

Rancang Bangun Sistem Temu Kembali Informasi (Information Retrieval System) Dokumen Berbahasa Jawa menggunakan Metode DICE Similarity

by 08032023 Purwatiningtyas

Submission date: 08-Mar-2023 03:30PM (UTC+0700)

Submission ID: 2031986388

File name: Temu_Kembali_Informasi_Information_Retrieval_System__dinamik.pdf (461.01K)

Word count: 3187

Character count: 19970

Rancang Bangun Sistem Temu Kembali Informasi (*Information Retrieval System*) Dokumen Berbahasa Jawa menggunakan Metode *DICE Similarity*

Fatkahul Amin, Purwatiningsy, Edy Winarno

Fakultas Teknologi Informasi, Universitas Stikubank Semarang
email: fatkhulamin@gmail.com, diba_ian@yahoo.com, edywin@edu.unisbank.ac.id

Abstrak

Implementasi dokumen teks bahasa jawa masih banyak dijumpai baik secara online maupun offline. Pencarian dokumen teks bahasa jawa dengan menggunakan mesin pencari atau Sistem Temu Kembali Informasi (STKI) yang tersedia menghasilkan suatu hasil pencarian dengan dokumen terambil yang banyak (*recall* tinggi) sehingga hasil menjadi kurang akurat (*precision* rendah). Tujuan pembuatan rancang bangun STKI dengan metode *DICE Similarity* agar *user* mudah dalam melakukan pencarian dokumen teks berbahasa Jawa. *Software* STKI dirancang untuk memberikan hasil pencarian dokumen dalam jumlah *recall* rendah dan *precision* tinggi menggunakan metode pemeringkatan *DICE Similarity*, sehingga *user* akan mendapatkan hasil pencarian cepat dan akurat. Metode *DICE Similarity* akan melakukan pembobotan tiap dokumen yang ada pada database sehingga antar dokumen memiliki bobot yang berbeda untuk menentukan dokumen mana yang paling mirip (similar) dengan *query*. Rangka teratas akan ditempati oleh dokumen dengan bobot tertinggi hasil pencarian Evaluasi hasil pencarian IRS dilakukan dengan uji *recall* dan *precision* dengan model persepsi. STKI mampu melakukan pencarian dokumen dan menampilkan hasil pencarian dokumen memiliki rata-rata *recall* 0,04 dan rata-rata *precision* 0,83. Hasil STKI dilengkapi dengan bobot tiap dokumen dan letaknya yang akan memudahkan *user* dalam pencarian dokumen teks bahasa Jawa.

Kata-Kunci: STKI, Basa Jawa, *Dice Similarity*

1. PENDAHULUAN

1.1 Latar Belakang

Bahasa Jawa sebagai bahasa yang paling banyak digunakan di wilayah Indonesia setelah bahasa Indonesia, dewasa ini mulai banyak ditinggalkan oleh kebanyakan orang. Media offline dan media online juga kurang mengangkat bahasa Jawa sehingga dikhawatirkan bahasa Jawa lama-kelamaan akan ditinggalkan oleh bangsa kita. Beberapa media online berbahasa Jawa ada, namun belum menggunakan atau belum menyediakan pencarian informasi menggunakan mesin pencari khusus berbahasa jawa. Bahasa daerah adalah bahasa yang terkait akan latar belakang etnis, suku, budaya, yang begitu kaya di Indonesia. Bahasa daerah mencerminkan identitas bangsa ini, cermin kita sebagai bangsa yang kaya akan budaya dan bahasa. Bangsa Indonesia memiliki sekitar 700 lebih bahasa daerah, tetapi yang tercatat oleh Kementrian

Pendidikan dan Kebudayaan (Kemendikbud) hanya sekitar 450 saja.

Ada dua faktor utama yang menyebabkan bahasa Jawa (bahasa daerah pada umumnya) ditinggalkan oleh masyarakat, yaitu faktor internal dan faktor eksternal. Adapun Faktor internal yang dimaksud; 1) Melemahnya Sosialisasi dalam Keluarga, 2) Disorientasi Kurikulum Pendidikan, dan 3) Kurangnya Kesadaran Generasi Muda. Sedangkan Faktor eksternal yang menjadi penyebabnya yaitu; 1) Modernisasi dan Globalisasi, 2) Eksistensi Bahasa Asing di Indonesia, dan 3) Dominasi Kultural.

