

Penerapan Algoritma Aritmathic Coding Untuk Kompresi Record Pada Database Penjualan PT. Octa Putra Jaya

Jesika Christi Simatupang^{1,*}, Rivalri K. Hondro², Sarwandi³, Alwin Fau⁴

¹Fakultas Ilmu Komputer dan Teknologi Informasi, Teknik Informatika, Universitas Budi Darma, Medan, Indonesia

²Fakultas Ilmu Komputer dan Teknologi Informasi, Manajemen Informatika, Universitas Budi Darma, Medan, Indonesia

⁴Fakultas Keguruan dan Ilmu Pendidikan, Pendidikan Teknologi Informasi, Universitas Budi Darma, Medan, Indonesia

³Fakultas Ilmu Komputer dan Teknologi Informasi, Sistem Informasi, Universitas Budi Darma, Medan, Indonesia

Email: ^{1,*}jesikachristi1912@gmail.com, ²alwinfau@gmail.com

Email Penulis Korespondensi : jesikachristi1912@gmail.com

Abstrak—Abstrak Besarnya jumlah data yang telah disimpan didalam memori atau pun hardisk akan menambah besarnya kapasitas pemakaian pada media penyimpanan yang digunakan. Ukuran file yang besar juga dapat mempengaruhi transfer data antara media penyimpanan, maka diperlukan salah satu teknik kompresi untuk memampatkan record pada basis data penjualan pada PT. Octa Putrajaya sehingga ukuran media penyimpanan yang diperlukan untuk penyimpanan lebih kecil. Adapun solusi dalam masalah ini adalah record basis data tersebut dapat dikompresi guna untuk mempercepat transmisi dan menghemat penggunaan media penyimpanan. Kompresi pada record basis data dapat dilakukan dengan memperkecil ukuran file dengan proses mengurangi bit pada record basis data, akan tetapi tidak menghilangkan atau membuat record basis data hilang atau rusak. Dengan melakukan kompresi, data yang besar akan berkurang ukurannya sehingga dapat menghemat ruang penyimpanan. Dalam penelitian ini, algoritma yang digunakan adalah algoritma aritmathic coding, dengan menggunakan algoritma tersebut, hasil kompresi dari nilai mempunyai hasil yang berbeda-beda dari setiap nilainya, dan hasil kompresi akan mempermudah dalam melakukan pengiriman dan pemindahan record basis data akan semakin mudah.

Kata Kunci: Kompresi; Record; Basis Data; Aritmathic Coding

Abstract—An abstract. The large amount of data that has been stored in memory or the hard disk will increase the usage capacity of the storage media used. Large file sizes can also affect data transfer between storage media, so a compression technique is needed to compress records in the sales database at PT. Octa Putrajaya so that the size of the storage media required for storage is smaller. The solution to this problem is that the database records can be compressed in order to speed up transmission and save on storage media usage. Compression of database records can be done by reducing the file size by reducing the bits in the database record, but not eliminating or causing the database record to be lost or damaged. By compressing large data, its size will be reduced so that it can save storage space. In this research, the algorithm used is an arithmetic coding algorithm. By using this algorithm, the compression results of the values have different results for each value, and the compression results will make it easier to send and transfer database records.

Keywords: Compression; Record; Database; Arithmathic Coding

1. PENDAHULUAN

Algoritma arithmetic coding melakukan penggantian satu deretan simbol input dengan sebuah bilangan floating point. Semakin panjang dan semakin kompleks pesan yang dikodekan, akan semakin banyak bit yang diperlukan untuk keperluan tersebut[1]. Output dari pengkodean arithmetic coding adalah satu angka yang lebih kecil dari angka 1 dan lebih besar atau sama dengan 0. Angka ini secara unik dapat di decoding, tiap simbol yang akan di encoding diberi satu set nilai probabilitas. Arithmetic coding dapat juga didefinisikan sebagai suatu bagian dari entropy encoding yang mengkonversi suatu data ke dalam bentuk data yang lain dengan lebih sering menggunakan sedikit bit dan jarang menggunakan lebih banyak bit karakter. Kompresi record database penjualan pada PT.Octa PutraJaya akan membantu meminimalisir penggunaan media penyimpanan serta lebih efisien dalam transmisi data. Kompresi record database penjualan pada PT.Octa PutraJaya yang dilakukan yaitu dengan memperkecil ukuran record dengan proses mengurangi bit pada record database, akan tetapi tidak menghilangkan atau membuat record database rusak.

