APLIKASI *INFORMATION RETRIEVAL* UNTUK PENCARIAN DOKUMEN LAPORAN PENELITIAN

Indri Tri Hapsari[1], Banni Satria Andoko[2], Cahya Rahmad[3]
[1,2] Jurusan Teknik Elektro, Program Studi Teknik Informatika, Politeknik Negeri Malang
[1]indri.tri.hapsari@gmail.com, [2]banniandoko@gmail.com, [3]cahyarahmad@gmail.com

Abstrak

Information retrieval atau temu kembali informasi merupakan sistem pencarian untuk menemukan kembali sebuah informasi. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem pencarian dokumen laporan penelitian sehingga dapat mempermudah dalam menemukan kembali dokumen yang diinginkan oleh pengguna. *Text mining* digunakan untuk mengolah teks atau *preprocessing* didalam dokumen sebagai kata kunci dan perhitungan *termfrequency – inverse document frequency* (*TF-IDF*) sebagai metode pembobotan setiap kata dalam dokumen sesuai dengan kata kunci yang diinputkan pengguna. *TF-IDF* dipengaruhi oleh frekuensi kemunculan kata pada sebuah dokumen dan frekuensi dari dokumen yang memiliki kata tersebut sehingga jika diimplementasikan sistem ini dapat menemukan kembali informasi dari dokumen laporan penelitian yang disimpan secara cepat dan efisien, serta dari hasil pencarian dapat diurutkan berdasarkan bobot informasinya.Hasil dari penelitian ini menunjukkan bahwa pembobotan kata dengan menggunakan TF-IDF dapat me-*retrieve* dokumen yang relevan dengan query masukan pengguna.

Kata kunci :*information retrieval, text mining, TF-IDF*, pencarian

3. Pendahuluan

Seiring dengan perkembangan informasi, masalah utama telah bergeser dari cara mengakses informasi menjadi memilih informasi utama yang berguna secara selektif. Pencarian atau pemilihan informasi ini tidak mungkin dilakukan secara manual karena kumpulan informasi yang sangat besar, banyak, dan beragam. Dibutuhkan suatu sistem otomatis untuk membantu *user* (pengguna) dalam menemukan informasi. *Search engine* (mesin pencari) dapat mengatasi permasalahan tersebut secara efektif. Setiap search engine menggunakan *proprietary algorithm* untuk menciptakan indeks-indeks yang ditampilkan dalam bentuk hasil pencarian (Sarwono, 2010:1).

Sistem pencarian dokumen umumnya menampilkan hasil pencarian berdasarkan kata kunci (*keyword*s) dan peringkatnya yang ditampilkan dalam daftar yang panjang. Sebagian search engine masih menggunakan metode tersebut dan memiliki karakteristik pencarian dokumen yang memiliki tingkat kecocokan rendah. Pencarian dokumen teks merupakan permasalahan yang mendasar dan penting. Didalam dokumen teks, tulisan yang terkandung merupakan bahasa dengan struktur yang kompleks dan memuat jumlah kata yang banyak. Dari permasalahan tersebut dikembangkanlah suatu ilmu yang diberi nama temu-kembali informasi *(information retrieval).*

Temu-kembali informasi berhubungan dengan penyimpanan, struktur dan akses dari dokumen-dokumen yang bertujuan untuk memudahkan pencarian suatu informasi. Representasi dari dokumen itu diharapkan harus mudah diakses oleh pengguna untuk mendapatkan informasi. Tujuan dari penelitian ini adalah mencoba menerapkan konsep temu-kembali informasi yang di terapkan di dalam sebuah sistem penyimpanan dokumen teks berbasis web. Dengan menerapkan konsep tersebut, diharapkan sistem dapat melakukan pencarian dokumen berdasarkan informasinya secara cepat dan mengetahui tingkat akurasi hasil pencarian dengan metode pembobotan kata *TF-IDF* (*TermFrequencyInverse DocumentFrequency*).

