***TEXT MINING* DALAM PENCARIAN TOPIK SKRIPSI PADA *E-LIBRARY*  STMIK DENPASAR DENGAN METODE *VECTOR SPACE MODEL* DAN *CRONTAB***

I Komang Novayadi

(08.51.054)

Jurusan Teknik Informatika STMIK Denpasar

Jl. Tukad Balian 15, Renon 80266 – Telp. (0361) 249781 Fax : (0361) 223251

**Abstrak**

Saat ini bidang pendidikan khususnya bidang komputer mengalami perkembangan yang cukup pesat. Dengan bertambahnya jumlah lulusan di setiap tahunnya maka kumpulan skripsi di perpustakaan akan semakin bertambah pula jumlah dan jenisnya. Dengan banyaknya skripsi yang disimpan di perpustakaan maka mahasiswa akan mengalami kesulitan dalam mencari sekelompok skripsi yang sesuai dengan kata penting tertentu. Maka dari itu untuk membantu melakukan pencarian dan untuk mengurutkan dokumen berdasarkan kata yang dicari oleh mahasiswa tentunya dibutuhkan suatu aplikasi.

Dalam penelitian ini, penulis melakukan penelitian di perpustakaan STMIK Denpasar dengan menggunakan metode wawancara kepada petugas perpustakaan serta mencari data pendukung sebagai data tambahan. Dalam membangun sistem pencarian ini penulis menggunakan *text mining* sebagai ekstraksi teks dan metode *Vector space model* untuk mengurutkan dokumen berdasarkan tingkat kemiripan antara bobot dokumen dengan kata kunci. Selain metode *vector space model* penulis juga menggunakan beberapa metode pendukung, seperti Nazief dan Adriani, TF/IDF (*Term Frequency - Inverse Document Frequency*) serta pemanfaatan perintah *crontab* sebagai *tools* untuk *indexing* data.

Sistem pencarian ini dapat membantu melakukan perangkingan data berdasarkan bobot kata kunci terhadap kumpulan dokumen dalam perpustakaan online STMIK Denpasar.

Kata Kunci : *Text mining*, *Vector Space Model*, Nazief dan Adriani, TF/IDF, *Crontab.*

**Abstract**

Currently, the computer education in particular has developed rapidly. By the increasing number of graduates in each year, the amount and type of the thesis in the library collection will be increased. With many thesis is stored in the library, the student will have difficulty in finding a group of thesis that appropriate with certain important words. Therefore to help conduct a search and to sort documents based on words searched by the students of course required an application.

In this research, the author conducted a research in the library of STMIK Denpasar by using interviews to the librarian and find supporting data as additional data. In process to make the retrieval system, the author use text mining as a method of text extraction and Vector space model to rank documents based on the degree of similarity in integrity of the documents by a keyword. Besides the methods vector space models, the author also use several methods of support, such as Nazief and Adriani, TF / IDF (Term Frequency - Inverse Document Frequency) and utilize crontab command as a tools for indexing data.  
 This search system can help to do data ranking based on the integrity of keywords to documents in the online library collection of STMIK Denpasar.

Keywords : Text mining, Vector Space Model, Nazief and Adriani, TF/IDF, Crontab.

1. **Latar Belakang**
   1. **Pendahuluan**

Saat ini dunia pendidikan mengalami perkembangan yang cukup pesat khususnya bidang ilmu komputer. Sekolah tinggi seperti STMIK Denpasar dengan jumlah mahasiswa yang semakin bertambah, telah menghasilkan banyak lulusan di setiap tahunnya. Tentunya buku skripsi sebagai syarat kelulusan mahasiswa disimpan untuk selanjutnya digunakan sebagai referensi bagi pihak-pihak yang membutuhkan.

Dalam hal ini, perpustakaaan STMIK Denpasar adalah salah satu wadah yang menyediakan fasilitas di mana mahasiswa dapat meminjam skripsi. Kumpulan skripsi tersebut semakin bertambah jumlah dan jenisnya setiap tahun. Dengan banyaknya skripsi yang disimpan di perpustakaan maka mahasiswa akan mengalami kesulitan dalam mencari sekelompok skripsi yang sesuai dengan kata penting tertentu.

Maka dari itu untuk membantu melakukan pencarian dan untuk mengurutkan dokumen berdasarkan kata yang dicari oleh mahasiswa tentunya dibutuhkan suatu aplikasi. Aplikasi yang dimaksud adalah aplikasi pencarian dan pengurutan hasil pencarian berdasarkan kata kunci yang paling berkaitan dengan dokumen. Tujuan dari penelitian ini adalah untuk mengembangkan aplikasi yang menerapkan *text mining, indexing* dokumen dengan pemanfaatan perintah *Crontab* dan perangkingan dokumen dengan *Vector Space Model* pada sekumpulan skripsi yang disimpan dalam *directory* situs e-library STMIK Denpasarberdasarkan kata kunci tertentu .

* 1. **Rumusan masalah**

Berdasarkan latar belakang diatas maka permasalahan yang didapat adalah :

1. Bagaimana implementasi *text mining* dan algoritma *Vector Space Model* dalam pencarian topik skripsi di *e-library* STMIK Denpasar?
2. Bagaimana mengurutkan informasi judul skripsi hasil dari pencarian?
   1. **Batasan masalah**

Dari pokok masalah diatas maka dapat ditentukan batasan masalah sebagai berikut :

1. Data presprocessing dilakukan pada abstrak skripsi STMIK Denpasar.
2. Abstrak yang dijadikan *target data* merupakan abtrak berbahasa Indonesia dengan asumsi kata asing sebagai kata penting.
3. Metode *vector space model* digunakan hanya untuk melakukan perangkingan hasil pencarian dengan pendekatan *cosine similarity.*
   1. **Tujuan dan manfaat penelitian**

Tujuan yang ingin dicapai adalah terciptanya sebuah sistem pencarian topik skripsi dari mahasiswa di e-library STMIK Denpasar dengan memanfaatkan *text mining*, perintah *crontab* dan metode *vector space model* berdasarkan asosiasinya dengan *keyword* yang di-*input*-kan oleh *user*.

