

PENCARIAN INFORMASI SKRIPSI PADA PUSTAKA FAKULTAS ILMU KOMPUTER UNIVERSITAS LANCANG KUNING

Indi Rahman

Program Studi Sistem Informasi Fakultas Ilmu Komputer Universitas Lancang Kuning

Jl. Yos Sudarso KM. 8 Rumbai, Pekanbaru, Riau, telp. 0811 753 2015

e-mail: indi.rahman@gmail.com

ABSTRAK

Pada beberapa tahun terakhir, seiring dengan peningkatan jumlah koleksi dokumen yang berisikan informasi elektronik kian banyak, kebutuhan untuk menemukan informasi yang relevan dalam bentuk digital semakin penting. Salah satu cara yang cukup populer untuk menemukan informasi tersebut adalah dengan menggunakan operator *basic boolean*, yaitu AND, OR dan NOT. Penerapan operator ini membantu dalam mendefinisikan kriteria pencarian dan mengendalikan logika pencarian. Mekanisme standard untuk mendukung query *basic boolean* adalah dengan menggunakan file inverted indek yang dikelompokkan berdasarkan nilai hexa desimalnya, digunakan langkah parsing, remove stopword, stemming dan weighting atau pembobotan.

Penerapan *basic boolean* untuk pencarian Skripsi pada Ilmu Komputer Universitas Lancang Kuning Pekanbaru, melakukan pencarian string query yang diinputkan terhadap data term inverted. Kemudian memberikan respon kepada user berupa judul, uraian singkat dan alamat dokumen sesuai dengan logika pencarian yang diinginkannya, serta mengurutkan dokumen yang relevan tersebut berdasarkan ranking dari term itu sendiri.

Kata kunci : *Basic boolean, retrieval, inverted index, parsing, remove stopword, stemming, weighting.*

ABSTRACT

In recent years, growth-up amount document collection which comprising electronic data information increasingly, requirement to find relevant information data digital progressively to be important. One of way popular to find the information is using boolean basic operator, that is AND, OR and NOT. Standard mechanism to support boolean basic query is using inverted index file. For the extract document and make the tables of term inverted index is grouping term value in hexa decimal, used by parsing step, remove stopword, stemming and weighting.

Applying basic boolean for the seeking Skripsi in Computer Science in Lancang Kuning University, conducting seeking of query string which is input from user to term inverted data. Then give respon to user in the form of title, brief description and document address as according to wanted seeking logic, and also sort the relevant document pursuant to ranking from itself term.

Keywords : *Basic boolean, retrieval, inverted index, parsing, remove stopword, stemming and weighting.*

I. PENDAHULUAN

Pada beberapa tahun terakhir, seiring dengan peningkatan jumlah koleksi dokumen elektronik Skripsi Mahasiswa kian banyak di Perpustakaan Fakultas Ilmu Komputer Universitas Lancang Kuning. Kebutuhan untuk menemukan informasi yang relevan dalam bentuk digital yang cepat dan efisien semakin penting.

Pencarian informasi pada umumnya dapat dirasakan dalam permasalahan ketika akan memilih teks dokumen dari suatu database yang besar. Dari segi pandangan ini, isu yang utama berhubungan dengan pemanggilan dan penyajian teks, query, dan teknik untuk membandingkan

dan bagaimana menyajikan query tersebut ke pengguna. Oleh karena itu diperlukan suatu solusi untuk mencari teks yang diperlukan oleh pengguna secara cepat dan mudah dengan menggunakan operator basic boolean dari suatu peta data.

Penggunaan Operator logika basic boolean "AND", "OR" dan "NOT" sangat membantu para pemakai untuk mendefinisikan kriteria pencarian dan mengendalikan logika pencarian tersebut. Semenjak operator logika Boolean diberlakukan bagi perhitungan persamaan jarak, teknik ini dapat dengan mudah diterapkan pada banyak sistem pencarian (Suzuki, Yaginuma dan, Yuji, 2003 ; hal 1).

