IMPLEMENTASI METODE TERM FREQUENCY INVERSED DOCUMENT FREQUENCE (TF-IDF) DAN VECTOR SPACE MODEL PADA APLIKASI PEMBERKASAN SKRIPSI BERBASIS WEB

Abdul Rokhim [1)] , **Achmad ainul yaqin** [2)]
[1)] Program Studi/Prodi Teknik Informatika, STMIK Yadika,
[2)] Program Studi Teknologi informasi,Sekolah Tinggi Teknik Surabaya
email: abd.rokhim@gmail.com
yaqien378@mhs.stmik-yadika.ac.id

Abstract: *Thesis is a major requirement for taking the graduation at the College of Computer Science Yadika Bangil. Thesis data is stored in a database by using the application filing thesis report. This study discusses the implementation methods Term Frequency Inversed Document Frequency (TF-IDF) and the Vector Space Model on the application filing thesis report. The purpose of this study to optimize the search results in order to provide information that is more relevant to the needs of the user. With the TF-IDF method that gives weight to each - each data and vector space model calculates the value of the similarity between the keyword and each - each weight data. The result of the calculation is represented by the degree of similarity data against keyword.*

Keywords: *Term Frequency Inversed Document Frequency (TF-IDF), Vector Space Model, Text preprocessing, laporan skripsi.*

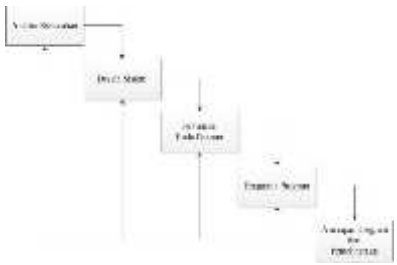
1. Pendahuluan

Skripsi merupakan syarat utama untuk menempuh kelulusan. Skripsi pada sekolah tinggi informatika menghasilkan sebuah program atau aplikasi dan sebuah laporan. Semakin tahun semakin banyak pula dokumen yang di simpan dalam bentuk dokumen fisik, maka di perlukan proses digitalisasi. Tujuannya digitalisasi adalah pengguna dapat melihat koleksi judul dan deskripsi dari skripsi dalam suatu database, akan mengalami kesulitan dalam pencarian judul skripsi yang begitu besar. Dibutuhkan banyak waktu untuk melakukan satu langkah pencarian data yang begitu besar untuk mendapatkan hasil pencarian yang relevan.

Diperlukan metode yang dapat mendukung pencarian dengan metode *term frequecny document inversed frequence* (TF-IDF) dan *Vector space model* (VSM). Dengan metode TF-IDF hasil pencarian akan dilakukan pembobotan kata yang di temukan dan di lanjutkan dengan metode VSM untuk melakukan perhitungan kemiripan hasil pencarian agar lebih relevan. Dengan demikian dapat dirumuskan masalah yang akan di bahas dalam penelitian ini yaitu bagaimana mengimplementasikan metode *Term Frequency Inversed Document Frequency* (TF-IDF) dan *Vector Space Model* untuk pencarian koleksi pada aplikasi pemberkasan skripsi yang berbasis website untuk mendapatkan hasil pencarian yang lebih relevan dengan kebutuhan user.

2. Metode Penelitian

Pada penelitian ini penulis menggunakan metode penelitian pengembangan sistem waterfall. metode *waterfall* adalah metode yang pekerjaan-pekerjaanya mengikuti suatu pola tertentu dan dilaksanakan dengan cara dari atas kebawah. Metode ini mempunyai tahapan seperti berikut. Analisis kebutuhan, Desain sistem, Penulisan kode program, Penerapan program dan pemeliharaan. Proses di lakukan secara berututan dari proses analisis hingga penerapan program. Gambar 1 menunjukkan proses metode waterfall.



Gambar 1. Metode *Waterfall.*

$$1 =$$

2.1. *Text Preprocessing*

Text prepocessing merupakan proses pengolahan teks menjadi kata dasar. Pada tahap Text preprocessing Dalam *text preprocessing* ada beberapa langkah yang perlu dilakukan untuk mendapatkan teks yang bebas derau (*noise*) atau bebas kata-kata yang tidak bermakna. Selain membebaskan dari derau, text preprocessing juga mengembalikan kata menjadi kata dasar atau root word.Langkah-langkah dalam Text preprocessing dalam bahasaIndonesia adalah : Proses *Tokenizing*, Proses *Filtering*, Proses *Stemming*. Tahap *tokenizing* adalah tahap pemotongan string input berdasarkan tiap kata yang menyusunnya. Tahap *filtering* yaitu tahap mengambil kata- kata penting dari hasil token dan menghapus kata – kata yang kurang penting. Tahap selanjutnya adalah tahap *stemming* adalah tahap mencari dasar kata dari tiap kata hasil filtering. Setiap kata yang memiliki imbuhan seperti imbuhan awalan dan akhiran maka akan diambil kata dasarnya. Hasil kata dasar tersebut akan digunakan dalam perhitungan TF-IDF.

