INFORMATION RETRIEVAL DOKUMEN TESIS UNTUK MENGETAHUI KEMIRIPANNYA DENGAN PENELITIAN YANG TELAH ADA

Monica Mayeni[1], Wing Wahyu Winarno[2,] Andi Sunyoto[3]
[1]Mahasiswa Pascasarjana MTI STMIK AMIKOM Yogyakarta
[2]Dosen MTI STMIK AMIKOM, Yogyakarta
[3]Dosen STMIK AMIKOM, Yogyakarta

Abstract

Information retrieval system is a system used in rediscovering ever previously stored information. This study discusses the design of information retrieval system by employing *Vector Space Model* in the tracking and also similarity measures of *Cosine Similarity* as the method to rank the documents found that match with the keywords/query. The purposes of this study are to facilitate students in drawing prediction whether the research will be proposed is accepted or rejected and also to design a system to show documents which are similar to the query entered as the search limitation. This study did not calculate the plagiarism degree of the selected documents. The focus of this study was on designing the tracking system; however, the prototype application was tested to show the result of the system implementation. Testing was conducted to 60 .pdf formatted documents. The application was executed in the desktop with a local host display and the database was stored in the computer's hard drives. The final result of this system was a sequence of selected documents which were relevant or similar to the user's query that later can also be used as a research reference. Due to the limitations of rules on *Porter stemmer algorithm,* a list of words that are not perfectly drawn during the stemming process needs to be added for the development of the system.

Keywords: *Information Retrieval System, Vector Space Model, Porter Stemmer Algorithm, Cosine Similarity*

Abstrak

Information retrieval system adalah sistem yang digunakan dalam menemukan kembali informasi yang pernah tersimpan sebelumnya. Penelitian ini membahas perancangan sistem temu kembali informasi dengan menggunakan *Vector Space Model* dalam penelusurannya serta menggunakan ukuran kesamaan *Cosine Similarity* sebagai metode dalam meranking dokumen yang ditemukan yang sesuai dengan kata kunci/*query*. Tujuan dari penelitian ini adalah untuk membantu memudahkan mahasiswa dalam memprediksi apakah penelitian yang akan diajukan tersebut diterima atau ditolak, dan merancang sistem untuk memperlihatkan dokumen-dokumen yang mirip dengan *query* yang dimasukkan sebagai batasan pencarian. Penelitian ini tidak menghitung tingkat plagiasi dari dokumen terpilih. Fokus penelitian adalah pada perancangan sistem penelusuran namun aplikasi *prototype* diuji untuk memperlihatkan hasil implementasi sistem. Uji coba dilakukan dengan menggunakan 60 dokumen berformat .pdf, aplikasi dieksekusi secara *desktop* dengan tampilan *localhost* dan *database* disimpan dalam *hardisk* komputer. Hasil akhir sistem adalah urutan dokumen-dokumen terpilih yang relevan atau mirip dengan *query user* yang nantinya dapat juga digunakan sebagai referensi penelitian. Karena keterbatasan aturan pada *Porter stemmer algorithm* maka

sebaiknya untuk pengembangan sistem perlu ditambahkan daftar kata yang tidak terambil secara sempurna pada saat dilakukan proses *stemming*.

Kata kunci: *Information Retrieval System, Vector Space Model, Porter Stemmer Algorithm, Cosine Similari*

1. Pendahuluan

Syarat akhir bagi mahasiswa untuk dapat lulus dari suatu perguruan tinggi adalah membuat suatu penelitian (skripsi, tesis ataupun tugas akhir). Sebagai mahasiswa, biasanya akan mengalami kendala dalam mencari judul maupun memperkirakan apakah judul ataupun penelitian yang akan diajukan tersebut akan diterima ataupun ditolak oleh program studi; terkait dengan sama, mirip atau tidaknya penelitian tersebut dengan penelitian yang pernah ada sebelumnya.

Sementara itu pemerintah telah mengatur batasan suatu penelitian dikategorikan masuk dalam tindak plagiarism dalam PERMEN no.17 tahun 2010, tentang pencegahan dan penanggulangan plagiat di perguruan tinggi. Intinya, jika dalam penelitian mengandung isi yang mengutip penelitian lain namun tidak mencantumkan rujukan kutipan, maka dapat dikatakan penelitian tersebut masuk kategori plagiat.

