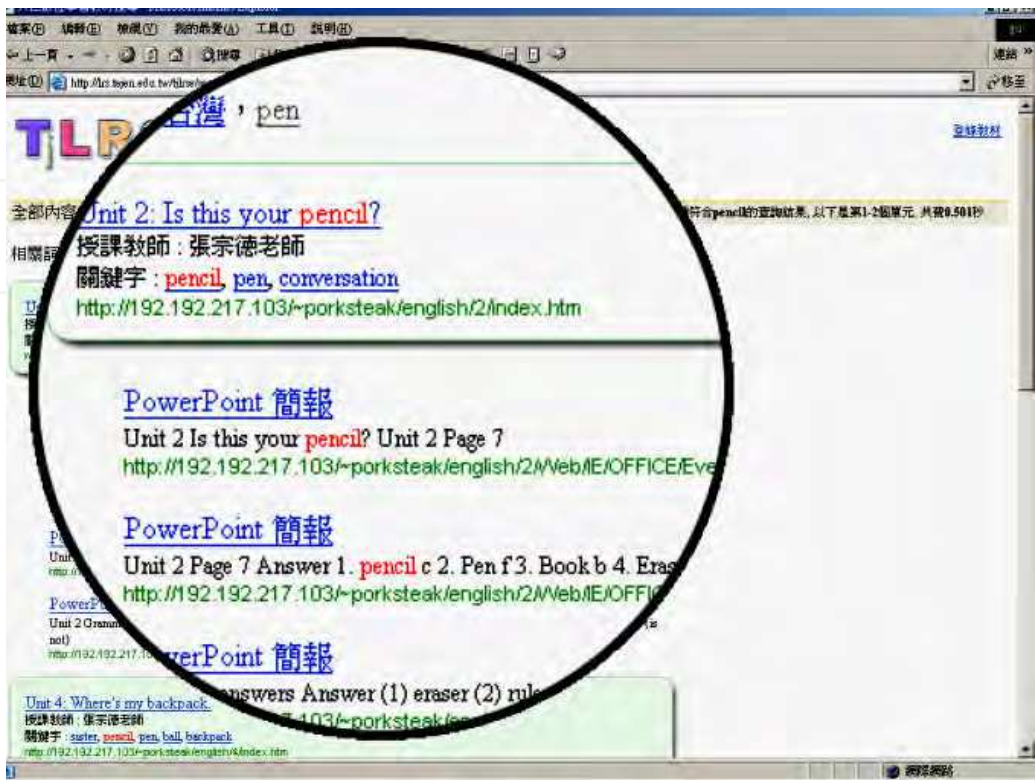

Fig. 7. The result conducted by searching with the "pencil" word.

Fig. 8. The associated words with the "pencil" word.

All results are presented in XML form and comply with the SCORM (SCORM 2004. http://www.adlnet.gov/) requirement. In Figure. 7, we demonstrate a search result for the keyword "pencil". Users first type in the keyword and the system reposes with "Unit 2: Is this your PENCIL?", which is a unit with the title containing target keyword. Below that, two keywords, "pencil" and "Conversation", defined by the teacher of Unit 2 or filtered by collaborative filtration, are cited also. Users can click on the hyperlinks and read the course content or explore the recommended topics.

Meanwhile, the system had found related topics generated from the user log. We can easily see these related topics in our system, previous users are interested with "pencil" and the teacher's name (of course, learners can trace down the link if he/she want to). As shown in Figure. 8, it shows the associated words with the "penceil" word on the top of screen.

While teachers upload their learning materials to the LMS systems, it will pass the pathname of uploaded material to the web spider for collecting index data. Then, we adopt the JAXB mechanism to transform the result generated by HtmlParser process into XML documents fitting to the defined schema. Bye Unmarshalling mechanism, it is easy to access data elements in documents.

There are currently more than 28 URLs of courses registered in the recommendation system. It contains more than 574 course units in total. According to the usage logs, users get into their content search by 2~3 keywords (averaged 2.39). And, each query request spends about 0.65 seconds.

To access the association efficiency, we observe the usage log in one hour period and found that 317 out of 517 queries followed system recommendation links, that is, approximately 60% or two-third of the users followed the system recommendation. This fact also implies a high "hit-rate" of association which we intent to exploit in a future survey.

