SNESTIK

Seminar Nasional Teknik Elektro, Sistem Informasi, dan Teknik Informatika

https://ejurnal.itats.ac.id/snestik dan https://snestik.itats.ac.id

Informasi Pelaksanaan :
SNESTIK V - Surabaya, 26 April 2025
Fakultas Teknik Elektro dan Teknologi Informasi, Institut Teknologi Adhi Tama Surabaya

Informasi Artikel:
DOI : 10.31284/p.snestik.2025.7676

Prosiding ISSN 2775-5126

Fakultas Teknik Elektro dan Teknologi Informasi-Institut Teknologi Adhi Tama Surabaya
Gedung A-ITATS, Jl. Arief Rachman Hakim 100 Surabaya 60117 Telp. (031) 5945043
Email : snestik@itats.ac.id

Perbandingan Algoritma Shannon-Fano dan Lempel Ziv Welch (LZW) untuk Kompresi Data Teks

Azhar Adi Dirgantara Setiawan, Citra Nurina Prabiantissa*

Institut Teknologi Adhi Tama Surabaya
e-mail:citranurina@itats.ac.id

ABSTRACT

In today's digital era, the volume of data generated and stored is increasing, so that efficiency in data management and delivery is very crucial. Optimal data management not only supports personal needs, but also supports activities in various sectors, from business to research. Therefore, a method is needed that can reduce the size of data without sacrificing important information contained therein. The data compression process is becoming increasingly important in the management and storage of digital information, especially in optimizing the use of storage space and accelerating data delivery. This study aims to compare the efficiency of the Shannon-Fano and Lempel Ziv Welch (LZW) algorithms in compressing text data. Both algorithms are analyzed based on parameters such as Compression Ratio (CR) and Redundancy (RD) to evaluate the effectiveness of the resulting compression. The implementation of the algorithm is carried out through encoding and decompressing text data, taking into account the frequency distribution of characters in the data. From the results of the tests carried out, the LZW algorithm proved to be superior to Shannon-Fano in compressing TXT files. LZW consistently produces compression ratios of 81 - 95% and lower RDs between 81 – 95%, especially for texts with complex repeating patterns..

Keywords*: Data Compression, Shannon-Fano, LZW, Compression Ratio, Algorithm Efficiency*

ABSTRAK

Di era digital saat ini, volume data yang dihasilkan dan disimpan semakin meningkat, sehingga efisiensi dalam pengelolaan dan pengiriman data menjadi hal yang sangat krusial. Pengelolaan data yang optimal tidak hanya mendukung kebutuhan pribadi, tetapi juga menunjang aktivitas di berbagai sektor, mulai dari bisnis hingga penelitian. Oleh karena itu, diperlukan suatu metode yang mampu mengurangi ukuran data tanpa mengorbankan informasi penting yang terkandung di dalamnya.Proses kompresi data menjadi semakin penting dalam pengelolaan dan penyimpanan informasi digital, terutama dalam mengoptimalkan penggunaan ruang penyimpanan dan mempercepat pengiriman data. Penelitian ini bertujuan untuk membandingkan efisiensi algoritma Shannon-Fano dan Lempel Ziv Welch (LZW) dalam mengompresi data teks. Kedua algoritma ini dianalisis berdasarkan parameter seperti Compression Ratio (CR) dan Redundancy (RD) untuk mengevaluasi efektivitas kompresi yang dihasilkan. Implementasi algoritma dilakukan melalui pengkodean dan dekompresi data teks, dengan memperhatikan distribusi frekuensi karakter dalam data. Dari hasil pengujian yang dilakukan Algoritma LZW terbukti lebih unggul dibandingkan Shannon-Fano dalam mengompresi file TXT. LZW secara konsisten menghasilkan compression ration sebanyak 81 - 95% dan RD yang lebih rendah antara 81 – 95%, terutama untuk teks dengan pola berulang yang kompleks.
Kata Kunci: Kompresi Data, Shannon-Fano, LZW, Rasio Kompresi, Efisiensi Algoritma

PENDAHULUAN

Di era digital saat ini, volume data yang dihasilkan dan disimpan semakin meningkat, sehingga efisiensi dalam pengelolaan dan pengiriman data menjadi hal yang sangat krusial [1]. Pengelolaan data yang optimal tidak hanya mendukung kebutuhan pribadi, tetapi juga menunjang aktivitas di berbagai sektor, mulai dari bisnis hingga penelitian. Oleh karena itu, diperlukan suatu metode yang mampu mengurangi ukuran data tanpa mengorbankan informasi penting yang terkandung di dalamnya.