Penelitian ini akan membuat rancangan sistem untuk menelusuri dokumen dalam suatu repository berdasarkan query yang dimasukkan user sehingga akan ditemukan dokumen-dokumen yang mirip dengan query tersebut, serta ranking/urutan dokumen yang ditemukan berdasarkan nilai kemiripannya.

2. Information Retrieval System (IRS)

Menurut Salton (1989), sistem temu kembali informasi merupakan suatu sistem yang menemukan *(retrieve)* informasi yang sesuai dengan kebutuhan *user* dari kumpulan informasi secara otomatis. Prinsip kerja sistem temu kembali informasi jika 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 (Amin, 2012).

Fungsi utama IRS seperti dikemukakan oleh Lancaster (1979) dan Kent (1971) adalah sebagai berikut:

1) Mengidentifikasi sumber informasi yang relevan dengan minat masyarakat pengguna yang ditargetkan.
2) Menganalisis isi sumber informasi (dokumen)
3) Merepresentasikan isi sumber informasi dengan cara tertentu yang memungkinkan untuk dipertemukan dengan pertanyaan (*query*) pengguna.
4) Merepresentasikan pertanyaan (*query*) pengguna dengan cara tertentu yang memungkinkan untuk dipertemukan sumber informasi yang terdapat dalam basis data.
5) Mempertemukan pernyataan pencarian dengan data yang tersimpan dalam basis data.
6) Menemu-kembalikan informasi yang relevan.
7) Menyempurnakan unjuk kerja sistem berdasarkan umpan balik yang diberikan oleh pengguna.

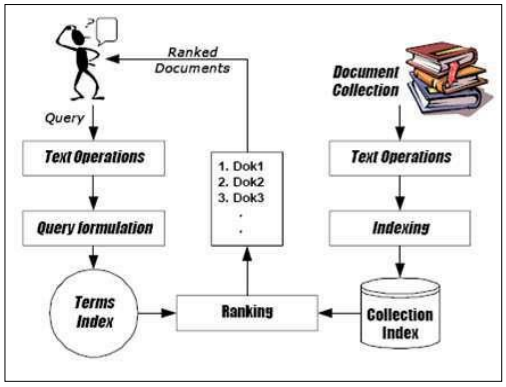
Penelitian mengenai IRS sebelumnya sudah banyak dilakukan, mulai dari temu kembali informasi untuk mencari informasi berbahasa

Indonesia (Karyono et al., 2012) maupun sistem deteksi plagiarsme dokumen bahasa Indonesia (Tudesman et al., 2014). Bahkan, Gadge et al. (2015) telah melakukan penelitian untuk mempercepat unjuk kerja model penelusuran dengan melakukan segmentasi terhadap dokumen yang akan ditelusuri. Untuk kategori penelusuran yang lebih spesifik, yaitu penelusuran untuk bidang pertanian, Luan et al. (2013) telah melakukan penelitian dengan menggunakan fitur multy_query agar identifikasi dokumen lebih baik.

Dalam proses pencarian informasi, diperlukan interaksi antara user dan sistem secara langsung maupun tidak. Seperti digambarkan pada Gambar 1, interaksi dan bagian-bagian dalam sistem ada yang memerlukan ekseskusi dari user dan juga sistem..

Menurut Kurniasih (2015), Tujuan IRS adalah sebagai berikut:

1) Suatu sistem temu kembali informasi bertujuan mengumpulkan dan mengorganisasikan informasi dalam sebuah sistem agar dapat memberikan informasi kepada pengguna secepat permintaannya.
2) Sistem temu kembali (IRS) tidak menunjukkan perkembangan ilmu pengetahuan, tetapi menunjukkan
3) ada tidaknya sebuah dokumen, sebagaimana dikemukakan oleh Lancaster : "*an information retrieval system does not inform (i.e., change the knowledge of) the user on the subject of his enquiry. It merely inform him of the existence (or non-existence) and where aboutsof documents relating to his request*." (Lancaster, 1979).



