早稲田大学審査学位論文
博士（人間科学）

ソーシャルネットワークモデルに基づいた統合アプローチによるラーニングオブジェクトの再利用

Facilitating Reuse of Learning Objects: An Integrated Approach Based on Social Network Mining and Analysis

２０１２年１月

早稲田大学大学院　人間科学研究科

厳昱文
YEN , NEIL YUWEN

ABSTRACT

Sharing resources and information on the Internet is becoming a popular and essential activity for education, and identifiesan emerging approach to achieve collective intelligence.Our repository (i.e., the MINE Registry), a branch of federated database systems under CORDRA architecture, has been developed for managing and sharing resources (or learning objects), approximate 23,000 at this stage, in the past five years. The learning objects are created and reused by users, especially instructors, for specific purposes. To enhance the reusability, an approach entitled Reusability Tree was proposed to track the alternation history, which is similar to the concept of version control in Software Engineering,of learning objects. Following the Reusability Tree, this thesis goes further to investigate the inner attributes of learning object and its correlations with others in a federated repository considering the main focus of human-centric support.

The features, as well as a solution to CORDRA enhancement,are summarized as follow.First, the LONET (Learning Object Network), a social-driven structure assystematic extension of the Reusability Tree, is proposed to clarify the vague reuse scenario of learning objects.The structure identifiesthe collaborative intelligencethrough past interactive usage experiences, and can be graphed in terms of implicit and explicit relations among learning objects. The metrics based on social network analysis are developed to quantify the relations (or interdependency),such as prerequisites, inheritance, reference and peer, and achieve further reuse.

Based on the LONET, the practical applications that facilitate the reuse processof learning object are then proposed from the perspective of web search. The

reference information, including citations and user feedbacks, are applied to develop a weighting/ranking algorithm with the focus of time-series information, and highlight the significance.The search guidance algorithm is proposed with the aim at leading users to appropriate direction(s) by providing progressive suggestions corresponding to the query. And, as a practical outcome,an interactive search algorithm is proposed to mine the experience in the LONET. It generates path(s), based on past usage experiences, by computing possible interactive input, such as search criteria and feedback from instructors, and generates the tentative template for lecture generation.

To demonstrate the feasibility, an empirical study is conducted by two system performance evaluationsand a usability testing experiment. The evaluation is performed based on TREC testing with a general Precision-Recall test for retrieval accuracy and a nIAP (non-Interpolated Average Precision) test for ranking accuracy. The usability experiment is conducted with around 50 users (e.g., professors, lecturers and teacher assistants) in several universities, andreaches an expected result.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

CHAPTER I. INTRODUCTION

A comprehensive introduction is addressed to get users involved in this study. Following the background, the motivation and major purposes are discussed. Thekey features of this thesis are then mentioned, and the organization of the thesis is left in the last section.

1.1 Background

Development of information systems (and thus knowledge system) isbriefly categorized into three portions: User Interface, Process Logic, and Data Storage. From the perspective of E-Learning, involved systems include Authoring Tool, LMS (Learning Management System) and Repository. Broadly speaking, websites for online learning with functionalities of administration, course management and assessment could be seen as a LMS. Authoring tool is used for the creation and management of courseware and metadata. And, an open database that provides data storage, searching, delivery, and exchange functions is called a repository.

In order to enhance the systematic reuse, the ADL (Advanced Distributed Learning Initiative) initiated a distance learning de facto standard named SCORM (Sharable Content Object Reference Model). SCORM includes the CAM (Content Aggregation Model) to enhance reusability of courseware (as known as learning object), RTE (Run-Time Environment) to support learning activities, and SN (Sequence and Navigation) to provide a series of sequence rules for adaptive learning purposes. SCORM-compliant learning objects can easily achieve sharability and reusability through the use of LOM (Learning Object Metadata) [IEEE Draft Standard for Learning Object Metadata 2002] described in CAM.

In distance learning (or E-Learning), a repository identifies not only a distributed storage for *Learning Objects*[Hani Sarip and Yahya 2008][Shih et al. 2006], any digital resources for educational purpose, but also a common platform for it to be shared and reused.Although the issues of common repository for online learning were addressed [Hasegawa et al. 2002][Sánchez Alonso 2009], representation of learning objects is another key issue that affects the design of repository architecture [Ternier et al. 2009].

Thanks to Web 2.0, a growing number of Internet resources can be selected, assigning corresponding metadata [Van Assche et al. 2003], for the use in specific learning activities. However, the situation also brings an open issue on how a huge number of learning objectscan be managed in an efficient manner [Koppi et al. 2005]. As a solution, CORDRA (Content Object Repository Discovery and Registration/Resolution Architecture) was proposed to build an interoperable federation for storing and sharing SCORM (Sharable Content Object Reference Model)-compliant learning objects between heterogeneous underlying repository infrastructures [Vassiliadis et al. 2003].

From the perspective of instructors, repositories can be regarded as knowledge bases in which useful learning objectsare obtained. Users (i.e., instructors) can reassemble obtained learning objects in specific sequences for different objectives; and meanwhile, the original organization of learning objects is changed. On the other hand, the creation of a new sequence indicates a series of new connections.

To ensure reusability, the metadata (i.e. IEEE LOM) has been utilized to facilitatelearning objectdiscovery [Kastner and Furtmüller 2007]. In an earlier work [Lin et al. 2009], a concept named Reusability Tree was proposed to track the organization alteration of learning objects in a federated repository (i.e., the MINE Registry). This work not only recorded the changes but proposed quantitative metrics (i.e., similarity and diversity functions) to represent the difference degree of inner organization (i.e., sequence) between the newly created or revisedlearning object and the original one. As to the learning object discovery, several approaches, such as ontological [Gasevic et al. 2007] and semantic [Tsai et al. 2006] representations, were adopted for facilitating the search process. Some metrics, such as ranking [Ochoa and Duval 2008] and inference rules [Khierbek et al. 2008] were also applied to develop customized search services. However, as a summary, three major challenging issues that the researchers may face:

- **Vague Significance of Learning Object**

  The search has become a common service implemented in any web systems (i.e., the repository) that allow users to interact with the database. However, the common search service concentrates on accuracy (e.g., precision-recall) and efficiency (e.g., time cost) of the retrieval results and, in some ways, omits the significance that a data should have.

- **Uncertain Reuse Approach**

  Though the search service provide a solution to learning object reuse, the approach to achieve above mention is still not clearly addressed. The common situation is that users have difficulty to obtain the expected information (or learning object in this study), and often leads to "cold start".The approach to learning objects is not clearly defined.

- **Time-Consuming Process of Lecture Generation**

  One query, in general, is expected to reach one result. That means a huge number of queries have to be sent once a lecture for specific purpose needs to be generated. Although built-in search service can serve the initial situation (but the above mentions still need to be concerned), the cost of time in selecting useful learning objects is still required.

To provide solutions to the challenges, integration with interdisciplinary approaches is concerned. The researches have shown that Social Network Analysis can be a promising approach to clarify the complex interactions among participants, human users and their associated information and knowledge as well, by quantifying potential interactive processes to an interconnected relation graph or network [Tolsdorf 1976]. As a result, it not only makes easier to enunciate interpersonal information, but, furthermore, enables researchers to discover those indirect relationships by mining information associated to common intersections [Palla et al. 2007]. The concept can also be applied to every domain in which the interactions may occur (e.g., links between web pages, citations between research papers), by considering customized factors and metrics [Kumar et al. 2002; Raghavan 2002]. Hence, from a perspective of knowledge sharing, it raises the emerging issues about patterning collective intelligence [O'Reilly 2007].

Considering the above mention, this study goes further to investigate the inner attributes and correlations of learning objects from interdisciplinary perspectives of social network, information retrieval and web information mining to achieve efficient reuse of learning objects systematically. The motivation and the purpose are addressed in the following section.

1.2Motivation and Purposes

Although noted international standards, such as SCORM and CORDRA, have provided preliminary solutions for searching and reusing learning objects, a few fundamental issues are still left unsolved. For instance, learning objects are reused by users (e.g., leaner and instructor) after being registered in a repository. The learning object is becoming connected with specific purposes through simultaneous use.To facilitate the discovery of learning object, mechanisms, such as adaptive search and recommendation, are applied to the search service implemented in a repository. However, as time goes on, a vast amount of learning objects may be created in the scenario and cause the difficulty of learning object retrieval in some situations.

In this study, the issues on efficient reuse of learning objects are concentrated.Three major purposes are involved. First, the relations, especially those frequently used but implicit,will be clarified to construct a knowledge network, with focus on the use of collective experiences, for further systematic reference. Second, the significance of learning objects within the constructed network is going to be investigated concerning the time-series information. And third, the automated mechanisms that may facilitate the search process are sought in the field of information retrieval and applied for further supports to the general users.

1.3Thesis Features

The interdisciplinary perspectives, including information retrieval, social network analysis and data mining, are considered to achieve efficient and systematic reuse of learning objects within federated repositories. The thesis is featured by:

1. **SocialModel for Learning Object Representation**

   Amodel based on social network is developed to clarify the correlations between learning objects in accordance with the interactions which are adopted by users (i.e., instructor) in the past five years. We consider this as an approach to achieve efficient reuse of learning objects based on collective experience.

2. **Time-Series Weighting/Ranking Approach**

   We develop a weighting/ranking algorithm to feature the significance of learning objects based on corresponding time-series information and the relevancy between target learning objects and the query instead of a common approach (e.g., attributes matching).

3. **Progressive Suggestion to Facilitate Search Process**

   We implement a progressive search to facilitate the difficulty while searching for learning objects. This algorithm aims at leading users to appropriate direction(s) by providing progressive suggestions corresponding to the query.

4. **Interactive Search for Lecture Template Generation**

   We propose an automated algorithm to search for the experiences in the learning object network and to generate the templateoflectures (i.e., organization of learning objects in specific sequence) based on a specific learning object.

1.4 Thesis Organization

This thesis is organized in seven chapters including:

- **Chapter I Introduction**

  The challenging issues on learning object retrieval and reuse are introduced in the beginning of the chapter.We concentrate on the objective nature of the mentioned issues and assume that the solutions are imminently requiredwith deep consideration of social factor including its technical aspects.

- **Chapter II Literature Review**

  This chapter discusses the related issues regarding the systematic reuse of learning objects with the main focus human-centric support. The survey on web information retrieval, social network analysis and time-series mining technologies are summarized as the bases that develop this thesis.

- **Chapter III Learning Object Network**

  The learning object network that is inspired by the concept of social network is proposed in attempt to clarify the reuse. The five-year usage experiences are applied to define the rules for investigating the correlations between learning objects. In addition, the metrics based on social network analysis are incorporated to quantify the relations and achieve the acquisition of collective experiences.

- **Chapter IV Applied Search in LONET**

  This chapter addresses the potential search applications, including the weighting/ranking algorithm, the search guidance and the lecture template generation, that facilitate the discovery and systematic reuse of learning objects within the federated repository.

- **Chapter V Experiments and Evaluation**

  The empirical study is conducted to demonstrate the feasibility of the proposed works. Two system evaluations and a usability experiment on around 50 users are involved.

- **Chapter VI Conclusion**

  This chapter concludes the thesis with the brief summary of contributions, and presents the expected challenging issues and the future working directions.

- **Appendix Implementation**

  The appendix demonstrates the concrete implementation of the proposed works mentioned in the Chapter III and Chapter IV. The system automated functions are walked through via a simple operation process of learning object searching.

CHAPTER II. LITERATURE REVIEW

Beginning from the fundamental issue, reusability of learning object, three related issues regarding conceptual and technical aspects are discussed in this chapter. First,the strategies that assist the analysis of social network are introduced. The methods widely adopted to search the social network are addressed. The models for time-series data mining that prompt search process and experience, and the general framework to enhance the service interoperability are specified in the last. A brief summary identifying the connection between the study and above is then given.

2.1 Reuse Issues of Learning Object

Three portions are involved in this section including: IEEE LOM (Learning Object Metadata) and CORDRA (Content Object Repository Discovery and Registration/Resolution Architecture),the Reusability Tree, and methods for learning object retrieval and filtering.

2.1.1 IEEE LOM & CORDRA

A five-layered framework was proposed by IEEE LTSC (Learning Technology Standard Committee) to make comprehensive descriptions oflearning objects. Following the framework,the LOM(IEEE 1484.12.1-2002 LOM v1.0)defined an exchangeable format(i.e. metadata)that achieves efficient storage, sharing, and reuse of learning objects [Muzlo et al. 2001]. As to technical perspective, CORDRA specified a standardized development process to design and implement interoperable federations of learning object repositories.Though the process resolved the issues on conflict of name space while indexing registered learning objects in federated repositories, an open issue on learning objects management, especially those diverse but share common attributes, was raised.As a preliminary solution, the Reusability Tree [Lin et al. 2009] was proposedto clarify the relations among reuse history of learning objects.

2.1.2 The Reusability Tree

The Reusability Tree is conceptually similar to a version-derivation tree (or version control in Software Engineering) which is applied to the Content Aggregation in LOM. It consists of nodes and links, where nodes stand for associated learning objects and links represent the sequence between learning objects. A child

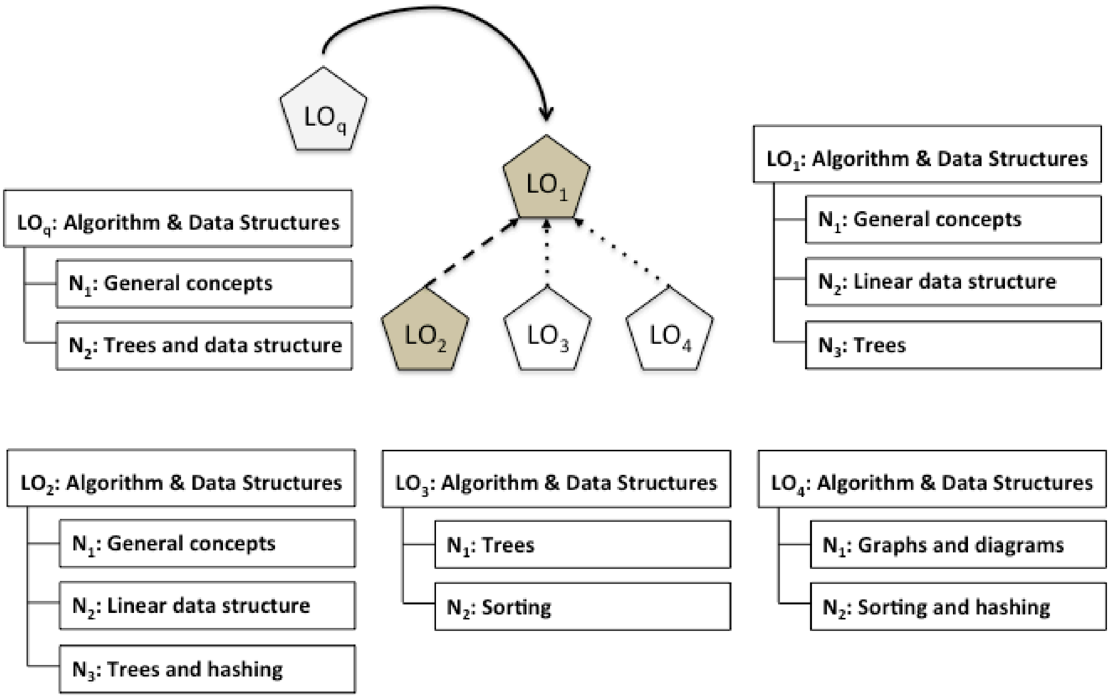


Figure 2-1 An Example of Reusability Tree

object thus sharessome attributes inherited from parent object andadds additional properties. When a learning object is being reused, several types of changes may be made*. And the changes are captured in the Reusability Tree. The metrics, such as similarity and diversity, defined in Reusability Tree can represent the relative information (i.e., degree of changes) in organization of learning objects. Taking Figure 2-1 as an example, four learning objects are in the scenario. LO_1 represents the original learning object with three nodes. Three learning objects, LO_2, LO_3 and LO_4, are created by modifying parts (i.e., nodes) of LO_1, and are considered as the nodes in the derivation tree from LO_1. In the example, it is not difficult to observe that LO_4 has a higher similarity with LO_1 (the original learning object), as compared with others. Once a query object (i.e., LO_q)is given, the best-matched object (LO_1) along with the derivative object(s) that contains higher similarity and less diversity are returned for further reuse.

*The changes are recorded by HardSCORM authoring tool while learning objects are accessed.

2.1.3 Methods for Learning Object Retrieval and Filtering

To facilitate learning object searching, usage of techniques in Information Retrieval (IR) is required. IR is usually applied as a filter to achieve retrieval and recommendation in specific domains. A method named Contextual Attention Metadata [Ochoa and Duval 2008] was proposed to rank learning objectbased onitscontainedattributes, such as course context and/or relations between learners and authors, during the lifecycle of learning object. To be specific, IR can be categorized in three major filtering models including

(1) User-based Filtering Model

This method classifies users based on the similarity discriminate factors including users' interests or hobbies, to generate the user model [Herlocker et al. 2002]. The system then follows the model to provide results (i.e. learning objects) to specific users adaptively;

(2) Item-based Filtering Model

The user-based filtering method exposes the problem of excessive computation time due to the increasing number of users. According to [Sarwar et al. 2001], an item similar to another one in which a user is interestedmay draw the attention of same user. Though it operates stable recommendation results, the difference among users is left unconsidered [Balabanovic and Shoham 1997];

(3) Model-based Filtering Model

This modelconcentrates on providing solutions to the limitations (e.g. information scarcity and scalability) that above two methods remained which may cause the provision of inaccurate results.In brief, this method adopts the advantage from user profile, same as the user models created by user-based filtering, and trains the following usage experiences to make prediction [Tolsdorf 1976][Umyarov and Tuzhilin 2008].

2.2 Social Network Analysis

SNA (Social Network Analysis) emphasizes on mining the exchange of resources among participants ranging from individual to organizations [Haythornthwalte 1996]. Briefly speaking, the architecture of Social Networks can be divided into two types, Homogenous and Heterogeneous, in accordance with their internal attributes. Graph Theory [Wellman and Berkowitz 1988] is often used to make a clear description of connections between participants within a specific network.

The original resources can be traced back to the studies on human relationships [Wegener 1991] around twenty years ago. And nowadays, the composition of networks, not limited to analysis of human beings, can be constructed by integrating customized factors into original mining methodologies [Bates and Peacock 1989]. To obtain expected, often implicit, information from such networks, it is necessary to take some factors defined in Graph Theory into consideration by using the concept of data mining for implicit information and/or patterns discovery [Jensen and Neville 2002].

The methods for determining the significance of nodes and links are PageRank [Langville and Meyer 2006], HITS (Hyperlink-Induced Topic Search) [Kleinberg 1999], and EigenRumor [Fujimura 2005] whose return values can be regarded as weights of measuring the degree of centrality [Freeman 1979]. The semantic methods [Weng et al. 2009; Wolf et al. 2009; Lin and Kao 2010] are also widely used for describing interdependency within the social structure. The semantics of a node is modeled using its surrounding labeled network structure, representing by the sequences of labels (i.e., paths) together with some statistical dependency measures associated with them. A scalable mining method [Albert and Barabashi 1999] is also applied to assist researchers in discovering and generating the possible connections.

As a practical contribution, the work in [Yang et al. 2010] addressed the importance of prioritization to specific interactivity such as exchanging email messages. The method to SNA, such as degree of centrality, and training methodology, SVM (Support Vector Machine), were adopted to cluster delivered email messages to specific groups, and to determine the order of email receipt in accordance with assigned priority weight. This concept is also referred by the latest introduced service “Priority Inbox” provided by Google Gmail.

2.3Searching in Social Network

To make efficient use of the analysis results, in this section, we go further to discuss the methods that may be applied to discover those indirect relationships between participants within social network. Research [Scott 1991; Wasserman and Faust 1994] indicates that Social Network complies with Small World theory and has feature of heavy tail distribution [Stanley 1967]. We put it differently to explain this phenomenon. That is, it is possible to identify the follow-up (i.e., path) when obtaining an expected node. In other words, how to navigate such structure has brought an open issue. Generally speaking, the methods to search social networks can be categorized into (1) Path-Oriented, and (2) Efficiency-Oriented. The first one emphasizes on discovering shortest and complete paths to connect different nodes. The second one pays more attention on improving the overall search efficiency to reduce the possible time cost. For example, Kleinberg [Kleinberg 2000; 2001] proved that a simple greedy strategy could achieve shortest path length, in $O((\ln N)^2)$, in a scalable range (i.e., hierarchically nested groups) by converting the structure into specific dimensions of space vector. To make improvement, several methods [Moody et al. 2005] were proposed to search the non-structured social networks. NeuroGrid [Joseph 2002] utilized Routing Tables generated by past query histories to achieve intelligent route selection. The same method was applied to a study [Rowstron and Druschel 2001] which utilized the tags to replace the routing tables in order to improve the completeness. As another instance, SSON (Semantic Social Overlay Networks) [Lin and Chalupsky 2008; Loser et al. 2007] made use of response frequencies to obtain nodes that might respond to the requests, and utilized the LRU

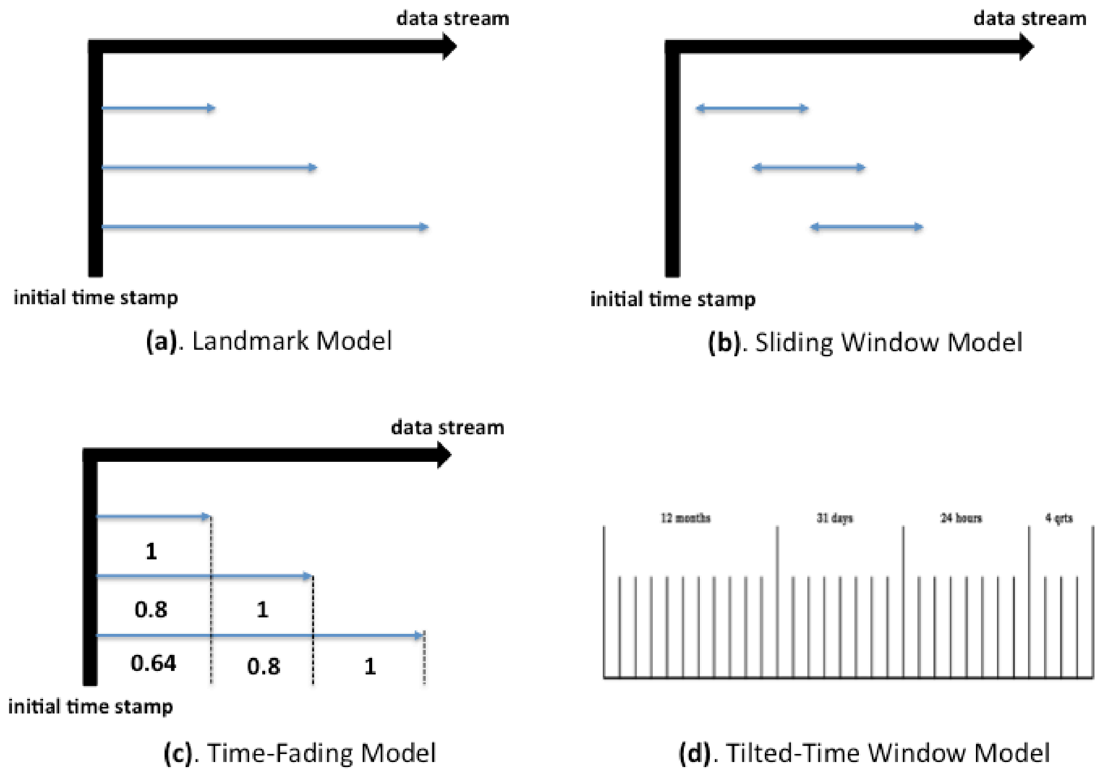
(Least Recently Used) to remove the redundant connections. As an extension, ESLP (Efficient Social-Like Peer-to-Peer Networks) [Lin et al. 2007] utilized query thresholds to determine the possibility of correct response to search results, and rank them for deciding how many query messages should be sent from a specific node. In addition, researchers [Ghanea-Hercock et al. 2006; Koo et al. 2006] posed the issue on neighborhood selection by assigning different weights to nodes. The Hebbian Rule was applied in the weighted network to revise the Swarm Intelligence algorithm for optimal neighborhood selection within a social-like peer-to-peer network [Liu et al. 2009].