2.2. Metode *Term Frequency Inversed*

Document Frequency (TF-IDF)

Metode TF-IDF (*Term Frequncy Inverse Document Frequency*) merupakan suatu cara untuk memberikan bobot hubungan suatu kata (*term*) terhadap dokumen. Metode ini Menggunakan dua konsep untuk perhitungan

Pada algoritma TF-IDF digunakan rumus untuk menghitung bobot (W) masing-masing dokumen terhadap kata kunci dengan rumus yaitu.

$W_{dt} = tf_{dt} * IDF_t$ .....(2-1)

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

IDF = Inversed Document Frequency

log = log (10 of)

D = total dokumen

df = banyak dokumen yang mengandung kata yang dicari

2.3. *Vector Space Model*

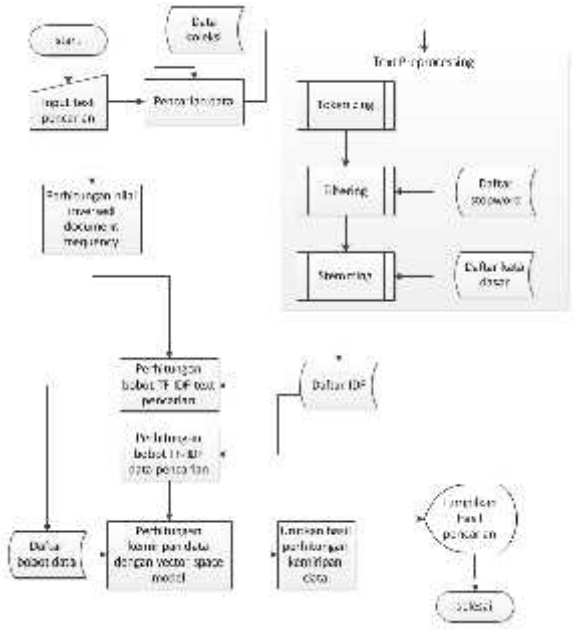
Vector space model adalah suatu model yang digunakan untuk mengukur kemiripan antara suatu dokumen dengan suatu query. Pada model ini, query dan dokumen dianggap sebagai vektor- vektor pada ruang n- dimensi, dimana n adalah jumlah dari seluruh term yang ada dalam leksikon. *Leksikon* adalah daftar semua *term* yang ada dalam indeks. Salah satu cara untuk mengatasi hal tersebut dalam *model vector space* adalah dengan cara melakukan perluasan vektor Proses perluasan dapat dilakukan pada vektor query, vektor dokumen, atau pada kedua vektor tersebut.

Pada metode *vector space model* gunakan rumus untuk mencari nilai cosinus sudut antara dua vector dari setiap bobot dokumen (WD) dan bobot dari kata kunci (WK). Rumus yang digunakan adalah sebagai berikut

$$sim(d_j, q) = \frac{d_j \cdot q}{|d_j| \cdot |q|} = \frac{\sum_{i=1}^{w} w_{ij} * w_{iq}}{\sqrt{\sum_{i=1}^{n} w^2_{ij}} * \sqrt{\sum_{i=1}^{w} w^2_{iq}}} \quad \ldots (2.2)$$

2.4. Flowchart

Untuk menjelaskan alur dan kinerja aplikasi yang penulis buat, dibutuhkan penjelasan mengenai proses berjalannya aplikasi yang telah dibuat. Salah satunya dengan menggunakan *Flowchart* , dengan menggunakan *flowchart* penulis dapat menjelaskan proses pencarian mulai dari tahap input hingga bagaimana sistem mencapai tahap output. Berikut flowchart dari aplikasi pemberkasan skripsi ini.

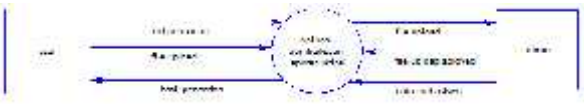


Gambar 2.1 Flowchart aplikasi pemberkasan skripsi.

Alur dimulai dari user melakukan input teks pencarian, kemudian dilakukan pencarian pada database dan hasil pencarian di olah dengan *text preprocessing* . Hasil pengolahan di lakukan perhitungan TF-IDF dan VSM . Hasil perhitungan di urutkan berdsarkan nilai kemiripan dari yang besar ke yang kecil dan di tampilkan pada user.