4. Metode

4.1 *Information Retrieval* (Temu-Kembali Informasi)

Temu-kembali informasi atau *Information Retrieval (IR)* adalah aktifitas utama yang dilakukan oleh sebuah penyedia informasi atau pusat pelayanan informasi, *term*asuk perpustakaan dan jenis dari layanan lainnya yang menyediakan informasi kepada masyarakat umum. Menurut sebuah ensiklopedia, temu-kembali informasi adalah seni dan ilmu dalam pencarian informasi di sekumpulan dokumen-dokumen, pencarian informasi di dokumen itu sendiri, pencarian metadata yang menjelaskan sekumpulan dokumen, atau pencarian di dalam basis data (Wikipedia, 2010). Temu kembali informasi

berhubungan dengan representasi informasi, data penyimpanan, pengorganisasian, dan akses untuk informasi tersebut (Baeza-Yates, 1999). Nantinya hasil akhir dari temu-kembali informasi adalah sebuah sistem yang dapat melakukan penemuan-kembali informasi atau disebut sistem temu-kembali informasi (Nadirman,2006).

4.2 *Text Mining*

Text Mining adalah suatu proses yang bertujuan untuk menemukan informasi dengan memproses dan menganalisa data dalam jumlah yang besar. Dalam menganalisa sebagian atau keseluruhan *unstructured text*, *text mining* mencoba untuk mengasosiasikan satu bagian teks dengan yang lainnya berdasarkan aturan tertentu. *Text mining* memiliki definisi menambang data yang berupa teks dimana sumber data biasanya didapatkan dari dokumen, dan tujuannya adalah mencari kata-kata yang dapat dilakukan analisa keterhubungan antar dokumen (Barakbah, 2013).

4.3 Pembobotan *TF-IDF*

Baeza-Yates dan Ribeiro-Neto (1999), menyebutkan bahwa pembobotan *TF-IDF* terdiri dari dua faktor, yaitu:

1. *TF (termfrequency)*

   TF adalah frekuensi kemunculan suatu istilah f_i di dalam sebuah dokumen $d_j$dibandingkan dengan frekuensi istilah f_l yang sering muncul pada dokumen itu. Jika dimasukkan dalam rumus matematika didapatkan:

$$f_{i,j} = \frac{freq_{i,j}}{max_l\, freq_{l,j}}$$

Gambar 1. Rumus *term frequency*

2. *IDF (inverse document frequency)*

   *IDF*adalah frekuensi kemunculan suatu istilah f_i di dalam seluruh dokumen. Penggunaan faktor *IDF* didasarkan pada istilah yang muncul pada setiap dokumen tidak memberikan suatu ciri khusus untuk menentukan dokumenyang relevan dari yang tidak relevan. Jika jumlah seluruh dokumen di dalam sistem dinyatakan dengan nilai *N* dan jumlah dokumen yang memiliki istilah f_i tersebut dinyatakan dengan ni, maka nilai IDF_i-nya dapat dinyatakan dengan:

$$idf_i = log\frac{N}{n_i}$$

Gambar 2. Rumus *inverse document frequency*

Keterangan :

IDF= inverse document frequency

N = jumlah kalimat yang berisi *term*(t)

n_i= jumlah kemunculan kata (*term*) terhadap d_j

Faktor pembobotan untuk tiap kata dalam dokumen didefinisikan sebagai kombinasi *termfrequency* dan *inverse document frequency*. Dari dua faktor tersebut maka pembobotan *TF-IDF* dapat dinyatakan dengan:

$$w_{i,j} = f_{i,j} \times log\frac{N}{n_i}$$

Gambar 3. Rumus *TF-IDF*

$$w_{ij} = tf_{ij} \times idf_j$$

Gambar 4. Rumus pembobotan *TF-IDF*

Keterangan:

w_{ij} = nilai bobot kata ke *j* dari dokumen *i*

tf_{ij} = *termfrequency*, yakni jumlah kemunculan kata t_j dalam dokumen D_i

df_j = document frequncy, yakni jumlah dokumen yang mengandung t_j

IDF_j = $log\ (\frac{d}{ni})$ dengan d adalah jumlah semua dokumen dalam koleksi. IDF_j adalah inverse document *frequency* (Anistyasari dkk, 2012).

