

## PENERAPAN SISTEM TEMU KEMBALI PADA SEBUAH SISTEM PENDUKUNG PENGAMBILAN KEPUTUSAN LOWONGAN PEKERJAAN BERBAHASA INDONESIA

<sup>1</sup>AG Hari Jati Nugraha , <sup>2</sup>Puspaningtyas Sanjaya Adi

<sup>1,2</sup>Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Sanata Dharma  
Kampus III Paingan, Maguwoharjo, Sleman Yogyakarta  
E-mail: <sup>1</sup>ari\_vdc212@yahoo.com, <sup>2</sup>puspa@staff.usd.ac.id

### ABSTRAKS

Dalam tulisan ini dijelaskan bagaimana mengintergrasi sistem temu kembali kedalam sebuah sistem pendukung pengambilan keputusan. Algoritma yang digunakan adalah TF-IDF menurut savoy (1993). Sistem yang dibangun dapat memberikan rekomendasi kepada pelamar pekerja lowongan perusahaan yang cocok dengan dokumen skill yang dimiliki pelamar. Dan untuk perusahaan sistem dapat memberikan rekomendasi pekerja yang sesuai dengan lowongan yang dimiliki perusahaan.

*Kata Kunci: sistem temu kembali, sistem pendukung pengambilan keputusan*

## 1. PENDAHULUAN

### 1.1 Latar Belakang

Banyak orang membutuhkan informasi lowongan kerja dengan cepat, akurat tanpa terhalang oleh jarak dan waktu. Demikian pula perusahaan juga memerlukan pekerja yang sesuai standar mereka. Perusahaan juga ingin mendapatkan pelamar-pelamar yang terbaik dan handal sesuai dengan kebutuhan perusahaan tersebut dengan waktu yang relatif singkat. Tidak adanya aturan khusus dalam penulisan CV (*curriculum vitae*) lowongan pekerjaan membuat informasi yang terkandung didalamnya terkadang terlalu detail ataupun terlalu umum. Hal ini menyebabkan sulit dalam penyaringan pelamar pekerjaan. Secara manual penyaringan pelamar dilaksanakan dengan cara membaca CV dari pencari kerja lalu perusahaan melakukan seleksi dan klasifikasi atas kesesuaian cv dengan lowongan pekerjaan. Kegiatan ini merupakan suatu model proses pemerolehan informasi atas data-data yang dimiliki oleh suatu organisasi atau lembaga. Bagaimana membangun sebuah rancangan sistem informasi yang dapat menjawab permasalahan ini

#### Rumusan Masalah

Judul yang baru saja Anda baca, juga dibuat dengan style Heading 2.

- Bagaimana memfasilitasi pencari kerja yang mencari lowongan ?
- Bagaimana memfasilitasi perusahaan yang mencari pekerja?
- Bagaimana membuat rekomendasi berdasarkan informasi yang tidak terstruktur?

### 1.2 Tujuan Penelitian

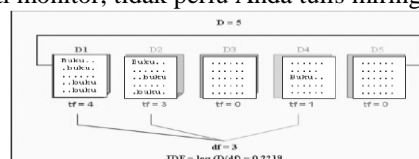
Tujuan dari skripsi ini adalah membangun sebuah software rekomendasi lowongan kerja yang berbasis aplikasi web, sedangkan manfaat software ini adalah memudahkan para pelamar mencari pekerjaan dan memudahkan perusahaan untuk mencari pekerja

## 2. METODE

### 2.1 Algoritma TF-IDF

Panjang tulisan antara 4-8 halaman, termasuk lampiran. Font yang digunakan adalah Times New Roman dengan style dan ukuran sebagaimana contoh dalam template ini, kecuali font untuk penulisan algoritma atau program yang akan dijelaskan lebih terperinci dalam bagian tersendiri. Untuk optimalisasi halaman, usahakan jumlah halaman genap.

Perhatikan penggunaan bahasa. Gunakan Bahasa Indonesia yang baku untuk ragam ilmiah. Jika Anda menggunakan istilah asing yang belum diserap ke dalam Bahasa Indonesia, tuliskan *italic* (miring). Jika istilah tersebut sudah terserap ke dalam Bahasa Indonesia atau sudah lazim di dunia informatika, seperti monitor, tidak perlu Anda tulis miring.