Penelitian yang penulis lakukan diharapkan mempunyai beberapa manfaat, antara lain:

1. Bagi Penulis, Penelitian ini dapat mengaplikasikan informasi mengenai ilmu yang penulis peroleh di bangku kuliah ke dunia nyata.
2. Bagi Masyarakat, Penelitian ini dapat mempermudah pencarian data dalam perpusatakaan bedasarkan peringkatan dokumen yang relevan.

Bagi STMIK Denpasar, menambah daftar bacaan di perpustakaan kampus dan ilmu bagi mahasiswa lainnya.

1. **Landasan teori**
   1. **Perpustakaan online**

Perpustakaan Online atau [Perpustakaan Digital](http://pustaka-ebook.com) Perpustakaan Digital adalah sebuah sistem yang memiliki berbagai layanan dan obyek informasi yang mendukung akses obyek informasi tesebut melalui perangkat digital (Sismanto, 2008).

* 1. **Text mining**

Text mining adalah salah satu bidang khusus dari *data mining*. Text mining dapat didefinisikan sebagai suatu proses menggali informasi dimana seorang user berinteraksi dengan sekumpulan dokumen menggunakan *tools* analisis yang merupakan komponen-komponen dalam *data mining* yang salah satunya adalah kategorisasi dan pengelompokan teks (Feldman*,*2006). Tujuan dari *text mining* adalah untuk mendapatkan informasi yang berguna dari sekumpulan dokumen (Feldman*,*2006). Jadi, sumber data yang digunakan pada *text mining* adalah kumpulan teks yang memiliki format yang tidak terstruktur atau minimal semi terstruktur. Adapun tugas khusus dari *text mining* antara lain yaitu pengkategorisasian teks *(text categorization)* dan pengelompokan teks *(text clustering)*.

* 1. **Ekstraksi dokumen**

Pada umumnya teks yang akan dilakukan proses memiliki struktur yang tidak baik. Cara yang digunakan untuk mempelajari suatu data teks, adalah dengan menentukan fitur-fitur yang mewakili setiap kata untuk setiap fitur yang ada pada dokumen. Sebelum menentukan fitur-fitur tersebut diperlukan tahap *pre-processing* yang dilakukan dalam text mining pada dokumen yaitu *case folding, tokenizing, filtering, stemming* dan *analyzing*.

1. ***Case folding***

*Case folding* adalah mengubah semua huruf dalam dokumen menjadi huruf kecil. Hanya huruf ‘a’ sampai dengan ‘z’ dan angka ’0’ sampai ’9’ yang diterima selain itu dianggap delimiter.

1. ***Tokenizing***

*Tokenizing* adalah tahap pemotongan string input berdasarkan tiap kata yang menyusunnya.

1. **Filtering**

Filtering adalah tahap mengambil kata - kata penting dari hasil token. Pada proses *filtering* ini kata-kata yang tidak penting akan dibuang. Kumpulan kata-kata tidak penting itu disebut *stoplist.* Contoh *stoplist* untuk bahasa Indonesia diantaranya “yang”, “di”, “ke”. *Stoplist* dapat bersifat fleksibel berdasarkan kebutuhan namun tidak menyimpang dari Kamus Besar Bahasa Indonesia dimana kata tersebut tidak mengandung makna atau tidak deskriptif yang dapat dibuang dalam pendekatan bag-of-words (Wibisono *et al*,2006).

1. ***Stemming***

*Stemming* merupakan suatu proses untuk menemukan kata dasar dari sebuah kata. Dengan menghilangkan semua imbuhan (*affixes)* baik yang terdiri dari awalan (*prefixes*), sisipan (*infixes*), akhiran (*suffixes*) dan *confixes* (kombinasi dari awalan dan akhiran) pada kata turunan. *Stemming* digunakan untuk mengganti bentuk dari suatu kata berimbuhan menjadi kata dasar dari kata tersebut yang sesuai dengan struktur morfologi Bahasa Indonesia yang baik dan benar. Banyak penelitian yang dilakukan untuk menemukan algoritma *stemming* yang tepat dan bagus dalam Bahasa Indonesia, antara lain algoritma Nazief dan Adriani, algoritma Arifin & Setiono, dan algoritma Vega. Menurut penelitian Jelita Asian, berdasarkan aturan morfologi Bahasa Indonesia dapat dinyatakan bahwa algoritma Nazief dan Adriani adalah algoritma yang memiliki hasil terbaik. (Novanta.2009).

**Algoritma Nazief dan Adriani**

Algoritma Nazief dan Adriani ini dikembangkan berdasarkan pada aturan morfologi Bahasa Indonesia yang mengelompokkan dan mengenkapsulasi imbuhan-imbuhan, termasuk di dalamnya adalah awalan (*prefix*), sisipan (*infix*), akhiran (*suffix*) dan gabungan awalan-akhiran (*confixes*).

1. ***Analyzing***

Dalam tahap analisa merupakan tahap untuk menentukan seberapa jauh keterhubungan antara kata-kata pada dokumen dengan kata kunci. Dalam tahap ini dilakukan proses pembobotan (W) tiap dokumen terhadap kata kunci.

Untuk mendapatkan nilai W dari tiap dokumen dan kata kunci digunakan algoritma *TF/IDF* (*Term Frequency – Inversed Document Frequency*) . *TF (Term Frequency)* merupakan frekuensi untuk menghitung bobot dari kata ke- *I* pada dokumen dan *IDF* (*Inversed Document Frequency*).

* 1. **Pembobotan TF/IDF (Term Frequency - Inverse Document Frequency)**

Pengurutan atau page rank dapat dilakukan dengan proses sorting berdasarkan W dari tiap dokumen tetapi apabila terdapat nilai W yang sama ambigu untuk membadingkan dokumen tersebut, untuk mengatasi ini, digunakan *Vector Space Model*. Dalam penelitian ini *TF/IDF* digunakan penulis untuk menghitung bobot dari dokumen dan kata kunci untuk selanjutnya digunakan dalam proses perhitungan *Cosine Similarity*.