Pada Metode ini pembobotan kata dalam sebuah dokumen dilakukan dengan mengalikan nilai *TF* dan *IDF*. Pembobotan diperoleh berdasarkan jumlah kemunculan *term* dalam kalimat (*TF*) dan jumlah kemunculan *term* pada seluruh kalimat dalam dokumen (*IDF*). Bobot suatu istilah semakin besar jika istilah tersebut sering muncul dalam suatu dokumen dan semakin kecil jika istilah tersebut muncul dalam banyak dokumen (Fatkhul, 2011).

Kemudian baru melakukan proses pengurutan (*sorting*) nilai kumulatif dari W untuk setiap kalimat. Tiga kalimat dengan nilai W terbesar dijadikan sebagai hasil dari ringkasan atau sebagai output dari peringkasan teks otomatis (Sarno, dkk, 2012).

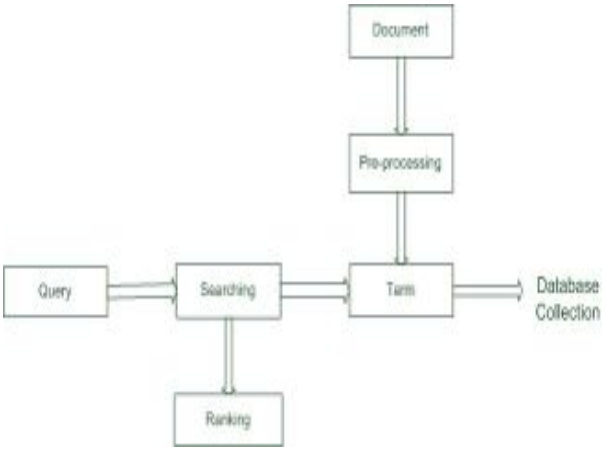
2.4 Rancangan Sistem

Fokus penelitian ini adalah bagaimana mengimplementasikan sistem pencarian dokumen secara otomatis dan efisien. Sistem temu kembali informasimemiliki dua tahapan besar, yaitu melakukan *preprocessing* terhadap *database* dan kemudian menerapakan metode tertentu untuk menghitung relevansi antara dokumen di dalam *database* yang telah di *preprocess* dengan *query* pengguna.

Proses *preprocessing* meliputi proses tokenisasi atau pemotongan kata dalam suatu kalimat,

filtering atau proses penyaringan kata hasil tokenisasi dimana kata yang tidak relevan dibuang seperti stoplist, *stemming* merupakan proses memecahkan setiap imbuhan suatu kata menjadi kata dasar, dan *termweighting* atau pembobotan kata.

Stemming adalah tahapmencari kata dasar dari setiap kata hasil *filtering*. Digunakan untuk mereduksi bentuk *term* untuk menghindari ketidakcocokan sehingga dapat mengurangi *recall*. Pembobotan kata (*termweighting*) dilakukan dengan menghitung frekuensi kemunculan *term* dalam dokumen. Frekuensi kemunculan (*termfrequency*) merepresentasikan isi dari suatu dokumen. Semakin besar kemunculan suatu *term* dalam dokumen akan memberikan nilai kesesuaian yang semakin besar. Selanjutnya akan dilakukan perangkingan berdasarkan bobotnya, dimana bobot yang tertinggi adalah yang dapat diasumsikan sebagai hasil pengujian.

