



Penerapan Algoritma VLBE Pada Aplikasi Kumpulan Cerita Rakyat Indonesia Berbasis Android

Sarifah Sidabutar

Program Studi Teknologi Informatika, Universitas Budi Darma, Medan, Indonesia
Email: sarifahsidabutar20@gmail.com

Abstrak—Permasalahan yang terjadi pada aplikasi kumpulan cerita rakyat Indonesia adalah banyaknya cerita yang tersimpan menyebabkan ukuran aplikasi menjadi cukup besar dan digunakan memiliki ruang memori yang besar, tetapi jika smartphone tersebut memiliki ruang memori yang kecil maka itu menjadi sebuah masalah yang cukup besar. Penerapan suatu algoritma kompresi pada aplikasi kumpulan cerita rakyat Indonesia akan mengatasi permasalahan ukuran aplikasi dan tentu saja dapat menampilkan cerita rakyat sehingga anak-anak dapat mengenal dan memahami cerita rakyat untuk mengurangi record yang akan dikompres dengan algoritma VLBE. Berdasarkan penerapan algoritma (VLBE), telah membuktikan bahwa suatu file cerita rakyat yang memiliki ukuran dengan rasio 50% yang besar dapat dikompres menjadi ukuran yang lebih kecil. Sehingga dapat memberikan proses yang lebih cepat.

Kata Kunci: Penerapan; Algoritma; VLBE; Berbasis Android

Abstract—The problem that occurs in the application of Indonesian folklore collection is the large number of stories stored causing the application size to be quite large and used to have a large memory space, but if the smartphone has a small memory space then it becomes a big enough problem. The application of a compression algorithm to the Indonesian folklore collection application will overcome the problem of application size and of course can display folklore so that children can recognize and understand folklore to reduce the record that will be compressed with the VLBE algorithm. Based on the application of the algorithm (VLBE), it has been proven that a folklore file that has a size with a large ratio of 50% can be compressed into smaller sizes. So that it can provide a faster process.

Keywords: Implementation; Algorithm; VLBE; Based on Android

1. PENDAHULUAN

Smartphone saat ini sudah menjadi alat bantu dalam melakukan berbagai pekerjaan dengan berbagai aplikasi yang terinstall didalamnya, Bukan hanya untuk melakukan pekerjaan saja, penggunaan smartphone juga digunakan untuk media belajar dan hiburan. Penggunaan smartphone pada anak-anak biasanya hanya untuk bermain game saja bukannya sebagai media belajar. Anak-anak zaman sekarang kurang menaruh minat pada cerita rakyat yang ada di Indonesia padahal cerita yang ada diberbagai daerah itu perlu diketahui oleh anak-anak zaman sekarang karena cerita itu sudah ada sejak zaman dahulu yang seharusnya dilestarikan.

Aplikasi yang berisikan kumpulan cerita rakyat Indonesia sangat bermanfaat bagi anak-anak ataupun orang dewasa yang ingin mempelajari ataupun mengetahui cerita rakyat yang ada di Indonesia. Pengguna aplikasi kumpulan cerita rakyat bisa saja orang tua yang ingin memberikan cerita sebelum tidur ke anaknya, atau bisa saja seorang guru yang ingin mengajarkan mengenai cerita rakyat ke murid-muridnya. Permasalahan yang terjadi pada aplikasi kumpulan cerita rakyat Indonesia adalah karena banyaknya cerita yang tersimpan menyebabkan ukuran aplikasi tersebut menjadi cukup besar. Permasalahan itu dapat terselesaikan jika smartphone yang digunakan memiliki ruang memori yang besar, tetapi jika smartphone tersebut memiliki ruang memori yang kecil maka itu menjadi sebuah masalah yang cukup besar, karena pengguna harus menghapus *file* atau aplikasi yang lain agar dapat menggunakan aplikasi cerita rakyat Indonesia. Ukuran aplikasi yang besar dapat diatasi dengan menggunakan salah satu teknik kompresi.