Berdasarkan penelitian yang dilakukan oleh Ratiya Tari, Surya Darma Nasution, dkk (2021) “Penerapan Algoritma Arithmetic Coding Untuk Mengkompresi Record Database Pada Aplikasi Ensiklopedia Flora Berbasis Android” menyimpulkan bahwa kompresi yang dilakukan berdasarkan algoritma arithmetic coding telah berhasil melakukan proses kompresi record database pada aplikasi ensiklopedia flora, sehingga proses kompresi dapat berjalan sesuai dengan teknik kompresi. Penerapan algoritma arithmetic coding untuk mengkompresi record database aplikasi ensiklopedia flora, berdasarkan contoh kompresi dalam penelitian ini menghasilkan 72bit setelah dikompresi, Ratio Compression 2,2, Compression Ratio 45%, Redudancy 55% [2].

Data berupa text, file teks, gambar, vidio, maupun audio, semakin besar ukuran file maka semakin besar juga ruang penyimpanan yang dibutuhkan. Untuk kebutuhan pengiriman data melalui transmisi, maka akan semakin lama waktu yang dibutuhkan proses pengiriman dan pengaksesan data tersebut. Salah satu alternative pemecahan masalah yang akan dilakukan adalah dengan melakukan kompresi data sehingga data tersebut menjadi lebih kecil ukurannya dan proses transmisi data akan semakin cepat. Besarnya jumlah data yang telah disimpan didalam memori ataupun hardisk akan mengakibatkan penambahan besarnya kapasitas pemakaian pada media penyimpanan tersebut. Database bisa menampung dan menyimpan data dalam skala besar dan juga jenis data yang disimpan ke dalam database tersebut bisa bervariasi. Banyaknya data yang tersimpan kedalam database maka akan mengakibatkan pengaksesan data kedalam

database tersebut semakin lama. Maka dengan itu perlu dilakukan sebuah Teknik kompresi record database agar lebih cepat dalam tingkat pengaksesan data.

Database memudahkan penggunaannya untuk mengakses dan menyimpan data, database memiliki beberapa fungsi diantaranya adalah mengelompokkan data untuk mempermudah identifikasi data, menghindari data ganda, memudahkan akses penyimpanan data dan menjaga keamanan data. Database secara umum dapat diartikan sebagai kumpulan dari berbagai macam data[3]. Data tersebut dapat berupa text, gambar, suara, video dan berbagai multimedia lainnya. Database sangat bermanfaat untuk mengatasi berbagai masalah yang sering terjadi dalam penyusunan data. Penggunaan Database dalam sebuah aplikasi dapat mempengaruhi ukuran (size) aplikasi itu sendiri. Aplikasi yang menggunakan database bertujuan untuk mengelompokkan data dalam mempermudah identifikasi data. Semakin banyak data-data di dalam database, maka semakin besar pula ukuran aplikasinya. Ukuran (size) dari database dipengaruhi oleh banyaknya record record database sehingga dengan hal tersebut diperlukan suatu Teknik kompresi untuk mengkompresi record database.

2. METODOLOGI PENELITIAN

2.1 Kompresi

Kompresi data atau pemampatan data (bahasa Inggris: data compression) adalah sebuah cara dalam ilmu komputer untuk memadatkan data sehingga hanya memerlukan ruangan penyimpanan lebih kecil sehingga lebih efisien dalam menyimpannya atau mempersingkat waktu pertukaran data tersebut[2]. Kompresi merupakan proses perubahan data yang memperkecil atau memadatkan ukuran berupa kumpulan karakter menjadi bentuk kode sehingga hanya memerlukan ruang penyimpanan yang lebih kecil dan lebih efisien dalam waktu transmisi data Hasil ukuran kompresi akan dipengaruhi oleh setiap file dan karakteristik dari file yang akan dimampatkan, sedangkan hasil kecepatan pemampatan dari sebuah file dipengaruhi oleh kapasitas dari ukuran file itu sendiri jika besar kapasitas file-nya, maka semakin lama waktu proses kompresi. Dalam mengkompresi suatu data terdapat faktor-faktor yang digunakan untuk menentukan kualitas data yang telah dikompresi, yaitu:

1. Ratio of compression (Rc)
Ratio of compression merupakan suatu cara untuk mengetahui nilai perbandingan dari suatu ukuran bit data file sebelum dilakukan proses kompresi dengan ukuran data file setelah melewati proses kompresi.
2. Compression Ratio (Cr)
Compression Ratio adalah hasil persentase perbandingan dari ukuran data yang telah dikompresi dengan ukuran data sebelum dilakukan kompresi.
3. Redundancy (Rd)
Redundancy merupakan nilai yang diperoleh dari hasil penilaian rasio dengan hasil nilai rasio setelah dikompresi.
4. Space Saving (Ss)
Space Saving adalah selisih antara data yang belum dikompresi dengan besar data yang sudah dikompresi

