INFORMATION RETRIEVAL SYSTEM ARTIKEL KESEHATAN MENGGUNAKAN PEMBOBOTAN TF.IDF DAN *LATENT SEMANTIC INDEXING*

SKRIPSI

Oleh :

MUHAMMAD ISMAIL HASAN
NIM. 13650047

JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2018

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

INFORMATION RETRIEVAL SYSTEM ARTIKEL KESEHATAN MENGGUNAKAN PEMBOBOTAN TF.IDF DAN *LATENT SEMANTIC INDEXING*

SKRIPSI

Diajukan kepada :
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :
MUHAMMAD ISMAIL HASAN
NIM.13650047

JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2018

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

LEMBAR PERSETUJUAN

INFORMATION RETRIEVAL SYSTEM* ARTIKEL KESEHATAN MENGGUNAKAN PEMBOBOTAN TF.IDF DAN *LATENT SEMANTIC INDEXING

SKRIPSI

Oleh :
MUHAMMAD ISMAIL HASAN
NIM. 13650047

Telah Diperiksa dan Disetujui untuk Diuji

Tanggal: 02 Juli 2018

Dosen Pembimbing I	Dosen Pembimbing II
Hani Nurhayati, M.T	Khadijah F.H. Holle, M.Kom
NIP. 19780625 200801 2 006	NIDT. 19900626 20160801 2 077

Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

LEMBAR PENGESAHAN

INFORMATION RETRIEVAL SYSTEM* ARTIKEL KESEHATAN MENGGUNAKAN PEMBOBOTAN TF.IDF DAN *LATENT SEMANTIC INDEXING

SKRIPSI

Oleh :

MUHAMMAD ISMAIL HASAN
NIM. 13650047

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 02 Juli 2018

Susunan Dewan Penguji			**Tanda Tangan**
Penguji Utama	:	Linda Salma Angreani, M.T NIP. 19770803 200912 2 005	( )
Ketua Penguji	:	Fresy Nugroho, M.T NIP. 19710722 201101 1 001	( )
Sekretaris Penguji	:	Hani Nurhayati, M.T NIP. 19780625 200801 2 006	( )
Anggota Penguji	:	Khadijah F.H. Holle, M. Kom NIDT. 19900626 20160801 2 077	( )

Mengesahkan,

Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

HALAMAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama	: Muhammad Ismail Hasan
NIM	: 13650047
Jurusan	: Teknik Informatika
Fakultas	: Sains dan Teknologi

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku.

Malang, 5 Juli 2018

Yang membuat pernyataan

Muhammad Ismail Hasan

NIM. 13650047

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

MOTTO

Jangan Kalah dengan Keadaan

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

HALAMAN PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

Dengan Rahmat Allah yang Maha Pengasih Lagi Maha Penyayang

Dengan ini saya persembahkan karya ini untuk

Bapak Muhamad Chozin dan Ibu Siti Chosidah

Sebagai tanda bakti, hormat, dan rasa terima kasih yang tiada terhingga kupersembahkan karya ini kepada bapak dan ibu yang telah memberikan kasih sayang, segala dukungan.

Kakakku Mohamad Zaenul Huda dan Adikku Muhammad Lukmanul Hakim

Meskipun jarang ada perbicaraan serius, tetapi adanya kakak dan adik selalu membuat betah dirumah dan menjadi candu dalam pertemuan. Terima kasih atas doa dan semangatnya selama ini.

Serta keluarga besar

Yang telah memberi semangat dan dukungan yang tiada henti-hentinya.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

KATA PENGANTAR

Assalamu'alaikum Wr.Wb.

Segala puji bagi Allah SWT tuhan semesta alam, karena atas segala rahmat dan karunia-Nya sehingga penulis mampu menyelesaikan skripsi dengan judul "*Information Retrieval System* Artikel Kesehatan Menggunakan Pembobotan TF.IDF dan *Latent Semantic Indexing*" dengan baik dan lancar. Shalawat serta salam selalu tercurah kepada tauladan terbaik Nabi Muhammad SAW yang telah membimbing umatnya dari zaman kebodohan menuju Islam yang *rahmatan lil alamiin.*

Dalam menyelesaikan skripsi ini, banyak pihak yang telah memberikan bantuan baik secara moril, nasihat dan semangat, maupun materiil. Atas segala bantuan yang telah diberikan, penulis ingin menyampaikan doa dan ucapan terimakasih yang sedalam-dalamnya kepada.

1. Dr. Bayyinatul Muchtaromah, drh. M.Si, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Cahyo Crysdian selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Hani Nurhayati, M.T selaku Pembimbing I yang telah meluangkan waktu untuk membimbing, memotivasi, mengarahkan, dan memberi masukan kepada penulis dalam pengerjaan skripsi ini hingga akhir.
4. Khadijah F.H. Holle, M.Kom selaku Pembimbing II yang telah meluangkan waktu, membimbing, dan mengarahkan penulis dalam pengerjaan skripsi ini hingga akhir.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

5. Segenap Dosen Teknik Informatika yang telah memberikan bimbingan keilmuan kepada penulis selama masa studi.
6. Kedua orang tua yang telah memberikan motivasi demi terselesainya tugas akhir ini.
7. Teman-teman kontrakan 53A yang telah bersama-sama hidup satu atap, susah senang bareng, makan makan bareng dan selalu membantu disaat di butuhkan.
8. Semua teman-teman FBI, Wana, Awwalia, Linda, Heni, Visa, Vita dkk yang tidak saya sebutkan satu bersatu, yang selalu mambantu saya bagi saya.
9. Teman-teman seperjuangan Teknik Informatika angkatan 2013.
10. Semua Saudara-saudara IKRAM Cabang Malang yang juga selalu memberikan semangat juang.
11. Desi Rahmawati yang selalu mengingatkan, memberikan semangat dan selalu mambantu dalam menghadapi skripsi ini.

Semoga apa yang menjadi kekurangan bisa disempurnakan oleh peneliti selanjutnya dan semoga karya ini senantiasa dapat memberi manfaat. Amin.

Wassalamualaikum Wr.Wb

Malang, 5 Juli 2018

Penulis

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

DAFTAR ISI

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

DAFTAR GAMBAR

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

DAFTAR TABEL

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

ABSTRAK

Hasan, Muhammad Ismail. 2018. ***Information Retrievel System* Artikel Kesehatan Berbahasa Indonesia Menggunakan Pembobotan TF.IDF dan *Latent Semantic Indexing*.** Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Pembimbing : (I) Hani Nurhayati, M.T, (II) Khadijah F.H. Holle, M.Kom

Kata Kunci : *Text Mining, Information Retrievel System, Term Weighting*

Dewasa ini, pertumbungan teknologi digital begitu pesat yang mengakibatkan semakin besar data dan informasi yang tersebar di internet. *Information Retrieval System* diperlukan agar mempermudah dalam melakukan pencarian data berdimensi yang besar tersebut. Salah satu pemanfaatan *information retrieval system* adalah digunakan untuk melakukan pencarian terhadap berbagai keluhan kesehatan yang dialami para pengguna internet. Salah satu metode yang sangat populer adalah perangkingan dokumen menggunakan *vector space model* berbasis pada nilai pembobotan TF.IDF. Motode tersebut hanya melakukan pembobotan berdasarkan frekuensi kemunculan kata dalam dokumen tanpa memerhatikan preferensi yang diinginkan pengguna. Metode pembobotan kata berdasarkan preferensi pengguna perlu mempertimbangkan hubungan semantik antar kata untuk meningkatkan relevansi hasil pencarian. *Latent Semantic Indexing* merupakan salah satu metode *indexing* dalam *information retrieval system* yang mempertimbangkan hubungan semantik antar kata. Penelitian ini mengembangkan metode pembobotan TF.IDF dengan menambahkan *Latent Semantic Indexing*. Data yang digunakan adalah kumpulan dokumen artikel kesehatan yang diambil dari beberapa *website* yang dipilih oleh peneliti. Hasil pengujian menunjukkan bahwa metode pembobotan TF.IDF.LSI menghasilkan nilai *Mean Average Precision* (MAP) yang lebih tinggi daripada metode TF.IDF. Secara berurut nilai MAP metode pembobotan TF.IDF.LSI pada *rank-k* 10 dan 15 adalah 86% dan 82.4%. Nilai tersebut lebih tinggi dari metode pembobotan TF.IDF yang menghasilkan nilai MAP pada *rank-k* 10 dan 15 adalah 82.8% dan 79%.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

ABSTRACT

Hasan, Muhammad Ismail. 2018. **Information Retrieval System Indonesian Health Articles Using TF.IDF and Latent Semantic Indexing.** Thesis. Department of Informatic Engineering. Faculty of Science and Technology. State Islamic University of Maulana Malik Ibrahim Malang.

Adviser : (I) Hani Nurhayati, M.T, (II) Khadijah F.H. Holle, M.Kom

Keywords : *Text Mining, Information Retrievel System, Term Weighting*

Recently, the growth of digital technology so rapidly that result in greater data and information spread on the internet. The Information Retrieval System is required to make it easier to search the large dimensioned data. One of the utilization of information retrieval system is used to search the various health complaints experienced by internet users. One of the most popular methods is document ranking using the vector space model based on the weighted value of TF.IDF. The method only performs weighting based on the frequency of occurrence of term in the document regardless of user preferences. The term weighting method based on user preferences needs to consider the semantic relation between term to improve the relevance of search results. Latent Semantic Indexing is one of the indexing methods in the information retrieval system that considers the semantic relation between term. This research develops the method of weighting TF.IDF by adding Latent Semantic Indexing. The data used is a collection of health article documents taken from some website selected by researchers. The results show that the TF.IDF.LSI produces a higher Mean Mean Precision (MAP) value than the TF.IDF. In sequential MAP values of TF.IDF.LSI at rank-k 10 and 15 are 86% and 82.4%. This value is higher than the TF.IDF that result MAP values at rank k-10 and 15 are 82.8% and 79%.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

ملخص البحث

حسن، محمد إسماعيل. 2018. نظام استرجاع المعلومات (*Information Retrieval System*) لمادة الصحية بالاندونيسية باستخدام الترجيح TF.IDF و فهرسة الدلالة الكامنة (*Latent Semantic Indexing*). البحث الجامعي. قسم المعلوماتية كلية العلوم والتكنولوجيا الجامعة الإسلامية الحكومية مولانا مالك إبراهيم مالانج.

الاشراف: هاني نورحياتي،الماجستير ، (2) خديجة ف.ح خولى، الماجستير

الكلمات الرئيسية: تعدين النص ، نظام استرجاع المعلومات ، ترجيح الألفاظ

اليوم، نمو التكنولوجيا الرقمية بسرعة كبيرة مما أدى إلى زيادة البيانات والمعلومات المنتشرة على شبكة الإنترنت. يطلب نظام استرجاع المعلومات لتسهيل البحث البيانات ذات الأبعاد الكبيرة. يستخدم أحد استخدام نظام استرجاع المعلومات للبحث الشكاوى الصحية المختلفة التي تتعرض لمستخدمو الإنترنت. واحدة من شعبيات كبيرة هي تصنيف الوثائق باستخدام نموذج فضاء المتجه على أساس القيمة المرجحة لـ TF.IDF. تقوم الطريقة فقط بإجراء عملية ترجيح استنادًا إلى تكرار حدوث الكلمات في المستند بغض النظر عن تفضيلات المستخدم. يجب أن ان يعتبر العلاقة الدلالية بين الكلمات لتحسين ملاءمة نتائج البحث. فهرسة الدلالة الكامنة هي واحدة من طرائق الفهرسة في نظام استرجاع المعلومات التي تعتبر العلاقة الدلالية بين الكلمات. تطور هذه الطريقة بترجيح TF.IDF بإضافة فهرسة الدلالة الكامنة. البيانات المستخدمة هي مجموعة من وثائق المقالات الصحية التى أخذت من العديد من الانترنيت التي اختارتها للباحث. ظهرت نتائج الاختبار أن أسلوب الترجيح TF.IDF.LSI حصلت قيمة متوسط الدقة (*Mean Average Precision*) (MAP) أعلى من الأسلوب TF.IDF في قيم MAP متسلسلة من أسلوب الترجيح TF.IDF.LSI في رتبة-ك 10 و 15 هو 86٪ و 82.4٪. هذه القيمة هي أعلى من طريقة ترجيح TF.IDFالتي حصلت قيمة MAP في رتبة-ك 10 و 15 يعنى 82.8٪ و 79٪.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan cepatnya pertumbuhan informasi digital, mengakibatkan semakin besarnya dimensi data atau informasi yang ada. Dimensi data yang besar dapat memicu informasi yang penting menyebar terlalu luas, sehingga menjadi kurang informatif. Hal tersebut juga mengakibatkan berkembangnya teknologi informasi yang dibutuhkan oleh pengguna sehingga mengakibatkan munculnya suatu cabang ilmu baru dalam teknologi informasi yaitu *information retrieval* (IR) (T. Wicaksono, 2005). Cabang ilmu tersebut tentu diperlukan agar mempermudah dalam melakukan pencarian data berdimensi yang besar tersebut.

Penerapan *information retrieval* (IR) yang sering dijumpai adalah pada mesin pencari (*search engine*). Mesin pencari (*search engine*) adalah kombinasi perangkat keras dan perangkat lunak komputer yang disediakan oleh perusahaan tertentu melalui *website* yang telah ditentukan (Abdillah, Falkner, & Hemer, 2010).

Salah satu pemanfaatan *search engine* ini adalah dalam bidang kesehatan. Dimana digunakan untuk melakukan pencarian terhadap berbagai keluhan kesehatan yang dialami para pengguna *internet*. Namun masalahnya adalah pemakaian mesin pencari masih sering mempersulit pengguna, diantaranya pada saat mengumpulkan artikel kesehatan, hasil percarian terlalu luas sehingga informasi yang dicari kurang spesifik. Disamping itu juga, banyak informasi yang hasilkan dari *website* yang masih diragukan ke-valid-annya karena tidak tertera ahli penanggungjawabnya. Dalam penelitian ini, akan diusulkan suatu sistem

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

information retrival (IR) yang spesifik untuk memudahkan *users* dalam melakukan pencarian tentang masalah kesehatan. Data yang digunakan diambil dari beberapa *website* kesehatan yang dipilih.

Secara garis besar, pokok pembahasan IR meliputi 2 hal yakni pembobotan dan perangkingan (Holle, Arifin, & Purwitasari, 2015). Perangkingan digunakan untuk mendapatkan dokumen-dokumen yang relevan dengan *query* pengguna diurutkan dari tingkat relevansinya. Perangkingan ini diperlukan karena mungkin dokumen yang sesuai dengan *query* pencarian lebih dari satu.

Salah satu metode yang dapat digunakan dalam perangkingan dokumen adalah *Vector Space Model* (VSM). Pada metode ini dokumen direpresentasikan sebagai sebuah vektor yang dibentuk dari nilai-nilai *term* yang menjadi indeknya. Harrag dkk (2008) menggunakan *vector space model* berbasis *term* weighting TF-IDF untuk melakukan perangkingan pada dokumen berbahasa Arab. TF.IDF merupakan kombinasi dari metode *Term Frequency* (TF) yang merupakan mengukur kepadatan *term* dalam dokumen, dengan metode *Inverse Term Frequency* (IDF) yang mengukur kepentingan sebuah *term*.

Untuk meningkatkan relevansi *query* terhadap hasil pencarian, dibutuhkan metode untuk mempertimbangkan hubungan semantic antar *term*. Maka digunakan metode *Latent Semantic Indexing* (LSI) yang mampu mengidentifikasi makna semantik antar *term* berdasarkan pola hubungan yang ada pada koleksi dokumen/artikel. Sifat utama LSI adalah kemampuannya untuk membangun hubungan antara istilah yang muncul dalam konteks yang serupa (Wahib, Pasnur,

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Santika, & Arifin, 2015). Selain itu, LSI diperlukan karena dalam pembahasan kesehatan, banyak sekali istilah-istilah yang bisa memiliki makna berbeda-beda, tergantung pada penerapan konteksnya.

Oleh karena itu, dalam penelitian ini diusulkan kombinasi metode TF.IDF dengan metode *Latent Semantic Indexing* (LSI) yang diimplementasikan pada sistem *information retrieval* (IR) artikel kesehatan berbahasa Indonesia. Penggunaan kedua metode TF.IDF dan LSI ini diharapkan dapat meningkatkan relevansi perangkingan artikel kesehatan yang tepat dan sesuai yang diharapkan *users*.

1.2 Identifikasi Masalah

Berdasarkan apa yang telah sebelumnya dijelaskan latar belakang maka permasalahan yang diangkat peneliti pada penelitian ini adalah seberapa akurat implementasi metode pembobotan TF.IDF dan LSI untuk *information retrieval system* artikel kesehatan berdasarkan perhitungan *Mean Average Precision* (MAP)?

1.3 Tujuan Penelitian

Berdasarkan pada indentifikasi masalah, tujuan pokok yang didapat dalam penelitian ini adalah untuk menemukan relevansi artikel dari *keyword* yang dimasukkan pengguna. Lebih spesifiknya adalah mengetahui tingkat akurasi *Mean Average Precision* (MAP) pada implementasi metode TF.IDF dan LSI untuk *information retrieval system* artikel kesehatan.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

1.4 Manfaat Penelitian

Berdasarkan uraian pada tujuan penelitian, secara garis besar penelitian ini diperuntukkan untuk memudahkan *users* dalam melakukan pencarian tentang masalah kesehatan. Kelebihannya adalah sistem ini lebih spesifik pada artikel kesehatan yang diambil dari *website-website* yang dipilih oleh peneliti. Sehingga output dari pencarian tidak terlalu luas.

Berdasarkan manfaat penggunaan metode TF.IDF dan LSI adalah dapat meningkatkan relevansi antara *query* dari *users* dan *corpus*, sehingga berdampak pada kepuasan *users* dalam melakukan pencarian menggunakan sistem ini. Selain itu juga, dapat mengoptimalkan waktu *users* dalam mencari informasi.

1.5 Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut :

1. Data artikel di kumpulakan dari beberapa *website* yang telah dipilih, yang memiliki url sebagai berikut : (1) *http://www.alodokter.com/*, (2) *https://www.dokter.id/*, dan (3) *https://doktersehat.com/*.
2. Data artikel yang digunakan adalah teks berbahasa Indonesia.
3. Data artikel yang diolah merupakan *file* berformat *textfile* yang di ambil secara manual dari *website* yang dipilih.
4. Mengecualikan kata yang mengandung frasa pada tahap *preprocessing*.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

`

BAB 2

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Endah Purwati (2015) dalam penelitiannya untuk mengetahui bagaimana penerapan sistem pencarian informasi dalam klasifikasi jurnal. Data penelitian berupa dokumen-dokumen jurnal, kemudian di kontruksi dengan teknik text mining, melakukan pembobotan setiap token hasil kontruksi, lalu menghitung kemiripan antar dokumen menggunakan *Vector Space Model* (kesamaan *Cosine Similarity*), dan kemudian di klasifikasi menggunakan *k-Nearest Neighbor*. Hasil penelitiannya menunjukkan dokumen dapat diklasifikasikan sesuai dengan kategori yang sebenarnya.

Yuita Arum Sari dan Eva Yulia Puspanigrum (2015), dalam penelitiannya mengenai sistem pencarian *semantic* dokumen berita. Data yang digunakan adalah dataset artikel berupa plaint text yang didapat dari *website* www.kompas.com. Pembobotan pada kata-kata yang penting menggunakan reduksi seleksi fitur yang merupakan kombinasi *Document Frequency* (DF) *thresholding*, dan *Information Gain* (IG). Untuk perhitungan dalam mencari dokumen yang relevan antara *query* dan *corpus* menggunakan metode *Essential Dimension Of Latent Semantic Indexing* (EDLSI).

Pemanfaatan metode *Vector Space Model* (VSM) dan metode *Cosine Similarity* juga dilakukan oleh Ana Triana dkk (2014), penerapannya adalah untuk mendeteksi hama dan penyakit tanaman padi. Untuk mendeteksi penyakit dibutuhkan *input*/*feedback* dari pengguna, namun karena banyaknya gejala dari

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

semua daftar hama dan penyakit membuat pengguna harus memberikan *input*/*feedback* sebanyak daftar semua gejala yang ada, maka pada proses masukan gejala pada fitur ini dibuat dengan berbasis tekstual sehingga pengguna dapat langsung memberikan input/feedback tanpa harus menjawab satu per satu. Metode *Vector Space Model* (VSM) digunakan untuk mendapatkan daftar gejala yang sesuai dengan *input*/*feedback* dengan menentukan kemiripan diantara keduanya. Hasilnya kemudian dihitung dengan metode *Cosine Similarity* untuk mendeteksi hama dan penyakit tanaman padi. Pengujian dengan 25 percobaan, metode*Vector Space Model* (VSM) dalam mengidentifikasi *input*/*feedback* menghasilkan akurasi sebesar 96%.