Penerapan suatu algoritma kompresi pada aplikasi kumpulan cerita rakyat Indonesia akan mengatasi permasalahan ukuran aplikasi dan tentu saja dapat menampilkan cerita rakyat sehingga anak-anak dapat mengenal dan memahami cerita rakyat serta lebih mengenal cerita rakyat dan melestarikannya. Terdapat banyak algoritma kompresi, salah satunya adalah algoritma *Variable Length Binary Encoding (VLBE)*. Algoritma VLBE diterapkan untuk mengkompresi *record* cerita rakyat yang tersimpan di *database*, sehingga ukuran aplikasi diharapkan menjadi lebih kecil.

Dari hasil penelitian sebelumnya yang dilakukan oleh Novelina Mancelina, Mesran aplikasi ini dirancang telah mampu melakukan kompresi string dengan menggunakan algoritma VLBE dan mampu melakukan dekompresi terhadap hasil kompresi menjadi string hasil[1].

Algoritma VLBE digunakan untuk mengkompresi *file* cerita rakyat Indonesia adalah kode yang memetakan simbol sumber untuk sejumlah *variable bit*. Kode *variable panjang* dapat memungkinkan sumber yang akan dikompresi dengan nol kesalahan (kompresi data *lossless*) dan masi dapat dibaca kembali oleh simbol – simbol. Algoritma VLBE merupakan jenis kompresi *lossless* yang digunakan dalam penelitian ini, yang mana akan diukur dengan rasio kompresi (*Ratio of compression dan Compression Ratio*), *space saving*, dan waktu yang dibutuhkan selama proses kompresi[1].

2. METEDOLOGI PENELITIAN

2.1 Kompresi

Kompresi berarti pemampatan atau mengecilkan ukuran. Sedangkan kompresi data adalah proses mengkodekan informasi menggunakan bit atau *information-bearing unit* yang lain yang lebih rendah daripada representasi data yang tidak terkodekan dengan suatu sistem *encoding* tertentu. Contoh kompresi sederhana yang biasa dilakukan misalnya adalah menyingkat kata-kata yang sering digunakan tapi sudah memiliki konvensi umum, misalnya : kata “yang” dikompres menjadi kata “yg”. Pengiriman data hasil kompresi dapat dilakukan jika pihak pengirim yang melakukan kompresi dan pihak penerima memiliki aturan yang sama dalam hal kompresi data[2].

2.2 Algoritma VLBE

Algoritma *VLBE* adalah algoritma kompresi data yang mengkodekan tiap karakter dengan mengurutkan karakter berdasarkan frekuensi dan disortir secara stabil sehingga membentuk beberapa rangkaian bit. Adapun langkah-langkah dalam menyelesaikan proses kompresi dengan *Aalgoritma Variable Length Binari Encoding* adalah sebagai berikut:

1. Nilai awal biner dengan 0 sebagai root untuk pembentukan biner selanjutnya .
2. Nilai awal biner sebelumnya ditambahkan dengan angka 1 setiap angka didepan awal biner.
3. Jika biner n terakhir nilai awal biner diubah menjadi angka 1[3]

2.3 Java

Java adalah sebuah bahasa pemrograman *scripting* yang sering digunakan dalam pembuatan aplikasi berbasis *handphone* dan juga dapat digunakan untuk menyediakan akses objek yang disisipkan di aplikasi lain. Java berfungsi sebagai penambah tingkah laku agar *widget* dapat tampil lebih atraktif.

Dalam bahasa pemrograman java, seluruh *source code* pertama kali ditulis dalam file teks biasa yang akan berubah menjadi ekstensi java. Lalu seluruh file *source* tersebut akan di compile menjadi *bytecode* dengan ekstensi, clas oleh *java Computer Bytecode* tersebut dapat langsung anda eksekusi di tiap *platform* menggunakan *java Virtual Machine (JVM)* atau *java runtime* sebagai interpreter sehingga java memerlukan pemrograman secara spesifik. *Java Virtual Machine* itu sendiri dapat di artikan sebagai salah satu komponen dari *java plstform* sendiri merupakan *hardware* juga *software*, tempat sebuah program dapat dijalankan seperti *Windows, Linux, Solaris, Mac OS*, dan lainnya