Pencarian informasi saat ini dilakukan dengan menggunakan sistem temu kembali informasi (STKI) atau mesin pencari, *user* menuliskan *query* dan mesin pencari akan menampilkan hasil pencarian. STKI yang sudah ada dan banyak digunakan saat ini memberikan hasil perolehan pencarian yang banyak (banyak

dokumen yang terambil), sehingga diperlukan waktu untuk menentukan hasil pencarian yang relevan. Menentukan hasil yang relevan sesuai dengan keinginan user dengan jumlah hasil pencarian yang banyak akan menyulitkan pengguna (*user*). Hal ini terjadi karena dokumen yang terambil oleh sistem jumlahnya banyak, maka sistem berkemungkinan menampilkan hasil pencarian yang tidak relevan. Banyaknya dokumen hasil pencarian ini membuat waktu yang dibutuhkan dalam pencarian menjadi lebih banyak dari yang diharapkan.

Perkembangan penelusuran informasi saat ini menghasilkan *recall* yang tinggi dan *precision* yang rendah. *Recall* yang tinggi diartikan bahwa dokumen yang dihasilkan dalam penelusuran dokumen adalah banyak, sedangkan *precision* rendah dapat diartikan bahwa dokumen yang diharapkan dapat ditemukan sedikit.

Solusi untuk mengatasi masalah ini adalah dengan membuat *software Information Retrieval System (IRS)* menggunakan metode *DICE Similarity* yang didukung oleh *Stemmer Bahasa Jawa*. Metode *Dice Similarity* dipilih karena cara kerja model ini efisien, mudah dalam representasi dan dapat diimplementasikan pada *document-matching*. *Software IRS* diharapkan menghasilkan *recall* rendah dan *precision* tinggi.

1.2 Rumusan Masalah

Bagaimana membuat Rancang Bangun Sistem Temu Kembali Informasi (*Information Retrieval System*) Dokumen Berbahasa Jawa Menggunakan Metode *DICE Similarity*.

1.3 Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini adalah Membuat Sistem Temu Kembali Informasi (*Information Retrieval System*) Dokumen Berbahasa Jawa Menggunakan Metode *DICE Similarity*

2. METODE

2.1. Pengumpulan data

Pengumpulan data dimaksudkan agar mendapatkan bahan-bahan yang relevan, akurat dan *reliable*. Maka teknik pengumpulan data

yang dilakukan dalam penelitian ini adalah sebagai berikut :

a. Observasi

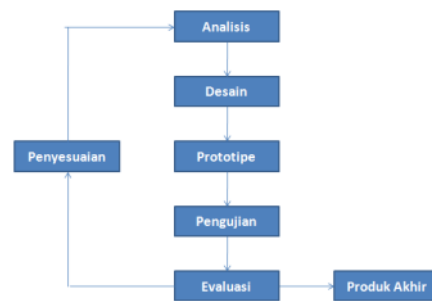
Melakukan pengamatan dan pencatatan secara sistematis tentang hal-hal yang berhubungan dengan basis data dokumen teks bahasa jawa yang sesuai kebutuhan. Melakukan pengamatan materi bahasa Bahasa Jawa dan menganalisis Struktur Bahasa Jawa pada majalah Penjebar Semangad.

b. Studi Pustaka

Pengumpulan data dari bahan-bahan referensi, arsip, dan dokumen yang berhubungan dengan permasalahan dalam penelitian ini. Studi pustaka dilakukan dengan cara online dengan melihat web dan video-video Bahasa Jawa.

2.2. Metode Pengembangan

Penelitian ini menggunakan model *prototype*. Di dalam model ini sistem dirancang dan dibangun secara bertahap dan untuk setiap tahap pengembangan dilakukan percobaan-percobaan untuk melihat apakah sistem sudah bekerja sesuai dengan yang diinginkan. Sistematis model *prototype* terdapat pada Gambar 1 memperlihatkan tahapan pada *prototype*.