Untuk memecahkan waktu yang terbuang sia-sia dan masalah pengenalan terhadap materi secara berulang-ulang setiap kali proses pencarian dilakukan, kita dapat menggunakan cara indeks sebagai alat untuk menempatkan informasi yang diinginkan. Indeks disiapkan bagi pemakai untuk membantu mereka di dalam menemukan informasi dengan cepat dan mudah. Suatu indeks bukan sekedar daftar terminologi yang utama di dalam suatu penerbitan. Melainkan, suatu peta dokumen yang terorganisir dan teratur. Salah satu cara melakukan indeks yang banyak diterapkan adalah inverted indeks (Medelyna, 2006 ; hal 2).

Berdasarkan uraian di atas, maka dibutuhkan pembangunan sebuah system untuk pencarian informasi skripsi mahasiswa yang tersimpan di komputer pada Perpustakaan Fakultas Ilmu Komputer Universitas Lancang Kuning Pekanbaru.

2. METODE PENELITIAN

2.1. Pencarian Informasi

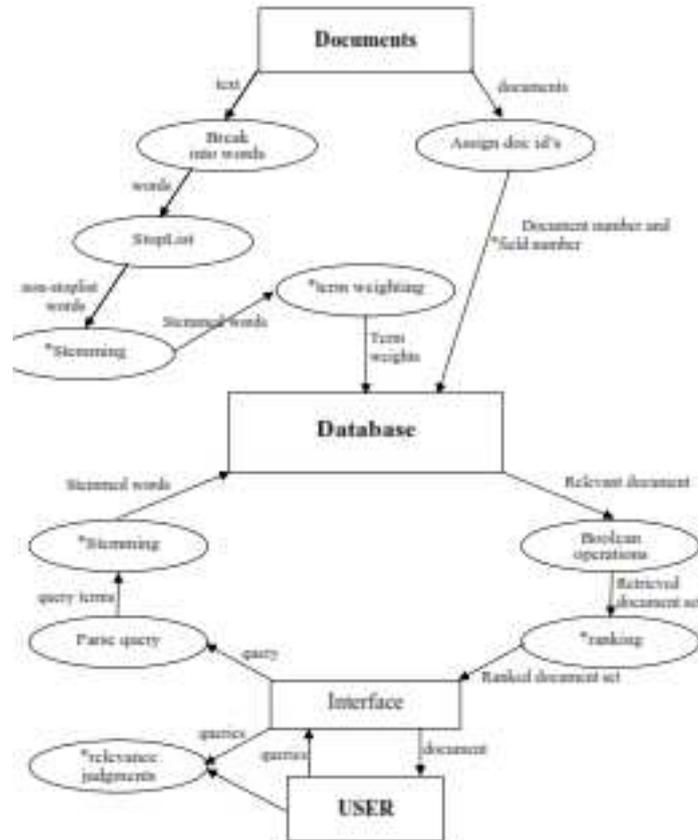
Sistem pencarian informasi pada prinsipnya adalah suatu sistem yang sederhana. Misalkan ada sebuah kumpulan dokumen dan seorang user yang memformulasikan sebuah pertanyaan (*request* atau *query*). Jawaban dari pertanyaan tersebut adalah sekumpulan dokumen yang relevan dan membuang dokumen yang tidak relevan. Secara matematis hal tersebut dapat dituliskan sebagai $Q \xrightarrow{2^n} D$, dimana Q = pertanyaan (queri), D = dokumen, n = jumlah dokumen, 2^n = jumlah kemungkinan himpunan bagian dari dokumen yang ditemukan. Sistem temu-kembali akan mengambil salah satu dari kemungkinan tersebut (Hasibuan dan Andri, 2007).

Sistem pencarian informasi merupakan sistem yang berfungsi untuk menemukan informasi yang relevan dengan kebutuhan pemakai. Salah satu hal yang perlu diingat adalah bahwa informasi yang diproses terkandung dalam sebuah dokumen yang bersifat tekstual. Dalam konteks ini, temu kembali informasi berkaitan dengan representasi, penyimpanan, dan akses terhadap dokumen representasi dokumen. (Hasibuan dan Andri, Yofi., 2007).

Sparck Jones 1997 menyatakan bahwa pencarian dokumen dikelompokkan pada dua aktivitas yang saling berkaitan satu dengan lainnya, yaitu: indeks dan pencarian. Indeks mengacu pada dokumen itu sendiri, yaitu informasi yang akan dipanggil, dan pencarian, yaitu pernyataan dari pemakai yang membutuhkan informasi, dengan tujuan untuk menampilkan informasi yang diinginkan (Rhodes, 2000; hal 45).

