International Journal of Grid Distribution Computing
Vol.8, No.2 (2015), pp.259-272
http://dx.doi.org/10.14257/ijgdc.2015.8.2.23

Research of Intelligent Web Courseware Model System Based on Ontology

Haixu Xi, WeiGao and JinQian

*School of Computer Engineering
Jiangsu University of Technology
Changzhou, China
just_xhx@126.com, 569964130@qq.com, qj@jsut.edu.cn*

Abstract

The development of distance education put forward higher requirements for WEB courseware, it should update teaching content based on the degree of students. This article proposes a resource sharing model of courseware based on semantic web so as to solve the problem of courseware in the teaching process. It takes Resource sharing based on SCORM courses object model as a framework and use XML to describe each learning content objects which makes teachers organize course content and create courseware more easy as well as makes excellent courseware resources can be shared in greatest degree. System defines two agents, one is the teacher interface agent for receiving input information from teachers and curriculum recommendation agent and another agent is communicate agent. Course recommendation agents mainly complete the selection of appropriate curriculum content recommended according to teachers in advance and adjust the course object order in accordance with the repository attribute settings.

Keywords: *Distance education, XML, intelligent Agent, semantic web, ontology*

1. Introduction

CAI is a kind of computer assisted instruction in teaching methods, it is computer-mediated by the interaction between the computer and the students achieve educational purposes. CAI can be used instead of teachers teaching the entire course, also be used as an additional means of conventional classroom teaching [1]. CAI and the traditional teacher-centered teaching compared to one-way communication, the most significant difference is that it's interactive, self-adaptability and integration [2]. Its interaction between teaching students and become a two-way communication between computers, students can control their own participate in teaching activities, thereby greatly enhancing interest in learning. Its integration to enable students to take advantage of a variety of instructional media for learning, and make teaching several processes (lectures, lectures, self-test and examination).

Courseware is clear teaching objectives, teaching content reflects textbook structure and have the appropriate teaching strategies and process systems [3]. Courseware for teaching activities, the use of computer language, writing system, or other writing tools generated by the computer software and the corresponding documentation, including educational programs for control and a computer program to help develop and maintain procedures for documentation and software with the use of textbooks and exercise books and so on [4]. Disadvantages of traditional courseware: First, CAI courseware poor portability. Teaching is integrated into teachers' teaching methods and teaching ideas, to show the level of teaching an art, not a technology. CAI courseware can be kind of dooring which this one teacher teaching ideas expressed, but this is only his own teaching ideas and ourselves. Other teachers make it difficult to be applied.

ISSN: 2005-4262 IJGDC

Teachers and students need their own combinations according to the teaching of information resources using multimedia teaching software systems [5]. It is not just technically the material library and teaching resources platform multimedia authoring simple sum, but ensemble platform and product combination, in which parts library integrated multimedia databases, micro teaching unit, the data presented way libraries, teaching and learning strategies libraries, online environment library of resources to teachers and students make use of combination teaching software platform provides ample material sources and a variety of effective ways. The research projects aimed at creating SCORM-based courses teaching resource object storage, the use of XML for different levels of knowledge of the contents of granularity attribute description, built on various levels in order to achieve the resource discovery and resource reuse. Using this resource storage model can be shared, reusable and other advantages, build an intelligent courseware management platform.

2. Relevant Standards and Educational Resources and Organizational

2.1. ADL-SCORM

ADL (Advanced Distributed Learning) reference model by the U.S. Department of Defense at 1997 was proposed. ADL is based on C/S network, through Internet as a web page presented to the user by the LMS, SCO and API together form a SCORM runtime environment to achieve. In the ADL system, all learning content classification according to uniform standards, creating a knowledge base for collecting, storing all learning content SCO, which base through the API interface is managed by the LMS system. This mode is mainly composed of three parts, as shown in Figure 1.

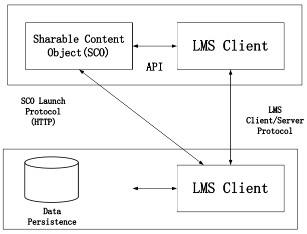

Sharable Content Object(SCO)
API
LMS Client
SCO Launch Protocol (HTTP)
LMS Client/Server Protocol
Data Persistence
LMS Client

Figure 1. The Consist of ADL System

Share content body (SCO), course content metadata specification, SCO is SCORM learning environment smallest reusable learning modules can be completed independently a learning objective, but also by multiple SCO together to accomplish a learning goal. This information through the SCORM runtime environment for the exchange of information with the LMS, SCORM established a set of rules to achieve a shared teaching system, the common use of learning content.

2.2. IMS LD

E-learning learning management system includes management of teaching content and resource, academic and administrative affairs, learning process supporting and tracking.

SCORM standards focus on sharing of teaching content and resource in different learning management system, while IMS LD (Global Learning Consortium Leering Design) Specification focus on design process of combining learning resources to a teaching unit, reuse and share of the higher level intelligent knowledge, such as learning design and teaching strategy.