Gambar 1. Bagian-bagian IRS

Berdasarkan Gambar 1, maka proses dalam IRS terdiri atas 3 bagian besar kegiatan, yaitu:

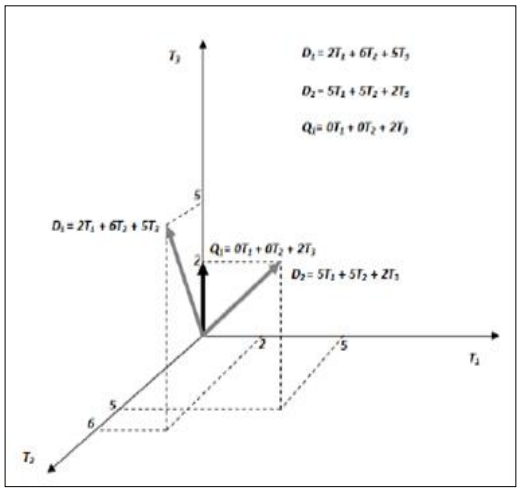
1. *Text Operation*, dimana kegiatan ini meliputi pemilihan kata-kata di dalam query maupun dokumen untuk ditransformasikan terhadap dokumen atau query tersebut dan menjadi terms index (indeks dari kata-kata).
2. *Query Formulation,* adalah formulasi terhadap query yang akan memberikan bobot pada indeks kata-kata query.
3. *Ranking*, yang merupakan pengurutan dokumen-dokumen yang ditelusuri dan relevan dengan query yang dimasukkan.
4. *Indexing*, merupakan proses membuat indeks dari koleksi dokumen. Proses ini adalah proses awal sebelum seluruh proses pada sistem temu kembali.

3. Vector Space Model (VSM)

Menurut Baeza (1999), *Vector Space Model (VSM)* adalah metode untuk melihat tingkat kedekatan atau kesamaan *(similarity)* term dengan cara pembobotan *term*. Dokumen dipandang sebagi sebuah vektor yang memiliki *magnitude* (jarak) dan *direction* (arah). Pada *Vector Space Model*, sebuah istilah direpresentasikan dengan sebuah dimensi dari ruang vektor. Relevansi

sebuah dokumen ke sebuah *query* didasarkan pada similaritas diantara vektor dokumen dan vektor *query* (Amin, 2012).

Dokumen dan query didalam VSM diekspresikan sebagai vektor dimensi di dalam koleksi dokumen, seperti dicontohkan pada Gambar 2.



Gambar 2. Model Ruang Vektor dengan dokumen D1 dan D2, serta query Q1

Pada VSM, database dari semua dokumen direpresentasikan oleh matriks *term-document* (atau matriks *term-frequency*), dimana setiap sel pada matriks kerkorespondensi dengan bobot yang diberikan dari suatu term. Nilai nol berarti bahwa term tidak terdapat dalam dokumen.

Term-frequency akan menghitung frekuensi term tersebut yang muncul dalam suatu dokumen, dengan persamaan (1).

$$tf = tf_{ij} \quad (1)$$

Untuk mengukur seberapa penting suatu term dalam dokumen secara keseluruhan, maka akan dihitung *idf (inverse document frequency)* dengan menggunakan persamaan (2).

$$idf_i = log \frac{N}{df_i} \quad (2)$$

Bobot w_{ij} yang merupakan bobot suatu term, dihitung menggunakan persamaan (3).

$$w_{ij} = tf_{ij} \times idf_i \quad (3)$$

Pada dokumen-dokumen biasanya panjangnya tidaklah seragam, sementara bobot term dihitung juga frekuensinya, untuk itu maka dilakukan normalisasi panjang query dan dokumen dengan persamaan (4) dan persamaan (5).

$$|q| = \sqrt{\sum_{j=1}^{t}(W_{i,q})^2} \quad (4)$$

Dimana |q| adalah jarak query dan w_{iq} bobot query dokumen ke-i.

$$|d_j| = \sqrt{\sum_{i=1}^{t}(W_{ij})^2} \quad (5)$$

Dengan $|d_j|$ adalah jarak dokumen.

Perhitungan pengukuran similaritas antara query dan dokumen dihitung dengan menggunakan persamaan (6).

$$sim(q, d_j) = \sum_{i=1}^{t} W_{iq} \cdot W_{ij} \quad (6)$$

4. Cosine Similarity

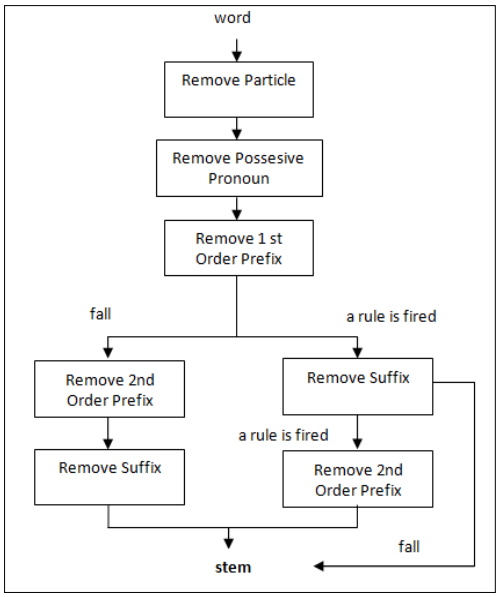
Salah satu ukuran kemiripan teks yang digunakan pada VSM untuk mengurutkan kemiripan dokumen adalah cosine similarity, yang menghitung nilai kosinus sudut antara vektor. Nilai cosine similarity didefinisikan dengan persamaan (7).