Gambar 1. prototipe

Berikut adalah tahapan yang dilakukan pada penelitian ini dengan metode pengembangan *prototype*

a. Analisa

Pada tahap ini dilakukan analisa tentang masalah penelitian dan menentukan pemecahan masalah yang tepat untuk menyelesaikannya. Menentukan tujuan pembuatan mesin pencari.

b. Disain

Pada tahap ini dibangun rancangan Sistem Temu Kembali Informasi bahasa jawa (DFD dan Flow Chart)

c. Prototype

Pada tahap ini dibangun Sistem Temu Kembali Informasi Bahasa Jawa. Tahap ini di mulai dari proses tokenisasi, Penyaringan (filtering), Pembuatan kata dasar bahasa jawa (stemming), tfidf, dan perhitungan *Dice Similarity* yang diaplikasikan dengan program PHP.

d. Pengujian

Pada tahap ini dilakukan pengujian *Recal* dan *Precision* dengan model Persepsi

e. Evaluasi

Pada tahap ini dilakukan evaluasi apakah performa aplikasi sudah sesuai dengan yang diharapkan, apabila belum maka dilakukan penyesuaian-penyesuaian secukupnya.

f. Penyesuaian

Tahap ini dilakukan apabila pada evaluasi performa aplikasi kurang memadai dan dibutuhkan perbaikan, tahap ini melakukan penyesuaian dan perbaikan pada aplikasi sesuai dengan kebutuhan

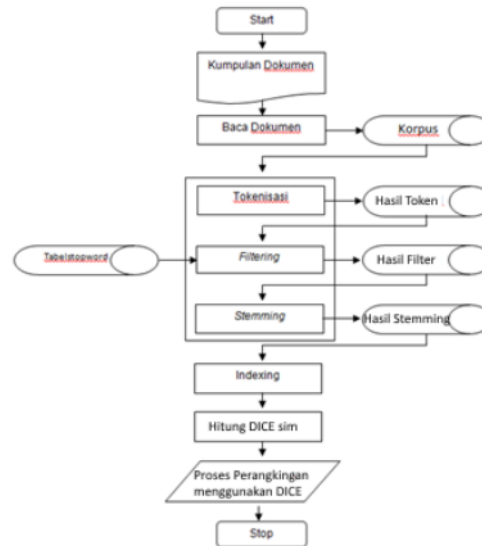
3. HASIL

3.1. Perancangan Sistem Temu Kembali Informasi (STKI)

a. Flow Chart STKI Jawa

Perancangan STKI didesain agar mudah digunakan oleh pengguna dalam hal ini oleh pengguna mesin pencari. STKI didesain untuk menemukan informasi secara akurat kepada pengguna (user). Proses STKI oleh sistem melalui proses-proses seperti gambar 2. *Flowchart* diawali dengan *input* dokumen-

dokumen kedalam korpus. Selanjutnya dokumen melalui proses preprosesing, dihitung bobotnya dan dibuat rankingnya berdasarkan bobot dokumen yang tertinggi. Hasil STKI adalah dokumen yang relevan dengan permintaan *user*.. Gambar 2 menunjukkan diagram alir STKI.



Gambar 2. Diagram Alir STKI.

b. Perancangan Tabel

Pada STKI ini menggunakan beberapa tabel untuk tempat meletakkan kumpulan data pada korpus, *term-term* hasil proses Tokenizing, *Filtering* dan *Stemming*. Selanjutnya untuk proses perhitungan dan pemeringkatan *Dice Similarity* digunakan tabel *freq* yaitu tabel yang berisi kumpulan *term-term* yang telah menjadi kata dasar. Berikut ini Rancangan tabel yang akan digunakan dalam STKI pada penelitian ini;

c. Rancangan Tampilan STKI Jawa

STKI bekerja melalui beberapa tahapan proses sebagai berikut; Proses Tokenisasi dilakukan dengan mekanisme jika dokumen pada korpus ditemukan spasi, maka *term* yang ada diantara spasi akan di *retrieved* (akan diambil oleh sistem) kemudian *term* ditempatkan dalam tabel tabelawal. Hasil proses berupa *term* asli (*term* yang masih memiliki imbuhan, tanda baca yang melekat, dan angka). Keunggulan proses ini waktu komputasi cepat.