2.4 Mining Time-Series Data

The time-series information has been widely applied to highlight the importance of data within specific periods. Four major models including Landmark Model, Sliding Window Model, Time-Fading Model and Tilted-Time Window Model (shown in Figure 2-2)which focus on different perspectives of mining usage are discussed.

Landmark [Perng et al. 2000] identifies the time stamp that the system is initialized. The process histories are recorded from the beginning of system time (i.e., the time stamp) until suspensions (i.e., the end time stamp of specific procedure). This model achieves comprehensive records regarding each procedure and is applicable to make further comparison and analysis. However, extreme workload of the system is revealed the shortcoming while overtime processing of specific schedules, and,in addition, not all of the recorded data within the timescale are useful. The garbage collection and processing are required to be concerned while Landmark Model is adopted.

The Sliding Window Model [Lin et al. 2003] improves the disadvantages in Landmark Model.Itconcentrates on data stream analysis within specific timescaleswhich is represented by a fixed width of window. The data will be loaded and processed in a specific timescale ahead of the current system time. After that, data elements are implicitly deleted from the specific sliding window, when it moves out of the window scope. However, the use of such a model focuses on the basis of timescale, and have to be adjusted based on specific constraints or different conditions and circumstances.



(a). Landmark Model **(b)**. Sliding Window Model

(c). Time-Fading Model **(d)**. Tilted-Time Window Model

Figure 2-2Illustration of Time-series Mining Models

The data stream is considered as the same in each timescale in previous models.In order to clarify the importance of data in each timescale, the Time-Fading Model [Chang and Lee 2003] which claimed that the distance of time (e.g., latest to oldest) as a key feature of measuring time-series data was proposed. It separates the processing time of each system schedule into isometric blocks, and assigns weight to each blockin decreasing orderprogressively(from the latest one to the oldest one). It improves the relationship between data and timescale, especially to those timeliness data. Taking Figure 2-2 (c) as an instance, the data in the later block will have higher weights than previous ones.

Using the presented models for mining time series data, the quantity of data has to be concernedbecause of the memory size. In Landmark Model, it may take

$$60(minutes) \cdot 24(hours) \cdot 31(days) = \mathbf{44640}(units)$$

to record the data in one month with the smallest measurement unit at one minute. In order to overcome this storage problem, the Tilted-Time Window Model [Chen et al. 2002] was proposed. The block obtained throughTime-Fading Model is further divided into different sections from the nearest one to the farthest one. The nearer sections will be given in more details as shown in Figure 2-2 (d).With the same example, the total memory costs will be:

$$60(minutes) + 24(hours) + 31(days) = \mathbf{115}(units)$$

In this study, we pay emphasis on the focus-to-date information, and to estimate the importance degree by using the mining methodologies, especially the Time-Fading Model and the Tilted-Time Window Model. The records of learning objectsin the past five years are considered in a macro perspective through the integration of these methodologies.

2.5 The Learning Tools Interoperability

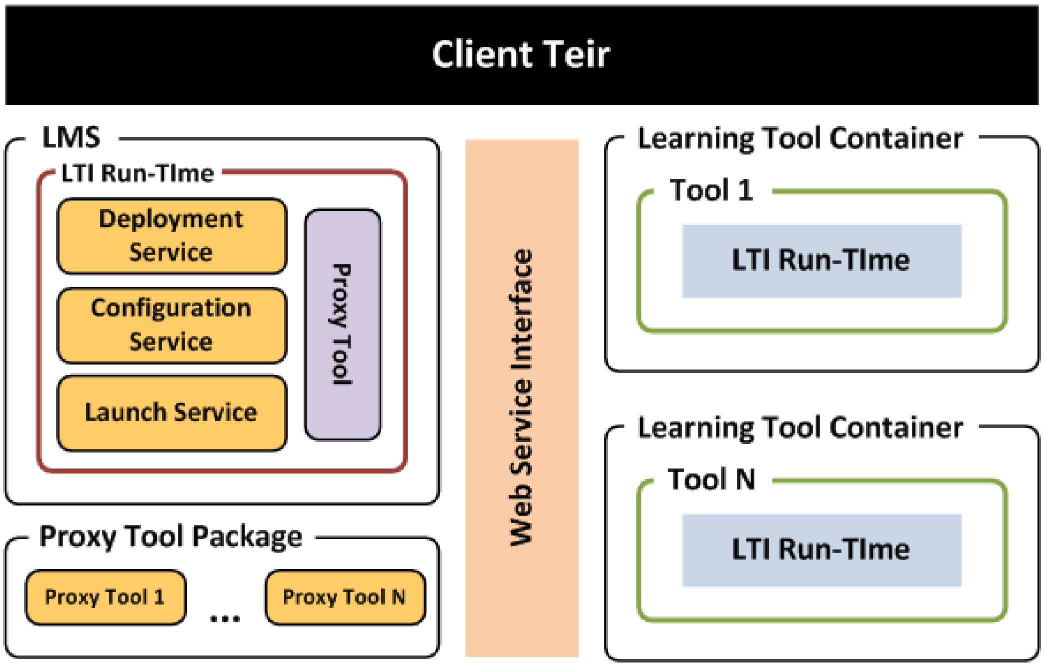


Figure 2-3 General framework of Learning Tools Interoperability

A web service framework named "Learning Tools Interoperability" which was initialized by an international learning society, IMS Global Learning Consortium, is to provide a solution for services to be reused. It aims at enhancing the functionality of learning systems through an external integration of web services (or any services in an existing system) but not adding the services into the original systems instead. The service is considered as a component and is applicable to be reorganized based on specific purposes.The adoption of Learning Tools Interoperability facilitates the sharing and exchange of resources, especially those are not designated to be in common definition, among the Internet, and achieve the purpose of sharability.

The general framework of Learning Tools Interoperability is shown in Figure 2-4. Three major portions are involved. First, the *Learning Tool Container* identifies the web service that is provided by any third-party developer. Second, the *Proxy Tool Package* reveals the method to communicate between main system and services and/or service and services. And the last, the LTI Run-Time which contains the core setting of the main system.The essential parts include Proxy Tool and the Run-Time which are specified as follows.

Proxy Tool

Proxy Tool refers to the run-time environment of Learning Tools Interoperability and the entrance of business logic of external services. The primary goal is to reduce the extra efforts (e.g., time, codes) which might be required to get interoperated. Proxy Tool will link up with the deployment descriptor tightly and it is also a bridge to communicate with main system (e.g., learning management system in this example) and third-party services. The transmission of data is through deployment package in the run-time environment of Proxy Tool. This package is considered as a kind of data folder. It includes a manifest file and deployment descriptor. The deployment descriptor is an XML-based file and contains one or more deployment profiles and external settings.

Learning Tools Interoperability Run-Time

The Learning Tools Interoperability Run-Timerefers to a set of services. The implemented services are easily to be deployed, modified and interacted with systems in different platforms. It serves as a common web service which sends out the requests, receives the responds and proceed the follow-ups. The details are discussed below.

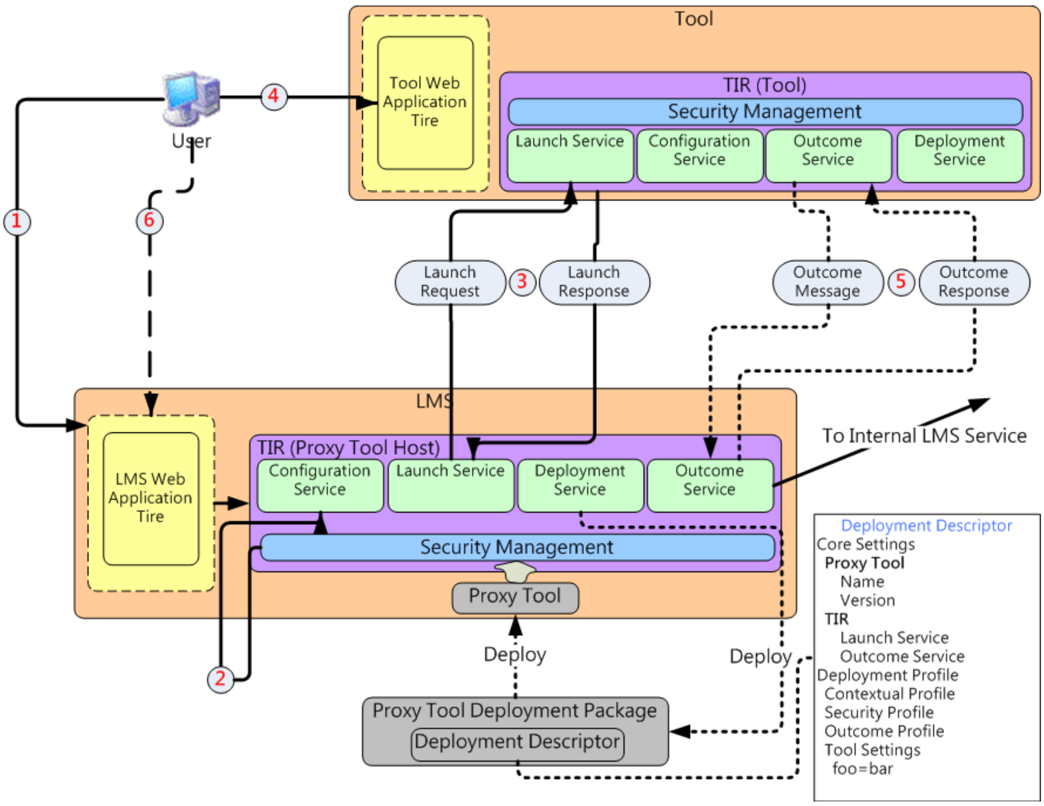


Figure 2-5 Communication Process of Learning Tools Interoperability

- **Deployment Service**

  It deploys applications in the main system and takes them as one of the learning objects through deployment package of Proxy Tool.

- **Launch Service**

  It generates the protocol and determines which security mechanism should be applied in main system side. In the service side, this service is responsible for retrieving the launch message or security message that is sent or utilized by the main system and responds the protocol.

- **Configuration Service**

  This service manages the settings of Proxy Tool. It makes the deployment service and launch service could operate normally.

- **Outcome Service**

  The outcome profile which is generated by applicationis sent to the run-time of main system and proceeds it through Proxy Tool. Then, the run-time of main system will respond it to the run-time of external services.

- **Security Management**

  A flexible authentication mechanism is developed to provide services when they generate a security profile and send the authentication information through Proxy Tool. In order to provide multi-forms for personal authorization, authentication modules or other services for main system, the deployment descriptors of Proxy Tool will have their own security profile. A message sent by the core protocol of Proxy Tool will extend to a specific security profile through SOAP.

- **Session Management**

  When users would like to use a new service, main system will provide a URL (Uniform Resource Locator) for user to click and then start a new browser window. The essential information is transmitted through session mechanism. The run-time of main systemthen activates session management for generated sessions. It is similar tothe main system redirects the user to the new URL of services, but all the operations are still performed in the main system.

2.6Summary

Research works have pointed out the trends of Social Network development, which mainly emphasize on how implicit information, such as neighborhood selection, associated to specific nodes is obtained within a specific network environment. This concept has also been applied in the scope of distance learning for learners to obtain appropriate models to follow. As to the perspective of instructors, the difficultyin lecture generationhas brought an open issue that discourages the popularity and reusability. In this study, the mentioned works are applied to facilitate learning object management, retrieval, and reuse. A socialized network structure, LONET (Learning Object Network), is proposed to clarify implicit usage experiences recorded in Reusability Tree. Then, four automated algorithms, such as weighting and ranking, search guidance, search path and lecture template, are developed concerning the time-series issues to assist instructors in searching for useful learning objects. In addition, the Learning Tools Interoperability is applied as an underlying framework to enhance the interoperability of implemented service, and prompts the integration and re-development of the proposed works.

CHAPTER III. LEARNING OBJECT NETWORK

Considering efficient reuse of learning objects, the recorded interactions, especially those implicit relations, in Reusability Tree are specified. The alteration histories and usage experiences on lecture generation in past years are utilized to develop the social structure, named Learning Object Network (LONET), in MINE Registry. In this chapter, the definition of LONET is introduced. The LONET modeling and the metrics for mining the LONET are then discussed. A concrete instance regarding the use of mentioned metrics is given in the last.

3.1 Definition

Learning Object Network (LONET) is conceptually similar to Social Network, but concentrates on the interactions of learning object. It consists of nodes and links, where nodes represent the participants (or learning objects) and the links stand for the correlations associated to two connected nodes. It is used to clarify the relationships between learning objects, especially those used for assembling lectures for specific objectives, and is constructed by the possible alterations recorded by Reusability Tree, while generating lectures. The main purpose of LONET is to meet the following needs.

- **To represent relationships effectively**

  The matter for composing learning objects to lectures varies in accordance with different teaching objectives. The relationships among learning objects are thought to be intricate. Once the organization of lecture changes, the correlations between related learning objects are changed (e.g., remove old or add new learning objects). The repeated alteration processes are difficult to be traced if an isolated hierarchical structure is the only method. Thus, how to effectively represent the relations among huge number of learning objects is the first priority.

- **To facilitate lecture generation**

  Search engine is regarded as a tool to assist instructors in retrieving resources from the repository. However, lots of queries are required to obtain expected learning object(s). Fortunately, the past experiences that instructors utilized to generate lectures for specific objectives are recorded in Reusability Tree. Therefore, it is applicable to provide an interactive search method to mine possible patterns (e.g., templates for lecture generation) from such network. The pattern is composed of consecutive learning objects, and available to be revised in accordance with customized needs from instructors. This process offers a solution to facilitate the difficulty in preparing lecture structure and corresponding supplementary materials.

3.2 Modeling the Learning Object Network

The shift of structure from Reusability Tree (past experiences) to LONET (social structure) is introduced in the beginning. A general expression that formulizes the LONET is then given.

3.2.1 From Reusability Tree to LONET

The past usage experiences are essential to construct the social structure. A simple example collected from MINE Registry is shown in Figure (a). Once a user sends a query, the system will return results in the form of tree-based structure that represents the relations, similarity and diversity, between learning objects from a macro perspective. Hundreds of such results will be shown to users for further usage, but it also takes time for filtering out irrelevant ones. Considering different compositions of Activity Tree in learning objects, they can be taken as ways for generating specific objectives. Thus, to make an improvement, we take each node Activity Tree as an independent learning object by giving corresponding metadata, and to use a directed graph to describe the correlations as shown in Figure (b). In this situation, learning objects which is not connected originally can be connected in accordance with the same objectives. This practice may strengthen the degree of binding between learning objects, and may achieve a continuous process of learning object discovery that facilitate specific purposes. In Figure 3-1, the elements for conversion from Reusability Tree to LONET are shown: the box identifies a subjective node as an empty collection (e.g., the Algorithm & Data Structure in this example); the circle is a common node as mentioned learning object that contains actual content; and the triangle is a reference node as any supplementary materials (e.g.,

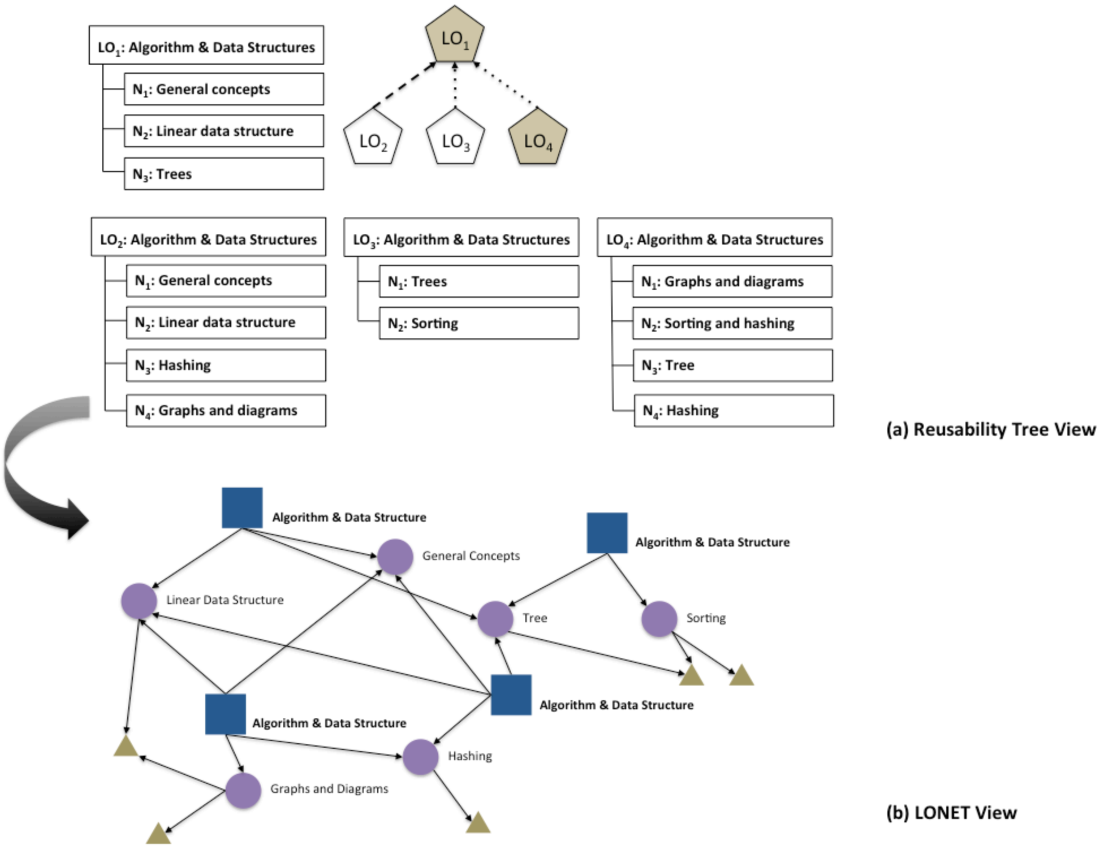


Figure 3-1Illustrations of Reusability Tree & LONET (Partially Selected)

video and web page) that support the main objective but is not shown in Reusability Tree.

3.2.2 The General Expression of LONET

With the relations from alteration history, inherited from Reusability Tree, the LONET is defined as a weighted digraph expressed by

$$G_{LONET} = (V, E, W) \tag{3.1}$$

The notation V indicates the vertice set, $V = \{v_1, v_2, \dots, v_n\}$, and is expressed by

$$V = \{x_i|_{x\in V, i\in\mathbb{N}}\}$$

The notation E represents the edge, appending with specific correlation, that connects the vertices in V so that we have $E: V \times V$ with general expression

$$E = \{e_{ij} = (x_i, x_j)|_{\forall\, x \in V, j \in \mathbb{N}}\}$$

The weight W associated with the edge is developed to identify the strength of specific relation. It represents as $W = \{w_1, w_2, \dots, w_n\}$ and refers to a real number, $W: E \to \mathbb{R}$, denoted by

$$W = \left\{w_{ij} = |e_{ij}||_{w_{ij} \in [-1.0, 1.0]}\right\}$$

The value of the connection weight decides the direction between vertices and equals to the absolute value of e_{ij} representing as $|e_{ij}|$. If w_{ij} is positive, the direction is from v_i to v_j and, on the contrary, is from v_j to v_i. For the case of 0, it represents no direct correlation between resources; that is, other resources are required to make the connection.

The possible correlation (or interdependency) that forms the link(s) between nodes can be summarized in four categories including:

- **Prerequisite Correlation**

  This relation contains a mandatory dependency. It identifies that some otherlearning objectsare required to triggered before accessing a specific learning object. If the mentioned object does not exist, the access is considered invalid. For instance, the learning object "Linked List" shall be taught before the learning object "Tree" while performing a lecture "Data Structure."