$$Sim(q, d_j) = \frac{q.d_j}{|q| * |d_j|} = \frac{\sum_{i=1}^{t} W_{iq} \cdot W_{ij}}{\sqrt{\sum_{j=1}^{t}(W_{iq})^2} * \sqrt{\sum_{i=1}^{t}(W_{ij})^2}} \quad (7)$$

5. Porter Stemmer Algorithm

Stemming adalah proses untuk memisahkan kata menjadi bentuk dasar dengan menghilangkan awalan, akhiran ataupun sisipan. Pada IRS proses stemming dilakukan dengan maksud memudahkan pencarian dan meningkatkan kualitas informasi yang didapatkan.

Algoritma Porter adalah algoritma yang dibuat oleh Martin Porter pada tahun 1980 untuk men-stemming teks dengan bahasa Inggris. Pada tahun 1992, W.B. Frakes melakukan beberapa modifikasi untuk stemming bahasa Indonesia. Frakes melakukan proses penghilangan awalan dan sisipan yang tidak ada didalam proses stemming teks berbahasa Inggris. Frakes juga membuat 5 (lima) tabel aturan untuk mendapatkan kata dasar bahaa Indonesia. Desain untuk teks berbahasa Indonesia tersebut dijelaskan dalam Gambar 3.



Gambar 3. Desain *Porter Stemmer* untuk bahasa Indonesia

6. Pembahasan

Sistem yang dirancang ini dapat digunakan oleh institusi manapun selama institusi telah menyimpan hasil penelitian mahasiswa dalam bentuk digital dan disimpan dalam satu repository.

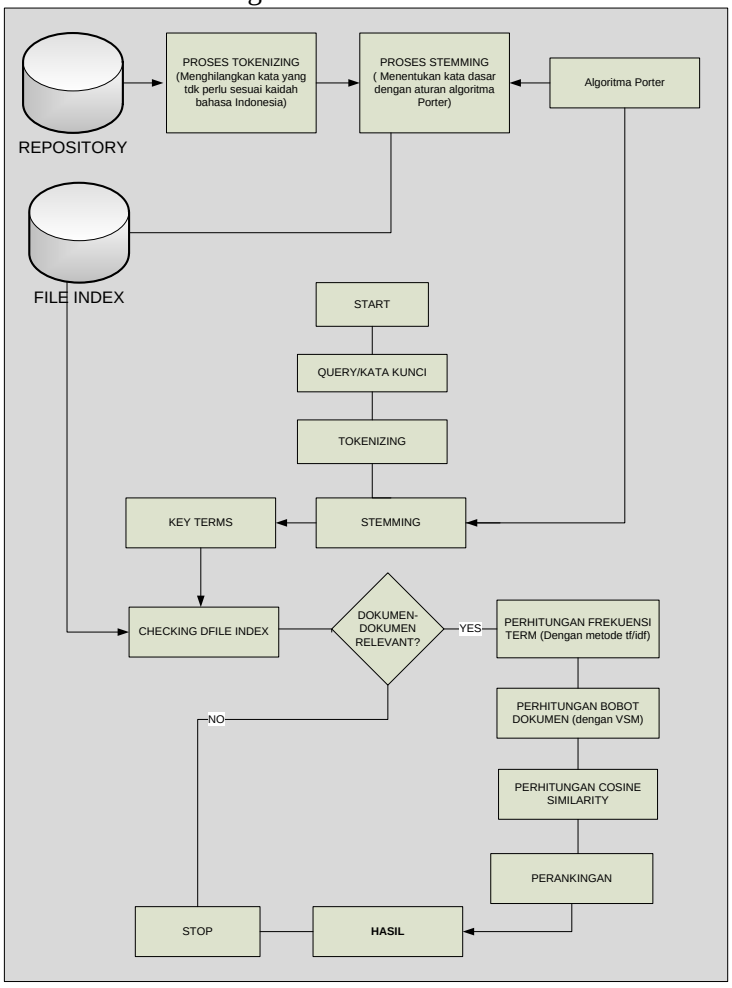
Sistem ini dirancang dengan mengambil database repository yang telah tersedia dan penggunaannya local untuk satu institusi itu saja.