* 1. **Vector Space Model**

*Vector space model* adalah suatu model yang digunakan untuk mengukur kemiripan antara suatu dokumen dengan suatu *query* (Harjono,2005). 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.

Ide dari metode ini adalah dengan menghitung kemiripan dengan menggunakan *cosine similarity* atau menghitung nilai *cosinus* sudut kemiripan dari dua vektor, yaitu W dari tiap dokumen dan W dari kata kunci.

Rumus yang digunakan adalah sebagai berikut (Halrian, 2006:14):

Co sin*q) = =………………(1)*

* 1. **Crontab**

*Crontab* adalah sebuah perintah yang mengendalikan *Cron* untuk menjalankan tugas-tugas yang terjadwal, sehingga akan mengurangi waktu administrasi. Cron adalah Daemon yang menjalankan perintah sesuai jadwal yang telah disusun (Anonim,2010).

Untuk menjalankan perintah-perintah *crontab* pada server linux dapat dilakukan dengan menggunakan sebuah aplikasi *cron job*. Dimana *cron job* merupakan aplikasi yang yang sudah terkonfigurasi langsung dengan *cPanel* pada hosting *berbasis* *linux* yang umumnya digunakan untuk penjadwalan *crontab*.

1. **Implementasi Hasil**
   1. **Analisis Proses**

**Contoh perhitungan Pencarian dan pengurutan data**

Dalam contoh kasus ini penulis akan menjabarkan proses demi proses terhadap 4 buah dokumen yang diambil dari potongan-potongan abstrak. Abstrak ini sengaja penulis potong-potong untuk mempermudah perhitungan.

**Contoh dokumen berasal dari Skripsi berikut :**

Table 3.1 Contoh dokumen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Dokumen ke** | **Judul** | **Penulis** | **Tahun** | **Abstrak** |
| Dokumen 1 | Sistem Pakar Mendiagnosis Penyakit Diabetes Mellitus Menggunakan Metode Forward Chaining Dan Backward Chaining | Kadek Dana | 2011 | Dalam penelitian ini, penulis menggunakan metode wawancara dengan pakar serta mencari data pendukung sebagai bahan tambahan data informasi. Metode penelusuran solusi yang di gunakan dalam membangun sistem pakar ini adalah penelurusuran runut maju (forward chainning) dan penelurusuran runut balik (backward chainning) serta pemodelan aturan memanfaatkan database Microsoft Access serta bahasa memprograman Embarcadero Delphi 2010. |
| Dokumen 2 | Sistem Pendukung Keputusan Untuk Menentukan Calon Penerima Beasiswa dengan Metode SAW dan FMADM | Made Agus Sadnyana | 2011 | Dalam penelitian ini, penulis menggunakan metode FMADM dan metode SAW untuk menentukan calon penerima beasiswa agar penentuan calon penerima beasiswa ini dapat lebih akurat, serta pemodelan aturan memanfaatkan database Microsoft Access serta bahasa pemrograman Embarcadero Delphi 2010 |
| Dokumen 3 | Implementasi Game Interaktif Sebagai Media Pembelajaran Mengenal Buah Dan Hewan Menggunakan Metode Scoring | A.A Ayu Dewi Utari | 2011 | Dewasa ini game berkembang begaitu pesatnya. Tetapi masih sangat jarang ada game yang ditujukan untuk meninggkatkan kemampuan anak dalam menerima pemebelajaran. Anak-anak cenderung ingin bermain daripada belajar. Hal ini sangatlah wajar, mengingat dunia anak adalah dunia bermain. Tetapi bagaimana kita dapat menyisipkan pembelajaran dalam permainan anak. Unutk itu, penulis mencoba memecahkan permasalahan tersebut dengan malakukan penelitian berjudul: “implementasi game interaktif sebagai media pembelajaran mengenal buah dan hewan menggunakan metode scoring”. |
| Dokumen4 | Dokumen 4 | Anonim | - | testing testing |

* 1. **Simulasi Penerapan text mining dalam proses preprocessing dokumen Tahap *Preprocessing***
     + 1. **Proses *preprocessing***

1. *Case folding*

Dalam tahap ini, dilakukan proses penerimaan karakter a-z dan angka 0-9 selain karakter dan angka tersebut akan dibuang dan dianggap sebagai delimeter. Hasil proses *case folding* untuk dokumen-dokumen di atas adalah sebagai berikut :

Tabel 3.2 hasil *case folding*

|  |  |  |
| --- | --- | --- |
| **Dokumen ke** | **Abstrak asli** | **Hasil *Case folding*** |
| Dokumen 1 | Dalam penelitian ini, penulis menggunakan metode wawancara dengan pakar serta mencari data pendukung sebagai bahan tambahan data informasi. Metode penelusuran solusi yang di gunakan dalam membangun sistem pakar ini adalah penelurusuran runut maju (forward chainning) dan penelurusuran runut balik (backward chainning) serta pemodelan aturan memanfaatkan database Microsoft Access serta bahasa memprograman Embarcadero Delphi 2010. | dalam penelitian ini penulis menggunakan metode wawancara dengan pakar serta mencari data pendukung sebagai bahan tambahan data informasivmetode penelusuran solusi yang di gunakan dalam membangun sistem pakar ini adalah penelurusuran runut maju forward chainning dan penelurusuran runut balik backward chainning serta pemodelan aturan memanfaatkan database microsoft access serta bahasa memprograman embarcadero delphi 2010 |
| Dokumen 2 | Dalam penelitian ini, penulis menggunakan metode FMADM dan metode SAW untuk menentukan calon penerima beasiswa agar penentuan calon penerima beasiswa ini dapat lebih akurat, serta pemodelan aturan memanfaatkan database Microsoft Access serta bahasa pemrograman Embarcadero Delphi 2010 | dalam penelitian ini, penulis menggunakan metode fmadm dan metode saw untuk menentukan calon penerima beasiswa agar penentuan calon penerima beasiswa ini dapat lebih akurat serta pemodelan aturan memanfaatkan database microsoft access serta bahasa pemrograman embarcadero delphi 2010 |
| Dokumen 3 | Dewasa ini game berkembang begaitu pesatnya. Tetapi masih sangat jarang ada game yang ditujukan untuk meninggkatkan kemampuan anak dalam menerima pemebelajaran. Anak-anak cenderung ingin bermain daripada belajar. Hal ini sangatlah wajar, mengingat dunia anak adalah dunia bermain. Tetapi bagaimana kita dapat menyisipkan pembelajaran dalam permainan anak. Unutk itu, penulis mencoba memecahkan permasalahan tersebut dengan malakukan penelitian berjudul: “implementasi game interaktif sebagai media pembelajaran mengenal buah dan hewan menggunakan metode scoring”. | dewasa ini game berkembang begaitu pesatnya tetapi masih sangat jarang ada game yang ditujukan untuk meninggkatkan kemampuan anak dalam menerima pemebelajaran anak anak cenderung ingin bermain daripada belajar hal ini sangatlah wajar mengingat dunia anak adalah dunia bermain tetapi bagaimana kita dapat menyisipkan pembelajaran dalam permainan anak. unutk itu penulis mencoba memecahkan permasalahan tersebut dengan malakukan penelitian berjudul implementasi game interaktif sebagai media pembelajaran mengenal buah dan hewan menggunakan metode scoring |
| Dokumen 4 | Testing testing | testing testing |