Darmawan & Wurijanto (2010), melakukan penelitian tentang aplikasi search engine tafsir Al Quran. Metode algoritma yang dipakai adalah *Vector Space Model* (VSM) dan *text mining* untuk pemrosesan datanya. Setiap Ayat Al Quran digunakan sebagai input data. Pembobotannya menggunakan TF-IDF dan *Vector Space Model* (VSM) sebagai algortima pemodelan, perhitungan *similarity* antara *query* dengan dokumen menggunakan *Cosine Similarity*. Pengujian dan evaluasi menghasilkan bahwa kemampuan aplikasi untuk mencari ayat yang relevan adalah sebesar 96,3% dari 35 pengujian *query*. Dan dilakukan evaluasi kepuasan terhadap 30 responden menghasilnya nilai 71,67%.

Penerapan metode pembobotan TF.IDF.ICF.IBF juga dilakukan oleh Wahib, Pasnur, Santika, & Arifin (2015), diterapkan untuk perangkingan dokumen Berbahasa Arab. Tetapi dalam penelitian ini, metode tersebut di kombinasikan dengan metode *Latenr Sematic Indexing* (LSI). Penggunaan metode tersebut

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

adalah karena pentingnya hubungan semantic antar *term* untuk meningkatkan relevansi hasil pencarian dokumen. Dataset yang digunakan diambil dari kumpulan dokumen pada perangkat lunak Maktabah Syamilah. Hasilnya menunjukkan bahwa metode yang diusulkan memberikan nilai evaluasi yang lebih baik dibandingkan dengan metode TF.IDF.ICF.IBF dengan nilai *f-measure* pada ambang *cosine similarity* 0.3, 0.4, dan 0.5 adalah 45%, 51%, dan 60%. Tetapi memiliki waktu komputasi yang lebih tinggi 2 menit 8 detik.

Wardhana, Yunianto, Arifin, & Purwitasari (2015) melakukan penelitian pada dokumen fiqh berbahasa Arab dengan menggunakan metode pembobotan kata yang berbasis preferensi pengguna berdasarkan hubungan semantic antar kata. Metode pembobotannya disebut Inverse Preference Indexing (IDF) yang kemudian dikombinasikan dengan TF.IDF.IBF sehingga menjadi TF.IDF.IBF.IPF. sedangkan semantiknya menggunakan metode *Latent Semantic Indexing (LSI).* Hasilnya menunjukkan nilai *precision* sebesar 88,75%, *recall* 89,72% dan *f-measure* 87,91%.

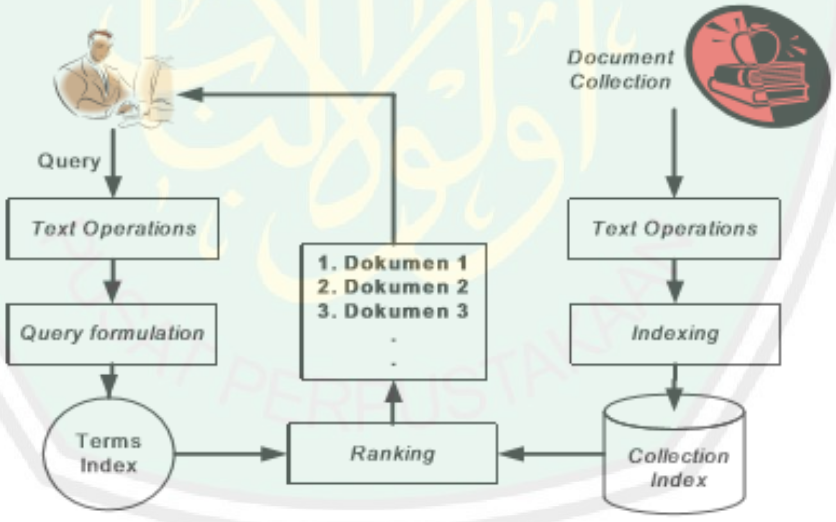
2.2 Information Retrieval (IR)

Information Retrieval (IR) adalah menemukan materi (biasanya dokumen) dari sebuah kumpulan data yang tidak terstruktur (biasanya teks) untuk memenuhi kebutuhan informasi dari koleksi yang besar (Manning dkk, 2008). Tujuan *Information Retrieval* (IR) adalah untuk menjawab kebutuhan informasi *user* dengan sumber informasi yang tersedia dalam kondisi seperti sebagai berikut :

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

1. Mempresentasikan sekumpulan ide dalam sebuah dokumen menggunakan sekumpulan konsep.
2. Terdapat beberapa pengguna yang memerlukan ide, tapi tidak dapat mengidentifikasikan dan menemukannya dengan baik.
3. *Information Retrieval (IR)* bertujuan untuk mempertemukan ide yang dikemukakan oleh penulis dalam dokumen dengan kebutuhan informasi penggguna yang dinyatakan dalam bentuk *keyword* dan *query* /istilah penelusuran.

Sebagai suatu sistem, sistem IR memiliki beberapa bagian yang membangun sistem secara keseluruhan. Bagian-bagian yang terdapat dalam IR digambarkan pada gambar 2.1.



Gambar 2. 1 Bagian-bagian dari sistem information retrievel

(Sumber : Bunyamin & Negara, 2008)

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

1. *Text Operations* (operasi terhadap teks) yang meliputi pemilihan kata-kata dalam *query* maupun dokumen (*term selection*) dalam pentransformasian dokumen atau *query* menjadi *term* index (indeks dari kata-kata).
2. *Query formulation* (formulasi terhadap *query*) yaitu memberi bobot pada indeks kata-kata *query*.
3. *Ranking* (perangkingan), mencari dokumen-dokumen yang relevan terhadap *query* dan mengurutkan dokumen tersebut berdasarkan kesesuaiannya dengan *query*.
4. *Indexing* (pengindeksan), membangun basis data indeks dari koleksi dokumen. Dilakukan terlebih dahulu sebelum pencarian dokumen dilakukan.

Prinsip kerja dari *information retrieval* (IR) adalah jika ada sebuah kumpulan dokumen dan seorang user yang memformulasikan sebuah pertanyaan (*query* dan *request*). Sistem IR menerima *query* dari pengguna, kemudian melakukan perangkingan terhadap dokumen pada koleksi berdasarkan kesesuaiannya dengan *query*. Hasil perangkingan yang diberikan kepada pengguna merupakan dokumen yang menurut sistem relevan dengan *query*.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

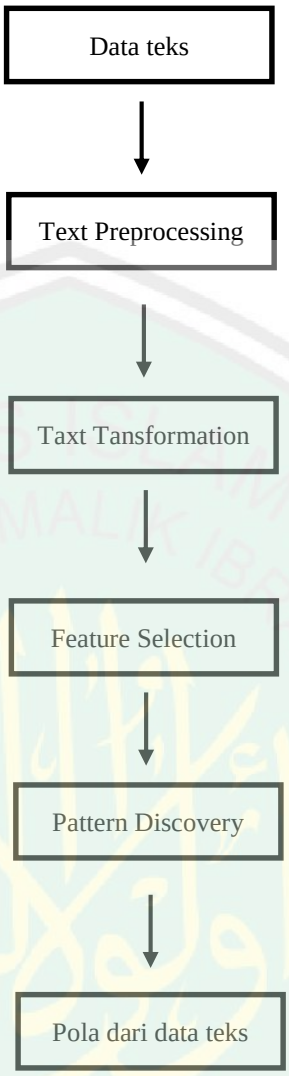
2.3 Text Mining

Menurut Feldman & Sanger (2007) , *Text Mining* adalah penambangan yang dilakukan oleh computer untuk mendapatkan sesuatu yang baru dalam bentuh sebuah informasi, sesuatu yang tidak diketahui sebelumnya atau menemukan kembali informasi yang tersirat secara implisit, yang berasal dari informasi yang diekstrak secara otomatis dari sumber-sumber data teks yang berbeda-beda.

Sebagian menganggap bahwa *text mining* adalah serupa dengan *data mining* karena keduanya memiliki tujuan yang sama yaitu untuk memperoleh informasi dan pengetahuan dari sekumpulan data yang sangat besar. Namun sebenarnya terdapat perbedaan dari keduanya, dalam hal pola, dimana *data mining* mencari suatu pola dari data yang tersetruktur, sedangkan pada *text mining* mencari pola pada data yang semi atau bahkan tidak tersetruktur. Tetapi secara garis beras, prinsip kerja dari *text mining* mengadopsi dari *data mining*.

Input dari *text mining* adalah berupa data teks dan menghasilkan output pola berupa taksiran. *Text mining* memiliki 4 tahapan yakni (1) *Text Preprocessing*, (2) *Text Transformation*, (3) *Feature Selection*, dan (4) *Pattern Discovery*. Bagannya sebagaimana digambarkan pada gambar 2.2.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG



Gambar 2. 2 Tahapan Proses *Text Mining*

2.3.1 Text Preprocessing

Tahap paling awal dalam melakukan *text mining* adalah mempersiapkan teks yang akan digunakan. *Text Preprocessing* adalah proses bagaimana mempersiapkan teks mentah yang akan diolah lebih lanjut. Suatu data teks atau karakter yang bersambung harus dipecah-pecat menjadi beberapa unsur yang berarti. Tahapan-tahapan text preprocessing adalah sebagai berikut :

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

1. Case Folding

Case Folding adalah mengubah semua huruf dalam dokumen menjadi huruf kecil. Hanya huruf a sampai z yang diterima. Karakter selain huruf dihilangkan dan dianggap delimiter (Salim & Anistyasari, 2017). Hal ini diperlukan untuk mengkonversi keseluruhan teks dalam dokumen menjadi bentuk standar (*lowercase*) agar mempermudah pencarian nantinya.

2. Tokenisasi

Tokenisasi atau parsing adalah proses memecahkan teks menjadi kalimat dan token (Feldman, 2007). Pada tahap ini dilakukan penghilangan beberapa karakter tertentu yang dianggap tidak penting, seperti kapitalisasi, keberadaan digit, tanda baca, karakter special dan lain sebagainya. Hasil dari tahap ini sangatkan penting karena akan digunakan dalam tahap selanjutnya.

2.3.2 Text Transformation

Text Transformation adalah tahapan yang digunakan untuk mengubah kata-kata kedalam bentuk dasar, sekaligus mengurani jumlah kata-kata tersebut. Secara garis besar melalui 2 tahap yakni :

1. Filtering/Stopword Removal

Proses Filtering/Stopword Removal merupakan proses penghapusan *term*.token yang tida memiliki arti atau tidak relevan. Eliminasi Filtering/Stopword Removal memiliki banyak mkeuntungan, yakni akan mengurangi *space* pada tabel *term index* hingga 40% atau lebih (Baeza & Ribeiro, 1999).

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

2. Stemming

Stemming adalah proses untuk menggabungkan atau memecahkan setiap varian-varian suatu kata menjadi kata dasar. Pada dasaranya proses ini merupakan pencarian *root* kata dari tiap kata hasil filtering. Proses stemming dilakukan dengan cara menghilangkan semua imbuhan (*affixes*) baik yang terdiri dari awalan (*prefixes*), sisipan (*infixes*), akhiran (*suffixes*) dan *confixes* (kombinasi dari awalan dan akhiran) pada kata turunan.

2.3.3 Feature Selection

Feature Selection bertujuan untuk mengurangi dimensi dari suatu kumpulan teks atau menghapus kata-kata yang dianggap tidak penting atau tidak menggambarkan isi dokumen berdasarkan frekuensi kemunculan kata tersebut. Tahapan ini merupakan tambahan jika kata yang tersisa belum mampu menggambarkan isi dari teks dokumen (Indranandita, Santoso, & Rachmat, 2008).

2.3.4 Pattern Discovery

Pattern Discocery merupakan tahapan final dan ini dari teknik *text mining*, yakni menemukan suatu pola pada sekumpulan dokumen teks.

2.4 Perangkingan Dokumen

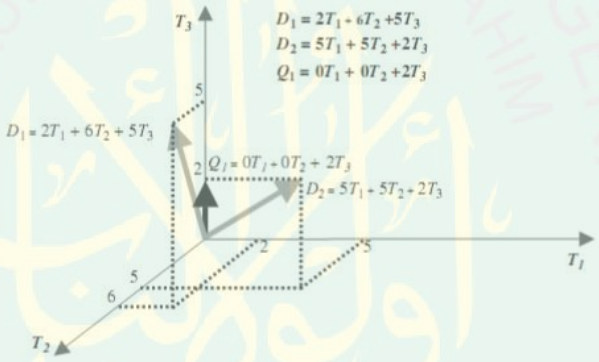
2.4.1 *Vector* Space Model (VSM)

Vector Space Model (VSM) adalah suatu model aljabar untuk mewakili dokumen teks sebagai suatu vektor pengenal, contohnya indeks kata. VSM biasanya digunakan dalam penyaringan informasi, temu balik informasi, pengindeksan, dan perankingan relevansi (Triana et al., 2014). Dalam *Vector Space*

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Model (VSM), dokumen yang ada di *database* direpresentasikan ke dalam *vector* multidimensi. Dimensi dari vektor berkorespondensi dengan jumlah setiap *term* dalam *database* dan kumpulan *term* tersebut membentuk suatu ruang vektor.

Setiap *term i*, didalam dokumen maupun *query*, diberikan suatu bobot yang bernilai rela w_{ij}. Dokumen dan *query* di ekpresikan sebagai *vector* t –dimensi $d_j = (w_{1j}, w_{2j}, \ldots, w_{ij})$ dan diasumsikan terdapat n dokumen di dalam *database*, yaitu $j = 1,2, \ldots, n$. Contohnya bisa dilihat pada gambar 2.3.

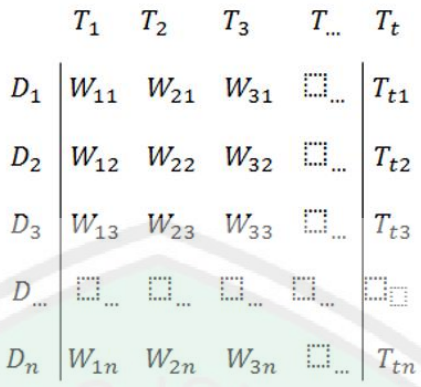


Gambar 2. 3 Model Ruang *Vector*

(Sumber : Cios, 2007).

Setiap sel dalam matrik bersesuaian dengan bobot yang diberikan dari suatu *term* dalam dokumen yang ditentukan. Nilai nol berarti bahwa *term* tersebut tidak hadir dalam dokumen. Gambar 2.4 femerupakan gambaran matrik dari *term*-document:

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG



Gambar 2. 4 Matrik *Term-Document*.

(Sumber : Darmawan & Wurijanto, 2010)

2.4.2 Pembobotan *Term*

Hal yang perlu diperhatikan dalam pencarian informasi dari koleksi dokumen yang heterogen adalah pembobotan *term*. *Term* dapat berupa kata, frase atau unit hasil indexing lainnya dalam suatu dokumen yang dapat digunakan untuk mengetahui konteks dari dokumen tersebut, maka untuk setiap kata tersebut diberikan indikator, yaitu *term weight*.

1. *Term* Frequency (TF)

   Hasil *term-term* dari tahap *preprocessing* yang telah disimpan di *database* dilakukan dua proses secara paralel yakni menghitung jumlah *term* dalam dokumen dan menghitung jumlah dokumen yang mengandung *term* tersebut. Jumlah dokumen keseluruhan juga dihitung sebagai variabel pendukung algoritma ini dan data-data dokumen diakses berasal dari *database* dokumen.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Dalam artian sebenarnya, proses diatas adalah proses *term frequency* (TF). Dimana merupakan salah satu metode untuk menghitung bobot tiap *term* dalam teks. TF menyatakan banyaknya suatu kata muncul dalam dokumen (Feldman, 2007). Setiap *term* diasumsikan memiliki kepentingan yang proporsional terhadap jumlah kemunculan *term* pada dokumen. Bobot sebuah *term* t pada sebuah teks dirumuskan dalam persamaan berikut :

$$W_{d,t} = tf_{d,t} \quad \text{..... (2.1).}$$

Term frequency dapat memperbaiki nilai *recall* pada *information retriviel* (IR), tetapi tidak selalu memperbaiki *precision*. Hal ini dapat diperbaiki dengan membuang *term* dari *termset* yang mempunyai nilai frekuensi tertinggi.

2. Inverse Document Frequency (IDF)

IDF (*Inverse Document Frequency*) merupakan didefinisikan sebagai logaritma dari rasio jumlah keseluruhan dokumen yang diproses dengan jumlah dokumen yang memiliki *term* bersangkutan (Darmawan & Wurijanto, 2010). IDF menunjukkan hubungan ketersediaan sebuah *term* dalam seluruh dokumen. Semakin sedikit jumlah dokumen yang mengandung *term* yang dimaksud, maka nilai IDF semakin besar. Berikut adalah Persamaan Inverse Document Frequency (IDF) :

$$IDF_t = 1 + \log\left(\frac{n}{df}\right) \quad \text{..... (2.2).}$$

Dimana d_t merupakan *frequency* dokumen dari *term i* atau sama dengan jumlah dokumen yang mengandung *term i* dan *n* adalah total dokumen di dalam *database*.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

3. *Term* Frequency-Inverse Document Frequency (TF.IDF).

Term Frequency (TF) merupakan algoritma yang menunjukan banyaknya suatu kata muncul dalam sebuah dokumen. Sedangkan *Inverse Document Frequency* (IDF) menunjukkan banyaknya dokumen yang mengandung suatu kata dalam satu segmen publikasi. Jadi algortima TF.IDF adalah suatu algoritma yang berdasarkan nilai statistic menunjukkan kemunculan suatu kata dalam dokumen.

Rumus yang digunakan untuk menyatakan bobot (*w*) masing-masing dokumen terhadap kata kunci adalah :

$$W_{d,t} = tf_{d,t} \text{ x } IDF_t \quad \text{..... (2.3).}$$

Dimana *d* adalah dokumen ke-*d*, *t* adalah kata ke-*t* dari kata kunci dan $W_{d,t}$ adalah bobot dokumen ke-*d* terhadap kata ke-*t*.

2.4.3 Latent Semantic Indexing (LSI)

Menurut Froud (2013), *Latent Semantic Indexing* (LSI) adalah metode *indexing* pada *information retrival* (IR) yang menggunakan teknik *singular value decomposition* (SVD) untuk mengidentifikasi makna semantik kata-kata berdasarkan pola dan hubungan antara istilah dan konsep-konsep yang terkandung dalam koleksi teks. Tujuan Utama LSI adalah untuk meningkatkan efektifitas dari sistem IR dengan mengembalikan dokumen yang lebih relevan terhadap *query* pengguna dengan memanipulasi matriks *term-document* dengan menggunakan aljabar linear SVD Matriks asli biasanya sangat besar bagi sumber daya komputasi yang tersedia.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

$$A = USV^T \quad (2.4).$$

Sebuah matrik M yang merupakan frekuensi *term* pada dokumen berukuran m x n dengan *rank* r dapat didekomposisi dengan SVD menjadi U, ∑, V. Hasil SVD adalah matriks U berukuran m x k dan matriks V berukuran n x k. Kedua matriks tersebut mempunyai kolom kolom orthogonal. Sedangkan ∑ adalah matriks diagonal, di mana semua elemen selain diagonalnya bernilai 0. Elemen diagonal dari matriks ∑ disebut sebagai *singular values* dari matrik M (Anand & Jeffrey, 2011).

2.4.4 *Cosine Similarity*

Cosine similarity adalah *similarity measure* yang umum digunakan dalam *information retrieval* (IR) dan merupakan ukuran sudut antara *vector* dokumen (Imbar, Ayub, Rehatta, & Adelia, 2014). Tiap vektor tersebut merepresentasikan setiap kata dalam setiap dokumen (teks). Ukuran ini menghitung nilai *cosinus* sudut antara dua *vector*.

Perhitungan kemiripan menghasilkan bobot dokumen yang mendekati nilai 1 atau menghasilkan bobot dokumen yang lebih besar dibandingkan dengan nilai yang dihasilkan dari perhitungan *inner product*. Persamaan *Cosine Similarity* adalah sebagai berikut :

$$sim\,(q,d) = \frac{q.d}{|q| * |d|} = \frac{\sum_{i=1}^{t} W_{iq} + W_{ij}}{\sqrt{\sum_{j=1}^{t}(W_{iq})^2} + \sqrt{\sum_{j=1}^{t}(W_{ij})^2}} \quad (2.5).$$

Similarity atau *sim* (q,d_j) antara *query* dan dokumen berbanding lurus terhadap

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

jumlah bobot *query* (*q*) dikali bobot dokumen (d_j) dan berbanding terbalik terhadap akar jumlah kuadrat *q (|q|)* dikali akar jumlah kuadrat dokumen *(|dj|).*

2.4.5 Perhitungan Mean Average *Precision* (MAP)

Precision adalah proporsi dokumen yang terambil oleh sistem adalah relevan. *Average precision* merupakan nilai yang didapatkan dari setiap nilai *precision* item relevan yang dihasilkan. Nilai *mean average precision* (MAP) merupakan nilai rata-rata dari *average precision*. Nilai *precision* untuk *average precision* dihitung dengan memperhatikan urutan item yang diberikan oleh sistem, sehingga nilai *precision* diberikan untuk setiap item yang dihasilkan oleh sistem (Parwita, 2015)

Maka pada perhitungan MAP merupakan rata-rata dari dokumen relevan yang berhasil ter-*retrieval* berdasarkan *query*, dokumen yang tidak relevan bernilai 0. Persamaannya adalah sebagai berikut :

$$MAP\ (Q) = \frac{1}{|Q|}\sum_{j=1}^{|Q|}\frac{1}{m}\sum_{k=1}^{mj} Precision\ (Rjk) \quad \text{..... (2.6).}$$

Dimana Q adalah banyaknya *query* atau kumpulan *query* yang di inputkan. R adalah item relevan yang dihasilkan oleh sistem. m adalah jumlah item relevan yang dihasilkan dari *query*. Nilai MAP antara 0 hingga 1. Dan dikatakan baik jika suatu sistem menghasilkan nilai mendekati 1.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

BAB 3

METODOLOGI PENELITIAN

Menjelaskan bagaimana tahapan-tahapan dan mekanisme yang dilakukan dalam penelitian ini. Dengan adanya metodologi penelitian ini maka diharapkan akan memberikan petunjuk tentang pemecahan masalah yang telah dirumuskan.

