Study on the Federal Search of the OPAC and e-Learning Resource Database[1]

Qinghua Zheng[1], Haifeng Dang[2], Huixian Bai[2], Jing Shao[3]

1 2 Computer Department of Xi'an Jiaotong University, China, 710049
3 Xi'an Jiaotong University Library, Xi'an, China ,710049

1 qhzheng@mail.xjtu.edu.cn
2 {xjtu_hfdang,baihuixian}@163.com
3 jshao@mail.lib.xjtu.edu.cn

Abstract

OPAC resource and e-Learning resource have many inner relations especially for e-Learning services. Unfortunately, nowadays they are independent of each other due to their heterogeneity, differences in data format, technical standard, storage mode, and supporting platform. On the basis of analyzing the characteristics of the two resources, we constructed an integration platform, presented a unified description for the exchange of heterogeneous resources, and a joining and ranking model of the searched results from OPAC and e-Learning resources. We have already developed a prototype of union retrieval system named *Fsearch* based on Java platform, and it has been tested on the heterogeneous and distributed resources of XJTU Library's OPAC and e-Learning resource database. Now learners can search the two resources in a one website, and the search response time is acceptable. From the test of *Fsearch*, it shows that such union retrieval systems can effectively improve the efficiency and quality of acquiring resources.

Keywords: OPAC Resource, e-Learning Resource, Resource Integration, Unified Resource Description, Result Ranking

I. Introduction

OPAC is the main bibliography information database in library, while e-Learning Resource Database is the basis of network education. Since these two kinds of resources are independent of each other, and different in technical standard, metadata format and database management, they have different retrieve mode. OPAC is stored and managed by library, while e-Learning resource is stored and maintained by network education school. Therefore, they are two heterogeneous and distributed resources.

But, in fact, these two resources have many relations. Both of them are teaching and learning materials, they have many inherent knowledge relations, they are complementary each other since resources from the OPAC are the set of formal publications while e-Learning resources are the set of multimedia courseware which are usually informal teaching materials. However, because of their heterogeneity and distribution, people who want to learn some curriculum say "computer network" over the Internet, have to get the courseware from e-Learning website in one hand, and get the teaching materials or references from library's OPAC in the other hand. It makes complex and

inconvenient for Internet users to learn with these two resources. So, integrate the two heterogeneous resources, so that learners can search teaching or learning materials through "one web site" and one query, which can not only improve the learning efficiency, but also enhance the learning quality by joining the two searched results together.

We have developed a prototype called *Fsearch* based on J2EE application framework, and it has been applied on the heterogeneous and distributed resources of XJTU(Xi'an JiaoTong University) Library's OPAC and e-Learning resource database. This paper will describe the architecture of *Fsearch*, a unified resource description, and the join of searched results, especially a similarity calculating algorithm.

The rest of this paper is organized as follows: section 2 discusses the related works of this research; section 3 introduces the architecture of *Fsearch* system; section 4, section 5 and section 6 describes three key technologies: unified Description of OPAC and e-Learning Resources, join of search outcomes from two resources and results ranking; section 7 shows the application and experiment; and the last section is the conclusion.

II. Related Work

Up to our knowledge, there are few researches on the integration of OPAC and e-Learning resources, but there are some mature technologies to integrate other heterogeneous resources. Those technologies can be summarized to two main approaches:

A. ***Federal search across databases directly on-line.***
They build a united platform, search each heterogeneous database concurrently, and provide a consistent result display by integrating the search results from each database. The advantage of this approach is that the search results always keep consistent with the data source; the disadvantage is that the response time may be much longer. C. Boyer developed HONselect to implement the federal search across some heterogeneous medical databases [1], where the databases are indexed according to medical classification, which can decrease the response time remarkably. Wang Lancheng from NanJing Political College constructs searching agents basing on XML and COM technology for federal search [2], one agent for one database, which is convenient for a new database to be added. Additional, MetaLib/SFX [3] from Ex Libris and MAP [4] from Innovative are two mature systems implemented also by this approach.

