

## PENCARIAN STRING MENGGUNAKAN ALGORITMA BOYER MOORE PADA DOKUMEN

Eza Rahmanita, S.T., M.T.

Jurusan Teknik Informatika, Fakultas Teknik, Universitas Trunojoyo  
Jl. Raya Telang PO. BOX 2 Kamal, Bangkalan, Madura, 691962 e-  
mail: eza\_rahmanita@yahoo.com

### ABSTRAK

Proses pencarian merupakan salah satu kegiatan penting dalam pemrosesan data. Proses ini dapat menghabiskan waktu dalam ruang pencarian yang besar sehingga diperlukan suatu teknik pencarian yang efisien. Algoritma Boyer Moore merupakan suatu solusi pencarian yang efisien dapat melakukan perbandingan pattern mulai dari kanan ke kiri. Jika terjadi ketidakcocokan string dari kanan pattern maka ketidakcocokan akan membantu kita untuk menggerakkan pattern tersebut dengan jarak yang lebih jauh. Gerakan melompat ini akan memberikan informasi berapa banyak pattern harus digeser untuk mencocokkan karakter terakhir yang cocok dengan kemunculan awal pattern. Artinya, akan lebih signifikan dalam mengurangi proses perbandingan, jika kita bisa melompati atau tidak melakukan perbandingan karakter yang diprediksi akan gagal. Algoritma Boyer Moore mempunyai keunggulan dalam waktu menemukan pattern yang akan dicari dalam ukuran file yang lebih besar. Pada file berekstensi .txt dengan file size 4.625 byte dengan varian keyword berebeda. Varian keyword yang sedikit dapat ditempuh dengan waktu lebih cepat pada pattern 'a' yaitu 0,228 detik dengan banyak pattern ditemukan 685. Pada file berekstensi .doc dengan file size 39.936 byte, pada pattern 'yang' dapat diproses dengan waktu 0,542 detik dengan ditemukannya pattern sebanyak 18. Sedangkan pada file berestensi .pdf optimalisasi dari pattern matching yang diproses adalah 0,103 detik dengan file size 15.804 byte dan banyak pattern = 1 untuk pencarian pattern 'suatu'.  
Kata kunci : string, pattern matching , Algoritma Boyer Moore

### ABSTRACT

The searching process is one of the important activities in data processing. This process could spend some time in the big area of searching so it is needed of an efficient searching technique. Boyer Moore's Algorithm is an efficient searching solution that could compare the pattern from right to left. If the string incompatibility was happen, so the incompatibility will be help us to move more distant. This jump moving will give some how much pattern information should remove to match the last character which is suitable with the appearance of pattern beginning. Which the significances will be more specific in decrease of ratio processing. If we able to jump or not doing the character ratio which is predict will be fail. Boyer Moore's work ability depend on the word's length which sought. For file extention .txt with file size 4.625 byte with a variant different keyword, a little keyword variant should be process in 0,228 s with "a" pattern and finded 685. For file extention .doc with file size 39936, with "yang" pattern should be process in 0,542 s and finded 18. For file exstention .pd,f from a pattern matchingshould be process in 0,103 s with file size 15804 and pattern finded = 1 for pattern 'suatu'.

Key words: String, Pattern matching, Boyer Moore's Algorithm.

## 1. Pendahuluan

Algoritma Boyer Moore adalah algoritma pencarian string yang paling efektif saat ini. Algoritma yang ditemukan oleh Bob Boyer dan J. Strother Moore ini telah menjadi standar untuk berbagai literatur pencarian string. Algoritma Boyer Moore akan menyimpan informasi pergeseran untuk melakukan pencarian string. Karakteristik utama dari algoritma Boyer Moore adalah algoritma ini melakukan pencocokan string mulai dari kanan ke kiri. Dengan karakteristik tersebut, ketidakcocokan saat terjadi perbandingan string akan membuat pergerakan pattern melompat lebih jauh untuk menghindari perbandingan karakter pada string yang diperkirakan gagal. Aplikasi ini dibuat dengan menggunakan metode string matching dan algoritma Boyer Moore, yang di harapkan bisa membantu dalam system pencarian data yang di inginkan user.