- **Inheritance Correlation**

  This relation identifies a weak dependency, and it is optional. It indicates a specific concept which claims"usually a core concept that can be applied to other learning objects for further usage." For instance, the learning object introducing “Sorting” shares the some similar core attributes to the learning object introducing “Sorting in Tree.”

- **Reference Correlation**

  This relation possesses no dependency, and exists independently. Its concept is similar to the Inheritance Correlation and Peer Correlation, and serves as an optional item. The difference is that the learning objects with Reference Correlation would or would not share the core attributes. For instance, it can be regarded as learning objects that contained several videos or figures for describing the learning objectfor “Sorting.”

- **Peer Correlation**

  This relation establishes through minor modification of specific learning objects. It can be regarded as connections among similar nodes within the networks because of their slight differences from each other (see LO_1 and LO_2 in Figure 2-1). The difference between them can be calculated through similarity coefficient [Cost and Salzberg 1993]. For example, several learning objects might have been created by different instructors for introducing the concept “Sorting.”

LONET can be constructed in accordance with mentioned correlations between learning objects. Generally speaking, nearly all of the information regarding correlations can be obtained from the past usage experiences of lecture generation. However, it is still insufficient to determine the “Prerequisite Correlation.” For instance, there may exist lots of possibility if there is a direction from v_i to v_j. Thus, in LONET construction, the metadata (IEEE LOM) is adopted to determine predefined correlation (i.e., Prerequisite), and the Reusability Tree is used to track for others (i.e., Inheritance, Reference, and Peer).

3.2.3 Quantifying the Correlations

To quantify the weight of connection, two factors, usage frequency and its corresponding time information, are considered. The equation, $W(E)$, that implements the function, $W\colon E \to \mathbb{R}$, can be expressed as follow.

$$W(E) = \sum_{e \in w^{+}} f(e)_s \cdot t(e)_s + \sum_{e \in w^{-}} f(e)_s \cdot t(e)_s \tag{3.2}$$

where

$$w^{+}(v) = \{e \in E | w(e) = (v, k), k \in V\}, \forall v \in V$$
$$w^{-}(v) = \{e \in E | w(e) = (k, v), k \in V\}, \forall v \in V$$

In Eq. (2),$f(e)_s$ represents the frequency of use within the period s and $t(e)_s$ is the corresponding weighted coefficient. The two operators, $w^{+}(v)$ and $w^{-}(v)$, are utilized to sum the weight of opposite direction of the relation.

3.2.3.1 Adding Time-series Information

Considering the time-series information, two models, Tilted-Time Window model and Time-Fading model, are employed with the focus on integration of timeframe. The notation t is utilized to represent the timestamp of the event that the relations taken place. We then have the basic definition of the time information as

$$t : t \in \mathbb{T}$$

where $\mathbb{T}$ represents the length of the time, initializing with the system service, and $t = \{t_0, t_1, t_2, \dots, t_n\}$ indicates the specific timestamp revealing the beginning and

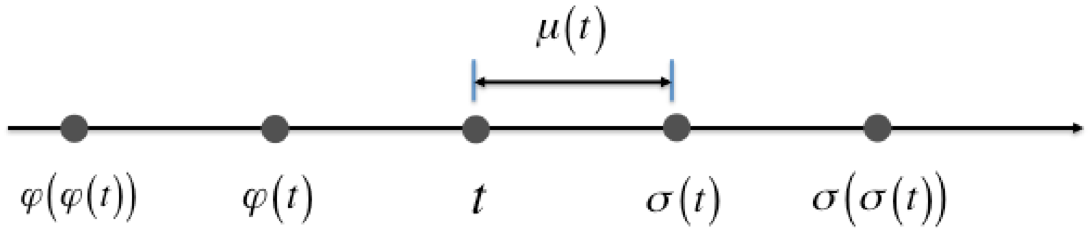


Figure 3-2 Illustration of the applied time-seriesinformation

end of the event. The time frame can be obtained through the difference between consecutive timestamps as shown in Figure 3-2. Note that$|t_{n+1} - t_n|$ may not equal to $|t_n - t_{n-1}|$ where the notation, n, stands for the index operator.

To make accurate the use of the time, two operators, $\sigma(t_n)$ and $\varphi(t_n)$, are defined to locate the exact timeframe, $\mu(t)_n$, forward or backward of the given time, t_n, respectively.

$$\sigma(t_n) = fw\{s \in \mathbb{T} : s > t_n, n \in \mathbb{N}\}$$
$$\varphi(t_n) = bw\{s \in \mathbb{T} : s < t_n, n \in \mathbb{N}\}$$

The $\sigma(t_n)$ is used to locate the time after the given time, t_n, while the $\varphi(t_n)$ is adopted for the $t_n$where the index operator $s$is utilized to compare with the current time and the given time. To obtain the timeframe, the following equations are applied.

$$\mu(t_n) = \sigma(t_n)_s - t_n = t_n - \varphi(t_n)_s$$

The timeframe is determined by the given time, t_n, and also causes the difference of extra weighted value adding to he original weight, W, of the edge in graph. The time function, $t : \mathbb{T} \to \mathbb{R}$, is proposed to obtain the weighted coefficient, $t(e)$, within specific timeframe $\mu(t_n)$. The function is generally expressed by

$$t(e) = \{t(e)_s | t(e) \in [0.0, 1.0]\}$$

The length of selected timeframe is divided into smaller units to determine the exact weighted coefficient of each period within the timeframe. For example, the unit 'hour' can be used if the length of $\mu(t_n)$ is a day. The 'day' is also acceptable if the original length of $\mu(t_n)$is set to 'month' and/or 'year'. That is, the weighted coefficient, appended on the timeframe, can be obtained dynamically in accordance with the period a connection (i.e. relation) exists and the occurrence frequency a connection has. The weighted coefficient is utilized to highlight the significant information during the specific period within timeframe.

With the definition, the weighted coefficient of each period within the given timeframe is available to be obtained. The following equation is applied to implement the function $t(e)$ expressed by

$$t(e)_s = \frac{D_{q-s+1}}{\sum_{s=1}^{q} D_s} \tag{3.3}$$

where D represents the unit, q is the sum of separate units and s is the indexer.

For instance, a length of timeframe, 3 months (assuming that 30 days per month in this instance) and 10 days, is given before the current timestamp t. The backward operator is then applied, and we have $\mu(t_1) = t_1 - \varphi(t_1) = 100$. The unit is set to a day by default. Then we separate the timeframe into smaller periods, say 1 day, 15 days, and 84 days, in backward order to compute the weighted coefficient for each of the period. Thus the parameters are listed

$$q = s = 3, D_1 = 1, D_1 = 15, D_3 = 84$$

and the weighted coefficient to each period is then obtained through Eq. (3).

$$t(e)_1 = \frac{D_{3-1+1}}{\sum D} = \frac{84}{1+15+84} = 0.84$$

The rests, $t(e)_2$ and $t(e)_3$, can be obtained through the same calculation processand should be 0.15 and 0.01 respectively.

3.2.3.2 The Usage Frequency of Correlation

After the time information, we go further to discuss the quantification of the frequency of use among the shared resources. Here the frequency of use of the resources concentrates on the co-occurrences, with other resources, the resource contains while performing a specific learning activity. The Hebbian rule (see Figure 3-3) is employed to highlight the frequently visitedlinks (or significant connections). The general expression is implemented as shown in Eq. (4).

$$f(e)_s = \frac{\sum \left(C(e_{ij}) \cdot (1 + H_{coe}) \right)}{m} \tag{3.4}$$

where$C(e_{ij})$ represents the involved correlation, m identifies the numbers of the homogenouscorrelation the selected connection e_{ij} has. According to the definition of correlations, the notation $C(e_{ij})$ is then expressed by

$$C(e_{ij}) = \{C_1 | C_2\}$$

where

$$C_1 = \begin{cases} 1, \text{ if prerequisite correlation exists between } v_i \text{ and } v_j \\ -1, \text{ if prerequisite correlation exists between } v_i \text{ and } v_j \\ 0, \text{ otherwise} \end{cases}$$

$$C_2 = K(e_{ij}), \text{ if prerequisite correlation exists between } v_i \text{ and } v_j$$

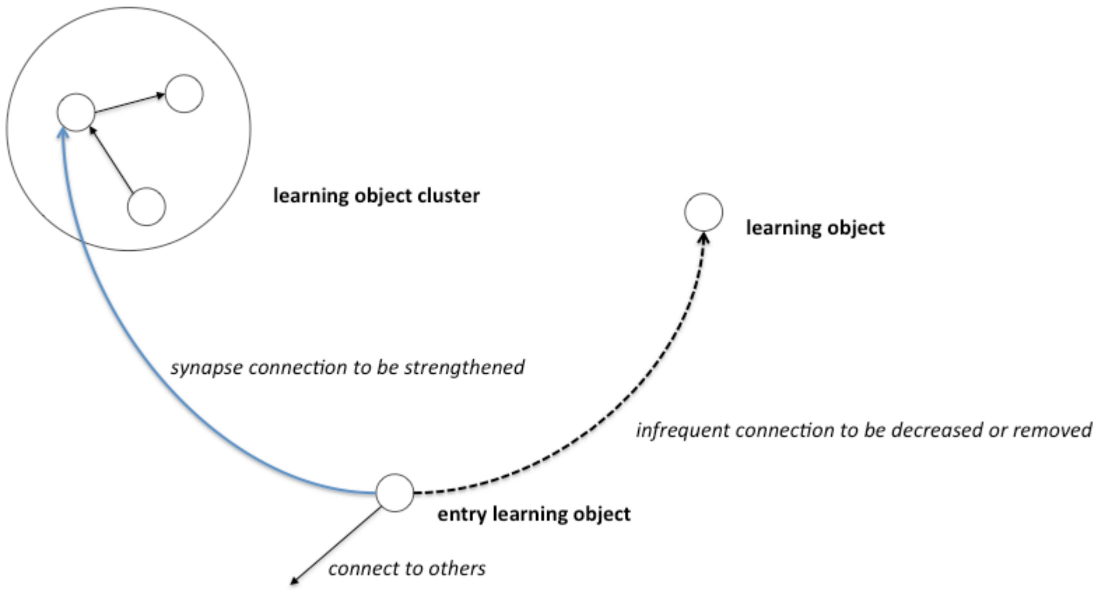


Figure 3-3 Illustration of hebbian rule

For $C(e_{ij})$, it can be categorized into two (i.e., C_1 and C_2) in accordance with the independence. A learning object with mandatory correlation indicates a connection that cannot be violated.It must comply with the direction while beingaccessed, and thus, has a higher weight than other correlations have. Hence, the correlation belongs to “Prerequisite” and “No Relation” is categorized in C_1 where the value to “Prerequisite” equals to 1 or -1, and 0 for “No Relation”. The weight of the rests (i.e., Inheritance, Reference, and Peer) are assigned by$K(e_{ij})$ where the range is located between -1 to 1 but not equals to both of them.

In the past experiences to lecture generation, lots of correlations are applicable to be used to connect learning objects. For instance, the learning object “Sorting in Tree” is inherited from the learning object “Sorting”; and the learning object “Sorting in Tree” can also be a reference to introduce its parent learning object. In other words, multiple types of correlation (i.e. numbers of connection) can exist between nodes in LONET. The variable m indicates the number of links between connected nodes.

The notation $f(e)$ is interpreted as the pattern, from v_i to v_j, applied between specific learning objects. Considering the Hebbian Rule, the weight of link shall be raised if it is triggered in accordance with existing patterns. On the contrary, it shall be reduced if the pattern is modified (i.e., only one of them is adopted). This approach is applied dynamically to adjust the weight of link and to highlight the pattern that is often visited. Thus, in Eq. (4), the additional weight, considering frequency of usage, is added to $f(e)$ by H_{coe} thatis expressed by

$$H_{coe} = \varphi \sum \overrightarrow{e_{ij}} \overrightarrow{e_{ij}}^{T} \tag{3.5}$$

The connection from v_i to v_j is converted to the vector, $\overrightarrow{e_{ij}}$, with the length representing the frequency of use. Note that the weight does not cause the difference on the length. In order to avoid the extreme value, the vector $\overrightarrow{e_{ij}}$ is then projected onto a virtual coordinate space to obtain additional increment or reduction to w_{ij} under normalization. That is, the obtained value will be relative to every other resource connected to the datum resource. Anequilibrium coefficient φ is set in accordance with the directional relation (or usage direction) between resources. Its value ranges from -1 to 1, but notequals to 0 since the existing pattern identifies that there is at least one existing connection between the designed resources on the graph. With the normalization, the value of H_{coe} is also regarded as the credibility of specific patterns (e.g. the significant connection).

3.3 Metrics for Network Mining and Analysis

The fundamental issue of Social Network is how nodes within a specific network can be described and quantified, to produce the corresponding positional importance [Wassermannand Faust 1994]. In this section, the metrics to mine the implicit information, as factors to achieve lecture generation in LONET, are addressed.

3.3.1 Degree Centrality

The *degree centrality* identifies the links, incoming and outgoing, which a specific node $v_i$may have within LONET. It isexpressed by

$$DegCent(v_i) = \frac{InDeg(v_i) + OutDeg(v_i)}{2} \tag{3.6}$$

where $InDeg(v_i) \neq 0$ or $OutDet(v_i) \neq 0$

The obtained value reflects the importance of nodes in LONET, and explains why a node is needed by other nodes or needs other nodes to support its existence. From the perspective of reusability, the path can be produced through the connection of links, which identify the possible sequence of construction of specific objectives associated to nodes.

For a single node v_i, its degree centrality will be the average of every incoming links, $InDeg(v_i)$, and outgoing links, $OutDeg(v_i)$. Note that the degree centrality of a specific node v_i is not equal tozero since every node was generated for specific objective(s), so that it must contain, at least, one incoming or outgoing link. The metrics for obtaining degree of incoming and outgoing will be discussed as follows.

3.3.1.1 InDegree

The number of incoming links indicates the popularity (or degree of attention) a node has. The node with many incoming links is regarded as a component often used for specific objectives. The PageRank algorithm is adopted to be the basis for quantifying the incoming links. In addition, the time-series issue is also concerned to be a factor that highlights the importance during different timescales. The formula can be simplified as in Eq. (7).

$$InDeg(v_i) = \delta \frac{\sum T(v_{ji}) R_{in}(Ts, v_{ji})}{R_{in}(v_{ji})} \tag{3.7}$$

where $R_{in}(Ts, v_{ji})$denotes the incoming links, from v_j to v_i, in a selected timescale Ts. The least measurement unit for Ts is assumed to be one day, but can be adjusted by an administrator or instructor. The variable δ (with default value is 1) is used to balance the outcome once if the extreme value happens. The variable $T(v_{ji}) \rightarrow \{t(e)_s\}$ indicates the additional weight that is assigned to incoming links to node v_i, and is applicable to be obtained through Eq. (3).

3.3.1.2 Out Degree

Similarly, for nodes used to generate lectures, they are bound to have outgoing links in accordance with the definition in the IMS Simple Sequencing [Yang et al. 2004]. The exception can only happen in the starting node, which may have lots of outgoing links and no incoming link. For the end nodes, the types of missing links are opposite. The outgoing link, from v_i to v_j, can be calculated through Eq. (8).

$$OutDeg(v_i) = \delta \frac{\sum T(v_{ij}) R_{out}(Ts, v_{ij})}{R_{out}(v_{ij})} \tag{3.8}$$

The major difference between Eq. (7) and (8) is the direction of links, and the weighted values for the links in different timescales.

3.3.2 The Eigenvalue

Eigenvalue is used to find out the contributing source within LONET by assigning relative values to each network participants. It is originated from the Eigenvector. From another viewpoint, a node with high Eigenvalue must have lots of incoming links or has been linked by nodes with higher InDegree (or authority). This value can be used to rank the nodes within LONET by

$$EiVal(v_i) = InDeg(v_i) + \frac{|w_{ij}| \cdot InDeg(v_j)}{\sum|w_{ij}|} \tag{3.9}$$

In Eq. (9), the direct interaction between two connected nodes is concentrated when obtaining the Eigenvalue. That is, there is no intermediate node existing. With this definition, the Eigenvalue of a node v_i depends on the percentage of InDegree value that the connected nodes (i.e. v_j) have. The percentage is then determined by the numbers of direct connection from v_j to v_i.In this study, the Eigenvalue is utilized to obtain the possible entry node when receiving the query, since a node with a high Eigenvalue is considered to have more links and can be the basis for deciding a search direction.

3.3.3 Betweenness Centrality

In general, one shortest path, at least, exists to connect any two nodes in LONET. The *betweenness centrality* is used to quantify how many times a specific node appears in the shortest paths connecting all other nodes. A node with high betweenness centrality is considered an important node within the network, and is regarded as a bridge that connects two learning objects or learning object clusters, which have no direct link. If the node is removed from the network, it may cause fracture. The value can be obtained by

$$BetCent(v_i) = \frac{\delta}{|R|^2} \sum_{j \neq i}^{|R|} \sum_{l \neq j, l \neq i}^{|R|} \frac{\kappa_{lj}(v_i)}{\kappa_{lj}} \tag{3.10}$$

where κ_{lj} indicates the sum of all pairs of shortest distances going from v_l to v_j, and $\kappa_{lj}(v_i)$ represents the sum of shortest distances from v_l to v_j via v_i. And $|R|$, which is equal to $|R_{in} + R_{out}|$, indicates the sum of links that v_i has. To obtain the average path lengths, we utilize $|R|^2$ as a denominator for normalization. The size of $BetCent(v_i)$ may affect the reusability of LONET (i.e.,the composition of lecture path generation). This situation often occurs in arrangement of nodes used to describe the same subject. For instance, it is the issue that how the sequence between "Linked List" and "Queue" can be arranged when introducing the "Data Structure."

3.3.4 Closeness Centrality

The nodes are connected through the links that can also be quantified to indicate the distance between nodes. The closeness centrality indicates a node that has the shortest path to access every other node within the network. In LONET, only the nodes in a specific area, close to a target cluster, will be calculated to obtain their closeness centrality. It will lead to a relative value for selecting alternative nodes when instructors make changes to the original lecture generation. The closeness centrality of a node can be obtained through

$$CloseCent(v_i) = \sum_{i \neq j} [d_{ij}]^{-1} \tag{3.11}$$

where d_{ij} denotes the distance between v_i and v_j and is formed by

$$d_{ij} = \sum \left| \vec{v_i} - \vec{v_j} \right| \tag{3.12}$$

The closeness centrality of a node is defined by the inverse of the average length of the shortest paths to/from all the other nodes in LONET. The value will be taken as the first visit neighborhood to reproduce the following path in LONET if the changes have been made. For instance, an instructor would like to change the original sequence, from “Linked List” to “Queue,” when generating lecture introducing “Data Structure.” In this situation, every nodes connected by “Queue” shall be rearranged (or the path of “Linked List” will be removed). The detailed algorithm regarding lecture template generation based on LONET is discussed in ChapterIV.

3.4 An Instance

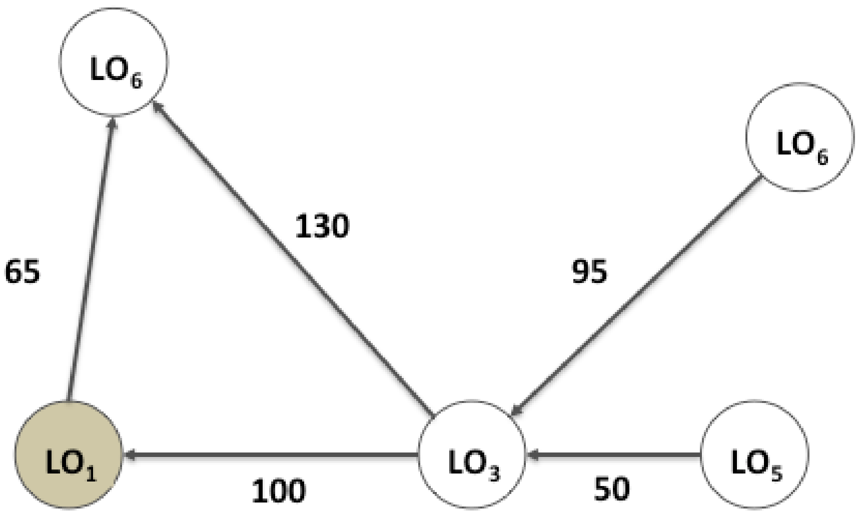


Figure 3-4 Nodes partially selected from LONET

An instance is given to demonstrate how the mining metrics are applied in LONET. Five nodes, v_1 (LO_1), v_2 (LO_2), v_3 (LO_3), v_5 (LO_5) and v_6 (LO_6), which form a small group are selected. The relations between them are shown in Figure 3-4. The node v_1 is taken as the basis. The incoming link of v_1, $R_{in}(v_{31})$, represents the partial link from v_3 to v_1 while the outgoing link of v_1, $R_{out}(v_{12})$, is the link from v_1 to v_2.

First, we assume that the age (i.e., existing timescale), of v_1 is three months (e.g., 30 days in a month), and 12.5 days, say 102.5 days in total. The incoming/outgoing links of v_1 are 55/30 times in the first three months, 25/20 times in following twelve days, and 20/15 times in the latest half-day. And the sum of outgoing links from v_3 to v_2 is 130. In this situation, we can separate the timescale into three parts:

$$D_1 = 0.5, D_2 = 12, D_3 = 90, s = 3$$

and the weight of each incoming/outgoing timescale can be obtained through Eq.(3):

$$t(v_{31})_1 = t(v_{12})_1 = \frac{D_{3-1+1}}{D_1 + D_2 + D_3} = \frac{90}{102.5} = 0.878$$

The weights to the rest timescales of v_1, 0.117 and 0.004 for the second and the third timescale respectively, can be obtained through the same process.Thus, the degree centrality of v_1 can be obtained. The detail process, with default value of variable δ, is given as follows.

$$InDeg(v_1) = \frac{\sum T(v_{31}) R_{in}(Ts, v_{31})}{R_{in}(v_{31})} = \frac{(0.878 \cdot 20 + 0.117 \cdot 20 + 0.004 \cdot 55)}{20 + 25 + 55} = 0.207$$

$$OutDeg(v_1) = \frac{\sum T(v_{12}) R_{out}(Ts, v_{12})}{R_{out}(v_{12})} = \frac{(0.878 \cdot 15 + 0.117 \cdot 20 + 0.004 \cdot 30)}{15 + 20 + 30} = 0.240$$

$$DegCent(v_1) == \frac{0.207 + 0.240}{2} = 0.224$$

After obtaining the degree centrality, including InDegree and OutDegree, we go further to calculate the Eigenvalue of v_1. The concept of Eigenvalue is similar to the ranking algorithm used in search engines. The authority of a source node will affect the value. Thus, to start to calculate the Eigenvalue of v_1, the InDegree of v_3 is required. We assume the nodes, v_3, v_5 and v_6, form another group as well in LONET, and the relations between them is shown in Figure .