Beberapa hal yang harus diperhatikan sehubungan dengan batasan penelitian ini adalah sebagai berikut:

a) Sistem yang dirancang adalah sistem temu kembali informasi dengan corpus dokumen .pdf dan teks berbahasa Indonesia.
b) Dokumen tersebut adalah dokumen dalam bentuk naskah publikasi, dimana format penulisan harus seragam, yaitu penulisan dibuat dalam dua kolom, dan struktur penulisan yang terurut.
c) Data utama adalah database repository, dimana koleksi dokumen naskah publikasi tersimpan.
d) Sistem akan membaca/mengambil koleksi dokumen tersebut tanpa memperhatikan bagaimana admin menyimpan dokumen ke dalam media penyimpanan.
e) Opsi pilihan pencarian yang dibuat adalah berdasarkan judul, batasan Masalah, abstrak dan kesuluruhan isi dokumen.
f) Prototype dibuat untuk dapat memperoleh gambaran hasil proses sistem yang dirancang.
g) Untuk mempercepat proses pembacaan maupun penelusuran, maka dokumen hasil index akan dipotong/kluster sesuai dengan opsi pilihan yang ditawarkan.
h) Pada aplikasi prototype ini, naskah publikasi yang digunakan sebagai bahan testing adalah dokumen dengan penulisan dua kolom, kecuali abstrak. Susunan penulisannya terdiri dari abstrak, latar belakang, batasan masalah, dan selanjutnya boleh struktur apa saja.

i) Hasil akhir dari sistem adalah dokumen teranking yang dianggap relevan dengan query, serta estimasi dalam bentuk persentase berdasarkan nilai kosinus yang dihasilkan sistem.

Gambaran sistem yang dirancang menggunakan VSM ini secara garis besar tahapan prosesnya terlihat pada alusr sistem pada Gambar 4. Repository adalah database yang dokumennya akan diambil dan ditelusuri dan file index adalah simpanan dokumen setelah dilakukan kluster sesuai dengan opsi pilihan.

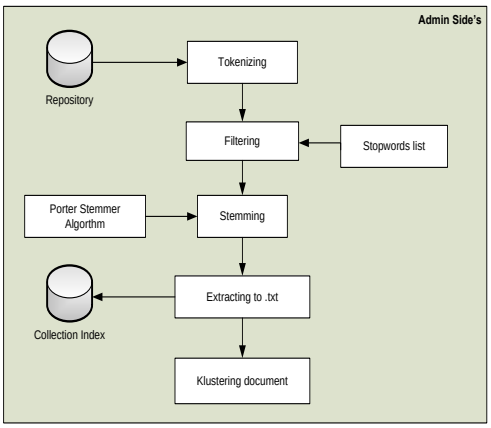


Gambar 4. Alur Sistem IRS dengan VSM

6.1 Rancangan

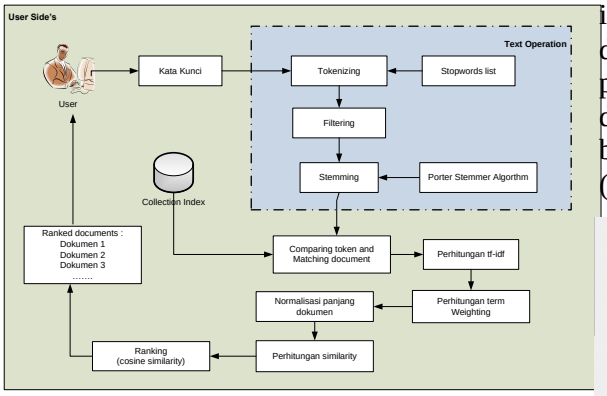
Sistem ini dirancang dengan dua sisi pengguna, sisi admin dan sisi user. Gambaran jelasnya seperti pada Gambar 5 dan Gambar 6.

Admin akan melakukan proses indexing dimana dalam proses ini akan dilakukan parsing dan stemming dokumen, yang setelah diextract ke dalam format .txt lalu dikelompokkan.