## 2. Tinjauan Pustaka

Algoritma Boyer Moore banyak digunakan untuk penyelesaian masalah pencarian string pada dokumen bahkan pencarian menggunakan search engine di internet telah menjadi sangat populer. Beberapa bidang diantaranya manufaktur, control process, dan ekonomi. Berikut adalah penelitian yang sudah dilakukan dengan menggunakan algoritma Boyer Moore :

1. Pengenalan Sidik Jari dengan menggunakan Algoritma Pencocokan String Boyer Moore [1]. Pada penelitian ini algoritma Boyer Moore digunakan untuk menangani pengenalan sidik jari dalam biometric untuk proses identifikasi seseorang. Algoritma pencocokan string Boyer-Moore dengan pattern dicocokkan dari kanan ke kiri memungkinkan informasi yang didapat akan lebih banyak. Pemanfaatan algoritma Boyer-Moore dalam struktur sidik jari sebagai string biner yang dicocokkan untuk menemukan suatu keakuratan. Image sidik jari diubah menjadi bilangan biner untuk selanjutnya diproses dengan algoritma Boyer Moore.
2. Kinerja algoritma paralel untuk Pencarian Kata dengan metode Boyer Moore menggunakan PVM [2]. Pada penelitian ini algoritma Boyer Moore digunakan untuk menangani permasalahan dalam kegiatan penting dalam pemrosesan data mengenai optimasi komputasi parallel dengan bantuan parallel virtual machine dalam mengkombinasikan pencarian string dengan metode Boyer Moore akan didapat peningkatan kinerja komputasi untuk kata kunci yang terdiri dari satu huruf saja.
3. Pencarian nama yang memiliki kesamaan fonetik menggunakan Algoritma Kesamaan String Boyer Moore[3]. Pada penelitian ini, kesamaan fonetik yang dibahas adalah merupakan ukuran kesamaan yang sangat tergantung pada bahasa yang digunakan dalam penerapannya pada perkembangan ejaan bahasa Indonesia. Perkembangan ejaan yang terjadi di Indonesia yaitu terdapat beberapa ejaan bahasa Indonesia, diantaranya
  - 1). Ejaan Van Ophuysen, Ejaan ini digunakan untuk menuliskan kata-kata Melayu menurut model yang dimengerti oleh orang Belanda, yaitu menggunakan [huruf Latin](#) dan bunyi yang mirip dengan tuturan Belanda, diantaranya adalah kata jang, pajah, sajang, goeroe, itoe, oemoer.
  - 2). Ejaan Republik (Ejaan Soewandi), adalah ketentuan ejaan dalam Bahasa Indonesia yang berlaku sejak 17 Maret 1947. Ejaan ini kemudian juga disebut dengan nama edjaan Soewandi, Menteri Pendidikan dan Kebudayaan kala itu. Ejaan ini

mengganti ejaan sebelumnya, yaitu Ejaan Van Ophuijsen yang mulai berlaku sejak tahun 1901. Perbedaan antara ejaan ini dengan ejaan Van Ophuijsen ialah huruf 'oe' menjadi 'u', seperti pada *goeroe* → *guru*, 3). Ejaan Melindo adalah sistem [ejaan Latin](#) yang termuat dalam Pengumuman Bersama Edjaan Bahasa Melaju-Indonesia (Melindo) (1959) sebagai hasil usaha penyatuan sistem ejaan dengan [huruf Latin](#) di [Indonesia](#) dan [Persekutuan Tanah Melayu](#), sebagai contoh kata “tj” dalam bahasa Indonesia dan “ch” dari bahasa Melayu berubah menjadi “c”, 4). Ejaan Baru (Ejaan LBK), Merupakan lanjutan dari rintisan panitia ejaan melindo, sebagai contohnya adalah kata “remadja” menjadi “remaja”, dan 5). Ejaan Bahasa Indonesia Yang di sempurnakan (EYD) sebagai contoh “wadjar” menjadi “wajar”. Jika hal ini terjadi pada penulisan nama, maka pendeteksian akan mengalami kesulitan dalam fonetik. Oleh sebab itu algoritma Boyer Moore digunakan untuk menentukan varian huruf berdasarkan ejaan yang baru. Jadi huruf-huruf yang tadinya masih menggunakan ejaan yang lama akan langsung diproses menjadi huruf dengan ejaan yang baru, yang sesuai dengan ejaan baru Bahasa Indonesia untuk kemudian diproses pencarian string-nya dengan cara kerja algoritma Boyer Moore, yaitu mengecek pattern dari kanan ke kiri.

