Perancangan Sistem Temu Kembali Informasi Menggunakan Metode Vector Space Model Pada Pencarian Dokumen Berbasis Teks Berita

Jamal Maulana Hudin[1], Achmad Rifai[2]

Abstract*—The information retrieval (IR) system or information retrieval system is used to rediscover the information relevant to the user's needs from an information set automatically. In the information retrieval there are many methods used, one of them is the vector space model method that can measure the similarity between the vectors with the keywords in the input by the user. By adding the frequency-inverse document frequency (TF-IDF) method the data will be calculated by its weight, if there is a term with almost the same weight, this data will be more in priority and will appear at the top of the search. Recall is declared as part of the relevant document in the found document, whereas Precision is declared as part of the relevant document found. In this study can be produced in performance, retrieval system developed is good enough because with average precision about 71.31973% which means the average on each recall point, 71.31973% documents successfully found-return Relevant to the given query

Intisari—Information retrieval (IR) system atau sistem temu kembali informasi digunakan untuk menemukan kembali informasi-informasi yang relevan terhadap kebutuhan pengguna dari suatu kumpulan informasi secara otomatis. Dalam information retrieval terdapat banyak sekali metode yang digunakan, salah satunya adalah metode vector space model yang dapat mengukur kemiripan antar vektor dengan kata kunci yang di inputkan oleh pengguna. Dengan menambahkan metode frequency-inverse document frequency (TF-IDF) data akan dihitung bobot nya, apabila ada istilah dengan bobot yang hampir sama, data ini akan lebih di utamakan dan akan muncul dibagian paling atas dalam pencarian. Recall dinyatakan sebagai bagian dari dokumen relevan dalam dokumen yang ditemukan, sedangkan *Precision* dinyatakan sebagai bagian dokumen relevan yang ditemukan. Dalam penelitian ini dapat di hasilkan secara kinerja, sistem temu-kembali yang dikembangkan sudah cukup baik karena dengan rata-rata average precision sekitar 71,31973% yang berarti rata-rata pada tiap recall point, 71,31973% dokumen yang berhasil ditemu-kembalikan relevan dengan query yang diberikan

Kata Kunci— **Sistem Informasi, Information Retrieval, Vector Space Model, Pencarian Teks.**

[1] *Program Studi Sistem Informasi, STMIK Nusa Mandiri, Jl Kramat Raya No.18 Jakarta Pusat; e-mail: jamal.jml @nusamandiri.ac.id)*

[2] *Program Studi Teknik Informatika, STMIK Nusa Mandiri, Jl. Damai No. 8 Warung Jati Barat, Jakarta Selatan; e-mail: penulis2:achmad.acf@gmail.com)*

I. PENDAHULUAN

Penggunaan sebuah komputer untuk menyimpan dokumen teks dalam bentuk file sampai saat ini sudah banyak dilakukan, penyimpanan teks atau informasi dalam sebuah komputer informasi tersebut dilakukan agar informasi tersebut dapat dengan mudah diakses atau digunakan [1]. Seiring banyak nya penyimpanan dalam datal menyebabkan jumlah informasi yang tersedia berlimpah [2], para Pengguna informasi cenderung menggunkan kosa kata untuk mencari sebuah informasi dalam sebuah sistem [3]. Sayangnya, tingkat keefektifanya tidak terlalu akurat, karena beberapa dokumen yang dituju mungkin tidak dapat diambil karena ketidakcocokan nya dalam penggunaan kosa kata [4].

Pencarian informasi berdasarkan query yang digunakan oleh pengguna, yang diharapkan dapat menemukan informasi yang relevan berdasarkan query yang digunakan dikenal dengan temu balik informasi (information retrieval) [5]. Dengan proses information retrieval, pada saat-saat tertentu informasi dapat ditemukan lagi dengan baik dan relevan [6]. Salah satu metode yang digunakan dalam information retrieval adalah Vector Space Model (VSM), Metode VSM digunakan untuk memudahkan pengguna dalam mencari dokumen [7]. VSM merupakan salah satu model yang digunakan untuk mengetahui kemiripan dokumen yang digunakan dalam FAQ otomatis [8]. Dalam metode VSM menampilkan hasil yang berbeda untuk setiap dokumen yang tersimpan dalam database, yang secara bergantian menampilkan dokumen yang paling mirip dengan kueri yang di inputkan oleh pengguna, dengan tingkat kemiripan palingtinggi otomatis akan ditempatkan di bagian paling atas pencarian [7].

Tahap awal dimulai dengan menggunakan pengolahan teks, pembobotan setiap token dengan menggunakan istilah term frequency-inverse document frequency (TF-IDF) [2]. Dengan menggunakan metode TF-IDF ini akan memberikan bobot untuk istilah yang lebih diutamakan [1], jadi dengan metode frequency-inverse document frequency (TF-IDF) dapat memberikan bobot pada setiap indeks dari kosa kata(Term) dan metode Vector Space Model (VSM) dapat mengukur kemiripan antar vektor dokumen dengan kata kunci. Pada pengujian terdapat tiga besaran performansi yang dihitung yaitu Recall untuk menemukan seluruh dokumen yang relevan dalam koleksi dokumen, Precision untuk menemukan hanya dokumne yang relevan saja dalam koleksi dan Interpolated Recall Precision untuk

mengukur performansi sistem dengan memepertimbngkan aspek keturutan dokumen relevan [9].