Teknik kompresi data telah muncul sebagai solusi efektif untuk mengatasi permasalahan keterbatasan kapasitas media penyimpanan dan kecepatan transfer data. Proses kompresi memungkinkan data diubah menjadi bentuk yang lebih ringkas, sehingga mengurangi ruang penyimpanan yang diperlukan dan mempercepat proses transmisi [2]. Dua algoritma kompresi yang cukup populer dan telah banyak diteliti adalah algoritma Shannon-Fano dan Lempel Ziv Welch (LZW).

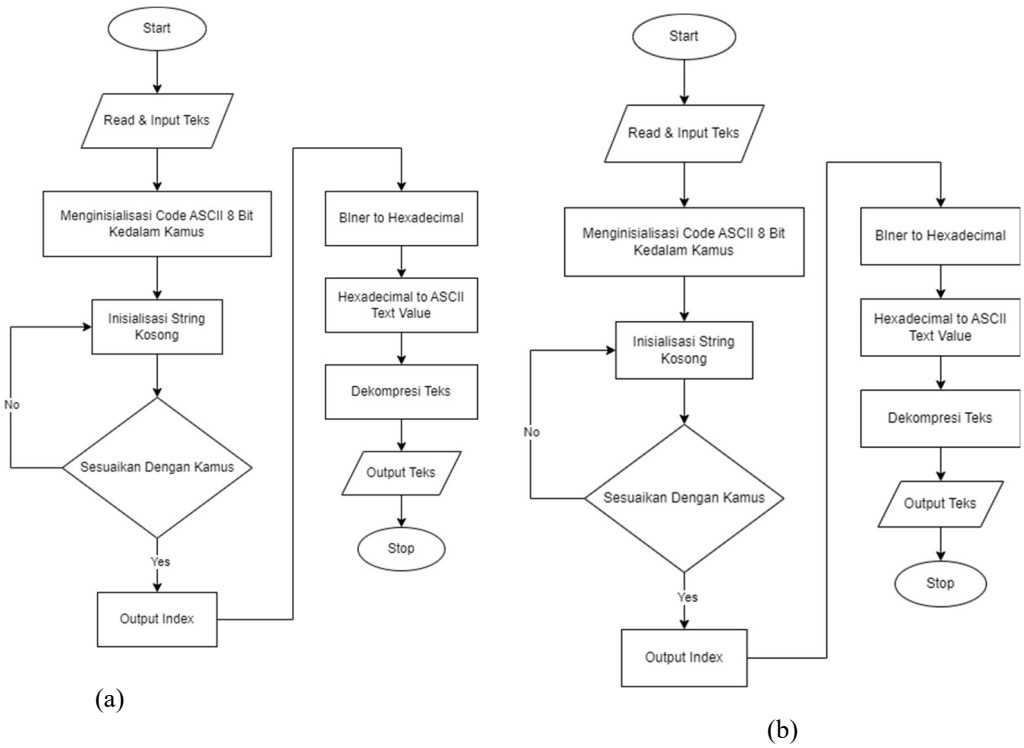
Algoritma Shannon-Fano bekerja dengan prinsip pengkodean variabel, di mana karakter-karakter yang sering muncul diberikan kode yang lebih pendek dibandingkan dengan karakter yang jarang muncul. Pendekatan ini memungkinkan efisiensi pengkodean pada data dengan distribusi frekuensi yang tidak merata [1]. Sementara itu, algoritma LZW memanfaatkan struktur kamus untuk menggantikan segmen data yang berulang dengan indeks yang lebih ringkas, sehingga menghasilkan proses kompresi yang adaptif dan cepat [2]. Perbandingan antara kedua algoritma ini menunjukkan bahwa keunggulan masing-masing sangat bergantung pada karakteristik data yang akan dikompresi [3], [4].

Penelitian ini bertujuan untuk menganalisis efektivitas kompresi data teks dengan mengimplementasikan algoritma Shannon-Fano dan LZW. Analisis akan difokuskan pada pengukuran parameter kinerja seperti Compression Ratio (CR) dan Redundancy (Rd) untuk memperoleh pemahaman mendalam mengenai kelebihan dan keterbatasan masing-masing algoritma. Hasil penelitian diharapkan dapat memberikan kontribusi dalam pemilihan metode kompresi yang paling sesuai untuk berbagai aplikasi pengolahan data di era digital [3].