IMS LD Specification define many elements for learning unit which is the main part of learning design, it is learning objectivity, role, activity, activity structure, environment, property and method. The process of learning design is selecting the reuse comments such as activity and role to form the learning activity, adding the external properties to realize adaptive learning based on learners' characteristics. Environment provides learning objects and learning services for learning activity. Such design process is platform-independent, and make us reuse and share these learning design [6]. Academic and administrative affairs management activity can refer to IMS LD Specification.

2.3. Instructional Management System

IMS is a global learning company's proposed joint learning technology system specification, has become a more influential industry standards. 19% of EDUCOM (U.S. Interuniversity Communications Committee) established a program called IMS (Learning Management System) research project, and later developed into a non-profit IMS Global Learning Associates. Instructional Systems Technology specializes in developing and promoting standards work is now in the UK, Australia, and Singapore with branch offices. IMS mainly develop educational content online publishing some of the criteria, including the storage and use of teaching content, progress tracking, and student progress reports.

IMS Content packaging Specifications in XML defines a standard format, that is, each course should be required to have a named Yeah msmanifest.xml of XML documents in this file defines the tutorial content arrangement, source of the content. The standard format describe a basic structure of distance education system, as it is defined in XML, it is very good scalability. Figure 2 is the IMS defines the data structure of the content structure.

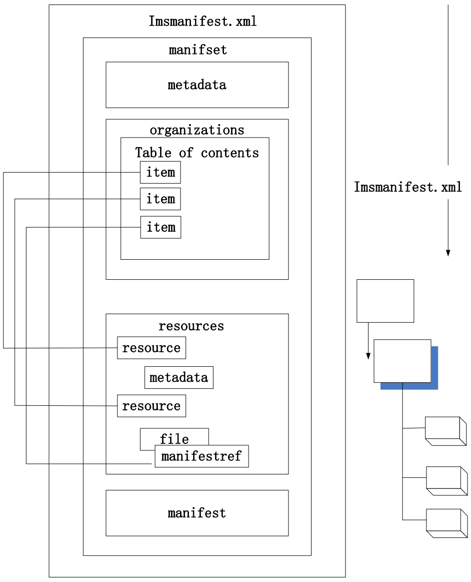


Figure 2. The File Structure Resistant of Imsmanifest.xml

2.4. Extensible Markup Language

Currently online course is still the main form of Web pages written in HTML. HTML is a markup language, although it was very widely applied, but it is undeniable that there are two drawbacks. First, HTML can't be expanded in accordance with the requirements of users. Second, HTML learning content and the content can't be separated from the description. XML structural units called entities package, each containing parsed or unparsed data. Parsed data constituted by the character can be seen as markers, the XML processing function processing; without parsed character data is not processed as the original text. It was shown in Figure 3.

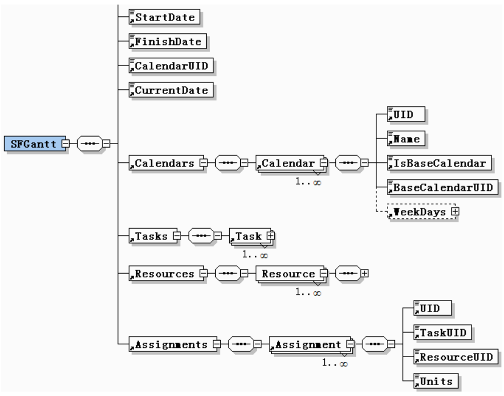


Figure 3. The Data Node of XML

3. Teaching Resource Sharing System Model Based on Semantic Web

Learning objects standard provide a better standard of teaching resource sharing model and its defined data structure and content communication method well support the sharing of learning content and strategy. Adding semantic description to the data model can realize teaching resources integration and mining based semantic rules. Figure 4 shows teaching resource sharing system model based on the Semantic Web

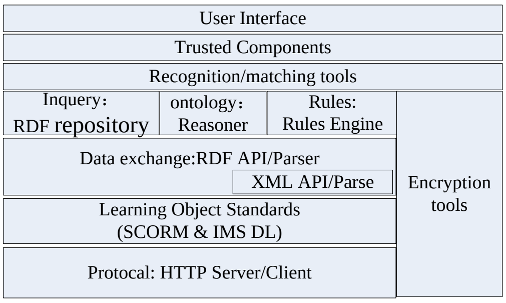


Figure 4. Teaching Resource Sharing System Model based on Semantic Web

Web server provides content services via the HTTP protocol, HTTP clients achieve browse function. Semantic Web systems use XML parser and application programming interface to process web content (such as SCORM standardized data packets) which does not satisfy the RDF standard, or use the RDF parser and programming interface to standardize resource description based on existing B/S architecture web application system.