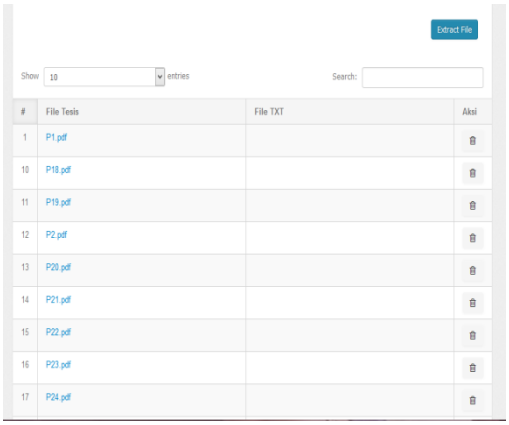
Gambar 5. Proses Indexing (sisi Admin)

Berikutnya adalah dari sisi user, dimana user hanya perlu memasukkan query untuk memulai penelusuran.



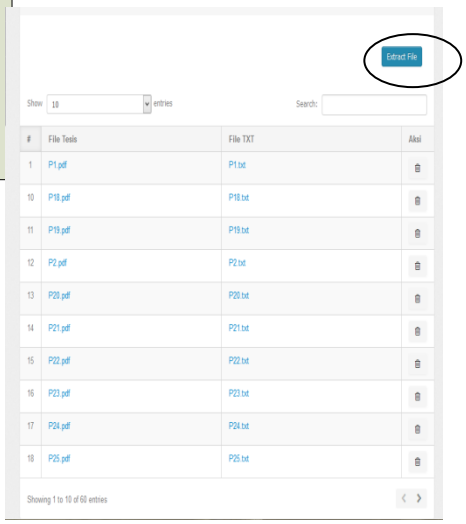
Gambar 6. Proses Testing (sisi User)

Interface sistem yang dirancang untuk aplikasi ini dikategorikan menjadi dua sesuai dengan kategori pengguna. Kategori admin, maka interface yang dirancang seperti pada Gambar 7, 8 dan 9.



Gambar 7. Interface Pengambilan dokumen

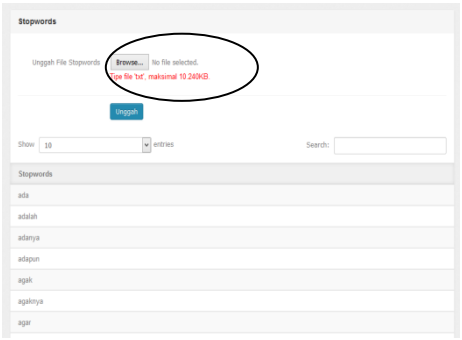
Gambar 7 menjelaskan dokumen yang dimasukkan dalam sistem adalah dokumen yang ada dalam repository tanpa ada perubahan apapun. Setelah itu, dokumen tersebut dikonversi atau di extract ke dalam bentuk .txt seperti pada Gambar 8. Proses ekstraksi dilakukan hanya dengan mngeksekusi button extract file pada aplikasi (Gambar 8, label a).



Gambar 8. Ekstraksi dokumen

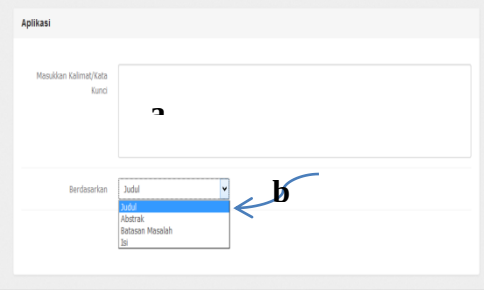
Untuk proses berikutnya yang masih menjadi hak admin adalah memasukkan data stopword, dan interface untuk memasukkan stopword ini terlihat dalam Gambar 9.

Memasukkan stopword dapat dilakukan dengan cara mengunggah file dengan format .txt dan dengan ukuran maksimal 1024KB (Gambar 9 label a).



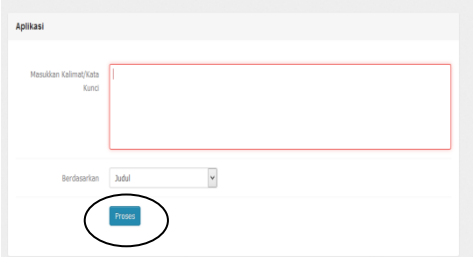
Gambar 9. Interface Unggah Stopword

Interface pertama bagi user adalah kolom untuk memasukkan query sebagai kata kunci untuk memulai pencarian, yang diperlihatkan pada Gambar 10 label a. Query yang dimasukkan adalah dalam bentuk kalimat-kalimat dan bukan file.