Same calculation is conducted to obtain the InDegree of v_3. Assuming that the age of v_5 is 1 month and 5 days, and the age of v_6 is 15.5 days. The incoming links from v_5 to v_3 are 40 in the first month and 10 in the following 5 days, and the incoming links from v_6 to v_3 are 60 in the first half-month (15 days) and 35 in the following 0.5 days. Through the definition, the weight of each timescale is calculated through Eq.(3) and returns that$t(v_{53})_1$ is 0.857, $t(v_{53})_2$ is 0.14; and $t(v_{63})_1$ is 0.968, $t(v_{63})_2$ is 0.033. The Eq.(7) is then utilized to obtain the InDegree of v_3 as follows.

$$InDeg(v_3) = \frac{\sum T(v_{53}) R_{in}(Ts, v_{53})}{R_{in}(v_{53})} + \frac{\sum T(v_{63}) R_{in}(Ts, v_{63})}{R_{in}(v_{63})}$$

$$= \frac{(0.85 \cdot 10 + 0.143 \cdot 40)}{40 + 10} + \frac{(0.968 \cdot 35 + 0.033 \cdot 60)}{35 + 65} = 0.663$$

The InDegree of $v_3$is divided into two according to the portion of its outgoing links, andthus, the one which is designated to add to v_1 can be obtained through Eq. (9). The detail process is shown below.

$$EiVal(v_1) = InDeg(v_1) + \frac{|w_{13}| \cdot InDeg(v_3)}{|w_{13}| + |w_{23}|} = 0.207 + \frac{100 \cdot 0.663}{100 + 130} = 0.495$$

The betweenness centrality and closeness centrality of v_1 is then considered. In accordance with the definition of Eq.(10), there are two kinds of links that connect v_2 and v_3. One goes directly between v_2 and v_3, while another one goes via v_1. In this situation, we can obtain $\kappa_{23}(v_1)$ equals to 165, the sum of $R_{\text{out}}(v_{12})$ and $R_{\text{in}}(v_{31})$, and κ_{23} equals to 130, the sum of $R_{\text{out}}(v_{32})$. Then, the value of betweenness centrality of v_1 can be obtained by:

$$BetCent(v_1) = \frac{1}{|165|^2} \cdot \frac{165}{130} = 0.007$$

And, lastly, the closeness centrality depends on the distance between connected nodes. In distance learning, the optimal way to calculate the distance between objects is to adopt the corresponding metadata elements. We assume the metadata (IEEE LOM format) of v_1 and v_2 are shown in Table 3-1.

Table 3-1Corresponding Metadata Elements of v_1 and v_2

Metadata of v_1 (Partially Selected)	Metadata of v_2 (Partially Selected)

… <general> <title>Algorithm & Data Structures</title> <language>eng</language> <keyword>data structure, intro., overview</keyword> <coverage>Higher education</coverage> </general> … <technical> <format>text/.html</format> <size>29234</size> <location> 2097/e4a921.html</location> </technical> <educational> <learningresType>lecture</learningresType> <IntendRole >University</IntendRole > <typageRange>19-22</typageRange> <difficulty>medium</difficulty> <typTime>3 hour/hours</typTime> </educational> … <annotation> <date> <datetime>2008/10/1</datetime> </date> <description> <string language="eng"> web content resource </string> </description> </annotation> …	… <general> <title>Algorithm & Data Structures</title> <language>eng, jpn, chi</language> <keyword> data structure, intro., hashing, overview</keyword> <coverage>Higher education</coverage> </general> … <technical> <format>text/.html</format> <size>30173</size> <location> 2097/ef349.html</location> </technical> <educational> <learningresType>lecture</learningresType> <typageRange>22-25</typageRange> <difficulty>medium</difficulty> <typTime>2.5 hour/hours</typTime> </educational> … <annotation> <date> <datetime>2008/12/21</datetime> </date> <description> <string language="eng">This is the web content resource from MINE Registry</string> </description> </annotation> …

The elements with value are listed since not all of the elements are required in IEEE LOM specification. In Table 3-1, three major categories, General, Educational and Annotation, are represented, and for those, LifeCycle, Meta Metadata, Rights, Relation and Classification, contain no exact value (given empty string by default while being adopted) are not represented.

To facilitate the calculation process, the simple cosine similarity [Wu et al. 2008] is applied. Each element in specific category is operated and is assigned the weight (i.e. 1 by default) once the value(s) is matched. The “<title>”, “<language>”, “<keyword>” and “coverage” are involved in this example.The detail is shown below.

$$d_{12\_General} = 1 + \frac{1}{3} + \frac{3}{4} + \frac{1}{4} = 0.771$$

The similarities in the rest metadata categories areoperated through the same process. They are “0” in LifeCycle, “0” in MetaMetadata, “1” in Technical, “0.567” in Education, “0” in Rights, “0” in Relation, “0.556” in Annotation and “0” in Classification, respectively. The distance, d_{12}, of v_1 is then obtained.

$$d_{12} = 0.771 + 0 + 0 + 1 + 0.567 + 0 + 0 + 0.556 + 0 = 2.894$$

And the closeness centrality of $v_1$is also applicable to be obtained through:

$$CloseCent(v_1) = {d_{12}}^{-1} = (2.894)^{-1} = 0.346$$

CHAPTER IV. APPLIED SEARCH IN LONET

Following the LONET, three automated mechanisms are proposed to facilitate the search process. The fundamental mechanism, weighting and ranking of learning object, is introduced after the definition of IEEE LOM-based search criteria. The search guidance mechanism that achieves progressive search suggestions provision is then addressed. In the end, the reusable path (i.e. lecture template and adaptive route) that is generated according to the past experiencesis discussed.

4.1 The Search Criteria

IEEE LOM specification has defined 77 elements in 9 categories to achieve reusability of learning objects. In accordance with the attribute of mentioned elements, five major groups [Lin et al. 2009] to assist the retrieval of learning object in MINE Registry are defined including Precise Criteria, Incremental Criteria, Precedence Criteria, Time/Duration Criteria, and Single/Multiple Choice Criteria.

As mentioned, not all of the elements defined in LOM are required. And thus, five elements [Rivera et al. 2004] are selected based on experiments on concrete usage scenario. The selected elements including keyword and coverage in “General” category, learningResourceType, Difficulty and typicalLearningTime in “Education” category are adopted to strengthen the educational usage.

4.2 Weighting & Ranking of Learning Object

Though the LONET gives a well-structured network for learning objects management, how the learning object is retrieved via online search service has not been mentioned. This section, based on systematic perspective, provides a solution to the mentioned issue. Three algorithms, weighting,ranking and adaptive re-ranking are involved.

4.2.1 Learning Object Weighting based on Time-series Issue

To start out, the factors that cause influence upon the weight of learning object is emphasized.The number of citation, as known as frequency of download in this study, is adopted concerning the reuse scenario of learning objects. The records are recorded through the access of implemented search service in the repository system.The first notation, $CR(v_i)$, denotes the *Citation Reference* of a learning object v_i which refers to a positive integer. The significance of a specific learning is then realized through $CR(v_i)$. It is obvious that the higher the $CR(v_i)$ is, the more popular the v_i is. In this situation, two corresponding notations are defined including:

- **Author Reference (AR):**

  The *Author Reference* identifies the authority of an instructor who creates the mentioned learning objectv_i, and depends on the number of citations the learning object v_i has. The citation of a newly created learning object is set, by default, to zero once it is selected for further operation.

- **Time Reference (TR):**

The Time Reference identifies the number of citations which is made by users within a specific timescale (i.e., Ts). This reference is applied to deal with those learning objects which obtain increasing number of citations within timescales but is accessed just a few times in the past or in the following days. This situation cannot be measured simply through the previous reference (i.e., $\boldsymbol{AR}$) since the overall situation should be taken into consideration. Thus, the time reference is applied to record duration that learning object exists and labels corresponding citations to improve the accuracy of the obtained weight.

With the defined reference factors (CR, AR and TR), two issues are then confronted without making normalization. That is, for examples, (a) the value of $CR(v_i)$ is increasing to extreme high (e.g., infinity) or extreme low (e.g., 1) if no united measurement unit is defined, and (b)it is required to remove some unnecessary data in the past to avoid the unexpected noises (e.g., inaccurate outcome). And thus, the factors CR and AR are re-defined according to the TR.

Inspired by the Web 2.0 and the issues of social network, the user feedback which is similar to applied evaluation strategy (e.g., Youtube and Google Social Search) is taken into consideration. The social factor represented by $fb(v_i)$ identifies the user feedback regarding a specific learning object v_i.

With the above definitions, the weighting algorithm can be expressed by

$$ref(v_\mathrm{i}) = \alpha \frac{\sum T(v_i) \cdot CR(Ts, v_i)}{\sum CR(v_\mathrm{i})} + \beta \frac{\sum T(v_i) \cdot AR(Ts, v_i)}{\sum AR(v_\mathrm{i})} + \gamma \frac{fb(Ts, v_\mathrm{i})}{\sum fb(v_\mathrm{i})} \qquad (4.1)$$

where$\alpha + \beta + \gamma = 1$

$CR(Ts, v_i)$denotes the citations for a specific learning object $v_i$withina selected timescale Ts, and $\sum CR(v_\mathrm{i})$ is the sum of citation that v_i has since it is created. The measurement unit for $Ts$is set to be aday by default andis applicable to be revised by an administrator or instructor.The variable $T(v_i)$ is same as one in Eq. (7) and is implemented by $t(e)_s$ in Eq. (3).Three thresholds, α,β and γ, are defined to avoid the extreme value while the use of specific timescale Ts provides a solution regarding removal of unnecessary data. The variable $fb(Ts, v_i)$is same as $CR(Ts, v_i)$ and $AR(Ts, v_i)$ and identifies the feedbacks, in form ofreal number, which are left by users. The type of feedback can be categorized into two, Negative and Positive, and represents in form of five relevancy coefficient (from -1 to 1). The implementation of the search interface as well as the collection of user feedback is shown in Chapter VI. The algorithm that is applied to compute the user feedback is described in Table 4-1 (top of next page).

Atemporary threshold $R_i$is utilized to represent the search results with three variables (Rv, Po, Ne) and examines the accuracy of the feedbacks for v_i in specific timescale Ts. Poidentifies the positive feedbacks $v_i$has.On the contrary, Ne is for the negative feedbacks. The variable Rv is the current scores thatv_i has. Two thresholds (Po, Ne) are used to normalize the polarization feedbacks which might cause the imbalance. The value of $Rv$is then mapped to $FB(Ts, LO_i)$.

Table 4-1 Algorithm for user feedback calculation

Input: record of user feedback from repository

Output:a normalized number Rv

1 : set rating scores array,Rs, $Rs = \{-1, -0.5, 0, 0.5, 1\}$

2 :given the number to search results R_i forv_i, $i = \{0, 1, 2, \dots, n\}$, and check selected timescale (Ts) forv_i

3 : initialize R_i with current rating values and numbers of positive and negative feedbacks,$R_i(Rv, Po, Ne)$

4 : check scores (ρ_t) given by users, $t = \{0, 1, 2, \dots, n\}$

5 : calculate the numbers of positive and negative feedbacks

if ρ_t< 0 { Ne += -1 }else { P_o += 1 }

6 : if Rv> 0

check if (($\sum \rho_t$ ≥ 0) & (Po>Ne)) {Rv+= $\frac{\sum \rho_t}{|Po|}$ }

else if (($\sum \rho_t$< 0) & (Po<Ne)) { Rv += $\frac{\sum \rho_t}{|Po-Ne|}$ }

else if Rv< 0

check if (($\sum \rho_t$ ≤ 0) & (Po<Ne)) { Rv+= $\frac{\sum \rho_t}{|Ne|}$ }

else if (($\sum \rho_t$> 0) & (Po>Ne)) { Rv += $\frac{\sum \rho_t}{|Ne-Po|}$ }

else

check if ($\sum \rho_t$> 0) { Rv+= $\frac{\sum \rho_t}{|Po|}$ }

else if ($\sum \rho_t$< 0) { Rv+= $\frac{\sum \rho_t}{|Ne|}$ }

7 : return Rv

For instance, the age of a learning object $v_1$is 2 month and 6.5 days (66.5 days in total) which is shown in Figure 4-1. The citations are 650 times in two month, 250 times in 6 days and 100 times in the last half day. In this situation, we can obtain the additional weight, through Eq. (3.3), for each timescale to be 0.902, 0.09 and 0.008 respectively from present to the past.

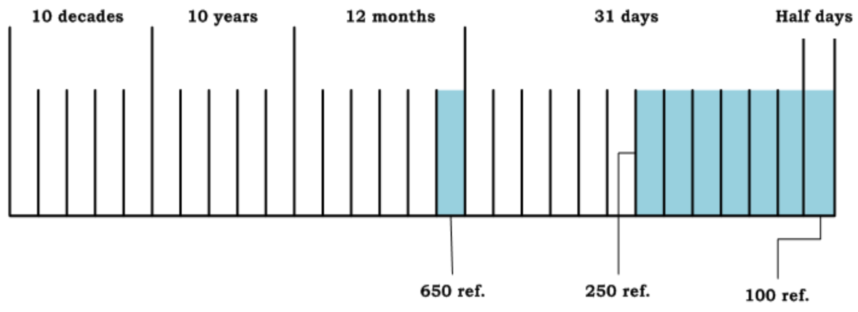


Figure 4-1 Age of v_1 in selected timescale

Assuming that v_1 is the only learning object created by an author,and thus the β shall be 0. According to the record inthe repository, there are 375 feedbacks given in recent 66.5 days wherethe sum is 500 regarding user feedbackof v_1 in a year. Thus, the selected timescale Ts for v_1 should be 0.18 (66.5 days in 365 days).With the conditions, the detail process of weighting process is shown in Table 4-2.

Table 4-2The calculation process for obtaining the weight of v_1

Weight of v_1

1. The period, citations and additional weight:

Period	Citations	Additional Weight
0.5 days	100	0.902
6 days	250	0.090
60 day (2 months)	650	0.008

2. The thresholds for the example:

   $\beta = 0$and assume that α = 0.5 and γ = 0.5

3. The user feedback of v_1 will be:

   $FB(0.18, v_1) = 375$

   $FB(1, v_1) = 500$

4. Placing the thresholds and parameters into the equation:

$$ref(v_1) = 0.5 \cdot \frac{0.902 \cdot 100 + 0.09 \cdot 250 + 0.008 \cdot 650}{100 + 250 + 650} + 0 + 0.5 \cdot \frac{375}{500} = 0.4340$$

4.2.2 Learning Objects Ranking

After obtaining the weight of learning object, the ranking issue is discussed.The PageRank algorithm is revised concerning the Reusability Tree for educational purpose.

To clarify the actual relation between learning objects in Reusability Tree, the use of similarity function is required. In this study, the similarity/diversity functions are refined from [Lin et al. 2009]. In addition, to highlight the educational usage, nine elements from IEEE LOM are selected including Title, Language, Keyword and Coverage in “General” category and learningResourceType, intendedEndUserRole, typicalAgeRange, difficulty andtypicalLearningTime in “Education” category.

Table 4-3 Similarity algorithm for learning objects comparison

---

Input:metadata of learning objects

Output: a numeric number sim_{tmp}

1: initialize elements array $E_m[r_w][s_x]$, $E_n[r_y][s_z]$ for LO_m and LO_n, wherer_w, r_y ∈ {LOM Categories}, and s_x, s_z∈ {elements in corresponding category}

2: initialize the selected elements array SE , SE ∈{title, language, …, typicalLearningTime}

3: for each element in E_m and E_n{

check if (r_w equals to r_y)

if (s_x not exists in SE)

$$Sim_{tmp} += \sum \frac{\overrightarrow{\mathrm{s_x}}\overrightarrow{\mathrm{s_z}}}{|\overrightarrow{\mathrm{s_x}}||\overrightarrow{\mathrm{s_z}}|}$$

else

$$Sim_{tmp} += \frac{\mathrm{SE}'}{\mathrm{SE}} \sum \frac{\overrightarrow{\mathrm{s_x}}\overrightarrow{\mathrm{s_z}}}{|\overrightarrow{\mathrm{s_x}}||\overrightarrow{\mathrm{s_z}}|}$$

4: return sim_{tmp}

In the algorithm, the cosine similarity is applied as the main metrics to integrate with selected IEEE LOM elements SE. The original similarity between learning objects is adjusted based on the existence of matched elements (SE').According to Eq. (4.1) and the similarity algorithm, the ranking formula for learning object is then defined as follow:

$$rank(v_{\mathrm{i}}) = ref(v_{\mathrm{i}}) + \frac{\sum sim(v_{\mathrm{i}}, v_{\mathrm{j}}) \cdot ref(v_{\mathrm{j}})}{n} \tag{4.2}$$

As shown, the equation contains two portions. The first is the original weight obtained through Eq. (4.1) while the second is the additional weight originated from the inheritance relation recorded in Reusability Tree.

The Eq. (4.1) and Eq. (4.2) are applied in the CORDRA-based repository (i.e., MINE Registry) to enhance the searchability of learning objects.The result of Eq. (4.1) specifies the significant degree that a learning object possesses within a specific timescale while the Eq. (4.2) identifies the order (e.g., in decrease order by default) that learning objects are displayed in the search result list by integrating the PageRank algorithm withthe inheritance relations recorded in the Reusability Tree.

As a continuative instance from section 4.2.1, we assume that v_2 is a derivative learning object from v_1.The metadata of v_1 and v_2 are the same as one shown in Table 3-1.We concentrate on the selected elements, and thus, we have 0.771 in General category and 0.567 in Education category, and obtain the similarity between v_1 and v_2 at 0.669 (average of the similarity obtained from the selected elements).

Based on the similarity, the rests are proceeded. The detail calculation process including the weight of v_2 and ranking issue is shown in Table 4-4.

Table 4-4The calculation process for weight of v_2 and ranking value of v_1

Weight of v_2

1. The period, citations and additional weight:

Period	Citations	Additional Weight
0.5 days	20	0.822
6 days	100	0.164
30 day (1 month)	300	0.014

2. Threshold for our proposed formula:

   β = 0 and assume thatα = 0.5 andγ = 0.5

3. The feedback information ofv_2 will be:

   $FB(0.10, LO_2) = 200$

   $FB(1, LO_2) = 500$

4. Placing the thresholds and parameters into the equation:

$$ref(v_2) = 0.5 \cdot \frac{(0.822 \cdot 20 + 0.164 \cdot 100 + 0.014 \cdot 300)}{20 + 100 + 300} + 0 + 0.5 \cdot \frac{200}{500} = \mathbf{0.2441}$$

RankingValue of v_1

1. The basic weight of v_1 andv_2

   $ref(v_1) = 0.4340$

   $ref(v_2) = 0.2441$

   $n = 1$(assuming that v_2 is the only object derivative from v_1)

The ranking value of LO_1 will be:

$$rank(v_1) = 0.4340 + \frac{0.669 \cdot 0.2441}{1} = \mathbf{0.5973}$$

In this example, two learning objects (i.e., v_1 and v_2) are considered, and thus the order should be $v_1 \rightarrow v_2$(in decrease order). Once other learning objects (e.g., v_6 and v_7) are existed and are derivative from v_2 with the weight at 0.40833 and 0.31719 and the similarity corresponding to v_2 at 0.5916 and 0.4230 respectively. The ranking value of v_2 is calculated in Table 4-5.

Table 4-5The ranking value of v_2 (updated)

Ranking Value of v_2(with the existance of v_6&v_7)

1. The basic setting:

$$ref(v_6) = 0.40833, sim(v_2, v_6) = 0.5916$$
$$ref(v_7) = 0.31719, sim(v_2, v_7) = 0.4230$$

$$Rank(v_2) = 0.2441 + \frac{0.5916 \cdot 0.40833}{1} + \frac{0.4230 \cdot 0.31719}{1} = \mathbf{0.6227}$$

This calculation process is updated from the previous one in Table 4-4. It is obvious that v_2 contains higher ranking value than v_1 once derivative objects (v_6 and v_7) exist. Considering the four learning objects in this scenario, the ranking result is going to be:$v_2 \rightarrow v_1 \rightarrow v_6 \rightarrow v_7$ (in decrease order). As a result, the ranking value of a specific learning object is dynamical and determined by the weight (based on the timescale) and the relations (derivation recorded in LONET).

4.2.3 Adaptive Re-Ranking

The adaption issue is discussed in this subsection based on the previous outcomes. We consider that the search results, as well as the ranking results, should be provided adaptive in accordance with essential context which may refer to the query preference, working item and expected resource types to different users.That is, each user should obtain different ranking results even the same query criteria is given. To discriminate with the recommendation system, a re-ranking algorithm, which concentrates on reorganizing initial search results but not providing recommendations instead, is then proposed.

In this thesis, the mentioned context that causes the change of search organization includes:

- **WorkingItem**

  The working item identifies the learning activity (or lecture) that user is accessed while using the search service. It is considered that additional learning objects are required to generate the lectures. And thus, to provide learning objects which contain higher similarity with the working item are recommended.

- **Resource Types**

  There are various types of resource in our repository. The returned resources in the search result should have certain correlation that corresponds to the working item or the last few learning objects accessed by the users.