5. Conclusion and Future Works

The proposed LMS supports systematic learning as well as constructive learning, which can effectively guide users through systematic browsing and inquiry. With this function, it works more as a dynamic researching tool than static learning material. On the other hand, the LMS sustains constructive learning. Although the functionality of a topic map is formulaic and systematic, it is also feasible for task-based learning. From the constructive point of view, learners need resources from multiple sources for the purpose of independent research. The mechanism can suffice the exploration of various learning styles, tendencies of interests, and professional abilities. More importantly, this guidance is not provided by teachers working in the classrooms, but by an autonomous system which is supported by a professional team with a wide array of resources. It turns learning into an information-guided dynamic. Therefore, this material helps users to "discover" new knowledge by presenting explicit and implicit knowledge so that they are able to see ideas and concepts that are most unexpected. This process matches the basic principles of constructivist learning.

In present E-learning scenes it is difficult to integrate all e-learning platforms from various vendors. However, as digital contents explosively grow, a resource-sharing mechanism should not be built solely for material-inquiring service across diverse e-learning LMS platforms. Thus, we proposed an integrated learning activity-based mechanism to assist users with automatic material recommendation. Thus, we established a prototype and

proposed an integrated learning activity-based recommendation system. Currently, we proceed to collect a large number of user learning logs and to evaluate the effectiveness of the material recommendation system. We believe such a deployment will be helpful in achieving the better learning performance and a higher learner's satisfaction.

The techniques of the LDAP and the JAXB greatly reduced the load of development of search engine and the complexity of the content parsing. The material recommendation system improves the learning performances based on the learning activities of previous learners. From the educational perspective, teachers can effectively use this system to collect, process, digest, and analyze information. Learners can gain an overview of the subject matters while surfing the course contents.

From the usability perspective, we see that the LMS can carry out autonomous processing and presentation. It provides teachers and learners with an autonomous abstract environment. Even facing substantial documents, computers can replace human labor to efficiently process the tedious algorithm, but maintain high-level humanistic and professional analysis. Teachers can apply it on the teaching websites, and let the system compile thematic materials for them, saving time of copying and pasting, coding, and rewriting. It is customizable and interactive. The interface interacts with users, opening up layers of information upon every selection and inquiry. Hence, the route taken by every user is different, and the system returns with different results upon every choice. The system has proper support for customized learning. Best of all, it is easily manipulated. Most search engines, websites, and databases are designed to carry documents with different formats, content areas, and inquiry methods. Users who are unfamiliar with each system can get lost in it, and every inquiry can take up much energy and time. By the way, it is easy to use, and simple for users to grasp the key terms generated from the knowledge content. Users do not need to spend much time and energy to get a hold of the main theme of the massive resources. Even learners without prerequisite knowledge or sufficient subject understanding to reach into the depth of content, can still get on the system quickly, starting from the search item.

Educational evaluations are scheduled in the forthcoming year after accumulating threshold amount of user-experience. We plan to investigate the usability and instructional value of the LMS, which includes the presentation of material categories and the trouble-free search for materials; the convenience of data retrieval; the meaningfulness and usefulness of resources; the value of the system's assistance in new knowledge discovery; the usefulness of instructional needs; the appraisal of the overall conceptual presentation of large amount of information; and the level of acceptance and comprehensive understanding of the learning materials, and so forth. At the same time, a focus group performs interview to gather more feedbacks in depth.

6. References

Agrawal, R., and Srikant, R. (1994) Fast Algorithms for Mining Association Rules. Proc. of the 20th VLDB Conference, Santiago.

Agrawal, R., Imielinski, T., and Swami, A. (1993) Database mining: A performance perspective. IEEE Transactions on Knowledge and Data Engineering, 5(6):914-925. Special Issue on Learning and Discovery in Knowledge-Based Databases.

Anderson, J. R. (1980). Cognitive Psychology and Its Implications. San Francisco: W. H. Freeman and Company.

Apache Software Foundation. Jakarta Project-HttpClient. Available at http://jakarta.apache.org/commons/httpclient/.

Ausubel, D. P. (1968) Educational Psychology: A Cognitive View. New York: Holt, Rinehart & Winston.