Baeza dan Ribeiro (1999 ; 10), menjelaskan bahwa untuk menguraikan proses pencarian informasi tersebut dapat dipahami dengan suatu arsitektur perangkat lunak yang umum dan sederhana. Pertama, sebelum sistem proses pencarian informasi digunakan, perluk untuk menguraikan database teks. Pada umumnya dilaksanakan oleh manajer database, yang menetapkan : (a) dokumen untuk digunakan (b) operasi untuk dilakukan pada teks dan (c) teks model (yaitu. struktur teks dan unsur-unsur apa yang dapat didapat kembali). Operasi Teks mengubah bentuk dokumen asli dan menghasilkan suatu tampilan.

Frakes (2005;15) menyatakan poses pencarian informasi dalam metode melakukan ekstrak dokumen ke dalam database dengan melalui beberapa filter yaitu stoplist, stemming dan kemudian dilakukan pembobotan. Proses stoplist, stemming dan pembobotan disebut sebagai proses pengindeksan. Selanjutnya database tersebut baru akan digunakan oleh sistem pencarian. Proses ini disebut sebagai Query. Untuk lebih jelasnya dapat dilihat pada gambar berikut,



Gambar 1 : Sistem Pencarian Informasi

Ketika membangun database, dokumen diambil satu persatu, dan teks yang terdapat di dalamnya akan dijadikan dalam satuan unit kata. Kata dari dokumen tersebut selanjutnya diuji dengan satu perangkat daftar kata henti (stopword list) yang tidak akan diberikan nilai indeks. Seluruh kata di dalam dokumen yang tidak ditemukan pada stoplist seterusnya akan dibendung (stemmed). Kata tersebut kemudian akan dihitung, yaitu bobot kata-kata di dalam dokumen dan di dalam database secara keseluruhan sering digunakan untuk mengatur dokumen didapat kembali.

Kata dan seluruh yang berhubungan dengan informasi dokumen, bidang di dalam dokumen, dan jumlah yang disimpan pada database. Database kemudian terdiri dari pengenalan dokumen (alamat dokumen) dan term. Struktur seperti itu disebut sebagai file dibalikkan (inverted file). Di dalam suatu sistem pencarian informasi, masing-masing dokumen harus memiliki sesuatu pengenalan yang unik, dan field nya, jika operasi field didukung, harus mempunyai nama field yang unik dan telah dilakukan indeks.

Dalam matematika, indeks merupakan notasi yang menunjuk pada unsur tertentu dalam suatu susunan unsur-unsur. Secara umum indeks adalah petunjuk yang sistematis kepada satuan-satuan yang terkandung di dalam, atau konsep yang diturunkan dari koleksi entitas atau basis data.

Menurut Lee (2006) Tidak ada mesin pencari sempat melakukan scan secara teliti terhadap koleksi dokumen. Sebagai gantinya, penting membuat index. Pekerjaan ini merupakan pemetaan dari masing-masing istilah berbeda ke daftar dokumen yang bermuatan tema. Arsitektur pembuat indeks berisi object kunci memiliki StopWordList, Stemming Tokenizer, InvertedIndex, dokumen, Parser. Algoritma inverted index menggunakan suatu jumlah ditetapkan bagi memori dan membangun sebanyak mungkin indeks memori yang memungkinkan dan kemudian disimpan ke dalam disk ketika memori telah penuh.

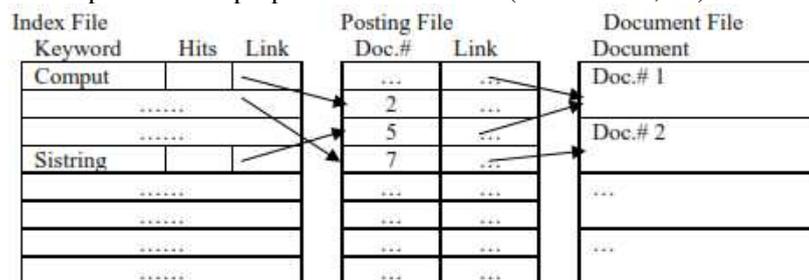
Parser melakukan *scan* terhadap file input (masukan) dan menghasilkan satu set objek dokumen. Suatu obyek dokumen berisi semua informasi yang tersusun dan unik untuk masing-masing dari dokumen. Parser menggunakan tokenizer untuk mengidentifikasi tanda unik di

dalam dokumen dan frekuensi istilah dari tanda ini. Diperlukan untuk mengatur keterkaitan (Lee, 2006).