3.1 Analisis Sistem

Analisis sistem dapat di definisikan sebagai penguraian dari suatu sistem yang utuh ke dalam bagian-bagian kompenennya dengan maksut untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan yang terjadi dan kebutuhan yang diharapkan sehingga dapat di usulkan perbaikan-perbaikan. Hasil dari tahapan dapat menggambarkan sistem yang telah dipelajari dan diketahui bentuk permasalahan serta rancangan sistem baru yang akan dibuat dan dikembangkan.

Data artikel diambil dari 3 website yang dipilih yakni aladokter.com, dokter.id, dan doktersehat.com. Secara garis besar, terdapat 2 metode pengumpulan data yakni metode *scraping* dan metode manual. Pada metode *scrapping*, pengambilan data dilakukan dengan algoritma yang mengambil semua elemen yang terdapat pada *website-website* yang ditentukan, yang tentunya akan memiliki lebih banyak *noise*. Lalu pada metode manual, pengambilan data dilakukan dengan menyalin artikel-artikel dalam website kedalam *file* berformat *textfile*. Hal ini tentunya memiliki *noise* yang sangat lebih sedikit. Secara tidak langsung juga mempengaruhi relevansi hasil pencarian. Metode *scraping* akan diuraikan pada subbab selanjutnya. Dan akan dilakukan pengujian dengan sampel, hasil terbaik

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

antara keduanya akan digunakan sebagai metode untuk melakukan pengambilan data.

Perangkingan dokumen dengan metode pembobotan TF.IDF telah umum digunakan seperti pada penelian dari Maarif (2015) dan Harjanto, Endah, & Bahtiar (2012). Metode tersebut hanya melakukan pembobotan term berdasarkan frekuensi kemunculannya pada dokumen. Hal tersebut tentunya kurang optimal dan berpengaruh pada relevansi pada artikel yang ter-retrieved. Untuk meningkatkan relevansi terhadap hasil pencarian tersebut maka dibutuhkan metode untuk mempertimbangkan hubungan semantic antar *term*. Maka digunakan metode *Latent Semantic Indexing* (LSI) yang mampu mengidentifikasi makna semantik antar *term* berdasarkan pola hubungan yang ada pada koleksi dokumen/artikel. Sifat utama LSI adalah kemampuannya untuk membangun hubungan antara istilah yang muncul dalam konteks yang serupa (Wahib, Pasnur, Santika, & Arifin, 2015)

Terdapat beberapa metode dalam hal perhitungan *similarity*. Dalam penelitian oleh Wahyuni & dkk (2017) terdapat 3 jenis perhitungan *similarity* yakni *cosine similarity, jaccard similarity dan k-nearest neighbor* (KNN). Penelitian dilakukan percobaan sebanyak 33 kali *query* yang berbeda yang menghasilkan nilai kemiripan dari *cosine similarity* adalah 41%, metode *jaccard similarity* menghasilkan nilai 19% dan KNN menghasilkan 40%. Hal tersebut karena metode *cosine similarity* mempunyai konsep normalisasi panjang vektor data dengan membandingkan N-gram yang sejajar satu sama lain dari 2 pembanding. Oleh sebab itu, pada penilitian ini akan menggunakan metode *cosine similarity* untuk perhitungan *similarity*-nya.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Berdasarkan uraian diatas, sistem yang akan dikembangkan dalam penelitian ini adalah sistem temu kembali (*information retrieval system)* yang berdasarkan pembobotan *term frequency inverse document frequency* (TF.IDF) dan *latent semantic indexing* (LSI) berbasis pada *cosine similarity* antara *query* dengan data dokumen artikel yang diolah.

3.2 Perancangan Sistem

Perancangan sistem digunakan dalam penggambaran sistem yang akan dibuat. Bagaimana sistem akan dibangun, berjalan, dan sekaligun pengolahan datanya.

Penelitian ini adalah mengenai perangkingan artikel kesehatan berbahasa Indonesia dengan berbasis pada nilai perhitungan *Cosine Similarity* yang berdasarkan pada pembobotan *Term Frequecny-Inverse Document Frequency* (TF.IDF) dan *Latent Semantic Indexing* (LSI) pada sebuah sistem *information retriviel* (IR). Dalam sistem ini, *users* melakukan pencarian sebuah kasus dengan memasukkan kata kunci *(query),* perangkingan artikel dilakukan berdasarkan perhitungan *similarity* antara *vector* indeks artikel dan *query*. Outputnya berupa perangkingan artikel-artikel secara *descending* sesuai dengan nilai *cosine similarity*.

Proses pencarian artikel dimulai dari *query* yang dimasukkan *users* akan di proses dengan menggunakan *text mining* pada tahap *preprocessing* yang meliputi *case folding, tokenizing, filtering, stemming*. Tahap *preprocessing* digunakan untuk mengubah data teks menjadi data numerik sehingga bisa dikomputasikan. Hasil *preprocessing* ini adalah berupa bentuk token atau *term* yang paling sederhana.

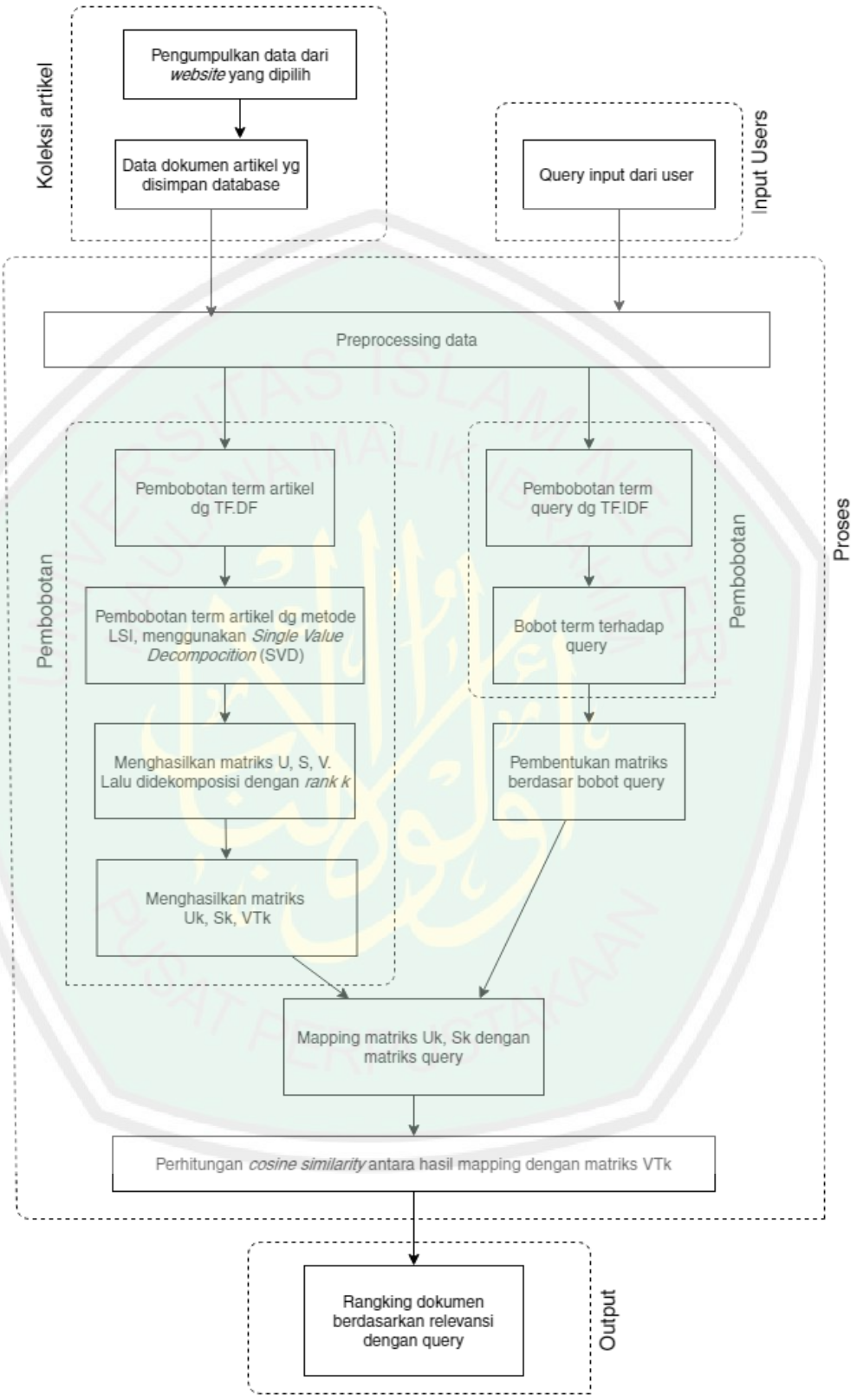
CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Kemudian token atau *term* tersebut akan diberikan bobot atau nilai dengan menggunakan metode *Term Frequecny-Inverse Document Frequency* (TF.IDF).

Banyaknya *dataset* artikel merupakan representasi dari kolom matriks, sedangkan jumlah *term* yang terbentuk dari tahap *preprocessing* merupakan representasi dari baris matriks, serta frekuesi kemunculan kata dalam artikel tertentu merupakan nilai matriks yang terletak sesuai dengan letak *term* dan artikel pada baris dan kolom terkait. Lalu matriks tersebut dilakukan dekomposisi menggunakan *single value decompotition* (SVD) berdasarkan pada *rank-k* pada tahapan LSI untuk mencari hubungan semantic antar *term.*

Lalu dilakukan perhitungan kemiripan dengan motode *cosine similarity* yang meghasilkan nilai *similarity* antara *query* dengan artikel. Dan tahap paling akhir adalah perangkingan kesesuaian antara *query* dengan artikel berdasarkan nilai *cosine similarity* dari yang tertinggi hingga terendah untuk ditampilkan kembali pada *users*.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

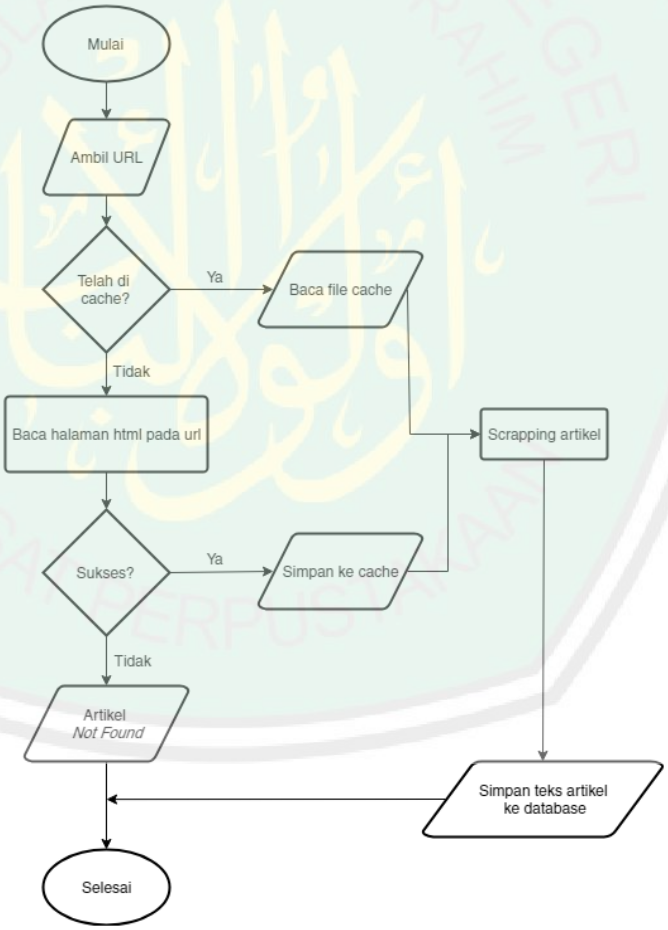


Gambar 3. 1 Diagram Rancangan Sistem.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

3.2.1 Pengumpulan Data

Data yang digunakan dalam penelitian ini adalah berupa artikel - artikel kesehatan berbahasa Indonesia yang diambil dari *website-website* kesehatan terpilih, diantaranya http://www.alodokter.com/, https://www.dokter.id/, https://doktersehat.com/. Cara pengambilan datanya menggunakan salah satu teknik *web scraping* yakni Regresi Linier. Data artikel hasil *scrap* kemudian diolah dengan teknik *text mining* lalu di simpan dalam suatu *database* dengan format *file* yang sifatnya plain text. Alur *web scraping* ditunjukkan pada gambar 3.2.



Gambar 3. 2 Flowchart Web Scrapping

(Sumber: Wibisono & Utomo, 2013)

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

3.2.2 Preprocessing Data

Tahap paling awal dalam pengolahan data adalah tahap *preprocessing*. Tahap ini bisa dikatakan penting karena hasil dari tahap ini akan sangat mempengaruhi terhadap hasil dari penelitian. Dalam tahap ini dilakukan pengolahan data dari yang semi- tersetruktur hasil *scrap* menjadi data yang siap pakai dalam implementasi metode TF.IDF dan LSI serta *Cosine Similarity*. Tahapannya digambarkan pada gambar 3.3.

Gambar 3. 3 Alur Preprocessing

Misalkan terdapat suatu artikel singkat :

> Aspirin merupakan obat penurun demam. Tetapi aspirin juga dapat digunakan sebagai cara menghilangkan jerawat di punggung. Aspirin mengandung anti inflamasi yang bisa membantu mencegah pori-pori kulit tersebut.

Gambar 3. 4 Contoh Artikel Kesehatan

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

1. Case Folding

Pada tahap ini dilakukan proses konversi teks manjadi suatu bentuk standart (*Lowercast)* karena Tidak semua data teks yang digunakan konsisten dalam penggunaan huruf capital. Case Folding adalah proses penyamaan *case* dalam sebuah dokumen. Sebagai mana contoh diatas, maka :

aspirin merupakan obat penurun demam. tetapi aspirin juga dapat digunakan sebagai cara menghilangkan jerawat di punggung. aspirin mengandung anti inflamasi yang bisa membantu mencegah pori-pori kulit tersebut.

Gambar 3. 5 Hasil Proses Case Folding

2. Tokenisasi

Pada tahap ini dilakukan pemotongan input *string* berdasarkan tiap kata penyusunnya. Token seringkali disebut sebagai istilah (*term*) atau kata. Sebagai mana contoh diatas, maka :

aspirin merupakan obat penurun demam	tetapi aspirin juga dapat digunakan sebagai cara menghilangkan jerawat di punggung	aspirin mengandung anti inflamasi yang bisa membantu mencegah pori-pori kulit tersebut

Gambar 3. 6 Hasil Proses Tokenisasi

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

3. Filtering atau Stopword Removal

Filtering adalah tahap mengambil kata-kata penting dari hasil token. Tahap filtering dalam penelitian ini akan menggunakan *algoritma stopword.* Algoritma *stopword* merupakan algoritma yang digunakan untuk mengeliminasi kata-kata yang tidak deskriptif. Proses *filtering* menggunakan daftar *stoplist* bahasa Indonesia yang berisi kata-kata seperti: ada, yang, ke, dan lain sebagainya. Sebagai mana contoh diatas, maka :

aspirin obat penurun demam	aspirin menghilangkan jerawat punggung	aspirin mengandung anti inflamasi membantu mencegah pori-pori kulit

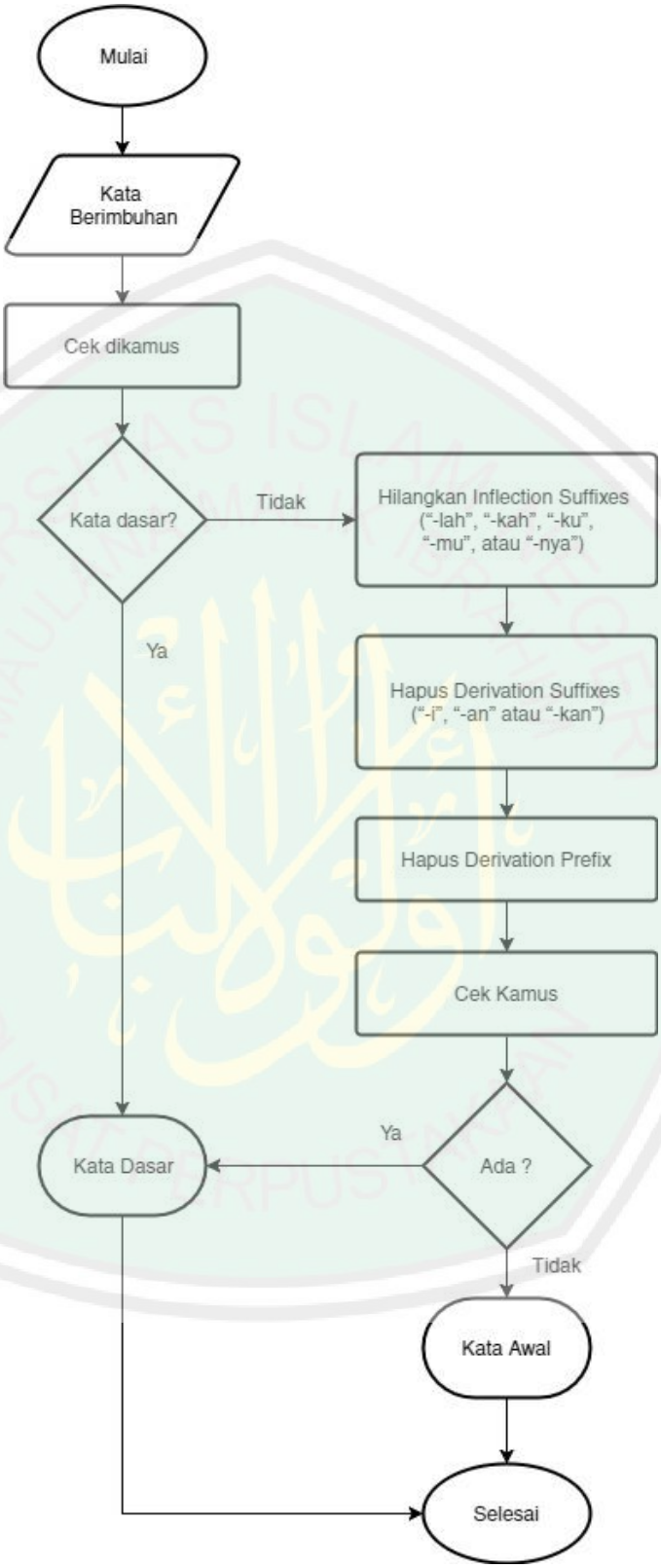
Gambar 3. 7 Hasil Proses *Fitering* dan *Stopword Removal*

4. Stemming

Pada dasaranya stemming merupakan pencarian *root* kata dari tiap kata hasil filtering. Proses stemming dilakukan dengan cara menghilangkan semua imbuhan (*affixes*) baik yang terdiri dari awalan (*prefixes*), sisipan (*infixes*), akhiran (*suffixes*) dan *confixes* (kombinasi dari awalan dan akhiran) pada kata turunan.

Dalam penelitian ini, tahap *stemming* menggunakan algoritma *Nazief & Adriani Stremmer* (2009). Berikut *flowchart* dari proses *stemming* :

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG



Gambar 3. 8 *Flowchart Algoritma Nazief & Adriani Stremmer*

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Sebagai mana contoh diatas, maka :

aspirin obat turun demam	aspirin hilang jerawat punggung	aspirin kandung anti inflamasi bantu cegah pori-pori kulit

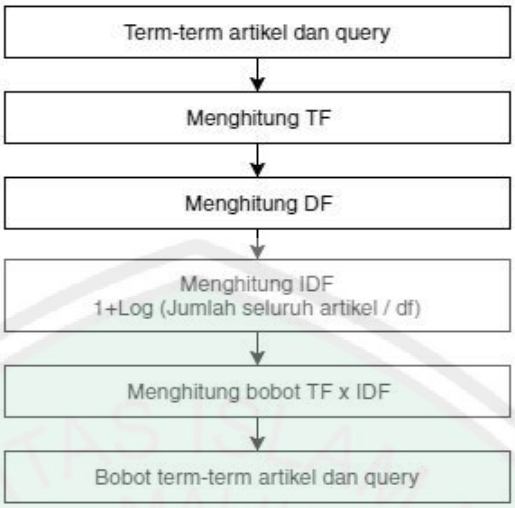
Gambar 3. 9 Hasil Proses *Stemming*

3.2.3 Pembobotan dengan *Term* Frequency Inverse Document Frequency (TF.IDF)

Penggunaan *Term Frequency* (TF) sering dijumpai pada sistem *information retriviel* (IR). Dalam kaitanya dengan fungsi *recall* dan *precision*. Ternyata, penggunaan TF saja hanya mampu memenuhi fungsi *recall*, tidak cukup mampu untuk memenuhi fungsi *precision*. *Precision* yang tinggi mengisaratkan kemampuan untuk membedakan suatu dokumen dengan dokumen yang lain untuk mencegah retrieval yang tidak diinginkan.

