Fu, Y., Mostafa, J. (2004). Toward Information Retrieval Web Services for Digital Libraries. Proceedings of IEEE/ACM Joint Conference on Digital Libraries 2004, Tuscon, Arizona

Toward Information Retrieval Web Services for Digital Libraries

Yueyu Fu
Laboratory of Applied Informatics Research
Indiana University, Bloomington
IN, 47405-3907
(812)856-4182, 01
yufu@indiana.edu

Javed Mostafa
Laboratory of Applied Informatics Research
Indiana University, Bloomington
IN, 47405-3907
(812)856-4182, 01
jm@indiana.edu

ABSTRACT
Information retrieval (IR) functions serve a critical role in many digital library systems. There are numerous mature IR algorithms that have been implemented and it will be a waste of resources and time to re-implement them. The implemented IR algorithms can be distributed or their functions made available through the framework of web services. Web services in the IR domain have not been widely tested. Concept extraction is an important area in traditional IR. We demonstrated that it can be easily adopted as IR web services and can be accessed in multiple ways. For the IR web services, we take advantage of a term representation database which was created as a result of a previous digital library project containing 31,928,892 terms found on 49,602,191 pages of the web.

Categories and Subject Descriptors
H.3.5 [**Information Storage and Retrieval**]: Online Information Services – *web-based services.*

General Terms
Design, Experimentation

Keywords
Web Services, Information Retrieval

1. INTRODUCTION
The field of IR is approximately 50 years old and many techniques and operations have been developed in IR that do not require radical changes or re-implementation. A key idea behind web services is that frequently used functions can be implemented once and offered to other application or software environments through programmatic interfaces. Not many web services exist for IR even though several common IR functions can be potentially offered through web services. In this paper, we concentrated on concept extraction and weighting as key IR functions and demonstrate how these functions can be offered as web services. The paper will briefly outline the architecture of our system. It will describe how to access our web services in multiple ways. It will also describe how the key concepts are determined and extracted. The paper will conclude with a discussion of future work.

2. ARCHITECTURE

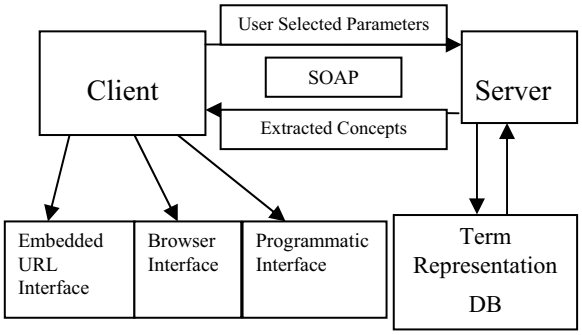

Figure 1. System Architecture

The system (see Figure 1) mainly consist of three components: the web service server, which is deployed in Apache Soap server; the client, which passes the user selected parameters to the server and retrieves the extracted concepts based on SOAP protocol, and web term representation database. We refer to our system as Library of User-Oriented Concepts for Access Services (LUCAS).

3. WEB SERVICES
The basic operation supported by LUCAS is the term extraction, term weighting, and ranking of the terms. LUCAS can generate results in different formats depending on parameters selected by clients. The related parameters are:

1) **docUrl** (required) - URL of the web page. The URL points to a web page from which the concepts will be extracted from.
2) **nTermMax** (required) - It specifies the number of terms which will be extracted from the top of the term list sorted in descending order based on the TF*IDF weights.
3) **Weights** (optional) **-** The weight parameter specifies if the user wants to retrieve term weight or not. For example, true or false.
4) **Format** (optional) **-** Three kinds of output formats are available: text, HTML and XML. Default is text.

There are multiple ways to invoke our web service. All of them are supported by the SOAP protocol. Based on our web service

description and the SOAP protocol, new client interfaces can be easily built according to users' preferences.