Two achieve the re-ranking process, two coefficients, AEC (Author Expect Coefficient) and RRC (Re-Ranking Coefficient), are defined to calculate and assign additional weight to the original weight, $ref(v_i)$, of learning object v_i. Before goring further, three element sets responsible for the record of mentioned factors are defined.

- **Match List (ML)**

  The selected LOM elements (see 4.1) are concerned to generate additional weight. Thus, the Match List refers to the selected LOM elements in learning objects and$ML = \{ML_n|_{n=1,2,\ldots,n}\}$.

- **Response List (RL)**

  The Response List represents the candidate learning objects that are corresponded to the input query in our repository. This process is executed in the background and returns a set of learning objects in form of an XML-based document with corresponding indexer $RL = \{RL_n|_{n=1,2,\ldots,n}\}$.

- **Match Item (MI)**

  The Match Item refers to the intersection between the ML and RL. Once the element in ML is matched to the one in RL, the statistic value, through a similarity calculation (Table 4-3), is returned. And the average through each returned item (MI_n) identifies the additional weight.

With the definition, the AEC is expressed as follow.

$$AEC(v_{\mathrm{i}}) = \pi \cdot sim(v_{\mathrm{i}}, v_{\mathrm{j}}) + (1-\pi) \cdot \frac{1}{|ML|} \sum_{i=1}^{n} \frac{MI_i}{|RI_i|} \tag{4.3}$$

where $\pi \in [0,1]$

In Eq.(4.3), we do cross matching based on different information. The different parameters lead to different weights. The thresholdπ is set to achieve the optimal balance between original query and extra weight for specific information.

Note that the AEC is not a final value to re-rank the search results. The original weight,$rank(v_{\mathrm{i}})$ through Eq.(4.2), is required to be concerned for obtaining objective result. Thus, considering the adaption issue, the weight of learning objects that is utilized for ranking should be expressed by:

$$RRC(v_{\mathrm{i}}) = rank(v_{\mathrm{i}}) + AEC(v_{\mathrm{i}}) \tag{4.4}$$

Although two equations, Eq. (4.2) and Eq. (4.4), serve for the same purpose (e.g., provide ranking result), the Eq. (4.2) is applied for the general purpose while the Eq. (4.4) is adopted when a specific user is determined.

4.3 Search Guidance

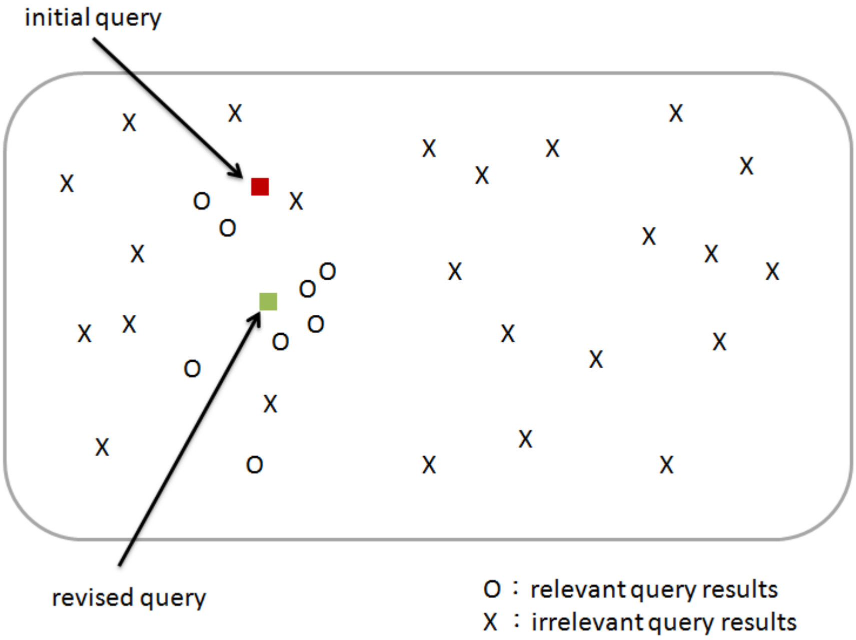


Figure 4-2 Illustration of search scenario

Following the fundamental issues (i.e., weighting and ranking), the first automated mechanism that facilitates the learning object retrieval in the repository is introduced.The *Search Guidance* conceptually identifies a progressive path that assists users in obtaining expected learning objects in the federated repositories.It aims at providing suggestion(s) instead of specified recommendations corresponding to the query. The search process is continued through adopting one of the suggestion(s) or starting over a new query.The scenario is shown inFigure 4-2. The red box represents the query (i.e., initial query for the first access). The green box represents the suggested query according to the attributes collected from the results led by initial query.To achieve this, the relevance feedback algorithm [Rocchio 1971][Roya et al. 2007] is revised based on selected LOM elements. The detail algorithm is shown in Table .

Table 4-6 Algorithm of search guidance

Input: a query, q

Output: suggestion set(s), $\overrightarrow{Q_n}$

1 : initialize the query vector, let $\overrightarrow{Q_m} \leftarrow q$

let L_q := IEEE LOM elements, $\overrightarrow{Q_m} \subseteq L_q$, D := resources in database

2 : for each search criteria C_i in $\overrightarrow{Q_m}$, ($\forall C_i \in \overrightarrow{Q_m}$)

return search results D_m, ($D_m \in D$)

3 : calculate the representative set $Z_p \in D_m$

3.1 check the similarity degree of Z_p

compare each results i in D_m, where $i = \{0,1,2, \dots, N\}$

if $Sim\left(Z_p, D_{(m,i)}\right) < Avg\left(Ref(D_m)\right)$ move to irrelevant set Z_n

3.2 calculate the suggestion coefficient S_{coe} for Z_p

for $i = 1; i++;\ i \leq number\ of\ results\ in\ D_m$

if $Ref\left(D_{(m,i)}\right) \neq 0$ $S_{coe} = \mathrm{Sim}\left(Z_p, D_{(m,i)}\right) \times Ref\left(D_{(m,i)}\right)$

else $S_{coe} = \mathrm{Sim}\left(Z_p, D_{(m,i)}\right) \times Avg\left(Ref(D_m)\right)$

3.3 reset the order in the suggestion set, $Z_p \rightarrow Z_{pr}$

3.4 calculate the coefficient of irrelevant set Z_n

check the average irrelevant degree (S_{rcoe})

$$S_{rcoe} = \sum Diversity\left(Z_n, D_{(m,i)}\right) / D_m$$

4 : check the occurrences

for each item k in Z_n, $k = \{1,2, \dots, N\}$

if ($Z_{(n,k)} \ni Z_{pr}$) remove k from Z_{pr}

5 : revise the initial query vector $\overrightarrow{Q_m}$

if ($Z_{pr} \not\ni \overrightarrow{Q_m}$) $\overrightarrow{Q_n}$ += top-10 elements from Z_{pr}

else if ($Z_n \ni \overrightarrow{Q_m}$) set $\overrightarrow{Q_n} = \overrightarrow{Q_m} - Z_n$

return $\overrightarrow{Q_n}$

The process flow can mainly be separated into the following steps:

STEP 1 :

The search criteria defined in Section 4.1 are adopted for beginning the search process in the repository. The query vector$\overrightarrow{Q_m}$ which contains the query (q) is initialized. In addition, the selected elementsL_q in LOM metadata are extracted from the existing learning objects D in repository.

STEP 2 :

The criteria $\mathrm{C_i}$ in q is utilized to compare with the resources in the repository. The matched learning objects D_m are returned in form of tree structure (i.e., the Reusability Tree).

STEP 3 :

In this step, we concentrate on obtaining the representative set (Z_p) which identifies the majority collection of returned results.The representative set simply refers to the intersection of D_m and ranked based on significant degree (S_{coe}) that is determined through the similarity comparison. Note that two major sets are concerned,suggestion set and irrelevant set, with corresponding reference coefficientS_{coe} and S_{rcoe} respectively.

STEP 4 :

With the coefficients, we then go further to determine whether the elements existed inZ_p also exists in Z_n. Once the co-occurrence exists, the matched element(s) is removed from Z_p and reorder the sequence in Z_{pr}.

STEP 5 :

The elements in representative setZ_{pr} are compared with query vector$\overrightarrow{Q_m}$. The algorithm mainly checks whether elements exist in the query vector or previous suggestion set. The top ten elements in Z_{pr} are returned in form of new query vector $\overrightarrow{Q_n}$ if the elements do not exist in $\overrightarrow{Q_m}$.Otherwise, the algorithm checks whether elements in Z_n exist in $\overrightarrow{Q_m}$ and proceed removal.

The purpose of Search Guidance is to give assistance to users, with progressive suggestions, in search process according to adaptive inputs. An instance is given that the keyword "Data Structure" is typed in the repository. The system returns the matched results to the input and generates corresponding suggestions. Assuming that the keyword "Hashing" is returned, and thus, the suggestion comes to be "Data Structure + Hashing." It is then applicable for user to follow the suggestion or to continue original query.

The mentioned processin this exampleis in a loop with interaction with users. It ends up with suspension by user. A preliminary testing result regarding the revision process is shown inTable 4-7.

Table 4-7 Instance of Query-Revision Process

Initial Query(Q_1):

keywords: photoshop

Retrieved Objects:

reusability tree: 84

learning objects: 5199

System Suggestions:

keywords: training, introduction

languages: en, zh, ja, ko

Revised Query(Q_2):

Keywords: photoshop + training

Languages: en

Retrieved Objects:

reusability tree: 63

learning objects: 4768

System Suggestions:

keywords: linear, introduction, graphic

languages: en, zh

Revised Query(Q_3):

Keywords: photoshop + introduction | training

Languages: en

Retrieved Objects:

reusability tree: 117

learning objects: 6379

System Suggestions:

keywords: toolbox, shape design

languages: en

4.4 Lecture Template Generation

A *lecture template*is considered as a possible re-composition of learning objects which have beenutilized to generate lectures for specific objectives. This concept is inspired by Search Guidance and is used to facilitate tedious process of lecture generation by reducing the time and efforts that cause toinstructors. According tothe experiences inlearning object searching, the results are sometimes isolated. In this section, in addition to providing learning object and related recommendations, the use of structural outcome (i.e., template) with educational purpose isaddressed. As the one mentioned in Section 4.3, instructors can choose to follow or make further revision of the structures. The revision will be taken as a new composition of lecture.

The lecture path is produced according to possible sequences within LONET. To achieve this, the first step is to analyze the queries from instructors, and to determine the entry node. The entry nodeis considered as a list of learning objectsreturned by searching services from repository. Each of the nodes refers to a center in a lecture web, and can be used to determine the direction to lecture path generation in accordance with its implicit properties. This iterative retrieval process will be kept to achieve continuous path. Two major portions, path generation and generated path reorganization, are involved.

4.4.1 Path Generation

The path generation refers to thebasic structural organization of learning objects in LONET. In this situation, the frequency of the learning objects usage, especially the sequence (i.e., link) between them, draws the concentration instead of the shortest path in LONET. The frequency determined by the user experience (i.e., instructor) identifies the credibility and is applicable to be the reference for path generation.The structure of the path is generated through following steps.

STEP 1 :

In the beginning, the attribute array, $Aq[r_w][s_x]$, is generated by converting the query from user in accordance with its corresponding metadata description. Then, to return the top-N items, $V_{ent}\,[v_m]$, the matched objects in LONET is ranked, from high to low, on the basis of their relative similarity. It refers to the algorithm shown in Table 4-8.

Table 4-8 Algorithm for STEP 1 of Lecture Template Generation

Input: users' query $\vec{Q}$with search criteria C_i

Output: corresponding results array, $V_{ent}\,[v_m]$, ranked by similarity

1: initialize query vector $\vec{Q}$

2: let L_q := IEEE LOM elements, where $\vec{Q} \subseteq L_q$

3: for each input criteria C_i in $\vec{Q}$

3.1 initialize query attribute array $Aq[r_w][s_x]$ for C_i, and node attribute array $Av[r_y][s_z]$ for v_i in V, where $r_w, r_y \in \{LOM\ category\}$ and

$$s_x, s_z \in \{elements\ in\ LOM\ category\}$$

3.2 calculate similarity degree between Aq and Av

for each i with $i < |V|$

$$sim_{coe}\,(Aq, Av_i) = \frac{\overrightarrow{Aq} \cdot \overrightarrow{Av}}{|\overrightarrow{Aq} \cdot||\overrightarrow{Av_i}|}$$

3.3 initialize entry nodes array, $V_{ent}\,[v_m]$, where $V_{ent} \in V$

add top-N nodes with high sim_{coe} to $V_{ent}\,[v_m]$

4 : return $V_{ent}\,[v_m]$

STEP 2:

The Swarm algorithm isrefined to be the basis for visiting every node that has connections with node v_m in V_{ent}. The nodes with a shorter path, with fewer nodes between v_m and $v_{(m,j)}$, and a higher OutDegree value will be the priority obtained. The main reason is that the node with a higher OutDegree identifies that more nodes can be accessed. In other words, the node offers more choices for the system to select. Thus, we set up a rule that if the visited node $v_{(m,j)}$ has a higher OutDegree value, a shorter distance than v_m will replace the original one. And a new entry node array $V_{ent}{'}[v_m{'}]$ will be generated. The process is shown in Table4-9.

Table 4-9 Algorithm for STEP2 of Lecture Template Generation

Input: ranked results array $V_{ent}[v_m]$

Output: entry array of candidate nodes $V_{ent}{'}[v_m{'}]$

1: reset V_{ent} by neighbor node of v_m with higher Out Degree

1.1 for each node $v_{(m,j)}$ in V_{ent}, find nearest node v_n with highest Out Degree

for *l*-th neighbor nodes $v^l_{(m,j)}, l = \{0,1,2, \dots, n\}$ of $v_{(m,j)}$

if ($InDeg(v^l_{(m,j)}) < InDeg(v_m)$ AND

$$OutDeg(v^l_{(m,j)}) > OutDeg(v_m)$$

$$v_m \leftarrow v_m{'} = v^l_{(m,j)}$$

1.2: reorder the $V_{ent}{'}[v_m{'}]$based on $OutDeg(v_m{'})$

2 : return $V_{ent}{'}[v_m{'}]$

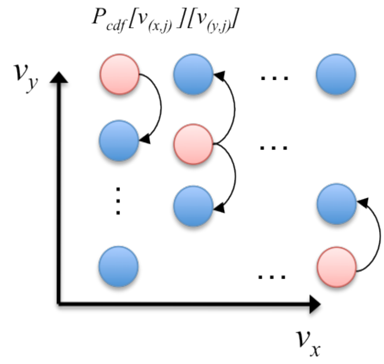


Figure 4-3Illustration of Lecture Path Generation

STEP 3 :

The returned entry array, $V_{ent}{}'[v_m{}']$, will be converted to a two-dimension vector space, $P_{cdt}[v_x][v_y]$, where v_x represents the entry nodes from $v_m{}'$ and v_y indicates the possible path to v_x as shown in Figure 4-3. The red circle represents the possible position of entry node where the others are default to $null$ for expansion. Three possible positions are involved. First, The position identifies a start node regarded as first object of a lecture (e.g., Introduction) in the left. The middle position stands for the node applicable for two-way expansions of a lecture. The right side represents the end node(e.g., Conclusion) opposite from the start node. The algorithm shown in Table 4-10 was applied.

Table 4-10 Algorithm for STEP3 of Lecture Template Generation

Input: revised candidate node array $V_{ent}{}'[v_m{}']$

Output: initialized vector space for lecture generation

1: initialize path array $P_{cdt}[v_x][v_y]$ where $v_x \in V_{ent}$, $v_y \in V$ and has direct edge e_{xy}

  convert $v_m{}'$ in V_{ent} to v_x in P_{cdt}

    for each j with $j < |v_m{}'|$

$$v_{(x,j)} \leftarrow v_{(m,j)}{}'$$

$$v_{(y,j)} \leftarrow null$$

STEP 4 :

As mentioned in STEP 3, nodes utilized for lecture generation are supposed to have correlations between each other.It is sometimes difficult to determine the entry node, the first node for path generation in LONET. An appropriate entry node is considered to have higher Out Degree than others, and may locate at any position in the path. Hence, the proposed metrics, the weight of link between two connected nodes and EigenValue, are utilized to determine the possible direction of path for those correlations that do not belong to “Prerequisite.” The possible directions can be tracked backward or forward. As stated in Chapter 3, the correlations aresummarized in four categories, and only one mandatory interdependency (the other three are optional) is existed. That is, the correlations except “Prerequisite” will not affect the generation rule of lecture path. Thus, we only consider the edge with Prerequisite correlation. If such kind of correlation exists, it will be added to the path without further consideration of the EigenValue, and the following direction will be determined. The detail algorithm is listed in Table 4-11.

Table 4-11 Algorithm for STEP4 of Lecture Template Generation

Input: revised candidate node array $V_{ent}{}'[v_m{}']$

Output: filled path array $P_{cdt}[v_x][v_y]$

1 : for each v_x

set atolerant threshold χ for candidate neighbor

nodes selection with direction dir

2 : for $\forall v_i \in V$ that has not been visited

for k-th neighbor nodes $v^k_{(x,j)}$, $k = \{0,1,2,\dots,n\}$

if $v^k_{(x,j)} \notin P_{cdt}$

obtain highest weight$|w_{(x,j)i}|$ from $v_{(x,j)}$ to v_i

if $e_{(x,j)i}$ contains no Prerequisite Information

if $EiVal(v_i) \geq \chi$

add v_i into P_{cdt}

if $w_{(x,j)i} > 0$

$$y = y - 1$$

let $v_{(y,j)} \leftarrow v_i$

$$dir \leftarrow backward$$

else

$$y = y + 1$$

let $v_{(y,j)} \leftarrow v_i$

$$dir \leftarrow forward$$

else

add v_i connected by $e_{(x,j)i}$

$$y = y - 1$$

$$v_{(y,j)} \leftarrow v_i$$

go to 2

3 : return $P_{cdt}[v_x][v_y]$

STEP 5 :

To avoid duplications (i.e., two same nodes appear in a same path), the mechanism named “Duplicated Nodes Removal” has been utilized to make cross-comparison to all the nodes within $P_{cdt}\,[v_x][v_y]$. If duplication exists, the node being added later will be removed directly, and the process will go back to the previous one to look for alternative nodes. If there is no existence of duplicated node, the produced candidate array $P_{cdt}{}'\,[v_x][v_y]$ will be converted to a list of single paths that will be the result responded to a user’s query. Then decisions can be made by users to make further modifications. The detail algorithm is shown in Table 4-12.

Table 4-12 Algorithm for STEP5 of Lecture Template Generation

Input:$P_{cdt}\,[v_x][v_y]$

Output:$P_{cdt}{}'\,[v_x][v_y]$

1: check for duplicated nodes v_d in same column of P_{cdt}

for each j with $j < |v_j|$

compare $v_{(y,j)}$ with rest nodes in $v_{(y,j+1)}$

if $v_{(y,j+1)}$is duplicated from $v_{(y,j)}$

remove $v_{(y,j+1)}$

go to STEP 4 (2)

2 : return $P_{cdt}{}'\,[v_x][v_y]$

4.4.2 Path Reorganization

With the basic structure of the lecture path, the sequence reorganization is then discussed to cope with the changes which may be made by end users. The changes simply identify the addendum and/or removal of the element(s), i.e. learning object, in the path. The sequence, especially the juncture, should be revised in accordance with the original attributes of connected objects. For instance, we assume that the sequence is shown as $v_1 \rightarrow v_2 \rightarrow v_3$. An object should be picked as an alternation if v_2 is removed. Otherwise, the operation causes the fracture between v_1 and v_3 and discontinues the path. The proposed metrics, betweenness centrality and closeness centrality, are applied to develop a relevance feedback algorithm to achieve the purpose which is shown in Table 4-13.

Table 4-13Algorithm for Lecture Template Reorganization

Input: User's Request $Req(\quad)$

Output: Revised Path $P_{rev}[v_x][v_y]$

1: check the status of input request

if $Req(\quad)$ = "*Addendum*" goto 2

else if $Req(\quad)$ = "*Removal*" goto 3

else return $P_{rev}[v_x][v_y]$

2: check the attribute of added node v_a

2.1 check $sim_{coe}\left(v_a, v_{y(y,j)}\right)$ between v_a and $v_{(y,j)}$

find appropriate position to add node v_a

$$P_{rev}[v_x][v_{(y,j+1)}] \leftarrow P_{rev}[v_x][v_{(y,j)}] + v_a$$

2.2 if $v_a \in V$add v_a to V

2.3 add correlation, $e_{a(j+1)}$ and $e_{a(j-1)}$, to nodes associated to v_a

3: check position of removal node $v_r$in $v_{(x,j)}$

for each *j* with $j < |v_j|$

if $v_r$equals to$v_{(y,j)}$

for nodes set V' connected to v_r

compute $CloseCent(v_z)$, where $v_z \in V'$

convert$N_c[v_z]$, where $N_c \in V'$

for each *p* with $p < |v_z|$

if $e_{(y,j)(z,p)}$ exists then $y_{(y,j)} \leftarrow v_{(z,p)}$

else$y_{(y,j)} \leftarrow v_{(z,p+1)}$

4: return $P_{rev}[v_x][v_y]$

The processes to path reorganizationis mainly categorized into two aspects. The first is to add nodes into a path, while the second is to remove nodes from the candidate path. In the adding process, we will firstly calculate the possible position of the nodes, which may be added, within the selected path through comparison of similarity between v_a and $v_{(y,j)}$. Then the node va will be directly added to the path, and the newly constructed correlations (i.e., $e_{a(j+1)}$, $e_{a(j-1)}$) will also be returned to LONET. Compared to the added process, the removal process will be more complicated. When the system receives the instruction to remove a specific node v_r, it will check which node (i.e., $v_{(y,j)}$) shall be removed. Then, the system will calculate the nodes (possible alternative nodes) that have a higher closeness centrality to $v_{(y,j)}$, and place them in a temporary set $V^{'}$. After that, the correlations associated to vr will be determined. If the correlation is not belonged to "Prerequisite," the corresponding node $v_{(y,j)}$ in the path will be removed directly. Otherwise, if correlation belongs to "Prerequisite," two of them will be removed together. And the alternative nodes will be selected again through the same process to reorganize the revised path.