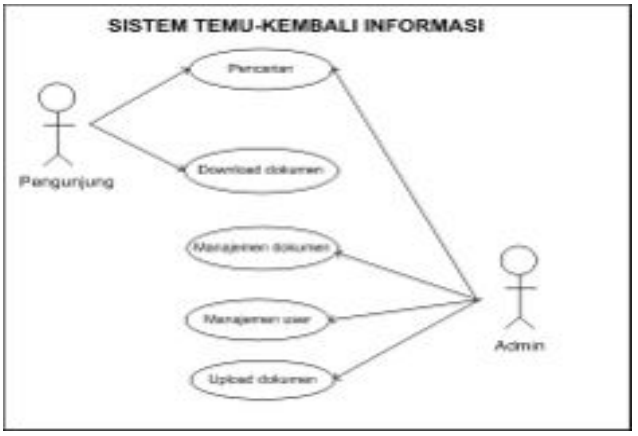


Gambar 5. *Block Diagram IR*

2.5 Pemodelan Sistem

Use Case Diagram

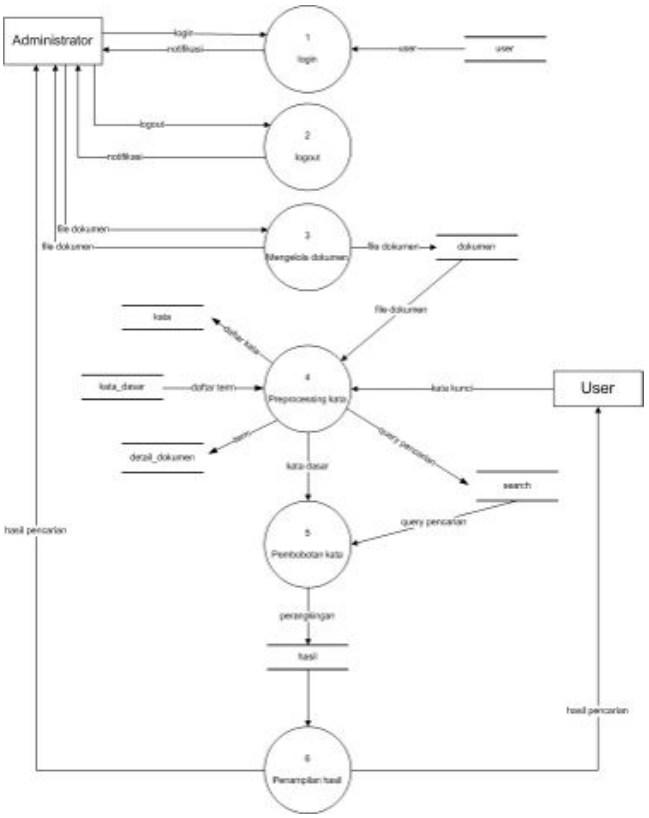
Pada sistem dua aktor yaitu *user* dan administrator. Administrator melakukan login sesuai dengan hak yang diperolehnya. Kemudian admin dapat meng-*upload* dokumen laporan penelitian dalam bentuk .docx. Admin dapat mengolah dokumen yang ada pada *database*. *User* memiliki hak untuk memasukkan kata kunci pada kolom pencarian untuk melakukan pencarian dokumen laporan penelitian dan mengunduh dokumen laporan tersebut.



Gambar 6. *Use Case Diagram*

Data Flow Diagram

Gambar 7. Diagram Konteks



Gambar 8. DFD Level 1

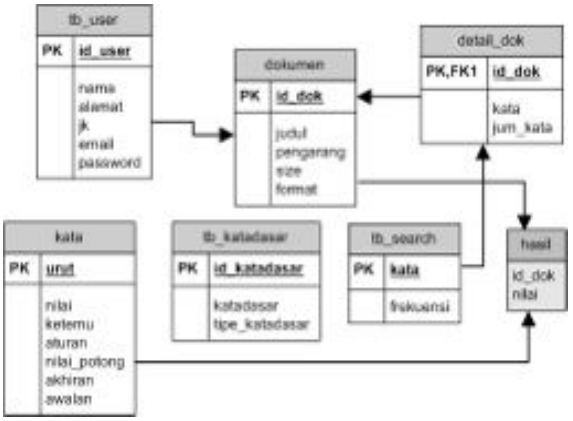
2.7 Rancangan *Interface*

Pada rancangan *interface*, terdapat menu Home yang menampilkan kolom pencarian dan kolom hasil pencarian. *User* memasukkan kata kunci pada kolom pencarian

yang menjadi dasar dalam perhitungan untuk merangking dokumen yang paling relevan.Data dokumen memuat arsip dokumen yang terdapat pada *database*.