Inverse Document Frequency (IDF) digunakan dalam meningkatkan fungsi *precision*. Hal tersebut telah di teliti oleh Spärck Jones dan menunjukkan bahwa penggunaan IDF akan menghasilkan performa *retrieval* yang lebih efektif jika dibandingkan dengan penggunaan frekuensi *term* saja. Kemudian untuk mengkombinasikan metode TF dan IDF, dengan mempertimbangkan frekuensi inter-dokumen dan frekuensi intradokumen dari suatu *term*. Alur prosesnya ditunjukkan pada gambar 3.10.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG



Gambar 3. 10 Alur pembobotan TF.IDF.

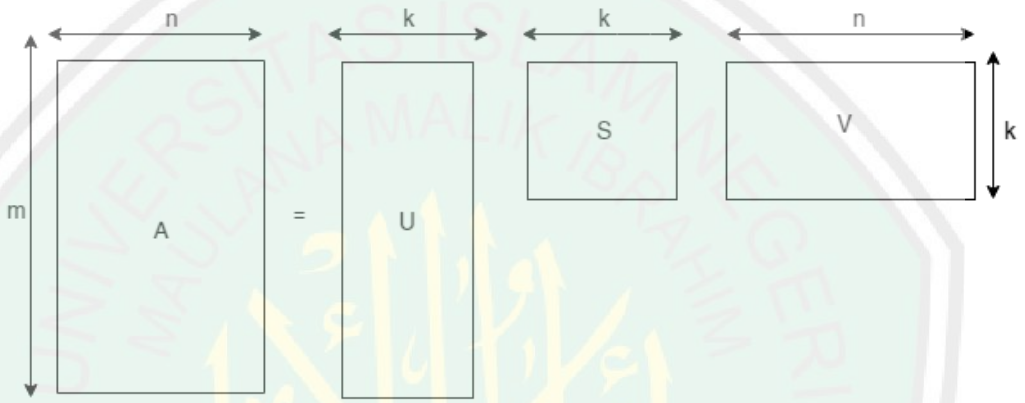
3.2.4 Pembobotan *Latent Semantic Indexing* (LSI)

Setelah *term-term* dilakukan pembobotan dan dibentuk matriks. Kemudian matriks tersebut dilakukan dekomposisi. Pada LSI, dekomposisi tersebut menggunakan *single value decomposition* (SVD) untuk memodelkan relasi asositif antara *term*. Secara konsep, SVD menerima kumpulan data berdimensi tinggi dan variable tinggi serta menguranginya ke dalam ruang dimensi yang berukuran lebih kecil. Proses SVD menghasilkan 3 matriks yakni U, S dan V^T. Lalu biasanya dilakukan tahap pengurangan ukuran dimensi, sebanyak k. Penentuan nilai k yang paling optimal memang perlu dilakukan percobaan. Dari situ, ketiga matriks tersebut akan menjadi U_k, S_k, dan V^T_k.

Ilustrasinya adalah sebagai berikut, matriks A merupakan frekuensi *term* pada dokumen berukuran m x n, dimana *m* menginisialisasikan jumlah *term* yang diproses dan *n* menginisialisasikan jumlah kolom atau dokumen artikel yang terlibat. Menghasilkan matriks U, S, V. Kemudian dengan nilai *k* dapat

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

didekomposisi dengan SVD menjadi U^k, S^k, V^k seperti pada Gambar 3.11. Hasil SVD adalah matriks Uk berukuran m x k dan matriks V^k berukuran n x k. Kedua matriks tersebut mempunyai kolom-kolom orthogonal. Matriks Sk merupakan matriks diagonal dimana elemen lain selain diagonalnya bernilai 0. Elemen diagonal dari matriks S disebut sebagai *singular values* dari matriks A.



Gambar 3. 11 Ilustrasi Dekomposisi dengan SVD

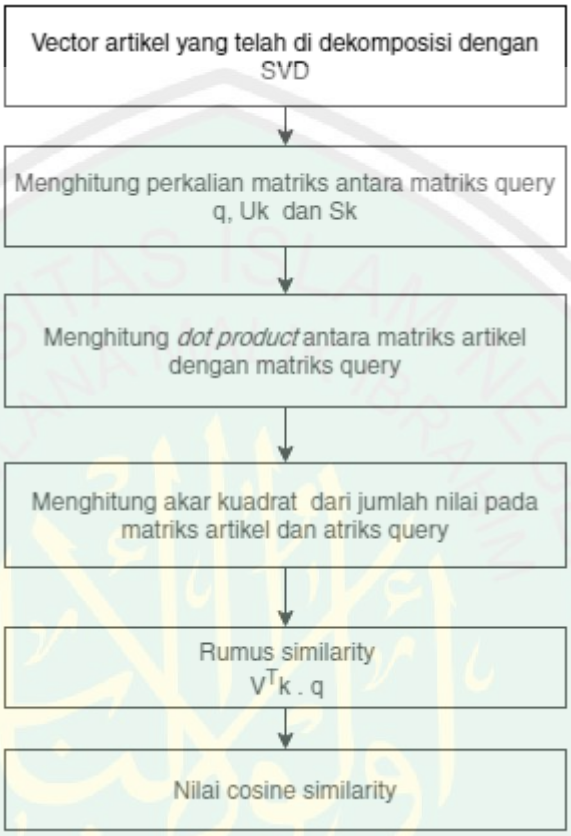
3.2.5 Perhitungan Kemiripan dengan *Cosine Similarity*

Setelah suatu data teks di bobotkan dengan TF.IDF dan di hitung hubungan semantic antar *term* dengan LSI. Hasilnya berupa *vector query* dan *vector* artikel lalu dilakukan perhitungan koefesien kemiripan antar artikel menggunakan *similarity measure*. Salah satu fungsi *similiarity measure* yang paling populer adalah *Cosine Similarity*. Secara konsep, *Cosine Similarity* digunakan untuk menghitung nilai *cosinus* sudut antara dua *vector*. Dalam penelitian ini, dokumen dan *query* dipresentasikan kedalam bentuk *vector*.

Hasil dari proses LSI menghasilkan output berupa matriks U_k, S_k, dan V^T_k. Matriks U_k dan S_k digunakan untuk menghitung bobot *query* sedangkan nilai V^T_k

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

digunakan untuk menghitung nilai *cosine similarity*. Alur prosesnya ditunjukkan pada gambar 3.12.



Gambar 3. 12 Alur Perhitungan *Cosine Similarity*

3.2.6 Output

Hasil dari penelitian ini adalah perangkingan artikel-artikel kesehatan berdasarkan pada nilai *Cosine Similarity*, yang diperoleh dari kesesuaian antara dataset dengan *query* dari pengguna. Semakin besar nilai *Cosine Similarity* maka semakin tinggi letak rangkingnya. Kemudian yang ditampilkan kepada pengguna adalah judul-judul dari artikel yang terangking tersebut.

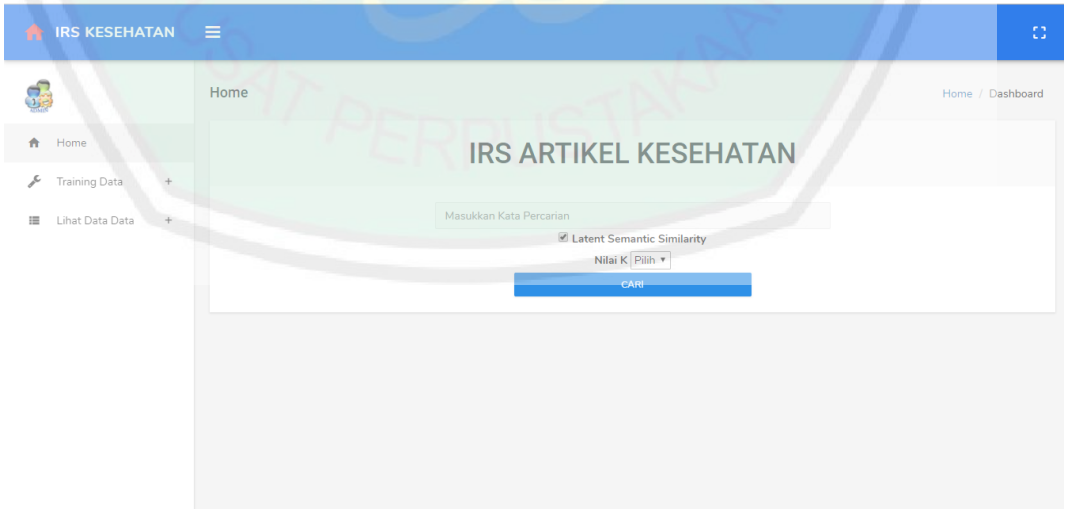
CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

3.2.7 Desain Interface

Implementasi *interface* merupakan penjelasan tentang *interface* aplikasi yang telah dibuat. Pada penelitian ini terdapat beberapa *interface* yang dapat digunakan untuk keperluan pengguna yakni *interface* home, *interface* untuk preprocessing, *interface* latent semantic indexing, dan *interface* lihat data hasil proses. Setiap *interface* juga digunakan untuk keperluan yang berbeda-beda tetapi memiliki keterkaitan antara satu dan lainya. Tampilan aplikasi adalah sebagai berikut :

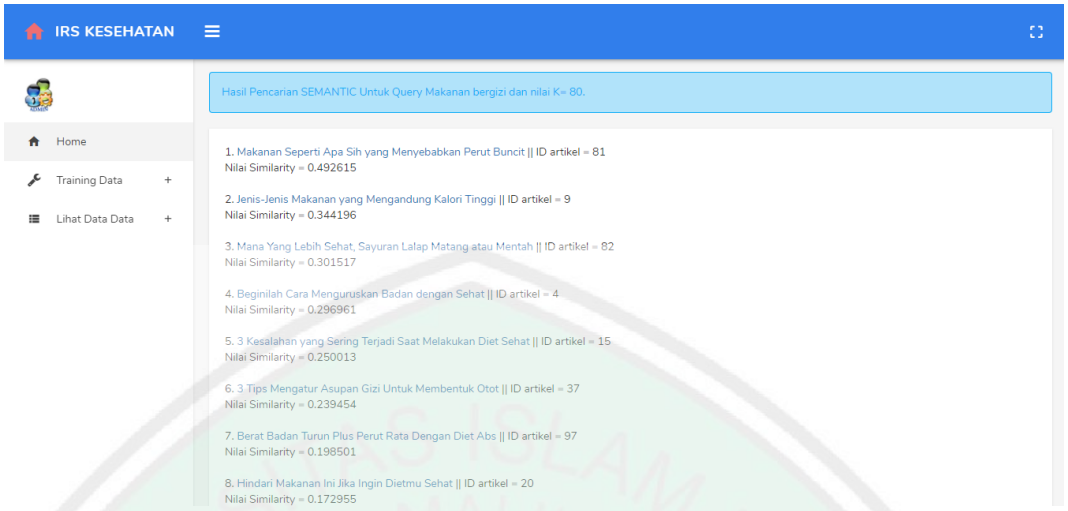
1. *Interface* Home

Pada *interface* ini digunakan untuk melakukan pencarian, dimana terdapat fitur input bertipe text untuk memasukkan *query*, fitur checklist untuk memilih apakah ingin menggunakan metode *Latent Semantic Indexing* (LSI) atau tanpa LSI, fitur pilihan menu untuk memilih nilai k yang digunakan untuk proses LSI, dan tombol cari untuk mulai melakukan pencarian. *Interface* home di tunjukkan pada gambar 3.13 dan gambar 3.14 untuk *interface* hasil pencarian.



Gambar 3. 13 Interface Home

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

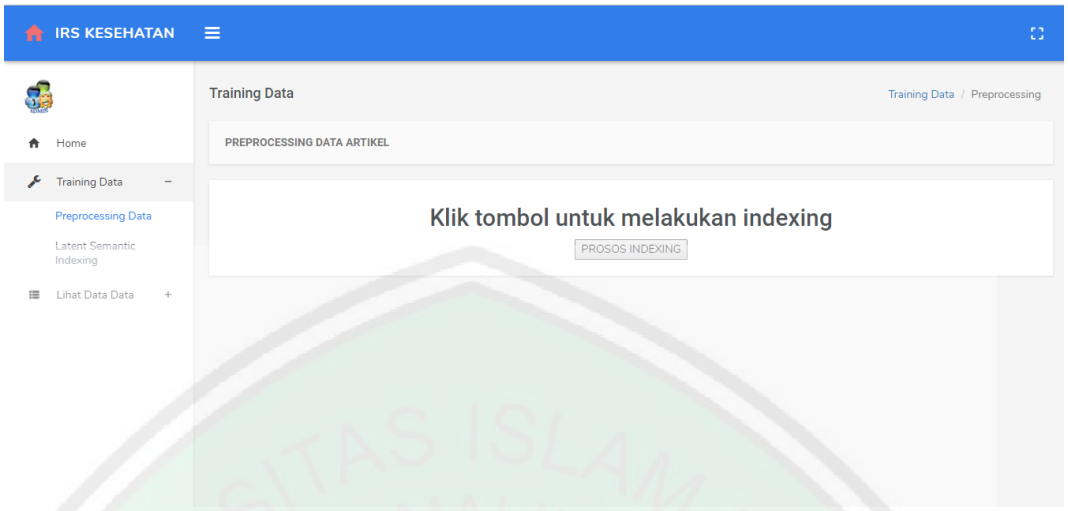


Gambar 3. 14 Interface untuk Hasil Pencarian

2. *Interface* Training Data

Pada menu *Interface* training data terdapat submenu *preprocessing* dan *latent semantic indexing*. Submenu *preprocessing* digunakan untuk training data dalam *preprocessing*. *Interface* terdapat fitur tombol indexing untuk mulai malakukan proses. Dan submenu *latent semantic indexing* digunakan untuk melakukan training data dalam proses *latent semantic indexing* (LSI). *Interface* terdapat fitur input text nilai k yang digunakan untuk input dari proses *latent semantic indexing* (LSI). *Interface preprocessing* di tunjukkan pada gambar 3.15 dan gambar 3.16 untuk *interface latent semantic indexing*.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG



Gambar 3. 15 Interface untuk Preprocessing Data

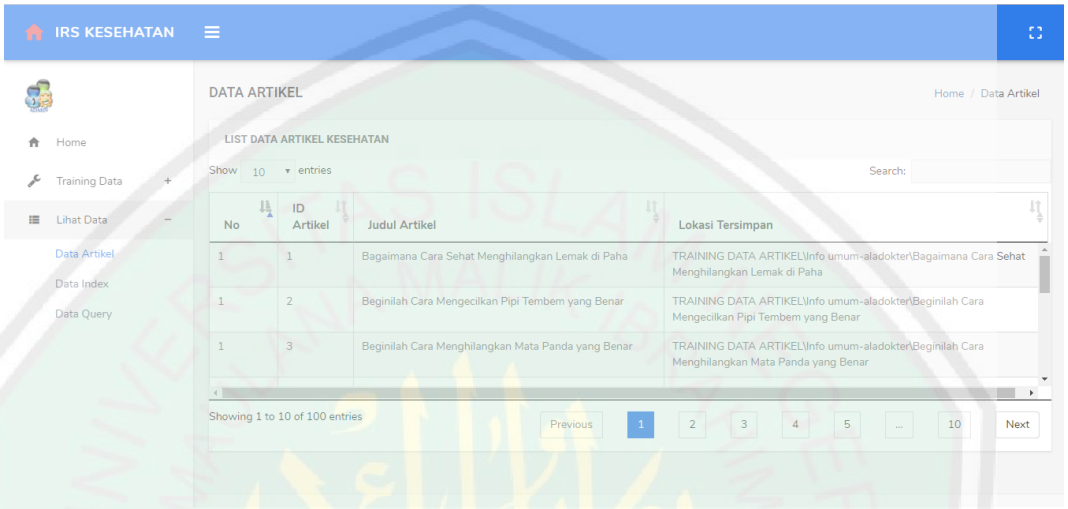
Gambar 3. 16 Interface untuk Latent Semantic Indexing

3. *Interface* Lihat Data

Pada menu *Interface* lihat data terdapat submenu data artikel, data index dan data *query*. Submenu data artikel digunakan untuk melihat semua artikel yang terlibat dalam penelitian. *Interface* terdapat fitur search untuk melakukan pencarian artikel yang diinginkan *users*. Submenu data index digunakan untuk melihat semua index *term* hasil tahap *preprocessing*. Submenu data *query*

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

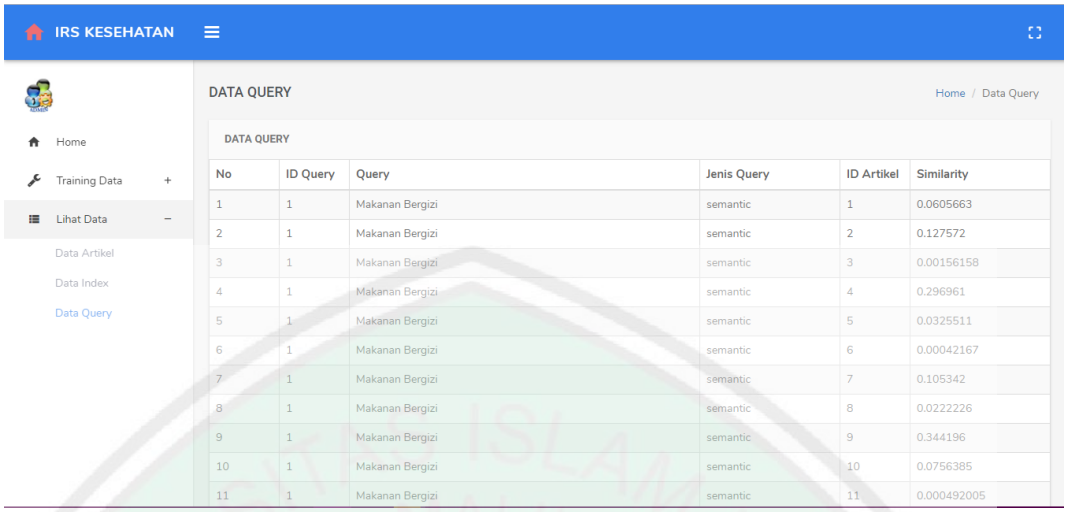
digunakan untuk melihat semua *query* yang digunakan untuk uji coba aplikasi ini. *Interface* data artikel di tunjukkan pada gambar 3.17, *interface* data index di tunjukkan pada gambar 3.18 dan gambar 3.19 untuk *interface* data *query*.



Gambar 3. 17 Interface untuk Lihat Data Artikel

Gambar 3. 18 Interface untuk Lihat Data Index Term

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG



Gambar 3. 19 Interface untuk Lihat Data Query

3.3 Skenario Uji coba

Pada subbab ini menjelaskan bagaimana melakukan pengujian dan mengevaluasi hasil pengujian dari penerapan metode pembobotan TF.IDF dan LSI terhadap masalah yang diangkat. Pada penelitian ini, pengujian dilakukan dengan menggunakan *Mean Average Precision* (MAP). Hasil dari penelitian yakni berupa nilai *similarity* dari *query* yang di uji coba dengan tiap-tiap dokumen artikel. Nilai *similarity* tersebut kemudian akan di rangking berdasarkan nilai terbesar sampai terkecil. Evaluasi sistem dengan MAP dilakukan berdasarkan pada ranking hasil setiap pencarian.

Dalam MAP, *precision* dihitung dengan membagi antara jumlah dokumen relevan yang *retrieved* pada *rank-k* dengan urutan jumlah dokumen yang *retrieved* pada *rank-k*. Penentuan dokumen artikel yang relevan didapat dari pakar yang merupakan seorang user yang memiliki latar belakang tentang masalah kesehatan. Kemudian perhitungan *Mean Precision* (MP) dilakukan dengan membagi *precision*

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

dari *query* pada *rank-k* dengan jumlah *query*. Lalu tahap terakhir yakni MAP yang diperoleh dengan mambagi *precision* dari MP masing-masing *query* dengan *rank-k* dokumen arikel yang terlibat.