RDF description data can be formed to RDF knowledge base, we can develop application to store, index RDF data and processing SPARQL queries on the basis of RDF repository. Inference engine can handle ontology description of ontology language and draw certain inferences. Similarly, the rule engine can process rules in RIF interchange format. When transferring sensitive personal data, you need to ensure that data transfer cannot be interrupted; the data cannot be obtained and illegally modified. Encryption tools include encryption and authentication technology to ensure data security exchange. Cryptographic modules, such as SSL processors, verify digital certificates and provide privacy and authentication password. Different data sources use different marks to represent the same thing in the real world, and Identification and alignment layer ensures data from different data sources tightly integrated. Confidence layer add extra information in data to ensure the privacy of user data. The user interface provides interface for user interaction with the Semantic Web data, it includes a common data graphical and domain-specific ontology management interface from the functional perspective [7].

3.1. Resource Description Framework

With the extensive application of metadata, each organization defines the semantics of metadata that reflect its specific needs, and sharing among applications require the common agreement of metadata about semantics, syntax and structure. Resource Description Framework thus has born [8].

Resource Description Framework is XML syntax represented resource model. It describes the characteristics of web resources and the relationship between resources. Basic concepts of Resource Description Framework is an abstract model to represent Web resources, RDF using an abstract model to divide information/knowledge into small fragments along with simple semantics (meaning) rules in order to provide a simple and flexible way to express any facts, therefore description of knowledge becomes easy to be operated by computer applications. Abstract model contains three parts: Description, Subject and Object Resources and Predicate. Knowledge fragments are the description, so knowledge can be represented by a series of descriptions. Each description is formatted in the Subject-Predicate-Object (triples) in a fixed order, Subject and Object are concrete or abstract resource name in real life, the Predicate is the name of the relationship between the two connection resources. As shown in Figure 5.

Subject
Predicate
Object

Figure 5. RDF Abstract Model

Figure 6 is an RDF teaching resource description of Multimedia Technology Course. ns0 and ns1 are the namespace which is used to distinguish different resource location. Teaching resources of Multimedia Technology Course is associated with the multimedia courseware resources through author, the associated rules is the same person have the same phone number.

3.2. Learning Object Metadata Packaged under Resource Description Framework

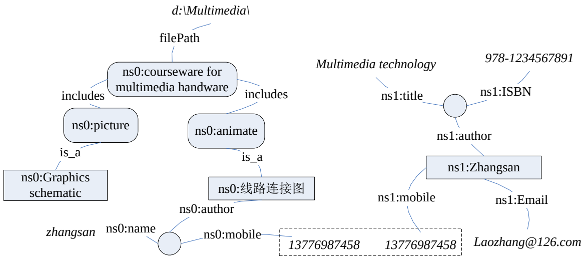


Figure 6. RDF Description of Teaching Resources

As shown in Figure 7, RDF description is added in the resource information part of the SCROM content package model [9], the relationship between the resource information are described so as to establish the relationship between resource information.