Proses *Filtering* dilakukan dengan mekanisme jika *term* pada tabel tabelawal ditemukan tanda baca, huruf kapital, dan angka. Maka program akan menghilangkan (tanda baca dan angka) dan mengganti (huruf kapital menjadi huruf kecil), kemudian memeriksa *term* dengan *stopwords*. Hasil proses berupa *term* pilihan (tanpa tanda baca, tanpa huruf kapital, dan bukan termasuk *stopwords*). Keunggulan proses ini sistem mampu mereduksi tanda baca, angka, merubah *term* menjadi huruf kecil, dan memeriksa *term* *stopwords* dengan waktu komputasi yang cepat. Proses *Stemming* dilakukan program dengan cara menghilangkan imbuhan yang terdapat pada *term* hasil *filtering*. Proses menghilangkan dilakukan dengan menghilangkan awalan, sisipan, dan akhiran. Hasil proses ini dimasukkan dalam tabel *tablefreq*. Hasil proses berupa *term* kata dasar.

Proses Pembobotan dokumen dengan metode VSM dilakukan dalam proses pencarian dokumen. Program akan bekerja ketika *user* melakukan *query*, selanjutnya program akan memproses *query* tersebut dengan perhitungan-perhitungan *tf*, *idf*, *tfidf*, jarak *query* dan dokumen, similaritas dan *cosine similarity*. Hasil proses pembobotan dengan metode VSM berupa dokumen hasil pencarian disertai dengan bobot dokumen, letak dokumen dan disusun *descending* (dokumen dengan bobot terbesar diletakkan di atas). Selengkapnya proses atau cara kerja STKI mulai dari persiapan dokumen hingga pencarian dokumen beserta hasilnya akan dibahas pada proses STKI.

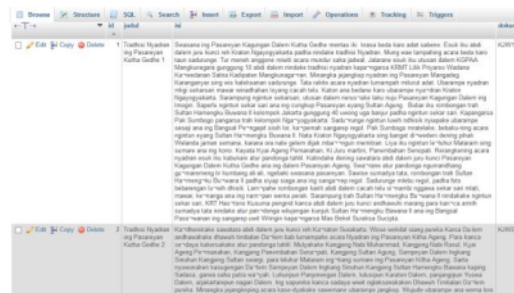
4. PEMBAHASAN

a. Implementasi STKI Bahasa Jawa metode Dice Similarity

Dokumen abstrak di *input* dengan cara manual dengan format dokumen teks. Proses ini dilakukan dengan cara memasukkan abstrak-abstrak skripsi bahan kajian penelitian kedalam tabel korpus. Sebelum dimasukkan kedalam tabel, dibuat satu tabel dengan nama tabel korpus yang digunakan sebagai tempat data. Tabel korpus ini memiliki *field-field* id, judul, isi dan dokumen. *Field* id berisi urutan data penelitian didalam korpus yang tersusun sesuai dengan urutan input data. *Field* judul berisi judul

skripsi. *Field* isi berisi abstrak skripsi dan *field* dokumen berisi nama dokumen dengan kode tertentu. Proses memasukkan dokumen ke dalam tabel korpus ini memerlukan waktu relative lama bergantung pada jumlah data yang akan di *input* kedalam tabel korpus (Tabel 1).