Data yang digunakan berupa 100 artikel kesehatan. Kemudian pada proses SVD, di tentukan nilai *k* yakni 80. Untuk *query* terdapat 20 kalimat yang di kategorikan menjadi 2 kategori yakni *short* dan *long*. Tujuannya adalah sebagai perbandingan pada perhitungan MAP-nya. Perhitungan MAP dilakukan pada *rank-k*=15 dan *rank-k*=10. Data *query* tersebut adalah sebagai berikut :

Tabel 3. 1 List Data *Query* untuk Uji Coba

No	*Query*	Kategori
1	Makanan bergizi	Short
2	Penyebab mulut bau	Short
3	Sariawan	Short
4	Manfaat tidur siang	Short
5	Kanker	Short
6	Mencegah diabetes	Short
7	Mengobati sakit gigi	Short
8	Diet yang ampuh	Short
9	Gigi dan mulut	Short
10	Gejala penyakit jantung	Short
11	Mencegah Serangan Jantung sejak dini	Long
12	Jerawat tidak kunjung sembuh	Long
13	Jenis makanan yang mengandung lemak	Long
14	Menghilangkan mata panda	Long
15	Menghilangkan komedo di wajah	Long
16	Menyembuhkan Flek hitam bekas jerawat	Long
17	Makanan yang mempunyai kalori tinggi	Long
18	Tips sehat berolahraga di pagi hari	Long
19	Tipe olahraga untuk menurunkan berat badan	Long
20	Cara mudah mengobati sakit kepala	Long

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

3.4 Lingkungan Pengembangan Sistem

Lingkungan pengembangan sistem pada penelitian ini adalah sebagai berikut:

4.1.1. Spesifikasi Perangkat Keras

1. Laptop Acer Aspire V5-471G
2. *Processor* Intel® Core™ i5-3337U CPU @ 1.80GHz dan NVIDIA GeForce 710M
3. RAM 4.00 GB

4.1.2. Spesifikasi Perangkat Lunak

1. Windows 10 Pro 64-bit Operating System
2. Sublime Text 2
3. *Database* MySQL

Sistem yang dibangun aplikasi berbasis web dengan menggunakan bahasa pemrograman *PHP* dan *database* MySQL.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

BAB 4

PEMBAHASAN

Pada bagian ini akan menjelaskan mengenai implementasi dari setiap langkah yang telah di jelaskan di bab sebelumnya dan memaparkan hasil uji coba yang telah dilakukan sesuai dengan scenario pengujian yakni dengan membandingkan tingkat akurasi dan presisi antara metode yang di gunakan dalam penelitian ini dengan metode yang telah digunakan dipenelitian sebelumnya. Kemudian pada bagian akhir bab akan dipaparkan evaluasi dan pembahasan dari hasil percobaan yang diperoleh. Input data dari penelitian ini adalah berupa artikel kesehatan, kemudian dilakukan proses *preprosesing*, pembobotan, perhitungan *semantic* dan perhitungan *similarity*. Dan outputnya adalah berupa perangkingan artikel kesehatan sesuai dengan kata kunci (*query*) yang dilakukan oleh pengguna.

4.1 Implementasi

Implemantasi digunakan untuk mengintregasikan metode yang digunakan kedalam langkah-langkah yang terencana sehingga diperoleh hasil uji coba yang dapat diukur tingkat akurasi dan presisinya. Metode yang digunakan di implementasikan menggunakan bahasa pemrograman PHP 5.6 dengan IDE yang digunakan adalah Sublime Text 2. *Database Server* menggunakan MySQL, dan package *Single Value Decomposition* (SVD) dan Stemming. Aplikasi di bangun menggunakan *Operating System* Windows 10 dengan spesifikasi *Intel Core* i5 dan memory 4 GB.

Langkah-langkah yang terencana di susun rapi pada bab 3. Langkah-langkah tersebut kemudian dijalankan sehingga dapat diperoleh nilai yang bisa digunakan

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

untuk mengukur tingkat keberhasilan penelitian. Pada bagian ini di jelaskan hasil implementasi dari setiap langkah-langkah beserta potongan-potongan *source code* yang penting.

4.1.1 Pembuatan Index Artikel pada Tahap Implementasi

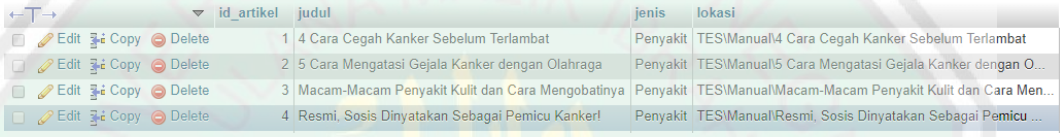
Langkah-langkah pembuatan index artikel meliputi beberapa tahap. Tahap pertaman adalah pengambilan data artikel kesehatan pada *website-website* yang telah di tentukan yakni (1) http://www.alodokter.com/, (2) https://www.dokter.id/, dan (3) https://doktersehat.com/ yang akan digunakan sebagai isi dari *database* yang akan di proses.

Tahap kedua yakni *preprocessing* data, yang meliputi *case folding*, *tokenisasi*, *stopword removal*, dan stemming. Output dari tahap ini adalah berupa index *term-term* dari seluruh artikel yang terlibat dalam *preprocessing*. Kemudian dilakukan tahap pembobotan dengan metode TF.IDF pada setiap *term*.

Perolehan data artikel dilakukan dengan metode manual yakni dengan mencari sejumlah artikel yang diperlukan, kemudian di simpan dalam pada *file* berformat *textfile*. Karena apabila perolehan data dilakukan dengan metode *scraping*, data artikel yang didapatkan terlalu banyak *noise* sehingga juga akan mempengaruhi hasil akhir penelitian. Data sampel hasil pengambilan artikel ditunjukkan pada gambar 4.2 dan 4.3. Hasil perbandingan, perolehan data dengan metode manual terbukti lebih optimal dengan hasil *preprocessing* dari 5 dokumen artikel yang diproses menghasilkan sebanyak 782 index *term*, sedangkan cara *scraping* menghasilkan 5798 index *term* yang sebagian besar merupakan *noise*.

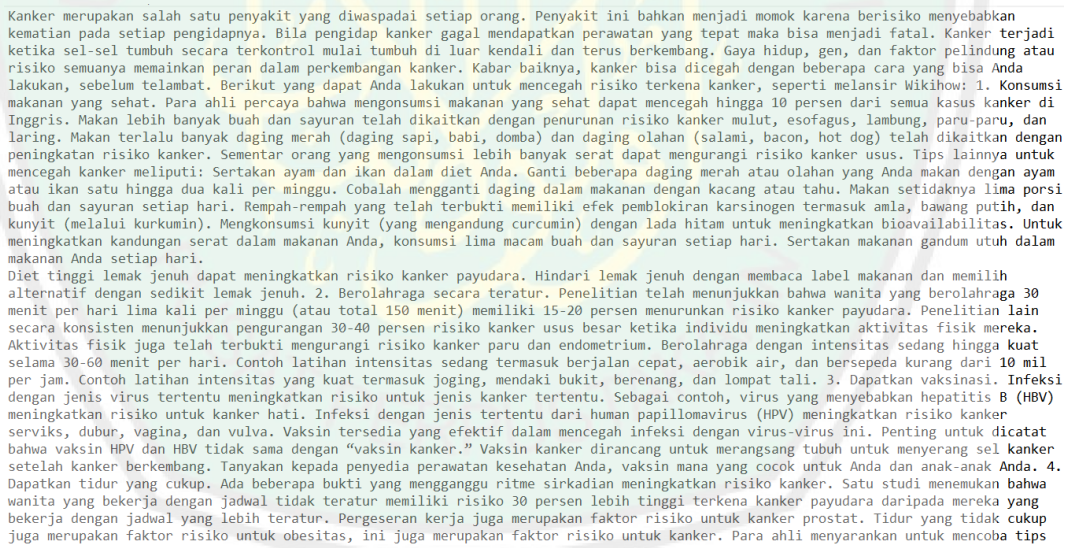
CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Selain itu, metode *scraping* membutuhkan waktu *proprocessing* yang relative lebih lama yakni sekitar 100 menit untuk 5 dokumen artikel yang di uji coba, sedangkan metode manual hanya membutuhkan waktu 10 menit untuk 5 dokumen artikel yang di uji coba. Dalam hal nili semilarity, metode manual juga terbukti menghasilkan nilai yang lebih tinggi dari pada metode *scraping*. Perbandingan nilai *similarity* antara perolehan data dengan metode manual dengan perolehan data dengan metode *scraping* ditunjukkan pada tabel 4.1.



	id_artikel	judul	jenis	lokasi
Edit Copy Delete	1	4 Cara Cegah Kanker Sebelum Terlambat	Penyakit	TES\Manual\4 Cara Cegah Kanker Sebelum Terlambat
Edit Copy Delete	2	5 Cara Mengatasi Gejala Kanker dengan Olahraga	Penyakit	TES\Manual\5 Cara Mengatasi Gejala Kanker dengan O...
Edit Copy Delete	3	Macam-Macam Penyakit Kulit dan Cara Mengobatinya	Penyakit	TES\Manual\Macam-Macam Penyakit Kulit dan Cara Men...
Edit Copy Delete	4	Resmi, Sosis Dinyatakan Sebagai Pemicu Kanker!	Penyakit	TES\Manual\Resmi, Sosis Dinyatakan Sebagai Pemicu ...

Gambar 4. 1 Sampel Dokumen Artikel dalam *Database*



Kanker merupakan salah satu penyakit yang diwaspadai setiap orang. Penyakit ini bahkan menjadi momok karena berisiko menyebabkan kematian pada setiap pengidapnya. Bila pengidap kanker gagal mendapatkan perawatan yang tepat maka bisa menjadi fatal. Kanker terjadi ketika sel-sel tumbuh secara terkontrol mulai tumbuh di luar kendali dan terus berkembang. Gaya hidup, gen, dan faktor pelindung atau risiko semuanya memainkan peran dalam perkembangan kanker. Kabar baiknya, kanker bisa dicegah dengan beberapa cara yang bisa Anda lakukan, sebelum telambat. Berikut yang dapat Anda lakukan untuk mencegah risiko terkena kanker, seperti melansir Wikihow: 1. Konsumsi makanan yang sehat. Para ahli percaya bahwa mengonsumsi makanan yang sehat dapat mencegah hingga 10 persen dari semua kasus kanker di Inggris. Makan lebih banyak buah dan sayuran telah dikaitkan dengan penurunan risiko kanker mulut, esofagus, lambung, paru-paru, dan laring. Makan terlalu banyak daging merah (daging sapi, babi, domba) dan daging olahan (salami, bacon, hot dog) telah dikaitkan dengan peningkatan risiko kanker. Sementar orang yang mengonsumsi lebih banyak serat dapat mengurangi risiko kanker usus. Tips lainnya untuk mencegah kanker meliputi: Sertakan ayam dan ikan dalam diet Anda. Ganti beberapa daging merah atau olahan yang Anda makan dengan ayam atau ikan satu hingga dua kali per minggu. Cobalah mengganti daging dalam makanan dengan kacang atau tahu. Makan setidaknya lima porsi buah dan sayuran setiap hari. Rempah-rempah yang telah terbukti memiliki efek pemblokiran karsinogen termasuk amla, bawang putih, dan kunyit (melalui kurkumin). Mengkonsumsi kunyit (yang mengandung curcumin) dengan lada hitam untuk meningkatkan bioavailabilitas. Untuk meningkatkan kandungan serat dalam makanan Anda, konsumsi lima macam buah dan sayuran setiap hari. Sertakan makanan gandum utuh dalam makanan Anda setiap hari.
Diet tinggi lemak jenuh dapat meningkatkan risiko kanker payudara. Hindari lemak jenuh dengan membaca label makanan dan memilih alternatif dengan sedikit lemak jenuh. 2. Berolahraga secara teratur. Penelitian telah menunjukkan bahwa wanita yang berolahraga 30 menit per hari lima kali per minggu (atau total 150 menit) memiliki 15-20 persen menurunkan risiko kanker payudara. Penelitian lain secara konsisten menunjukkan pengurangan 30-40 persen risiko kanker usus besar ketika individu meningkatkan aktivitas fisik mereka. Aktivitas fisik juga telah terbukti mengurangi risiko kanker paru dan endometrium. Berolahraga dengan intensitas sedang hingga kuat selama 30-60 menit per hari. Contoh latihan intensitas sedang termasuk berjalan cepat, aerobik air, dan bersepeda kurang dari 10 mil per jam. Contoh latihan intensitas yang kuat termasuk joging, mendaki bukit, berenang, dan lompat tali. 3. Dapatkan vaksinasi. Infeksi dengan jenis virus tertentu meningkatkan risiko untuk jenis kanker tertentu. Sebagai contoh, virus yang menyebabkan hepatitis B (HBV) meningkatkan risiko untuk kanker hati. Infeksi dengan jenis tertentu dari human papillomavirus (HPV) meningkatkan risiko kanker serviks, dubur, vagina, dan vulva. Vaksin tersedia yang efektif dalam mencegah infeksi dengan virus-virus ini. Penting untuk dicatat bahwa vaksin HPV dan HBV tidak sama dengan "vaksin kanker." Vaksin kanker dirancang untuk merangsang tubuh untuk menyerang sel kanker setelah kanker berkembang. Tanyakan kepada penyedia perawatan kesehatan Anda, vaksin mana yang cocok untuk Anda dan anak-anak Anda. 4. Dapatkan tidur yang cukup. Ada beberapa bukti yang mengganggu ritme sirkadian meningkatkan risiko kanker. Satu studi menemukan bahwa wanita yang bekerja dengan jadwal tidak teratur memiliki risiko 30 persen lebih tinggi terkena kanker payudara daripada mereka yang bekerja dengan jadwal yang lebih teratur. Pergeseran kerja juga merupakan faktor risiko untuk kanker prostat. Tidur yang tidak cukup juga merupakan faktor risiko untuk obesitas, ini juga merupakan faktor risiko untuk kanker. Para ahli menyarankan untuk mencoba tips

Gambar 4. 2 Sampel Dokumen Artikel dengan Pengambilan Data Manual

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Gambar 4. 3 Sampel Dokumen Artikel dengan Pengambilan Scraping

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Tabel 4. 1 Perbandingan Hasil Pencarian antara Perolehan Manual dengan *Scraping*

Metode Perolehan data dengan *query* = kanker	**Id Artikel**				
	1	**2**	**3**	**4**	**5**
Manual	0.391	0.270	0.0075	0.161	0
Scraping	0.120	0.089	0.025	0.048	0.016

4.1.2 Pengambilan Data dari *Database* pada Tahap Implementasi

Data yang telah di ambil dari beberapa *website* yang telah di sebutkan sebelumya, kemudian di simpan di dalam *database* MySQL yang terdiri dari beberapa tabel yakni tabel artikel, index, kadar, kesehatan, *query*, dan stopword. Tujuannya adalah agar mudah saat di lakukan proses selanjutnya. Proses koneksi ke *database* tersebut terdapat pada *file* bernama koneksi.php, dimana *file* tersebut akan di *include* kan pada setiap proses yang didalamnya berhubungan dengan *database*. Proses koneksi menggunakan bahas pemrogaman PHP karena aplikasi yang dibangun berbasis web.

Terdapat beberapa atribut dan method yang digunakan dalam operasi koneksi ke *database*. Pertama, untuk membuat koneksksi ke *database* membutuhkan penginisialisai *host, database username* dan *password*. Lalu method *mysqli_connect()* yang digunakan membuat koneksi *database*. Kemudian terdapat juga *mysqli_query* yang digunakan untuk menjalankan sebuah *query* pada setiap fungsi managemen basis data yang meliputi *insert, select, update,* dan *delete*. Method *mysqli_fetch_array()* digunakan untuk memanggil data yang sudah terpanggil di method *mysqli_query()*. Dan method *mysqli_nums_row()* yang

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

digunakan untuk menampikan jumlah isi dari *database* yang di panggil di method *mysqli_query()*.

```
[1] $host='localhost';
[2] $user='root';
[3] $pass='';
[4] $database='skripsi';
[5] $conn = mysqli_connect ($host,$user,$pass,$database);
[6] ini_set('display_errors', 1);
```

Gambar 4. 4 Method untuk koneksi ke *database*

4.1.3 Preprocessing Data pada Tahap Implementasi

Setelah data artikel kesehatan berhasil di simpan di *database*. Tahap selanjutnya adalah *preprocessing*. Hasil dari preprocessing ini adalah berupa index *term-term* dari semua artikel kesehatan yang terproses. Tahap ini meliputi *case folding*, *stopword removal*, *tokenisasi*, dan *stemming*.

1. Case Folding

   Tujuan dari proses ini adalah agar lebih mempermudah di proses selanjutnya. *Case folding* adalah merubah semua karakter huruf menjadi huruf kecil. Proses *case folding* ditunjukan pada gambar 4.5.

```
//case folding
[1] $file = strtolower(trim($artikel));
```

Gambar 4. 5 Proses untuk Case Folding Artikel

2. Tokenisasi

   Tokenisasi adalah proses penghilangan tanda baca dan pemotongan kata pada kalimat dalam dokumen. Proses ini melakukan pemotongan data yang berupa *string* menjadi kata-kata tunggal berdasarkan spasi. Terdapat tahap

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

filtering yang ditunjukkan pada gambar 4.6 baris ke-1 sampai ke-13 dan tahap kemudian tahap tokenisasi di tunjukkan pada baris ke-14.

```
[1] $teks = str_replace("?", " ", $file);
[2] $teks = str_replace("'", " ", $teks);
[3] $teks = str_replace("-", " ", $teks);
[4] $teks = str_replace(")", " ", $teks);
[5] $teks = str_replace("(", " ", $teks);
[6] $teks = str_replace("\"", " ", $teks);
[7] $teks = str_replace("/", " ", $teks);
[8] $teks = str_replace("=", " ", $teks);
[9] $teks = str_replace(".", " ", $teks);
[10] $teks = str_replace(",", " ", $teks);
[11] $teks = str_replace(":", " ", $teks);
[12] $teks = str_replace(";", " ", $teks);
[13] $teks = str_replace("!", " ", $teks);
[14] $token1 = explode(" ", $teks);
```

Gambar 4. 6 Proses untuk Tokenisasi

3. Stopword Removal

Setelah dilakukan pemotongan per kata dalam kalimat. Tahap selanjutnya adalah proses kata-kata yang *term*asuk stopword. Daftar stopword terdapat di tabel stopword di *database*. Proses Stopword Removal di tunjukkan pada gambar 4.7

```
[1] for ($i = 0; $i < count ( $token1 ); $i++ ) {
[2]   $result = mysqli_query($conn,"SELECT * FROM
tb_stopword WHERE kata_stopword = '$token1[$i]' LIMIT 1");
[3]   $result2 = mysqli_num_rows($result);
[4]    if($result2 > 0 ){
[5]      $y[$i] = '';
[6]    } else {
[7]      $y[$i] = $token1[$i];
[8]     };
[9]    }
[10] $artikel = array_empty_remover($y);
[11] $artikel = implode(" ", $artikel);
[12] $artikel = strtolower(trim($artikel));
[13] return $artikel;
[14] }
```

Gambar 4. 7 Proses untuk *Stopword Removal*

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

4. Stemming

Stemming adalah proses mengubah kata ke bentuk dasarnya dengan cara menghilangkan imbuhan-imbuhan pada kata dalam dokumen. Dalam penelitian ini, stemming menggunakan algoritma Nazief Adriani. Proses stemming di tunjukan pada gambar 4.8.

Proses stemming dilakukan dengan menghilangkan *suflix* dan *prefix*. Pada tahap ini akan di jelaskan satu satu yakni *suflix*. *Sulfix* adalah penghilangan imbuhan yang ada di akhir kata. *Source code* dari proses di *suflix* di tunjukan pada gambar 48 baris ke-11 sampai bari ke-25. Pada *suflix* dilakukan penghilangan akhiran *i, an* yang di cek pada baris ke-13 dan akhiran *kan* pada baris ke-17 dengan method *preg_match*, kemudian dihilangkan dengan method *preg_replace* pada baris ke-14 dan ke-18. Hasilnya akan dilakukan pengecekan pada tabel kata dasar yang telah di simpan pada *database*. Jika kata tersebut terdapat di *database* atau *return true* maka kata tersebut sudah berbentuk berupa kata dasar. Jika tidak ada di *database* atau *return false* maka dilakukan proses stemming (*suflix* dan *prefix*).