2.5. Diagram konteks

Diagram konteks merupakan diagram yang menggambarkan kondisi sistem yang ada baik input maupun output serta menyertakan terminator yang terlibat dalam penggunaan sistem. Diagram ini akan memberi gambaran tentang keseluruhan sistem seperti pada gambar berikut.

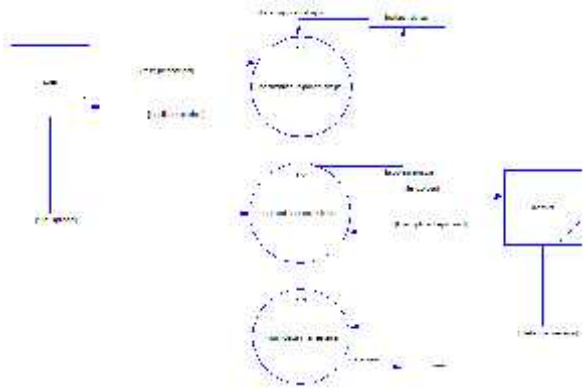


Gambar 2.2 Diagram konteks

Terdapat dua entitas yaiut User dan Admin. Admin melakukan input data mahasiswa dan verifikasi upload. User melakukan upload berkas laporan skripsi dan melakukan pencarian data pada sistem.

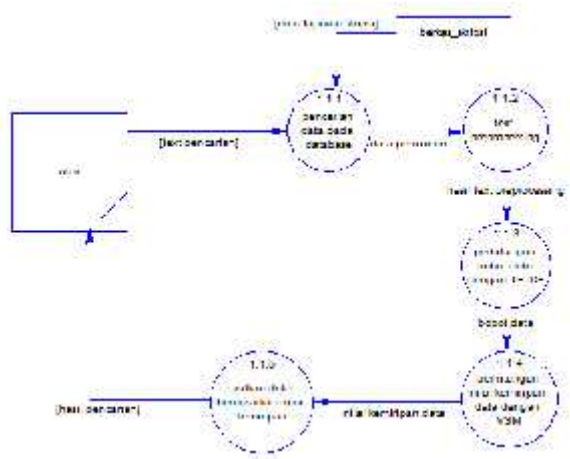
2.6. DFD (*Data Flow Diagram*)

Data Flow Diagram merupakan suatu cara atau metode dalam merancang sebuah sistem, biasanya digunakan untuk menggambar sistem dari proses-proses secara fungsional, yang dilakukan dengan merancang aliran data dan proses yang berhubungan antara satu dengan yang lainnya oleh aliran data. Berikut DFD Level 1 aplikasi pemberkasan skripsi.



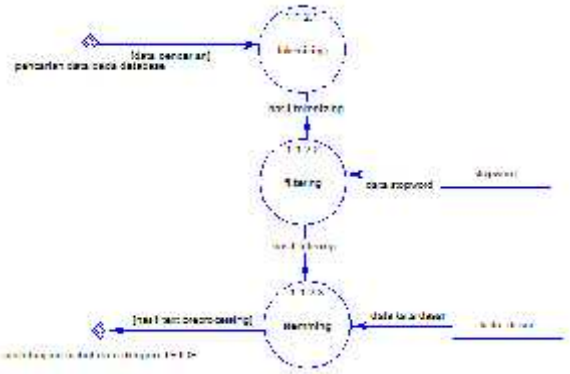
Gambar 2.3 DFD Level 1

Pada DFD level 1 ini terdapat 3 proses utama yaitu pencarian laporan skripsi dimana user melakuukan input teks pencarian kemudian di olah oleh sistem dan di tampilkan kepada user, yang kedua proses upload file dimana user melakukan upload file berkas laporan skripsi dan di verifikasi oleh admin, dan yang ketiga input data mahasiswa dimana admin menginputkan data mahasiswa untuk mendapatkan hak ases untuk melakukan upload file berkas laporan skripsi. Pada pencarian laporan skripsi terdapat sub proses yang akan diuraikan pada DFD Level 2 berikut.



Gambar 2.4 DFD Level 2

Pada DFD level 2 ini terdapat 4 proses utama yaitu pencarian data pada database dimana kata kunci akan di cari pada tabel berkas_skripsi. Kemudian hasil pencarian di olah dengan proses *text preprocessing* untuk mendapatkan index kata dasar, proses yang ketiga proses perhitungan TF-IDF yang menghasilkan bobot pada masing – masing data. Pada proses empat di hitung nilai kemiripan dari masing – masing data dengan kata kunci dan hasil perhitungan tersebut di urutkan pada proses kelima yaitu proses pengurutan data berdasarkan nilai kemiripan dan di tampilkan pada user. Pada proses *text preprocessing* terdapat sub proses yang akan di jelaskan pada DFD level 3 berikut.