II. METODE PENELITIAN

A. *Metode Vector Space Model*

Vector space model adalah suatu model yang digunakan untuk mengukur kemiripan antara suatu dokumen dengan suatu query [10]. Pada model ini, query dan dokumen dianggap sebagai vektor-vektor pada ruang n-dimensi, di mana n adalah jumlah dari seluruh term yang ada dalam leksikon. Leksikon adalah daftar semua term yang ada dalam indeks.

Metode *vector space model* menggunakan rumus untuk mencari nilai cosinus sudut antara dua vektor dari setiap bobot dokumen dan bobot dari kata kunci. Jika terdapat dua vektor dokumen dj dan *query* q, serta *term* yang diekstrak dari koleksi dokumen maka nilai cosines antara d_j dan q dapat didefinisikan sebagai berikut:

$$similarity\ (d_j,q) = \frac{d_j \bullet q}{|d_j| \bullet |q|} = \frac{\sum_{i=1}^{t} w_{ij} \bullet w_{iq}}{\sqrt{\sum_{i=1}^{t} w_{ij}^2 \bullet \sum_{i=1}^{t} w_{iq}^2}} \quad (1)$$

B. Metode TF-IDF

Term Frequency-Inverse Document Frequency adalah salah satu perhitungan bobot dari frekuensi kemunculan sebuah term pada dokumen. TF-IDF digunakan untuk mencari nilai bobot dari dokumen [11].

TF-IDF akan memeriksa kemunculan tiap kata pada isi dokumen dari hasil tokenisasi, filtering dari kemuculan tiap kata pada isi dokumen. Adapun rumus yang digunakan untuk perhitungan TF- IDF adalah sebagai berikut:

$$\text{IDF} = \log \frac{D}{df} \quad (2)$$

$$W_{(d,t)} = tf_{(d,t)} \text{ x } IDF_{(t)} \quad (3)$$

Dimana:

IDF = *inversed document frequency*
D = jumlah dokumen
df = banyak dokumen yang mengandung kata yang dicari
d = dokumen ke-d
t = kata ke-t dari kata kunci
W = bobot dokumen ke-d terhadap kata ke-t
tf = banyaknya kata yang dicari pada sebuah dokumen

C. Sistem Temu Kembali Informasi (*Information Retrieval)*

Information retrieval (IR) system atau sistem temu kembali informasi digunakan untuk menemukan kembali informasi-informasi yang relevan terhadap kebutuhan pengguna dari suatu kumpulan informasi secara otomatis [12]. Salah satu aplikasi umum dari IR system adalah search engine atau mesin pencari yang terdapat pada jaringan internet. Pengguna dapat mencari halaman-halaman web yang dibutuhkannya melalui search engine.

Tujuan dari sistem temu kembali informasi yang ideal adalah:

1. Menemukan seluruh dokumen yang relevan terhadap suatu query.
2. Hanya menemukan dokumen relevan saja, artinya tidak terdapat dokumen yang tidak relevan pada dokumen hasil pencarian.

Dua keadaan tersebut digunakan untuk menghitung performansi sistem temu kembali, yaitu recall dan precision. Recall dinyatakan sebagai bagian dari dokumen relevan dalam dokumen yang ditemukan. Recall dapat dihitung dengan persamaan:

$$\text{Recall (r)} = \frac{\text{Jumlah Dokumen Relevan ditemukan}}{\text{Jumlah Dokumen Relevan Dalam Koleksi}} \quad (4)$$

Nilai recall tertinggi adalah 1, yang berarti seluruh dokumen dalam koleksi berhasil ditemukan [13].

Precision dinyatakan sebagai bagian dokumen relevan yang ditemukan. *Precision* dapat dihitung dengan persamaan:

$$\textit{Precision}\ (P) = \frac{\text{Jumlah Dokumen Relevan ditemukan}}{\text{Jumlah Dokumen Ditemukan}} \quad (5)$$

Nilai *precision* tertinggi adalah 1, yang berarti seluruh dokumen yang ditemukan adalah relevan [13].

Pengukuran *recall* dan *precision* ini merupakan perhitungan yang dilakukan terhadap kumpulan dokumen hasil pencarian (*set based measure*) secara keseluruhan. Pengukuran dengan menggunakan *set based measure* ini tidak dapat menggambarkan performansi sistem temu kembali informasi mengenai urutan dari dokumen relevan. Pengukuran performansi dengan mempertimbangkan aspek keterurutan atau rangking dapat dilakukan dengan melakukan interpolasi antara *precision* dan *recall.* Nilai rata-rata *interpolated precision* dapat mencerminkan urutan dari dokumen yang relevan pada perangkingan.