B. ***Centralized search based on metadata harvest.***
It is called OAI-PMH (Open Archives Initiative - Protocol for Metadata Harvesting). They maintain a local metadata database which contains a duplication of the metadata of each heterogeneous database, harvest metadata to update the local database timely, and when a search occurs they only search the local metadata database to return the results. The advantage of this approach is that its search speed is much higher relatively; The disadvantage comprises the follow points: 1) A large local metadata database must be maintained, and it must be updated timely, so the maintenance cost must be high. 2) The results returned may be not consistent with the records in the original metadata database. 3) The duplicating of the whole metadata database may invoke some copyright problems. The most typical application of this approach is CALIS (China Academic Library & Information System) [5], which integrates the OPAC resources of over 150 academic libraries throughout China. CALIS constructs a central database, where each academic library uploads and updates its own data, and every search will be executed on the central database. Therefore, CALIS is a large project, it is labor and time consuming, but of course, it can provide rapid and comprehensive searches that any single academic library could not achieve.

We implement our system, Fsearch, using the approach having the same manner with the first one mentioned above, with the consideration of the following:

1. We only integrate two kinds of resources, and they are based on high-speed university campus

network environment, so the response time will not be too long;
2. The resource databases of OPAC and e-Learning are updated frequently.

III. The Architecture of Fsearch System

OPAC resource and e-Learning resource are two distributed and heterogeneous resources, they have different information retrieval mode due to their differences in technical standard, metadata format, database management, and so on. Table 1 shows the differences between OPAC resource and e-Learning resource.

Table 1. The comparison of OPAC and e-Learning Resources

	OPAC	e-Learning
Management system	Z39.50 Server(e.g. INNOPAC)	common commercial DBMS(e.g. ORACLE)
Access approach	Z39.50 Gateway	database connectivity interface (e.g. JDBC)
Search expression	RPN (Reverse Polish Notation)	SQL query sentence
Search result	ISO2709 record	SQL ResultSet
Metadata standardization	MARC	CELTS [6]

As a result, to integrate these two kinds of resources, we construct a middleware named USE (Union Search Engine), which is based on J2EE component technology and J2EE/JSP three layers application.

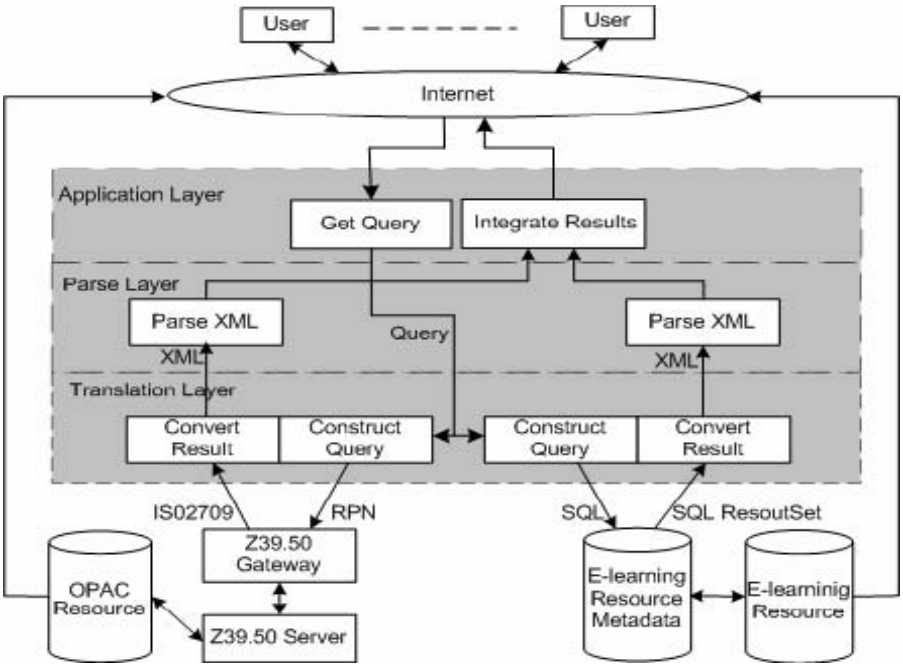

Fig.1. *Fsearch* System Architecture

Union Search Engine is composed of three layers: Application Layer, Parse Layer and Translation Layer. It runs as following process:

Step1. The Application Layer accepts user's query and submits it to the Translation Layer.

Step2. The Translation Layer constructs two search expressions, one for the search on OPAC as RPN expression, which is submitted to Z39.50 gateway, the other is for the search on e-Learning resource database as SQL expression, which is submitted to the search engine of e-

Learning resource.