Gambar 2.5 DFD Level 3

Pada DFD level 3 ini terdapat 3 proses utama yaitu proses *tokenizing* yang melakukan pemecahan teks menjadi per kata. Proses *filtering* yang menyaring kata – kata yang tidak penting untuk di hapus. Dan yang terakhir proses *stemming* dimana hasil filtering akan buang imbuhan baik awalan maupun akhiran menjadi kata dasar.

3. Hasil dan Pembahasan

Dari hasil penelitian didapatkan hasil pengujian sebagai berikut :

Tabel 3.1 Tabel perbandingan pencarian.

No	Test Case	*Query* biasa	TF-IDF dan *VSM*
1	Hasil pencarian	6 data	6 data
3	Urutan tampilan data	Hasil pencarian di urutkan berdasarkan alfabet secara ascending.	Hasil pencarian diurutkan berdasarkan nilai kemiripan data terhadap kata kunci. Data diurutkan dari nilai kemiripan yang besar k yang terkecil.
3	Kebutuhan user	User ingin mencari data yang sesuai dengan *keyword* ,Data hasil pencarian yang sesuai di temukan di tampilkan berdasarkan urutan alfabet. User masih harus memilah – milah kembali data yang benar – benar sesuai dengan kebutuhan.	Data hasil pencarian yang sesuai dengan *keyword* di tampilkan. Urutan menampilkan data dengan metode ini berdasarkan hasil perhitungan nilai kemiripan pada masing data kemudian di urutkan dari nilai kemiripan yang besar ke yang terkecil. Dengan cara ini data yang paling sesuai dengan kebutuhan user akan di tampilkan paling atas sendiri.

Sesuai dengan tabel 3.1 di atas , dapat di ketahui bahwa dengan menggunakan metode TF-IDF dan

vector space model mempunyai keunggulan data yang mempunyai kemiripan dengan kata kunci di tampilkan paling atas. Dengan cara ini user dapat dengan mudah menemukan data yang paling sesuai dengan kata kunci.

Hasil pengujian tersebut jika di implementasikan pada system aplikasi pemberkasan skripsi berbasis web, dapat di lihat pada gambar 3.1 yang menampilkan hasil pencarian yang mempunyai nilai kemiripan dengan kata kunci dan di tampilkan dengan urutan dari nilai kemiripan yang terbesar ke yang terkecil.

Gambar 3.1 Tampilan hasil pencarian

4. Simpulan

Berdasarkan hasil penelitian dan perancangan Aplikasi pemberkasan laporan skrispsi dengan metode TF-IDF dan VSM dapat di simpulkan sebagai berikut yaitu Aplikasi pemberkasan laporan skrispsi dengan mengimplementasikan metode TF-IDF dan VSM dapat menampilkan hasil pencarian sesuai dengan kebutuhan user. Dengan menberikan nilai kemiripan pada masing – masing data terhadap kata kunci dan di urutkan berdasarkan nilai terbesar hingga terkecil. User dapat dengan mudah mengetahui hasil pencarian yang sesuai dengan kebutuhan dengan cepat.

Daftar Pustaka

- Abdul kadir, 2013. *Pemrograman Database MySQL untuk Pemula :* MediaKom, Jakarta.
- Adhit Herwansyah, *Aplikasi Pengkategorian Dokumen Dan Pengukuran Tingkat Similaritas Dokumen Menggunakan Kata Kunci Pada Dokumen Penulisan Ilmiah Universitas Gunadarma*, Universitas Gunadarma.
- Giat Karyono, Fandy Setyo Utomo. *Temu Balik Informasi Pada Teks Berbahasa Indonesia Dengan Metode Vector Space Retrieval Model*, Seminar Nasional Teknologi Informasi & Komunikasi Terapan 2012, Juni 2012.
- Heru Suryono, *Aplikasi Menentukan Kemiripan Situs Web Pada Sistem Temu Balik Informasi Berbasis Web Menggunakan Metode Term Frequency Inverse Document Frequency*,Universitas Muhammadiyah Sidoarjo.
- https://id.wikipedia.org/wiki/ diakses pada tanggal 12 Juli 2016.
- Indrajani, S.Kom, MM. 2011. *Perancangan Basis Data Dalam Allin1* : Penerbit Elex Media Komputindo. Jakarta.
- Jogiyanto, HM, 1999. *Analisis dan Desain Sistem Informasi*. Yogyakarta : Penerbit Andi.
- Kristhoper David Harjono, *Perluasan Vector Pada Metode Search Vector Space*, INTEGRAL Vol. 10 No. 2, Juli 2005.
- Yudi Priyadi, 2014. *Kolaborasi SQL dan ERD dalam Implementasi Database :* Andi, Yogyakarta.