Gambar 1. Ilustrasi Algoritma TF-IDF

- D1, D2, D3, D4, D5 = dokumen
- D = Total Dokumen
- TF= merupakan frekuensi dari sebuah istilah dalam sebuah dokumen
- IDF= jumlah dokumen yang mengandung istilah  
 $W_{ik} = ntf_{ik} * nidf_k$ ,

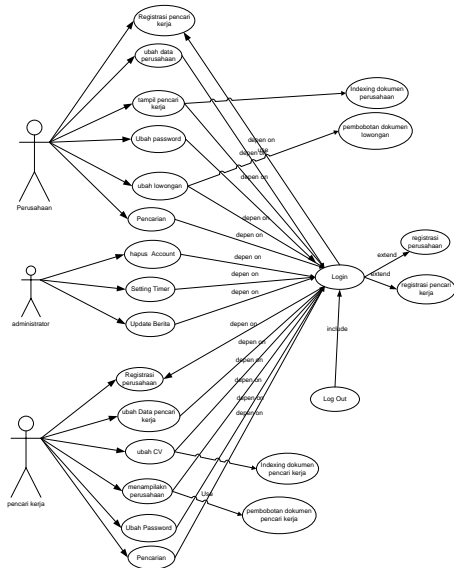
$$\text{dimana } ntf_{ik} = \frac{tf_{ik}}{\text{Max}_j tf_{ij}} \text{ dan } nidf_k =$$

$$\frac{\log\left[\frac{n}{df_k}\right]}{\log(n)}$$

- e.  $Wd = Wik + Wjk + Whk \dots + Wnk$
- f. Query = i or j or h or.....n
- g. Setelah bobot masing dokumen (Wd) diketahui maka dilakukan pengurutan atau sorting dimana semakin besar bobot dokumen (Wd) maka semakin besar tingkat kemiripan (similaritas) dokumen terhadap kata (query) yang dicari, dan demikian sebaliknya.

### 3. ANALISA KEBUTUHAN, PERANCANGAN LOGIS, PENGUJIAN

#### 3.1 Analisa kebutuhan



Gambar 2 Use Case Actor

Dibawah ini tabel penjelasan Gambar 2 Use Case Actor

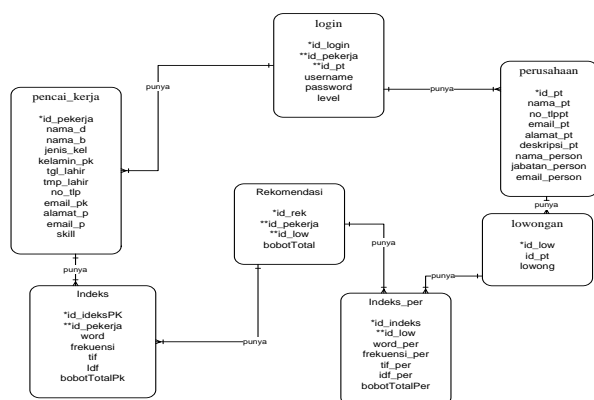
Tabel 1. Penjelasan Use Case Actor

Aktor	Keterangan
Admin	<ul style="list-style-type: none"> <li>a. Orang yang berperan sebagai administrator</li> <li>b. Menangani manajemen account.</li> <li>c. Orang dapat menambah, mengedit, menghapus berita.</li> <li>d. Orang yang dapat menyetting waktu pembuatan rekomendasi.</li> <li>e. Melakukan update pada website ini</li> </ul>
Pencari Kerja	<ul style="list-style-type: none"> <li>a. Adalah pencari kerja yang mendaftar sebagai member pada situs atau website ini.</li> <li>b. Dapat melakukan login sebagai pencari kerja.</li> <li>c. Dapat melakukan Ubah data pribadi (harus melakukan login sebagai pencari kerja).</li> <li>d. Dapat melihat lowongan</li> </ul>

	<ul style="list-style-type: none"> <li>pekerjaan yang sesuai dengan kriteria yang ia punya (harus melakukan login sebagai pencari kerja).</li> <li>e. Dapat melakukan pencarian lowongan sesuai dengan kalimat masukan pencari kerja.</li> <li>f.</li> </ul>
Perusahaan	<ul style="list-style-type: none"> <li>a. Adalah Badan Usaha yang mendaftar sebagai member perusahaan.</li> <li>b. Dapat melakukan login sebagai perusahaan.</li> <li>c. Dapat melihat lowongan pekerjaan.</li> <li>d. Dapat melihat daftar pencari kerja yang sesuai dengan klasifikasi lowongan yang di-upload perusahaan (harus melakukan login sebagai perusahaan).</li> <li>e. Dapat meng-upload lowongan pekerjaan (harus melakukan login sebagai perusahaan)</li> <li>f. Dapat menambah lowongan pekerjaan (harus melakukan login sebagai perusahaan)</li> <li>g. Dapat melakukan pencarian pencari kerja sesuai kalimat kunci masukan perusahaan.</li> </ul>

