

Implementasi Algoritma Turbo Boyor Moore Pada Aplikasi Kamus Linguistik

Windi Sri Utami Saragih, Indri Handayani, Lamsah*

Prodi Teknik Informatika, S1, STMIK Budi Darma, Medan Indonesia

Email: ¹windisriutamisaragih@gmail.com, ^{2*}indrihandayani34@gmail.com, ^{3*}lamsahlex2017@gmail.com

Abstrak

Kamus merupakan alat bantu yang digunakan untuk menterjemahkan suatu bahasa. Kamus sangat dibutuhkan dalam dunia pendidikan untuk mencari tahu makna dari kata yang ingin kita cari. Banyak kata yang tidak dapat kita mengerti apa maknanya. Pada aplikasi kamus Linguistik terdapat banyak kata dan maknanya sehingga di dalam pencarian kata yang ingin kita cari harus menggunakan pencocokan string untuk mempercepat pencarian dan untuk mendapatkan hasil pencocokan string yang lebih efisien. Pada aplikasi kamus Linguistik pada pencocokan string digunakan metode Turbo Boyer Moore.

Kata Kunci: Algoritma, String Matching, Turbo Boyer Moore, Kamus Linguistik.

1. PENDAHULUAN

Kamus merupakan buku yang berisi beberapa kumpulan istilah, kata atau nama yang disusun menurut abjad beserta dengan penjelasan makna dan pemakaiannya.

Algoritma pencocokan *String* merupakan algoritma yang digunakan untuk melakukan pencarian sebuah *string* yang terdiri dari beberapa karakter yang bisa disebut dengan *pattern* dalam sejumlah besar text. Dalam pencocokan *string* pada aplikasi Kamus Linguistik menggunakan algoritma Turbo Boyer Moore.

Algoritma merupakan urutan langkah-langkah logis pada penyelesaian masalah yang disusun secara sistematis. Masalah dapat berupa apa saja, dengan catatan untuk setiap masalah ada syarat kondisi awal yang harus dipenuhi sebelum menjalankan algoritma. Algoritma ini diklasifikasikan sebagai algoritma yang membaca *string* dari kiri ke kanan. Algoritma pencocokan *string* merupakan algoritma yang digunakan untuk melakukan pencarian sebuah *string* yang terdiri dari beberapa karakter. Proses pencarian dalam sebuah aplikasi Kamus Linguistik sangatlah penting, namun terkadang proses pencarian itu lambat. Untuk mempercepat dan mempermudah suatu proses pencarian, dibutuhkan suatu algoritma yang dapat memaksimalkan proses pencarian tersebut.

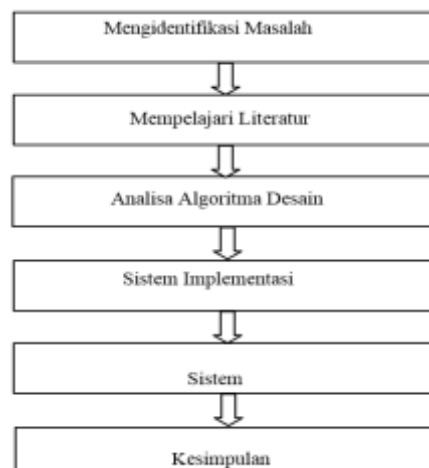
2. METODOLOGI PENELITIAN

Dalam metode ini penulis akan memberikan gambaran langkah-langkah yang mencakup dari awal penelitian sampai dengan akhir penelitian. Agar penelitian dapat terlaksana dengan terstruktur dan sistematis maka perlu untuk menyusun kerangka kerja penelitian. Masing-masing tahapan dalam kerangka kerja tersebut kemudian dijelaskan tentang pelaksanaannya. Hakekat penelitian dapat dipahamkan dengan mempelajari berbagai aspek yang mendorong penelitian untuk melakukan penelitian. Setiap orang mempunyai motivasi yang berbeda dalam melakukan penelitian, keinginan untuk mendapatkan dan mengembangkan pengetahuan merupakan kebutuhan dasar manusia yang umumnya menjadi motivasi untuk melakukan penelitian.

2.2 Kerangka Kerja

Kerangka kerja dalam penelitian ini adalah uraian secara rinci terhadap masing-masing kerangka kerja yang telah disusun agar penelitian yang dilakukan dapat terlaksana secara terstruktur dan jelas.

Kerangka kerja dalam penelitian diperlihatkan pada gambar 1.



Gambar 1. Kerangka Kerja



Uraian langkah kerja dari gambar 1, sebagai berikut :