Sedangkan pada penelitian ini, algoritma Boyer Moore digunakan untuk menentukan optimasi pencarian string dalam dokumen berekstensi .txt .pdf dan .doc.

Algoritma Boyer-Moore diperkenalkan oleh Bob Boyer dan J.S. Moore pada tahun 1977. Pada algoritma ini pencocokan kata dimulai dari karakter terakhir kata kunci menuju karakter awalnya. Jika terjadi perbedaan antara karakter terakhir kata kunci dengan kata yang dicocokkan maka karakter-karakter dalam potongan kata yang dicocokkan tadi akan diperiksa satu per satu. Hal ini dimaksudkan untuk mendeteksi apakah ada karakter dalam potongan kata tersebut yang sama dengan karakter yang ada pada kata kunci. Apabila terdapat kesamaan, maka kata kunci akan digeser sedemikian rupa sehingga posisi karakter yang sama terletak sejajar, dan kemudian dilakukan kembali pencocokan karakter terakhir dari kata kunci. Sebaliknya jika tidak terdapat kesamaan karakter, maka seluruh karakter kata kunci akan bergeser ke kanan sebanyak  $m$  karakter, di mana  $m$  adalah panjang karakter dari kata kunci.

Booyer-Moore merupakan salah satu Algoritma Pattern Matching yang cukup terkenal. Algoritma ini menggunakan beberapa kasus pengecekan teks (input karakter yang akan dibaca) dengan Pattern (pola yang akan disaring). Algoritma Boyer-Moore adalah algoritma pencarian string yang mencari dengan cara membandingkan sebuah huruf dengan huruf yang ada di pattern yang dicari, dan menggeser pattern tersebut hingga posisinya sama dengan teks yang dicari dan membandingkan kata tersebut. Cara ini disebut character jump.

Algoritma Pattern Matching Booyer-Moore ini berbasis pada 2 metode yaitu :

1. The Looking-Glass Technique

The Looking-Glass Technique melakukan perbandingan suatu karakter akhir pada kata  $w$  dengan suatu karakter pada teks  $s$ . Jika karakter tersebut sama maka jendela karakter akan berjalan mundur pada kedua string dan mengecek kembali kedua karakter. Mencari Suatu kecocokan String pada Teks dengan pola yang akan dicari dengan cara memindahkan atau menggesernya sampai Teks string selesai.

2. The Character-Jump Technique

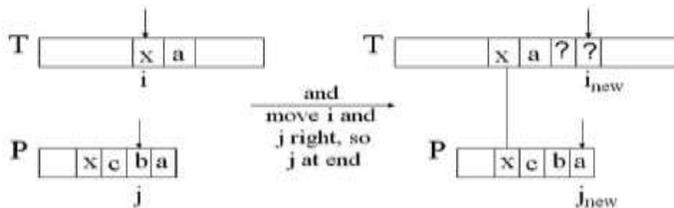
Character-jump Technique melakukan suatu aksi ketika perbandingan antara dua karakter yang berbeda. Ada dua aksi yang tergantung pada teks s dan kata w yang dimiliki; jika p yaitu karakter pada s yang sedang diproses yang tidak cocok maka ada dua kemungkinan aksi. Mencari karakter yang sesuai dan cara penggeseran sebuah karakter perbandingan terakhir.

Beberapa kasus yang ada pada algoritma ini antara lain:

1. Jika suatu karakter Pola (P) mengandung karakter x dimana x adalah anggota dari Teks yang telah dibandingkan maka perbandingan karakter selanjutnya dimulai karakter P yang sama dengan Misal :

T = ..xa..??

P = xcba



Gambar 1. Kinerja Algoritma BM dalam Pencarian String

Keterangan Gambar :

Pada Gambar 1 dapat dijelaskan bahwa T yang akan dicocokkan dengan pattern P adalah kata “xa”. Dengan menggunakan algoritma Boyer Moore, pencocokan akan dimulai dari kanan ke kiri sesuai dengan T yang diminta yaitu “xa” pada pattern P “xcba”. Jika T mengalami ketidakcocokan maka T akan melompat sejauh n karakter T pada Pattern P. jika terjadi kecocokan (match) maka proses akan berhenti.

