Chapter X

Structure- and Content-Based Retrieval for XML Documents

Jae-Woo Chang, Seoul National University of Korea
Du-Seok Jin, Chonbuk National University of Korea

ABSTRACT

As the number of XML documents is dramatically increasing, it is necessary to develop an XML document retrieval system that can support both structure-based retrieval and content-based retrieval. In order to support the structure-based retrieval, we design four efficient index structures, i.e., keyword, structure, element and attribute index, by indexing XML documents based on a basic element unit. In order to support the content-based retrieval, we design a high-dimensional index structure based on the X-tree so as to store and retrieve both color and shape feature vectors efficiently. Finally, we do the performance evaluation of our XML document retrieval system in terms of system efficiency, such as retrieval time, insertion time, and storage overhead, as well as system effectiveness, such as recall and precision measures.

INTRODUCTION

The XML (eXtensible Markup Language) was proposed as a standard markup language to make Web documents in 1996[W3C, 2000]. It has as good expressive power as SGML and is also easy to use like HTML. Recently, it has been common for users to acquire through the Web a variety of multimedia documents written by XML. Meanwhile, because the number of XML documents is dramatically increasing, it is difficult to reach a specific XML document required by users. Moreover, an XML document not only has a logical and hierarchical structure commonly, but also contains its multimedia data, such as image and video. Thus, it is necessary to develop an XML document retrieval system that can support both the retrieval based on document structure and the retrieval based on image content.

In general, since the conventional XML document retrieval systems support only structure-based retrieval, it is impossible to deal with a user query which requires both structure- and content-based retrieval for XML document. In this chapter, we design and implement an XML document retrieval system that can efficiently retrieve XML documents based on both document structure and image content. In order to support the structure-based retrieval, we design four efficient index structures, i.e., keyword, structure, element and attribute index, by indexing XML documents based on a basic element unit and implement
them by using the o2store storage system. For supporting the content-based retrieval, we
design a high-dimensional index structure based on the X-tree so as to store and retrieve both
color and shape feature vectors efficiently.

This chapter is organized as follows. We introduce related works in the area of
structure-based and content-based document retrieval systems. We design an XML docu-
ment retrieval system supporting structure-based and content-based retrieval. We show the
interface of our XML document retrieval system. In Section 5, we present the performance
analysis of our system. Finally, we draw conclusions and provide some issues for future
research.

**RELATED WORK**

**Structure-Based Retrieval**

Because an element is a basic unit that constitutes a structured document (i.e., SGML
or XML document), it is essential to support not only retrieval based on element units but
also retrieval based on logical inclusion relationships among elements. Since there are a lot
of studies on SGML documents, we, in this section, describe some related work on the
representation of SGML document structures. First, RMIT in Australia proposed five query
types for structure-based retrieval that should be supported in SGML information retrieval
(Sacks-Davis, Arnold-Moore and Zobel, 1994). Most of the types consist of retrieval on
upper-level elements (e.g., parent element), or on lower-level elements (e.g., child ele-
ments) from a given element. For supporting the five types of queries, RMIT proposed a
subtree model which indexes all the elements in a SGML document and stores all the terms
which are appeared in the elements (Lowe, Zobel and Sacks-Davis, 1995). Although the
model supports efficient retrieval on a specific query, it has disadvantages of long indexing
time and high storage overhead because index information should be repeatedly stored
according to a tree depth. Secondly, RMIT proposed a SCL structure that extends the GCL
structure (Dao and Sacks-Davis, 1996). After assigning numbers to both terms and markups
in SGML documents, they use the SCL structure to store term interval, markups and
inclusion relationships among elements. The SCL structure has an advantage that it can
handle graph-structured documents, but it has two disadvantages that it requires a deletion
operation and it cannot represent the depth of the elements effectively. Finally, SERI in
South Korea proposed a K-ary Complete Tree Structure which represents a document as a
K-ary complete tree (Han, Son, Chang and Zhoo, 1999). In this method, each element
corresponds to a node in a K-ary tree. Therefore, a relationship between two elements can
be acquired by calculation. This method has an advantage that it is fast to find an element
including a given logical relation by calculation. But, as the depth of a K-ary tree is deeper,
the number of nodes is increasing exponentially with a large number of unused nodes. In the
cases of partial insert and deletion, almost all of nodes should be changed in their assigned
number.

**Content-Based Retrieval**

There have been many researches on content-based retrieval techniques in multimedia
DBMSs. The key issues of the studies include keyword extraction for text-based retrieval,
image-processing techniques used for feature extraction of images, and multi-dimensional
indexing techniques for fast retrieval, and content-based image retrieval based on color
histogram, texture, and shape. First, QBIC (Query By Image Content) project (Niblack,
A Unicode Keyboard for African Languages: The Case of Dagaare and Twi
*International Journal of Technology and Human Interaction* (pp. 1-20).
[www.igi-global.com/article/unicode-keyboard-african-languages/2875?camid=4v1a](www.igi-global.com/article/unicode-keyboard-african-languages/2875?camid=4v1a)