Data Compression Using Text Encryption

Holger Kruse   Amar Mukherjee
kruse@cs.ucf.edu  amar@cs.ucf.edu

Department of Computer Science
University of Central Florida
Orlando, FL 32816

Abstract

In this paper we discuss the use of a new algorithm to preprocess text in order to improve the compression ratio of textual documents, in particular online documents such as web pages on the World Wide Web. The algorithm was first introduced in an earlier paper, and in this paper we discuss the applicability of our algorithm in Internet and Intranet environments, and present additional performance measurements regarding compression ratios, memory requirements and run time. Our results show that our preprocessing algorithm usually leads to a significantly improved compression ratio.

Our algorithm requires a static dictionary shared by the compressor and the decompressor. The basic idea of the algorithm is to define a unique encryption or signature for each word in the dictionary, and to replace each word in the input text by its signature. Each signature consists mostly of the special character '*' plus as many alphabetic characters as necessary to make the signature unique among all words of the same length in the dictionary.

In the resulting cryptic text the most frequently used character is typically the '*' character, and standard compression algorithms like LZW applied to the cryptic text can exploit this redundancy in order to achieve better compression ratios.

We compared the performance of our algorithm to other text compression algorithms, including standard compression algorithms such as gzip, Unix 'compress' and PPM, and to one text compression algorithm which uses a static dictionary. The compression ratio of our method combined with a standard compression algorithm is typically 5% to 20% higher than that of the same standard compression algorithm alone.

In this paper we also describe how our algorithm can be efficiently used to aid in the compression of large texts for transmission across public data networks, in particular how our scheme can interoperate with standards such as HTTP and HTML on the World Wide Web.

Since the static dictionary needs to be shared by the compressor and decompressor, the cost for the dictionary is amortized over the compression savings. In our paper we also describe ways how dictionaries can be uniquely identified and maintained in Internet and Intranet environments.