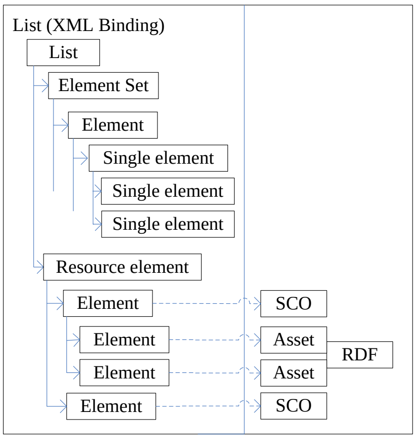


Figure 7. Improved SCORM Content Packaging Information Model

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:myCourseware="http://www.xxx.com/courseware#">
<manifest identifier="MANIFEST1" version="1.1">
<metadata>...</metadata>
<organizations>...</organizations>
<resources>
<resource identifier=”>...</resource>
</resources>
</manifest>
<rdf:Description
rdf:about="http://www.xxx.com/courseware#handwareCourseware">
<myCourseware:includes
rdf:resource="http://www.xxx.com/courseware#PIC"/>
</rdf:Description>
</rdf:RDF>
```

4. The Design of Intelligent XML-based System Based on Ontology

4.1. Courseware Making the Overall Structure of the System

Courseware template system is divided into two parts, based on SCORM standards of teaching resources and courseware design templates, the resource content and template functions separated to improve learning content reusability, scalability and portability. System architecture was shown in Figure 8.

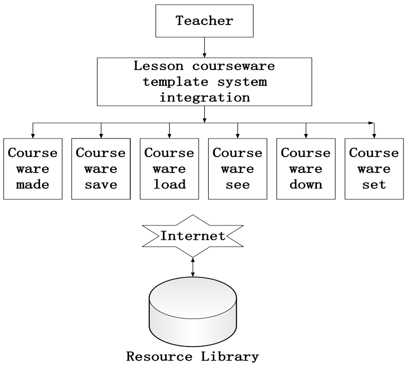


Figure 8. The Structure of Courseware Template System

4.2. Standardization of Learning Resources

Strictly speaking, the global investment in education courseware cannot say much funding, but due to the lack of early object oriented program specification and description, so that the material is difficult to integrate, *ie*, course components cannot play reuse purposes. In Schlichter model, the object is actually a course after finishing by the organization's information object components. This information is used when the problem is solved, the actual knowledge that is part of belonging to a subset. Resulting question is: How in the curriculum design, the effective integration of these information objects.

4.3. System Development Environment

Semantic Web application components are divided into main components of the Semantic Web and Semantic Web-related tools. The main components of the Semantic Web include the Semantic Web statement, the uniform resource identifier, semantic Web language, and ontology and instance data [10]. Teaching resource sharing system based on Semantic Web development use a number of tools and frameworks. Tools include the construction tool, asking tools, inference engine and rules engine, semantic framework package above-mentioned tools together to form an integrated unit.

System development environment is distributed as follows: compilation and execution tools-Java SDK; code editing tools-Eclipse IDE; ontology editing tools-Protégé; Semantic Web programming framework-Jena Semantic programming framework; Ontology Inference Engine-Pellet. Ontology editing tools is used to create, edit, and merge the ontology; Jena framework includes a query engine, storage, ontology management, inference engine and rules engine that supports interaction with the Semantic Web data through programming.

4.4. Semantic Web Systems Development Process

Firstly, Add Jena and Pellet library in project.

Secondly, precede the next four steps:

- Programming to query instance resources;
- Aligning the instances resources and the domain ontology;
- Reasoning for resource description data by integrating multiple resource database management system and domain ontology;
- Determine the constraints and rules.

Finally, realize resource integration function in multiple systems which meet certain query conditions. The program process consists of four main steps: obtaining a structured storage space or model, using the Semantic Web data to populate the model, and possibly execute data proceed (queries, logic proceed, etc.), and finally output data that meet the conditions.

- Obtain model and populate data

```
public class RDFPopulateModel {
static string defaultNameSpace=”http://www.xxx.com/courseware#”
private Model_resources=null;
private void populateResourceRDF() {
Model_resources=ModelFactory.createOntologyModel();
InputStream inResourceInstance=FileManager.get().open(”Ontologies/resource.rdf”);
Model_resources.read(inResourceInstance,defaultNameSpace);
inResourceInstance.close();}}
```

- Data proceed (instance query) and output