Step3. Executes two searches on OPAC and e-Learning resource databases respectively, and then collects the OPAC results as ISO2709 format and the e-Learning database results as SQL ResultSet to the Translation Layer respectively.

Step4. Convert the two kinds of results into a unified XML representation, and then pass it to the Parse Layer.

Step5. The Parse Layer passes these XML documents to The Application Layer.

Step6. The Application Layer completes the join of two results and displays the final result to user.

Step7. End and go to next federal search.

In this process, two key issues are included, one is how to design a unified Description of heterogeneous resources based on XML, and the other is how to join the searched result of OPAC with the one of e-Learning resource.

Ⅳ. The Unified Description of OPAC and e-Learning Resources

For the sake of the unified processing and the exchange of OPAC and e-Learning resources, the first problem is to define a unified representation. Of course, XML is the best tool to describe the heterogeneous resources at present, because it has a good extensibility, is independent of any system platform, and has the ability to express its content by itself [7]. In Fsearch, we use XML to describe the two heterogeneous resources, OPAC resource and e-Learning resource. But before we introduce the XML into resource representation, we must understand the standard representation of these two resources.

OPAC resources use MARC as the metadata. An OPAC resource is represented by a MARC record, which is constituted by several fields distinguished by their tags. A field is composed of five kinds of elements: its field tag, identifier1, identifier2, code, and content. It must be explained that a field may have one single content (code has a null value) or several contents with each one having a code value. More information about MARC standard could be found at http://www.loc.gov/marc/.

CELTS is the Chinese specification of e-Learning resource. According to CELTS-31 (Education Resource Construct Specification), a description of an e-Learning resource record is composed of several metadata items, each of which belongs to one of three categories: the core data elements, the common data elements, and the expended data elements. Where, the core data elements are defined according to IEEE LOM (Learning Object Metadata) standard. More information about CELTS could be acquired at http://www.edu.cn/html/keyanfz/yuanchengjiaoyu.shtml.

Having known the standard representation of each resource, then we will show the definition of the unified XML representation [8]. The DTD (Document Type Definition) of it is shown as follow:

```
<!ELEMENT record    (field+)>
<!ATTLIST record    type            #REQUIRED>
<!ELEMENT field     (#PCDATA | code+)>
<!ATTLIST field     fieldtag        CDATA>
<!ATTLIST field     etype (1|2|3)>
<!ATTLIST field     identifier1     CDATA>
<!ATTLIST field     identifier2     CDATA>
<!ELEMENT code      (#PCDATA)>
<!ATTLIST code        codetag       CDATA       #REQUIRED>
```

Where, the attribute type indicates the resource type, its value could be "OPAC" or the type name of any e-Learning resource (e.g., multimedia courseware, test paper etc.). If the resource is e-Learning, that is to say the value of type is not "OPAC", then the value of fieldtag is the Name of data element of an e-Learning record, the value of etype only can be selected from 1, 2, and 3, which respectively denotes the data element is a core data element, common data element or expended data element, and no identifier1, identifier2 attributes and code element appear. If the resource is OPAC, that is to say the value of type is "OPAC", then the value of fieldtag is the tag value of each field in the MARC record, the values of identifier1, identifier2 and code are same to their values in the MARC record respectively, and no etype attribute.

Ⅴ. Join of Two Outcomes

The destination of the federal search is to generate an integrated result, which is joined from the outcome of the search on OPAC database and the one of e-Learning database. The join process includes the following steps:

1) Collect two outcomes into one result set and use a uniform attribute set to represent each result;
2) Sort the results in the result set according to the similarity of the query and each result;
3) Discover the correlations among results and associate correlating results;
4) Display the joint results to users.

In this process, we have 3 key issues:

1) Union of two resources' attribute sets. The XML representation discussed in section 4 realizes the unified description of resources in form only, for the uniform result display to the users we must also design a uniform attribute set to represent the two kinds of resources. Suppose that, the attribute set of OPAC resource is Attro (A1, A2, …, Am) and the attribute set of e-Learning resource is Attre (B1, B2, …, Bn). The simplest way to buildup the union attribute set is to combine the two attribute sets of each resource, that is to say the union attribute set is Attru(A1, A2, …, Am, B1, B2, …, Bn). But, some attributes of the two resources have the same meaning, for example, the attribute A1 may be the title of OPAC resource in Attro and the attribute B1 may be also the title of e-Learning resource in Attre. In this condition, the attributes in Attru appear repeated attributes, A1 and B1. Therefore, it is necessary to remove such repetitions.