1. *Tokenizing*

Setelah dokumen melewati proses *case folding* maka dokumen tersebut akan di-*parsing* berdasarkan spasi agar terbentuk kumpulan kata.

1. *Filtering*

Pada proses *filtering* ini kata-kata yang tidak penting akan dibuang. Kumpulan kata-kata tidak penting itu disebut *stoplist.* Contoh *stoplist* untuk bahasa Indonesia diantaranya “yang”, “di”, “ke”. *Stoplist* dapat bersifat fleksibel berdasarkan kebutuhan namun tidak menyimpang dari Kamus Besar Bahasa Indonesia dimana kata tersebut tidak mengandung makna atau tidak deskriptif (Wibisono *et al*,2006).

Pada contoh kasus ini penulis menghilangkan kata-kata yang dianggap sebagai stoplist berdasarkan daftar kata-kata yang tidak penting pada [penelitian](http://fpmipa.upi.edu/staff/yudi/stop_words_list.txt) sebelumnya maka akan mendapat hasil seperti berikut, sesuai lampiran *stoplist*.

Tabel 3.3 Tabel *Filtering* Dokumen 1:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **no** | **kata** | **no** | **kata** | **no** | **kata** |
| 1 | penelitian | 14 | penelusuran | 27 | backward |
| 2 | penulis | 15 | solusi | 28 | chaining |
| 3 | metode | 16 | membangun | 29 | pemodelan |
| 4 | wawancara | 17 | sistem | 30 | aturan |
| 5 | pakar | 18 | pakar | 31 | memanfaatkan |
| 6 | mencari | 19 | penelusuran | 32 | database |
| 7 | data | 20 | runut | 33 | microsoft |
| 8 | pendukung | 21 | maju | 34 | access |
| 9 | bahan | 22 | forward | 35 | bahasa |
| 10 | tambahan | 23 | chainning | 36 | memprograman |
| 11 | data | 24 | penelusuran | 37 | embarcadero |
| 12 | informasi | 25 | runut | 38 | delphi |
| 13 | metode | 26 | balik | 39 | 2010 |

Tabel 3.4 Tabel *Filtering* Dokumen 2:

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Kata** | **No** | **Kata** |
| 1 | *penelitian* | 13 | akurat |
| 2 | penulis | 14 | pemodelan |
| 3 | metode | 15 | aturan |
| 4 | fmadm | 16 | memanfaatkan |
| 5 | metode | 17 | database |
| 6 | saw | 18 | microsoft |
| 7 | calon | 19 | access |
| 8 | penerima | 20 | bahasa |
| 9 | beasiswa | 21 | pemprograman |
| 10 | calon | 22 | embarcadero |
| 11 | penerima | 23 | delphi |
| 12 | beasiswa | 24 | 2010 |

Tabel 3.5 Tabel *Filtering* Dokumen 3:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Kata** | **No** | **Kata** | **No** | **Kata** |
| 1 | dewasa | 16 | mengingat | 31 | implementasi |
| 2 | game | 17 | anak | 32 | game |
| 3 | berkembang | 18 | bermain | 33 | interaktif |
| 4 | pesat | 19 | kita | 34 | media |
| 5 | game | 20 | menyisipkan | 35 | pembelajaran |
| 6 | ditujukan | 21 | pembelajaran | 36 | mengenal |
| 7 | meningkatkan | 22 | permainan | 37 | buah |
| 8 | kemampuan | 23 | anak | 38 | hewan |
| 9 | anak | 24 | penulis | 39 | metode |
| 10 | menerima | 25 | mencoba | 40 | scoring |
| 11 | pembelajaran | 26 | memecahkan |  |  |
| 12 | anak | 27 | tersebut |  |  |
| 13 | bermain | 28 | melakukan |  |  |
| 14 | belajar | 29 | penelitian |  |  |
| 15 | wajar | 30 | berjudul |  |  |

1. *Stemming*

Pada proses *stemming* ini tiap kata akan dihilangkan awalan dan akhiran sehingga membentuk kata dasar dengan pendekatan Nazief dan Adriani. Untuk penelitian ini penulis menggunakan kamus kata dasar dari [http://bahtera.org](http://bahtera.org/). Hasil dari proses stemming ini dapat dilihat pada Tabel berikut :

Tabel 3.6. Tabel *Stemming* Dokumen 1:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Kata** | **No** | **Kata** | **No** | **Kata** |
| 1 | teliti | 14 | telusur | 27 | backward |
| 2 | tulis | 15 | solusi | 28 | chainning |
| 3 | metote | 16 | bangun | 29 | model |
| 4 | wawancara | 17 | sistem | 30 | atur |
| 5 | pakar | 18 | pakar | 31 | manfaat |
| 6 | cari | 19 | telusur | 32 | database |
| 7 | data | 20 | runut | 33 | microsoft |
| 8 | dukung | 21 | maju | 34 | access |
| 9 | bahan | 22 | forward | 35 | bahasa |
| 10 | tambahan | 23 | chainning | 36 | program |
| 11 | data | 24 | telusur | 37 | embarcadero |
| 12 | informasi | 25 | runut | 38 | delphi |
| 13 | metode | 26 | balik | 39 | 2010 |