```
private void runQuery(String queryRequest,Model model) {
//create and run Query
Query query=QueryFactory.Create(queryStr.toString());
QueryExecution qexec=QueryExecutionFactory.create(query,model);
//select in query result
ResultSet response=qexec.execSelect();
While(response.hasNext){
QuerySolution soln=response.nextSolution();
RDFNode url=soln.get(“?url”);
if(url!=null){
//output data
……
//
}}}
```

- Align the instance and ontology

At first, we should create model, two ontologies related with two resources and two resource instances data are filled into the model, and then create an equivalent or specialization relationships statement will align the two data sets and add them into the model. Secondly, bound the model with Jena inference engine and finally perform data processing (query instance) and outputting.

- Determine the constraints and rules

Firstly, create a specific concept or class, to form a constraint that describes the logic relationship between members for limited class, for example, it limits that courseware must contain a picture descriptions. Secondly, the constraints are readied into the model bounded with inference engine and process the query. In limited areas beyond the construct ability of the ontology, we can establish a rule using the rule language, when the rule is satisfied, the inference engine add a statement in the model, to associate a new ontology class with instances and finally perform data processing (query instance) and output.

5. Ontology Matching Method Based on Approximate Subgraph Isomorphism

5.1. Overall Framework

Overall framework based e-Learning resource ontology matching approximate subgraph isomorphism (Ontology Matching based Subgraph Isomorphism, SIOM) is shown in Figure 9 below. SIOM is a sequence matcher which mainly includes four major steps, they are anchor select and subgraph extraction, sub-graph structure similarity calculation, subgraph approximate isomorphism determination and ontology matching based on similar isomorphic subgraphs.

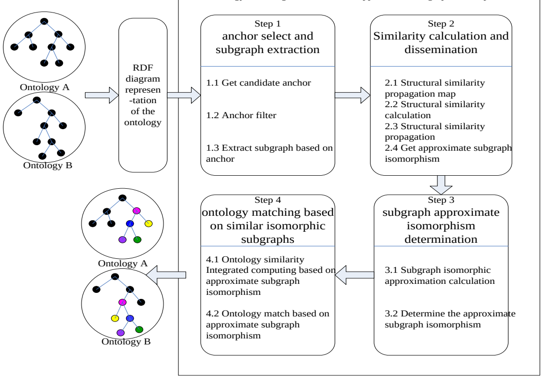

Ontology matching method base on approximate subgraph isomorphism
Ontology A
Ontology B
RDF diagram represen -tation of the ontology
Step 1
anchor select and subgraph extraction
1.1 Get candidate anchor
1.2 Anchor filter
1.3 Extract subgraph based on anchor
Step 2
Similarity calculation and dissemination
2.1 Structural similarity propagation map
2.2 Structural similarity calculation
2.3 Structural similarity propagation
2.4 Get approximate subgraph isomorphism
Step 3
subgraph approximate isomorphism determination
3.1 Subgraph isomorphic approximation calculation
3.2 Determine the approximate subgraph isomorphism
Step 4
ontology matching based on similar isomorphic subgraphs
4.1 Ontology similarity Integrated computing based on approximate subgraph isomorphism
4.2 Ontology match based on approximate subgraph isomorphism
Ontology A
Ontology B

Figure 9. Ontology Matching Framework based on Approximate Subgraph Isomorphism

5.2. Anchor Selection and Subgraph Extraction

Anchor, referred to the first pair similar concept be determined in ontology A and B which to be matched, just is the first pair determined matched nodes on the markers on the map of ontology. Defined as follows:

Definition 1: (Anchor) Given two ontologies A, B to be matched, corresponding structure map is $G(A)$, $G(B)$, if one node $x \in C_A$ in $G(A)$, always make one node $y \in C_B$ exist in $G(B)$ to be:

- $OM\ (x,y)$, that is x matched y ;
- $\forall I(x) \in I_A, P(x) \in P_A, Hc\ (x) \in Hc_A, R(x) \in R_A, A_A^0(x) \in A_A^0$,

$\exists I(y) \in I_B, P(y) \in P_B, Hc\ (y) \in Hc_B, R(y) \in R_B, A_B^0(y) \in A_B^0$, can be

$$OM\ (I(x), I(y)) \wedge OM\ (P(x), P(y)) \wedge OM\ (R(x), R(y)) \\ \wedge OM\ (Hc\ (x), Hc\ (y)) \wedge OM\ (A_A^0(x), A_B^0(y)) \qquad (1)$$

$< x, y >$ is a pair anchor of A、B, x and y is called anchor concept.

Anchors can be divided into the following nine kinds of situations depending on the anchor at the location of the ontology hierarchy:

- x , y are root nodes of $G(A)$, $G(B)$ respectively;
- x is root node of $G(A)$, y is middle node of $G(B)$;