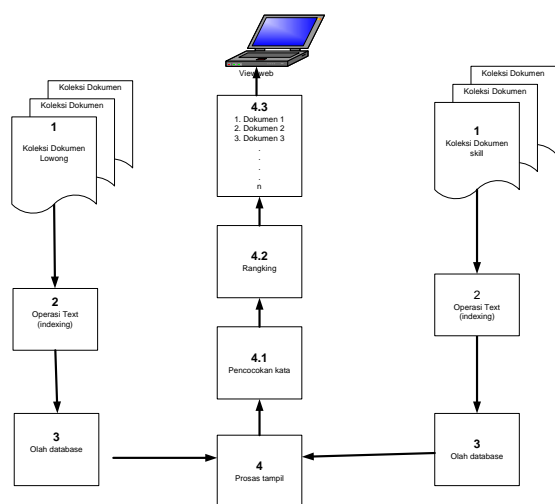
#### 3.2 Perancangan logis

Program ini akan menghasilkan sebuah sistem informasi berbasis website. Seorang pelamar maupun perusaha akan mendapatkan rekomendasi jika sudah mendaftarkan diri sebagai member pada website ini. Pelamar memasukkan CV dan skill yang dia punya sedangkan perusahaan memasukkan data perusahaan serta lowongan pekerjaan yang ia punya. Setelah sistem menjalankan proses pengolahan data maka sistem dapat memberikan rekomendasi lowongan kepada pelamar pekerjaan, dan untuk perusahaan akan mendapat rekomendasi pelamar. Gambar 3 merupakan rancangan database relational pada sistem ini



Gambar 3 Perancangan Logis

### 3.3 Proses kerja aplikasi



Urutan pada gambar yaitu proses 1, 2, 3, dan 4. Proses pembuatan rekomendasi ini dilakukan oleh sistem secara berkala. Proses rekomendasi tahap-tahapnya sebagai berikut :

#### 3.3.1 Pengumpulan dokumen

Proses pengumpulan dilakukan dilakukan saat pencari kerja mendaftar pada website dan saat perusahaan memasukkan lowongan pekerjaan pada website.

#### 3.3.2 Operasi teks

Operasi teks dilakukan pada kumpulan dokumen skill yang dimiliki oleh pencari kerja dan pada dokumen lowongan yang dimiliki perusahaan, ada tiga proses teks yang dilakukan

- Parsing**  
Parsing adalah proses pengambilan kata-kata pada sekumpulan dokumen.
- Stolist**  
Adalah proses penghilangan kata kata yang tidak berguna seperti : tetapi, dan, mengapa, apakah dan lain-lain. Kata kata yang tidak berguna ini tersimpan dalam sistem.
- Stemming**

Proses stemming adalah proses pengilangan imbuhan kata-kata menjadi sebuah kata dasar. Setelah proses stemming dilakukan hasilnya adalah term indeks dan disimpan dalam tabel indeks untuk dokumen skill dari pencari kerja dan untuk dokumen lowongan tersimpan pada tabel indeks\_per. Tabel tersebut berisi kata dan frekuensi kemunculan kata tersebut dalam sebuah dokumen skill pencari kerja. proses ini menggunakan algoritma Bobby Nazief dan Mirna Adriani.

#### 3.3.3 Olah database

Olah database adalah memberikan bobot pada sebuah kata (*term index*) berdasarkan rumus savoy(1993). Bobot adalah perkalian antara nilai *tif* dan *idf*. Lalu nilai bobot untuk kata-kata dokumen skill pencari kerja disimpan dalam sistem .

#### 3.3.4 Proses Tampil

Proses menampilkan dalam rekomendasi ada dua yaitu dari sisi pencari kerja dan perusahaan. Proses ini terdiri dua tahap

##### 3.3.4.1 Proses tampil untuk pencari kerja

- Pencocokan kata**  
Pencocokan kata adalah proses mencocokkan kata dari *term indeks* hasil dari proses skill pencari kerja dengan *term indeks* hasil dari proses pada dokumen lowongan perusahaan. Algoritma yang digunakan adalah algoritma TF-IDF.

