

ABSTRAK

PERBANDINGAN ALGORITME PEMAMPATAN DATA TEKS : ALGORITME HUFFMAN DAN ALGORITME LEMPEL ZIV

Pemampatan data teks adalah suatu teknik untuk mentransformasi berkas simbol ke berkas sandi yang ukurannya lebih kecil daripada ukuran berkas simbol. Secara garis besar algoritme pemampatan data teks dapat dibagi dalam dua kelompok pemodelan, yaitu pemodelan berbasis *statistik* dan pemodelan berbasis *kamus*. Pemodelan berbasis statistik melakukan pengumpulan data – data statistik sebelum pemampatan dilakukan, sedangkan pemodelan kamus menyandikan panjang untai yang bervariasi dari simbol sebagai *token* tunggal. Terdapat dua algoritme pemampatan yang populer dan banyak mendasari terbentuknya algoritme pemampatan yang ada saat ini, yaitu algoritme Huffman dan algoritme Lempel Ziv. Algoritme Huffman menggunakan pemodelan *statistik* sedangkan algoritme Lempel Ziv pemodelan kamus.

Analisis dan perbandingan terhadap dua algoritme pemampatan tersebut dilakukan terhadap kompleksitas waktu pada keadaan terburuk, tingkat pertumbuhan algoritme, rasio pemampatan, serta kecepatan program dalam hal pemampatan dan pemekaran data. Dari analisis yang dilakukan disimpulkan bahwa algoritme Huffman mempunyai kompleksitas waktu $T(n) = n \lg n$, sedangkan algoritme Lempel Ziv Storer Symanski (LZSS) dan Algoritme Lempel Ziv Welch (LZW) mempunyai kompleksitas waktu $T(n) = n$. Dapat dibuktikan bahwa tingkat pertumbuhan ketiga algoritme tersebut adalah sama dengan kompleksitas waktunya. Rasio pemampatan terbaik dicapai oleh algoritme LZSS, kecepatan pemampatan terbaik dicapai oleh LZW, sedangkan kecepatan pemekaran terbaik dicapai oleh LZSS.

ABSTRACT

Comparison of Text Data Compression Algorithms: The Huffman Algorithm and The Lempel Ziv Algorithm

Text data compression is a technique to transfer symbol files into codeword files, which are smaller than the symbol files in size. Basically text data compression algorithms can be classified into two modeling groups, namely statistical-based modeling and dictionary-based modeling. Statistical-based modeling gathers data before doing the compression, while dictionary-based modeling encodes various string lengths of symbol as a single token. There are two compression algorithms which are popular and serve as the basis of the formation of existing compression algorithms, namely Huffman algorithm and Lempel Ziv algorithm. Huffman algorithm uses statistical – based modeling, whereas Lempel Ziv algorithm dictionary-based modeling.

Analysis and comparison between the two text data compression algorithms were done with respect to time complexity on the worst case, algorithm growth rate, compression ratio, and time efficiency of the program in terms of compression and decompression speed. From the analysis results it can be concluded that the time complexity of Huffman algorithm is $T(n) = n \lg n$, while the time complexity of Lempel Ziv Storer Symanski (LZSS) algorithm and Lempel Ziv Welch (LZW) algorithm are both $T(n) = n$. It can be proved that the growth rate of the three algorithms are the same as the time complexity. The best compression ratio was reached by LZSS, the best compression speed was reached by LZW, while the best decompression speed was reached by LZSS.