1. Langkah awal dalam penelitian ini adalah mengidentifikasi masalah dari penelitian ini, masalah yang diteliti harus ditentukan terlebih dahulu, karena tanpa menentukan serta mendefinisikan masalah yang akan diteliti, maka tidak akan pernah ada solusi untuk masalah tersebut.
2. Mempelajari literatur, untuk mencapai tujuan penelitian, kemudian literatur-literatur yang dipelajari diseleksi untuk dapat menentukan literatur yang akan digunakan dalam penelitian. melalui studi literatur dipelajari teori-teori yang berhubungan dengan pengenalan pola (*pattern*), dan algoritma Turbo Boyer Moore. Teori-teori tersebut berdasarkan sumber dari buku, jurnal, prosiding, maupun situs internet.
3. Analisa algoritma dan analisa sistem, dari studi literatur yang telah dilakukan selanjutnya dilakukan taha analisis dengan menentukan algoritma Turbo Boyer Moore yang digunakan untuk menentukan pencocokan *string* yang telah dicari.
4. Desain sistem, pada langkah ini analisa yang telah dilakukan dan ditetapkan menjadi desain sistem. Hasil dari tahap ini adalah untuk mendapatkan model perangkat lunak.
5. Implementasi algoritma, pada tahap ini dilakukan implementasi algoritma kedalam desain sistem yang telah dibuat sebelumnya.
6. Kesimpulan, setelah melakukan langkah-langkah sebelumnya maka langkah selanjutnya adalah menarik kesimpulan. Kesimpulan diambil untuk dapat menganalisa tingkat keberhasilan algoritma Turbo Boyer Moore dalam melakukan pencocokan *string*.

3. ANALISA DAN PEMBAHASAN

Algoritma *Turbo Boyer Moore* adalah sebuah algoritma pencocokan pola tertentu terhadap suatu kalimat atau paragraf. Algoritma ini merupakan varian dari algoritma *Boyer Moore* yang memungkinkan terjadinya ‘lomatan’ melewati segmen yang tidak memerlukan *prepossessing* tambahan. Algoritma Turbo Boyer Moore mengambil metode pencarian *string boyer moore* yaitu *good suffix* dan *bad character suffix*. Berikut penerapan algoritma *Turbo Boyer Moore* untuk pencarian *pattern* dalam teks :

Teks : SELAMAT TAHUN BARU

Pattern : BARU

Tabel 1. Tabel BMBC dan BMGS

Indeks	0	1	2	3
Pattern	B	A	R	U
BMBC	3	2	1	0
Suffix	0	0	0	4
BMGS	3	3	3	1

$$BMBC = M - i - 1$$

$$BMGS = M - 1 - i \text{ suffix}$$

Ket :

M = Panjang pattern

I = posisi pattern pada teks

Tabel 2. Langkah ke-1

S	E	L	A	M	A	T		T	A	H	U	N		B	A	R	U
B	A	R	U														

$$BMBC[A] - M + (\text{Indeks bawah} + 1) = 2 - 4 + 4 = 2$$

$$BMGS[3] = 1$$

Sehingga attern digeser sebanyak 2 diambil dari nilai maksimal dari kedua perhitungan.

Tabel 3. Langkah ke-2

S	E	L	A	M	A	T		T	A	H	U	N		B	A	R	U
		B	A	R	U												

$$BMBC[A] - M + (\text{Indeks bawah} + 1) = 2 - 4 + 4 = 2$$

$$BMGS[3] = 1$$

Sehingga attern digeser sebanyak 4 diambil dari nilai maksimal dari kedua perhitungan.

Tabel 4. Langkah ke-3

S	E	L	A	M	A	T		T	A	H	U	N		B	A	R	U
				B	A	R	U										

$$BMBC[] - M + (\text{Indeks bawah} + 1) = 4 - 4 + 4 = 4$$



BMGS[3] = 1

Sehingga attern digeser sebanyak 4 diambil dari nilai maksimal dari kedua perhitungan.

Tabel 5. Langkah ke-4

S	E	L	A	M	A	T		T	A	H	U	N		B	A	R	U
								B	A	R	U						

BMBC[H] - M + (Indeks bawah + 1) = 4 - 4 + 4 = 4

BMGS[2] = 3

Sehingga attern digeser sebanyak 4 diambil dari nilai maksimal dari kedua perhitungan.

Tabel 6. Langkah ke-5

S	E	L	A	M	A	T		T	A	H	U	N		B	A	R	U
												B	A	R	U		

BMBC[A] - M + (Indeks bawah + 1) = 2 - 4 + 4 = 2

BMGS[3] = 1

Sehingga attern digeser sebanyak 2 diambil dari nilai maksimal dari kedua perhitungan.

Tabel 7. Langkah ke-6

S	E	L	A	M	A	T		T	A	H	U	N		B	A	R	U
														B	A	R	U

Pada langkah yang ke-6, tidak perlu dilakukan pergeseran lagi karena sudah sampai pada indeks terakhir

4. KESIMPULAN

Dari hasil penulisan dapat diambil kesimpulan-kesimpulan sebagai berikut:

1. Pencocokan *string* meruakan proses pencarian karakter pada pattern terhadap karakter ada teks yang dicari.
2. Algoritma *Tubo Boyer Moore* dapat diterapkan pada aplikasi kamus linguistik untuk mempermudah proses pencarian dan pencocokan karakter .

REFERENCES

- [1] Argakusumah dan Hansun, *Implementasi Algoritma Boyer-Moore Pada Aplikasi Kamus Kedokteran berbasis Android*. 2014.
- [2] Riyanarto Samo, Yeni Anistyasari dn Rahimi Fitri, *Semantic Search*. Andi, 2012.
- [3] Alvin Susanto, *Penggunaan Algoritma Hungarian Dalam Menyelesaikan Persoalan Matriks Berbobot*, *Journal Fakultas Teknik Informatika*. 2009.
- [4] Abu-Zaid dan El-ayyes, *Parallel Search Using KMP Algorithm in Arabic String*, *International Journal of Science and Technology*.2012.