Standar yang biasa digunakan adalah 11 standar tingkat *recall*, yaitu $r_j \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$. Misalkan r_j, $j \in \{0,1,2,.10\}$ adalah tingkat standar *recall* ke-j maka:

$$P(r_j) = \max r_j \le r \le r_{j+1} P(r) \quad (6)$$

Aturan interpolasi adalah recall standard ke-j memiliki nilai interpolated precision sebesar maksimum precision pada recall yang lebih besar dari recall standard ke-j.

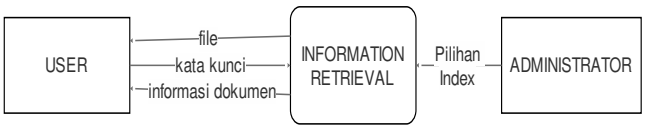
III. HASIL DAN PEMBAHASAN

A. Perancangan Data Flow Diagram

Sistem temu-kembali informasi yang akan dibangun menggunakan model ruang vektor (Vector Space Model). Proses utama yang digunakan oleh sistem temu-kembali

informasi adalah indexing yang lebih lanjut akan dijelaskan pada tahap perancangan.

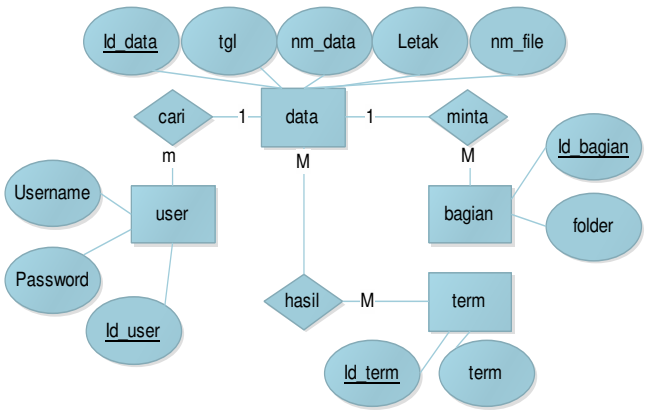
Sistem ini nantinya akan dibagi menjadi dua bagian besar, yaitu proses indexing yang berhubungan dengan dokumen-dokumen dan proses query yang berhubungan dengan pengguna. Pengaturan indexing dokumen berbasis teks menjadi kumpulan indeks istilah akan dilakukan oleh administrator. Sedangkan proses query pengguna akan direpresentasikan melalui pengiriman kata kunci berupa teks untuk diproses menjadi query yang dapat digunakan oleh sistem ini untuk mencari informasi di dalam dokumen yang disimpan. Gambaran mengenai sistem temu-kembali informasi pada sistem penyimpanan data dapat dilihat pada Gambar 1.



Gbr. 1 Diagram Konteks

Dari diagaram konteks maka dapat diturunkan menjadi Data Flow Diagram (DFD) level 1. DFD adalah sebuah teknik grafis yang menggambarkan aliran data yang bergerak dari input ke output. Selain itu DFD juga menyajikan fungsi-fungsi sistem yang mengolah data input dan menghasilkan data output. Diagram alur data dapat digunakan untuk menyajikan suatu sistem perangkat lunak pada setiap tingkat abstraksi.

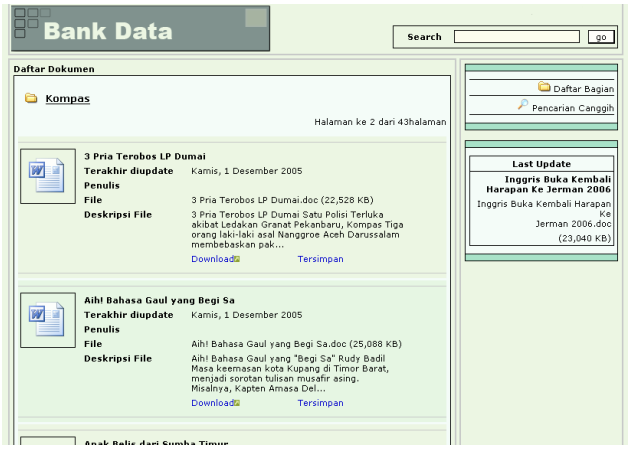
B. Perancangan Basis Data



Gbr. 2 *Entity Relation Diagram*

C. Perancangan Tampilan

Pada halaman ini dapat melakukan pencarian dokumen yang memiliki suatu informasi tertentu dengan memasukan kata kunci ke dalam form. Form ini terletak pada bagian kanan atas pada setiap halaman yang dilihat oleh pengguna. Selain itu pengguna juga dapat mencari informasi dari dokumen dengan fasilitas pencarian canggih. Pada fasilitas pencarian canggih, pengguna dapat mencari dokumen secara spesifik dengan memasukan kata kunci yang berupa kata dan kata tepat, penulis, serta bagian tertentu dari dokumen yang akandicari.