METODE

Penelitian ini bertujuan untuk mengevaluasi efisiensi kompresi yang dihasilkan oleh kedua algoritma tersebut ketika diterapkan pada data teks yang sama [5]. Fokus utamanya adalah untuk menentukan algoritma mana yang lebih efektif dalam mengurangi ukuran file. Pada tahap

awal, kedua algoritma diuji untuk mengompresi data teks secara terpisah, dan hasil kompresi dibandingkan untuk menilai keefektifannya . Selanjutnya, dilakukan evaluasi terhadap kemampuan kedua algoritma dalam mendekompresi data yang telah terkompresi untuk melihat keefektifan kompresi dan dekompresinya [6]. Gambar 1 berikut ini merupakan flowchart perbandingan algoritma shannon – fano dan LZW:



Gambar 1. (a) Flowchart Algoritma Shannon Fano, (b) Flowchart Algoritma LZW

Pengujian algoritma Shannon-Fano dan LZW akan dievaluasi berdasarkan *Ratio Compression* (RC), *Compression Ratio* (CR), dan *Redundancy Compression* (RD) untuk menentukan efisiensi kompresi data teks, dengan masing-masing algoritma memiliki metode dan karakteristik performa yang unik.

Tahap awal kompresi data teks melibatkan konversi setiap karakter menjadi kode biner 8-bit menggunakan *encoding ASCII*, menghasilkan urutan bit unik seperti "D" yang direpresentasikan sebagai "1000100", yang kemudian digunakan oleh algoritma kompresi seperti Shannon-Fano [7]. Tabel 1 berikut ini merupakan penerapan konversi karakter ke binary code:

Tabel 1. Konversi karakter ke Binary Code

Huruf	Kode Binary	Ukuran (Bit)
D	1000100	8 bit
I	1001001	8 bit
R	1010010	8 bit
G	1000111	8 bit
A	1000001	8 bit
N	1001110	8 bit

T	1010100	8 bit
A	1000001	8 bit
R	1010010	8 bit
A	1000001	8 bit

Kompresi Shannon-Fano

Pada algoritma Shannon – Fano, kode biner setiap karakter ditentukan oleh susunan simpul daun dalam pohon *Huffman*. Untuk menghitung total jumlah bit yang dibutuhkan, jumlah bit untuk setiap karakter dikalikan dengan frekuensi kemunculannya [8]. Misalnya, karakter A dengan kode 2 bit (00) dan frekuensi 3 akan membutuhkan 6 bit (2 x 3). Perhitungan jumlah bit menggunakan Shannon - Fano dapat dilihat pada tabel 2:

Tabel 2. Penghitungan Jumlah Bit menggunakan Shannon Fano

Huruf	Frekuensi Huruf	Kode	Jumlah Bit
D	1	1000	4*1=4
I	1	101	3*1=3
R	2	01	2*2=4
G	1	1001	4*1=4
A	3	00	2*3-6
N	1	110	3*1=3
T	1	111	3*1=3
Total	**10**		**27**

Kompresi Lempel Ziv Welch

Algoritma Kompresi LZW merupakan teknik yang sangat efisien untuk mengurangi jumlah data dengan memanfaatkan pola yang sering muncul dalam data teks [9]. Setelah prosedur kompresi selesai, hasilnya adalah serangkaian kode yang mewakili teks yang telah dikompresi. Tabel 3 menujukkan perubahan huruf menjadi kode biner menggunakan kompresi algoritma LZW:

Tabel 3. Kompresi LZW

Huruf	Frekuensi Huruf	Kode
T	84	01010100
H	72	01001000
I	73	01001001
S	83	01010011
IS	258	100000010
TH	256	100000000
E	69	01000101
Total		**58 bit**

Hasil kompresi menggunakan algoritma *LZW* menunjukkan bahwa setiap karakter dan kombinasi karakter diberikan kode unik, seperti "T" dengan kode 84 (biner 01010100) dan "H" dengan kode 72 (biner 01001000). Kombinasi "IS" diberi kode 258 (biner 100000010), menggunakan kode 9-bit yang lebih panjang dibandingkan karakter tunggal. Sebelum kompresi, teks memerlukan 72 bit, sementara setelah kompresi, jumlah bit berkurang menjadi 58 bit.