One way to access our web services is through a command-line programmatic interface. The following figure shows an example.

```
[yufu@tara strive]$ java ParseClient lair.indiana.edu 5 true text
Terms:
mostafa
86.24616043641782
jcdl
40.4131509661718 34
d-lib
35.43909112821328
large-scale
35.43909112821328
information
34.50069273428786
```

Figure 2. Programmatic interface

In the above example (see Figure 2), the command syntax contains the java class name of the client program "ParseClient", the URL "lair.indiana.edu", the number of terms "5", the weight parameter "true", and the output format "text". The result includes the extracted terms followed by their weights. Demo client code can be accessed at:
http://tara.slis.indiana.edu/~yufu/ce/ParseClient.java

Another possibility is by an embedded URL. For example:
http://tara.slis.indiana.edu:9080/soap/ce/Search.jsp?docUrl=www.jcdl2004.org&nTermMax=10&weights=true&format=text

URL Link: www.jcdl2004.org

Term	Google Links	Weight
digital	Google Link	84.44502184000616
jcdl	Google Link	80.82630193234367
library	Google Link	35.6986719514484
award	Google Link	24.397592864146688
committee	Google Link	23.12193450991064
conference	Google Link	22.53767642879499
organizing	Google Link	19.98040918357409
preservation	Google Link	19.50341646494154
poster	Google Link	18.37006469091709
first-served	Google Link	17.71954556410664

Figure 3. Access the service by a URL

In the above example (see Figure 3), the URL can point to any page on the web (top frame) and upon receiving a "click" from the user our web service can produce key concepts and weights and generate a web page with these results (bottom frame).

Finally, users can create a customized interface to LUCAS using HTML. Below we show an example. The HTML-based interface allows access to all parameters for retrieval through interactive means. The online demo is available at:
http://tara.slis.indiana.edu/~yufu/ce/ (see Figure 4).

CE - Concept Extraction Web Service

Web Page URL: www.jcdl2004.org
Maximum number of terms: 5 Weight: Format: Text
Go Reset

by Yueyu Fu and Javed Mostafa.

Term	Google Links	Weight
digital	Google Link	84.44502184000616
jcdl	Google Link	80.82630193234367
library	Google Link	35.6986719514484
award	Google Link	24.397592864146688
committee	Google Link	23.12193450991064

Figure 4. HTML-based user interface

4. CONCEPT EXTRACTION

The term representation database was generated by a previous digital library project at UC Berkeley and Stanford. It contains document frequency of 31,928,892 terms found on 49,602,191 pages of the Web (http://elib.cs.berkeley.edu/docfreq/). The terms in the database were used to refine and re-weight terms extracted from client pages to make the results more representative in relation to the overall distribution of terms in the web.

Concept extraction algorithm is a mature procedure in IR. It was implemented as a web service. The service executes the following steps:

1) Retrieve a web page by its URL and parse its content.
2) Extract all the terms except the stop words and determine term frequency.
3) Retrieve the IDF values from the term representation database for all the extracted terms and compute TF*IDF weights. The IDF values are generated from the term representation database.
4) Sort the terms based on the TF*IDF values and select the top terms specified by the user.

5. CONCLUSION

Our system demonstrated that mature IR algorithms can be successfully turned into web services and can be accessed in multiple ways. A future direction of the project is to process documents in other common formats such as PDF, PS, and MSWORD. The efficiency and effectiveness of the services will also have to be carefully measured. Web services such as these have great potential for supporting digital library operations.

6. ACKNOWLEDGMENTS

This work was partially supported through a grant from the National Science Foundation Award#:0333623.

7. REFERENCES

[1] Chen, H. Semantic Research for Digital Libraries. *D-Lib Magazine*, 5(10). 1999.

[2] Tilley, S., Gerdes, J., Hamilton, T., and etc. Adoption Challenges in Migrating to Web Services. *Proceedings of the Fourth International Workshop on Web Site Evolution*. 2002.

[3] Truner, M., Budgen, D., and Brereton, P. Turning Software into a Service. *IEEE Computer*, 36(10), 2003.

[4] Web Term Document Frequency and Rank <http://elib.cs.berkeley.edu/docfreq/>