File masukan diperoleh dari parameter yang diteruskan ke index dan file dikirim ke parser untuk diuraikan. Ketika dokumen diuraikan, suatu kejadian parsing mengembalikan daftar dokumen object ke parser. Sebagai dokumen object yang datangnya tidak secara bersamaan, pembuat index mengirimkan daftar object dokumen ke metoda tambahan di dalam *InvertedIndex*. Selanjutnya pembuat index memanggil metode tulis dalam *InvertedIndex* yang melakukan pencatatan inverted indeks ke disk (untuk suatu koleksi dokumen cukup besar, ini mungkin mengakibatkan menggabungkan beberapa file temporer membangun banyak panggilan untuk menambahkan suatu dokumen) (Lee, 2006).

Suatu inverted file adalah seperti file indeks. Struktur dari suatu masukan file inverted pada umumnya menggunakan term, document-ID, field-ID. Suatu kata kunci merupakan suatu indeks term yang bersumber dari dokumen yang diekstrak, document-ID adalah suatu unik identifier untuk suatu dokumen, dan field-ID adalah suatu nama unik yang menandai field di dalam dokumen kata kunci. (Frake, 12).

Konsep file *invert* type indeks sebagai berikut. Asumsikan satu set dokumen. Masing-Masing dokumen diberikan daftar term atau kata kunci atau atribut, dengan pilihan yang memiliki hubungan dengan kata kunci masing-masing. Suatu file invert merupakan daftar yang telah diurutkan (atau indeks) berdasarkan term (atribut), dengan masing-masing kata kunci memiliki hubungan dengan dokumen yang berisi kata kunci tersebut. Ini adalah seperti menemukan indeks pada sistem perpustakaan komersil (Harman etc, 38).

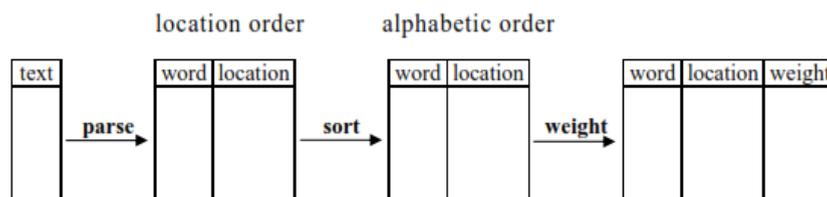


Gambar 2 : Implementasi Inverted Dengan Array Terurut

Penting untuk melakukan pembatasan bobot indeks yang akan dilakukan tersebut dan sebagai konsekuensi pada pencarian kemudiannya. Bentuknya adalah pengendalian kosa kata, yang merupakan koleksi dari term atau kata kunci yang akan di indeks. Sedangkan daftar stopwords (artikel, preposisi, dll.) sebagai bahan pertimbangan untuk ukuran ketepatan dan daya ingat tidak akan tercakup di indeks, dan juga diperlukan perangkat aturan yang memutuskan permulaan suatu kata atau suatu potongan teks yang dapat indeks. Aturan ini berhadapan dengan spasi, tanda baca, atau beberapa awalan baku, dan mungkin punya dampak penting pada term indeks.

Suatu pencarian di dalam suatu file inverted terdapat dua komposisi algoritma pencarian ; suatu pencarian untuk suatu term (kata kunci), yang dikembalikan sebagai index, dan kemudian suatu pencarian mungkin pada index itu sendiri untuk atribut tertentu. Hasil suatu pencarian pada file inverted adalah satu set record (atau pointer ke record).

File yang di inverted diperlakukan sebagai struktur array menyimpan daftar dengan jenis term di dalam suatu array, mencakup banyaknya dokumen berhubungan dengan masing-masing term dan suatu mata rantai kepada dokumen yang berisi term tersebut. Untuk menghasilkan jenis array file *inverted* dapat dibagi menjadi dua atau tiga langkah seperti yang ditampilkan lebih lanjut pada gambar berikut. Pertama, teks yang dimasukkan harus diuraikan ke dalam daftar kata-kata, bersamaan dengan penempatan mereka di dalam teks.



