Toward a Distributed Terabyte Text Retrieval System in China-US Million Book Digital Library

Bin Liu Wen Gao Ling Zhang
Digital Media Research Center
Institute of Computing Technology
Chinese Academy of Sciences
Beijing 100080, P.R.China
{bliu, wgao, lzhang} @ict.ac.cn

Tie-jun Huang Xiao-ming Zhang
Digital Media Research Center
Graduate School of
Chinese Academy of Sciences
Beijing 100080, P.R.China
{tjhuang, xmzhang} @gscas.ac.cn

Jun Cheng
The Library of
Chinese Academy of Sciences
Beijing 100080, P.R.China
jcheng@gscas.ac.cn

ABSTRACT

In China-US Million Book Digital Library, output of the digitalization process is more than one terabyte of text in OEB and PDF format. To access these data quickly and accurately, we are developing a distributed terabyte text retrieval system. With the query cache, system can search less data while maintaining acceptable retrieval accuracy. From the OEB package, we get its metadata and structural information to implement multi-scale indexing and retrieval. We are to explore some new retrieval models and text clustering approaches in the Digital Library.

Categories and Subject Descriptors

H.3.7 [Information Storage and Retrieval] Information Search and Retrieval - Retrieval models, Search process; Digital Libraries – Standards, Systems issues

General Terms: Algorithms, Design

Keywords: Terabyte text retrieval, query caching, Open eBook

1. INTRODUCTION

As the content in the Internet and intranets grows, traditional full-text search engines cannot cope with the increasing data scale. Therefore, many research and commercial distributed databases and IR systems come out. Hawking[1] used multiprocessor systems to response a single query less than a second on 100GB collection. Zhihong Lu[2] used simulation to experiment with a terabyte of text data, giving some promising performance of partial replication distributed IR system. TREC (Text Retrieval Conference) also provides the VLC2 (Very Large Collections version 2) that contains about 100GB text data for study.

In China-US Million Book Digital Library, output of the digitalization process is more than one terabyte of text in OEB (Open eBook) format[3] for browsing, and the PDF format files for downloading. To provide information and knowledge service to any people at any time in any place, it is important to access any text data in the library quickly and accurately, with a terabyte text retrieval system.

The remainder of this paper is organized as follows. Section 2 describes the system framework and the query cache mechanism. Section 3 shows our approach in retrieving OEB documents. Section 4 gives the conclusions and our future work.

JCDL'02, July 13-17, 2002, Portland, Oregon, USA.

2. DISTRIBUTED FRAMEWORK

The system has millions of books in OEB format, which contains numerous pages in XML format. These files and PDF files are stored in the source library such as disk array. Fig.1 shows the framework of the distributed text retrieval system.

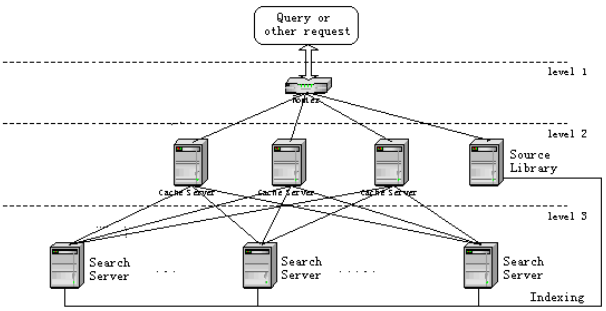

Fig.1 Framework of the distributed text retrieval system

Although we use Gigabit Ethernet to transmit data, it cannot keep up with increasing users and expanding book collections. Distributing excessive workloads and searching as little data as possible while maintaining acceptable retrieval accuracy are the two ways to improve the performance and scalability of a distributed IR system. For data distributing, we put the indexes of different books on different search servers. Each search server can deal with about tenths of gigabyte text data with a full-text search engine. To search less data, we built a query cache, which contains query logic as well as a subset of the documents or their summaries.

Caching is based on query and document locality, repeated access to the same part of the text database. Recent research results[4,5] indicate the locality in search engine queries, based on which we build query cache. To cache enough query for the service, we use three cache servers, each covers part of the query cache. The router in level 1 balances the workloads of the cache servers in level 2. It also keeps the system from attacking. When the user sends a query, the router check which cache server can provide the service and forward it to the proper one. If the query is not in the query cache, the router forwards it to an idle server in rolling means. Then the cache server sends searching commands to each search server in level 3. The cache server merges the result from search servers, ranks them by the relevance, stores the new query in the cache and returns the search result to the user.