Tabel 1. Tabel Korpus



b. Proses Tokenizing

Proses *scanner* dokumen korpus menggunakan format teks dilakukan dengan cara masuk kedalam dokumen korpus melalui perantara program php ke dalam database mysql. Proses *scanner* data dilakukan dengan cara *scanner* baris per baris, untuk tiap-tiap file naskah yang ada di dokumen. Tokenizing dimulai dengan memisahkan *term-term* yang ada pada dokumen korpus menjadi kumpulan *term* melalui proses *scanner* dengan dasar spasi. Selanjutnya *term* hasil proses Tokenizing di masukkan kedalam tabel awal dengan menyertakan *field-field* judul, *term* dan dokumen. Proses Tokenizing dilakukan dengan dua tahap yaitu tahap *scanner term* pada korpus kemudian *term* hasil *scanner* dimasukkan ke tabelawal dan tahap berikutnya adalah *scanner term* pada tabel awal dan menempatkan *term* hasil *scanner* di tabel kedua. Hasil *scanner file* pada proses (tabel 2)

Tabel 1. Tabel awal

judul	term	dokumen
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	swasana	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	ing	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	pasareyan	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	kagungan	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	dalem	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	kutha	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	gedhe	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	mentas	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	iki	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	krasa	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	beda	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	karo	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	adat	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	sabene.	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	esuk	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	iku	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	abdi	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	dalem	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	juru	KJW1

c. Proses Filtering

Proses selanjutnya setelah proses Tokenizing adalah proses *filtering*. Proses filtering dilakukan untuk menghilangkan *term-term* yang tidak memiliki arti dengan menggunakan *stopword list* tala. Proses filtering adalah proses baca tabel kedua untuk diperiksa apakah semua term memiliki term-term yang termasuk dalam *stopword list* menurut tala. Jika dalam tabel kedua terdapat *term-term* yang termasuk dalam *stopword*, maka akan dilakukan penghilangan *term-term* tersebut. Hasil proses filtering selanjutnya dimasukkan dalam tabel freq (tabel 2)

Tabel 2. Proses Filtering

judul	term	dokumen
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	swasana	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	ing	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	pasareyan	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	kagungan	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	dalem	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	kutha	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	mentas	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	beda	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	adat	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	sabene.	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	esuk	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	iku	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	abdi	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	dalem	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	juru	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	kunci	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	reh	KJW1
Tradhisi Nyadran ing Pasareyan Kutha Gedhe 1	kraton	KJW1

d. Proses stopword Removal

Proses membuang *stopword (stopword removal)* dilakukan untuk menghilangkan *term-term* yang tidak memiliki arti dengan menggunakan *stopword jawa*. Proses ini dilakukan dengan cara *scanner* dan scanner tabel kedua

e. Proses Stemming

Proses *stemming* yang digunakan adalah proses *stemmer* menggunakan *stemmer* untuk bahasa Jawa ngoko berdasarkan *stemmer* bahasa Indonesia yang dibuat Tala. Proses stemming dengan menggunakan *stemmer* jawa melalui beberapa tahapan dan untuk mendukung proses ini juga digunakan *stopword list jawa*. Hasil akhir dari proses stemming adalah kumpulan *term* yang sudah menjadi kata dasar yang diinput dalam tabel *freq*. Proses *stemming* menghasilkan kumpulan term berupa kata dasar hasil scanner *term* pada tabel kedua. Proses *stemming* didukung *stopword jawa* yang digunakan untuk mengurangi term yang ada pada tabel kedua. Selanjutnya *term* hasil *stemming* di letakkan pada tabel *freq*

Tabel 3. Proses Stemming

judul	term	freq	freqpangkat
Upacara Ruwatan lan Crita Murwakala	crita	11	121
Upacara Ruwatan lan Crita Murwakala	bab	8	64
Upacara Ruwatan lan Crita Murwakala	lelako	1	1
Upacara Ruwatan lan Crita Murwakala	bethar	1	1
Upacara Ruwatan lan Crita Murwakala	kala	4	16
Upacara Ruwatan lan Crita Murwakala	kamot	2	4
Upacara Ruwatan lan Crita Murwakala	ing	134	17956
Upacara Ruwatan lan Crita Murwakala	buku	6	36
Upacara Ruwatan lan Crita Murwakala	*pakem	5	25
Upacara Ruwatan lan Crita Murwakala	gruwat	3	9
Upacara Ruwatan lan Crita Murwakala	murwakala*	4	16
Upacara Ruwatan lan Crita Murwakala	karang	1	1
Upacara Ruwatan lan Crita Murwakala	kyai	9	81
Upacara Ruwatan lan Crita Murwakala	demang	1	1
Upacara Ruwatan lan Crita Murwakala	reditanay	3	9
Upacara Ruwatan lan Crita Murwakala	nuduh	2	4