Gambar 3 : Skema Pembuatan Inverted dan mengurutkannya

Pentingnya weights (pembobotan) untuk term dokumen dan query term, dapat membantu mengenali term yang berguna bagi retrieval. Weights dapat ditambahkan ke file inverted. Dokumen berbeda mempunyai persamaan yang berbeda dalam query dan dokumen dapat di ranking pada waktu *retrieval* di dalam mengurangi persamaan (Lu, 78)

Formula yang biasanya digunakan untuk menghitung weight dari term biasa disebut dengan tf.idf, yaitu :

$$W_{ij} = tf_{ij} * \log (N/df_i)$$

Keterangan :

W_{ij} adalah weights dari term j di dalam dokumen i.

tf_{ij} adalah frekuensi term j di dalam dokumen i

N adalah total jumlah koleksi dokumen

df_i adalah jumlah dokumen yang memiliki term j

Formula ini selanjutnya akan dipergunakan dalam memberikan weight untuk setiap term yang akan disimpan pada database retrieval. Meskipun banyak percobaan lain yang telah dilakukan untuk term weights ini, seperti Dumais (1992) dengan entropy weighting, Pyle (1999) dengan TfIDf yang menggunakan fungsi logistik dan lain sebagainya.

Untuk mencari informasi, seorang pengguna dapat memasukan sebuah query terdiri dari satu set kata kunci yang dihubungkan oleh Operator Boolean (AND, OR, NOT). Query diuraikan ke dalam terminologi unsur nya dan Terminologi Operator Boolean. Kemudian terms akan dilihat di dalam file inverted dan daftar pengenalan dokumen sesuai dengan kombinasinya menurut Operator Boolean yang telah ditetapkan. Jika frekwensi informasi telah dikenali, memungkinkan untuk menyusun pengaturan pencarian informasi menurut keterkaitannya. Hasil pencarian kemudian diberikan kepada pengguna. Dalam beberapa sistem, pengguna membuat pertimbangan tentang keterkaitan dokumen yang di cari, dan informasi ini digunakan untuk memodifikasi query.

Model Boolean dalam retrieval merupakan model yang paling sederhana. Model ini berdasarkan teori himpunan dan aljabar Boolean. Dokumen adalah himpunan dari istilah (term) dan query adalah pernyataan Boolean yang ditulis pada term. Dokumen diprediksi apakah relevan atau tidak. Model ini menggunakan operator boolean. Istilah (term) dalam sebuah query dihubungkan dengan menggunakan operator AND, OR atau NOT. Metode ini merupakan metode yang paling sering digunakan pada mesin penelusur (search engine) karena kecepatannya (Hasibuan dan Andri, Yofi., 2007).

3. HASIL DAN PEMBAHASAN

3.1. Analisa Sistem

Sistem pencarian informasi yang dibuat memiliki kemampuan membaca file dokumen Skripsi Mahasiswa Ilmu Komputer Universitas Lancang Kuning dalam format Document (*.Doc). Selain itu sistem ini harus mampu membuat database indexing dalam bentuk tabel inverted indeks yang akan dipergunakan dalam operasi query.

Selanjutnya sistem harus memiliki fasilitas query dengan menggunakan metode basic boolean dan mampu untuk mencari kecocokan dengan database informasi retrieval. Sistem harus mampu memberikan pembobotan untuk setiap term yang disimpan pada database informasi retrieval dan dapat memanfaatkannya untuk mengurutkan tampilan dokumen relevan pada saat hasil pencarian sesuai dengan nilai bobotnya.