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

```
[1]   function stemming($artikel){
[2]         $tampung = explode(" ", $artikel);
[3]               for ($i = 0; $i < count($tampung); $i++){
[4]               $tampung[$i] = stemmer_proses
($tampung[$i]);
[5]         }
[6]   $artikel = implode(" ", $tampung);
[7]   $artikel = strtolower(trim($artikel));
[8]   return $artikel;
[9]   }
[10]
[11]  function Del_Derivation_Suffixes($kata){
[12]  $kataAsal = $kata;
[13]  if(preg_match('/(i|an)\z/i',$kata)){ // Cek Suffixe
[14]        $__kata = preg_replace('/(i|an)\z/i','',$kata);
[15]        if(cekKamus($__kata)){ // Cek Kamus
[16]              return $__kata;
[17]        }else if(preg_match('/(kan)\z/i',$kata)){
[18]        $__kata = preg_replace('/(kan)\z/i','',$kata);
[19]              if(cekKamus($__kata)){
[20]                    return $__kata;
[21]              }
[22]        }
[23]  }
[24]  return $kataAsal;
[25]  }
[26]  function cekKamus($kata){
[27]        global $conn;
[28]        $sql = mysqli_query($conn," SELECT * from
tb_kadar where kata_kadar ='$kata' LIMIT 1");
[29]        $result = mysqli_num_rows($sql);
[30]        if($result==1){
[31]              return true; // True jika ada
[32]        }else{
[33]              return false; // jika tidak ada FALSE
[34]        }
[35]  }
```

Gambar 4. 8 Proses untuk Stemming

Hasil akhir dari tahap *preprocessing* adalah berupa index *term-term* dari artikel kesehatan yang terlibat. Kemudian index tersebut di simpan di dalam *database* pada tabel tb_index. Dari 100 artikel kesehatan yang terproses, menghasilkan 11.854 *term*. Sampel hasilnya ditunjukkan pada gambar 4.9.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

	id_index	id_artikel	term
Edit Copy Delete	1	1	sehat
Edit Copy Delete	2	1	hilang
Edit Copy Delete	3	1	lemak
Edit Copy Delete	4	1	paha
Edit Copy Delete	5	1	gemuk
Edit Copy Delete	6	1	gelambir
Edit Copy Delete	7	1	sungguh
Edit Copy Delete	8	1	rusak
Edit Copy Delete	9	1	amil
Edit Copy Delete	10	1	wanita
Edit Copy Delete	11	1	timbun
Edit Copy Delete	12	1	sebab
Edit Copy Delete	13	1	utama
Edit Copy Delete	14	1	tarik
Edit Copy Delete	15	1	artikel
Edit Copy Delete	16	1	kali
Edit Copy Delete	17	1	bahas
Edit Copy Delete	18	1	aman
Edit Copy Delete	19	1	wajah
Edit Copy Delete	20	1	tubuh
Edit Copy Delete	21	1	seringkali
Edit Copy Delete	22	1	pusat
Edit Copy Delete	23	1	hati
Edit Copy Delete	24	1	orang
Edit Copy Delete	25	1	milik

Gambar 4. 9 Sampel Index Term Hasil Preprocessing

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

4.1.4 Pembobotan TF.IDF pada Tahap Implementasi

Setelah *term-term* hasil *preprocessing* tersimpan di *database*, tahap selanjutnya adalah pembobobatan tiap-tiap *term*. Dalam penelitlian ini, pembobotan dilakukan menggunakan *Term Frequecy-Inverse Document Frecuency* (TF.IDF).

Pembobotan TF dilakukan dengan menghitung jumlah kemunculan *term* pada masing-masing artikel. Sedangkan perhitungan IDF dilakukan dengan menghitung jumlah kemunculan *term* pada semua artikel yang terlibat dalam proses. Kemudian hasil dari TF dan IDF di kalikan berdasarkan masing-masing *term* sehingga menghasilkan bobot TF.IDF. Hasil dari perhitungan dan pembobotan di simpan ke *database*. Proses TF di tunjukan pada gambar 4.9. Proses IDF di tunjukan pada gambar 4.10. Dan output dari TF.IDF ditunjukan pada Tabel 4.11. Dan sampel hasinya ditunjukkan pada gambar 4.12.

```
[1] $ambil_u = file_get_contents("$lokasi.txt");
[2] $gabung = $artikel.'. '.$ambil_u;
[3] $gabung = prepro($gabung);
[4] $gabung = stemming($gabung);
[5] $art_term = explode(" ", trim($gabung));
[6] $juml_term = count($art_term);
[7] foreach ($art_term as $a => $value) {
[8]   if ($art_term[$a] != "") {
[9]    $cek = mysqli_query($conn,"SELECT tf FROM tb_index
WHERE term='$art_term[$a]' AND id_artikel=$id_artikel");
[10]    if (mysqli_num_rows($cek) > 0){
[11]     $baris = mysqli_fetch_array($cek, MYSQL_ASSOC);
[12]        $juml_baris = $baris['tf'];
[13]        $juml_baris++;
[14]        mysqli_query($conn,"UPDATE tb_index SET tf =
$juml_baris WHERE term='$art_term[$a]' AND
id_artikel=$id_artikel");
[15]        } else {
[16]        mysqli_query($conn,"INSERT INTO tb_index
(id_artikel, term, tf, idf, bobot) VALUES ($id_artikel,
'$art_term[$a]', 1, 0, 0)");
[17]        }
[18]    }
[20] }
```

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Gambar 4. 10 Proses untuk *Term Frequency* (TF)

```
[1] function idf(){
[2]   global $conn;
[3]   $ambil_n = mysqli_query($conn,"SELECT DISTINCT
id_artikel FROM tb_index");
[4]   $ambil_term = mysqli_query($conn,"SELECT * FROM
tb_index ORDER BY id_index");
[5]   $n = mysqli_num_rows($ambil_n);
[6]   while ($row = mysqli_fetch_array($ambil_term,
MYSQL_ASSOC)) {
[7]         $id_index = $row['id_index'];
[8]         $term = $row['term'];
[9]         $ambil_N = mysqli_query($conn,"SELECT COUNT(*)
as N FROM tb_index WHERE term='$term'");
[10]        $temp_N = mysqli_fetch_array($ambil_N,
MYSQL_ASSOC);
[11]        $N = $temp_N['N'];
[12]              if ($N >= 0){
[13]                    $idf = 1+log($n/$N);
[14]                    mysqli_query($conn,"UPDATE tb_index
SET idf=$idf WHERE id_index=$id_index");
[15]              }
[16]        }
[17]  }
```

Gambar 4. 11 Proses untuk *Inverse Document Frequency* (IDF)

```
[1] function bobot(){
[2]   global $conn;
[3]   $ambil = mysqli_query($conn,"SELECT * FROM tb_index
ORDER BY id_index");
[4]   while ($row = mysqli_fetch_array($ambil, MYSQL_ASSOC))
{
[5]    $id_index = $row['id_index'];
[6]    $tf = $row['tf'];
[7]    $idf = $row['idf']     ;
[8]     if ($tf || $idf >= 0){
[9]         $bobot  = $tf * $idf;
[10]        }
[11]        mysqli_query($conn,"UPDATE tb_index SET
bobot=$bobot WHERE id_index=$id_index");
[12]        }
[13]  }
```

Gambar 4. 12 Proses untuk TF.IDF

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

	id_index	id_artikel	term	tf	idf	bobot
Edit Copy Delete	1	1	sehat	6	1.16252	6.97512
Edit Copy Delete	2	1	hilang	12	2.02165	24.2598
Edit Copy Delete	3	1	lemak	30	2.27297	68.1891
Edit Copy Delete	4	1	paha	44	4.21888	185.631
Edit Copy Delete	5	1	gemuk	2	3.81341	7.62682
Edit Copy Delete	6	1	gelambir	1	5.60517	5.60517
Edit Copy Delete	7	1	sungguh	1	5.60517	5.60517
Edit Copy Delete	8	1	rusak	1	2.66073	2.66073
Edit Copy Delete	9	1	amil	1	5.60517	5.60517
Edit Copy Delete	10	1	wanita	3	2.51413	7.54239
Edit Copy Delete	11	1	timbun	1	4.50656	4.50656
Edit Copy Delete	12	1	sebab	7	1.38566	9.69962
Edit Copy Delete	13	1	utama	1	2.51413	2.51413
Edit Copy Delete	14	1	tarik	1	3.20727	3.20727
Edit Copy Delete	15	1	artikel	1	4.50656	4.50656
Edit Copy Delete	16	1	kali	2	2.13943	4.27886
Edit Copy Delete	17	1	bahas	3	2.96611	8.89833
Edit Copy Delete	18	1	aman	1	3.20727	3.20727
Edit Copy Delete	19	1	wajah	1	2.42712	2.42712
Edit Copy Delete	20	1	tubuh	13	1.22314	15.9008
Edit Copy Delete	21	1	seringkali	1	3.30259	3.30259
Edit Copy Delete	22	1	pusat	1	4.50656	4.50656
Edit Copy Delete	23	1	hati	3	1.96758	5.90274
Edit Copy Delete	24	1	orang	6	1.31471	7.88826
Edit Copy Delete	25	1	milik	7	1.27444	8.92108
Edit Copy Delete	26	1	ukur	19	2.89712	55.0453
Edit Copy Delete	27	1	percaya	1	2.42712	2.42712
Edit Copy Delete	28	1	bebas	1	3.12026	3.12026
Edit Copy Delete	29	1	maka	1	3.20727	3.20727
Edit Copy Delete	30	1	aneka	3	3.40795	10.2239

Gambar 4. 13 Sampel data Pembobotan TF.IDF Index *term*

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

4.1.5 Pembobotan *Latent Semantic Indexing* (LSI) pada Tahap Implentasi

Setelah masing-masing *term* di hitung bobotnya. Tahap selanjutnya adalah perhitungan LSI. Proses ini menggunakan operasi *Single Value Decomposition* (SVD). Semua index *term* yang ada di *database* ditampilkan dan dibentuk sebagai sebuah matriks untuk dapat di proses menggunakan SVD yakni sebuah matriks A. Proses SVD menghasilkan 3 matriks yakni U, S, dan V. Proses pembetukan matriks *term* ditunjukan pada gambar 4.13. Hasil pembentukan matriks *term* artikel dan matriks *term* uery di tunjukkan pada gambar 4.14.

```
[1] $A = array();
[2] $ambil_termnya = mysqli_query($conn, "SELECT DISTINCT term
FROM tb_index");
[3] $b = mysqli_num_rows($ambil_termnya);
[4] $ambil_art = mysqli_query($conn,"SELECT DISTINCT id_artikel
FROM tb_index");
[5] $hitung_art = mysqli_num_rows($ambil_art);
[6] echo "Jumlah Artikel = ".$hitung_art."</br>";
[7]   for ($row=0; $row < $b; $row++) {
[8]    $a = mysqli_fetch_array($ambil_termnya);
[9]    $terms[] = $a['term'];
[10]   for ($col=0; $col < $hitung_art ; $col++) {
[11]        $col_plus = $col+1;
[12]        $temp = mysqli_query($conn, "SELECT bobot FROM
tb_index WHERE term='$terms[$row]' AND
id_artikel='$col_plus'");
[13]        $am_temp = mysqli_fetch_array($temp, MYSQL_ASSOC);
[14]        $t = $am_temp['bobot'];
[15]        $A [$row][$col] = $t;
[16]  }
[17]}
```

Gambar 4. 14 Proses untuk Membentuk Matriks *Term*

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

2.602	2.602	0.000	0.000
1.041	0.000	1.041	1.041
1.301	0.000	1.301	0.000
1.602	0.000	0.000	0.000
1.602	0.000	0.000	0.000
1.602	0.000	0.000	0.000
1.602	0.000	0.000	0.000
1.041	0.000	1.041	1.041
0.000	0.000	3.204	0.000
0.000	1.602	0.000	0.000
0.000	1.602	0.000	0.000
0.000	1.602	0.000	0.000
0.000	1.602	0.000	0.000
0.000	1.602	0.000	0.000
0.000	1.602	0.000	0.000
0.000	1.602	0.000	0.000
0.000	1.602	0.000	0.000
0.000	0.000	3.204	0.000
0.000	0.000	3.204	0.000
0.000	0.000	3.204	0.000
0.000	0.000	3.204	0.000
0.000	0.000	3.204	0.000
0.000	0.000	2.602	1.301
0.000	0.000	1.602	0.000
0.000	0.000	0.000	1.602
0.000	0.000	0.000	1.602
0.000	0.000	0.000	1.602

(1)

0.000	1.041	1.301	0.000	0.000
0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000
0.000	0.000	1.301	0.000	0.000
0.000	0.000			

(2)

Gambar 4. 15 (1) Sampel matriks data *term* (2) matriks *query*

Setelah matriks *term* terbentuk, selanjutnya matirks tersebut akan di dekomposisi menggunan SVD. Dan menghasilkan matriks U, S, V. Dalam kaitannya agar mempermudah perhitungan selanjutnya, matriks hasil SVD di lakukan pengurangan dimensi, sebanyak nilai *k*. Dalam penelitian ini akan di uji coba dengan nilai k=80. Prosesnya adalah mengambil *k* kolom pertama dari matriks

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

U dan *V*. dan mengambil *k* baris dan kolom pertama pada matriks *S*. Sehingga matriks hasil SVD di sebut matriks U_k, S_k, V^T_k.

-0.026	0.637	0.182	-0.115
-0.139	0.096	0.250	0.208
-0.160	0.110	0.232	-0.124
-0.014	0.151	0.317	-0.119
-0.014	0.151	0.317	-0.119
-0.014	0.151	0.317	-0.119
-0.014	0.151	0.317	-0.119
-0.139	0.096	0.250	0.208
-0.366	-0.030	-0.061	-0.068
-0.002	0.241	-0.204	0.048
-0.002	0.241	-0.204	0.048
-0.002	0.241	-0.204	0.048
-0.002	0.241	-0.204	0.048
-0.002	0.241	-0.204	0.048
-0.002	0.241	-0.204	0.048
-0.002	0.241	-0.204	0.048
-0.002	0.241	-0.204	0.048
-0.366	-0.030	-0.061	-0.068
-0.366	-0.030	-0.061	-0.068
-0.366	-0.030	-0.061	-0.068
-0.366	-0.030	-0.061	-0.068
-0.366	-0.030	-0.061	-0.068
-0.311	-0.014	0.030	0.330
-0.183	-0.015	-0.030	-0.034
-0.016	0.012	0.098	0.473
-0.016	0.012	0.098	0.473
-0.016	0.012	0.098	0.473

Gambar 4. 16 Sampel matriks U

8.694	0.000	0.000	0.000
0.000	5.615	0.000	0.000
0.000	0.000	4.102	0.000
0.000	0.000	0.000	3.259

Gambar 4. 17 Sampel matriks S

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

-0.075	0.528	0.810	-0.242
-0.011	0.846	-0.524	0.098
-0.993	-0.053	-0.078	-0.069
-0.089	0.043	0.251	0.963

Gambar 4. 18 Sampel matriks V

U_k

-0.026	0.637
-0.139	0.096
-0.160	0.110
-0.014	0.151
-0.014	0.151
-0.014	0.151
-0.014	0.151
-0.139	0.096
-0.366	-0.030
-0.002	0.241
-0.002	0.241
-0.002	0.241
-0.002	0.241
-0.002	0.241
-0.002	0.241
-0.002	0.241
-0.002	0.241
-0.366	-0.030
-0.366	-0.030
-0.366	-0.030
-0.366	-0.030
-0.366	-0.030
-0.311	-0.014
-0.183	-0.015
-0.016	0.012
-0.016	0.012
-0.016	0.012

S_k

8.694	0.000
0.000	5.615

V_k

-0.075	-0.011	-0.993	-0.089
0.528	0.846	-0.053	0.043

V^T_k

-0.075	0.528
-0.011	0.846
-0.993	-0.053
-0.089	0.043

Gambar 4. 19 Sampel matriks hasil dekomposisi

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

```
[1] $input = new Matrix($A);
[2] $svd = $input->svd();
[3] $U = $svd->getU();
[4] $S = $svd->getS();
[5] $V = $svd->getV();
[6] $VT = $V -> transpose();
[7] $to_int_k = intval($ambil_k);
[8] $Uk = $svd->getUr($to_int_k);
[9] $Sk = $svd->getSr($to_int_k);
[10] $Vk = $svd->getVr($to_int_k);
[11] $VTk = $VT->inverse();
[12] $VTk = $VTk->reducerow($input->rank());
[13] $string_data1 = serialize($Uk);
[14] $string_data2 = serialize($Sk);
[15] $string_data3 = serialize($VTk);
[16] simpan_uk($string_data1, $ambil_k);
[17] simpan_sk($string_data2, $ambil_k);
[18] simpan_vtk($string_data3, $ambil_k);
```

Gambar 4. 20 Proses untuk dekomposisi dengan SVD

[-6.581 1.263]

Gambar 4. 21 Matriks hasil mapping

Kemudian matriks hasil proses SVD yakni U_k, S_k, V^T_k di simpan dalam bentuk text*file*. Tujuannya adalah agar mudah di gunakan untuk proses selanjutnya yakni perhitungan *similarity*.

4.1.6 Perhitungan *Similarity* pada Tahap Implementasi

Pada tahap perhitungan *similarity*, perhitungan menggunakan *cosine similarity*. Perhitungan ini adalah antara *query* yang dimasukkan pengguna dengan masing-masing dokumen artikel yang didasarkan dengan matriks hasil pembobotan dan LSI. Perhitungan dilakukan dengan cara meghitung kedekatan antara matriks *vector space model* pada *query* dengan matriks *vector space model* pada dokumen artikel. Semakin besar kedekatannya, maka semakin besar pula nilai *similarity*-nya.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Sama halnya dengan data artikel, untuk dapat di hitung *similarity*, *query* juga dilakukan tahap *preprocessing* dan pembobotan. Yang hasilnya berupa matriks *query* dan kemudian di simpan di dalam *file* berformat text *file* untuk nantinya di proses bersama *file* berformat text *file* dari artikel hasil dari LSI.

Setelah *query* di proses di tahap preprocesing dan mendapatkan matriks *query*. Kemudian matriks *query* di kalikan dengan matriks U_k, dan matriks S_k. Yang hasilnya adalah matriks *query* baru berukuran 1x2. Matriks *query* baru tersebut lalu di lakukan proses perkalian dengan matriks V^T_k. Jumlah kolom pada matriks V^T_k menginisialisasikan jumlah artikel yang terlibat, misalkan kolom pertama pada matriks V^T_k menginisialisasikan dokumen artikel pertama. Proses perkalian matriks ditunjukkan pada gambar 4.22.

Proses *similarity* dilakukan dengan menghitung panjang *vector* matriks *query* dan matriks *vector* V^T_k. Panjang *vector* didapat dari nilai akar dari jumlah kuadrat matriks *query* baru dan V^T_k. Proses perhitungan panjang *vector* di tunjukkan pada gambar 4.23, mulai baris ke-14 sampai baris ke-26. Kemudian juga menghitung perkalian antara bobot *vector query* baru dengan bobot matriks $V^T_{k.}$ Proses perhitungan perkalian antara kedua *vector* tersebut di tunjukkan pada gambar 4.24, mulai baris ke-1 sampai baris ke-13. Hasil akhirnya atau nilai *similarity* diperoleh dengan membagi jumlah perkalian bobot dengan perkalian panjang *vector query* dan panjang *vector* V^T_k

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

```
[1] $ambil_uk = file_get_contents("Matriks Uk.txt");
[2] $ambil_sk = file_get_contents("Matriks Sk.txt");
[3] $ambil_vtk = file_get_contents("Matriks VTk.txt");
[4] $ambil_bobot = file_get_contents("Matriks Bobot.txt");
[5] $array_uk = unserialize($ambil_uk);
[6] $Uk = array();
[7] foreach ($array_u as $key => $value) {
[8]  foreach ($value as $key2 => $value2) {
[9]   foreach ($value2 as $key3 => $value3) {
[10]    $Uk[$key2][$key3] = $value3;
[11]   }
[12]  }
[13] }
[14] function perkalian_matriks($matriks_a, $matriks_b) {
[15]  $hasil = array();
[16]   for ($i=0; $i<sizeof($matriks_a); $i++) {
[17]    for ($j=0; $j<sizeof($matriks_b[0]); $j++) {
[18]    $temp = 0;
[19]     for ($k=0; $k<sizeof($matriks_b); $k++) {
[20]      $temp += $matriks_a[$i][$k] * $matriks_b[$k][$j];
[21]     }
[22]    $hasil[$i][$j] = $temp;
[23]   }
[24]  }
[25] return $hasil;
[27] }
```

Gambar 4. 22 Proses untuk Perkalian Matriks *Term*

```
[1] function kuadrat ($mat_a, $mat_b){
[2]  $hasil1 = array();
[3]   for ($a=0; $a < sizeof($mat_b[0]) ; $a++) {
[4]    for ($b=0; $b < sizeof($mat_a) ; $b++) {
[5]     $temp = 0;
[6]       for ($c=0; $c < sizeof($mat_b); $c++) {
[7]         $temp += $mat_a[$b][$c] * $mat_b[$c][$a];
[8]         }
[9]     }
[10]  $hasil1[$a] = $temp;
[11]  }
[12]  return $hasil1;
[13] }
[14] function vector($matriks_a){
[15]  $temp1 = array();
[16]   for ($i=0; $i < sizeof($matriks_a); $i++) {
[17]    $hasil = 0;
[18]     for ($j=0; $j < sizeof($matriks_a[0]) ; $j++) {
[19]        $temp = pow($matriks_a[$i][$j], 2);
[20]        $hasil += $temp;
[21]      }
[22]        $hasil_a = sqrt($hasil);
[23]        $temp1[$i] = $hasil_a;
[24]      }
[25]      return $temp1;
[26]  }
```

Gambar 4. 23 Proses untuk Perkalian Kolom dan Panjang *Vector* Matriks *term*

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

```
[1]   function hitung_vector($kali, $vector_query,
$vector_dok){
[2]    $hasil = array();
[3]    for ($i=0; $i < sizeof($vector_dok) ; $i++) {
[4]     $temp = 0;
[5]     $temp = $kali[$i] / ($vector_query[0]*$vector_dok[$i]);
[6]     $hasil[$i] = $temp;
[7]    }
[8]   return $hasil;
[9]   }
```