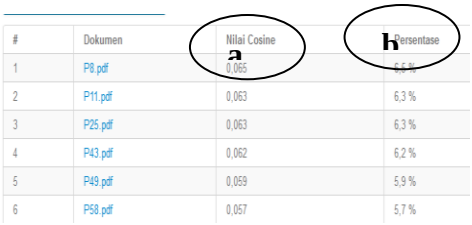


Gambar 10. Interface input query

Langkah selanjutnya setelah memasukkan query, user dapat memilih opsi pencarian yang terdiri dari 4 (empat) pilihan seperti dijelaskan pada Gambar 10 label b. Jika user telah memilih salah satu opsi maka akan nampak button proses, yang akan memulai eksekusi setelah user melakukan clicking seperti pada Gambar 11.



Gambar 11. Interface button Proses



Gambar 12. Interface hasil penelusuran

Gambar 12 adalah hasil hasil yang akan diberikan kepada user, dimana dokumen-dokumen tersebut telah diranking sesuai dengan nilai kemiripannya dengan query. Nilai kosinus (Gambar 12 label a) merupakan nilai kemiripan dengan nilai maksimal adalah 1 (satu). Semakin mendekati nilai 1 maka semakin mirip dokumen tersebut dengan query yang dimasukkan oleh user. Sedangkan persentase (label b) merupakan nilai kosinus yang dipersenkan.

6.2 Testing

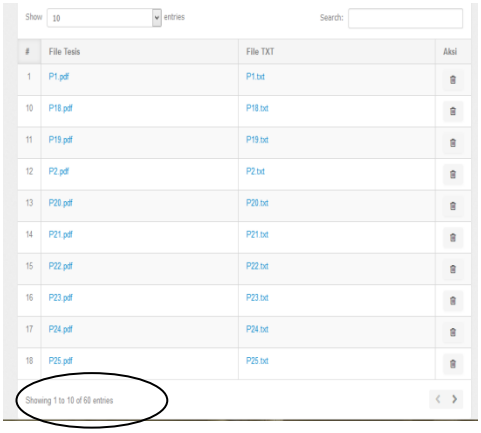
Untuk menguji rancangan sistem, penelitian ini membuat prototype dengan menggunakan 60 dokumen sebagai bahan testing seperti terlihat pada Gambar 13 label a. 60 dokumen tersebut memiliki kriteria yang terinci dalam Tabel 1.

Tabel 1. Daftar dokumen Testing

No.	Jumlah	Format	Keterangan
1.	50 (P1 - P50)	a. Dua kolom b. Sesuai Format	Dokumen ini adalah dokumen dengan format sesuai dengan item pilihan pencarian yang dibuat dalam penelitian ini, dimana tersedia abstrak, judul, serta batasan masalah.
2.	8 (P51 - P58)	a. Dua kolom b. Tidak sesuai Format	Dokumen ini dibuat dalam dua kolom tetapi satu item pilihan pencarian tidak ada, yaitu bataasan masalah.
3.	2 (P59 - P60)	a. Satu kolom b. Tidak sesuai format	Dokumen dengan format penulisan satu kolom/format IEEE ini juga tidak memiliki item batasan masalah.

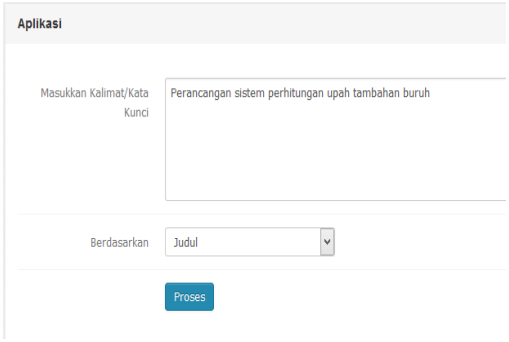
Dokumen yang tidak sesaui format dimasukkan sebagai data testing untuk mengetahui apakah dokumen tersebut memiliki kemungkinan untuk ditelusuri oleh sistem.

Testing dilakukan dengan semua opsi pilihan, namun yang disampaikan disini hanya untuk pencarian judul dan batasan masalah saja.