Tabel 3.7 Tabel *Stemming* Dokumen 2:

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Kata** | **No** | **Kata** |
| 1 | teliti | 13 | akurat |
| 2 | tulis | 14 | model |
| 3 | metode | 15 | atur |
| 4 | fmadm | 16 | manfaat |
| 5 | metode | 17 | database |
| 6 | saw | 18 | microsoft |
| 7 | calon | 19 | access |
| 8 | terima | 20 | bahasa |
| 9 | beasiswa | 21 | program |
| 10 | calon | 22 | embarcadero |
| 11 | terima | 23 | delphi |
| 12 | beasiswa | 24 | 2010 |

Tabel 3.8 Tabel *Stemming* Dokumen 3:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Kata** | **No** | **Kata** | **No** | **Kata** |
| 1 | dewasa | 16 | ingat | 31 | implementasi |
| 2 | game | 17 | anak | 32 | game |
| 3 | kembang | 18 | main | 33 | interaktif |
| 4 | pesat | 19 | kita | 34 | media |
| 5 | game | 20 | sisip | 35 | ajar |
| 6 | tuju | 21 | ajar | 36 | kenal |
| 7 | tingkat | 22 | main | 37 | buah |
| 8 | mampu | 23 | anak | 38 | hewan |
| 9 | anak | 24 | tulis | 39 | metode |
| Tabel 3.8 Lanjutan | | | | | |
| **No** | **Kata** | **No** | **Kata** | **No** | **Kata** |
| 10 | terima | 25 | coba | 40 | scoring |
| 11 | ajar | 26 | pecah |  |  |
| 12 | anak | 27 | sebut |  |  |
| 13 | main | 28 | laku |  |  |
| 14 | ajar | 29 | teliti |  |  |
| 15 | wajar | 30 | judul |  |  |

1. **Proses *Crontab* dalam *indexing* dokumen**

Setelah melalui beberapa proses diatas maka setaiap dokumen disimpan dalam database dan masing-masing mempunyai file XML yang berisikan informasi tersetruktrur dari dokumen tersebut.

Dari file-file XML tersebut dihitung bobot setiap kata atau frase yang didapat dengan *term frequency*(tf), hasil dari perhitungan ini disimpan dalam file *cache*. Perhitungan ini dilakukan melalui aplikasi *cronjob* pada *cPanel hosting linux* yang dilakukan secara berkala, tujuan dilakukan proses ini adalah untuk meng-*indexing* dokumen-dokumen dan diharapkan mampu memperingan pencarian.

**4.3 Simulasi Penerapan metode *TF/IDF* dan *Vector Space Model* dalam proses pembobotan dokumen**

Pada tahap text mining di atas telah dihasilkan 3 buah data yang telah melewati preprocessing dan file *cache* hasil *crontab*.

Kata kunci : ”manfaat metode penelitian”

Kata kunci kemudian dimasukkan dalam proses *preprocessing,* sehingga menghasilkan kata kunci “***manfaat metode penelitian***”.

1. Algoritma TF/IDF

Formula yang digunakan untuk menghitung bobot(w) masing-masing dokumen terhadap kata kunci adalah rumus pada no. 1 dan 2:

Untuk memudahkan perhitungan berikut ini tabel perhitungan frekuensi kata:

Tabel 3.9 Tabel perhitungan TF/IDF

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Token** | **Tf** | | | | | **df** | **D/df** | **IDF** | **W** | | | | |
| **Kk** | **D1** | **D2** | **D3** | **D4** | **Kk** | **D1** | **D2** | **D3** | **D4** |
| testing |  |  |  |  | 2 | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.000 | 1.398 |
| **teliti** | **1** | **1** | **1** | **1** |  | **3.000** | **1.667** | **0.222** | **0.222** | **0.222** | **0.222** | **0.222** | **0.000** |
| tulis |  | 1 | 1 | 1 |  | 3.000 | 1.667 | 0.222 | 0.000 | 0.222 | 0.222 | 0.222 | 0.000 |
| **metode** | **1** | **2** | **2** | **1** |  | **3.000** | **1.667** | **0.222** | **0.222** | **0.444** | **0.444** | **0.222** | **0.000** |
| fmadm |  |  | 1 |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.699 | 0.000 | 0.000 |
| saw |  |  | 1 |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.699 | 0.000 | 0.000 |
| calon |  |  | 2 |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 1.398 | 0.000 | 0.000 |
| terima |  |  | 2 | 1 |  | 2.000 | 2.500 | 0.398 | 0.000 | 0.000 | 0.796 | 0.398 | 0.000 |
| beasiswa |  |  | 2 |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 1.398 | 0.000 | 0.000 |
| akurat |  |  | 1 |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.699 | 0.000 | 0.000 |
| model |  | 1 | 1 |  |  | 2.000 | 2.500 | 0.398 | 0.000 | 0.398 | 0.398 | 0.000 | 0.000 |
| atur |  | 1 | 1 |  |  | 2.000 | 2.500 | 0.398 | 0.000 | 0.398 | 0.398 | 0.000 | 0.000 |
| **manfaat** | **1** | **1** | **1** |  |  | **2.000** | **2.500** | **0.398** | **0.398** | **0.398** | **0.398** | **0.000** | **0.000** |
| database |  | 1 | 1 |  |  | 2.000 | 2.500 | 0.398 | 0.000 | 0.398 | 0.398 | 0.000 | 0.000 |
| microsoft |  | 1 | 1 |  |  | 2.000 | 2.500 | 0.398 | 0.000 | 0.398 | 0.398 | 0.000 | 0.000 |
| access |  | 1 | 1 |  |  | 2.000 | 2.500 | 0.398 | 0.000 | 0.398 | 0.398 | 0.000 | 0.000 |
| Tabel 3.9 Lanjutan | | | | | | | | | | | | | |
| **Token** | **Tf** | | | | | **df** | **D/df** | **IDF** | **W** | | | | |
| **Kk** | **D1** | **D2** | **D3** | **D4** | **Kk** | **D1** | **D2** | **D3** | **D4** |
| bahasa |  | 1 | 1 |  |  | 2.000 | 2.500 | 0.398 | 0.000 | 0.398 | 0.398 | 0.000 | 0.000 |
| program |  | 1 | 1 |  |  | 2.000 | 2.500 | 0.398 | 0.000 | 0.398 | 0.398 | 0.000 | 0.000 |
| embarcadero |  | 1 | 1 |  |  | 2.000 | 2.500 | 0.398 | 0.000 | 0.398 | 0.398 | 0.000 | 0.000 |
| delphi |  | 1 | 1 |  |  | 2.000 | 2.500 | 0.398 | 0.000 | 0.398 | 0.398 | 0.000 | 0.000 |
| 2010 |  | 1 | 1 |  |  | 2.000 | 2.500 | 0.398 | 0.000 | 0.398 | 0.398 | 0.000 | 0.000 |
| dewasa |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| game |  |  |  | 3 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 2.097 | 0.000 |
| kembang |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| pesat |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| tuju |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| tingkat |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| mampu |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| anak |  |  |  | 4 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 2.796 | 0.000 |
| ajar |  |  |  | 4 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 2.796 | 0.000 |
| main |  |  |  | 3 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 2.097 | 0.000 |
| wajar |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| ingat |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| kita |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| sisip |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| coba |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| pecah |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| sebut |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| laku |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| judul |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| implementasi |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| interaktif |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| media |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| kenal |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| buah |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| hewan |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| scoring |  |  |  | 1 |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.000 | 0.000 | 0.699 | 0.000 |
| wawancara |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |
| pakar |  | 2 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 1.398 | 0.000 | 0.000 | 0.000 |
| cari |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |
| data |  | 2 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 1.398 | 0.000 | 0.000 | 0.000 |
| dukung |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |
| bahan |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |
| tambah |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |
| informasi |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |
| telusur |  | 3 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 2.097 | 0.000 | 0.000 | 0.000 |
| solusi |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Tabel 3.9 Lanjutan | | | | | | | | | | | | | |
| **Token** | **TF** | | | | | **df** | **D/df** | **IDF** | **W** | | | | |
| **kk** | **D1** | **D2** | **D3** | **D4** | **kk** | **D1** | **D2** | **D3** | **D4** |
| bangun |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |
| sistem |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |
| runut |  | 2 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 1.398 | 0.000 | 0.000 | 0.000 |
| maju |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |
| forward |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |
| chainning |  | 2 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 1.398 | 0.000 | 0.000 | 0.000 |
| balik |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |
| backward |  | 1 |  |  |  | 1.000 | 5.000 | 0.699 | 0.000 | 0.699 | 0.000 | 0.000 | 0.000 |

Bobot (W) untuk Dokumen 1 : 0.222+0.444+ 0.398 = 1.063

Bobot (W) untuk Dokumen 2 : 0.222+0.444+0.398 = 1.063

Bobot (W) untuk Dokumen 3 : 0.222+0.222+0 = 0.444

Maka dengan hasil di atas dapat diurutkan peringkat dokumen dengan kata kunci ”manfaat metode penelitian” hanya mendapatkan rangking 2.

Dalam penelitian ini penulis tidak menghentikan perhitungan hanya sampai tahap ini tetapi dilanjutkan dengan proses *Vector space model* walaupun dengan perhitungan di atas sudah didapatkan urutan yang tidak kembar. Hal ini bertujuan untuk mendapatkan perangkingan yang maksimal berdasarkan bobot kata kunci terhadap bobot tiap dokumen.

1. *Vector Space Model* (*Cosine Similarity*)

Untuk mempermudah perhitungan, maka akan digambarkan dalam tabel perhitungan *vector space model* . Pada tahap ini dihitung *cosine similirity* tiap bobot dokumen dengan bobot kata kunci dimana bobot-bobot tersebut digambarkan sebagai vektor. Untuk menghitung *cosinus* sudut tiap dokumen terhadap kata kunci maka dicari token-token dari dokumen. Token-token diperoleh dari pencarian kata dalam dokumen setelah proses preprocessing. Token diambil satu kata jika mempunyai dua atau lebih kata yang sama pada kumpulan dokumen.

Kemudian setiap token-token akan dihitung bobotnya terhadap dokumen-dokumen yang ada dan terhadap kata kunci yang dimasukkan. Setelah mendapatkan bobot tiap token, dicari bobot kuadrat dari bobot dokumen dan perkalian bobot kata kunci terhadap bobot dokumen

Karena setengah dari proses *Vector Space model* telah dilakukan pada proses *TF/IDF* dengan menghasilkan bobot dari masing-masing kata dalam tiap dokumen maka proses selanjutnya adalah mengubah bobot tiap kata menjadi vektor untuk selanjutnya dihitung tingkat kemiripan kata kunci terhadap masing-masing dokumen.

Sesuai dengan rumus no.3 maka yang digunakan untuk menghitung tingkat cosine similarity suatu kata terhadap dokumen maka dihasilkan data seperti tabel 4.8.