2. Jika Perbandingan karakter terakhir pada suatu pola sama dengan teks adalah sama.

Maka pergeseran karakter selanjutnya bergeser satu kali.

Misal :

T = ..wax..??

P = cwax

Tabel 1. Kinerja Algoritma BM dalam Pencarian String

P	C	w	A	x
T	w	a	X	
		w	A	x

Keterangan Tabel :

Pada tabel di atas dapat dijelaskan bahwa T yang akan dicocokkan dengan pattern P adalah kata “cwax”. Dengan menggunakan algoritma Boyer Moore, pencocokan akan dimulai dari kanan ke kiri sesuai dengan T yang diminta yaitu “wax” pada pattern P “cwax”. Jika T mengalami ketidakcocokan maka T akan melompat sejauh n karakter T pada Pattern P untuk selanjutnya memulai pencocokan. jika terjadi kecocokan (match) maka proses akan berhenti.

Algoritma Boyer-Moore ini adalah perbandingan karakter dalam sebuah string yang dilakukan dari belakang ke depan atau dari kanan ke kiri karakter. Jika algoritma Boyer Moore

membandingkan teks “MAKALAH” misalnya, algoritma ini melakukan pengecekan apakah karakter ke tujuh dari teks yang dibandingkan adalah karakter ‘H’. Jika karakter ketujuh adalah ‘H’ maka ia akan melakukan pengecekan apakah karakter sebelumnya (ke-6) adalah ‘A’. Demikian seterusnya hingga menemukan bahwa karakter pertama adalah ‘M’.

Alasan kenapa Boyer-Moore melakukan pengecekan dari belakang akan lebih jelas jika kita mengamati apa yang terjadi jika pengecekan menghasilkan nilai yang tidak sama. Misalnya, algoritma mendapati karakter ke-8 dengan karakter ‘X’ bukannya ‘H’. ‘H’ tidak muncul sama sekali pada “MAKALAH”, dan ini berarti tidak ada kesamaan sama sekali karakter ‘X’ dengan semua karakter dalam “MAKALAH”. Sehingga, dengan hanya melakukan sekali pengecekan pada karakter ke-8 kita dapat mengabaikan pengecekan karakter ke-1 sampai ke-7 dan langsung melanjutkan pengecekan karakter dimulai dari karakter ke-9, tepat setelah ‘X’. Algoritma ini pada awalnya melakukan perhitungan sebuah tabel untuk menentukan banyaknya ‘loncatan’ karakter yang akan dilakukan setelah mendapati sebuah perbandingan yang tidak cocok.

Karakteristik utama dari algoritma Boyer-Moore adalah algoritma ini melakukan pencocokan string mulai dari kanan (belakang). dengan karakteristik tersebut, ketidakcocokan saat terjadi perbandingan string akan membuat pergerakan pattern melompat lebih jauh untuk menghindari perbandingan karakter pada string yang diperkirakan gagal. Sehingga proses pencarian string akan lebih optimal.

Contoh cara kerja algoritma Boyer Moore ini adalah sebagai berikut :

Teks : mendaftar di unijoyo

Pattern : daftar

	m	e	n	A	f	t	a	r	d	i		U	n	i	j	o	y	o
1.	d	a	f	t	A	r												
2.		d	a	f	T	a	r											
3.			d	a	F	t	a	r										
4.				i	A	f	t	a	r									