3.2. Prosedur Pencarian Informasi

3.2.1. Indexing

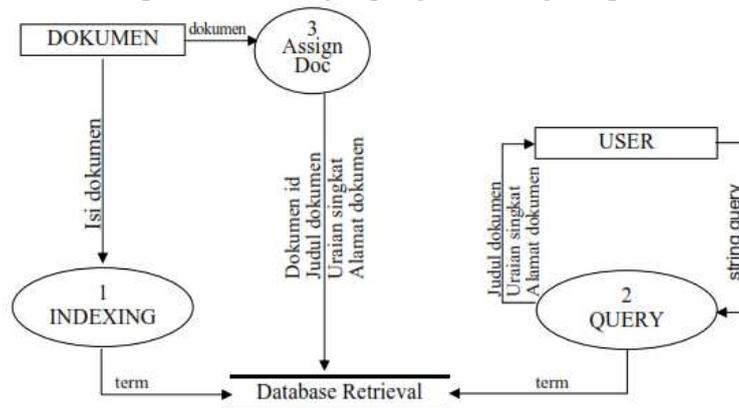
Indexing dilakukan terhadap Skripsi yang dikenali dengan langkah sebagai berikut : Pertama. dokumen yang dimiliki akan dilakukan parsing yang dilewati sebagai masukan file teks sebelum di indeks; Kedua dokumen tersebut selanjutnya akan di filter dengan menggunakan daftar stopwords berupa daftar kata-kata yang akan diabaikan dan stemming untuk menemukan kata dasar, serta melakukan weighting (pembobotan) untuk masing-masing term. Ketiga dokumen akan di indek dalam bentuk inverted indek dan akan disimpan pada database informasi retrieval yang memiliki muatan keseluruhan struktur akses sekunder yang dibangun secara efisien.

3.2.2. Query

Query harus mampu memberikan kemudahan bagi user untuk melakukan pekerjaannya dalam bentuk GUI yang mampu mendukung karakter yang akan dipergunakan dalam proses pencarian dengan menggunakan metode basic boolean. Selanjutnya query harus mampu memberikan tanggapan dalam logika operasi basic boolean dalam bentuk tampilan hasil pencarian yang diinginkan oleh user. Dalam hal ini query harus mampu menciptakan object Hasil dan melakukan pengurutan hasil tampilan (ranking) yang dibuat ketika query mengakhiri proses.

3.2.3. Basis Data

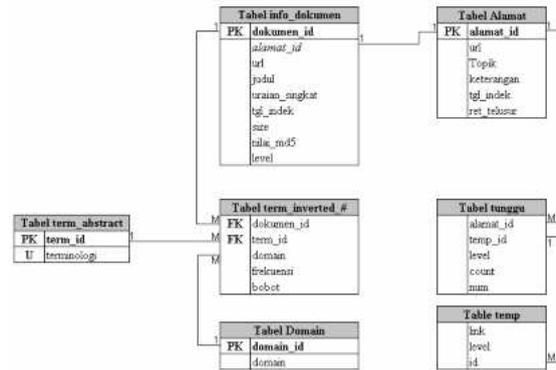
Basis data dibuat untuk menjawab serangkaian pertanyaan yang spesifik yang relevan dengan berbagai permrosesan data yang terkait dengan kebutuhan system. Dalam hal ini diuraikan dalam bentuk Diagram Alir Data yang dapat dilihat pada gambar berikut :



Gambar 4 : Diagram Alir Data Pencarian Informasi Skripsi Mahasiswa

Pada proses Indexing input berasal dari entitas dokumen yaitu berupa isi dokumen yang akan diproses. Proses indexing akan melakukan parsing, stopwords, stemming dan weighting atau pembobotan yang akan menghasilkan output berupa term, selanjutnya disimpan pada datastore database_retrieval. Pada sisi lain Input pada proses Query berasal dari eksternal entitas user berupa string_query yang akan dibandingkan untuk mencocokkan dengan datastore database_indeks yaitu term. Setelah masukan diolah pada proses query maka output yang dihasilkan ditujukan ke entitas user berupa judul dokumen, uraian singkat dan alamat dokumen.

Adapun skema Basis Data dari system pencarian informasi tersebut dapat dilihat pada gambar berikut :



Gambar 5 : Skema Basis Data Pencarian Informasi Skripsi Mahasiswa

3.3. Implementasi

3.3.1. Implementasi Sistem

Tahap implementasi sistem merupakan tahap meletakkan sistem supaya siap untuk dioperasikan pada keadaan yang sebenarnya, sehingga akan diketahui apakah sistem yang dibuat benar-benar dapat menghasilkan tujuan yang diinginkan. Adapun perangkat lunak yang digunakan adalah Sistem Operasi Microsoft Windows XP ; Apache sebagai server untuk localhost ; Bahasa pemrograman untuk pembuatan aplikasi dengan menggunakan HTML dan PHP ; Perangkat lunak untuk basis data yang akan di jalankan pada server menggunakan MySQL versi 5.