After analyzing the metadata of the two resources, we extract 12 pairs of similar attributes and map them to the corresponding attributes of DC (Dublin Core). The mapping relation of DC, USMARC and LOM is shown in Table 4[9]:

Table 4. Corresponding relations of attributes

DC	Creator	Title	Subject	Description
USMARC	700$a	245$a	650ab	520$a
LOM	Lifecycle: Contribute: Entity	General: Title	General: Keywords	General: Description
DC	Date	Format	Identifier	Rights
USMARC	260$c	"bibliography"	024$a	540$a
LOM	Lifecycle: Date	Technical: Format	General: Identifier	Rights
DC	Source	Language	Coverage	Relation
USMARC	856$u	546$a	651	530ab
LOM	Technical: Location	General: Language	General: Coverage	Relation

These 12 attributes are treated as the core attributes of the union attribute set, and the rest attributes of each resource are regarded as extended attributes for the united resources.

2) Results ranking: will be discussed in section 6.
3) Results Associating and hyperlink. OPAC resource and e-Learning resource have many inherent relations. In *Fsearch* system, we considered two ways to associate them. First, associate a result to all the results that have a similar theme with it, which can be weighed by a threshold and the similarity between the considered result and the other results. The similarity of two records can be computed using the similar method with the similarity computation between query and record, which will be introduced in section 6. Associate the considered result to all the results whose similarity with the considered result bigger than the threshold. Second, most e-Learning resources have designated books and reference books, which may be searched and acquired in OPAC resource. So, we also add a hyperlink to each designated book and each reference book of e-Learning resource results for searching it in the OPAC, so that user can check the library information of these books and acquire them (when authorized) in OPAC when viewing an e-Learning resource result.

Ⅵ. Results Ranking

Among all the attributes of a result, Title, Author, Subject and Description have the most ability to express the content of the result. And they are also the search points provided by Fsearch. So, we rank results according to the similarity between the query and the values of these four attributes of results. Since all of the values are texts, we must map them to corresponding vectors, which can be done with VSM (Vector Space Model) [10].

Fsearch provides four search points: Title, Author, Subject and Description. So, the query from a user is the logical combination of some of the following formulas: Title=*QTValue*, Author=*QAValue*, Subject=*QSValue*, Description=*QDValue*. For example,

"Title=*computer network* ∧ (Subject=*Tcp/ip* ∨ Subject=*network security*)" (6-1)

We also use the four attributes to represent a result, that is, a result can be represented by a formula:

"Title=*TValue* ∧ Author=*AValue* ∧ Subject=*SValue* ∧ Description=*DValue"* (6-2)

Our algorithm for calculating the similarity between a query and a result is shown below:

Input: query Q, a result R_x

Output: the similarity between Q and R_x

Step1.

FQ=toCNF(Q), represent FQ as:

$$FQ = \bigvee_{i=1}^{n}(Title = TV_i \wedge Author = AV_i \wedge Subject = SV_i \wedge Descriptio\ n = DV_i);$$

Step2.

RT=getTitle(R_x), RA=getAuthor(R_x), RS=getSubject(R_x), RD=getDescription(R_x);

$\mathbf{RV_1}$=toVector(RT), $\mathbf{RV_2}$=toVector(RA), $\mathbf{RV_3}$=toVector(RS), $\mathbf{RV_4}$=toVector(RD);

RVG=($\mathbf{RV_1}$, $\mathbf{RV_2}$, $\mathbf{RV_3}$, $\mathbf{RV_4}$);

Step3.

$\mathbf{QV_{i1}}$=toVector(TV_i), $\mathbf{QV_{i2}}$=toVector(AV_i), $\mathbf{QV_{i3}}$=toVector(SV_i), $\mathbf{QV_{i4}}$=toVector(DV_i);

QVG_i=($\mathbf{QV_{i1}}$, $\mathbf{QV_{i2}}$, $\mathbf{QV_{i3}}$, $\mathbf{QV_{i4}}$);

Step4.

for(i=1; i≤n; i++)

$$VSim(RVG, QVG_i) = \frac{\sum_{j=1}^{4}\left[M(RV_j, QV_{ij}) \times Cos(RV_j, QV_{ij})\right]}{\sum_{j=1}^{4} M(RV_j, QV_{ij})} \quad (6\text{-}3)$$

Step5.

$$\mathrm{Sim}(Q, R_x) = \max_{i=1}^{n}(VSim(RVG, QVG_i)) \quad (6\text{-}4)$$

Step6.

return Sim(Q, R_x).