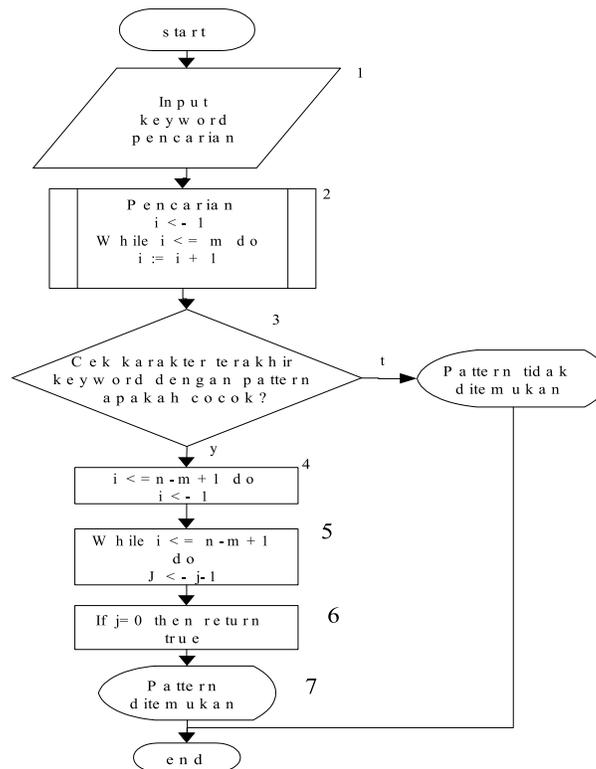
Gambar 2. Contoh Algoritma Boyer Moore

Dalam Gambar 2 dengan menggunakan Algoritma Boyer-Moore dalam pencarian string yang mencari dengan cara membandingkan sebuah huruf dengan huruf yang ada di pattern (m karakter = daftar) yang dicari, dan menggeser pattern sejauh m karakter tersebut hingga posisinya sama dengan teks = n karakter yang dicari dan membandingkan kata tersebut. Dalam pergeserannya, m karakter akan mencocokkan huruf dimulai dari kanan ke kiri. Jika huruf di kanan tidak ada kecocokan dengan n karakter maka m karakter atau pattern akan menggeser sejauh m karakter untuk mencocokkan kembali huruf yang dimaksud, sehingga menjadi cocok.

### 3. Metode Penelitian

Aplikasi yang dibuat adalah aplikasi pencarian string dengan menggunakan Algoritma Boyer Moore, yang bertujuan untuk mempermudah pencarian kata pada dokumen berextensi .txt, .doc dan .pdf dalam beberapa directory dengan sekali proses.

Proses awal adalah menentukan kata yang akan di cari pada file data berextensi .txt, .doc dan .pdf. Setelah itu load data berextensi .txt, .doc dan .pdf pada directory. Nantinya file data extensi .txt, .doc dan .pdf akan diproses dan muncul dalam satu halaman dengan berextensi data yang berbeda-beda. File-file tersebut nantinya akan diproses dengan menggunakan algoritma Boyer Moore agar dapat menentukan pencarian string yang dimaksud.



Gambar 3. Flowchart dari Sistem

### 4. Hasil dan Pembahasan

Data yang digunakan dalam sistem ini adalah data yang berupa file berextensi .txt, .doc, dan .pdf. Dalam hal ini penggunaan file berekstensi .txt dapat langsung diupload oleh web, karena file berekstensi .txt merupakan file yang hanya berisi karakter-karakter dasar tanpa tambahan elemen apapun pada halamannya, sehingga waktu file berekstensi .txt di load, system akan dapat membaca isi file tanpa bantuan apapun. Sedangkan file .doc dan .pdf, untuk membukanya diperlukan beberapa kode-kode berformat tertentu untuk dapat dibuka, tetapi dalam hal ini tampilan dari file berekstensi .doc dan .pdf berbeda atau terlihat sedikit kacau. Karena file berekstensi .doc dan .pdf menyertakan kode-kode terformat untuk elemen-elemen semacam font, margin, tabel, grafik, dan styles yang dikhususkan selama pembuatan dokumen, dalam kasus pembacaan dari file dalam php. Selain itu, terdapat sebuah fungsi yang akan

dipaksa meng-generate isi file, dan memisahkan antara karakter-karakter terbaca dan kode-kode khusus, lalu dilakukan filterisasi dengan hanya menampilkan karakter-karakter yg terbaca oleh sistem dan disimpan ke dalam bentuk dasar berupa txt, itulah sebabnya file .doc & .pdf dapat terbaca oleh system.

File berekstensi lain selain file yang sudah dijelaskan di atas adalah file berekstensi .docx, yang belum dapat dibuka dalam php. Hal ini dikarenakan file berekstensi .docx mempunyai sistem kompresi file dan pengkodean yang lebih rumit dan kompleks, sehingga membutuhkan suatu komponen dasar tambahan yang harus diinstal dalam sistem agar File bisa terbaca dan menterjemahkan kode-kode format tersebut. Dan apabila dibuka secara paksa hanya akan menampilkan kode-kode acak yang susah untuk diterjemahkan oleh scripting php saat ini.