Gbr. 3 Halaman Dokumen

Pada halaman ini dapat melakukan pencarian dokumen yang memiliki suatu informasi tertentu dengan memasukan kata kunci ke dalam form. Form ini terletak pada bagian kanan atas pada setiap halaman yang dilihat oleh pengguna. Selain itu pengguna juga dapat mencari informasi dari dokumen dengan fasilitas pencarian canggih. Pada fasilitas pencarian canggih, pengguna dapat mencari dokumen secara spesifik dengan memasukan kata kunci yang berupa kata dan kata tepat, penulis, serta bagian tertentu dari dokumen yang akandicari.

Gbr. 4 Halaman Pencarian Canggih

D. Modul Pengindeksan

File dari modul pengindeksan diberi nama dengan nama tokenize.php dan digunakan hanya pada saat sistem temu-kembali melakukan pengindeksan dokumen. Di dalam modul ini akan dilakukan parsing dan penghilangan stopwords dari dokumen-dokumen yang akan diindeks dilanjutkan dengan penghitungan nilai variabel tf dan idf. Hasil dari pengindeksan akan disimpan ke dalam basis data yang akan digunakan pada saat pencarian dokumen.

Modul yang digunakan pada saat pengindeksan ada dua buah, yaitu modul untuk parsing dan penghilangan stopwords serta modul perhitungan variabel idf. Dokumen yang diindeks diambil dari media penyimpanan sesuai dengan pilihan yang diberikan oleh administrator. Namun

sebelum itu, untuk mengetahui letak dokumen di media penyimpanan maka terlebih dahulu akan dilakukan pengambilan informasi dokumen dari tabel t_data pada basis data.

Awal dari modul pengindeksan adalah pendefinisian lamanya waktu indeks yang akan digunakan. Hal ini dimaksud agar waktu batas maksimum eksekusi file php yang sudah di set pada server web disesuaikan dengan lamanya waktu eksekusi pengindeksan dokumen. Perlu diingat pembuatan indeks sekumpulan dokumen dapat memakan waktu berjam-jam bahkan mungkin berhari-hari tergantung dari jumlah kata yang terdapat pada seluruh dokumen. Pada penelitian ini, untuk menghindari agar server web tidak menghetikan proses pengindeksan sebelum semua dokumen selesai diindeks maka batas maksimum waktu pengindeksan akan diatur menjadi 86400 detik yaitu selama satu hari penuh.

Setelah mengatur batas waktu maksimum maka dilakukan pengambilan daftar kata stopwords yang disimpan didalam sebuah file bernama stopwords.txt di dalam media penyimpanan. Isi dari daftar kata-kata stopwords yang disimpan di dalam file txt tersebut dapat dilihat pada Tabel I (Lampiran).

Kata-kata yang terdapat pada Tabel I tersebut tidak akan digunakan sebagai kata di dalam indeks serta pada query pencarian. Penggunaan sebuah file stopwords.txt dikarenakan kata-kata tersebut tidak diperbarui secara rutin, walaupun ada penambahan kata stopwords dapat dilakukan secara mudah oleh administrator. Selanjutnya, kata-kata yang terdapat di dalam file stopwords.txt akan dimasukan kedalam sebuah variabel bertipe array untuk dapat digunakan oleh sistem secara mudah.

Setelah pendefinisian lama waktu eksekusi dan pengambilan daftar kata stopwords, maka proses pengindeksan akan dilanjutkan ke proses pemilihan dokumen yang akan diindeks. Selanjutnya akan dilakukan pengambilan informasi dokumen yang akan diindeks dari basis data. Informasi yang diambil hanya letak alamat dari file sementara, bukan letak dari file aslinya.

E. Evaluasi Sistem Temu-Kembali Informasi

Pada Bagian ini akan dijelaskan mengenai pengujian sistem temu-kembali informasi. Dokumen sebagai bahan dari pengujian ini didapatkan dari artikel berita utama kompas edisi cetak (http://kompas.com/kompas-cetak/) dan republika edisi cetak (http://www.republika.co.id) selama bulan Oktober tahun 2016, kecuali untuk tanggal 29 Oktober 2016 tidak terdapat dokumen pada website republika.

Dokumen akan dipisah menjadi dua bagian sesuai dengan nama sumber dari dokumen tersebut yaitu bagian kompas dan republika. Bagian kompas terdiri dari 212 dokumen dan republika terdiri dari 180 dokumen, jadi total seluruh dokumen sebanyak 392 dokumen. Deskripsi isi dari dokumen yang akan dijadikan bahan uji di dalam sistem dapat dilihat pada Tabel III.

Proses pengindeksan memperoleh jumlah kata unik untuk dijadikan indeks sebanyak 16.828 kata, dan terdapat sebanyak 79.345 hubungan antara kata yang diindeks dengan dokumen. Proses pengindeksan memakan waktu selama 2 Jam 35 menit 29 detik. Ukuran dari basis data untuk pengindeksan sebesar 4.115,2 KB.