Tabel 3.10 Tabel perhitungan *Vector Space Model*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Token** | **W2** | | | | | **KK\*Di** | | | |
| **kk** | **D1** | **D2** | **D3** | **D4** | **D1** | **D2** | **D3** | **D4** |
| testing | 0.000 | 0.000 | 0.000 | 0.000 | 1.95 | 0.000 | 0.000 | 0.000 | 0.000 |
| **teliti** | **0.049** | **0.049** | **0.049** | **0.049** | **0.00** | **0.049** | **0.049** | **0.049** | **0.000** |
| tulis | 0.000 | 0.049 | 0.049 | 0.049 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| **metode** | **0.049** | **0.197** | **0.197** | **0.049** | **0.00** | **0.098** | **0.098** | **0.049** | **0.000** |
| fmadm | 0.000 | 0.000 | 0.489 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| saw | 0.000 | 0.000 | 0.489 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
|  |  |  |  |  |  |  |  |  |  |
| Tabel 3.10 Lanjutan | | | | | | | | | |
| **Token** | **W2** | | | | | **KK\*Di** | | | |
| **kk** | **D1** | **D2** | **D3** | **D4** | **D1** | **D2** | **D3** | **D4** |
| calon | 0.000 | 0.000 | 1.954 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| terima | 0.000 | 0.000 | 0.633 | 0.158 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| beasiswa | 0.000 | 0.000 | 1.954 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| akurat | 0.000 | 0.000 | 0.489 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| model | 0.000 | 0.158 | 0.158 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| atur | 0.000 | 0.158 | 0.158 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| **manfaat** | **0.158** | **0.158** | **0.158** | **0.000** | **0.00** | **0.158** | **0.158** | **0.000** | **0.000** |
| database | 0.000 | 0.158 | 0.158 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| microsoft | 0.000 | 0.158 | 0.158 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| access | 0.000 | 0.158 | 0.158 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| bahasa | 0.000 | 0.158 | 0.158 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| program | 0.000 | 0.158 | 0.158 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| embarcadero | 0.000 | 0.158 | 0.158 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| delphi | 0.000 | 0.158 | 0.158 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| 2010 | 0.000 | 0.158 | 0.158 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| dewasa | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| game | 0.000 | 0.000 | 0.000 | 4.397 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| kembang | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| pesat | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| tuju | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| tingkat | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| mampu | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| anak | 0.000 | 0.000 | 0.000 | 7.817 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| ajar | 0.000 | 0.000 | 0.000 | 7.817 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| main | 0.000 | 0.000 | 0.000 | 4.397 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| wajar | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| ingat | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| kita | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| sisip | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| coba | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| pecah | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| sebut | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| laku | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| judul | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| implementasi | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| interaktif | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| media | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| kenal | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| buah | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| hewan | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| scoring | 0.000 | 0.000 | 0.000 | 0.489 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| wawancara | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| pakar | 0.000 | 1.954 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| Tabel 3.10 Lanjutan | | | | | | | | | |
| **Token** | **W2** | | | | | **KK\*Di** | | | |
| **kk** | **D1** | **D2** | **D3** | **D4** | **D1** | **D2** | **D3** | **D4** |
| cari | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| data | 0.000 | 1.954 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| dukung | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| bahan | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| tambah | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| informasi | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| telusur | 0.000 | 4.397 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| solusi | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| bangun | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| sistem | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| runut | 0.000 | 1.954 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| maju | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| forward | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| chainning | 0.000 | 1.954 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| balik | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| backward | 0.000 | 0.489 | 0.000 | 0.000 | 0.00 | 0.000 | 0.000 | 0.000 | 0.000 |
| **Jumlah** | **0.257** | **20.602** | **8.045** | **35.482** | **1.954** | **0.306** | **0.306** | **0.098** | **0.000** |

***Formula : Cosine (Di) = sum(kk . Di)/([sqrt(kk2)]\*[sqrt(Di2)])***

Perhitungan bobot tiap dokumen.

Cosine (D1) = 0.306/(sqrt(0.257)\* sqrt(20.602))

= 0.306/(0.507 \* 4.539)

= 0.306/2.3

= **0.1330**

Cosine (D2) = 0.306/(sqrt(0.257)\* sqrt(8.045))

= 0.306/(0.507 \* 2.836)

= 0.306/ 1.437

= **0.2129**

Cosine (D3) = 0.098/(sqrt(0.257)\* sqrt(35.482))

= 0.098/(/(0.507 \*5.957)

= 0.098/3.019

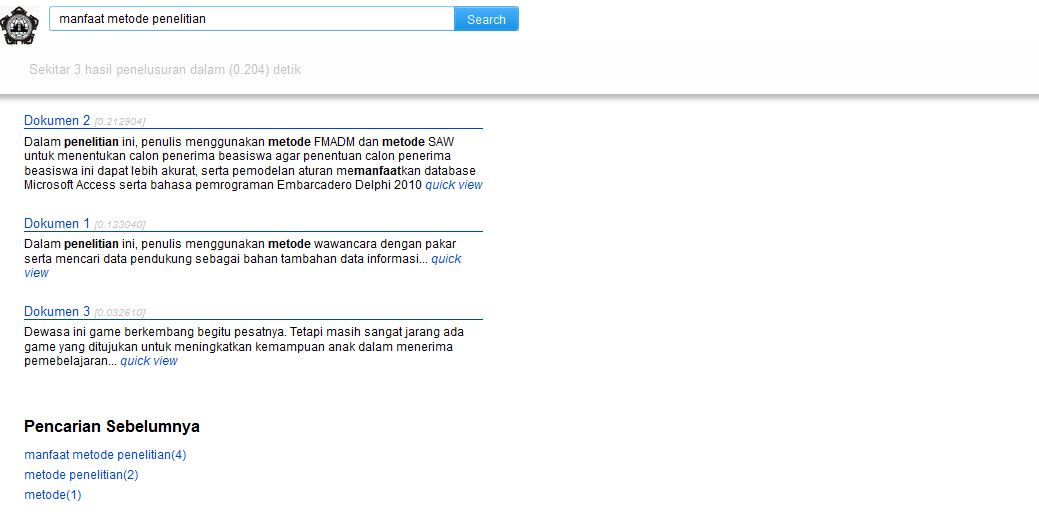
= **0.0326**

Setelah mendapatkan nilai *cosine* tiap dokumen, maka dokumen-dokumen akan diurutkan dari nilai sudut yang paling mendekati nol atau nilai yang mendekati 1, sehingga hasil dari perangkingan dokumen dapat dilihat pada tabel di bawah ini.