5. Hasil

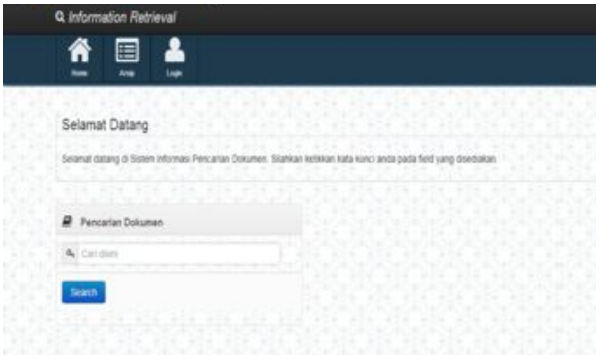
5.1 Implementasi Basis Data



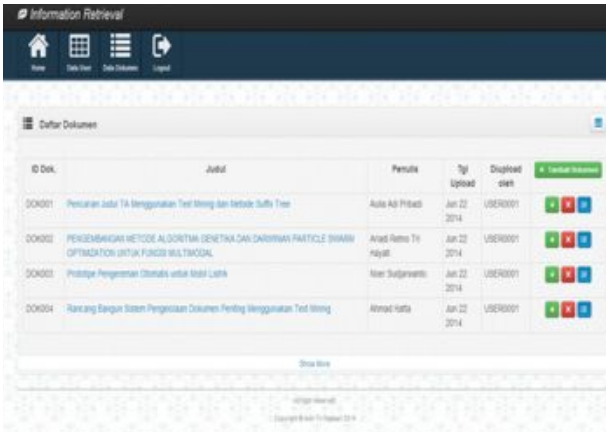
Gambar 9. *Database* Diagram Aplikasi IR

3.2 Implementasi Antarmuka

Berikut merupakan tampilan dari Aplikasi *Information Retrieval*. Terdapat menu Home yang berisi kolom pencarian, Arsip untuk melihat koleksi dokumen, Login untuk administrator.



Gambar 10. Tampilan Home Aplikasi IR



Gambar 11. Tampilan Data Dokumen

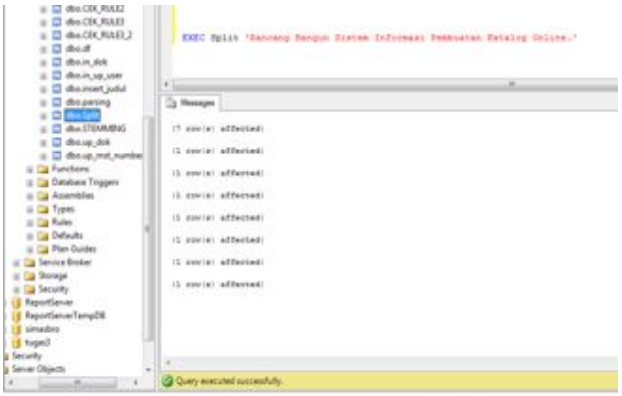
Gambar 12. Tampilan Data *User*

6. Pembahasan

6.1 Proses Tokenisasi

Hasil implementasi information retrieval saat proses *preprocessing* dokumen dan pembobotan *TF-IDF* adalah me-*retrieve* dokumen atau menemukan kembali dokumen yang diinginkan. Berikut merupakan pengujian tokenisasi dan *filtering* dengan memberikan inputan kalimat yang mengandung tanda baca dan kata yang *term*asuk stoplist.

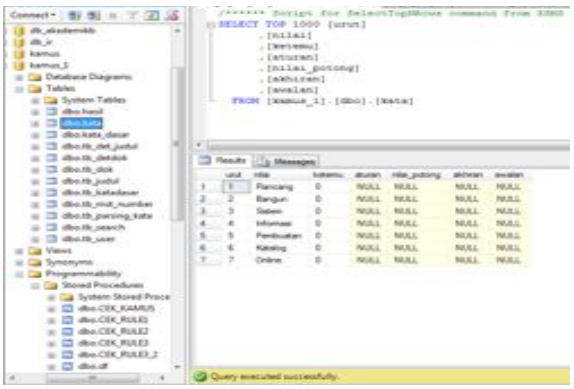
Teks Awal : "Rancang Bangun Sistem Informasi Pembuatan Katalog Online."



Gambar 13. Hasil Pengujian Tokenisasi

6.2 Proses *Filtering*

Teks Awal : "Rancang Bangun Sistem Informasi Pembuatan Katalog Online."