TABEL II
EVALUASI RECALL DAN PRECISION

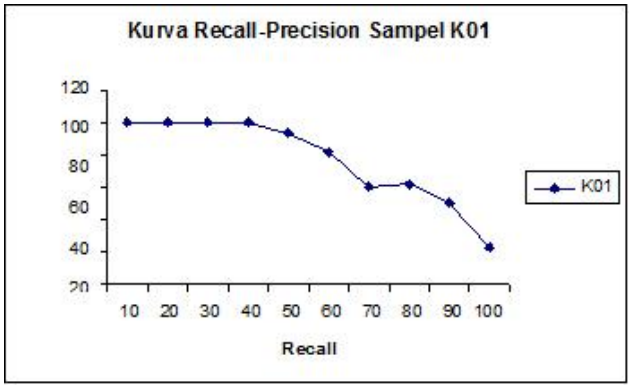
Pengujian	Kata Kunci	Dokumen Relevan		Total
		Kompas	Republika	
K01	Bantuan Langsung Tunai (BLT)	20	11	31
K02	“Kenaikan Harga BBM”	24	16	40
K03	Kasus Suap “Makamah Agung	18	6	24
K04	Bom Bunuh Diri Bali	22	12	34
K05	Gempa Pakistan	3	6	9
K06	Arus Mudik Lebaran	6	10	16
	Lain-lain	119	119	238
	Jumlah seluruh Dokumen	212	180	392

Evaluasi sistem akan dilakukan dengan metode recall dan precision. Pengujian dilakukan dengan memasukan kata kunci sesuai dengan deskripsi. Sampel pengujian diambil sebanyak 6 buah sesuai tertera pada Tabel III yaitu K01 sampai dengan K06.

TABEL III
PENGUJIAN SAMPEL

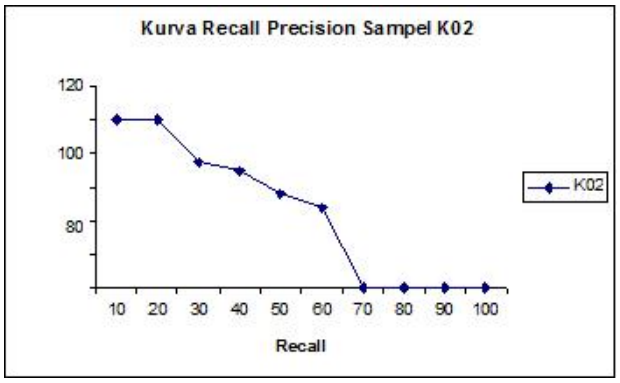
Recall	*P*					
	K01	**K02**	**K03**	**K04**	**K05**	**K06**
10	100	100	66,666	100	50	100
20	100	100	83,33	100	66,666	100
30	100	75	58,333	100	75	83,33
40	100	69,565	62,5	100	80	85,714
50	93,75	55,55	66,666	100	83,333	72,727
60	81,818	48	70	100	83,333	76,923
70	60	0	42,5	96	85,71	78,571
80	61,538	0	34,545	93,103	87,5	76,471
90	50,94	0	0	91,176	88,888	73,684
100	22,627	0	0	85	90	72,727
Average Precisio	77,0673	44,8115	48,454	96,5279	79,043	82,0147

Pengujian dengan sampel K01 didapatkan sebanyak 161 dokumen dengan rata-rata lama proses pencarian selama 3,8 detik. Pada Tabel III.3 dapat dilihat bahwa sampel K01 pada titik *recall* 10% memiliki nilai *precision* sebesar 100%, selanjutnya pada titik *recall* 20% memiliki nilai *precision* sebesar 100% dan begitu seterusnya sehingga dari data tersebut dapat dibentuk kurva *recall- precision* untuk sampel K01 yang dapat dilihat pada Gambar 5.



Gbr. 5 Kurva Recall-Precision Sampel K01

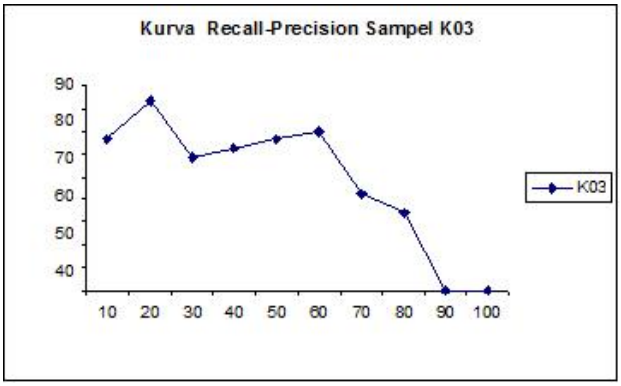
Pengujian dengan sampel kata kunci K02 didapatkan 70 dokumen dengan rata-rata lama proses pencarian 2,2 detik. Kurva *recall-precision* untuk sampel K02 dapat dilihat pada Gambar 6.