Namun, hal itu bukan berarti file berekstensi .docx tidak dapat dibuka, file berekstensi .docx dapat dibuka, akan tetapi sekarang ini hanya tersedia dalam pemrograman desktop. Dikarenakan pemrograman desktop yang lebih leluasa dalam berkomunikasi atau berintegrasi dengan sistem dan registry.

Pada hasil pengujian dan analisis ini, Pengujian sistem pada pencarian string melalui file berekstensi .txt, .doc, .pdf dapat kita lihat sebagai berikut :

File berekstensi .txt

1. Pencarian String menggunakan keyword “universitas” ditampilkan dengan tanda kuning disetiap kata “universitas” yang dimaksud. Dan dibagian bawah file akan ditampilkan berapa banyak pattern yang sesuai dan total karakter terbaca dalam file data yang diproses, dengan total waktu yang dibutuhkan dalam proses adalah 0,0611 second.



Gambar 4. Tampilan Hasil Search Pattern dengan Keyword “universitas” pada File.txt

2. Berikut tampilan pencarian string dengan keyword “a” :



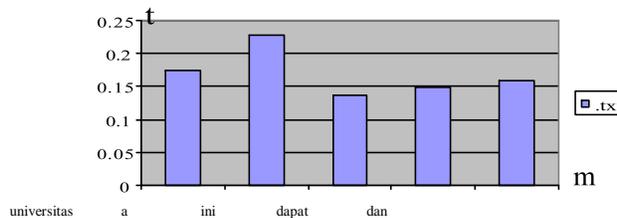
Gambar 5. Tampilan Hasil Search Pattern dengan Keyword “a” pada File txt

Dari uji coba pada contoh Gambar 4 dan Gambar 5, didapat kesimpulan bahwa terdapat pengaruh dari varian keyword dengan waktu proses. Hal ini dipengaruhi oleh 1). Jika jumlah keyword atau string yang di inputkan semakin sedikit, maka semakin cepat waktu prosesnya, 2). Jika semakin banyak pattern yang ditemukan, maka semakin cepat waktu prosesnya. Hal ini disebabkan oleh, jika semakin sedikit keyword yang diinputkan, maka algoritma Boyer Moore

hanya akan melakukan 1 pengecekan saja, sehingga algoritma Boyer Moore akan langsung melompat ke pattern selanjutnya untuk memulai pengecekan kembali.

Tabel 2. Pencarian String dengan Varian Huruf Berbeda pada File Berexstensi .txt

.txt	universitas	a	ini	dapat	Dan
File size	4.625	4.625	4.625	4.625	4.625
Pattern match	3	685	6	2	33
Jumlah karakter	4.625	4.625	4.625	4.625	4.625
Waktu (s)	0.175	0.228	0.137	0.148	0.159



Gambar 6. Perbandingan m dan t dalam file berekstensi .txt

File berekstensi .doc

Untuk pencarian string dengan file berekstensi .doc dapat diuji dengan file berekstensi .txt di atas. Berikut cara untuk pencarian string dengan file berekstensi .doc dengan kata kunci “yang”, “atau”, “untuk”, “guna” dan “dengan” dapat diuji dengan menekan tombol Boyer Moore kemudian pilih Search Pattern. Maka akan muncul di layar seperti gambar dibawah ini.

1. Pencarian String menggunakan keyword “yang”



Gambar 7. Tampilan Hasil Search Pattern dengan Keyword “yang” pada File doc

2. Pencarian String menggunakan keyword “atau”



Gambar 8. Tampilan Hasil Search Pattern dengan Keyword “atau” pada File doc

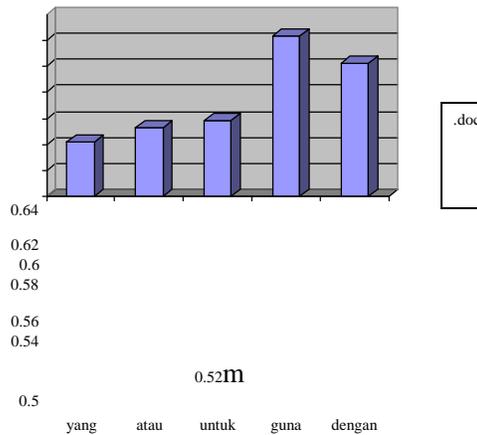
Dari uji coba pada contoh Gambar 7 dan Gambar 8, didapat kesimpulan bahwa terdapat pengaruh dari varian keyword dengan waktu proses. Hal ini dipengaruhi oleh 1). Jika jumlah keyword atau string yang di inputkan semakin sedikit, maka semakin cepat waktu prosesnya, 2). Jika semakin banyak pattern yang ditemukan, maka semakin cepat waktu prosesnya. Hal ini