An instance of lecture template generation is demonstrated through a query q. The cluster "Algorithms & Data Structure cluster" is utilized instead of using the whole LONET. Assuming that the entry node "1" is obtained through similarity calculation as show in Figure 4-4. Then, the algorithm looks for the neighbor nodes associated to node "1." Through this figure, it is obvious that node "2" has the highest OutDegree through the number of outgoing links to node "1." Thus, the direction, compared to node "1", is headed to the left. Then, we go further to check the EigenValue and the correlations between following candidate nodes in

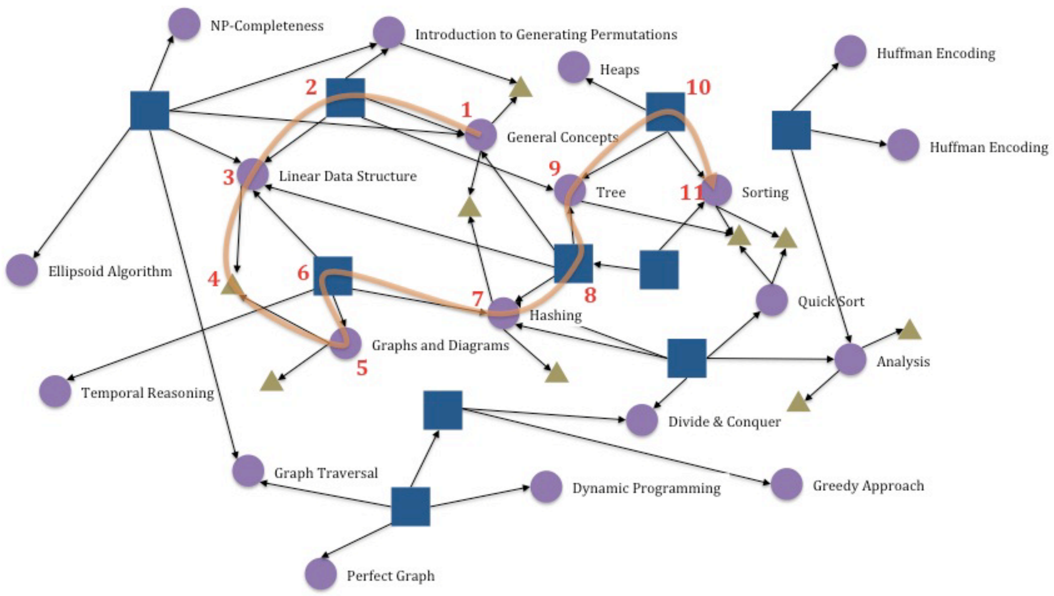


Figure 4-4An example oflecture templategeneration with Query q

accordance with the processes addressed in previous sections. A preliminary path to generate lectureis constructed through the interactive iterative search process. Through the query, the generated path has 11 nodes, and contains three different kinds of attributes. That is, there are four subjective nodes (blue color), one reference node (gray color), and six common nodes (purple color).The common nodes (node with content) are concentrated, and thus, the lecture path corresponding to the input query should be "1→3→5→7→9→11". The generation process will be ended when the next obtained common nodes have no direct usage history.

4.5 Adaptive Route

Following the experience on lecture template generation, we go further to discuss adaptive application of the LONET.Differentiating from Lecture Template, the *Adaptive Route* is more like the obtainment of optimal heading direction(s) based on the selected learning object. That is, we concentrate on turning the suggestions (Chapter 4.3) into actual route(s) containing only the selected learning object and its follow-up object in first level (i.e., direct link) as a further support to individual's search process.

The considered scenario is illustrated in Figure 4-5. According to Chapter 4.4, the learning objects are connected based on original usage experiences. However, not all of the connected objects are applicable to be returned to the users. In this case, the distance between the original query which leads to the selected learning object (red circle in this example) and the connected learning objects (black circles) are concentrated. It is because that the learning objects with similar attributes are clustered in same area (i.e., a conceptual group).

In Figure 4-5, three learning objects (learning object 1, learning object 2, and learning object 3) and corresponding connections are obtained according to existing correlations. Once the distance of original query is determined, the connections would be modified to satisfy the tolerable moving distance. And thus, two connections that do not pass the examination are removed.

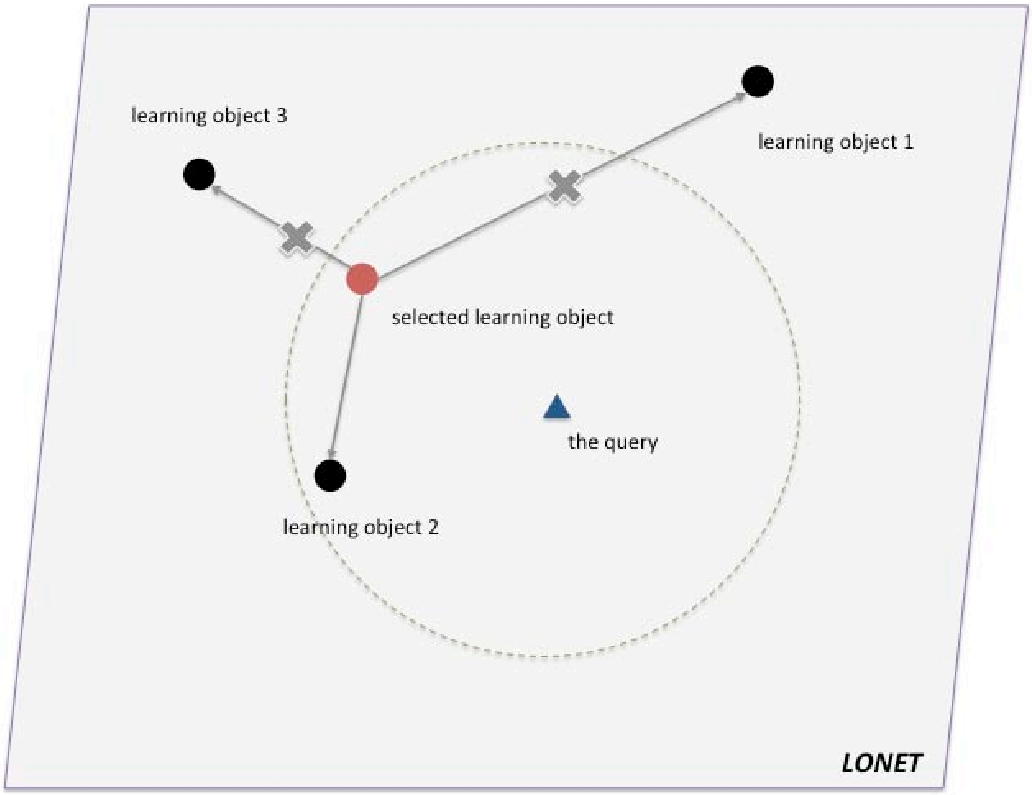


Figure 4-5 Illustration of Adaptive Route Generation

The algorithm to achieve adaptive route generation and optimization is shown in Table 4-14.The algorithm represents the procedure of adaptive route generation from the existing learning objects in LONET starting from a selected learning objectv_i to a set of connected objects (C_v). Note that the concentration is to obtain the optimal object(s) that has direct connection with v_i. Seven steps are involved. In Step 1, the resource v_j in V which has direct connection with v_i is found based on existing correlationsdefined in previous chapter. In the beginning, four categories of correlations are utilized. The correlation(s) between each connected resource v_j is then determined in Step 2. The *Bag* is adopted instead the *Set* since it is possible for connected resources to contain more than one correlation. After that, the correlations in E_i are sorted by the priority of correlation

(Prerequisite>Inheritance>Reference = Peer). The priority is determined in accordance with the number of

Table 4-14 Algorithm for connected resources retrieval

Input: A selected resource v_i

Output: One or more connected resources C_v

Step 1: Find resource $v_j \in V$ connected directly to v_i.

Step 2: Find entry resource Set $G_i \subset G \bullet \forall e_{ij}, j \in \mathbb{N}$. Let E_i denotes the *Edge Bag* and make $E_i \leftarrow E_i \cup e_{ij}$.

Step 3: For $\forall\, v_j \in G_i$, find connected resources, $v_k \in V$. Let $G_i \leftarrow G_i \cup v_k$, and $E_i \leftarrow E_i \cup e_{jk}$.

Step 4: Use edges and its containing correlations in E_i to generate the continuous route S_v starting from v_i.

Step 5: Check the location information of given query$q_{distance} = (q_{d_x}, q_{d_y})$ compute the distance, d, between user and v_i.

Step 6: Take d as the radium to generate a tolerant range Θ and filter out those resources v_j connected with v_i that are not within the range.

Step 7: Return C_v according to the given resource v_i, and display the first-level connection

learning objects the correlation could lead to. For example, the Prerequisite belongs to mandatory correlation, and that is, two learning objects, at least, are obtained once this correlation existed. For the rests, they are optional and are applicable to be obtained separately. In Step 3 and Step 4, the sub-graph G_i is generated to record the candidate resources until the similarity between default attributes of compared resources reach large disparity. The algorithm that computes the similarity coefficient between learning objects is shown in Table 4-3.

In Step 5, the learning objects in LONET are converted into a coordinate system to obtain the distance. The information of query $q_{distance}$ is also represented by a two-dimensional system $\left(q_{d\_x}, q_{d\_y}\right)$. With the information, the distance d between selected learning object and the query is determined. We then compute the appropriate distance d to be utilized to obtain the route from v_i. The method to compute the distance is applied by Eq. (6).

$$d = 2r \cdot \sin^{-1} \Delta \tag{4.5}$$

where

$$\Delta = \sin^2 \theta + \cos q_{d\_x}^{A} \cos q_{d\_x}^{B} \sin^2 \theta'$$

and

$$\theta = \frac{q_{d\_x}^{B} - q_{d\_x}^{A}}{2}, \theta' = \frac{q_{d\_y}^{B} - q_{d\_y}^{A}}{2}$$

The variable r is the fixed length as known as the radium of the earth. The obtained distance d is then considered as new radium to draw a great circle with the resource v_i. The area of the drawn circle is considered as the tolerant range for user to reach. After that, the resources in G_i are compared with v_i by Eq. (4.5) to filter out those resources connected but not within the range. The rest resources will be included in the set C_v to be returned.

CHAPTER V. EXPERIMENTS & EVALUATION

The empirical study is developed to demonstrate the feasibility of proposed works. First, the target dataset is introduced.The metrics in information retrieval are applied to evaluate the performance of implemented search service.The performance of path generation and the weighting/ranking algorithm is then discussed. In the last, the experiments with the focus on usability testing are conducted with around 50 users.

5.1 The Data Set

Our repository system, the MINE Registry, has stored and shared around 23,000 learning objects which are mostly collected from Open University, United Kingdom, and Advanced Distributed Learning Initiative, United States, in the past five years. Most of the learning objects are meant for specific educational trainings.

The learning objects in the MINE Registry are categorized into four major clusters including Photoshop, Algorithms & Data Structure, Computer Science, and others. Specifically, each of the categoriescontains6,000, 9,000, 5,000, and 2,500learning objects, respectively,which are the nodes that construct the LONET. In addition, the learning objects in our repository have been applied for generating the lectures (set) and/or becoming the supplementary materials (only single object is adopted)included in other course packages.

5.2 Overall Performance Evaluation

In this study, we concentrate on the efficiency and accuracy of the automated algorithms which are implemented in our repository. Two experiments are conducted. In the beginning, the TREC [Buckley and Voorhees 2004] was adopted to evaluate the search performance stored in the MINE Registry without further data process. The metrics [Jarvelin and Kekalainen 2000] for estimating the performance are shown as follows, where P', Eq. (5.1), represents the precision value and the R', Eq. (5.2), indicates the recall value. The $nIAP$, named non-Interpolated Precision, is used to calculate the precision of retrieved learning objectin a ranked result. In $nIAP$, the variable r_j is the sum of queries sent to system, the $v_j(i)$ is the order of retrieved learning object in the list, and the i is the order of query. To perform $nIAP$, the cut-off, number of documents in a result list, needs to be concerned. In this research, it was set to 10 by default. The results can be found in Table VIII.

$$P' = \frac{number\ of\ relevant\ learning\ objects\ retrieved}{number\ of\ learning\ objects\ retrieved} \tag{5.1}$$

$$R' = \frac{number\ of\ relevant\ learning\ objects\ retrieved}{number\ of\ relevant\ learning\ objects\ in\ dataset} \tag{5.2}$$

$$nIAP = \frac{\sum \frac{i}{v_j(i)}}{r_j} \tag{5.3}$$

Table 5-1Experiment Results from TREC Evaluation

object Retrieved	Precision	Relevant Retrieved	Raw Data		Non-Interpolated Data	
			Recall	Precision	Recall	Precision
100	1.0000	100	0.1190	1.0000	0.0	1.0000
200	0.9200	184	0.2143	0.9200	0.1	1.0000
400	0.9475	379	0.3333	0.9475	0.2	1.0000
600	0.8317	499	0.3810	0.8317	0.3	0.9200
2000	0.7675	1535	0.5476	0.7675	0.4	0.8095
4000	0.3705	1482	0.8810	0.3705	0.5	0.7857
10000	0.2001	2001	0.9286	0.2001	0.6	0.7568
20000	0.0500	1000	0.9524	0.0500	0.7	0.6977
					0.8	0.4789
					0.9	0.3585
					1.0	0.0000

In this evaluation, two different kinds of data are examined. The Raw Data represents the evaluation results for those learning objects without any pre-process in our repository. The Non-Interpolated Data then represents for the evaluation results that should be ranked in specific order. To clarify the results, the Precision-Recall curve for both Raw Data and Non-Interpolated Data are illustrated in Figure 5-1 (a) and Figure 5-1 (b).

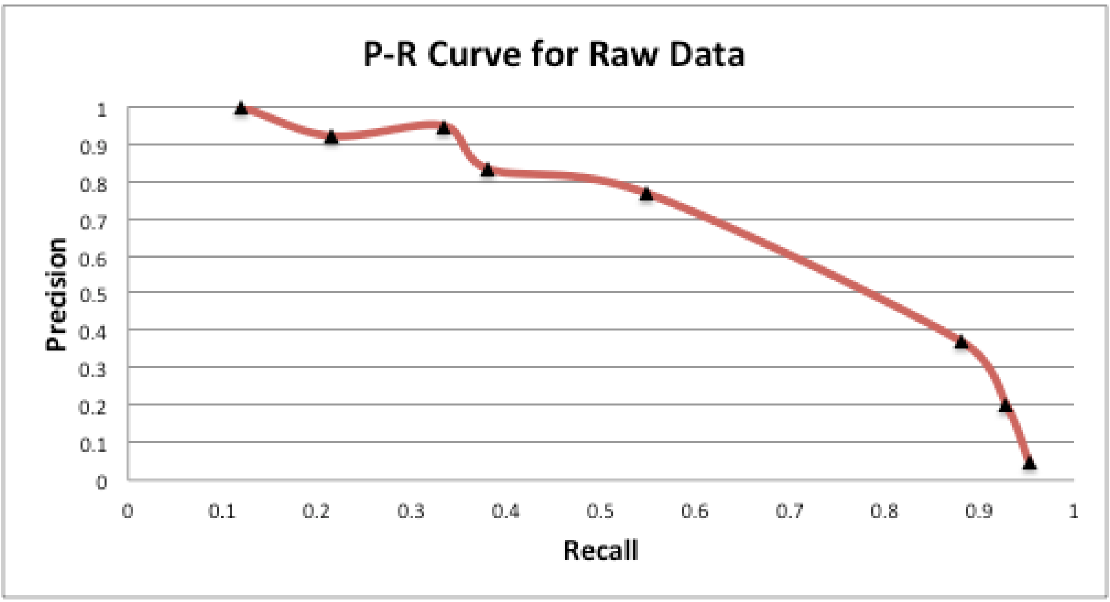


Figure 5-1 (a)P-R Curve for Performance of Search Service in MINE Registry

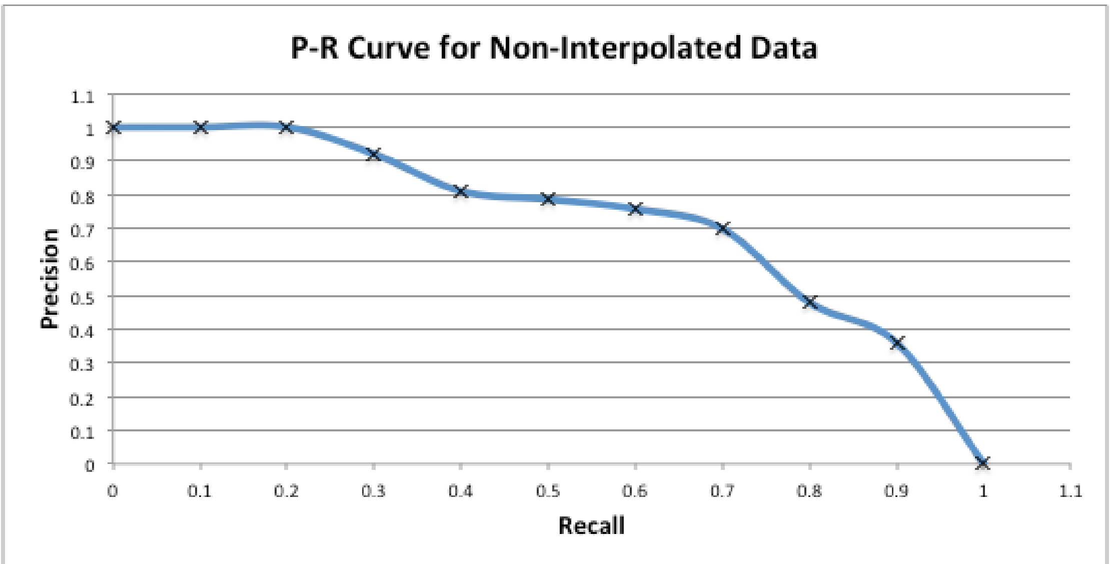


Figure 5-1 (b) P-R Curve for Performance of Search Service in MINE Registry

In addition, to enhance the searchability for specific categories in the MINE Registry, the k-nearest method was adopted to make classification of the learning objects in accordance with the corresponding metadata. As stated, three classifications, Photoshop, Algorithms & Data Structure, and Computer Science, were used in Precision-Recall evaluation.

In this experiment, three queries, based on predefined search criteria (keyword as example), were used to perform the experiments. The first query (Q_1) is to obtain related learning objects from major type “Photoshop” with a query term “photoshop.” The second query (Q_2) is for the major type “Algorithms & Data Structure” with term “algorithm + intro”. The third query (Q_3) is with term “cs + agenda” to query the major type “Computer Science”. The corresponding results to these queries are shown in Table 5-2.

According to the results, it is worth mentioning that the precision and recall value reach, in average, 90% and 94%, respectively, which reveal that our search mechanism can assist users in obtaining related learning objects in an efficient way. And this result also shows that the entry nodes, retrieved by our search service, have certain creditability for being the basis of generated lecture paths.

Table 5-2P-R Evaluation for learning objects with pre-classification

	Q_1	Q_2	Q_3
Total LOs in Repository		22141	
Relevant LOs in Repository	5962	8713	4948
Retrieved LOs	6235	9214	5123
Relevant LOs Retrieved	5602	8313	4601
Precision	89.85%	90.22%	89.81%
Recall	93.96%	95.41%	92.99%

5.3 Performance of Proposed Algorithms

The overall performance of the implemented service reaches the average as expected. We go further to examine the performance, specifically,on the path generation and the weighting/ranking algorithm.

5.3.1 Performance ofPath Generation

The accuracy of the retrieved nodes (or learning objects) for lecture path generation is concentrated.The nodes are compared with the ones in candidate dataset. This relation is used to generate a Precision-Recall curve, with two P-R curves as the baseline in previous experiment, to compare with the interactive search algorithm that achieve lecture path generation. In Figure 5-2, the curve with circle dots on it is the result while using interactive search algorithm. The linear dashed line goes across these three curves is the average trend. It is obvious that there is no prominent portions in this curve through comparison with the baselines. But, by and large, the proposed search method can reach the average accuracy.

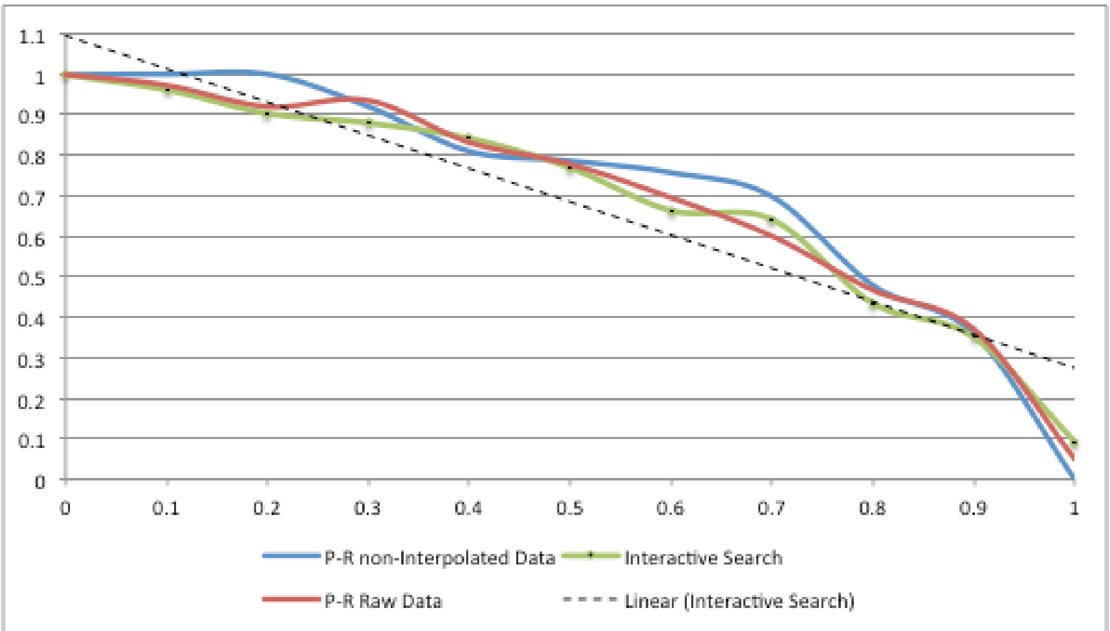


Figure 5-2 Precision-Recall Curve of Selected Nodes in Lecture Path

5.3.2 Performance of Weighting/Ranking Algorithms

We also conducted an experiment to compare the proposed weighting/ranking algorithms. Researches with same focus are compared with our approach as show in Table 5-3.We utilized the evaluation results obtained in the previous section. The precision-recall graph is shown in Figure 5-3. It is worth mentioning that the results between different approaches are close, and our approach is, at some degrees, higher than others.In addition, this evaluation reveals that the use of metadata as the basis of searching criteria, in some situations, will lead to a better result than the ontological approach which describes the learning object and provides search criteria in a non-united way.