Gbr. 6 Kurva Recall-Precision Sampel K02

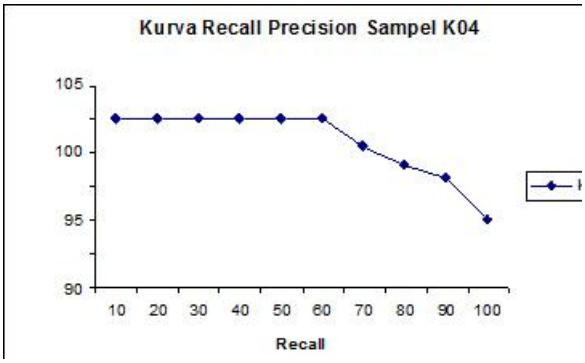
Pada sampel K02 dapat dilihat pada nilai *recall* 70%, 80%, 90% dan 100% tidak didapatkan nilai persentase dari *precision*. ini berarti bahwa tidak seluruh dokumen yang memiliki informasi mengenai "'kenaikan harga bbm'" dapat ditemu-kembalikan oleh sistem pencarian

Pengujian dengan sampel kata kunci K03 didapatkan 119 dokumen dengan rata-rata lama proses pencarian 2,8 detik. Pada sampel K03 ini didapatkan nilai *precision* 0% pada *recall* 90% dan 100%. Ini juga berarti tidak seluruh dokumen yang memiliki informasi mengenai "kasus suap 'makamah agung'" berhasil ditemu-kembalikan oleh sistem. Kurva *recall-precision* untuk sampel K03 dapat dilihat pada Gambar 7.



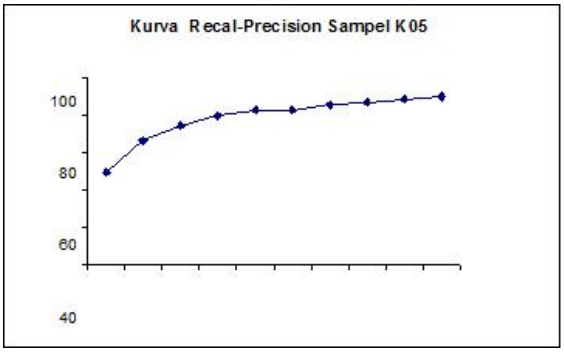
Gbr. 7 Kurva Recall-Precision Sampel K03

Pengujian dengan sampel kata kunci K04 didapatkan 141 dokumen dengan rata-rata lama proses pencarian 3,5 detik dengan rata-rata nilai *precision* sebesar 96,5279% pada setiap titik *recall*. Hal ini terjadi karena kata kunci yang dimasukan memiliki keunikan dibandingkan kata kunci lainnya sehingga sistem dapat melakukan proses temu-kembali hampir mendekati maksimal Grafik *recall-precision* untuk sampel K04 dapat dilihat pada Gambar 8.



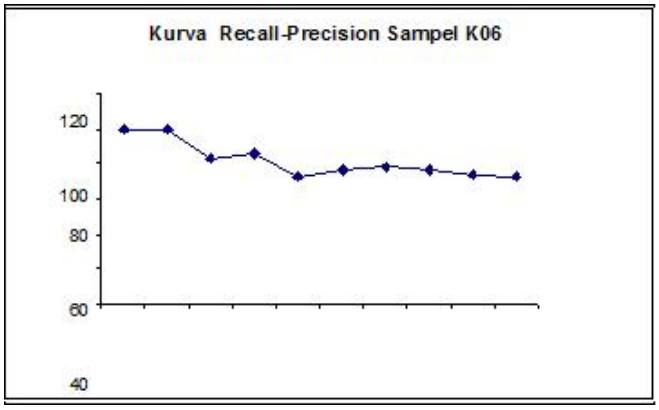
Gbr. 8 Kurva Recall-Precision Sampel K04

Pengujian dengan sampel kata kunci K05 didapatkan 13 dokumen dengan rata-rata lama proses pencarian 0,2 detik dengan rata-rata nilai precision sebesar 79,043% pada setiap titik recall. Kurva recall-precision untuk sampel K05 dapat dilihat pada Gambar 9.



Gbr. 9 Kurva Recall-Precision Sampel K05

Pengujian dengan sampel kata kunci K06 didapatkan 52 dokumen dengan rata-rata lama proses pencarian 1,04 detik dengan rata-rata nilai *precision* sebesar 82,0147% pada setiap titik *recall*. Kurva *recall-precision* untuk sampel K06 dapat dilihat pada Gambar 10.



Gbr. 10 Kurva Recall-Precision Sampel K06

Dari hasil pengujian sampel dokumen sesuai dengan kata kunci K01 sampai K06 dapat dilihat perbedaan waktu pencarian yang ditempuh. Perbedaan waktu pencarian ini didapatkan dari dua faktor, banyaknya dokumen yang berhasil ditemu-kembalikan dan banyaknya kata kunci yang dimasukan. Sedangkan untuk keakuratan hasil pencarian untuk mendapatkan dokumen yang relevan tergantung pada keunikan kata kunci yang diberikan oleh pengguna.