f. Proses Indexing

Proses *indexing* dilakukan untuk mengambil atau meretrieve *term-term* yang ada pada tabelfreq untuk selanjutnya diproses pada saat pencarian dilakukan oleh STKI. Proses perhitungan dilakukan langsung pada STKI saat *query* diproses oleh sistem. User memasukkan Kata Kunci (*query*) pada mesin pencari, kemudian setelah kata kunci ditulis mesin pencari akan melakukan pencarian *query* pada *database* dengan mengolahnya terlebih dahulu sesuai dengan arsitektur mesin pencari menggunakan metode *vector space model* dan memberikan hasil pencarian.

g. Proses Perhitungan Dice Similarity

STKI metode Dice akan melakukan proses perhitungan dimulai dari menghitung *tfidf*, menghitung jarak *query* dan jarak dokumen, menghitung similaritas produk, dan menghitung bobot dokumen. STKI akan mengeksekusi *query* dari *user* dan akan mengolah *query* tersebut. *Query* yang di *input* oleh user selanjutnya akan dilakukan pencarian pada tabel freq kemudian dilakukan perhitungan pembobotan menggunakan metode **Dice Similarity**. Perhitungan dilakukan dalam sistem pencarian, sistem pencarian akan melakukan perhitungan kemudian akan menampilkan hasilnya. Hasil pencarian akan menampilkan nama dokumen di korpus, kemudian bobot similaritas dan disusun

berdasarkan perankingan. Bobot terbesar akan menempati ranking teratas pada hasil pencarian.

Dice Similarity adalah metode untuk melihat tingkat kedekatan atau kesamaan (*smilarity*) term dengan cara pembobotan *term*. Dokumen dipandang sebagai 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*. VSM memberikan sebuah kerangka pencocokan parsial adalah mungkin. Hal ini dicapai dengan menetapkan bobot non-biner untuk istilah indeks dalam *query* dan dokumen. Bobot istilah yang akhirnya digunakan untuk menghitung tingkat kesamaan antara setiap dokumen yang tersimpan dalam sistem dan permintaan user. Dokumen yang terambil disortir dalam urutan yang memiliki kemiripan, model vektor memperhitungkan pertimbangan dokumen yang relevan dengan permintaan user. Hasilnya adalah himpunan dokumen yang terambil jauh lebih akurat (dalam arti sesuai dengan informasi yang dibutuhkan oleh *user*).

Dice similarity merupakan metode yang digunakan untuk menghitung tingkat kesamaan (*similarity*) antar dua buah objek. Untuk notasi himpunan dapat digunakan rumus (1):

$$S_{Dice} = \frac{2 \sum_{i=1}^d P_i Q_i}{\sum_{i=1}^d P_i^2 + \sum_{i=1}^d Q_i^2}$$

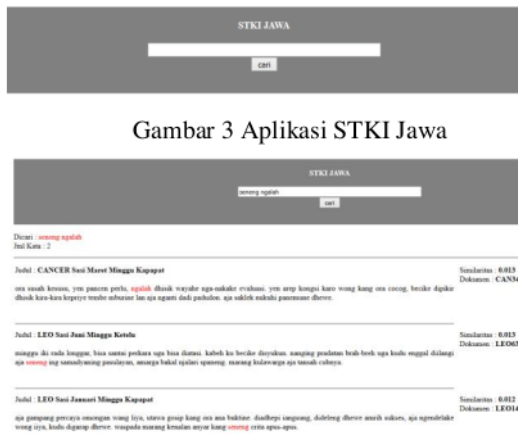
dimana p dan q adalah dokumen yang berbeda. *p_i* adalah term i yang ada di dokumen p *q_i* adalah term i yang ada di dokumen q.

h. Aplikasi STKI

STKI Dice dirancang agar *user* mudah menggunakan dalam mencari dokumen yang relevan. Tampilan (*interface*) juga dirancang seperti mesin pencari pada umumnya, sehingga siapapun usernya akan langsung mudah beradaptasi dalam menggunakan mesin pencari. Prosedur menggunakan STKI ini sangat mudah, yaitu *user* hanya perlu menuliskan *query* atau

kata kunci yang akan di cari pada kotak dialog kemudian setelah *query* di masukkan *user* tinggal mengklik tombol cari atau tekan *enter*.