In the algorithm, toCNF(Q) means to return the corresponding CNF (Conjectured Normal Formula, the disjunction format of some conjectured formulas) of Q. For instance, the CNF of (6-1) is "(Title=*computer network* ∧ Subject=*Tcp/ip*) ∨ (Title=*computer network* ∧ Subject=*network security*)"; getTitle(R_x), getAuthor(R_x), getSubject(R_x) and getDescription(R_x) mean to return the values of title, author, subject and description of R_x respectively; toVector(S) means to return the corresponding vector of S, which is a piece of text, and if S is null return **0**(zero vector); all the bold symbols mean that they represent vectors; Cos (V_1, V2) is the cosine of the angle between vector V1 and vector V2; M(V_1,V_2) is defined as follow:

$$M(V_1, V_2) = \begin{cases} 0 & V_1 \text{ or } V_2 \text{ is zero vector} \\ 1 & \text{others} \end{cases} \quad (6\text{-}5)$$

Using this algorithm, we calculate the similarity between the query and each result, and then the results can be ranked by the values of the similarities.

Ⅶ. Application and Experiment

We developed Fsearch prototype based on JAVA platform [11], utilized JAVA2 1.4 for Foundation Class development, JSP for web page implementing, JZKit (http://www.k-int.com) component and JAVA oracle component to interact with the library Z39.50 gateway and e-Learning resource

database respectively. Now Fsearch system has been tested on the OPAC database of XJTU Library's and e-Learning resource databases. The network model of Fsearch is shown in Fig. 2:

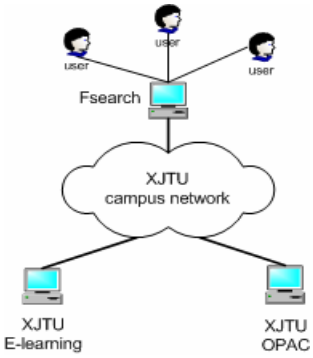

Fig. 2. Network Model of Fsearch

The Integrated Library System (ILS) of XJTU library is INNOPAC Millennium, which is a product of Innovative Interfaces Inc. (http://www.iii.com). The XJTU library holds the collections of over 3 million volumes (items) and over 14 thousand unique full-text electronic journals available online for users to access. The collections cover the fields of science, engineering, information technology, medicine, finance, economics and so on. The e-Learning materials management system of XJTU is ORACLE 9i, which contains about 180 multimedia courseware and 99 stream media courseware, total volume is more than 2000GB. Besides, there are also many materials, test papers, literatures and so on. The hardware information of Fsearch server is shown in Table 5:

Table 5. Experiment environment (*Fsearch* Server)

CPU	Piv 2.4G *2
Network	100M LAN connected in XJTU campus network
Operation System	windows 2003 server
Web Server	Resin2.0

Under this network model, we ran two experiments to test the ART (average response time) of the separate search on XJTU OPAC, separate search on XJTU e-Learning, and federal search on *Fsearch* with a computer interacted in XJTU campus network. The experiments are shown below:

Experiment1:

Query: Title=computer architecture
Results number of OPAC's separate search: 10;
Results number of e-Learning's separate search: 3;
Results number of *Fsearch*'s federal search: 13;

Table. 6 Result of Experiment 1

Concurrent users	1	5	15	50	100	300
OPAC ART (ms)	116	236	534	1408	2821	8157
e-Learning ART (ms)	200	351	434	1324	2810	8863
Fsearch ART (ms)	381	565	879	1980	5214	16621

Experiment2:

Query: Title=physics
Results number of OPAC's separate search: 164;
Results number of e-Learning's separate search: 10;
Results number of *Fsearch*'s federal search: 174;

Table. 7 Result of Experiment 2

Concurrent users	1	5	15	50	100	300
OPAC ART (ms)	170	282	566	1991	4360	14664
e-Learning ART (ms)	212	384	500	1546	3005	9512
Fsearch ART (ms)	406	602	1221	2515	6562	23853

The response time of *Fsearch* equals the maximum of OPAC response time and the e-Learning response time adding the join time. That is:

$$T_{Fsearch} = Max(T_{OPAC}, T_{E-learning}) + T_{join} \quad (7\text{-}1)$$

So, the response time could be reduced from the following facets:

1) Utilize a high-performance computer as the server, in order to reduce the computing time for result joining.
2) Optimize the code of result joining, especially the code of result ranking, which can also improve the system performance.
3) Reduce the transmission time on the network. Reduce the transmission length, improve the network bandwidth and use better switchers.