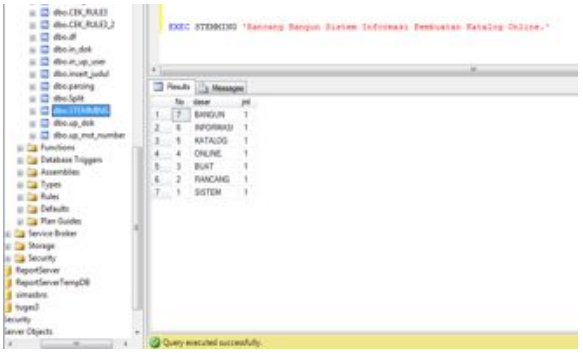


Gambar 14. Hasil Pengujian *Filtering*

6.3 Proses *Stemming*

Kalimat dalam Bahasa Indonesia sering tersusun dari kata-kata yang berimbuhan. *Stemming* merupakan sebuah proses pengembalian kata-kata tersebut ke bentuk dasar. Pada pengujian ini, metode stemming yang digunkan adalah algoritma stemming Nazief-Adriani dan akan diberikan inputan suatu kalimat yang tersusun dari kata berimbuhan. Hasil yang benar pada pengujian ini adalah jika kata berimbuhan dalam kalimat inputan berubah ke bentuk dasar.

Pada **Gambar 15** menunjukkan bahwa kata-kata yang sudah di tokenisasi dan di *filtering* dikembalikan ke bentuk dasar.

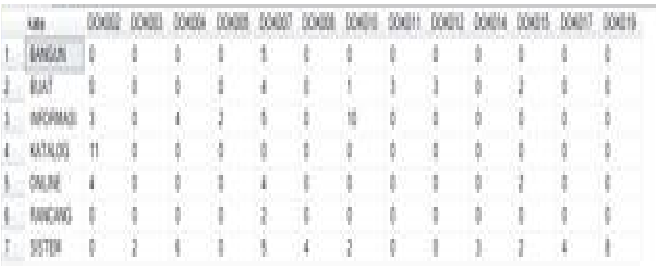


Gambar 15. Hasil Pengujian *Stemming*

6.4 Pembobotan *TF-IDF*

a. Perhitungan *TF*

Kata kunci : Rancang Bangun Sistem Informasi Pembuatan Katalog Online



Gambar 16. Hasil Perhitungan *tf*

Setelah hasil perhitungan *tf* didapatkan, langkah selanjutnya adalah perhitungan *idf* tiap *term* untuk menghitung bobot *term*.

b. Perhitungan *IDF*

Rumus *idf*:

$$idf = (1/df)$$

Gambar 17. Hasil Perhitungan *idf*

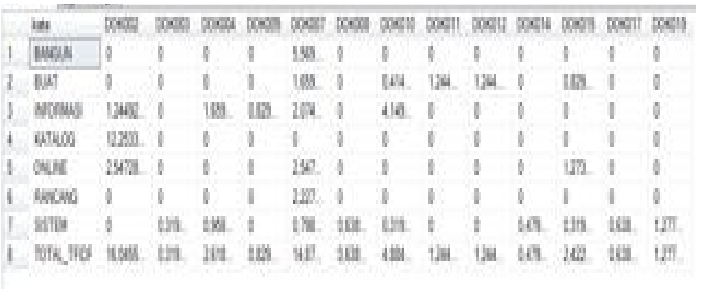
Selanjutnya, setelah nilai *tf* dan *idf* telah didapatkan, kemudian dimasukkan dalam perhitungan *tf-idfweighting* untuk menghitung bobot hubungan suatu *term* di dalam dokumen.