Studi kasus pada aplikasi STKI Dice ini menggunakan dokumen-dokumen Basa Jawa pada Majalah Online Penjebar Semangad yang terdapat pada 3 kategori yaitu; Kejawan, kebatinan, dan pasujarahan. *Query* yang dimasukkan pada Information Retrieval System adalah *keyword* dengan 1 *term* yaitu "*crita*", "*uwong*", 2 *term* "*Crita Rakyat*", 3 *term* "*crita rakyat surakarta*", 4 *term* "*crita rakyat uwong surakarta*" 5 *term* "*Juru kunci pasareyan kutha gedhe*". Gambar 3 dan 4. Menunjukkan Aplikasi STKI.



Gambar 3 Aplikasi STKI Jawa

Gambar 4 Hasil Pencarian *keyword*

Hasil pencarian dokumen dengan *keyword* "*seneng ngalah*", menunjukkan dokumen dengan bobot tertinggi adalah dokumen letak dokumen CAN34 (bobot 0,013). Dokumen CAN34 (dokumen Palintangan CANCER Sasi Maret Minggu Kapapat nomer 34) memiliki bobot tertinggi atau memiliki tingkat kemiripan tertinggi dibandingkan dengan dokumen lain yang ada pada korpus.

i. **Pengujian recall dan precision**

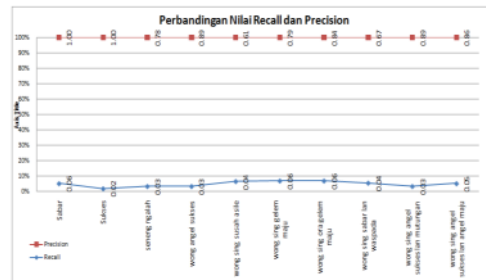
Pengujian *recall* (*P*) dan *precision* (*R*) dilakukan dengan cara *input query* ke dalam Information Retrieval System *input* 1 *term*, 2 *term* dan 3 *term*, 4 *term*, dan 5 *term*. Perhitungan *recall* dan *precision* menggunakan persamaan (8) dan persamaan (9). Hasil pengujian *recall*

dan *precision* dengan menguji 1 *term*, 2 *term* dan 3 *term* sampai dengan 5 *term* menunjukkan bahwa jika *recall* rendah maka *precision* akan tinggi, selengkapnya terlihat pada tabel 1.

Tabel 1 Hasil Pengujian *Recall* dan *Precision*

No	Query	Recall	Precision
1	Sabar	0.06	1.00
2	Sukses	0.02	1.00
3	seneng ngalah	0.03	0.78
4	wong angel sukses	0.03	0.89
5	wong sing susah asile	0.04	0.61
6	wong sing gelem maju	0.06	0.79
7	wong sing ora gelem maju	0.06	0.84
8	wong sing sabar lan waspada	0.04	0.67
9	wong sing angel sukses lan mutungan	0.03	0.89
10	wong sing angel sukses lan angel maju	0.05	0.86

Hasil uji recall dan precision berdasarkan persepsi bisa dilihat pada gambar 5



Gambar 5 Diagram hasil uji Recall dan precision

5. SIMPULAN DAN SARAN

5.1. Kesimpulan

- a. STKI mampu melakukan pencarian dokumen teks bahasa jawa dan menampilkan hasil pencarian dokumen teks berbahasa Jawa dengan disertai bobot tiap dokumen beserta letak dokumen dengan metode DICE Similarity.
- b. Hasil Uji *recall* dan *precision* STKI menunjukkan hasil pencarian dokumen teks memiliki rata-rata *recall* = 0,04 dan rata-rata *precision* = 0,83

5.2. Saran

- a. Stemmer Jawa masih perlu perbaikan untuk Proses *stemming* perlu diperbaiki karena hasil yang didapatkan masih belum bisa

sepenuhnya membuat semua *term* kedalam bentuk *term* kata dasar dengan benar.