2.2 Metode dan Tahapannya

Algoritma arithmetic coding melakukan penggantian satu deretan simbol input dengan sebuah bilangan floating point. Semakin panjang dan semakin kompleks pesan yang dikodekan, akan semakin banyak bit yang diperlukan untuk keperluan tersebut. Output dari pengkodean arithmetic coding adalah satu angka yang lebih kecil dari angka 1 dan lebih besar atau sama dengan 0. Angka ini secara unik dapat di decoding, tiap simbol yang akan di encoding diberi satu set nilai probabilitas. Arithmetic coding dapat juga didefinisikan sebagai suatu bagian dari entropy encoding yang mengkonversi suatu data ke dalam bentuk data yang lain dengan lebih sering menggunakan sedikit bit dan jarang menggunakan lebih banyak bit karakter.

Algoritma arithmetic coding menggunakan dua variabel low dan high untuk mendefinisikan interval [low, high). Berikut ini proses encoding dari algoritma arithmetic coding:

1. Set low = 0,0 (kondisi awal)
2. Set high = 1.0 (kondisi awal)
3. While (simbol input masih ada)
4. Ambil simbol input
5. $CR = high - low$
6. $High = low + CR * high_range (symbol)$
7. $Low = low + CR * low_range (symbol)$
8. End while
9. Cetak low

Proses encoding algoritma arithmetic coding, sebagai berikut:

1. Ambil encoded-symbol (ES)
2. Do
3. Cari range dari simbol yang melingkupi ES
4. Cetak simbol
5. $RC = high_range - low_range$

6. ES = ES-low_range
7. ES = ES / CR
8. Until symbol habis

3. HASIL DAN PEMBAHASAN

Transaksi yang dilakukan dalam penjualan dan pembelian maka tentunya akan mengelolah data. PT. Octa Putrajaya adalah salah satu perusahaan yang bergerak dibidang produk pakaian wanita khususnya pakaian remaja. Setiap hasil produksi yang telah selesai dilakukan oleh bagian produksi, kemudian produk tersebut dipasarkan dan dijual diseluruh cabang maupun outlet yang menjadi bagian dari PT. Octra Putrajaya. Setiap transaksi yang dilakukan oleh cabang tertentu akan disimpan dan ditampung kedalam satu database penjualan khususnya di PT. Octa Putrajaya. Sehubungan dengan data yang telah saya peroleh bahwa transaksi yang dilakukan untuk setiap harinya cukuplah banyak. Data transaksi yang terjadi disetiap harinya disimpan kedalam sebuah database penjualan PT. Octa PutraJaya. Banyaknya transaksi yang terjadi setiap harinya maka semakin banyak data yang akan tersimpan kedalam database.

3.1 Penerapan Metode

Pada penelitian ini akan membahas 2 proses utama dalam melakukan pengkompresian record database penjualan yaitu proses kompresi dan dekompresi. Penulis akan melakukan kompresi record database penjualan dengan menerapkan algoritma atimathic coding yang merupakan salah satu teknik kompresi lossless.

3.1.1 Proses Kompresi

- a. Membuat daftar frekuensi kemunculan setiap karakter
 Daftar frekuensi kemunculan setiap karakter dibuat untuk menghitung setiap kemunculan nilai nilai dari heksa yang sama.

Tabel 1. Pendataan Frekuensi Karakter Arithmetic Coding

No	Karakter	Heksa	Frekuensi
1	t	116	1
2	h	104	1
3	2	50	1
4	3	51	1
5	1	49	2
6	4	52	2
Jumlah	8		

- b. Menghitung probabilitas kemunculan setiap karakter
- c. Setiap nilai probabilitas yang diperoleh dilakukan dengan membagi nilai frekuensi kemunculan setiap nilai heksa dari karakter dengan jumlah keseluruhan dari nilai heksa. Keseluruhan nilai heksa adalah sebanyak delapan nilai heksa

Tabel 2. Probabilitas Frekuensi Kemunculan Karakter

No	Karakter	Heksa	Frekuensi	Probabilitas
1	t	116	1	1/8 = 0,125
2	h	104	1	1/8 = 0,125
3	2	50	1	1/8 = 0,125
4	3	51	1	1/8 = 0,125
5	1	49	2	2/8 = 0,25
6	4	52	2	2/8 = 0,25
Jumlah		8	1	