Berry, M. A., & Linoff, G. S. (2000). Mastering Data Mining. NY: John Wiley & Sons.

Brin, S., Motwai, R., Ullman, J.D., and Tsur, S. (1997) Dynamic Itemset Counting and Implication Rules for Market Basket Data. ACM SIGMOD Conference on Management of Data, pp. 265-276.

Bruno, D., & Richmond, H. (2003). The truth about taxonomies. Information Management Journal, 37(2): 44-51.

Chen, H., Houston, A.L., Sewell, R.R., & Schatz, B.R. (1998). Internet browsing and searching: User evaluation of category map and concept space techniques. Journal of the American Society for Information Science, 49(7): 582-603.

Chen, K. J., & Liu, S. H. (1992). Word identification for Mandarin Chinese sentences. Proceedings of the 15th International Conference on Computational Linguistics. Nantes, France.

Chien, L. F. (1997). PAT-Tree-Based Keyword Extraction for Chinese Information Retrieval. Proceedings of the 20 International ACM SIGIR Conference on Research and Development in Information Retrieval. Philadelphia, PA, USA, July 27-31, 50-58.

CyberLink StreamAuthor. Available at http://www.cyberlink.com/multi/products/main_7_ENU.html.

Derrickoswald and Somik. HTML Parser. Available at http://htmlparser.sourceforge.net/

Dittenbach, M., Merkl, D., & Rauber, A. (2000). The Growing Hierarchical Self-Organizing Map. Proceedings of the Int'l Joint Conference on Neural Networks (IJCNN'2000), vol. VI: 15-19. Como, Italy: IEEE Computer Society Press.

Dringus, L. P., & Ellis, T. (2005). Using data mining as a strategy for assessing asynchronous discussion forums. Computer & Education, 45, 141-160.

Ed Ort, and Bhakti Mehta (2003) Java Architecture for XML Binding (JAXB). Available at http://java.sun.com/developer/technicalArticles/WebServices/jaxb.

Fisher, K.M., Wandersee, J. H., & Moody, D. (2000). Mapping Biology Knowledge. Boston: Kluwer Academic.

Fox, E., Hix, D. L., Nowell, T., Brueni, D. J., & Rao, D. (1993). Users, User Interfaces, and Objects: Envision, A Digital Library. Journal of the American Society for Information Science, 44(8), 480-491.

Govindasamy, T. (2001). Successful implementation of e-learning; pedagogical considerations. The Internet and Higher Education, 4(3-4), 287-299.

Han J., J. Pei, and Y. Yin (2000) Mining frequent patterns without candidate generation. Proc. of the ACM SIGMOD Conference on Management of data.

J.S Park, C.Ming-Syan, and S.Yu.Philip (1995) An Effective Hash Based Algorithm for Mining Association Rules. Proc. of ACM SGMOD, pp. 175-185.

Kiang, M. Y., & Kumar, A. (2001). An evaluation of self-organizing map networks a robust alternative to factor analysis in data mining applications. Information Systems Research, 12(2), 177-194.

Klemke, R. (2000). Context framework – an open approach to enhance organizational memory systems with context modeling techniques. Proceedings of the Third International Conference on Practical Aspects of Knowledge Management (PAKM2000), Basel, Switzerland, 30-31 October.

Kohonen, T. (2001). Self-organizing Maps. (3rd ed.). New York: Springer-Verlag.

Li, Sheng-Tun and Lin, Chu-Hung (2005) On the distributed management of SCORM-compliant course contents. Advanced Information Networking and Applications, AINA 2005. 19th International Conference on Volume 1, Issue, 28-30 March, pp. 221 - 226 vol.1.

Lin, X., Soergel, D., & Marchionini, G. (1991). A self-organizing semantic map for information retrieval. Proceeding of 14th the ACM SIGIR Conference Research and Development in Information Retrieval. Chicago, Illinois, USA, October 13-16, 262-269.

Liu, D.-R., & Hsu, C. (2004). Project-based knowledge maps: combining project mining and XML-enabled topic maps. Internet Research,14(3), 254-266.