c. Perhitungan *TF-IDF*

Rumus pembobotan *tf-idf*:

$$W_{t,d} = tf_{t,d} * idf_t$$

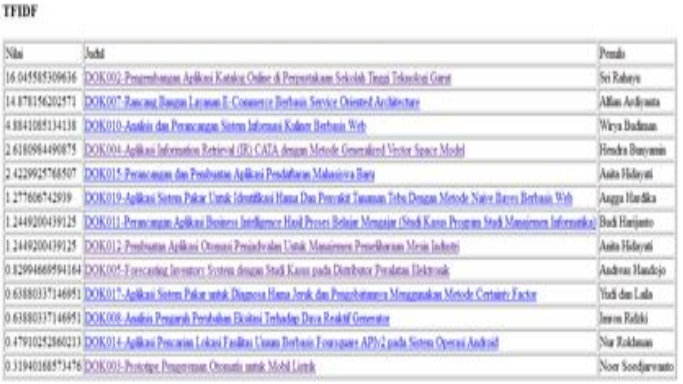
W = bobot dokumen ke-d terhadap kata ke-t



Gambar 18. Hasil Perhitungan *TF-IDF*

d. Hasil Aplikasi

Aplikasi menemu-kembalikan dokumen yang relevan dengan kata kunci masukan pengguna dalam sebuah daftar dokumen.



Gambar 19. Halaman Hasil Pencarian

Dokumen yang memiliki bobot paling tinggi adalah DOK002 dengan total bobot TF-IDF sebesar 16,04. Selain itu, aplikasi juga menemu-kembalikan dokumen yang relevan dengan kata kunci masukan.

7. Kesimpulan dan Saran

7.1 Kesimpulan

Setelah melalui tahap implementasi dan uji coba, maka dapat ditarik kesimpulan:

1. Sistem temu-kembali informasi yang dibuat dapat mencari informasi dari isi file dokumen yang disimpan di dalam sistem. Dokumen di dalam sistem temu-kembali informasi yang dikembangkan melalui beberapa tahapan prapemrosesan dokumen, yaitu tokenisasi, *filtering*, dan penggunaan *stemming.*
2. Pada proses pencarian agar sistem dapat menemu-kembalikan dokumen yang relevan maka, kata kunci melalui beberapa tahapan pemrosesan yaitu tokenisasi, *filtering*, *stemming*, dan pembobotan kata dengan menggunakan metode *tf-idf* untuk mendapatkan rangking berdasarkan nilai bobot setiap dokumen yang akan dicari.

7.2 Saran

Saran penulis yang diusulkan untuk penelitian dan pengembangan sistem temu-kembali informasi berikutnya yaitu, dalam temu-kembali informasi, belum cukup hanya mendapatkan dokumen yang relevan. Sistem harus dapat mendapatkan dokumen relevan dan tidak mendapatkan dokumen yang tidak relevan, maka sistem perlu dilakukan pengembangan setelah melakukan pembobotan kata yaitu pengurutan dokumen dengan menggunakan metode model ruang vektor dan atau model probabilistik.

8. Daftar Rujukan

Amin, Fatkhul. 2011: Implementasi *Search Engine* (Mesin Pencari) Menggunakan Metode *Vector Space Model*, Vol. V No. 1, Hal 45-48, Dinamika Teknik.

Baeza-Yates, Ricardo., Ribeiro-Neto, Berthier. 1999 : *Modern Information Retrieval,* ACM Press Books, New York.

Barakbah, Ali Ridho.2013. *Text Mining*. EEPIS-ITS, Surabaya.

Hatta, Ramadijanti. N, Helen. A, 2010: Rancang Bangun Sistem Pengelolaan Dokumen-Dokumen Penting Menggunakan *Text Mining*.

Nadirman, Firnas. 2006: Sistem Temu-Kembali Informasi dengan Metode Vector Space Model pada Pencarian File Dokumen Berbasis Teks. Yogyakarta, Indonesia: Universitas Gadjah Mada.

Sarno, R., Anistyasari, Y. dan Fitri, R. 2012: *SEMANTIC SEARCH* Pencarian Berdasarkan Konten, Penerbit ANDI, Yogyakarta.

Sarwono, Jonathan. 2010: *SEARCH ENGINE*, Penerbit ANDI, Yogyakarta.

Wikipedia, 2014, Sistem Temu-Kembali Informasi, EnsiklopediaBebas http://id.wikipedia.org/wiki/Sistem_temu_balik_informasi Wikipedia [diakses pada 10 Mei 2014]