- Perangkingan dokumen**

Proses perangkingannya menggunakan algoritma TF-IDF dengan menjumlahkan bobot dari kata (*term index*) dari masing dokumen lowongan sesuai dengan query kata skill dan mengurutkannya dari jumlah tertinggi ke rendah.

##### 3.3.4.2 Proses tampil untuk perusahaan

- Pencocokan kata**  
Pencocokan kata adalah proses mencocokkan kata dari *term indeks* hasil dari dokumen skill pencari kerja dengan *term indeks* hasil dari proses pada dokumen lowongan perusahaan. Algoritma yang digunakan adalah algoritma TF-IDF.
- Perangkingan dokumen**  
Proses perangkingannya menggunakan algoritma TF-IDF dengan menjumlahkan

bobot dari kata (*term index*) dari masing dokumen skill sesuai dengan query kata lowongan yang dimiliki perusahaan kemudian mengurutkannya dari jumlah tertinggi ke rendah .

#### 4. PENGUJIAN

Pengujian Hasil rekomendasi

##### 4.1 Persiapan

- Mencari contoh dokumen lowongan dan cv melalui internet. melakukan pencarian contoh dokumen lowongan pada website www.jobindo.com, www.bursakerjabali.com
- Aplikasi rekomendasi ini memiliki dokumen lowongan sebanyak 27 dokumen dan 30 dokumen CV pencari kerja Metode yang dilakukan dalam pengujian hasil temu informasi pada sistem ini dengan memberikan kuisisioner kepada responden.

##### 4.2 Pengolahan hasil kuisisioner

###### 4.2.1 Memilih metode untuk pengolahan kuisisioner.

Metode dalam pengolah hasil kuisisioner ini adalah *recall and precision*. Dari hasil pengolahan tersebut pembuat sistem dapat mengukur hasil informasi rekomendasi

###### 4.2.2 Hal-hal yang diuji

- Jumlah dokumen yang direkomendasikan
- Jumlah dokumen yang baik kesesuaiannya
- Jumlah dokumen yang tidak sesuai kesesuaiannya

###### 4.2.3 Membuat kuisisioner.

###### 4.2.4 Mencari responden.

Karakteristik dari responden yang pilih untuk mengisikan kuisisioner ini adalah mahasiswa, alumni dan karyawan. Jumlah responden yang menguji sistem ini adalah 28 responden yang terdiri 15 responden untuk sebagai perusahaan dan 13 responden sebagai pencari kerja. Dibawah ini merupakan tabel hasil pengolahan kuisisioner.

Tabel 2 Hasil Pengolahan Kuisisioner

No	Relevan	Not relevan	Jumlah dokumen	Precision
1	9	16	25	0,36
2	8	3	11	0.727273
3	5	22	27	0.185185
4	6	23	29	0.206897
5	7	18	25	0.28
6	3	27	30	0.1
7	7	22	29	0.241379
8	17	7	24	0.708333

9	9	16	25	0.36
10	5	25	30	0.166667
11	3	0	3	1
12	13	14	27	0.481481
13	8	11	19	0.421053
14	3	0	3	1
15	6	13	19	0.421053
16	4	10	14	0.285714
17	4	18	22	0.181818
18	2	16	18	0.111111
19	5	6	11	0.454545
20	2	10	12	0.166667
21	1	23	24	0.041667
22	5	18	23	0.217391
23	4	19	23	0.173913
24	6	16	22	0.272727
25	4	19	23	0.173913
26	2	22	24	0.083333
27	3	21	24	0.125
28	4	19	23	0.173913
Jumlah				8.761033
Rata-rata				0.312894

Dibawah ini merupakan tabel 5 perbandingan antara query kalimat yang baik dan tidak baik

Tabel 2 Perbandingan

No	Query kalimat	Relevan	Not relevan	Jml dok	Precision
1	programer java, PHP, makromedia flash, Linux.	3	0	3	1(baik)
2	dibutuhkan Administrasi Service Center , pendidikan d1, d2, d3, s1, Mempunyai pengalaman sebagai administrasi / customer service Dapat mengoperasikan komputer, excel dan office.	3	27	30	0.1( tidak baik)

##### 4.2.5 Menarik kesimpulan dari analisa kuisisioner.

- Hasil recall diperoleh dari rumus :

$$\text{Recall} = \frac{\text{Jumlah dokumen relevan Yang Terambil}}{\text{jumlah dokumen relevan yang ada dalam database (file)}}$$

Karena semua dokumen relevan dalam database terambil baik dari dokumen lowongan pekerjaan maupun dokumen kemampuan pekerja maka hasil recall dari sistem rekomendasi ini adalah 1.