Pada implementasi program aplikasi ini menggunakan 2 tingkatan pemakai yaitu user dan administrator. User hanya memiliki hak untuk melakukan pencarian terhadap koleksi data dokumen yang dimiliki oleh sistem informasi. Sedangkan admin memiliki hak untuk dapat melakukan manipulasi database.

3.3.2. Implementasi Antar Muka

Implementasi antarmuka pemakai pada sistem pencarian informasi, mengenai penerapan perangkat lunak secara keseluruhan. Implementasi ini meliputi penerapan fungsi-fungsi yang ada pada perangkat lunak secara keseluruhan.

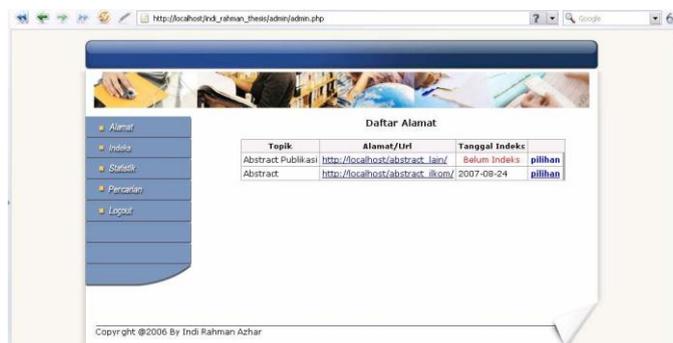
User dan administrator pertama kali akan melihat tampilan utama halaman pencarian. Pada tampilan ini user dan admin dapat melakukan pencarian langsung terhadap database retrieval yang dimiliki oleh sistem informasi retrieval.



Gambar 6 : Antarmuka Sistem Pencarian Dengan Menggunakan Metode Basic Boolean

Penandaan string query pencarian ditujukan untuk memilih salah satu operator AND, OR dan NOT jika ditemukan spasi, maka spasi tersebut akan dianggap sebagai operator basic boolean. Sedangkan untuk pencarian bebas, user dapat melakukan pencarian dengan melakukan input operator basic Boolean. Untuk operator AND dapat diinputkan dengan simbol perkalian (*), operator OR diinputkan dengan simbol penjumlahan (+) dan operator NOT dengan simbol minus (-) sebelum string query yang ingin dicari oleh user.

Sedangkan Administrator dalam melakukan dokumen untuk di indeks dapat dilihat pada gambar berikut :



Gambar 4.4 : Antarmuka Alamat

Antar muka ini ditujukan bagi administrator untuk melihat alamat pencarian yang sudah terdata. Jika diperlukan administrator dapat melakukan proses pencarian baru dengan menentukan alamat pencarian dokumen berikutnya. Administrator akan dapat melakukan edit terhadap alamat yang dimasukkan, indeks ulang untuk alamat yang telah diindeks, namun jika belum diindeks maka opsi yang keluar adalah Indeks. Selanjutnya admin dapat melihat file pada folder, menghapus dan melihat statistik dari alamat yang belum atau telah di indeks.

3.4. Testing dan Evaluasi

Setelah dilakukan proses implementasi, penulis mengindeks file data Skripsi Mahasiswa yang berjumlah 21 file dalam format Document. Sistem telah mampu untuk tidak melakukan indeks terhadap kata umum dan telah membentuk kata dasar dari tiap term yang bersumber dari dokumen tersebut. Selanjutnya setiap term telah dihitung term frekuensinya dan diberikan pembobotan dengan menggunakan formula TFIDF, dan selanjutnya term tersebut disimpan pada database retrieval.

Selanjutnya dalam pengujian terhadap string query, penulis telah melakukan 2 inputan string query dan kemudian mengujinya dengan metode basic boolean, yaitu AND, OR dan NOT. Hasil dari pengujian string query information system dalam katagori AND penulis mendapatkan hasil sebanyak 9 dokumen. Selanjutnya berturut-turut untuk operator OR mendapatkan hasil 17 dokumen dan operator NOT penulis hanya mendapatkan 4 dokumen.

4. PENUTUP

4.1. Kelebihan

Sistem mampu untuk melakukan pencarian dengan menggunakan operator basic boolean tanpa perlu melakukan tambahan kata operator itu sendiri, yaitu dengan memilih operator tersebut. Sehingga setiap pemisah (spasi) yang diinputkan oleh user dapat diartikan sebagai operator yang dipilihnya.