Compression Ratio (CR)

Compression Ratio merupakan persentase perbandingan antara ukuran data setelah dikompresi dengan ukuran data sebelum dikompresi[10]. Persamaan 1 berikut ini merupakan perhitungan menggunakan compression ratio:

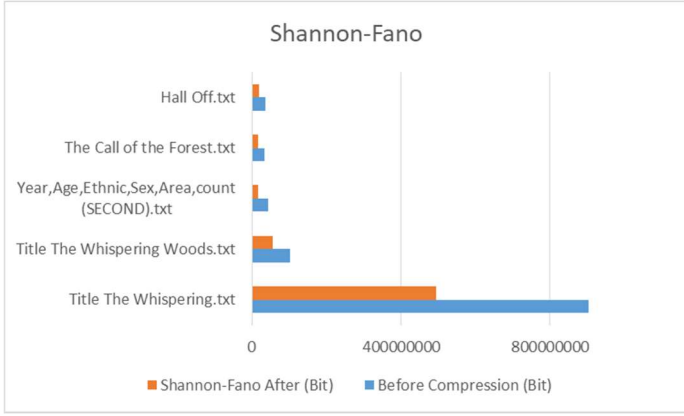
$$Compression\ Ratio = \frac{Jumla\quad bit\ yang\ sudah\ di\ kompresi}{Jumlah\ bit\ sebelum\ dikompresi}\ x\ 100\% \quad (1)$$

HASIL DAN PEMBAHASAN

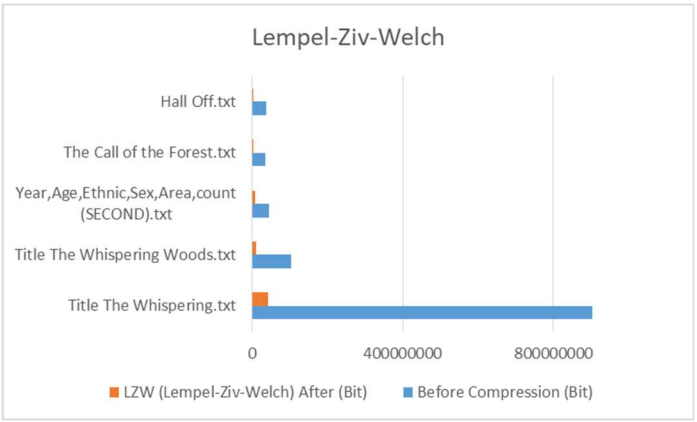
Hasil pengujian kompresi pada file TXT membandingkan algoritma Shannon-Fano dan Lempel-Ziv-Welch (LZW) berdasarkan ukuran file sebelum dan sesudah kompresi. Efektivitas algoritma dinilai menggunakan Compression Ratio (CR), yang membandingkan ukuran file asli dengan file setelah kompresi, serta Redundancy, yang mengukur informasi berlebih yang dapat dihilangkan tanpa kehilangan data penting. Kedua parameter ini menggambarkan efisiensi masing-masing algoritma dalam mengurangi ukuran data. Efisiensi algoritma tersebut dapat dilihat pada tabel 4 berikut ini:

Tabel 4. Hasil Pengujian Perbandingan Kompresi

Nama File	Sebelum Kompresi (Bit)	Shannon - Fano			LZQ (Lempel-Ziv-Welch)		
		Setelah Kompresi (Bit)	Compression Ratio (%)	Redundancy (%)	Setelah Kompresi (Bit)	Compression Ratio (%)	Redundancy (%)
Title The Whispering.txt	904296960	494229600	45.35	45	41075738	95.46	95
Title The Whispering Woods.txt	103315960	56465788	45.35	45	10785946	89.56	90
Year,Age,Ethnic,Sex,Area,count (SECOND).txt	43575424	18081144	58.51	59	8176094	81.24	81
Year,Age,Ethnic,Sex,Area,count (SECOND).txt	43575424	18081144	58.51	59	8176094	81.24	81
The Call of the Forest.txt	33360784	18019421	45.99	46	1858678	94.43	94
Hall Off.txt	37716480	20377300	45.97	46	1990978	94.72	95



Gambar 2. Grafik Kompresi Algoritma Shannon-Fano (TXT)



Gambar 3. Grafik Kompresi Algoritma Lempel Ziv Welch (TXT)