Sedangkan untuk pencarian dengan menggunakan frasa tepat lebih mengurangi jumlah recall dari hasil pencarian. Akan tetapi, penggunaan frasa tepat ini tidak menaikan nilai precision sehingga tidak seluruh dokumen yang relevan dapat ditemu-kembalikan ke pengguna.

Kinerja dari sistem temu-kembali pada sampel dokumen K01 sampai K06 dapat dilihat pada Tabel III. Secara kinerja sistem temu-kembali yang dikembangkan sudah cukup baik karena dengan rata-rata average precision sekitar 71,31973% yang berarti rata-rata tiap recall point, 71,31973% dokumen yang berhasil ditemu-kembalikan relevan dengan query yang diberikan.

IV. KESIMPULAN

Berdasarkan hasil penelitian yang berupa pengembangan sistem pencarian dengan metode temu-kembali informasi dapat diambil sebuah kesimpulan yaitu:

1. Sistem temu-kembali informasi yang dibuat dapat mencari informasi dari isi file dokumen yang disimpan di dalam sistem.
2. Proses pengindeksan dokumen didalam sistem temu-kembali informasi yang dikembangkan melalui beberapa tahapan pemrosesan teks, yaitu parsing, penghilangan stopwords dan penghitungan nilai bobot setiap kata yang akan dijadikan indeks. Sedangkan untuk proses pencariannya juga melalui beberapa tahapan proses yang yang hampir sama dengan proses pengindeksan, yaitu parsing, penghilangan stopwords, cek frasa dan yang terakhir adalah penghitungan fungsi kesamaan untuk mendapatkan nilai bobot setiap dokumen yang akan dicari.
3. Kecepatan pencarian sebuah informasi tergantung dari jumlah dokumen yang dihasilkan dan jumlah kata kunci yang digunakan sebagai query pencarian.
4. Pencarian dengan menggunakan frasa tepat dapat mengurangi nilai recall dari hasil pencarian. Akan tetapi hal ini akan menyebabkan nilai precision menurun, karena tidak semua dokumen yang memiliki informasi yang relevan dapat ditemu-kembalikan. Dan juga penggunaan frasa tepat dapat memperlama proses pencarian karena kata kunci harus diproses untuk mencocokan dokumen.
5. Secara kinerja, sistem temu-kembali yang dikembangkan sudah cukup baik karena dengan rata-rata average precision sekitar 71,31973% yang berarti rata-rata pada tiap recall point, 71,31973% dokumen yang berhasil ditemu-kembalikan relevan dengan query yang diberikan

REFERENSI

[1] Fitri, M. (2013). Perancangan Sistem Temu Balik Informasi dengan Metode Pembobotan Kombinasi TF-IDF untuk Pencarian Dokumen Berbahasa Indonesia. Jurnal Sistem dan Teknologi Informasi (JustIN), 1(1).

[2] Zaman, B., Purwanti, E., & Sukma, A. (2016). Information Retrieval Document Classified with K-Nearest Neighbor. Record and Library Journal, 1(2), 129-138.

[3] Steven. (2026). Perancangan Information Retrieval System Untuk Dokumen Berbahasa Indonesia Dengan Menggunakan Extended Boolean. Jurnal Ilmu Komputer dan Sistem Informasi, 1 (1), 183-189.

[4] Karnalim, O. (2015). Extended vector space model with semantic relatedness on java archive search engine. Jurnal Teknik Informatika dan Sistem Informasi, 1(2).

[5] Karyono, G., & Utomo, F. S. (2012). Temu Balik Informasi Pada Dokumen Teks Berbahasa Indonesia Dengan Metode Vector Space Retrieval Model. Semantik 2012, 282-289.

[6] Heriyanto, H. (2015, December). Pencarian Dokumen Teks Arsip Surat Dengan Metode Indexing Dan Query. In Seminar Nasional Informatika (SEMNASIF) (Vol. 1, No. 1).

[7] Amin, F. (2013). Sistem Temu Kembali Informasi dengan Pemeringkatan Metode Vector Space Model.

[8] Aziz, A., Saptono, R., & Suryajaya, K. P. (2016). Implementasi Vector Space Model dalam Pembangkitan Frequently Asked Questions Otomatis dan Solusi yang Relevan untuk Keluhan Pelanggan. Scientific Journal of Informatics, 2(2), 111-121.

[9] Kafatan, S., Riyanto, D. E., & Saputra, R. (2014). Sistem Informasi Pengelolaan Arsip Statis Pada Badan Arsip Dan Perpustakaan Provinsi Jawa Tengah Menggunakan Vector Space Model. Jurnal Masyarakat Informatika, 5(9), 45-52.