- b. Hasil precision diperoleh dari rumus :

$$\text{Precision} = \frac{\text{Jumlah Dokumen relevan yang Terambil}}{\text{jumlah dokumen yang terambil dalam pencarian}}$$

- Berdasarkan analisa hasil perhitungan nilai rata-rata *precision* adalah 0.312894
  - Untuk nilai rasio rekomendasi antara recall dan *precision* berdasarkan kuisioner 1: 0.312894
  - Hasil kedua rasio recall dan *precision* diatas kurang baik karena hasil yang baik adalah perbandingan antara *recall* dan *precision* 1:1 (Lee Pao, 1989 : 229). Selain itu, suatu sistem temu kembali dinyatakan efektif apabila hasil penelusuran mampu menunjukkan ketepatan (*precision*) yang tinggi sekalipun perolehannya rendah (Rowley, 1992 : 172).
- c. Sistem mampu memberikan *precision* informasi yang baik apabila query kata spesifik contoh itu dapat dilihat pada tabel
- d. Sistem memberikan nilai *precision* tidak baik apabila query kata masukan tidak spesifik artinya sebuah query yang mencantumkan kata-kata yang mempunyai idf rendah. Tingkat *precision* rekomendasi tidak baik dapat dilihat dari tabel 2. Nilai *precision* tidak baik dipengaruhi oleh nilai Idf yang rendah .

## 5. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan yang diambil setelah melakukan implemmentasi aplikasi web temu kembali lowongan kerja :

- a. Pencari kerja mendapatkan rekomendasi lowongan dengan mudah sesuai dengan kemampuannya. Dan perusahaan mendapatkan rekomendasi pencari kerja sesuai dengan lowongan yang perusahaan punya.
- b. Kemampuan sistem menemukan rekomendasi didukung oleh tingkat *recall* sistem yang bernilai satu artinya semua dokumen yang relevan ditemukan oleh sistem.
- c. Hasil nilai *precision* rata-rata yang dikeluarkan sistem berdasarkan dokumen lowongan adalah 42% dan untuk dokumen kemampuan pekerja adalah 19%.

- d. Semakin spesifik query maka rekomendasi yang dihasilkan semakin baik . hal ini ditunjukkan oleh tingginya nilai *precision*.
- e. Tingkat spesifik sebuah query term diukur berdasarkan nilai idf dari term query. Semakin tinggi nilai idf sebuah term maka semakin spesifik term tersebut.
- f. Nilai idf berbanding lurus dengan nilai TF-IDF.

### 5.2 Dari hasil analisa saran yang dapat diberikan :

- a. Penambahan kamus kata dasar dapat memperbaiki hasil keluaran rekomendasi dari sistem
- b. Penambahan status pada dokumen lowongan dan pekerja dapat mengurangi proses pemerolehan rekomendasi.

## PUSTAKA

- Fitriyanti, Masayu. *Sistem Temu-kembali Informasi dengan Mengimplementasikan Operasi Boolean, Sistem Peringkat, Perbaikan Query, dan Pemanfaatan Tesaurus*. Fakultas Ilmu Komputer, Universitas Indonesia. Skripsi. 1997.
- Herbert A. Simon. *Causality in Bayesian belief networks*. In *Proceedings of the Ninth Annual Conference on Uncertainty in Artificial Intelligence (UAI'93)* San Francisco, CA, 1993. Morgan Kaufmann Publishers, Inc
- Nazief Bobby dan Mirna Adriani (1996), *Confix-Stripping: Approach to Stemming Algorithm for Bahasa Indonesia*, Faculty of Computer Science University of Indonesia.
- Pao, Miranda Lee. *Concepts of Information Retrieval*. Libraries Unlimited, Englewood, Colorado, 1989
- Savoy, J. "A Learning Scheme for Information Retrieval in Hypertext". *Information Processing & Management*, 30(4), 515-533. 1993.
- Whitten, Jeffrey L.; Bentley, Lonnie D.; Dittman, Kevin C., 2005, *Metode Desain & Analisis Sistem, edisi 6*, Penerbit Andi Offset, Yogyakarta
- Zainal A. Hasibuan, Penerapan Berbagai Teknik Sistem Temu-Kembali Informasi Berbasis Hiperteks, Fakultas Ilmu Komputer, Universitas Indonesia.