ABSTRACT

This thesis involves comprehensive study and implementation of text compression techniques useful for direct searching the phrases in compressed form.

In the initial part of this thesis, we summarize our comprehensive study of different types of compression methods including Arithmetic Coding method, Bzip2, Prediction by Partial Match and Lempel-Ziv Markov-chain Algorithm.

In subsequent part, two categories of text compression techniques are implemented with an objective of improved compression ratio and optimized for searching and retrieval of strings randomly from compressed file.

We implement text compression techniques using three different types of dictionaries viz. static, semi-dynamic and dynamic. We also study the string-matching algorithms such as Karp-Rabin, Knuth-Morris-Pratt, Brute-Force, Boyer-Moore and Quick-Search Algorithms.

Major contribution of the thesis is to propose pre-text compression technique Word based Text Compression Technique using semi-dynamic dictionary (WBTC-C). This method gives a better compression ratio when used as a pre-stage compression to standards methods such as Bzip2, PPMd, PPMII and LZMA, and is also useful for searching the strings directly from the compressed files. The decompression time is also improved in WBTC-C method as compared to Bzip2 and PPMd. Other methods such as CBTC-A, CBTC-B, WBTC-A, WBTC-B, WBTC-D and WBTC-E are also implemented, which differ from WBTC-C. Those methods are implemented using single dimension dictionary, double dimension dictionary and using static dictionary and dynamic dictionary.

The techniques are useful for direct searching the pattern in the compressed form. The text compression techniques implemented by us uses single and double dimension dictionary. The text compression techniques are used as pre-stage compression to existing standard methods such as Arithmetic Coding, Bzip2, PPMd, PPMII and LZMA. The compression ratio is improved when our techniques are used as pre-stage to those methods.

All techniques are implemented in VC++ 6.0 version.