- d. Menghitung jangkauan setiap karakter dengan menentukan titik terendah (low) adalah 0,0 dan titik tertinggi adalah (hight) adalah 1,0. Nilai height diberikan pada nilai heksa untuk karakter yang pertama. Kemudian nilai low diberikan pada nilai heksa karakter yang terakhir. Setelah menentukan nilai high dan low, maka selanjutnya melakukan perhitungan secara berurut yaitu dengan cara nilai high (1.0) pada heksa pertama dikurang probabilitas. Hasil pengurangan merupakan nilai low di heksa pertama dan nilai high di heksa kedua, begitu perhitungan seterusnya sampai pada heksa yang terakhir. Perhitungan jangkauan setiap simbol dapat dilihat pada tabel 4.4.
- e. Diinisialisasi nilai titik terendah (low) sebagai 0,0 dan titik tertinggi (high) sebagai 1,0. Kemudian dilakukan perhitungan low dan high sesuai data setiap karakter dengan rumus kompresi arithmetic coding, berikut ini pseudocode algoritma arithmetic coding: algoritma arithmeric coding: Set low = 0.0 Set high = 1.0
 While (karakter input masih ada) do
 Ambil karakter input
 Coderange = high – low
 High = low + Coderange *high_range (symbol)

Low = low + Coderange *low_range (symbol)
 End While

Cetak Output dari proses arithmetic coding. Kompresi arithmetic coding dilakukan dengan mengelompokkan dua nilai heksa untuk mempermudah mencari low dan high. Nilai low dan high yang diambil adalah perhitungan yang kedua yaitu (0,75-0,875). Perhitungan nilai kedua diambil dikarenakan melakukan pengelompokkan nilai heksa. Nilai perhitungan kedua inilah yang nanti dilakukan rescaling untuk mencari nilai biner.

Tabel 3. Hasil Scalling nilai

No.	Binner Antara	Binner
1	0,875-0,1	00011111
2	0,75-0,1093	00010101
3	0,3496-0,2694	00010110
4	0,3095-0,2994	00011101
5	0,3069-0,3044	00010111
6	0,3069-0,306275	00011101

Nilai biner yang telah diperoleh dari proses scalling kemudian dibagungkan menjadi satu kesatuan sehingga menjadi 00011111000101010001011000011101 0001011100011101. Berdasarkan perhitungan dengan menggunakan arithmetic coding string bit diatas berjumlah 30 bit. Maka Langkah selanjutnya yang perlu dilakukan adalah dengan membagi jumlah bit tersebut menjadi 8 bit setiap kelompok. Jumlah 48 bit tidak habis dibagi delapan dan menyisakan hasil 6 bit atau $48 \text{ mod } 8 = 6$. Nyatakan hasil pembagian dengan “n”. Langkah selanjutnya yang perlu dilakukan adalah dengan menambahkan padding dan flagging. Laku masukan rumus berikut untuk menambah padding:

$$7 - n + "1"$$

$$7 - 0 + "1" = 0$$

Masukan rumus berikut ini untuk menambahkan flagging

$$9 - n$$

$$9 - 6 = 3 = 00000011$$

Total panjang bit keseluruhan bit setelah dilakukan penambahan padding dan flagging adalah $30 + 1 + 8 = 39$ bit. 00011111000101010001011000011101 0001011100011101**00000011**.

Langkah selanjutnya membagi string bit menjadi per 8 bit, kemudian merubahnya menjadi karakter. Hasil pembagian bit dan karakter dihasilkan dapat dilihat pada table dibawah 4.

Tabel 4. Pembagian string bit dan karakter yang dihasilkan

No	Nilai Binner (8 bit)	Desimal	Karakter
1	11111101	253	2
2	01101101	109	m
3	11011011	219	■
4	11110110	246	÷
5	0000011	3	—

f. Pengukuran hasil kompresi

Langkah selanjutnya adalah menghitung kinerja kompresi data. Total bit yang diperoleh sebelum dikompresi adalah sebanyak 48bit dan ukuran setelah dikompresi adalah sebanyak 39 bit. Berdasarkan hasil pengukuran kinerja kompresi, bahwa nilai Rc menunjukkan hasil dari ukuran data sebelum dikompresi dan setelah dikompresi mengalami pengurangan rasio 0,9. Redudancy menunjukkan kelebihan yang terdapat didalam data sebelum dikompresi. Hasil dari redundancy yaitu selisih antara ukuran data sebelum dikompresi dengan hasil compression ratio Compression ratio (Ss) menunjukkan persentase dari data yang telah terkompresi dengan data yang belum di kompresi dan dinyatakan dengan persen, yaitu 99,1 %.