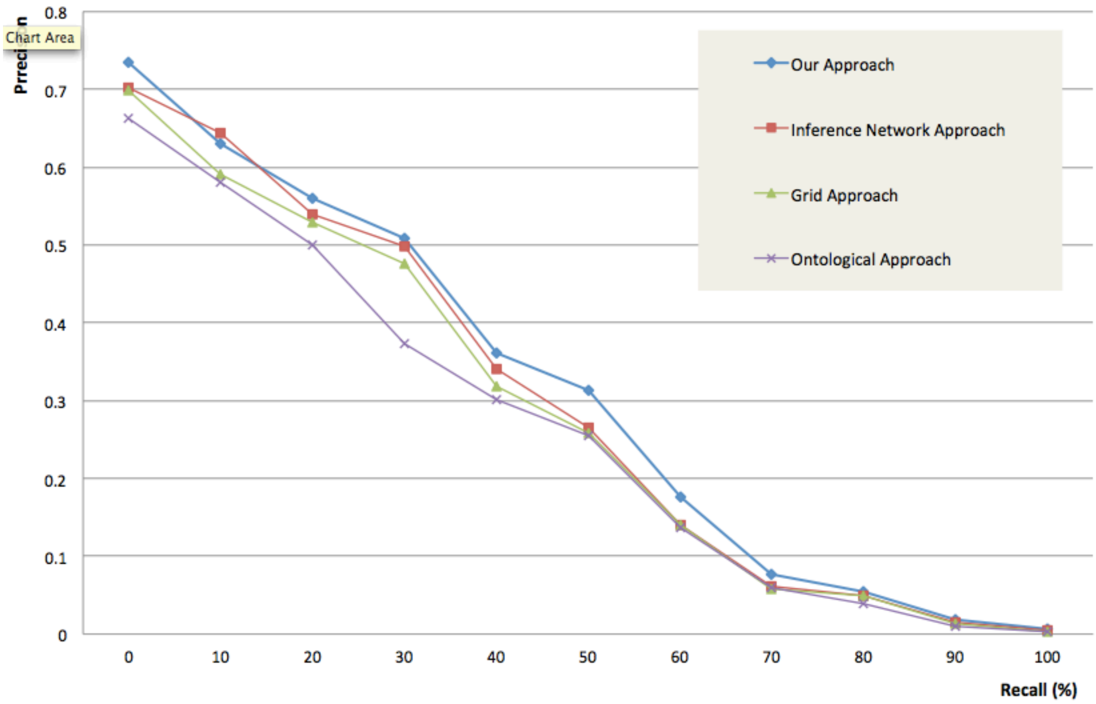


Figure 5-3 P-R performance compared with other related approaches

Table 5-3 Comparison with related researches

Topics / References	Overall Strategy	Our Approach
Search Suggestion [Catteau et al. 2008] [Khierbek et al. 2008]	The goal is to find out the relationship among learning objects and create the connections based on the elements defined in LOM. However, the search criteria are limited in keywords. In addition, the use of "Annotation" can only represent the usage situations of authors, instead of end-users.	The query criteria are categorized into six and the usage experiences are recorded. Besides, based on the weighting result, the search guidance providesprogressive suggestions to the adaptive input.
Query Methods [Hilera et al. 2009] [Shih et al. 2006]	The concept of middleware is proposed to search for learning objects in federated repositories based on LOM. But, due to the large amounts of elements, the performance needs to be improved. In addition, no common standard is applied and make resources isolated.	We follow the CORDRA to develop the repository and share the resources (i.e., learning objects) with external branches through a central registry system.Nine significant (or commonly adopted) elements are selected to increase the specific use.
Recommendation [Tsai et al. 2006] [Ochoa and Duval 2006]	The contextual and ontological information are applied to be the evaluation methods. However, the significance of learning objectsshould be revised concerning the time-series information. In addition, the relations between learning objects shall also be considered when ranking process.	Three major factors are utilized to evaluate thelearning objects based on different time-series information.Learning objects are ranked based on the derivation relations recorded in Reusability Tree. The relations between search results are also considered.

5.4 The Usability Testing

Experiments on usability testing are conducted to receive concrete feedbacks from users. In the experiment of interactive search algorithm (i.e., lecture path generation), we focus on implicit feedbacks from users. We collect the usage data from 25 instructors containing 10 lecturers and 15 teacher assistants. Three major factors were collected while accessing the search service. The first one is the "number of visit per page" to the item listed in the result list. In our search engine, the returned items within a result page are 10. The second one is the "number of selection" to the returned items. The third one is the "number of query revision" that an instructor will send to obtain an expected item. The comparison between difference search paradigms, basic search and interactive search, is shown in Table 5-4.

The result reveals that the interactive search can reduce the number of query revision at least 1.08, at average (i.e., (5.71 – 11.09) / 5), comparing with the basic search that follows the general core of search engine. In addition, the interactive search can raise the actual usage of returned items within the result list because the average selection is up to 0.2 (i.e., (11.19 – 10.19) / 5) in average. It is applicable to be found that preferable number of keywords, in this case, is ranging from 2 to 3.

Table 5-4Comparison of feedbacks between difference search paradigms

(Unit: Item)

	Basic Search			Interactive Search		
No. of Terms	Visit / Page	Selection	Revision	Visit / Page	Selection	Revision
1	5.07	3.16	2.12	7.18	3.60	2.06
2	5.66	3.32	3.67	6.33	4.13	1.54
3	3.53	1.85	3.39	4.11	2.33	1.37
4	2.24	1.10	1.43	1.97	0.79	0.47
5	1.51	0.76	0.48	1.01	0.34	0.27

Table 5-5 Feedbacks of concrete usage of generated lecture template

Initial Length	Final Length	Number of Revision	
		Add	Remove
3	4.47	3.13	0.19
4	5.13	2.79	0.54
5	5.46	1.68	0.77
6	5.57	0.46	1.32
7	5.96	0.13	2.21

The number of nodes in a lecture path is also recorded. The selected items in the result list are utilized to generate the initial path, and collected the revision scenario of the generated paths. The length of a lecture path is, by default, set to be between 3 and 7. In this experiment, 200 paths, equal to 200 selected LOs, were utilized to reach the results. There are 40 in length 3, 50 in length 4, 35 in length 5, 35 in length 6, and 40 in length 7. We can observe that the preferable length of generated path is around 6, by rounding off the numbers in this experiment. The detail results can be found in Table 5-5.

As to the search guidance, a pre-post experiment on 40 users, including 6 professors and 34 teacher assistants, who have accessed our repository to reuse learning objects. In order to obtain the objective results, users are asked to search for a specific learning object which is the only one object entitled "Introduction to Photoshop" in the dataset. Note that the inheritance relations are removed from the mentioned objects to avoid the possibility of indirect obtainment in this situation. Two major factors, query frequency and cost of time, are recorded to compare with the assistance of search guidance (post-test) and without the search guidance (pre-test). First, two substantial assumptions are made before going into the experiment.First, we made two substantial assumptions before the experiment:

- **Assumption 1:**

  The cost of time to obtain the expected learning object in the post-test is not better thanthe one in the pretest.

- **Assumption 2:**

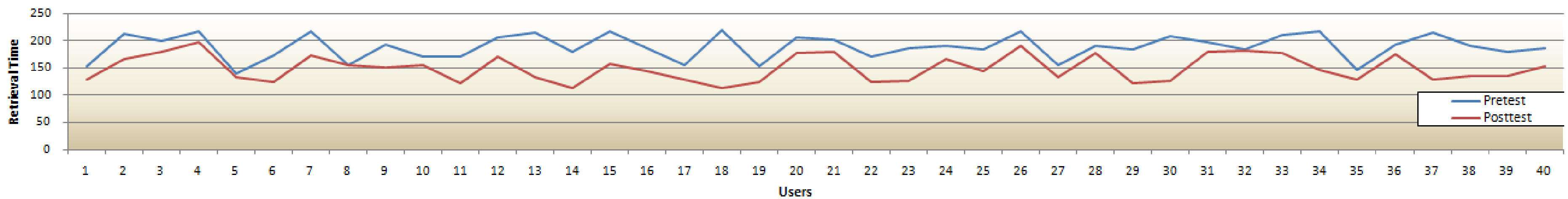
  The cost of time to obtain the expected learning object in the post-test is better thanthe one in the pretest.

To verify the assumptions, an One-Tailed Test (at $\alpha = 0.05$) is applied where the degree of freedom (df) is set to be 39 according to the examinee. The comparison of the pre-post test and the average cost of time are illustrated in Figure 5-4. The d-value - that is, the pretest score minus the posttest score of user identified as number 1 is 24 (i.e., 152-128). According to the t-distribution table, which assesses whether the means of two groups statistically differ from each other, $df = 39$ maps to t value 1.685. The obtained t value is at 10.098, which is greater than 1.685, through the formula. Thus, the assumption 2 is accepted and reveals that the users who utilize the search guidance would reduce the cost of time, around 40 seconds in this case, in looking for the specific learning object. The detail statistic resultsareshown in Table 5-6.

Table 5-6 Pre-Post Test for Search Guidance

	PreTest	PostTest
Mean	188.4	149.2
Variance	524.092	580.421
Observations	40	40
Pearson Correlation Coefficient	0.455	
Degree of Freedom	39	
t Stat.	10.098	
t Critical one-tail	1.685	
t Critical two-tail	2.023	

In addition, the query frequency that users have while searching for the target learning object is considered. Though the results do not remark a significance, the average query frequency still reach 3.5 per learning object (3.5 queries to reach the target learning object) where the most query frequency is at 7 and the least one is at 2.



Unit: Seconds

No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Pretest	152	213	199	216	139	172	217	155	192	171	171	205	215	180	216	185	156	219	153	205
Posttest	128	165	179	198	133	124	172	155	150	155	122	170	132	113	157	143	129	113	124	176
d	24	48	20	18	6	48	45	0	42	16	49	35	83	67	59	42	27	106	29	29

No.	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
Pretest	201	170	186	190	184	217	155	190	184	209	198	184	211	218	146	193	215	190	179	185
Posttest	180	125	127	165	143	191	133	178	121	126	180	182	177	147	129	175	128	136	134	153
d	21	45	59	25	41	26	22	12	63	83	18	2	34	71	17	18	87	54	45	32

Figure 5-4Results of Pre-Post test to the search guidance approach

CHAPTER VI. CONCLUSIONS

6.1 Summary of Thesis
6.2 The Limitations
6.3 Future Work

This study is concluded in three major portions. In the beginning, the brief review of the thesis is given. The challenging issues are then discussed. In the last, the possible future work is addressed.

6.1 Summary of Thesis

Issues regarding learning object management, sharing and reuse have been widely discussed in the literatures related to distance learning technologies. Although a simple approach to realize reusability relies on finding useful learning objects in a repository, the use scenario of the obtained learning objects could lead to another dimension of reusability.

In this thesis, an integrated approach considering the human-centric support through interdisciplinary perspectives is proposed to make enhancement to the international standards, such as SCORM and CORDRA, based on a systematic reexamination. The past achievement on learning object management (i.e., the Reusability Tree) is followed. We collect and analyze the user experiences on learning object reuse through our repository (or the MINE Registry) within the last five years. We then introduce the concept of LONET (Learning Object Network), an extension of the Reusability Tree, to clarify the correlations among learning objects and define a general graph model based on time-series information for further usage in Chapter III. The customized metrics based on social network analysis are proposed to mine the model through quantifying the correlations in the same chapter.

Based on the LONET, the practical applications that facilitate the reuse process of learning object are then proposed from the perspective of information retrieval in Chapter IV. The reference information, including citations and user feedbacks, are applied to develop a weighting/ranking algorithm with the focus of time-series information, and highlight the significance. The search guidance algorithm is proposed with the aim at leading users to appropriate direction(s) by providing progressive suggestions corresponding to the query. An interactive search algorithm, revised by Swarm Intelligence, is developed to mine the experiences in the LONET. It generates adaptive path(s), based on past usage experiences, by computing possible interactive input, such as search criteria and feedback from instructors, and generates the tentative template for lecture generation.

To demonstrate the feasibility, an empirical study is conducted by two system performance evaluations and a usability testing experiment. The evaluation is performed based on TREC testing with a general Precision-Recall test for retrieval accuracy and a nIAP test for ranking accuracy. The usability experiment is conducted with around 50 users (e.g., professors, lecturers and teacher assistants) in three universities, including Waseda University, Japan, Tamkang University, Taiwan, Athabasca University, Canada, and reaches an expected result.

We highly expect that the contributions of this studymay cause further impacts in the international community related to distance learning technology.

6.2 The Limitations

As stated, this study concentrates on the issues of learning object search and reuse based on a systematic reexamination. Although the study benefits from some useful standards such as IEEE LOM and/or efficient system architecture like CORDRA, some critical challenges remain unsolved.

- **Quantities of Learning Objects**

  Although there are huge amounts of learning objects in our repository, lots of them are derivations from same learning object(s). That is, learning objects contain similar attributes and may lead to inaccurate retrieval performance.

- **External Connections**

  Each branch existed in a repository federation defined by CORDRA should have, at least, one connection with others to facilitate the resources sharing. The searchability among a large scale of repositories may cause the difference and lead to low efficiency.

- **Concerning Time-series Information**

  The selected timescale determines the importance of learning objects. In this study, as a preliminary implementation, a three-year timescale is selected. The length of the timescale is required to be normalized to provide appropriate search results.

- **User Experiences**

  The collective intelligence depends on the rich user experiences. The more records to be mined the more accurate the outcomes will be. In this study, only limited prototype is demonstrated due to the limitation of user experiences.

6.3 Future Work

In the future, in addition to solving the above limitations, an enhanced mechanism, based on instructional objectives, for lecture path generation is considered in a progressive way. Although the educational theories tell us how lectures are designed to fit instructional purposes, a quantitative approach to measure effectiveness in the generation of learning activities is still difficult. We will try to give explicit description to LONET, and/or its sub-network, to achieve specific instructional purposes, and continuously provide solutions to the challenges and limitations mentioned above.

APPENDIX. THE IMPLEMENTATION

The MINE Registry, a branch of federated repositories under CORDRA architecture, has stored around 23,000 learning objects and corresponding usage experiences, especially those on lecture generation, in the past five years. In the appendix, a concrete example on learning object searching is given to demonstrate the basics of our repository and the implementation of proposed works in previous chapters.

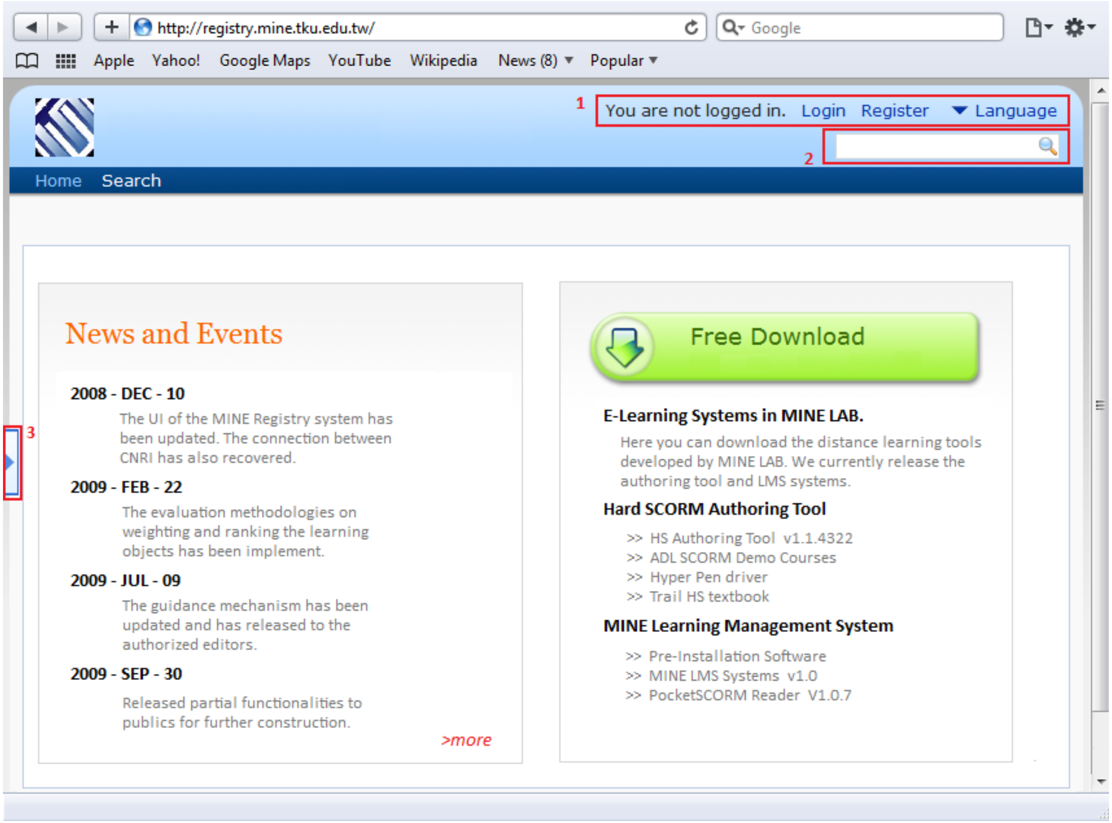


Figure A-1 The Interface of MINE Registry - 1

1. **Registration / Language Selection**

   The user has to login to the system for utilizing the advanced functionalities such as upload/download learning objects, personalized search history, and search path analysis, etc.

2. **Basic Search Toolbar**

   The user can utilize this toolbar to search learning objects only based on keywords.

3. **Side Toolbar**

   The user can click on this toolbar to open the sitemap or other related services developed by our research group.

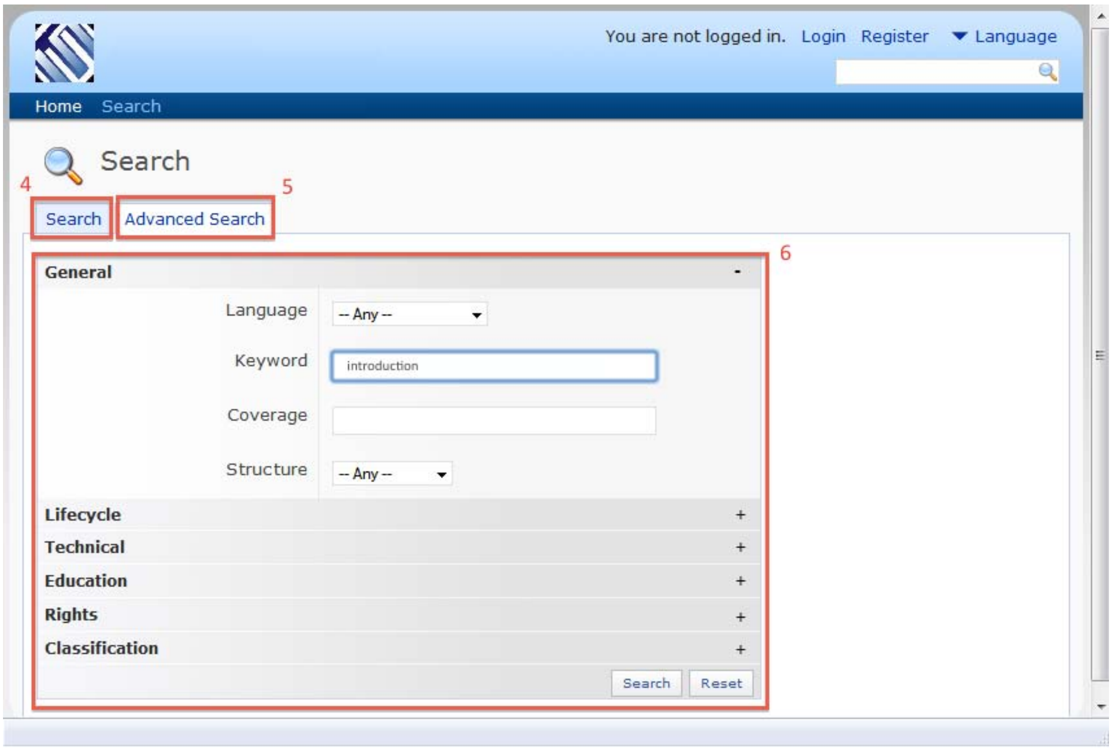


Figure A-2 The Interface of MINE Registry - 2

4. Search Service

This tab represents the basic search service implemented in the repository. In this example, the keyword “photoshop + intro + example” is

5. Advanced Search Service

This service allows users to search for learning objects based on IEEE LOM.

6. Query Criteria Selection

The users can choose the criteria and input corresponding values to search for learning objects in the repository.

