

Pencarian Kata Dalam Konten Tugas Akhir Dengan Implementasi Algoritma Knuth Morris Pratt (KMP)

Diana¹, Amri², Husaini^{3*}

^{1,3} Jurusan Tekniknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe
Jln. B.Aceh Medan Km.280 Buketrata 24301 INDONESIA

Email: ¹Dianasalmar0@gmail.com, ²jurusan@tik.pnl.ac.id, ³Husaini@pnl.ac.id

Abstrak— Mesin pencari atau *search engine* adalah program komputer yang dirancang untuk melakukan pencarian terhadap berkas-berkas yang tersimpan dalam computer ataupun komputer server dalam suatu jaringan. Dalam perancangan aplikasi ini membutuhkan algoritma pencocokan string (*string matchin*). Algoritma pencocokan string pada saat ini sudah sangat banyak ditemukan, salah satunya adalah algoritma *Knuth Morris Pratt*. Algoritma *Knuth Morris Pratt* adalah algoritma yang melakukan pencocokan string mulai dari kiri ke kanan. Dengan karakteristik tersebut, ketidakcocokan saat terjadi perbandingan string akan membuat pergerakan *pattern* melompat lebih jauh untuk menghindari perbandingan karakter pada string yang diperkirakan gagal. Algoritma *Knuth Morris Pratt* (KMP) yang diterapkan pada aplikasi pencarian ini dapat menghasilkan pencarian yang sesuai dengan kata kunci yang dimasukkan oleh user berdasarkan kata kunci yang ada pada masing-masing dokumen tugas akhir. Pengujian yang dilakukan pada aplikasi pencarian kata dalam konten tugas akhir tersebut terbukti mampu menampilkan hasil pencarian yang sesuai dengan dokumen yang dicari berdasarkan kata kunci.

Kata kunci: Mesin pencari, Tugas akhir, *Knuth Morris Pratt*, *Pattern*.

I. PENDAHULUAN

Saat ini mesin pencari atau *search engine* sudah tidak asing lagi di kalangan masyarakat apalagi di kalangan mahasiswa, mesin pencari dibuat untuk memudahkan pencarian baik itu pencarian sebuah file, dokumen yang ada di dalam komputer maupun pada situs internet. Pencarian dengan menggunakan mesin pencari akan lebih tepat dan cepat dibandingkan dengan melakukan pencarian tanpa menggunakannya. Khususnya dalam melakukan pencarian kata dalam konten tugas akhir, cukup dengan memasukkan satu kata kunci saja maka akan keluar semua konten yang berkaitan dengan kata kunci tersebut. Kecepatan dan ketepatan pencarian sebuah mesin pencari sangat tergantung kepada cara pengindeksan dan algoritma pencarian yang digunakan. Algoritma pencocokan string pada mesin pencari tentu sudah menjadi bagian yang sangat penting. Oleh karena itu mesin pencari membutuhkan sebuah algoritma *Knuth Morris Pratt* untuk melakukan pencarian.

Dalam pencarian topik penelitian atau judul skripsi beberapa mahasiswa sering kali mengalami kesulitan dan kendala sehingga beberapa kali harus mengganti dan merubah judul skripsi. Pengajuan judul skripsi oleh mahasiswa terkadang diterima ataupun ditolak berdasarkan pertimbangan yang salah satunya adalah judul skripsi sudah ada dan kendala utamanya adalah menghindari indikasi adanya kesamaan judul dan adanya indikasi plagiat. Adanya kesamaan judul skripsi

antara mahasiswa satu dengan lain tanpa dilakukan pengecekan dapat terjadi redundancy judul yang tentunya dapat dihindari apabila ada kontrol dari dosen atau jurusan yang dapat melakukan pengecekan sebelum judul tersebut di setujui. Salah satu upaya yang dapat dilakukan dalam menentukan judul skripsi ialah dengan melakukan pencocokan string (*string matching* atau *pattern matching*) dalam teks yang terdapat pada judul skripsi yang diusulkan. [1]

Dari latar belakang yang telah di uraikan, penulis ingin membuat sebuah mesin pencari untuk mempercepat proses pencarian, mesin pencari ini akan di implementasikan sebuah algoritma *Knuth Morris Pratt* di dalamnya agar pencarian lebih cepat dan tepat. Di samping itu pada mesin pencari juga akan di tambahkan sebuah fitur *word suggestion* untuk member saran kata pada saat melakukan penginputan kata kunci pada mesin pencari. Oleh karena itu penulis mengangkat tugas akhir (TA) dengan judul “Pencarian kata dalam konten tugas akhir dengan implementasi algoritma *Knuth Morris Pratt* (KMP)”.