3.1.2 Dekompresi Record Database

Proses dekompresi dilakukan setelah hasil kompresi diperoleh. Beberapa hasil kompresi yang telah diperoleh, terdapat beberapa karakter yang tidak bisa dibaca. Dekompresi adalah salah satu langka yang dapat dilakukan untuk mengembalikan setiap karakter yang merupakan hasil kompresi. Berikut ini langkah langkah yang dilakukan untuk demopresi record database. Nilai keseluruhan bit 00011111 00010101 0001011 000011101 00010111 00011101 **00000011** dari hasil dekompresi record database. Langkah selanjutnya yang perlu dilakukan adalah dengan menghilangkan nilai padding dan juga nilai flagging dari nilai biner yang terlah peroleh setelah kompresi. untuk menghilangkan padding dan flagging dengan cara mengambil 8 bit terakhir dan merubahnya menjadi bilangan desimal, kemudian nyatakan dengan “n”. $n = 00000011 = 3$

kemudian dengan menggunakan rumus “7+n” seperti dibawah ini: $7+n = 7+3 = 10$ (hilangkan string bit terakhir sebanyak 10 bit). Berdasarkan dari langkah-langkah untuk menghilangkan padding dan flagging, maka jumlah string bit yaitu, berjumlah 63bit seperti diawal, yaitu: 00011111 00010101 0001011 000011101 00010111 00011101 **00000011**. Dilakukan pembacaan string bit dari kiri ke kanan lalu ikuti tabel rescaling hingga menemukan nilai biner

antara. Adapun tabel pengecekan bit adalah sebagai berikut: 00011111 00010101 00010110 00011101 00010111 00011101.

Tabel 5. Hasil Dekompresi

No	Nilai	Keterangan
1	0	Tidak ada
2	00	Tidak ada
3	000	Tidak ada
4	0001	Tidak ada
5	00011	Tidak ada
6	000111	Tidak ada
7	0001111	Tidak ada
8	00011111	Nilai Binner Antara (0,875-0,1)
9	0	Tidak ada
10	00	Tidak ada
11	000	Tidak ada
12	0001	Tidak ada
13	00010	Tidak ada
14	000101	Tidak ada

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan mulai dari tahap Analisa, implementasi dan pengujian sistem dengan menggunakan algoritma aritmathic coding dapat disimpulkan, Algoritma Algoritma Aritmathic Coding dapat diimplementasikan pada pengkompresian record basis data, Algoritma Aritmathic Coding memiliki nilai dari Ratio of Compression 1,2, Compression Ratio 0,9, dan Redudancy 7,1 dan Space Saving 99,1 sehingga dari nilai tersebut diperoleh presentasi kompresi record basis data adalah 99%. hasil pengujian Algoritma Aritmathic Coding untuk tahapan dekompresi record basis dapat dikembalikan kembali nilainya tanpa merubah, hilang atau merusak nilai karakter yang terkandung didalam record basis data tersebut

REFERENCES

- [1] R. Tari, S. D. Nasution, and T. Zebua, "Penerapan Algoritma Arithmetic Coding Untuk Mengkompresi Record Database Pada Aplikasi Ensiklopedia Flora Berbasis Android," vol. 2, no. 7, 2021.
- [2] F. A. Sianturi, "Kompresi File Citra Digital Dengan Arithmetic Coding," J. Tek. Inform. Unika St. Thomas, vol. 03, no. 1, pp. 45–51, 2018, [Online]. Available: <http://ejournal.ust.ac.id/index.php/JTIUST/article/view/245/263>.
- [3] S. K. Lidya, M. A. Budiman, and R. F. Rahmat, "Implementasi dan Analisis Kinerja Algoritma Arithmetic Coding dan Shannon-Fano pada Kompresi Citra BMP," Snastikom, no. May 2016, 2013.
- [4] H. T. Sihotang, "Perancangan Dan Implementasi Algoritma Arithmetic Coding Untuk Aplikasi Kompresi Data Video Dan Audio," J. Mantik Penusa, vol. 2, no. 1, pp. 58–64, 2018.
- [5] M. Octiviani, "Kompresi Ayat Pada Aplikasi Buku Ende Menggunakan Algoritma Arithmetic Coding," Pelita Inform. Inf. dan ..., vol. 9, pp. 97–103, 2020, [Online]. Available: <http://ejurnal.stmik-budidarma.ac.id/index.php/pelita/article/view/2858>.