

1. Introduction

1.1 Background

LiMaCS (Library Information Monitoring and Control System) is proposed as library system model for the future. The target of this project aims at building a multi-media, distributed heterogeneous database for providing, monitoring and controlling library information over the wide-area-network.

In general, there are multiple libraries located in different areas of a region. It will be convenient to the clients if they can browse the catalogues of all these libraries in a central location instead of traveling to each of them. LiMaCS aims to provide these features. It stores all catalogues of libraries in a centralized database. Clients can browse the centralized catalogue over the Internet.

1.2 Approach of the thesis

The centralized database collects all documents for all local libraries within a region. These catalogues is a huge database. In order to provide an efficient mechanism for storing and searching, designing a good and suitable index structure and compressing this huge database are necessary.

We have studied several compression, indexing and searching approaches. For the compression, Huffman coding, arithmetic and LZ77 are analyzed. Huffman coding and arithmetic coding are two-pass methods. They work by estimating the probabilities of symbols appeared in the file, using shorter codewords for high-probability symbols and longer codewords for low-probability symbols. LZ77 is one-pass method. It uses a pointer to replace a string; this pointer refers to where the string has appeared previously.

Indexing method of inverted files, signature files and bitmaps are studied. The inverted file is the most natural indexing method, it is similar to the index of a book. An inverted file contains a list of terms and corresponding pointers which point to the position of the main text where the term appears. Signature file is a probabilistic method for indexing text. Each text file contains an associated descriptor, where several hash values are generated by every indexed term. Then the bits of the descriptor corresponding to those hash values are set to one. Bitmaps is a special case of inverted files. The lexicon of bitmaps contains a list of terms and their bitvector in which each bit indicates one document. If the term appears in the text file, the corresponding bit is set to one, otherwise it is set to zero.

Two Searching method which relates to the database searching are studied. Boolean query is a list of terms which are connected with one or more logical operator. The lexicon is searched for each term in the Boolean query, and all related inverted file entries are retrieved. These entries are merged together and taken set operation. Ranked query retrieves a set of documents with respect to the query, and sort this set of documents by decreasing order of frequency. Then the top of the ranked documents in terms of the most relevant to the query are retrieved as the list of answer.

In order to minimize the storage, the centralized database should be compressed. The static compression method which is discussed in chapter 4 can not be applied directly to the database which is changed frequently. Therefore, a dynamic compression method should be developed. For these reasons, we need to improve the static method to a dynamic method, and implement a prototype.

Among several approaches for compression, Huffman coding is the one and only one method which is improved to compress database. Arithmetic coding is not easy to start decoding in the middle of the text. However, decoding individual bibliographic record should be require for the bibliographic file. The various LZ77 models represent a string by reference to a previous appearing of the string; but a few reference of the string can be found within a bibliographic record. Huffman coding can start decoding in middle of the text, and it is efficient than LZ77 where only a few reference can be found for encoding.

1.3 Outline of this thesis

Chapter 2 analyses the domain in a library system. Three main components – library, client and document provider – , functions and operations are described. Document is refereed to all categories which stored inside the library. This term is often used in this thesis.

Chapter 3 overviews the LiMaCS in database aspect. It defines a collection of library components and their interfaces, and the system requirement. The entities and relationship are derived from these components and requirement.

Chapter 4 describes the existing methods about text compression, indexing and searching. Text compression uses codewords to represent the source text (character or word or string) by using a shorter one (codeword) to replace a longer one (word). Therefore, the required disk space for storing information is reduced. An index is used for retrieving information from a long text or a database. It can provide fast access to a given term in a text/database. A search is to retrieve all documents in which there are one or more words matched the given search condition from a text/database.

Compression and indexing techniques are chosen and improved for implementing the LiMaCS, and are described in Chapter 5. These techniques are selected from various alternatives which are discussed in chapter 4. The improved compression method and the evaluation of implementing a prototype of compression are described. This compression method which we call *extendible word-based Huffman coding* is developed based on the static character-based Huffman coding. For the performance concern, we use the raw file system to store the huge file.

In chapter 6, we summarize our contribution and discuss topics for limitations and future research.