3. HASIL DAN PEMBAHASAN

Berdasarkan penelitian ini, akan dilakukan analisa dan perancangan perangkat lunak pengkompresian *file* kumpulan cerita rakyat dengan menggunakan algoritma *Variable Length Binary Encoding*. Algoritma *Variable Length Binary Encoding* merupakan salah satu teknik kompresi *lossless* yang dapat memperkecil suatu data berdasarkan dengan frekuensi karakter pada objek yang akan dilakukan proses kompresi. Penggunaan algoritma *Variable Length Binary Encoding* akan dilakukan berdasarkan karakter yang sering muncul dan akan memiliki jumlah bit terkecil berdasarkan kode *Variable Length Binary Encoding*, sedangkan karakter paling sedikit muncul akan memiliki jumlah bit terpanjang..

Tahapan analisa terhadap suatu sistem dilakukan sebelum tahapan perancangan dilakukan. Dalam hal ini penerapan algoritma *Variable Length Binary Encoding* dilakukan terhadap *file* kumpulan cerita rakyat yang memiliki ukuran yang relatif besar. Dengan begitu proses kompresi data sangat diperlukan. Kompresi data merupakan teknik pemampatan data, sehingga diperoleh ukuran *file* yang lebih kecil dari aslinya. *File* kumpulan cerita rakyat memiliki ukuran relatif besar, semakin baik kualitas suatu gambar yang dihasilkan, sehingga jumlah *record* yang dibutuhkan untuk proses transmisi semakin besar, dan membutuhkan alokasi penyimpanan yang besar.

Berikut adalah langkah untuk mengkompres dan dekompres *record database*. Contoh sampelnya **Mereka hidup bahagia** untuk mengetahui ukuran *string* tersebut dapat dilihat pada tabel 3.1

Data *string* = Mereka hidup bahagia

Karakter = $\sum = \{M, e, r, e, k, a, s, p, h, i, d, u, p, s, p, b, g\}$

Frekuensi karakter=

M=1	i=2
e=2	d=1
r=1	u=1
k=1	p=1
a=4	b=1
h=2	g=1

Tabel 1. String yang sudah dikompres dengan Variable Length Binary Encoding

Karakter	Variable Length Binary Encoding	Bit	Frekuensi	Bit x Frekuensi
A	0	1	2	2

Karakter	Variable Length Binary Encoding	Bit	Frekuensi	Bit x Frekuensi
E	10	2	4	12
H	110	3	2	6
I	1110	4	2	8
Sp	11110	5	2	10
M	111110	6	1	6

Tabel 2. Keterangan Nilai Variable Length Binary Encoding

Karakte	Variable Length Binary Encoding	Bit	Frekuensi	Bit x Frekuensi
R	1111110	7	1	7
K	11111110	8	1	8
D	111111110	9	1	9
U	1111111110	10	1	10
P	11111111110	11	1	11
B	111111111110	12	1	12
G	111111111111	12	1	12
Total bit				113

Dari tabel dapat dibentuk string bit dari string sebelumnya di kompresi yaitu "Mereka hidup bahagia" menjadi string bit
 "1111101101111110110111111010011101111011111110111111110111111111001111111111010111010111111111111010"

Dengan uraian sebagai berikut:

<u>111110</u>	<u>10</u>	<u>11111010</u>	<u>11111110</u>	<u>0</u>	<u>11110</u>	<u>110</u>	<u>1110</u>		
M	e	r	e	k	a	sp	h	i	
<u>111111110</u>	<u>111111110</u>	<u>1111111110</u>	<u>11110</u>	<u>11111111110</u>	<u>0</u>				
d		u	p	sp	b	a			
<u>110</u>	<u>0</u>	<u>11111111111</u>	<u>1110</u>	<u>0</u>					
h	a	g	i	a					