disebabkan oleh, jika semakin sedikit keyword yang di inputkan, maka algoritma Boyer Moore hanya akan melakukan 1 pengecekan saja, sehingga algoritma Boyer Moore akan langsung melompat ke pattern selanjutnya untuk memulai pengecekan kembali.

Tabel 3. Pencarian String dengan Varian Huruf Berbeda pada File berekstensi .doc

.doc	yang	Atau	untuk	guna	dengan
File size	39936	39936	39936	39936	39936
Pattern match	18	11	6	15	12
Jumlah karakter	10582	10582	10582	10582	10582
Waktu (s)	0.542	0.553	0.558	0.623	0.603

t



Gambar 9. Perbandingan m dan t dalam file berekstensi doc

Pada sistem, pencarian keyword akan ditampilkan dengan tanda kuning di setiap kata atau keyword yang di maksud. Dan di bagian bawah file akan ditampilkan berapa banyak pattern yang sesuai dan total karakter terbaca dalam file data yang diproses, dengan total waktu yang dibutuhkan dalam proses.

File berekstensi .pdf

Untuk pencarian string dengan file berekstensi .pdf dapat diuji dengan dengan kata kunci “ini”, “c”, “dari”, “suatu” dan “maka” dapat diuji dengan menekan tombol Boyer Moore kemudian pilih Search Pattern. Maka akan muncul di layar seperti gambar dibawah ini.

1. Pencarian String menggunakan keyword “ini”



Gambar 10. Tampilan Hasil Search Pattern dengan Keyword “ini” pada File pdf

2. Pencarian String menggunakan keyword “c”

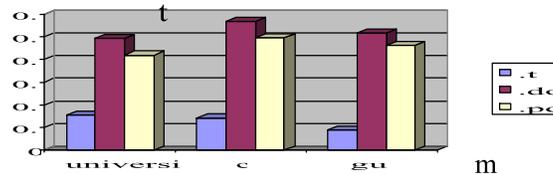


Gambar 11. Tampilan Hasil Search Pattern dengan Keyword “c” pada File pdf

Dalam sistem yang bukan termasuk file berekstensi .txt, .doc dan .pdf tidak dapat diproses dengan tampilnya peringatan bahwa yang dapat diproses dalam system ini hanya file berekstensi .txt, .doc dan .pdf.

Tabel 4. Pencarian String dengan Varian Huruf Berbeda pada File txt

.pdf	ini	c	dari	Suatu	Maka
File size	15804	15804	15804	15804	15804
Pattern match	7	24	11	1	4
Jumlah karakter	6516	6516	6516	6516	6516
Waktu (s)	0.493	0,577	0.485	0.103	0.502



Gambar 12. Perbandingan m dan t dalam file berekstensi .txt , .doc dan .pdf

Dalam hal ini terdapat perbedaan waktu antara pencarian pattern yang berekstensi file .txt, .doc dan .pdf dalam masing-masing pencarian keyword “universitas”, “c”, dan “guna” dengan masing-masing mempunyai file size 4.625 byte pada file berekstensi .txt, file size 40.448 byte pada file berekstensi .doc, dan file size 10.458 byte pada file berekstensi .pdf, ditemukan 3 pattern yang sesuai dari 4.625 karakter dengan waktu 0,661 detik. Sedangkan untuk file berekstensi .doc dalam pencarian keyword “c” dengan file size 40.448 byte, ditemukan 2.896 pattern yang sesuai dari 36.508 karakter dengan waktu 0,734 detik. Dan untuk file berekstensi .pdf dalam pencarian keyword “guna” dengan file size 10.458 byte, ditemukan 6 pattern yang sesuai dari 2.587 karakter dengan waktu 0,576 detik. Dari perbandingan tersebut metode Algoritma Boyer Moore jelas lebih unggul dalam pencarian file yang mempunyai file size lebih besar. Hal ini terbukti dari perbandingan file berekstensi .doc yang mempunyai file size lebih besar dari file berekstensi lainnya.