Tabel 4 menunjukkan perbandingan efektivitas algoritma *Shannon-Fano* dan *Lempel-Ziv Welch* (LZW) dalam mengompresi data teks. Pengujian dilakukan dengan menganalisis ukuran file setelah kompresi, *Compression Ratio* (CR), dan tingkat redundansi (RD) LZW menghasilkan ukuran file yang lebih kecil dan *Compression Ratio* yang lebih tinggi, berkisar antara 81,24% hingga 95,46%, dibandingkan *Shannon-Fano* yang mencapai 45,35% hingga 58,51%. Tingkat redundansi data juga lebih rendah pada LZW, yaitu 0,81 hingga 0,95. Pada gambar 2 dan gambar 3, dari 5 data teks yang yang diuji coba dengan ukuran file yang berbeda, menunjukkan penggunaan algoritma LZW lebih unggul dalam jumlah bit setelah dilakukan kompresi jika dibandingkan dengan algoritma Shannon – Fano. Keunggulan LZW ini disebabkan oleh pembentukan kamus dinamis untuk pola berulang, sehingga lebih efektif dalam mengurangi ukuran file dibandingkan *Shannon-Fano* yang hanya mengandalkan frekuensi karakter.

KESIMPULAN

Pengujian kompresi file TXT menggunakan algoritma Shannon-Fano dan LZW memberikan wawasan tentang keunggulan dan keterbatasan masing-masing algoritma dalam mengompresi data teks, yang diukur berdasarkan ukuran file setelah kompresi, Compression Ratio (CR) dan Redundancy (RD). Secara khusus Algoritma LZW terbukti lebih unggul dibandingkan Shannon-Fano dalam mengompresi file TXT. LZW secara konsisten menghasilkan compression ration sebanyak 81 - 95% dan RD yang lebih rendah antara 81 – 95%, terutama untuk teks dengan pola berulang yang kompleks. Shannon-Fano cenderung kurang efektif dalam mengompresi teks dengan struktur data yang kompleks, namun dapat lebih optimal untuk file TXT dengan karakteristik sederhana atau tanpa pola berulang. Dengan demikian, untuk pengompresan file TXT, algoritma LZW direkomendasikan apabila teks memiliki pola pengulangan yang signifikan.

DAFTAR PUSTAKA

[1] D. Cahayati, A. M. H. Pardede, and H. Khair, “Implementasi Algoritma Elias Gamma Kompresi Pada File Teks.”

[2] C. Lamorahan, B. Pinontoan, and N. Nainggolan, “Data Compression Using Shannon-Fano Algorithm.”

[3] C. P. Nugraha, R. Gunawan Santosa, and L. Chrisantyo, “PERBANDINGAN METODE LZ77, METODE HUFFMAN DAN METODE DEFLATE TERHADAP KOMPRESI DATA TEKS,” 2014.

[4] M. Iqbal, R. A. Prayogi, and D. Yunitasari, "ANALISIS KOMPRESI FILE TEKS MENGGUNAKAN ALGORITMA LEMPEL ZIV WELCH (LZW)," 2022. [Online]. Available: http://journal.unnes.ac.id/sju/index.php/ujm

[5] S. Staf *et al.*, "DINAMIKA Jurnal Ilmiah Teknik Mesin IMPLEMENTASI DAN ANALISIS KINERJA ALGORITMA SHANNON-FANO UNTUK KOMPRESI FILE TEXT," vol. 6, no. 1, 2014.

[6] R. Wiryadinata, "DATA COMPRESSION CODING USING STATIC AND DYNAMIC METHOD OF SHANNON-FANO ALGORITHM," *Media Informatika*, vol. 5, no. 2, pp. 129–139, 2007, [Online]. Available: http://wiryadinata.web.id;

[7] A. Sihotang, "Implementasi Algoritma Prefix Codes untuk Kompresi File Video Hasil Ekstra Aplikasi Kinemaster," 2022. [Online]. Available: https://journal.grahamitra.id/index.php/biostech

[8] G. Eko and R. Ningrum, "PENERAPAN METODE SHANNO FANO DALAM PENGKOMPRESIAN DATA TEKS," 2019.

[9] A. Satyapratama and M. Yunus, "ANALISIS PERBANDINGAN ALGORITMA LZW DAN HUFFMAN PADA KOMPRESI FILE GAMBAR BMP DAN PNG."

[10] I. M. Sianturi, "Perancangan Aplikasi Kompresi File Gambar Dengan Menggunakan Algoritma Stout Code," 2024.

Halaman ini sengaja dikosonkan