Kemudian sebelum di dapatkan hasil keseluruhan akhir kompresi dilakukan penambahan *string bit* itu sendiri yaitu *padding* bit dan *flag* bit. Jika sisa bagi panjang *string* bit terhadap 8 adalah 0 maka tambahan 00000001. Nyatakan dengan bit akhir. Sedangkan jika sisa bagi panjang *string* bit terhadap 8 adalah $n(1,2,3,4,5,6,7)$ maka tambahkan 0 sebanyak $7 - n + "1"$ di akhir *string bit*. Nyatakan dengan L. Lalu tambahkan bilangan biner dari $9 - n$. nyatakan dengan bit akhir. karena jumlah string bit 165 tidak habis dibagi delapan dan sisanya 3 bit, nyatakan sisa bagi tersebut dengan nilai n . maka tambahkan 0 sebanyak 0 sebanyak $7 - n + "1"$ di akhir string bit. Nyatakan dengan L.Lalu tambahkan bilangan biner dari $9-n$.Nyatakan dengan bit akhir.

$$7 - n + "1"$$

$$7 - 3 + "1" = \mathbf{00001}$$

Bit Akhir $9 - n$

$$\text{Bit Akhir} = 9 - 3 = 6 = \mathbf{0000110}$$

Gambar 1. Perhitungan Penambahan Bit

111110110111111011011111101001110111101111111011111111101111111111001111111111101011110101111111111111101000010000110

Gambar 2. String Bit Yang Telah Dilakukan Penambahan

Total panjang bit keseluruhan setelah ada penambahan bit adalah $107+5+8=120$. Selanjutnya lakukan pemisahan bit menjadi beberapa kelompok. Setiap kelompok terdiri dari 8 bit seperti gambar di bawah ini.

11111010	11111101	01111111	00111101	10111011
11111101	11111111	01111111	11101111	01111111
11110011	00111111	11111111	01000001	00000110

Gambar 3. Pembagian String Bit

Berdasarkan pada pembagian kelompok nilai biner, didapatkan 15 kelompok nilai biner baru yang sudah terkompresi beserta nilai biner penambahan bit. Setelah pembagian dilakukan, maka desimal yang sudah dibagi dirubah kedalam suatu karakter dengan terlebih dahulu mencari nilai desimal dari *string bit* tersebut menggunakan

kode ASCII untuk mengetahui nilai dari desimal yang sudah terkompresi. Adapun nilai desimal yang sudah terkompresi dapat dilihat pada tabel di bawah ini:

<u>11111010</u>	<u>11111101</u>	<u>01111111</u>	<u>00111101</u>	<u>10111011</u>
ü	ü		=	»
<u>11111101</u>	<u>11111111</u>	<u>01111111</u>	<u>11101111</u>	<u>01111111</u>
ÿ	ÿ		ï	
<u>11110011</u>	<u>00111111</u>	<u>11111111</u>	<u>01000001</u>	<u>00000110</u>
ó	?	ÿ	A	ACK

Dari hasil kompresi dengan *Variable Length Binary Encoding* diatas dapat dihitung kinerja kompresi yaitu:

a. *Compression Ratio* (Cr)

$$Cr = \frac{\text{Ukuran Data Sesudah Dikompresi}}{\text{Ukuran Data Sebelum Dikompresi}} \times 100\%$$

$$Cr = \frac{120}{160} \times 100\%$$

$$Cr = 75\%$$

Selanjutnya proses dekompresi setelah teks dikompresi maka akan menghasilkan *file* baru yang berisi informasi berupa tabel karakter dan tabel kode *Variable Length Binary Encoding* yang digunakan serta string bit hasil proses kompresi. Dekompresi dilakukan dengan cara membaca string bit dari indeks 1 sampai ke indeks dari panjang string bit tersebut. Setiap satu indeks pembacaan string bit dicek pada tabel kode *Variable Length Binary Encoding* apakah cocok atau sama, jika cocok maka dituliskan karakter pada tabel karakter dengan indeks yang sama pada tabel kode *Variable Length Binary Encoding* tersebut.