Tabel 3.11 Tabel Rangking Dokumen

|  |  |  |
| --- | --- | --- |
| **Rangking** | **Dokumen ke -** | **Bobot** |
| 1 | 2 | 0.2129 |
| 2 | 1 | 0.1330 |
| 3 | 3 | 0.0326 |

* 1. **Hasil dari proses Pencarian dan pengurutan dengan sistem adalah sebagai berikut**

****

Gambar 3.1 Form Hasil Pencarian dengan metode vector space model dan crontab

Dari hasil perhitungan sistem ini menghasilkan bobot dan perangkingan yang sama dengan perhitungan secara manual dimana menghasilkan bobot dan perangkingan yang sama dimana dokumen 1 mempunyai bobot 0.1330, dokumen 2 berbobot 0.2129 dan dokumen 3 berbobot 0.0326.

1. **Kesimpulan dan Saran**

**4.1 Kesimpulan**

Kesimpulan yang diperoleh dari hasil penelitian tentang *Text mining* dalam pencarian topik skripsi dengan metode *Vector Space Model* dan *crontab* yaitu:

* + - 1. Sistem pencarian dengan implementasi *text* *mining* dan metode *Vector* *space* *model* serta pemanfaatan *indexing* data dengan *crontab* dapat membatu melakukan perangkingan data bedasarkan bobot kata kunci terhadap kumpulan dokumen teks abstrak dalam perpustakaan online STMIK Denpasar.
      2. Sistem pencarian dengan implementasi *text* *mining* dan metode *Vector* *space* *model* serta pemanfaatan *indexing* data dengan *crontab* dapat membatu melakukan perangkingan data berdasarkan bobot kata kunci terhadap kumpulan dokumen teks abstrak dalam perpustakaan online STMIK Denpasar.
      3. Pengurutan hasil pencarian dengan menggunakan metode *vector space model* ini dilakukan dengan mengurutkan bobot dokumen secara *descending.*
      4. Semakin besar bobot suatu dokumen maka semakin tinggi tingkat kemiripan dokumen tersebut terhadap suatu kata kunci.
  1. **Saran**

Adapun saran yang hendak penulis sampaikan yaitu :

1. Diharapkan agar sistem pencarian ini dapat dikembangkan lebih jauh lagi tidak hanya meliputi pencarian dalam satu kategori tetapi mencakup lebih banyak kategori, misalnya buku-buku.
2. Untuk penelitian selanjutnya diharapkan untuk perbaikan dalam pembangunan sistem pencarian maupun sistem *indexing* data agar proses pencarian dapat lebih baik lagi.
3. **Daftar pustaka**
4. Afriyudi, 2008. *Pemrograman Web Dinamis dengan Kolaborasi PHP & Java*. Yogyakarta : Penerbit Andi.
5. Agusta, L, 2009. *Perbandingan Algoritma Stemming Porter dengan Algoritma Nazief & Adriani Untuk Stemming Dokumen Teks Bahasa Indonesia.* Bali: Konferensi Nasional Sistem dan Informatika.
6. Anonin, 2010, Analisis dan konfigurasi skrip sedot untuk mirroring repository Linux. [*http://opensource.telkomspeedy.com/wiki/index.php/Cron*](http://opensource.telkomspeedy.com/wiki/index.php/Cron). Diakses tanggal 30 mei 2012, 08:30.
7. Anonim, 2010 Pemprograman web. [*http://id.wikipedia.org/wiki/Pemrograman\_web*](http://id.wikipedia.org/wiki/Pemrograman_web). Diakses tanggal 30 mai 2012, 08:30.
8. Cosine similarity, 2012.[*http://en.wikipedia.org/wiki/Cosine\_similarity*](http://en.wikipedia.org/wiki/Cosine_similarity)*.* Diakses tanggal 30 mei 2012.
9. Feldman, Ronen 2006. Applied Text mining. www.cis.upnn.edu/~ungar/KDD/KDD\_tutorial.pdf. [Diunduh tanggal 30 juni 2012].
10. Halrian, C, Mikha, 2006. Text mining.[Online]. [*http://lecturer.eepis-its.edu/~iwanarif/kuliah/dm/6Text%20Mining.pdf*](http://lecturer.eepis-its.edu/~iwanarif/kuliah/dm/6Text%20Mining.pdf) [Diunduh tanggal 01 Juni 2012].
11. Harjono, K. D, 2005.Perluasan vektor pada metode search vector space.[Online].*http://www.google.co.id/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CEgQFjAA&url=http%3A%2F%2Fhome.unpar.ac.id%2F~integral%2FVolume%252010%2FIntegral%252010%2520No.%25202%2FPerluasan%2520Vektor.pdf&ei=LGEGUMerGoiyrAeB7KjFBg&usg=AFQjCNHk7lUUNEPXAeeLnX6LHiauw-ajtQ&sig2=TbtkPql\_iTMPi2EVqt7spw* [Diunduh tanggal 30 juni 2012].
12. Ladjamudin, Al Bahra, 2006. *Rekayasa Perangkat Lunak.* Yogyakarta : Graha Ilmu.
13. MySQL , 2012. [*http://id.wikipedia.org/wiki/MySQL*](http://id.wikipedia.org/wiki/MySQL)*.* (Diakses tanggal 23 april 2012).
14. MVC, 2012. http://id.wikipedia.org/wiki/MVC.(Diakses tanggal 24 april 2012).
15. Novanta, Audi, 2009. *Pendeteksian Plagiarisme Pada Dokumen Teks Dengan Menggunakan Algoritma Smith-Waterman*. Medan : Universitas Sumatera

Utara .

1. Sismanto , 2007. Manajemen Perpustakaan Digital. Jakarta: Afifa Pustaka.
2. Simarmata, Janner, 2007. *Perancangan Basis Data*. Yogyakarta : Penerbit Andi.
3. XML, 2012. *http://id.wikipedia.org/wiki/EML.* Diakses tanggal 30 mei 2012, 16.36.
4. Wibisono, Y., Khodra, M. L. 2006. Clustering Berita Berbahasa Indonesia.*www*.fpmipa.upi.edu/staff/yudi/KNSI\_Clustering\_yudi\_masayu.pdf. [Diunduh tanggal 30 juni 2012].