3. RETRIEVING OEB DOCUMENTS

OEB documents must be valid XML documents conforming to OEB package DTD and document DTD. An informal outline of the package is as follows:

<?xml version="1.0"?>

<!DOCTYPE package PUBLIC "+//ISBN 0-9673008-1-9//DTD OEB 1.0.1 Package//EN" "http://openebook.org/dtds/oeb-1.0.1/oebpkg101.dtd">

<package>

metadata | *manifest* | *spine* | tours | *guide*

</package>

The major parts of the OEB package file are:

Package Identity—A unique identifier for the OEB publication as a whole.

Metadata—Publication metadata (title, author, publisher, etc.).

Manifest—A list of files (documents, images, style sheets, etc.) that make up the publication. The manifest also includes fallback declarations for files of types not supported by this specification.

Spine—An arrangement of documents providing a linear reading order.

Tours—A set of alternate reading sequences through the publication, such as selective views for various reading purposes, reader expertise levels, etc.

Guide—A set of references to fundamental structural features of the publication, such as table of contents, foreword, bibliography, etc.

From the OEB package, we get publication metadata that conforms to the Dublin Core and structural information such as title, abstract and content, which are indexed for retrieval. We provide two retrieval means: book retrieval and page retrieval. The classified displaying tree in Fig.2 is an example of book retrieval in the area of computer science.

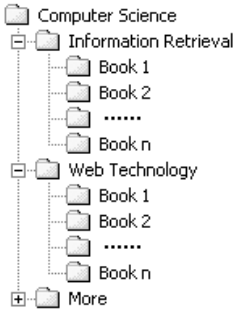

Fig.2 The result of book retrieval

Books ranking is based on their relevance to the query. When clicking one of the books, the user can browse it, as is shown in Fig.3. The left column shows the outline of an OEB book, including metadata, content and other supplementary information. When the user clicks one of the items, its content will be shown in the right column.

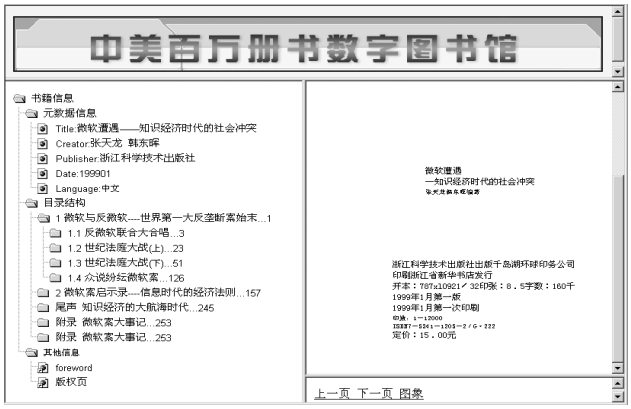

Fig.3 Browsing a book

Fig.4 shows the two results in page retrieval after inputting "Intelligent Information Retrieval" in Chinese. Users can perform further search in the result.

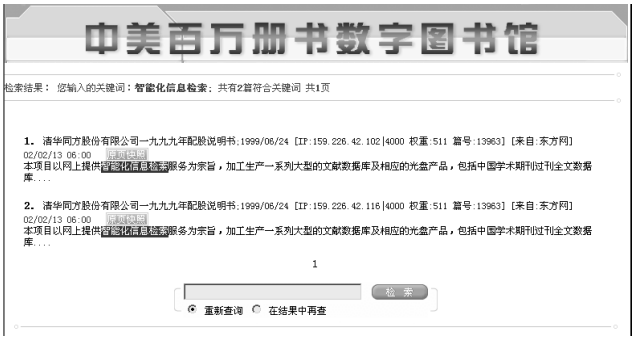

Fig.4 The result of page retrieval

4. CONCLUSIONS AND FUTURE WORK

A terabyte text retrieval system in China-US Million Book Digital Library should meet both functions and performance needs. This article presents our work on query caching in distributed framework and XML-based indexing that supports multi-scale OEB retrieval. We are to explore some new retrieval models and text clustering approaches in the Digital Library. The system is available at http://159.226.42.116/isearch.

5. REFERENCES

[1] Peter Bailey, David Hawking, A Parallel Architecture for Query Processing Over A Terabyte of Text. Australian National University, Technical Report, 1996

[2] Zhihong Lu, Scalable Distributed Architectures for Information Retrieval. University of Massachusetts at Amherst, Ph.D. dissertation, 1999

[3] Open eBook Publication Structure Specification 1.0.1. http://openebook.org/oebps/oebps1.0.1/download

[4] Evangelos P. Markatos, On Caching Search Engine Query Results. Computer Communications, 24(2), Jan 2001, pp137-143

[5] Yinglian Xie, David O'Hallaron, Locality in Search Engine Queries and Its Implications for Caching. Carnegie Mellon University, Technical Report, 2001