[10] Putri, W. M. I. (2016). Kombinasi Metode Vector Space Model Dan Teknik Hierarchical Agglomerative Clustering Single Linkagedalam Rancang Bangun Information Retrievalpada Perpustakaan Digital (Doctoral dissertation, Universitas Islam Negeri Sultan Syarif Kasim Riau).

[11] Hidayat, A. (2016). Implementasi Metode Terms Frequency–Inverse Document Frequency (Tf-Idf) Dan Maximum Marginal Relevance Untuk Monitoring Diskusi Online (Doctoral dissertation, Universitas Islam Negeri Sultan Syarif Kasim Riau).
[12] Wahyudi, I. P., Wurijanto, T., & Sulistiowati, S. (2017). Design Aplications To Increase Relevance To Search Thesis (A Case Study In Institute of Business and Information Stikom Surabaya Library). Jurnal JSIKA, 5(8).
[13] Novianti, K. D. P., & Diaz, R. A. N. (2017). Sistem Pencarian Program Studi Pada Perguruan Tinggi Di Bali Berbasis Semantik. JST (Jurnal Sains dan Teknologi), 6(1).

Jamal Maulana Hudin, M.Kom memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Sistem Informasi STMIK Nusa Mandiri Sukabumi, lulus Tahun 2012 dan memperoleh gelas Magister Komputer(M.Kom) Program Pasca Sarjana Magister Ilmu Komputer STMIK nusa Mandiri Jakarta, Lulus Tahun 2016.

Achmad Rifai memperoleh gelar Sarjana Komputer (S.Kom), Program Studi Sistem Informasi STMIK Nusa Mandiri Jakarta, lulus tahun 2011 dan memperoleh gelar Magister Komputer (M.Kom) Program Pasca Sarjana Magister Ilmu Komputer STMIK Nusa Mandiri Jakarta, lulus tahun 2015.

LAMPIRAN

TABEL I
Kumpulan Stopwords

Stopwords					
ada	biasanya	kalau	menunjukkan	Sedang	seusai
adalah	bila	kalian	menurut	Sedangkan	sewaktu
adanya	bilamana	kami	mereka	Sedikit	si
adapun	buat	kamu	merupakan	Segera	siapa
aduh	bukan	karena	Meski	Sehabis	siapakah
agar	dalam	kata	meskipun	Sehingga	siapapun
ah	dan	katanya	misalnya	sehubungan	suatu
akan	dapat	kau	mungkin	Sejak	sudah
aku	dari	ke	Namun	Sejumlah	supaya
alih-alih	daripada	kebanyakan	Nanti	Sekarang	tak
anda	dekat	kecuali	Nyaris	Sekeliling	tanpa
andai	demi	kemanakah	Oleh	Seketika	tapi
antar	demikian	kemudian	Pada	Sekitar	tatkala
antara	dengan	kenapa	padahal	Sekonyong-konyong	telah
apa	depan	kenapakah	Para	Selagi	tengah
apakah	di	kepada	Pasti	Selain	tentang
apalagi	dia	ketika	pelbagai	Selalu	tentu
asalkan	dikatakan	ketimbang	Per	Selama	tentunya
atas	dilakukan	kini	Peri	Selanjutnya	tergolong
atau	dkk	kita	perihal	Selesai	terhadap
ataupun	dll	lagi	pinggir	Seluruh	terjadi
Bagai	dsb	lain	Pula	Seluruhnya	terkadang
bagaikan	engkau	lain-lain	Pun	Semakin	terlalu
bagaimana	hal	lainnya	Saat	Semenjak	terlebih
bagaimanakah	hampir	lalu	Saja	Sementara	termasuk
bagaimanapun	hanya	lebih	Sambal	Semua	ternyata
bagi	harus	lepas	sampai	Semuanya	tersebut
bahkan	hingga	lewat	samping	Seorang	tertentu
bahwa	ia	maka	Sang	Sepanjang	tetap
balik	ialah	makin	Sangat	Seperti	tetapi
banyak	ini	manakala	sangatlah	Sepertinya	tiap
barangkali	itu	masih	Saya	Seputar	tiba-tiba
bawah	iya	masing-masing	Seakan	Seraya	tidak
beberapa	jadi	masing-	seakan-akan	Sering	ujar
begini	jangan	maupun	seantero	Seringkali	ujarnya
begitu	jarang	melainkan	Sebab	Serta	umumnya
belakang	jauh	melakukan	sebabnya	Sesuai	untuk
belum	jika	melalui	sebagai	Sesuatu	walau
berapa	jikalau	memang	sebagaimana	Sesudah	walaupun
berbagai	juga	mengatakan	sebagainya	Sesudahnya	ya
bersama	jumlah	mengenai	sebelum	sesungguhnya	yaitu
beserta	justru	menjadi	sebelumnya	Setelah	yakni
betapa	kadang	menjelang	Sebuah	Seterusnya	yang
biar	kadang-kadang	menuju	Secara	Setiap	