- x is root node of $G(A)$, y is leaf node of $G(B)$;
- x is middle node of $G(A)$, y is root node of $G(B)$;
- x is middle node of $G(A)$, y is middle node of $G(B)$;
- x is middle node of $G(A)$, y is leaf node of $G(B)$;
- x is leaf node of $G(A)$, y is root node of $G(B)$;
- x is leaf node of $G(A)$, y is middle node of $G(B)$;
- x is leaf node of $G(A)$, y is leaf node of $G(B)$.

Definition 2: Given ontology:O, x is the anchor concept of O, and the ontology O^x is derived by anchor can be expressed as pentad: $O^x = (C^x, I^X, P^x, Hc^x, R^x, A_x^0)$, in:

- $C^x = \{c \in C \mid (c\ Hc\ x) \vee (x\ Hc\ c) \vee (c\ R\ x) \vee (x\ R\ c)\}$ are sets of Concepts.
- $P^x = \{P \to \{C^x\}\}$ 、 $I^x = \{I \to \{C^x\}\}$ is Attribute Set and Instance Set.
- $Hc^x = \{Hc \to \{C^x\}\}$ is Hierarchy Set of Concepts in O^x .
- $R^x = \{R \to \{C^x\}\}$ is other Hierarchy Set of Concepts in O^x .

Inference 1: Given Ontology:O and Ontology O^x is derived by its anchor concept, if digraphs: $G(O)$, $G(O^x)$ are corresponding digraph structure representation of referred ontologies, there is

$$G(O^x) \subseteq G(O) \qquad (2)$$

Proof: By definition 1,2,7, Inference 1 is established.

Inference 2: Given Ontology:O and Ontology O^x is derived by its anchor concept, there is something for digraphs: $G(O)$, $G(O^x)$:

- if x is root node of $G(A)$, there is $G(O^x) = G(O)$;
- if x is not root node of $G(A)$, there is $G(O^x) \subset G(O)$.

Particularly, when x is leaf node of $G(A)$, $G(O^x)$ is the child node of $G(O)$.

Proof: From the above analysis and inference concept of anchor location in the ontology hierarchy and Inference 1, Inference 2 is established.

5.3. Ontology's Digraph Structure Similarity Calculation

The Similarity Calculation for Ontologies A 、 B and their digraph representations $G(A)$, $G(B)$, $G(A)$ and $G(B)$ is composed by four parts:

- node edit distance similarity;
- similarity between nodes in a hierarchical relationship;
- similarity between nodes in other relationships;
- Integrated structure similarity in digraph.

There are interpreted as follows:

- Edit distance similarity calculation: conceptual similarity, attribute similarity consolidated been represented by the node. Specific methods are as follows: Let x and y, respectively, are two nodes of $G(A)$,. $G(B)$, $S_e^c(x,y)$ is edit distance of x and y, $S_e^p(x,y)=\frac{2\,|\,p\,|_{p\in P^A \mathrm{I} P^B}}{|P^A|+|P^B|}\sum_p S(p(x),\ p(x))$ is total property edit distance of x and y, calculating by appling Reference 1, the similarity between the node x and y is calculated as follows:

$$S_e(x,y)=\alpha\bullet S_e^c(x,y)+\beta\bullet S_e^p(x,y) \tag{3}$$

α , β is the weight adjustment parameter, and $0<\alpha,\beta\le 1\wedge\alpha+\beta=1$.

- Similarity hierarchical relationships between nodes: Let the penetration set of hierarchical relationships of x in $G(A)$ is $x_{in}=\{x_i\in V(A)\,|\,\exists x_i\ Hc\ \ x\}$, out of the set of x in $G(A)$ is $x_{out}=\{x_j\in V(A)\,|\,\exists x\ Hc\ \ x_j\}$, analogously, the prnetration and out set of hierarchical relationships of y in $G(B)$ is y_{in} , y_{out} respectively. The hierarchical relationship similarity between x and y is calculated as:

$$S_{Hc}(x,y)=\frac{|x_{in}\Leftrightarrow y_{in}|+|x_{out}\Leftrightarrow y_{out}|}{|x_{in}|+|x_{out}|+|y_{in}|+|y_{out}|-|x_{in}\cap x_{out}|-|y_{in}\cap y_{out}|} \tag{4}$$

$x_{out}\Leftrightarrow y_{out}=\{x\,|\,x\in x_{out},y\in y_{out}:S_e(x,y)\in OM\ (x,y)\}$ represent can be matched set of nodes in child nodes which have a hierarchical relationship with x and y .

Other relationships similarity between nodes: there is a set of nodes have relations R between x and y :

$$x^R=\{x'\in V(A)\,|\,r\in R^A:\exists\,(x'\ r\ \ x)\vee(x\ r\ \ x')\}\quad y^R=\{y'\in V(B)\,|\,r\in R^B:\exists(y'\ r\ \ y)\vee(y\ r\ \ y')\}$$

If $\exists r_1\in R^A,r_2\in R^B$, there is

$$((x'\ r_1\ \ x)\wedge(y'\ r_2\ \ y)\wedge OM\ (x',y'))\vee((x\ r_1\ \ x')\wedge(y\ r_2\ \ y')\wedge OM\ (x',y')) \tag{5}$$

Denoted a set of nodes meet the formula (5) to be $x^R\Leftrightarrow y^R$, Combined weight adjustment factor, and the other relationships similarity between nodes can be expressed as:

$$S_R(x,y)=\sum_{r\in R^A}\gamma_i\frac{|x^r\Leftrightarrow y^r|}{|x^r|+|y^r|}+\sum_{r\in R^B}\lambda_i\frac{|x^r\Leftrightarrow y^r|}{|x^r|+|y^r|}-\sum_{r\in R^A\cap R^B}(\gamma_i+\lambda_i)\frac{|x^r\Leftrightarrow y^r|}{|x^r|+|y^r|} \tag{6}$$

Weight adjustment factor γ_i , λ_i meet $0<\gamma_i,\lambda_i\le 1\wedge\sum_i r_i=1\wedge\sum_i\lambda_i=1$ 。

Structural similarity of digraphs：ontologies to be matched A and B and their digraph $G(A)$ and $G(B)$, $<x,y>$ is a pair of anchor points of A, B, the similarity between digraph $G(x)$ and $G(y)$ exported by x and y can be expressed as:

$$S(G(x),G(y))=\eta\bullet S_e(x,y)+\varphi\bullet S_{Hc}(x,y)+\xi\bullet S_R(x,y) \tag{7}$$

$\eta+\varphi+\xi=1$ is weight adjustment factor.

5.4. Ontology-based Approximate Subgraph Isomorphism Matching Algorithm

Definition 3: If there are points to keep the correspondence between the map G and G' , and keep the correspondence between the edges, furthermore, keep the same relationship between corresponding points and the corresponding side, we call G and G' is isomorphic, denoted by $G\cong G'$.

Since it is generally difficult to achieve matching strict correspondence of the ontologies, as long as the degree of similarity between the ontologies meet a preset threshold, can be determined for the match, so this paper presents the approximately isomorphic structure of the ontology digraph.

Definition 4: Given target ontolgoy A and to be matched ontology B, which is represented as a directed graph $G(A)$, $G(B)$, if

- exists a node b in $G(B)$, there has a pair of anchor $<a,b>$ in A and B for the root node of $G(A)$;
- For Digraph $G(B^b)$ of ontology B^b exported by $G(A)$ and b , there has

$$V(A) \subseteq V(B^b), E(A) \subseteq E(B^b) ;$$

$$\forall x \in V(A) : \exists y \in V(B^b) : OM(x, y) ;$$

$$\forall e \in E(A) : \exists e' \in E(B^b) : OM(e, e') ;$$

For matching threshold θ , there has $S(G(A), G(B^b)) \geq \theta$;

A and B are called ontology diagram approximation isomorphic, denoted by $G(A) \approx G(B)$.

Based on the approximate isomorphic of figure, the main idea of SIOM algorithm is: Traversal order according to the figure scope preferentially, Achieve alternating matching between nodes and edges of figure based on access node degree after determining a matching anchor node, in order that we can find an approximate isomorphic subgraphs between $G(A)$ and the graph $G(B)$ of ontology B to be matched. The key steps are: First, determine the root node a in $G(A)$ corresponding to anchor node b in $G(B)$; then generate anchor B and export ontology B^b and its diagraph $G(B^b)$; Furthermore, execute figure isomorphism approximate calculation between $G(A)$ and $G(B^b)$. If they meet the approximate isomorphic relationship, called A, B are matched, otherwise, the iterative process is executed until they archive convergence requirements. Table 1 shows the Pseudo-code description of main operations of the algorithm.

5.5. Time Complexity Analysis of Algorithm

The time complexity of Pseudo-code description Algorithm given in Table 1 depends on the three-layer nested loop approximation subgraph isomorphic.

If there have $|V(Q)| = n$ and $|E(Q)| = m$ in digraph $G(Q)$ of ontology Q , $|V(Q')| = N$ 、 $|E(Q')| = M$ in digraph $G(Q')$ of ontology Q' , the main operational unit of time required for the algorithm to complete the matching of ontology Q' and Q were:

The required time anchor for the first amount of node matching is $n * N$;

For node $<x, y>$, required time for edge matching is $E(x) * E(y)$;

For structure similarity of child digraph of $G(x)$ and $G(y)$, the require time is:

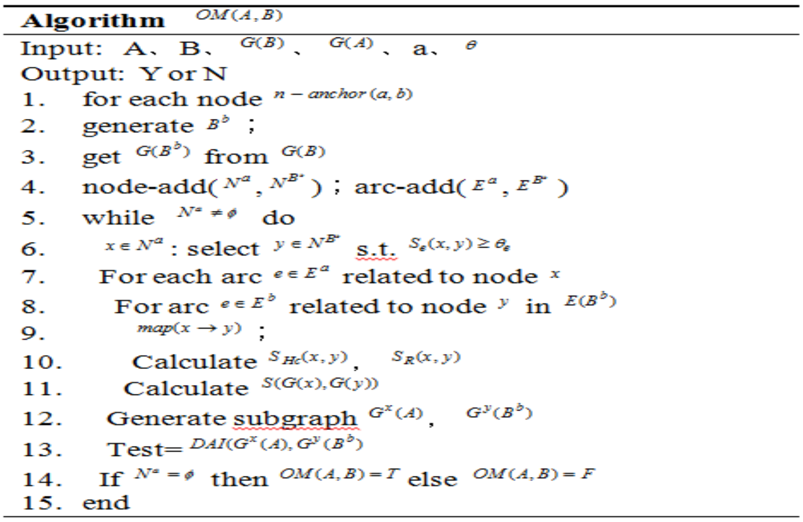
$C^2_{|V(G(x))|} * C^2_{|V(G(Y))|} * E(x) * E(y)$

When amount of nodes and edges are n 、 N 、 m 、 M respectively, require time of algorithm is

$$\frac{n(n-1)}{2} * \frac{N(N-1)}{2} * m * M \qquad (8)$$

Therefore, the time complexity of the algorithm is $O(n^6)$ level, algorithm is valid.

Table 1. Ontology-based Approximate Subgraph Isomorphism Matching Algorithm