Besides, a professional web server can improve the stability and the performance of system. As the recommended concurrent user limit of Resin2.0 is 100, the response times in the experiments shown above increase rapidly when the concurrent users exceeding 100.

Ⅷ. Conclusion

Motivated by the integration and share of OPAC bibliography database and e-Learning resource warehouse, one possible solution to realize federal search on OPAC and e-Learning databases is proposed in this paper. We also discussed two key technologies, unified description of two resources based on XML and the join of two retrieved results, where we especially introduced our similarity calculation algorithm of a query and a result. The integration of these two kinds of resource is a vital but challengeable work, because few people do such research and developing as we know. In fact, these two kinds of resources have many inherent relations, especially in the e-Learning application. Now we have already developed a prototype of federal search system named *Fsearch*, which is based on J2EE component technology and J2EE/JSP three layers application frame, and *Fsearch* system has been tested on the heterogeneous and distributed resources of XJTU Library's OPAC and e-Learning material databases. The result shows that *Fsearch* can effectively improve the efficiency and quality of acquiring knowledge, especially it can provide users with one website searching service of crossing database.

However, the research and developing of *Fsearch* system has just started, many inherent relations or hyperlinks between OPAC database and e-Learning resource database haven't been constructed, of course, this workload is very heavy, next we will research on how to find their deep relations and then construct hyperlink automatically. Moreover, the research on how to join the two different structure results from two databases is still a problem because when the number of search result records is more than 80, the performance will drop dramatically.

References

[1] C. Boyer, V. Baujard, V. Griesser, J.R. Scherrer. HONselect: A Multilingual and Intelligent Search Tool Integrating Heterogeneous Web Resources. International Journal of Medical Informatics. 2001, 64: 253–258;

[2] Wang Lancheng, Ao Yi. Realization of Multi-Database Searching Agent of Digital Library. Journal of ShangHai Jiaotong University. 2003.9, Vol. 37 Sup, 188-194;

[3] Bob Gerrity. Theresa Lyman. Ed Tallent. Blurring Services and Resources: Boston College's Implementation of MetaLib and SFX. Reference Services Review. 2002， 30（3）: 229-241;
[4] MAP (Millennium Access Plus). http://www.iii.com/mill/digital.shtml#map;
[5] CALIS(China Academic Library & Information System), http://www.calis.edu.cn/;
[6] CELTS, http://www.edu.cn/html/keyanfz/yuanchengjiaoyu.shtml;
[7] http://www.ucc.ie/xml/;
[8] Xun Jiyuan, Yang CuiE, Yang Liming.: Implementation of the Extraction and Parsing of Database Information Based on J2EE and XML Technologies. Journal of EIC. 2004, 11(1):93-94;
[9] http://www.loc.gov/marc/marc2dc.html;
[10] Wang Juanqin.: Studies on Three Retrieval Models: Boolean Retrieve Model, Probability Retrieval Model, Vector Retrieval Model. Journal of Information science. 1998,16(3):225-230;
[11] Jameela Al-Jaroodi, Nader Mohamed, Hong Jiang, David Swanson.: Middleware Infrastructure for Parallel and Distributed Programming Models in Heterogeneous Systems. IEEE Transactions on Parallel and Distributed Systems. 2003, 14(11), 1100-1111;

Qinghua Zheng, Ph.D, was born in 1969, works in computer science department at Xi'an Jiaotong University as a mentor. Researches are mainly on Natural Language Processing, Network Security and theory of e-Learning.

Haifeng Dang, graduate student in Computer Science of Xi'an Jiaotong University; Researches are mainly on Information Retrieval, Natural Language Processing, and Text Mining.

Huixian Bai, graduate student in Computer Science of Xi'an Jiaotong University; Researches are mainly on Information Retrieval, Natural Language Processing, and Text Mining.

Jing Shao, Associate Professor, Deputy Director of Xi'an Jiaotong University Library; Researches are mainly on library science.