Sistem memiliki kemampuan untuk mengurutkan hasil pencarian string query inputan dengan berdasarkan pada penjumlahan nilai bobot. Sehingga memudahkan user untuk memilih dokumen yang lebih relevan dengan pencariannya.

4.2. Kelemahan Aplikasi

Sistem ini tidak dirancang untuk menemukan kata dalam bentuk frase. Sehingga user yang menginginkan dua kata yang berdekatan, akan kesulitan untuk menemukannya. Selanjutnya system ini tidak disiapkan untuk user yang ingin melakukan modifikasi dalam bentuk pencarian operator basic boolean yang kompleks.

String query user akan diproses jika user memasukkan string query sebanyak 3 karakter atau lebih, sehingga untuk 2 karakter string akan diabaikan oleh sistem atau dengan kata lain tidak akan diproses oleh system.

Secara umum proses database kurang efektif jika kita melakukan indek terhadap bahasa Indonesia, karena penggunaan stemming dan stopword list dalam bahasa Inggris. Karena penulis belum menemukan algoritma stemming dan stopword list dalam bahasa Indonesia.

Untuk penerapan fungsi NOT, sistem informasi retrieval mengenali NOT untuk string query kata pertama. Tujuannya untuk mengantisipasi jumlah kata yang diinputkan oleh user lebih dari 2 kata. Serta pemilihan daftar kata umum atau stopword yang tidak tepat dapat mengakibatkan user tidak akan menemukan string query yang diinputkan olehnya.

DAFTAR PUSTAKA

- [1] Baeza-Yates, R., Ribeiro-Neto, B., 1999, *Modern Information Retrieval*, Addison-Wesley, London.
 - [2] Belkin, N J., *Interaction with Texts: Information Retrieval as Information-Seeking Behavior*, Paper presented at the School of Communication, Information & Library Studies, Rutgers University, New Brunswick.
 - [3] Boerthes et. al, 2007., *Multimedia Information Services An Architectural Approach*, The work described in this paper was supported by the EURESCOM project TALMUD (Technologies and Architectures for a Leap in MULTimedia Databases) and the Dutch National BTS project SUMMER (Secure Multimedia Retrieval).
 - [4] Forman, G., 2003, *An Extensive Empirical Study of Feature Selection Metrics for Text Classification*.
 - [5] Frakes, W. B, 2005., *Stemming Algorithms*.
 - [6] Hasibuan, Z. A dan Andri, Yofi., 2007, *Penerapan Berbagai Teknik Sistem Temu-Kembali Informasi Berbasis Hiperteks*.
 - [7] Harman, D., 2005, *Inverted Files*.
 - [8] Hasan, M. M., 2001, *Cross-language Information Retrieval, Document Alignment and Visualization – A Study with Japanese and Chinese*, Ph.D. Thesis Graduate School of Information Science Nara Institute of Science and Technology.
 - [9] Lu, G. J., 1999, *Multimedia Database Management Systems*, Artech House INC, Boston
 - [10] McKinley, T., 1997, *How To Make Information Instantly Accessible*, Printed in the United States of America by GAC Shepard Poorman, Indianapolis, Indiana.
 - [11] Ralston, A., Reiley, E D., 1993. *Encyclopedia of Computer Science*, New York : Van Nostrand Reinhold
 - [12] Rhodes, B J., 2000., *Just-In-Time Information Retrieval*, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge.
 - [13] Sormunen, E., Halttunen, K and Keskustalo. H., 2002, *Query Performance Analyser - a tool for bridging information retrieval research and instruction*, Department of Information Studies, University of Tampere, Finland.
 - [14] Susmikanti, Mike., 2006, *Komputasi Komponen Terhubung Dan Jalur Terpendek Dalam Algoritma Graf Paralel*, Prosiding Semiloka Teknologi Simulasi dan Komputasi serta Aplikasi.
 - [15] Suzuki, M. T., Yaginuma, Y., Sugimoto., Y. Y., 2003., *Implementation Techniques of Boolean Logic Operators in 3D Model Search Systems*.
 - [16] Teevan, J B., 2001, *Improving Information Retrieval with Textual Analysis: Bayesian Models and Beyond*, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge.
 - [17] Wartik, S., 2005, *Boolean Operation*.
-