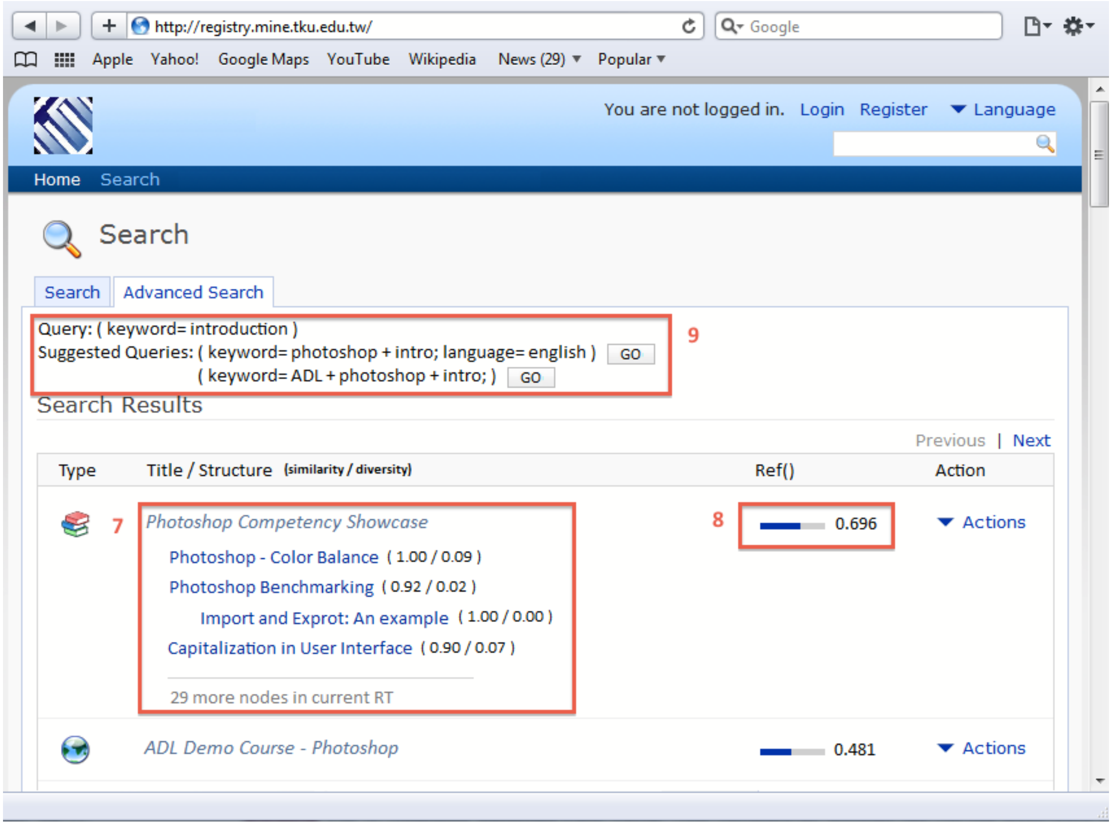


Figure A-3 The Interface of MINE Registry - 3

7. Reusability Tree

The reusability tree shows the relation between those learning objects. The values in the angle brackets represent the value of similarity and diversity.

8. Ranking (Weighting) Value of Learning Object

The ranking or weighting value is calculated and shown here, and affects the order of search results.

9. Search Guidance

The algorithm is generated the corresponding suggestion in accordance with the input query and the mining results of search results. In this example, two suggestions are given. User can follow the suggestion simply by clicking the “go” or to start over a new query.

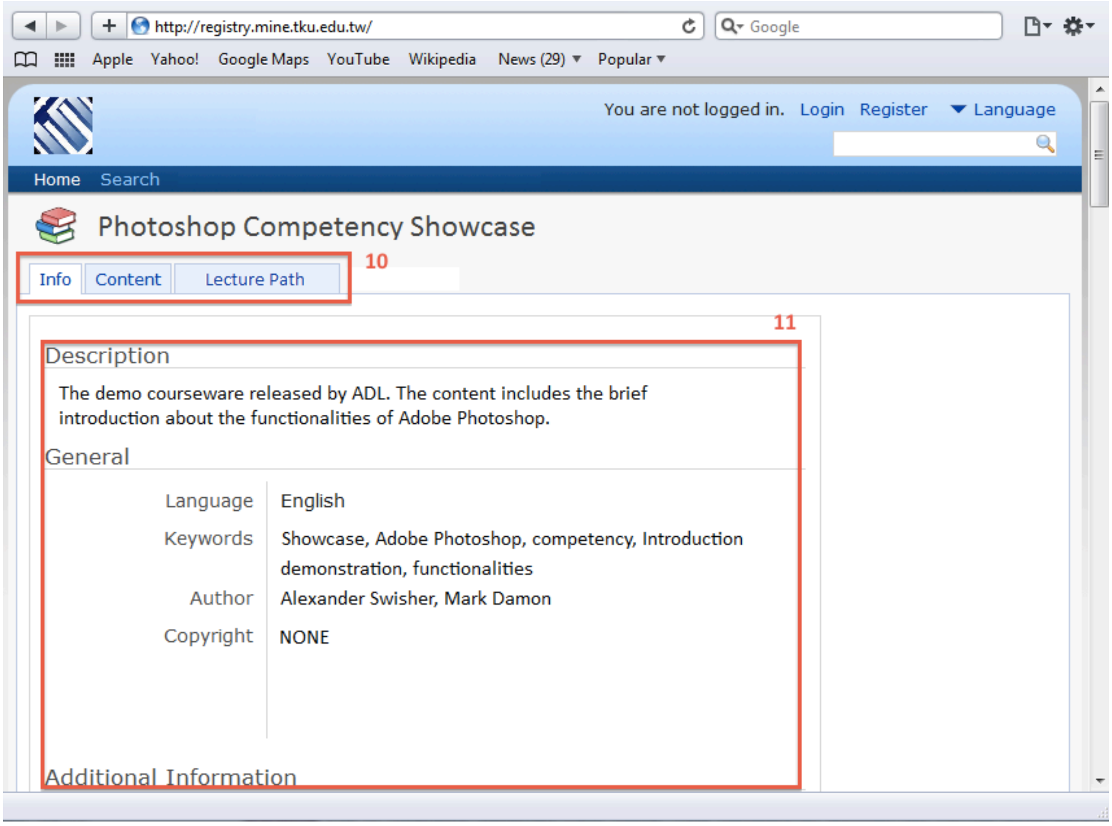


Figure A-4 The Interface of MINE Registry - 4

10. Information Toolbar

The information toolbar is represented as a tab menu including three major functions. The simple run-time environment (Content Tab) is also implemented through a web service powered by the learning management system.

11. LOM Metadata

The detail description of the selected learning object is shown in the middle of the page.

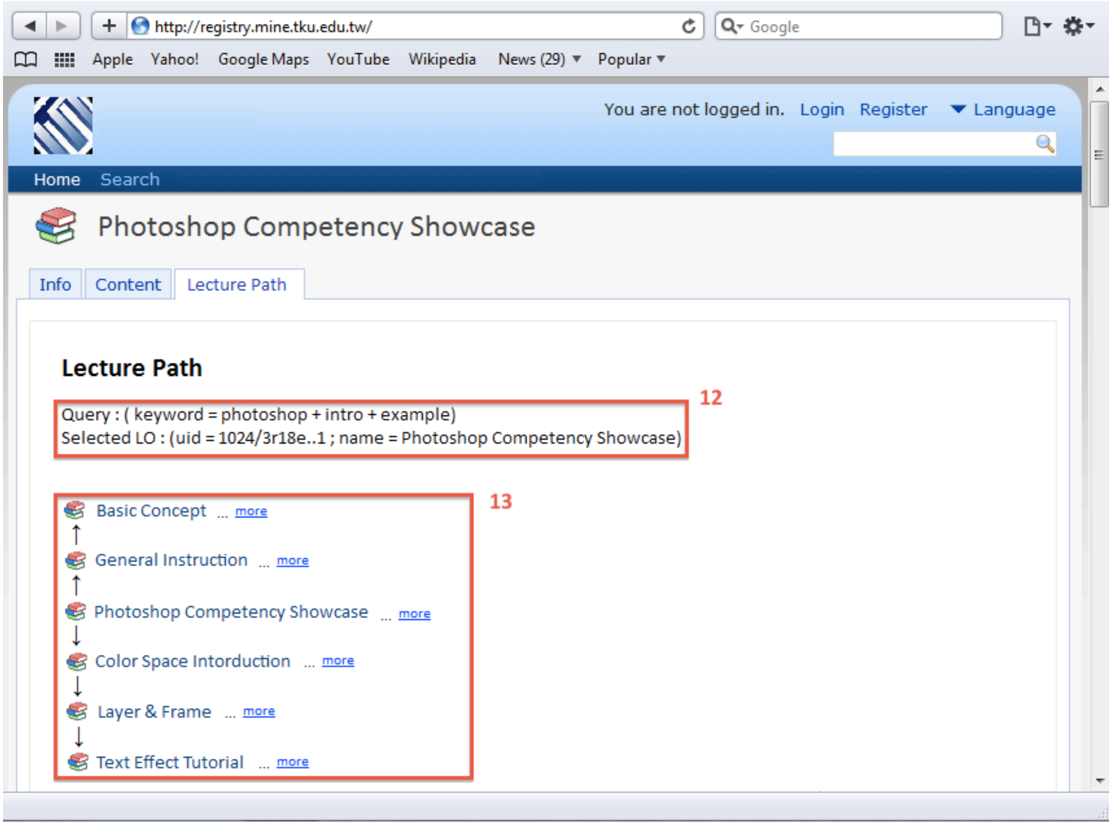


Figure A-5 The Interface of MINE Registry - 5

12. Information Details

The detail information regarding the input query and the selected learning object for lecture generation is shown. User is applicable to note down the information (e.g., the unique identification of learning object in “uid”) for future re-access.

13. Lecture Template

The lecture template is generated according to the selected learning object (Photoshop Competency Showcase in this example) and the user can make further modification through the “more” button at the end of each line.

ACKNOWLEDGEMENT

This thesis was supported by the RONPAKU Program of JSPS (Japan Society for the Promotion of Science), Japan, from April 2010 to March 2012.

The work was also partly supported by the National Science Council of Taiwan under Contract No. “NSC99-2511-S-468-004-,” “NSC100-2511-S-008 -010-MY2,” and “NSC100-2631-S-008-003-CC2,”the Institute of Information Industry of Taiwan, and 2010-2012 Waseda University Advanced Research Center for Human Sciences Project “Distributed Collaborative Knowledge Information Sharing Systems for Growing Campus.”

BIBLIOGRAPHY

Albert, R. and Barabashi, A.L. 1999. Emergence of Scaling in Random Networks. *Sciences*, 286, 509-512

Balabanovic, M. and Shoham, Y. 1997. Fab: content-based, collaborative recommendation. *Communication of the ACM*, 40, 3, 66-72

Bates and Peacock, 1989. Conceptualizing Social Structure: The misuse of classification in structural modeling. *American Sociological Review*, 54, pp. 565–577

Buckley, C. and Voorhees, E.M. 2004. Retrieval Evaluation with Incomplete Information. Proc. of *International Conference on Research and Development in Information Retrieval*, 25-32

Catteau, O., Vidal, P. and Broisin, J. 2008. Learning Object Virtualization Allowing for Learning Object Assessments and Suggestions for Use. Proc. of IEEE International Conference on Advanced Learning Technologies, 579-583

Chang, J.H.,and Lee, W.S. 2003. Finding Recent Frequent Itemsets Adaptively over Online Data Streams. Proc. of ACM *International Conference on Knowledge Discovery and Data Mining*, 487-492

Chen, Y., Dong, G., Han, J., Wah, B.W. and Wang, J. 2002. Multidimensional Regression Analysis of Time-Series Data Streams. Proc. of *International Conference on Very Large Data Bases*, 323-324

Fujimura, K., Inoue, T. and Sugisaki, M. 2005. The EigenRumor Algorithm for Ranking Blogs. Proc. of *World Wide Web Conf*erence

Freeman, L.C. 1979. Centrality in social networks: Conceptual clarification. *Social Networks*, 1, 3, 215-239

Gaševic, D., Jovanovic, J. and Devedžic, V. 2007. Ontology-based annotation of learning object content. *Interactive Learning Environments*, 15, 1, 1-26

Ghanea-Hercock R.A., Wang F. and Sun Y. 2006. Self-Organizing and Adaptive Peer-to-Peer Network. *IEEE Transactions on Systems, Man, and Cybernetics*, 36, 6,1230-1236

Hani Sarip, M., Yahya, Y. 2008.“LORIuMET: Learning Object Repositories Interoperability using Metadata,” Proc. of International Symposium on Information Technology, 1-5

Hasegawa, S., Kashihara, A., Toyoca, J. 2002. “An e-Learning Library on the Web,” Proc. of the International Conference on Computers in Education, 1281-1282

Haythornthwalte, C. 1996. Social network analysis: An approach and technique for the study of information exchange. *Library & Information Science Research*, 18, 4, 323-342

Herlocker, J., Konstan, J. and Riedl, J. 2002. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Information Retrieval*, 5, 4, 287-310

Hilera, J.R., Oton, S., Ortiz, A., De Marcos, L., Martinez, J.J., Gutierrez, J.A., Gutierrez, J.M. and Barchino, R. 2009. Evaluating Simple Query Interface Compliance in Public Repositories. Proc. of IEEE International Conference on Advanced Learning Technologies, 306-310

IEEE Draft Standard for Learning Object Metadata, IEEE P1484.12.1/d6.4, 2002

Jarvelin, K. and Kekalainen, J. 2000. IR Evaluation Methods for Retrieving Highly Relevant Documents. Proc. of *ACM International Conference on Information Retrieval*, 41-48

Jensen, D. and Neville, J. 2002. Data mining in social networks. Proc. of *National Academy of Sciences Symposium on Dynamic Social Network Analysis*, 290-301

Langville, A.N. and Meyer, C.D. 2006. Google's PageRank and Beyond: The Science of Search Engine Rankings. *Princeton Univ. Press*.

Lin, S. and Chalupsky, H. 2008. Discovering and Explaining Abnormal Nodes in Semantic Graphs. *IEEE Transactions on Knowledge and Data Engineering*, 20, 8, 1039-1052

Lin, C.L. and Kao, H.Y. 2010. Blog Popularity Mining Using Social Interconnection Analysis. *IEEE Internet Computing*, 14, 4, 41-49

Lin, F.H., Shih, T.K. and Kim, W. 2009. An Implementation of the CORDRA Architecture Enhanced for Systematic Reuse of Learning Objects. *IEEE Transactions on Knowledge and Data Engineering*, 21, 6, 925-938

Lin, Q., Agrawal, D., El Abbadi, A. 2003. Supporting sliding window queries for continuous data streams. Proc. of *International Conference on Scientific & Statistical Database Management*, 85-94

Liu, L., Antonopoulos, N. and Mackin, S. 2007. Social peer-to-peer for resource discovery. Proc. of *Int'l Conf. on Parallel, Distributed and Network-based Processing*, 459-466

Liu, H., Abraham, A. and Badr, Y. 2009. Neighbor Selection in Peer-to-Peer Overlay Networks: A Swarm Intelligence Approach. *Computer Communications and Networks*, 4, 405-431

Loser, A., Staab, S. and Tempich, C. 2007. Semantic Social Overlay Networks. *IEEE Journal on Selected Areas in Communications*, 25, 1, 5-14

Joseph, S. 2002. NeuroGrid: Semantically Routing Queries in Peer-to-Peer Networks. Proc. of *Int'l Workshop on Peer-to-Peer Computing*, 202-214

Kastner, M. and Furtmüller, G. 2007. Operationalization of the Metadata Element "Difficulty". Proc. of IEEE International Conference on Advanced Learning Technologies, 608-612

Khierbek, A., Salloum, S. and Tannous, A. 2008. An Inference Network Model for Retrieving Reusable Learning Objects. Proc. of International Conference on Information & Communication Technologies, 1-5

Kleinberg, J. 1999. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46, 5, 604-632

Kleinberg, J. 2000. Navigation in a small world. *Nature*, 406, 845.

Kleinberg, J. 2001. Small-world phenomena and the dynamics of information. *Advances in Neural Information Processing Systems*, 14, 1-14.

Koo, S.G., Kannan, K. and Lee, C.S. 2006. A Genetic-algorithm-based Neighbor-selection Strategy for Hybrid Peer-to-Peer Networks. *Future Generation Computer Systems*, 22, 732-741

Koppi, T., Bogle, L. and Bogle, M. 2005. Learning Objects, Repositories, Sharing and Reusability. *Open Learning: The Journal of Open and Distance Learning*, 20, 1, 83-91

Kumar, R., Ragbavan, P., Rajagopalan, S., and Tomkins, A. 2002. The Web and social networks. *IEEE Computer*, 35, 11, 32-36

Muzlo, J.A., Helns T. and Mundell, R. 2001. Experiences with reusable E-learning objects: From theory to practice. *The Internet and Higher Education*, 5, 1, 21-34

Moody, J., McFarland, D. and Bender-DeMoll, S. 2005. Dynamic Network Visualization. *American Journal of Sociology*, 110, 4, 1206-1241

Ochoa, X. and Duval, E. 2006. Use of Contextualized Attention Metadata for Ranking and Recommending Learning Objects. ACM International Conference on Information and Knowledge Management, 9-16

Ochoa, X. and Duval, E. 2008. Relevance Ranking Metrics for Learning Objects. *IEEE Transactions on Learning Technologies*, 1, 1, 34-48

O'Reilly, T. 2007. What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software. *International Journal of Digital Economics*, 65, 17-37.

Palla, G., Barabasi, A.L. and Vicsek, T. 2007. Quantifying Social Group Evolution. *Nature*, 446, 664-667

Perng, C.S., Wang, H., Zhang, S.R., Stott Parker, D. 2000. Landmarks: A New Model for Similarity-Based Pattern Querying in Time Series. Proc. of *International Conference on Data Engineering*, 33-42

Raghavan, P. 2002. Social networks: from the Web to the enterprise. *IEEE Internet Computing*, 6, 1, 91-94

Rivera, G.M., Simon, B., Quemada, J. and Salvachua, J. 2004. Improving LOM-Based Interoperability of Learning Repositories. *On The Move to Meaningful Internet Ssystems*, 690-699

Rocchio, J. 1971. Relevance Feedback Information Retrieval. The Smart Retrieval System - Experiments in Automatic Document Processing, 313 – 323

Rowstron, A. and Druschel, P. 2001. Pastry: Scalable, Distributed Object Location and Routing for Large-scale Peer-to-Peer Systems. Proc. of *IFIP/ACM Int'l Conf. on Distributed Systems Platforms*, 329-350

Roya, M., Chang, R., Qi, X. 2007. Learning From Relevance Feedback Sessions Using A K-Nearest-Neighbor-Based Semantic Repository. Proc. of*IEEE International Conference on Multimedia and Expo*, 1994-1997

Sánchez Alonso, S. 2009. "Enhancing Availability of Learning Resources on Organic Agriculture and Agroecology. The Electronic Library, 27, 5, 792-813

Sarwar, B., Karypis, G., Konstan, J. and Riedl, J. 2001. Item-Based Collaborative Filtering Recommendation Algorithms. Proc. of *World Wide Web Conference*, 285-295

Scott, J. 1991. Social Network Analysis: A Handbook. Sage Publication

Shih, T.K., Chang, C.C., Lin, H.W. 2006."Reusability on Learning Object Repository," Proc. of the 5th International Conference on Web-based Learning, 203-214

Shih, W.C., Yang, C.T., Chen, P.I. and Tseng, S.S. 2006. Performance-Based Content Retrieval for Learning Object Repositories on Grid Environments. Proc. of International Conference on Parallel & Distributed Computing, Applications & Technologies, 515-520

Stanley, M. 1967. The Small World Problem. *Psychology Today*, 1, 1, 60-67

Tolsdorf, C.C. 1976. Social Networks, Support, and Coping: An Exploratory Study. Family Process,15, 407-417

Tsai, K.H., Chiu, T.K., Lee, M.C. and Wang, T.I. 2006. A Learning Objects Recommendation Model based on the Preference and Ontological Approaches. Proc. of International Conference on Advanced Learning Technologies, 36-40

Ternier, S., Verbert, K., Vandeputte, B., Klerkx, J., Duval, E., Ordoez, V., Ochoa, X. 2009. “The Ariadne Infrastructure for Managing and Storing Metadata,” IEEE Internet Computing, 13, 4, 18-25

Umyarov, A., Tuzhilin, A. 2008. Improving Collaborative Filtering Recommendations Using External Data. Proc. of *IEEE International Conference on Data Mining*, 618-627

Van Assche, F., Campbell, L.M., Rifon, L.A. and Willem, M. 2003. Semantic Interoperability: Use of Vocabularies with Learning Object Metadata. Proc. of *IEEE International Conference of Advanced Learning Technologies*, 511-514

Vassiliadis, N., Kokoras, F., Vlahavas, I. and Sampson, D. 2003. An Intelligent Educational Metadata Repository. Databases and Learning Systems, CRC Press

Wasserman, S. and Faust, K. 1994. Social Network Analysis: Methods and Applications. Cambridge University Press

Wegener, B. 1991. Job mobility and social ties: Social resources, prior job, and status attainment. *American Sociological Review*, 56, 60–71.

Weng, C.Y., Chu, W.T., and Wu, J.L. 2009. RoleNet: Movie Analysis from the Perspective of Social Netowrks. *IEEE Transactions on Multimedia*, 11, 2, 256-271

Wellman, B. and Berkowitz, S.D. eds. 1988. Social Structures: A Network Approach. Cambridge: Cambridge University Press

Wolf, T., Schroter, A., Damian, D., Panjer, L.D. and Nguyen, T.H.D. 2009. Mining Task-Based Social Networks to Explore Collaboration in Software Teams. *IEEE Software*, 26, 1, 58-66

Wu, H.C., Luk, R.W.P., Wong, K.F. and Kwok, K.L. 2008. Interpreting TF-IDF Term Weights as Making Relevance Decisions. *ACM Transactions on Information Systems*, 26, 3, 13-49

Yang, T.D., Chiu, C.H., Tsai, C.Y. and Wu, T.H. 2004. Visualized online simple sequencing authoring tool for SCORM-compliant content package. Proc. of *IEEE International Conference on Advanced Learning Technologies*, 609-613

Yang, Y., Yoo, S.J., Lin, F. and Moon, I.C. 2010. Personalized Email Prioritization Based on Content and Social Network Analysis. *IEEE Intelligent Systems*, 25, 4, 12-18