- b. Penulisan kata atau dokumen dalam bahasa Jawa harus dilakukan dengan benar agar mendapatkan hasil tem kata dasar yang benar

6. UCAPAN TERIMA KASIH

Ucapan Terima kasih kami haturkan kepada;

- a. Bapak Dr. H. Hasan Abdul Rozak, S.H., C.N.,M.M. selaku Rektor Universitas Stikubank (Unisbank) Semarang.
- b. Ibu Dr. Endang Tjahjaningsih, S.E., M.Kom, selaku Ketua Lembaga Penelitian dan pengabdian Masyarakat (LPPM) Universitas Stikubank (Unisbank) Semarang.
- c. Bapak Dr. Drs. Y. Suhari, M.MSI, selaku Dekan fakultas Teknologi Informasi Universitas Stikubank (Unisbank) Semarang.
- d. Rekan-rekan dosen yang telah memberikan masukan-masukan untuk perbaikan dan kesempurnaan penulisan laporan ini.

7. DAFTAR PUSTAKA

- Budi, I., Aji, R.F., 2006. Efektifitas Seleksi Fitur dalam Sistem Temu Kembali Informasi. Seminar Nasional Aplikasi Teknologi Informasi (SNATI), ISSN : 1907-5022.
- Bum, K.Y., 2010. *An autonomous assessment system based on combined latent semantic kernels. Expert Systems with Applications: An International Journal* , Volume 37 Issue 4.
- Kadir, A., 2001. Dasar Pemrograman Web Dinamis menggunakan PHP. Penerbit Andi. Yogyakarta.
- Khuat Thanh Tung , (2016) A Comparison of Algorithms used to measure the Similarity between two documents, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 4 Issue 4, April 2015
- Manning, C., Raghavan, P., 2007. An Introduction to Information Retrieval, Stanford. USA.
- Manoj Chahal, 2016. Information Retrieval using Dice Similarity Coefficient , International Journal of Advanced Research in Computer Science and Software Engineering, Volume 6, Issue 6, June 2016, ISSN: 2277 128
- Meadow, C.T., 1997. *Text Information Retrieval Systems*. Academic Press. New York.
- Tala, F.Z., 2003, *A Study of Stemming Effects on Information Retrieval in bahasa Indonesia*. Institut for logic, Language and Computation Universiteit van Amsterdam The Netherlands.
- R. Umamaheswari, K. Rajesh, 2014, Text Clustering Using Cosine Similarity and Matrix Factorization Cosine Similarity, International Journal of Research in Computer and Communication Technology, Vol 3, Issue 10, October - 2014
- Salton, G., 1989, *Automatic Text Processing, The Transformation, Analysis, and Retrieval of information by computer*. Addison – Wesley Publishing Company, Inc. USA.
- Vikas Thada , 2015. Comparison of Jaccard, Dice, Cosine Similarity Coefficient To Find Best Fitness Value for Web, Department of Computer Science and Engineering Dr. K.N.M University, Newai, Rajasthan, India
- Yates, R.B, 1999. *Modern Information Retrieval*, Addison Wesley-Pearson international edition, Boston. USA.

Rancang Bangun Sistem Temu Kembali Informasi (Information Retrieval System) Dokumen Berbahasa Jawa menggunakan Metode DICE Similarity

ORIGINALITY REPORT

16%

SIMILARITY INDEX

13%

INTERNET SOURCES

7%

PUBLICATIONS

3%

STUDENT PAPERS

MATCH ALL SOURCES (ONLY SELECTED SOURCE PRINTED)

2%

★ publikasi.uyelindo.ac.id

Internet Source

Exclude quotes On

Exclude matches < 1%

Exclude bibliography On