Liu, F.J. & Shih, B.J., E-learning activity-based material recommendation system, Interactive Technology and Smart Education, VOl. 4, Issue: 4, P.200-P.207, 2007, 11.

Malhotra, Y. (1998). Deciphering the knowledge management hype. Journal for Quality & Participation, 21(4), 58-60.

Marx, R. W., Blumenfeld, P. C., Krajcik, J. S., & Soloway, E. (1998). New technologies for teacher professional development. Teaching and Teacher Education, 14, 33-52.

Masiello, I., Ramberg, R., & Lonka, K. (2005) Attitudes to the application of a Web-based learning system in a microbiology course. Computer & Education, 45, pp. 171-185.

Mladenic, D., & Grobelnik, M. (2003) Feature selection on hierarchy of web documents. Decision Support Systems, 35, pp. 45-87.

Nonaka, I., & Takeuchi, H. (1995) The Knowledge-Creating Company – How Japanese Companies Create the Dynamics of Innovation. NY: Oxford University Press.

Noy, N. F., & McGuinness, D. L. (2001). Ontology development 101: A guide to creating your first ontology. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, March.

Putnam, R. T., & Borko, H. (2000). What do new views of knowledge and thinking have to say about research on teacher learning? Educational Researcher, 29, 4-15.

Rasmussen. E. (1992). Clustering Algorithms. In W. B. Frakes and R. Baeza-Yates (eds.) Information Retrieval Data Structures and Algorithms. NJ: Prentice Hall.

Rath, H. H., & Pepper, S. (1999). Topic maps: introduction and allegro. Proceedings of the Markup Technologies 99, Philadelphia.

Rauber, A., & Merkl, D. (1999). Automatic Labeling of Self-Organizing Maps: Making a Treasure-Map Reveal its Secrets. Proceedings of the 3. Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'99). Beijing, China, April 26-28.

Roussinov, D., & Zhao, J. L., (2003). Automatic discovery of similarity relationships through Web mining. Decision Support Systems, 35, 149-166.

Salton, G. (1989). Automatic text processing. MA: Addison-Wesley.

SCORM 2004 3rd Edition. Available at http://www.adlnet.gov/.

Shih, B. -J., Shih, J. -L., & Chen, R. -L. (2007) Organizing learning materials through hierarchical topic maps: An illustration through Chinese herb medication. Journal of Computer Assisted Learning, Dec 07 issue, Volume 23, Issue 6, pp. 477-490.

Smith, K. A., & Ng, A. (2003). Web page clustering using a self-organizing map of user navigation patters. Decision Support Systems, 35, 245-256.

Veerasamy, A., & Belkin N. J. (1996). Evaluation of a Tool for Visualization of Information Retrieval Results. Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval, (SIGIR), 85-92.

Wong, K. F., & Li, W. J. (1998). Intelligent Chinese Information Retrieval - Why Is It So Difficult? Proceedings of the First Asia Digital Library Workshop. 47-56.

Yang, C. C., Chen, H., & Hong, K. (April, 2003). Visualization of Large Category Map for Internet Browsing. Decision Support Systems, Special Issue on Web Retrieval and Mining, 35(1), 89-102.

Yoon, B.-U.,Yoon, C. -B., & Park, Y. -T. (2002). On the development and application of a self-organizing feature map-based patent map. R&D management, 32(4), 291-300.

Zamir, O., & Etzioni, O. (May, 1999) Grouper: A Dynamic Clustering Interface to Web Search Results. Computer Networks, 31(11-16), 1361-1374.

E-learning Experiences and Future

Edited by Safeeullah Soomro

This book is consisting of 24 chapters which are focusing on the basic and applied research regarding e-learning systems. Authors made efforts to provide theoretical as well as practical approaches to solve open problems through their elite research work. This book increases knowledge in the following topics such as e-learning, e-Government, Data mining in e-learning based systems, LMS systems, security in e-learning based systems, surveys regarding teachers to use e-learning systems, analysis of intelligent agents using e-learning, assessment methods for e-learning and barriers to use of effective e-learning systems in education. Basically this book is an open platform for creative discussion for future e-learning based systems which are essential to understand for the students, researchers, academic personals and industry related people to enhance their capabilities to capture new ideas and provides valuable solution to an international community.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

INTECH
open science | open minds