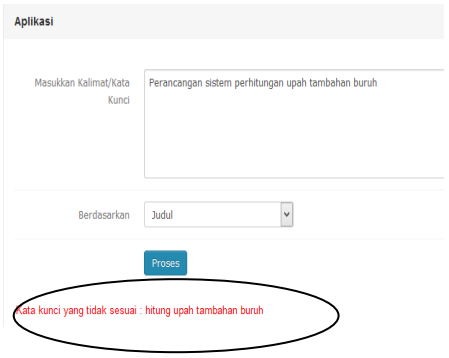
Gambar 13. Koleksi dokumen testing

Query untuk opsi pencarian judul, yang dimasukkan adalah 'Perancangan sistem perhitungan upah tambahan buruh", dijelaskan pada Gambar 14.



Gambar 14. Opsi penelusuran judul

Setelah dilakukan proses maka akan terlihat kata-kata dalam query yang tidak terdapat dalam koleksi dokumen seperti digambarkan pada Gambar 15 label a.



Gambar 15. Query yang tidak terdapat dalam kumpulan dokumen

Hasil akhir yang diberikan kepada user adalah urutan dokumen-dokumen sesuai dengan nilai kemiripannya dengan query user, seperti dijelaskan pada Gambar 16. Dokumen yang paling mirip adalah P8 dengan nilai kemiripan 6.5%.

#	Dokumen	Nilai Cosine	Persentase
1	P8.pdf	0,065	6,5 %
2	P11.pdf	0,063	6,3 %
3	P25.pdf	0,063	6,3 %
4	P43.pdf	0,062	6,2 %
5	P49.pdf	0,059	5,9 %
6	P58.pdf	0,057	5,7 %
7	P44.pdf	0,054	5,4 %
8	P17.pdf	0,053	5,3 %
9	P28.pdf	0,051	5,1 %
10	P54.pdf	0,050	5,0 %

Gambar 16. Hasi penelusuran

7. Penutup

Setelah dilakukan testing maka rancangan sistem ini dapat digunakan sebagai alat bantu untuk mendapatkan referensi dokumen yang mirip dengan query yang dimasukkan sehingga user dapat memperkirakan sejauh mana kemiripan penelitian yang akan dilakukannya.

Prototype masih memiliki error sistem ketika opsi pilihan user adalah isi, hal ini dimungkinkan karena keterbatasan hardware serta time limit execution dari bahasa pemrograman yang digunakan.

Untuk pengembangan penelitian, jika akan menggunakan algoritma Porter sebaiknya tambahkan tabel pelengkap kata bahasa Indonesia agar semua kata dapat dikenali sistem. Koleksi dokumen yang digunakan juga sebaiknya dengan format yang beragam dan akan lebih maksimal jika menggunakan database yang lebih luas sehingga aplikasi dapat digunakan secara bersama antar institusi.

8. Referensi

Agusta, Ledy; 2009, Perbandingan Algoritma Stemming Porter Dan Algoritma Nazief & Adriani Untuk Stemming Dokumen Teks Bahasa Indonesia, Konferensi Nasional Sistem dan Informatika, 14 November 2009, hal. 196-201, KNS&I09-036, https://yudiagusta.files.wordpress.com

Amin, Fatkhul; Pebruari 2012, Sistem Temu Kembali Informasi dengan Metode Vector Space Model, Jurnal Sistem Informasi Bisnis 02(2012)

Karyono, Giat; Fandy Setyo Utomo, 2012, Temu Balik Informasi Pada Dokumen Teks Berbahasa Indonesia Dengan Metode Vector Space Retrieval Model, Seminar Nasional Teknologi Informasi & Komunikasi Terapan (SEMANTIK), ISSN: 979-26-0255-0, 23 Juni 2012

Kurniasih, Nuning, Konsep Dasar Temu Kembali Informaasi, 10 Nopember 2015, https://www.academia.edu/6138333

Luan, RuPeng; Qian Zhang; JunFeng Zhang; Feng Yu, 2013, An Improved Vector Space Model to Retrieval Systems for Content Matching in Agricultural Information, 10th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), 978-1-4673-5253-6/13, June 2013

Riadi, Muchlisin; Information Retrieval System (IRS), 2012, 10 Nopember 2015, http://www.kajianpustaka.com

Salton, G., 1989, *Automatic Text Processing, The Transformation, Analysis, and Retrieval of information by computer*, Addison – Wesly Publishing Company, Inc. USA.

Tudesman ; Oktalina, Enny; Tinaliah; Yoannita; Sistem Plagiarisme Dokumen Bahasa Indonesia Menggunakan Metode *Vector Space Model*, 2014, http://eprints.mdp.ac.id