Berdasarkan latar belakang pemilihan judul, maka yang menjadi permasalahan adalah :

- Berapa panjang *pattern* yang digunakan untuk melakukan pencarian?
- Berapa lama waktu yang dibutuhkan untuk menemukan hasil?

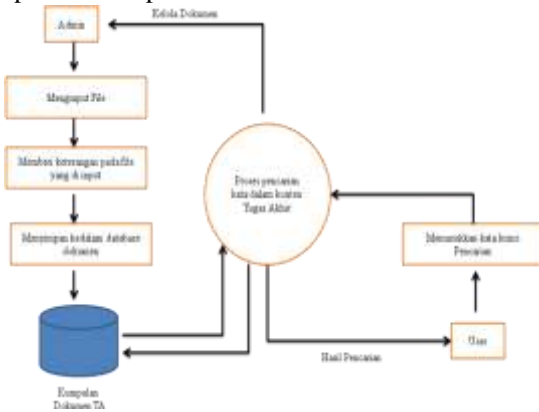
Berdasarkan rumusan masalah, maka yang menjadi tujuan adalah :

- a. Untuk mengetahui panjang *pattern* yang digunakan pada pencarian.
- b. Untuk mengetahui lama waktu yang dibutuhkan untuk menemukan hasil?

II. METODOLOGI PENELITIAN

2.1 Gambaran Umum Sistem

Sistem pencarian kata dalam konten tugas akhir yang dirancang adalah untuk mengatur dan memudahkan mahasiswa dalam melakukan pencarian kata, dengan menggunakan algoritma *knuth morris pratt*. Berikut gambaran umum pada sistem pencarian.

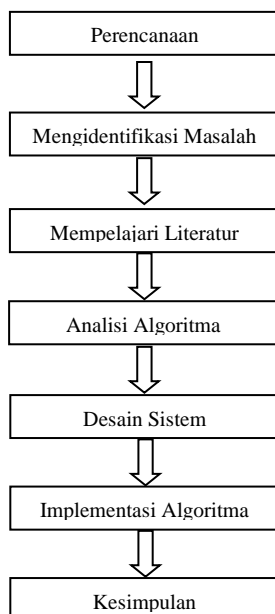


Gambar 1. Gambaran Umum Sistem

2.2 Perancangan Sistem

Perancangan merupakan langkah pertama dalam fase pengembangan rekayasa suatu sistem. Perancangan juga bisa dikatakan sebagai proses dalam menggunakan berbagai prinsip dan teknik untuk mendefinisikan suatu perangkat, proses atau suatu sistem hingga detail sehingga memungkinkan relasi berbentuk fisiknya. Perancangan sistem yang dibuat akan meliputi Diagram Konteks, Kerangka Kerja dan *Flowchart*.

- a. Adapun kerangka kerja dapat dilihat seperti dibawah gambar 2 dibawah ini.



Gambar 2 Kerangka Kerja

Berdasarkan gambar 2 maka uraian langkah kerja dapat ditulis sebagai berikut:

- 1) Langkah awal dalam penelitian ini adalah Perencanaan. Pada tahap ini, ditentukan tujuan yang ingin dicapai dan merencanakan strategi umum untuk memperoleh dan menganalisa data bagi penelitian.
- 2) mengidentifikasi masalah dari ruang lingkup masalah, yaitu mempelajari masalah yang akan diteliti. Ruang lingkup masalah yang akan diteliti harus ditentukan terlebih dahulu, karena tanpa menentukan serta mendefinisikan ruang lingkup masalah yang akan diteliti, maka tidak akan pernah ada solusi untuk masalah tersebut.
- 3) Mempelajari literatur, untuk mencapai tujuan yang telah ditentukan, maka dipelajari beberapa literatur yang digunakan. Kemudian literature-literatur yang dipelajari diseleksi untuk dapat menentukan literatur yang akan digunakan dalam penelitian. Melalui studi literature dipelajari teori-teori yang berhubungan dengan pengenalan pola (*pattern*), *Word Suggestion* dan algoritma *Knuth Morris Pratt*. Teori-teori tersebut berdasarkan sumber berupa buku, jurnal, prosiding maupun dari situs internet.
- 4) Analisa algoritma, dari studi literatur yang telah dijelaskan selanjutnya akan dijelaskan tahap analisis dengan menentukan algoritma *Knuth Morris Pratt* yang digunakan untuk melakukan pencocokan string.
- 5) Desain sistem, pada langkah ini analisa yang akan dilakukan dan ditetapkan menjadi desain sistem untuk mendapatkan model perangkat lunak, model perangkat lunak ini akan digunakan untuk implementasi algoritma tersebut.
- 6) Implementasi algoritma, pada tahap ini dilakukan implementasi algoritma ke dalam desain. Pada tahap implementasi ini algoritma *Knuth Morris Pratt* diterapkan kedalam desain sistem menggunakan bahasa pemrograman PHP dan database MySQL
- 7) Kesimpulan, setelah melakukan langkah-langkah sebelumnya maka langkah selanjutnya adalah menarik kesimpulan dari sistem yang akan dibuat. Diharapkan agar hasil dari penelitian yang akan dilakukan sesuai dengan tujuan yang telah ditetapkan dan dapat menjadi solusi atas masalah-masalah yang dibahas.