5. Kesimpulan

Adapun kesimpulan dari penelitian ini adalah sebagai berikut :

1. Algoritma Boyer Moore mempunyai keunggulan dalam waktu menemukan pattern yang akan dicari dalam ukuran file yang lebih besar, pada file berekstensi .txt dengan file size 4.625 byte dengan varian keyword berbeda, varian keyword yang sedikit dapat ditempuh dengan waktu lebih cepat pada pattern 'a' yaitu 0,228 detik dengan banyak pattern ditemukan 685. Pada file berekstensi .doc dengan file size 39936 byte pada pattern 'yang' dapat diproses dengan waktu 0,542 detik dengan ditemukannya pattern sebanyak 18. Sedangkan pada file berekstensi .pdf optimalisasi dari pattern Matching yang diproses adalah 0,103 detik dengan file size 15.804 byte dan banyak pattern = 1 untuk pencarian pattern 'suatu'.
2. Algoritma ini dapat digunakan untuk teknologi mesin pencari.
3. Dari hasil uji coba, ditemukan bahwa efektifitas Algoritma Boyer Moore tergantung pada panjang kata yang dicari. Semakin panjang kata yang dicari, maka semakin hemat waktu yang ditempuh pada file berekstensi .txt. namun pada file berekstensi .doc dan .pdf hal ini merupakan kebalikannya, artinya semakin sedikit m yang ditemukan maka semakin cepat waktu prosesnya, semakin banyak m yang ditemukan maka akan semakin lambat. M yang banyak ditemukan membutuhkan waktu yang lebih lama disbanding m yang lebih sedikit, hal ini dikarenakan ada dua perhitungan yang membuat proses waktu berjalan lebih lambat; yaitu 1) memproses P dengan Boyer Moore terlebih dahulu, 2) menghitung banyaknya pattern yang ditemukan. Hal inilah yang membuat berjalannya proses waktu lebih lambat dan merupakan kelemahan PHP dibanding pemrograman desktop.
4. Parameter seperti panjang karakter per kata dan perulangan karakter dalam kata dapat mempengaruhi efektivitas algoritma pencarian String.

#### 6. Saran

Saran untuk pengembangan aplikasi ini adalah data penunjang dokumen file dapat diisi dengan lebih detail sehingga proses pencarian dengan menggunakan Algoritma Boyer Moore. Diharapkan pengembangan aplikasi ini dapat digunakan secara maksimal pada pencarian string dokumen yang berindeks dokumen medis ICD-10.

#### Daftar Pustaka

- [1]. Muntaha, Amir. Juli 2013. Pengenalan Sidik Jari Dengan Menggunakan Algoritma Pencocokan String Boyer Moore,  
[URL:http://informatika.stei.itb.ac.id/~rinaldi.munir/Penelitian/Makalah-KNIF-2010.pdf](http://informatika.stei.itb.ac.id/~rinaldi.munir/Penelitian/Makalah-KNIF-2010.pdf)
- [2]. Vandika, Stania., K.A, Maria. Juli 2013. Kinerja Algoritma Paralel Untuk Pencarian Kata Dengan Metode Boyer Moore Menggunakan PVM,  
[URL:http://informatika.stei.itb.ac.id/~rinaldi.munir/Penelitian/Makalah-KNIF2010.pdf](http://informatika.stei.itb.ac.id/~rinaldi.munir/Penelitian/Makalah-KNIF2010.pdf)
- [3]. Adisantoso, Julio., Rambe, A., dan Kaliana, Indra. Juli 2013. Pencarian Nama Yang Memiliki Kesamaan Fonetik Menggunakan Algoritma Kesamaan String Boyer Moore,  
[URL:http://informatika.stei.itb.ac.id/~rinaldi.munir/Penelitian/Makalahilmukomputer-2010.pdf](http://informatika.stei.itb.ac.id/~rinaldi.munir/Penelitian/Makalahilmukomputer-2010.pdf)