```
Algorithm $OM(A,B)$
Input: A、B、$G(B)$、$G(A)$、a、$\theta$
Output: Y or N
for each node $n-anchor(a,b)$
generate $B^b$ ;
get $G(B^b)$ from $G(B)$
node-add( $N^a$, $N^{B^b}$ ) ; arc-add( $E^a$, $E^{B^b}$ )
while $N^a \neq \phi$ do
   $x \in N^a$ : select $y \in N^{B^b}$ s.t. $S_e(x,y) \geq \theta_e$
    For each arc $e \in E^a$ related to node $x$
      For arc $e \in E^b$ related to node $y$ in $E(B^b)$
        $map(x \to y)$ ;
        Calculate $S_{Hc}(x,y)$, $S_R(x,y)$
       Calculate $S(G(x),G(y))$
     Generate subgraph $G^x(A)$, $G^y(B^b)$
     Test= $DAI(G^x(A),G^y(B^b))$
    If $N^a = \phi$ then $OM(A,B)=T$ else $OM(A,B)=F$
end
```

6. Summary

Courseware is clear teaching objectives, teaching content reflects textbook structure and have the appropriate teaching strategies and process systems. System defines two agents, one for the teacher interface agent for receiving input information teachers and curriculum recommendation agent with another agent to communicate. Course recommendation agent mainly to complete the selection according to teachers in advance in accordance with the repository attribute settings, adjust the course object order and, where appropriate curriculum content recommended. The research projects aimed at creating SCORM-based courses teaching resource object storage, the use of XML for different levels of knowledge of the contents of granularity attribute description, built on various levels in order to achieve the resource discovery and resource reuse.

By adding semantic association information into existing learning techniques standards, we can inquiry, integrate and mining distributed learning resources more intelligently. This study take the Multimedia Technology course's teaching resource management for example to execute practice study and the associated recommended completeness of course's teaching materials and reference materials has been greatly improved over traditional resource retrieval method, types of teaching resources related with teaching content can be fully integrated and displayed in the end.

Acknowledgements

This work was financially supported by the National Science Education 12th Five-Year Plan key project of Ministry of education under Grant No.: DCA110190, the Natural Science Foundation of Jiangsu Province under Grant No. BK20141152, the Key Laboratory of Cloud Computing and Intelligent Information Processing of Changzhou City under Grant No: CM20123004, Qing Lan Project of Jiangsu Province of China, the Natural Science Foundation and Doctoral Research Foundation of Jiangsu University of Technology under Grant Nos: kyy12018 and kyy13003.

References

[1] C. Sun, "Intelligent Surveillance Using Inductive Vehicle Sensors", Intellimotion, vol. 8, no. 3, **(2001)**, pp. 10-100.

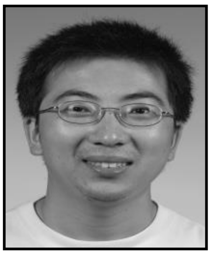
[2] J. Z. Li, J. B. Li and S. F. Shi, "Concepts, Issues and Advance of Sensor Networks and Data Management of Sensor Networks", Journal of Software, vol. 14, no. 10, **(2003)**, pp. 1717-1727.

[3] S. Tilak, N. B. Abu-Ghazaleh and W. Heinzelman, "A Taxonomy of Wireless Micro-sensor Network models", Mobile Computing and Commuications Review, vol. 1, no. 2, **(2002)**, pp. 1-8.
[4] D. Culler, D. Estrin and M. Srivastava, "Overview of Sensor Networks", IEEE Computer Society Officers, vol. 11, no. 10, **(2005)**, pp. 15-25.
[5] T. Matsuo, Y. Kaneko and M. Matano, "Introduction of Intelligent Vehicle Detection Sensors", IEEE/IEEJ/JSAI International Conference on Intelligent Transportation Systems, vol. 5, no. 12, **(1999)**, pp. 709-713.
[6] X. Cao and K. He, "New Developments in Learning Design and Learning Management Systems", Modern Educational Technology, vol. 16, no. 4, **(2006)**, pp. 5-8.
[7] X. Yang and S. Yu, "Learning Resource Information Model Building in Ubiquitous Learning Environment", China Educational Technology, no. 9, **(2010)**, pp. 72-78.
[8] L. Yu, "A Developer's Guide to the Semantic Web", New York: Springer-Verlag Berlin Heidelberg, **(2001)**.
[9] G. Antoniou and F. V. Harmelen, "A Semantic Web Primer", 2rd ed., London: The MIT Press, **(2008)**.
[10] K. Zhu, "Research on key technologies of the Semantic Web Teaching Resources Management System", Thesis of Tianjin University of Technology, **(2010)** January.
[11] D. Allemang and J. Hendler, "Semantic Web for the Working Ontologist: effective modeling in RDFS and OWL, 2rd ed., Waltham: Morgan Kaufmann Publishers, **(2011)**.
[12] B. Chen, "On the Resource Description Framework RDF: a highly portable tool vitality metadata", Library and Information Technology, no. 6, **(2000)**, pp. 62-64.
[13] J. Hebeler, M. Fisher and R. Blace, *et al*., "Semantic Web Programmin", Indianapolis: Wiley Publishing, Inc., **(2009)**.

Authors

Haixu Xi received the master's degree in Educational Technology from Nanjing normal University in 2006. Currently, he is a lecturer at the School of Computer School of Computer Engineering in Jiangsu University of Technology. His interests are in digital teaching resource development and multimedia technology.

Wei Gao received the master's degree in Educational Technology from YangZhou University in 2013.Currently, he is an lecturer at the School of Computer School of Computer Engineering Jiangsu Teachers University of Technology. His interests are in digital teaching resource development and multimedia technology.



Jin Qian received his Ph.D. degree in pattern recognition and intelligent system from Tongji University in 2013. Now he is an associate professor at Jiangsu University of Technology. His research interests include big data, rough set theory, data mining and cloud computing, *etc*.

.