2.3 Cara kerja Algoritma *Knuth Morris Pratt*

Adapun cara kerja algoritma *knuth morris pratt* adalah sebagai berikut:

- a. Algoritma *Knuth Morris Pratt* mulai mencocokkan *pattern* pada awal teks.
- b. Di mulai dari kiri ke kanan, algoritma ini akan mencocokkan karakter per karakter *pattern* dengan karakter di teks yang bersesuaian, sampai salah satu kondisi berikut dipenuhi:

1. Semua karakter di pattern cocok (match). Kemudian algoritma akan memberitahukan pencarian berhasil.
 2. Sebuah karakter di pattern dan teks yang dibandingkan terdapat ketidakcocokan (mismatch).
- c. Apabila pattern belum ditemukan kecocokannya dan teks belum habis, maka pattern akan menggeser satu karakter ke kanan dan mengulangi langkah 2 sampai pattern berada di ujung teks.

2.4 Fungsi Pinggiran

Fungsi pinggiran pada algoritma knuth morris pratt ini berguna untuk mencegah pergeseran yang tidak berguna. Fungsi pinggiran hanya berlaku pada karakter yang terdapat dala pattern bukan dalam teks yang akan dicari. Adapun cara untuk mengetahui nilai pada fungsi pinggiran adalah sebagai berikut.

Tabel 1 Nilai Pinggiran

j	0	1	2	3	4	5	6
P(j)	a	b	c	a	b	c	d
B(j)	0	0	0	1	2	3	0

Pada tabel diatas dijelaskan bahwa nilai pinggiran dihitung berdasarkan karakter yang terdapat pada pattern. Pada **p(j) a = 0** , a merupakan karakter pertama yang terdapat pada pattern sehingga nilai pinggiran pada pattern **a = 0**, kemudian pada pattern **b** dan **c = 0**, dikarenakan tidak ada pattern yang sama dengan pattern **b** dan **c**. Pada **j(3)** nilai pinggiran **j(3) = a = 1** karena pattern ini sama dengan pattern **j(0)= a= 0**. Selanjutnya pada **j(4) = b= 2** dan **j(5) = c = 3**.

Berikut adalah pseudo-code dari algoritma KMP secara umum :

```

S1 = S2 = 0
DoWhile(S1 < count(S2))
while(S1 < count(S2))
while(S1 < count(S2) && Spattern[S1] == S2[S1])
S1 = Slongat(S1)
S1++
S2++
if(S1 == count(S2))
break
while(S1 < count(S2))
S1 = Slongat(S1)
S1++
return S1
function preKMP(Spattern)
S1 = 0
S2 = Slongat[0] = -1
while(S1 < count(Spattern))
while(S1 < count(Spattern) && Spattern[S1] == Spattern[S2])
S2 = Slongat(S2)
S1++
S2++
if(Spattern[S1] == Spattern[S2])
Slongat[S1] = Slongat[S2]
else
Slongat[S1] = S2
return Slongat
    
```

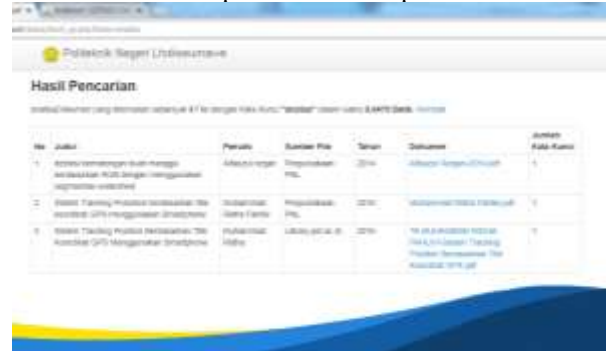
Gambar 3 psoude-code

III. HASIL DAN PEMBAHASAN

3.1 Hasil Pengujian Aplikasi

Dari pengujian aplikasi yang telah di lakukan yaitu pengujian admin yang mengelola sistem dan user sebagai penggunanya. Pada aplikasi ini dilakukan pengujian secara *admin interfase* maupun *user interface* yang bertujuan untuk melihat kesesuaian aplikasi terhadap desain interface antar muka yang telah dilakukan sebelumnya dan kesesuaian tampilan yang sudah berjalan dengan seharusnya.