Berikut ini adalah contoh proses dekompresi dari file hasil kompresi pada Tabel 3.3 dapatkan string bit hasil kompresi dengan kode *Variable Length Binary Encoding* yaitu:

111110110111111011011111101001110111101111111011111111011111111001111111111010111010111111111111010” Sebelum dilakukan pembacaan string, awalnya tentukan dahulu berapa panjang string bit yang harus dibaca. Pada delapan bit terakhir di dalam bit string terdapat flag yang memberikan sebuah tanda banyaknya padding yang diberikan pada String bit pada saat proses kompresi. Delapan bit itu adalah “11110110” atau sama dengan 4 pada bilangan desimal. Maka dapat ditentukan indeks terakhir untuk proses pembacaan string adalah total panjang string bit seluruhnya dikurangi dengan flag ditambah 4 atau dapat ditulis $n = \text{panjang String bit} - (\text{flag} + 4)$. Setelah nilai n ditentukan maka dilakukan pembacaan String bit dari indeks ke-1 sampai indeks ke- n . Indeks ke-1 = “0” adalah karakter “M”. Indeks ke-2 = 1 tidak ada pada tabel maka ditambahkan dengan indeks ke-3 menjadi “10” karakter “e” maka ditambahkan karakter e ke dalam string yang telah dibuat tadi sehingga menjadi “Me” dan begitu seterusnya sampai indeks ke- n sehingga dihasilkan string semula yaitu “**Mereka hidup bahagia**”

Dengan uraian tersebut sebagai berikut.

<u>111110</u>	<u>10</u>	<u>11111010</u>	<u>11111110</u>	<u>0</u>	<u>11110</u>	<u>110</u>	<u>1110</u>	
M	e	r	e	k	a	sp	h	i
<u>11111110</u>	<u>111111110</u>	<u>1111111110</u>	<u>11110</u>	<u>11111111110</u>	<u>0</u>			
d	u	p	sp	b	a			
<u>110</u>	<u>0</u>	<u>111111111111</u>	<u>1110</u>	<u>0</u>				
h	a	g	i	a				

4. KESIMPULAN

Berdasarkan dari penelitian yang telah dilakukan, maka hasil akhir dari penelitian tersebut dapat diambil beberapa kesimpulan dari pembahasan sebelumnya. Adapun kesimpulan yaitu berdasarkan tahap kompresi dengan menggunakan algoritma *Variable Length Binary Encoding* (VLBE), telah berhasil melakukan proses kompresi *file* cerita rakyat seperti karakter lalu menghitung nilai frekuensi menjadi string bit dalam proses kompresi tersebut sehingga proses kompresi dapat berjalan sesuai dengan teknik kompresi, berdasarkan penerapan algoritma *Variable Length Binary Encoding* (VLBE). telah membuktikan bahwa suatu *file* cerita rakyat yang memiliki ukuran dengan rasio 50% yang besar dapat dikompres menjadi ukuran yang lebih kecil. sehingga dapat memberikan proses yang lebih cepat dan Kompresi *file* cerita rakyat dapat dirancang dan dibangun dengan menggunakan aplikasi *eclipse juno* dengan menerapkan algoritma *Variable Length Binary Encoding* (VLBE), sehingga diharapkan memudahkan penulis dalam mengkompresi *record* cerita rakyat.

REFERENCES

- [1] K. Edy Wibowo, Hananto., Sakti Wijayanto, Indra., Herucahyono, Heru., .
- [2] D. Salomon, *Handbook of Data Compression*. London: Springer-Verlag London Limited, 2010.
- [3] “JURNAL GRAMATIKA Jurnal Penelitian Bahasa dan Sastra Indonesia V1.i2 (183-192),” vol. 2, pp. 183–192, 2015.
- [4] Rosa A.S. M. Shalahuddin, *Rekayasa Perangkat Lunak*. Bandung, 2015.



- [5] A. F. Sallaby, F. H. Utami, and Y. Arliando, "Aplikasi widget berbasis java," vol. 11, no. 2, 2015.
- [6] J. Ilmiah, I. Komputa, E. Volume, A. Issn, and A. Juansyah, "PEMBANGUNAN APLIKASI CHILD TRACKER BERBASIS ASSISTED – GLOBAL POSITIONING SYSTEM (A-GPS) DENGAN PLATFORM ANDROID Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)," 2015.
- [7] S. K. Alfa Satyaputra, M.sc, Eva Maulina Aritonang, *No Title JAVA for Beginners with eclipse 4.2 Juno*. Jakarta: PT Elex Media Komputindo, 2012.