Pengujian ini melakukan penelitian tentang pencarian kata yang akan di lakukan pada sistem pencarian dengan panjang pattern yang di input dan lama waktu pencarian. Gambar 4 akan menampilkan hasil dari pencarian kata.



Gambar 4 Hasil Pencarian

Pada gambar diatas dapat diuraikan bahwa hasil pencarian dari kata kunci “Analisa” terdapat 3 file yang ditemukan dimana pada masing-masing file tersebut mengandung kata analisa didalamnya.

3.2 Analisa Hasil Pengujian

Pengujian yang dilakukan pada aplikasi pencarian kata dalam konten tugas akhir dengan implementasi algoritma *knuth morris pratt* dengan hasil pengujian yang menghitung string dari karaterr perkarakter berdasarkan pattern dan teks yang disesuaikan.

3.2.1 Hasil Pengujian Tokenizing dan Steeming

Pengujian yang dilakukan menggunakan tokenizing dan steeming, sebelumnya file yang diinput berupa PDF. File asli sebelum diinputkan kedalam sistem dan sebelum terjadinya proses tokenizing dan steeming. Kemudian file tersebut akan diinput kedalam sistem dimana file ini kana tersimpan kedalam database MySQL. Pada saat penginputan file terjadi proses tokenizing dan steeming.

Pada tahap tokenizing semua huruf dalam dokumen akan diubah menjadi huruf kecil dan hanya “a” sampai huruf “z” yang diterima. Karakter selain huruf dihilangkan dan dianggap delimiter.

Potongan dari koding steeming yang bekerja untuk menghapus semua kata imbuhan pada file PDF yang diinput sehingga ketika file PDF yang telah berhasil di input kedatabase hanya kata dasar saja. gunanya menghindari banyaknya pengulangan pada saat perhitungan string. Adapun file yang telah di steaming akan masuk kedalam database

datatxt dan berelasi dengan batchcode. Pada gambar diatas menunjukkan hasil dari proses tokenizing dan steeming dimana semua kata imbuhan telah di hapus. Untuk lebih akurat bisa di lihat kembali PDF dari file yang di input dalam data base.

IV. SIMPULAN

Adapun simpulan yang dapat penulis simpulkan setelah melakukan penelitian dan pembahasan mengenai sistem pendeteksi informasi buku yang dibaca diperpustakaan menggunakan *radio frekuensi identification* (RFID) yaitu:

- a. Berdasarkan pengujian yang telah dilakukan panjang pattern sesuai dengan kata kunci yang di cari.
- b. Waktu yang dibutuhkan untuk melakukan pencarian tidak melebihi 1 menit. Waktu yang dibutuhkan paling tinggi 0.0690 detik dan paling rendah 0.0430 detik berdasarkan pengujian yang dilakukan pada aplikasi pencarian kata.

REFERENSI

- [1] Astuti, W. (2017). Analisis string matching pada judul skripsi dengan algoritma knuth morris pratt (KMP). Ilmiah.
- [2] Barakbah, A. R., Karlita, T., & Ahsan, A. S. (2013). *LOGIKA DAN ALGORITMA*. Surabaya.
- [3] Betha Sidik, I. (2012). *Pemrograman Web dengan PHP*. Bandung: Informatika.
- [4] Eko, S., & Muslim, H. K. (2010). *Teknologi Informasi dan Komunikasi*. Jakarta: PT Sinergi Pustaka Indonesia .
- [5] KBBI. (2018). Retrieved May 11, 2018, from Kamus Besar Bahasa Indonesia: <http://www.kbbi.web.id>
- [6] Pradhana, F. (2012). Penerapan String Matching pada Fitur Auto Correct dan Fitur Auto Text di Smart Phones.
- [7] Riyad, T., & ilmukomputer.com. (2009). Jenis-jenis Search Engine. *Taufan Riyadi* (p. 1). -: ilmukomputer.com.