Gambar 4. 24 Proses untuk Operasi *Similarity*

Kemudian hasilnya akan di simpan di *database* riwayat pencarian. Tujuannya adalah agar mudah di akses apabila data *query* di butuhkan lagi. Proses menyimpan *query* dan nilai *similarity* di tunjukkan pada gambar 4.25.

```
[1] function simpan_query($hasil_akhir, $keyw, $jenis, $k){
[2]  global $conn;
[3]  $cekkk = mysqli_query($conn, "SELECT * FROM tb_query WHERE
query ='$keyw' AND jenis_query = '$jenis' AND nilai_k = '$k'");
[4]  $cekkk2 = mysqli_num_rows($cekkk);
[5]  if ($cekkk2 == 0) {
[6]   for ($i=0; $i < sizeof($hasil_akhir) ; $i++) {
[7]    $ii = $i+1;
[8]    $cek = mysqli_query($conn, "SELECT DISTINCT (id) FROM
tb_query WHERE query = '$keyw' AND jenis_query ='$jenis' AND
nilai_k = $k");
[9]    $cekin = mysqli_query($conn, "SELECT max(id) FROM
tb_query");
[10]    $cek2 = mysqli_num_rows($cek);
[11]     if ($cek2 == 0) {
[12]      while ($row = mysqli_fetch_array($cekin,
MYSQL_ASSOC)) {
[13]       $idd = $row['max(id)'];
[14]       $idd++;
[15]       mysqli_query($conn, "INSERT INTO tb_query (id,
query, jenis_query, nilai_k, id_artikel, similarity) VALUES
('$idd', '$keyw', '$jenis', '$k', '$ii', '$hasil_akhir[$i]')");
[16]       }
[17]      } else if ($cek2 > 0) {
[18]       foreach ($cek as $value) {
[19]       $id = $value['id'];
[20]       mysqli_query($conn, "INSERT INTO tb_query (id,
query, jenis_query, nilai_k, id_artikel, similarity) VALUES
('$id', '$keyw', '$jenis', '$k', '$ii', '$hasil_akhir[$i]')");
[21]      }
[22]     }
[23]  }
```

Gambar 4. 25 Proses untuk Menyimpan Nilai *Similarity*

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Hasil nilai *similarity* yang di simpan di *database* di tunjukkan pada Gambar 4.26.

	nomor_query	id	query	jenis_query	nilai_k	id_artikel	similarity
Edit Copy Delete	1	1	Makanan Bergizi	semantic	80	1	0.0605663
Edit Copy Delete	2	1	Makanan Bergizi	semantic	80	2	0.127572
Edit Copy Delete	3	1	Makanan Bergizi	semantic	80	3	0.00156158
Edit Copy Delete	4	1	Makanan Bergizi	semantic	80	4	0.296961
Edit Copy Delete	5	1	Makanan Bergizi	semantic	80	5	0.0325511
Edit Copy Delete	6	1	Makanan Bergizi	semantic	80	6	0.00042167
Edit Copy Delete	7	1	Makanan Bergizi	semantic	80	7	0.105342
Edit Copy Delete	8	1	Makanan Bergizi	semantic	80	8	0.0222226
Edit Copy Delete	9	1	Makanan Bergizi	semantic	80	9	0.344196
Edit Copy Delete	10	1	Makanan Bergizi	semantic	80	10	0.0756385
Edit Copy Delete	11	1	Makanan Bergizi	semantic	80	11	0.000492005
Edit Copy Delete	12	1	Makanan Bergizi	semantic	80	12	0.00277662
Edit Copy Delete	13	1	Makanan Bergizi	semantic	80	13	0.028209
Edit Copy Delete	14	1	Makanan Bergizi	semantic	80	14	0.0126994
Edit Copy Delete	15	1	Makanan Bergizi	semantic	80	15	0.250013
Edit Copy Delete	16	1	Makanan Bergizi	semantic	80	16	0.0740994
Edit Copy Delete	17	1	Makanan Bergizi	semantic	80	17	0.145813
Edit Copy Delete	18	1	Makanan Bergizi	semantic	80	18	-0.00145992
Edit Copy Delete	19	1	Makanan Bergizi	semantic	80	19	0.00109873
Edit Copy Delete	20	1	Makanan Bergizi	semantic	80	20	0.172955
Edit Copy Delete	101	2	Penyebab mulut bau	semantic	80	1	0.0253727
Edit Copy Delete	102	2	Penyebab mulut bau	semantic	80	2	0.115242
Edit Copy Delete	103	2	Penyebab mulut bau	semantic	80	3	0.0331446
Edit Copy Delete	104	2	Penyebab mulut bau	semantic	80	4	0.00275083
Edit Copy Delete	105	2	Penyebab mulut bau	semantic	80	5	0.0100797
Edit Copy Delete	106	2	Penyebab mulut bau	semantic	80	6	0.0243063
Edit Copy Delete	107	2	Penyebab mulut bau	semantic	80	7	0.021588
Edit Copy Delete	108	2	Penyebab mulut bau	semantic	80	8	0.0181164
Edit Copy Delete	109	2	Penyebab mulut bau	semantic	80	9	0.000325206
Edit Copy Delete	110	2	Penyebab mulut bau	semantic	80	10	0.0175848
Edit Copy Delete	111	2	Penyebab mulut bau	semantic	80	11	0.0435191
Edit Copy Delete	112	2	Penyebab mulut bau	semantic	80	12	0.00308233
Edit Copy Delete	113	2	Penyebab mulut bau	semantic	80	13	0.0101081
Edit Copy Delete	114	2	Penyebab mulut bau	semantic	80	14	0.000939562
Edit Copy Delete	115	2	Penyebab mulut bau	semantic	80	15	0.00181219
Edit Copy Delete	116	2	Penyebab mulut bau	semantic	80	16	0.0199152
Edit Copy Delete	117	2	Penyebab mulut bau	semantic	80	17	-0.00116261
Edit Copy Delete	118	2	Penyebab mulut bau	semantic	80	18	0.00193379
Edit Copy Delete	119	2	Penyebab mulut bau	semantic	80	19	0.00864557
Edit Copy Delete	120	2	Penyebab mulut bau	semantic	80	20	0.000456142
Edit Copy Delete	121	2	Penyebab mulut bau	semantic	80	21	0.00169963

Gambar 4. 26 Sampel Nilai *Similarity*

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

4.2 Hasil dan Analisa Uji coba

Uji coba bertujuan untuk mengukur kemampuan kinerja aplikasi, dimana dilakukan dengan menggunakan 20 *query* yang telah ditetapkan. Dari *query* tersebut, dapat dihitung dan dievalusi dengan mendapatkan nilai *precision*, kemudian dilakukan perhitungan rata-rata dengan metode *Mean Average Precision* (MAP) untuk mengetahui tingkat kemampuan sistem yang telah dibuat. Daftar *query* di tunjukkan pada tabel 3.1. Data yang digunakan dalam uji coba ini merupakan kumpulan dokumen artikel kesehatan yang berjumlah 100 artikel, yang menghasilkan 11.854 *term* dari tahap *preprocessing* data, dan menghasilkan matriks U, S, V^T dari hasil *Single Value Decompotition* (SVD) dari tahap *Latent Semantic Indexing* (LSI).

Dokumen artikel yang terlibat akan dilakukan tahap *preprocessing* data meliputi *case folding*, *tokenisasi*, *stopword removal* dan *stemming*, yang mana akan menhasilkan *term-term dokumen*. Kemudian juga dilakukan proses pembobotan pada *term-term dokumen artikel* yang ada. Hasil pembobotan *term-term dokumen* tersebut dibentuk menjadi matriks bobot dokumen dan kemudian akan diproses dengan matriks SVD yang menghasilkan matriks U, S, V^T. Kemudian ketiga matriks tersebut di kurangi dimensinya dengan tujuan untuk mempercepat proses, sebanyak *k*. Nilai *k* yang di gunakan adalah 80. Sehinggan menjadi matriks U_k, S_k dan V^T_k.

Setiap *query* juga akan dilakukan *preprocessing* dan pembobotan yang menghasilkan matriks bobot *query*. Kemudian matriks bobot *query* dikalikan dengan matriks U_k dan S_k dari matriks bobot dokumen. Hasil dari perkalian tersebut

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

kemudian akan dikalikan dengan matriks V^T_k, yang mana setiap kolom pada matriks V^T_k menginisialisasikan dokumen artikel. Hasil akhirnya berupa nilai *similarity* antara *query* dengan setiap dokumen artikel.

Evaluasi kemampuan sistem dalam perangkingan dokumen dapat dilakukan dengan melihat posisi dokumen yang relevan yang terdapat pada urutan pertama terhadap ke-20 input *query* yang di ujikan. Salah satu contoh pencarian di tunjukkan pada gambar 4.27. Data artikel yang ditampilkan di batasi 15 dan atau 10 teratas diurutkan berdasarkan pada tinggi nilai *similarity*-nya untuk uji coba.

Gambar 4. 27 Contoh Pencarian Artikel

Semua input *query* akan diujikan seperti yang dilakukan pada gambar 4.27 kemudian dari 15 dan atau 10 dokumen artikel yang di *retrieve* pada setiap *query* akan di hitung nilai *precision*-nya. Kemudian nilai *precision* tersebut akan di proses dengan metode MAP. Pengujian ini dilakukan dengan membandingkan metode yang di ajukan yakni TF.IDF.LSI dengan metode yang ada sebelumnya yakni TF.IDF.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Nilai *precision* di peroleh dengan cara pembagian antara dokumen artikel yang relevan dengan urutan dokumen artikel yang ter-*retrieve*. Penentuan dokumen artikel yang relevan didapat dari pakar yang merupakan seorang *user* yang memiliki latar belakang tentang masalah kesehatan. Perhitungan *precision* dilakukan pada setiap dokumen artikel yang te-*retrieve* pada semua input *query* yang diujikan. Nilai MAP di peroleh dari rata-rata nilai *precision* dari seluruh input *query* yang diujikan.

Hasil MAP dari TF.IDF pada *rank-k* 10 ditunjukan pada tabel 4.2, TF.IDF pada *rank-k* 15 ditunjukan pada tabel 4.3, TF.IDF.LSI pada *rank-k* 10 ditunjukan pada tabel 4.4, TF.IDF.LSI pada *rank-k* 15 ditunjukan pada tabel 4.5. Nilai MAP dari TF.IDF *rank-k* 15 adalah 79 % dan lebih kecil 3.4% dari nilai MAP dari metode *rank-k* 15 TF.IDF.LSI yang memiliki nilai 82.4%. Sedangkan nilai MAP dari TF.IDF *rank-k* 10 adalah 82.8 % dan lebih kecil 3.2% dari nilai MAP dari metode *rank-k* 10 TF.IDF.LSI yang memiliki nilai 86%.

Berdasarkan perhitungan nilai MAP jika di golongkan berdasarkan pada jenis *query* yakni *short* dan *long*. Dimana menunjukkan bahwa nilai MAP dari metode TF.IDF pada jenis *query* short *rank-k* 15 adalah 81.3 % dan *rank-k* 10 adalah 82.8 % dan pada jenis *query* long *rank-k* 15 adalah 78.2 % dan *rank-k* 10 adalah 82.9 %, yang tunjukan pada tabel 4.6 dan tabel 4.8. Kemudian nilai MAP dari metode TF.IDF.LSI pada jenis *query* short *rank-k* 15 adalah 78.2 %, *rank-k* 10 adalah 82.5 % dan pada jenis *query* long *rank-k* 15 adalah 86.6% dan *rank-k* 10 adalah 89 %, yang tunjukan pada tabel 4.6 dan tabel 4.8. Berdasarkan pada nilai tersebut, dapat di ketahui bahwa pada *query* yang pendek, akurasi dari metode TF.IDF lebih tinggi dari pada metode TF.IDF.LSI. Dan sebaliknya, pada *query* yang panjang metode

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

TF.IDF.LSI terbukti lebih tinggi nilai akurasinya dibandingkan dengan metode TF.IDF. Juga pada *rank-k* 10 pada pada setiap uji coba memiliki nilai yang lebih tinggi.

Kemudian hasil pengujian memperlihatkan bahwa metode TF.IDF.LSI terdapat kecerendungan dimana semakin panjang atau banyaknya kata dalam dokumen artikel, maka nilai *similarity*-nya semakin tinggi. Dan sebaliknya pada metode TF.IDF sebagian besar artikel yang pendek atau memiliki jumlah kata yang sedikit, maka semakin tinggi nilai *similarity*-nya. Hal tersebut terjadi karena pada metode TF.IDF.LSI memperhatikan hubungan *semantic* antar *term*. Hubungan semantic antar *term* menjadi menjadi lebih bervariasi atau banyak apabila dokumen artikel memiliki jumlah kata yang banyak pula. Maka otomatis akan mempengaruhi juga pada relevansi hasil pencarian.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Tabel 4. 2 Perhitungan MAP TF.IDF *rank-k*= 10

NO	*QUERY*	NILAI *PRECISION* PER ARTIKEL PER ARTIKEL										MP
		1	2	3	4	5	6	7	8	9	10	
1	Makanan bergizi	1		0.67		0.6	0.67	0.7				0.728
2	Penyebab mulut bau	1	1					0.4	0.5			0.725
3	Mencegah Serangan Jantung sejak dini		0.5		0.5	0.6		0.57	0.63		0.6	0.57
4	Sariawan	1	1	1	1	1						1
5	Jerawat tidak kunjung sembuh	1	1	1	1	1	1		0.88			0.98
6	Manfaat tidur siang	1	1	1		0.8	0.83					0.92
7	Kanker	1	1	1	1	1	1	1	1			1
8	Mencegah diabetes	1	1	1			0.66					0.73
9	Jenis makanan yang mengandung lemak	1	1	1	1		0.83			0.67		0.916
10	Menghilangkan mata panda	1	1	1			0.67	0.71				0.88
11	Menghilangkan komedo di wajah	1	1		0.75				0.5			0.81
12	Menyembuhkan Flek hitam bekas jerawat	1		0.67	0.75	0.8	0.83	0.86		0.78		0.79
13	Makanan yang mempunyai kalori tinggi	1	1		0.75	0.8	0.8		0.75	0.78		0.84
14	Tips sehat berolahraga di pagi hari	1	1	1	1	1	1				0.7	0.95
15	Tipe olahraga untuk menurunkan berat badan	1					0.33			0.33		0.55
16	Mengobati sakit gigi	1		0.67	0.75	0.8	0.8					0.804
17	Cara mudah mengobati sakit kepala	1	1	1	1			1				1
18	Diet yang ampuh	1	1	1	1		0.8		0.75		0.7	0.89
19	Gigi dan Mulut		0.5	0.67	0.75	0.8	0.83	0.86	0.88			0.756
20	Gejala penyakit jantung	1		0.67	0.75			0.58	0.62			0.73
TOTAL												**16.56**
MAP												**0.828**

LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF

Tabel 4. 3 Perhitungan MAP TF.IDF *rank-k*= 15

NO	*QUERY*	NILAI *PRECISION* PER ARTIKEL YANG RELEVAN															MP
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
1	Makanan bergizi	1		0.67		0.6	0.67	0.7				0.55				0.47	**0.67**
2	Penyebab mulut bau	1	1					0.4	0.5								**0.73**
3	Mencegah Serangan Jantung sejak dini		0.5		0.5	0.6		0.57	0.63		0.6						**0.57**
4	Sariawan	1	1	1	1	1											**1**
5	Jerawat tidak kunjung sembuh	1	1	1	1	1	1		0.88								**0.98**
6	Manfaat tidur siang	1	1	1		0.8	0.83										**0.92**
7	Kanker	1	1	1	1	1	1	1	1								**1**
8	Mencegah diabetes	1	1	1			0.66										**0.9**
9	Jenis makanan yang mengandung lemak	1	1	1	1		0.83			0.67						0.46	**0.85**
10	Menghilangkan mata panda	1	1	1			0.67	0.71									**0.87**
11	Menghilangkan komedo di wajah	1	1		0.75				0.5			0.46	0.5				**0.7**
12	Menyembuhkan Flek hitam bekas jerawat	1		0.67	0.75	0.8	0.83	0.86		0.78							**0.81**
13	Makanan yang mempunyai kalori tinggi	1	1		0.75	0.8	0.8		0.75	0.78		0.7	0.75		0.7		**0.8**
14	Tips sehat berolahraga di pagi hari	1	1	1	1	1	1				0.7	0.73		0.7			**0.8**
15	Tipe olahraga untuk menurunkan berat badan	1					0.33			0.33			0.33				**0.5**
16	Mengobati sakit gigi	1		0.67	0.75	0.8	0.8							0.46			**0.64**
17	Cara mudah mengobati sakit kepala	1	1	1	1			1									**0.94**
18	Diet yang ampuh	1	1	1	1		0.8		0.75		0.7					0.53	**0.85**
19	Gigi dan Mulut		0.5	0.67	0.75	0.8	0.83	0.86	0.88								**0.75**
20	Gejala penyakit jantung	1		0.67	0.75			0.58	0.62							0.4	**0.67**
TOTAL																	**15.95**
MAP																	**0.790**

LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF

Tabel 4. 4 Perhitungan MAP TF.IDF.LSI *rank-k*=10

NO	*QUERY*	NILAI PRESISI PER ARTIKEL YANG RELEVAN										MP
		1	2	3	4	5	6	7	8	9	10	
1	Makanan bergizi		0.5	0.67			0.5		0.5	0.56	0.6	0.56
2	Penyebab mulut bau	1	1	1						0.44		0.86
3	Mencegah Serangan Jantung sejak dini	1	1	1	1			0.71			0.6	0.89
4	Sariawan	1	1	1	1							1
5	Jerawat tidak kunjung sembuh	1	1	1	1	1	1			0.78	0.8	0.95
6	Manfaat tidur siang	1	1		0.75				0.5			0.81
7	Kanker	1	1	1	1	1	1	1	1			1
8	Mencegah diabetes	1	1	1	1	1						1
9	Jenis makanan yang mengandung lemak	1	1	1	1		0.83	0.86	0.88			0.94
10	Menghilangkan mata panda	1	1	1					0.5		0.5	0.8
11	Menghilangkan komedo di wajah	1	1	1			0.67			0.56		0.85
12	Menyembuhkan Flek hitam bekas jerawat	1	1	1	1	1	1		0.87			0.98
13	Makanan yang mempunyai kalori tinggi	1	1			0.3	0.67	0.71	0.75			0.63
14	Tips sehat berolahraga di pagi hari	1	1	1	1	1				0.67	0.7	0.91
15	Tipe olahraga untuk menurunkan berat badan	1	1	1	1	1	1		0.88		0.8	0.96
16	Mengobati sakit gigi	1		0.67		0.6	0.67		0.625			0.712
17	Cara mudah mengobati sakit kepala	1	1	1	1	1	1	1				1
18	Diet yang ampuh	1	1	1	1	1	1		0.87	0.9	0.9	0.96
19	Gigi dan Mulut	1	1	1	1	1	1	1	1	1	1	1
20	Gejala penyakit jantung		0.5						0.25		0.3	0.35
TOTAL												**17.16**
MAP												**0.86**

PERPUSTAKAAN UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Tabel 4. 5 Perhitungan MAP TF.IDF.LSI *rank-k*=15

NO	*QUERY*	NILAI PRESISI PER ARTIKEL YANG RELEVAN															MP
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
1	Makanan bergizi		0.5	0.67			0.5		0.5	0.56	0.6			0.54			**0.55**
2	Penyebab mulut bau	1	1	1						0.44						0.33	**0.76**
3	Mencegah Serangan Jantung sejak dini	1	1	1	1			0.71			0.6				0.5		**0.83**
4	Sariawan	1	1	1	1												**1**
5	Jerawat tidak kunjung sembuh	1	1	1	1	1	1			0.78	0.8						**0.95**
6	Manfaat tidur siang	1	1		0.75				0.5					0.31			**0.71**
7	Kanker	1	1	1	1	1	1	1	1								**1**
8	Mencegah diabetes	1	1	1	1	1							0.5			0.47	**0.85**
9	Jenis makanan yang mengandung lemak	1	1	1	1		0.83	0.86	0.88				0.67				**0.91**
10	Menghilangkan mata panda	1	1	1					0.5		0.5				0.43		**0.74**
11	Menghilangkan komedo di wajah	1	1	1			0.67			0.56			0.5	0.54			**0.76**
12	Menyembuhkan Flek hitam bekas jerawat	1	1	1	1	1	1		0.87			0.73					**0.95**
13	Makanan yang mempunyai kalori tinggi	1	1			0.3	0.67	0.71	0.75			0.64	0.67		0.64	0.67	**0.74**
14	Tips sehat berolahraga di pagi hari	1	1	1	1	1				0.67	0.7		0.67	0.69			**0.86**
15	Tipe olahraga untuk menurunkan berat badan	1	1	1	1	1	1		0.88		0.8	0.82			0.71		**0.92**
16	Mengobati sakit gigi	1		0.67		0.6	0.67		0.625						0.43		**0.67**
17	Cara mudah mengobati sakit kepala	1	1	1	1	1	1	1									**1**
18	Diet yang ampuh	1	1	1	1	1	1		0.87	0.89	0.9				0.78		**0.94**
19	Gigi dan Mulut	1	1	1	1	1	1	1	1	1	1						**1**
20	Gejala penyakit jantung		0.5						0.25		0.3			0.3	0.36		**0.34**
TOTAL																	**16.48**
MAP																	**0.824**

LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF

Tabel 4. 6 Nilai perhitungan MAP TF.IDF berdasarkan jenis *query* pada *rank-k*=10

No	*Query*	Kategori	Presisi
1	Makanan bergizi	Short	0.728
2	Penyebab mulut bau	Short	0.725
3	Sariawan	Short	1
4	Manfaat tidur siang	Short	0.92
5	Kanker	Short	1
6	Mencegah diabetes	Short	0.73
7	Mengobati sakit gigi	Short	0.804
8	Diet yang ampuh	Short	0.89
9	Gigi dan mulut	Short	0.756
10	Gejala penyakit jantung	Short	0.73
MAP Jenis Short			**0.828**
11	Mencegah Serangan Jantung sejak dini	Long	0.57
12	Jerawat tidak kunjung sembuh	Long	0.98
13	Jenis makanan yang mengandung lemak	Long	0.916
14	Menghilangkan mata panda	Long	0.88
15	Menghilangkan komedo di wajah	Long	0.81
16	Menyembuhkan Flek hitam bekas jerawat	Long	0.79
17	Makanan yang mempunyai kalori tinggi	Long	0.84
18	Tips sehat berolahraga di pagi hari	Long	0.95
19	Tipe olahraga untuk menurunkan berat badan	Long	0.55
20	Cara mudah mengobati sakit kepala	Long	1
MAP Jenis Long			**0.829**

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Tabel 4. 7 Nilai perhitungan MAP TF.IDF.LSI berdasarkan jenis *query* pada *rank*-k=10

No	*Query*	Kategori	Presisi
1	Makanan bergizi	Short	0.56
2	Penyebab mulut bau	Short	0.86
3	Sariawan	Short	1
4	Manfaat tidur siang	Short	0.81
5	Kanker	Short	1
6	Mencegah diabetes	Short	1
7	Mengobati sakit gigi	Short	0.712
8	Diet yang ampuh	Short	0.96
9	Gigi dan mulut	Short	1
10	Gejala penyakit jantung	Short	0.35
MAP Jenis Short			**0.825**
11	Mencegah Serangan Jantung sejak dini	Long	0.89
12	Jerawat tidak kunjung sembuh	Long	0.95
13	Jenis makanan yang mengandung lemak	Long	0.94
14	Menghilangkan mata panda	Long	0.8
15	Menghilangkan komedo di wajah	Long	0.85
16	Menyembuhkan Flek hitam bekas jerawat	Long	0.98
17	Makanan yang mempunyai kalori tinggi	Long	0.63
18	Tips sehat berolahraga di pagi hari	Long	0,91
19	Tipe olahraga untuk menurunkan berat badan	Long	0.96
20	Cara mudah mengobati sakit kepala	Long	1
MAP Jenis Long			**0.89**

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Tabel 4. 8 Nilai perhitungan MAP TF.IDF berdasarkan jenis *query* pada *rank-k*=15

No	*Query*	Kategori	Presisi
1	Makanan bergizi	Short	0.67
2	Penyebab mulut bau	Short	0.73
3	Sariawan	Short	1
4	Manfaat tidur siang	Short	0.92
5	Kanker	Short	1
6	Mencegah diabetes	Short	0.9
7	Mengobati sakit gigi	Short	0.64
8	Diet yang ampuh	Short	0.85
9	Gigi dan mulut	Short	0.75
10	Gejala penyakit jantung	Short	0.67
MAP Jenis Short			**0.813**
11	Mencegah Serangan Jantung sejak dini	Long	0.57
12	Jerawat tidak kunjung sembuh	Long	0.98
13	Jenis makanan yang mengandung lemak	Long	0.85
14	Menghilangkan mata panda	Long	0.87
15	Menghilangkan komedo di wajah	Long	0.7
16	Menyembuhkan Flek hitam bekas jerawat	Long	0.81
17	Makanan yang mempunyai kalori tinggi	Long	0.8
18	Tips sehat berolahraga di pagi hari	Long	0.8
19	Tipe olahraga untuk menurunkan berat badan	Long	0.5
20	Cara mudah mengobati sakit kepala	Long	0.94
MAP Jenis Long			**0.782**

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Tabel 4. 9 Nilai perhitungan MAP TF.IDF.LSI berdasarkan jenis *query* pada *rank-k*=15

No	*Query*	Kategori	Presisi
1	Makanan bergizi	Short	0.55
2	Penyebab mulut bau	Short	0.76
3	Sariawan	Short	1
4	Manfaat tidur siang	Short	0.71
5	Kanker	Short	1
6	Mencegah diabetes	Short	0.85
7	Mengobati sakit gigi	Short	0.67
8	Diet yang ampuh	Short	0.94
9	Gigi dan mulut	Short	1
10	Gejala penyakit jantung	Short	0.34
MAP Jenis Short			**0.782**
11	Mencegah Serangan Jantung sejak dini	Long	0.83
12	Jerawat tidak kunjung sembuh	Long	0.95
13	Jenis makanan yang mengandung lemak	Long	0.91
14	Menghilangkan mata panda	Long	0.74
15	Menghilangkan komedo di wajah	Long	0.76
16	Menyembuhkan Flek hitam bekas jerawat	Long	0.95
17	Makanan yang mempunyai kalori tinggi	Long	0.74
18	Tips sehat berolahraga di pagi hari	Long	0.86
19	Tipe olahraga untuk menurunkan berat badan	Long	0.92
20	Cara mudah mengobati sakit kepala	Long	1
MAP Jenis Long			**0.866**

Berdasarkan pada tabel 4.6, 4.7, 4.8 dan 4.9 telah di tujukkan mengenai perhitungan dari nilai MAP dari kedua metode yang di uji coba beserta dengan jenis *query* yang digunakan. Pada tabel 4.10 dan 4.11 ditunjukkan nilai presentasi dari semua percobaan yang telah dilakukan. Dan ditunjukkan juga grafik dari perbandingan nilai presentasi tersebut pada gambar 4.34.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

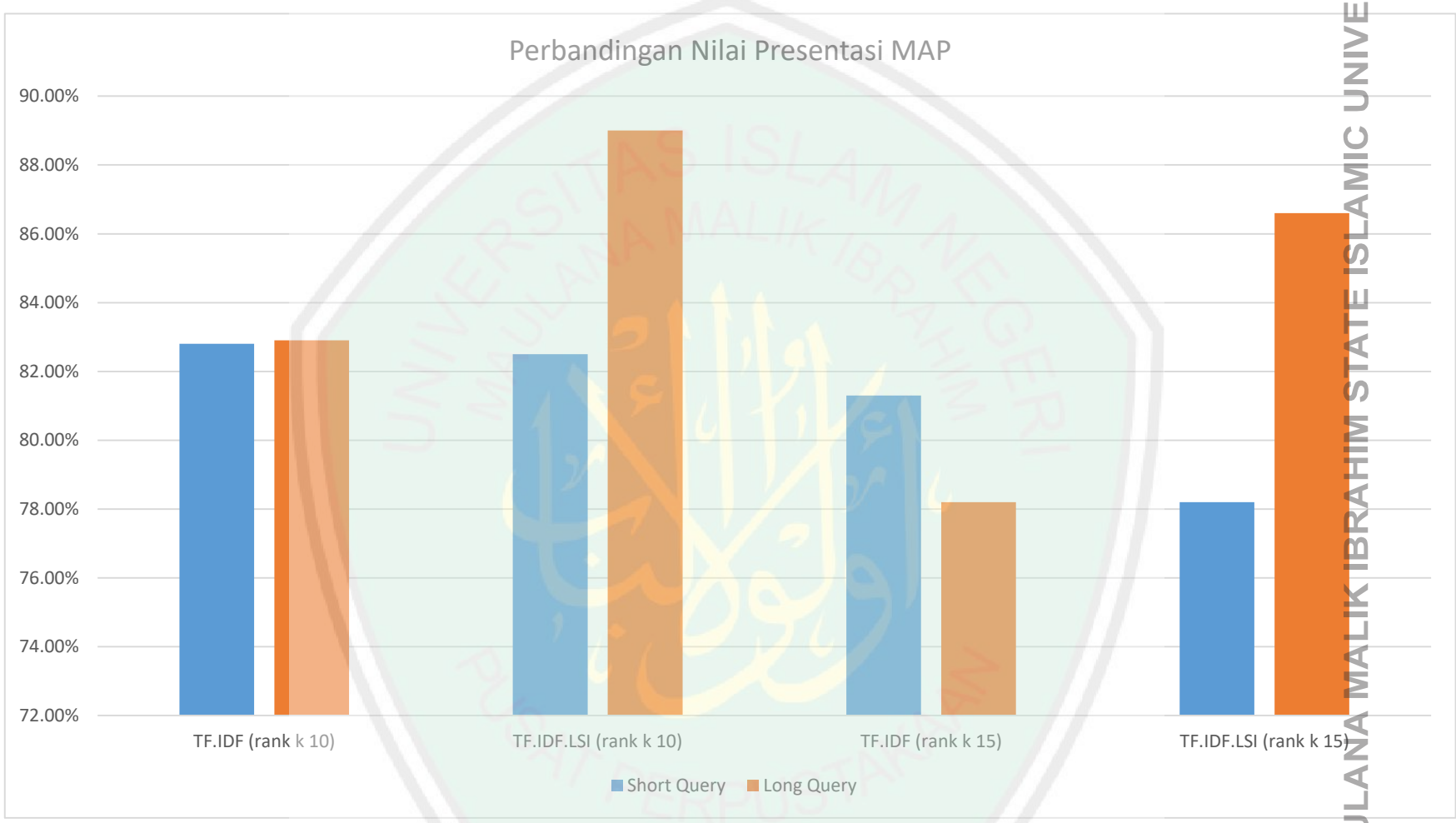
Tabel 4. 10 Presentasi Nilai MAP metode TF.IDF

Kategori *Query*	***Rank-k*=10**	***Rank-k*=15**
Short	82.8 %	81.3%
Long	82.9 %	78.2%
MAP	**82.8 %**	**79 %**

Tabel 4. 11 Presentasi Nilai MAP metode TF.IDF.LSI

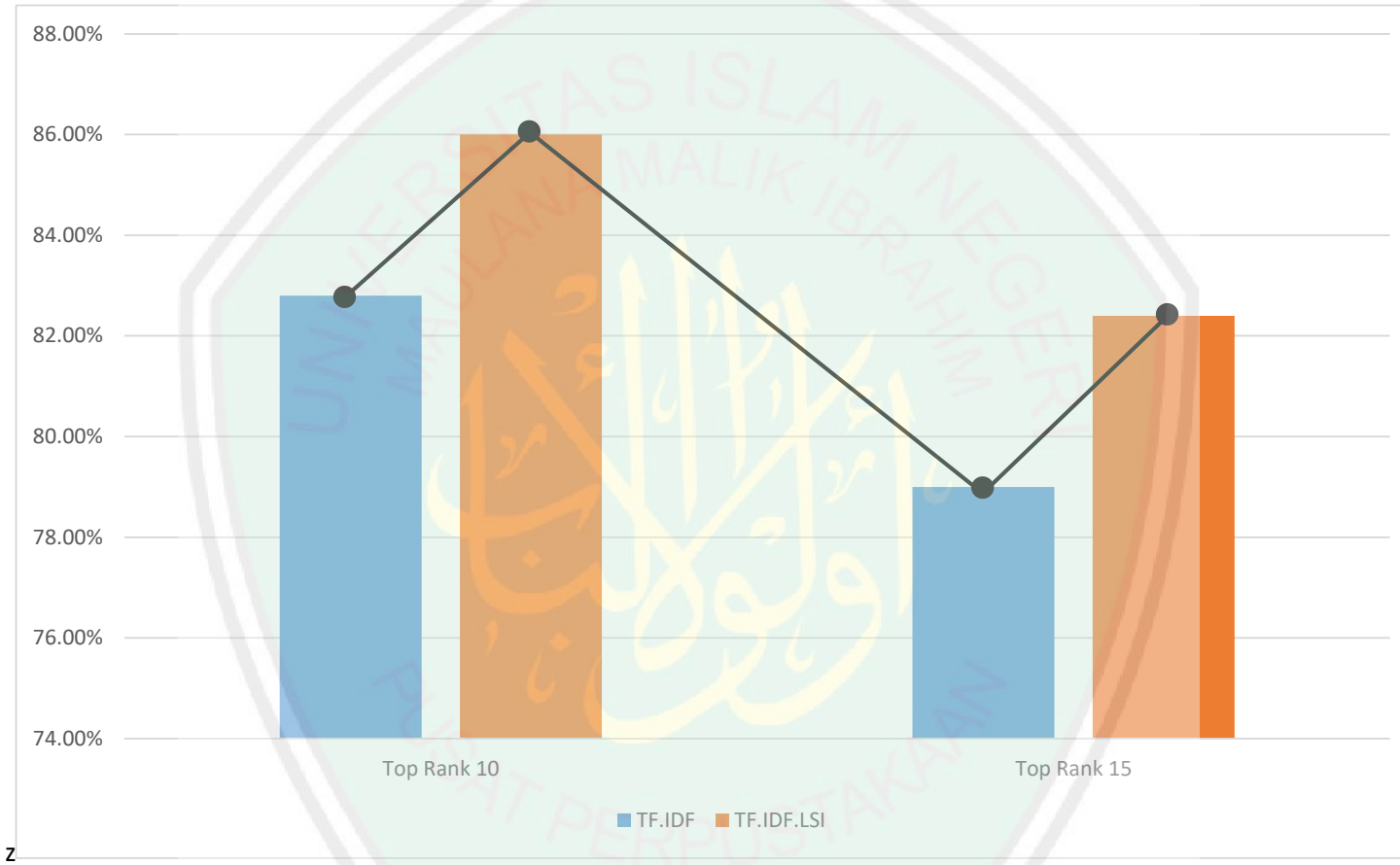
Kategori *Query*	***Rank-k*=10**	***Rank-k*=15**
Short	82.5 %	78.2 %%
Long	89 %	86.6 %
MAP	**86 %**	**82.4 %**

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG



Gambar 4. 28 Grafik presentasi nilai MAP berdasarkan jenis *query*.

LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF



Gambar 4. 29 Grafik Presentase Nilai MAP pada k = 80

UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
PUSAT PERPUSTAKAAN
LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF

4.3 Integrasi dengan Al Quran

Islam menetapkan tujuan pokok kehadirannya untuk memelihara agama, jiwa, akal, jasmani, harta, dan keturunan. Setidaknya tiga dari yang disebut di atas berkaitan dengan kesehatan yakni jiwa, akam dan jasmani. Kesehatan merupakan hal yang sangat diperhatikan dalam ajaran islam. Karena kesehatan menjadi modal awal untuk beribadah kepada Allah secara optimal. Anjuran untuk menjaga kesehatan sesuai dengan Firman Allah SWT surat Al-Baqarah ayat 222.

وَ يَسْـَٔلُوْنَكَ عَنِ الْمَحِيْضِ قُلْ هُوَ اَذًى فَاعْتَزِلُوا النِّسَآءَ فِى الْمَحِيْضِ وَلَا تَقْرَبُوْهُنَّ حَتّٰى يَطْهُرْنَ فَاِذَا تَطَهَّرْنَ فَاْتُوْهُنَّ مِنْ حَيْثُ اَمَرَكُمُ اللّٰهُ اِنَّ اللّٰهَ يُحِبُّ التَّوَّابِيْنَ وَيُحِبُّ الْمُتَطَهِّرِيْنَ ﴿٢٢٢﴾

Artinya :

"Mereka bertanya kepadamu tentang haidh. Katakanlah: Haidh itu adalah suatu kotoran. Oleh sebab itu hendaklah kamu menjauhkan diri dari wanita di waktu haidh; dan janganlah kamu mendekati mereka, sebelum mereka suci. Apabila mereka telah Suci, Maka campurilah mereka itu di tempat yang diperintahkan Allah kepadamu. Sesungguhnya Allah menyukai orang-orang yang bertaubat dan menyukai orang-orang yang mensucikan diri."

Dalam ayat diatas dijelaskan bahwa sebagai orang islam diharuskan untuk menjaga kebersihan, bahkan dalam hal yang berhubungan dengan badan manusia. Apabila kebersihan telah terwujud, maka secara tidak langsung akan berimbas pada kesehatan diri, dimana tidak banyak penyakit yang menghampiri badan. Bahkan

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

kebersihan tersebut juga sebagian dari harta seorang muslim. Sebagaimana Hadist berikut :

اَلنَّظَا فَةُ مِنَ الْإِيْمَانِ

Artinya :

"Kebersihan itu adalah satu sudut dari iman" (HR. Imam Ahmad dan Turmudzi).

Namun, meskipun kita sebagai muslim telah semaksimal mungkin menjaga kesehatan. Terkadang masih juga ada penyakit yang menghampiri diri kita. Karena memang hal tersebut merupakan sesuatu yang di ciptakan oleh dan dari Allah SWT. Selain penyakit, Allah juga menciptakan obat yang dapat untuk menyembuhkan penyakit tersebut. Seperti yang di jelaskan oleh oleh Allah SWT dalam Firman-NYA pada surat Yunus ayat 57 :

يٰٓاَيُّهَا النَّاسُ قَدْ جَاۤءَتْكُمْ مَّوْعِظَةٌ مِّنْ رَّبِّكُمْ وَشِفَاۤءٌ لِّمَا فِى الصُّدُوْرِۙ وَهُدًى وَّرَحْمَةٌ لِّلْمُؤْمِنِيْنَ ﴿٥٧﴾

Artinya :

“Wahai manusia! Sungguh telah, datang kepadamu pelajaran Al-Quran) dari Tuhanmu, penyembuh bagi penyakit yang ada dalam dada, dan petunjuk serta rahmat bagi orang yang beriman.”

Dari ibnu Mas’ud, bahwa Rasulullah bersabda :

إِنَّ اللهَ لَمْ يَنْزِلْ دَاءً إِلاَّ أَنْزَلَ لَهُ شِفَاءً، عَلِمَهُ مَنْ عَلِمَهُ وَجَهِلَهُ مَنْ جَهِلَهُ

"Sesungguhnya Allah Subhanahu wa Ta'ala tidaklah menurunkan sebuah penyakit melainkan menurunkan pula obatnya. Obat itu diketahui oleh orang yang bisa

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

mengetahuinya dan tidak diketahui oleh orang yang tidak bisa mengetahuinya." (HR. Ahmad, Ibnu Majah, dan Al-Hakim).

Ayat dan hadits di atas menunjukkan bahwa setiap penyakit pasti ada obatnya, dan hendaklah manusia melakukan perawatan sakitnya atau berobat kepada yang mengetahuaninya atau ahlinya. Tetapi obat dan dokter hanyalah cara kesembuhan, sedangkan kesembuhan hanya datang dari Allah. Karena Allah menyatakan, "Dialah yang menciptakan segala sesuatu." Semujarab apapun obat dan sehebat apapun dokternya, namun jika Allah tidak menghendaki kesembuhan, maka kesembuhan itu tidak akan didapat. Terdapat beragam cara yang bisa di gunakan dalam pencari pengobatan. Keanekaragaan tersebut sebagaimana di jelaskan dalam surat Ar Rum ayat 13.

وَمِنْ اٰيٰتِهٖ خَلْقُ السَّمٰوٰتِ وَالْاَرْضِ وَاخْتِلَافُ اَلْسِنَتِكُمْ وَاَلْوَانِكُمْ اِنَّ فِيْ ذٰلِكَ لَاٰيٰتٍ لِّلْعٰلِمِيْنَ ﴿٢٢﴾

Artinya :

“Dan di antara tanda-tanda kekuasaan-Nya ialah menciptakan langit dan bumi dan berlain-lainan bahasamu dan warna kulitmu. Sesungguhnya pada yang demikan itu benar-benar terdapat tanda-tanda bagi orang-orang yang mengetahui.”

Dalam hal mencari pengobatan, ada banyak cara yang bisa digunakan yakni salah satunya dengan mencari metode pengobatan dari artikel-artikel kesehatan yang di tulis oleh orang ahli di bidangnya.

Hasil akhir dari penelitian ini adalah perangkingan artikel kesehatan berbahasa Indonesia berdasarkan kesesuaiannya dengan *query* yang dimasukkan oleh pengguna. Hasil tersebut diperoleh dari menghitung *similarity* antara bobot

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

query dengan bobot dokumen artikel. Semakin besar nilai kesamaan atau semakin banyak ciri-ciri yang sesuai, maka semakin tinggi nilai *similarity* yang di hasilkan sehingga semakin dianggap mirip juga. Hal tersebut juga sesuai pada surat Al Hujuraat ayat 13 :

يٰٓاَيُّهَا النَّاسُ اِنَّا خَلَقْنٰكُمْ مِّنْ ذَكَرٍ وَّاُنْثٰى وَجَعَلْنٰكُمْ شُعُوْبًا وَّقَبَاۤىِٕلَ لِتَعَارَفُوْا ۚ اِنَّ اَكْرَمَكُمْ عِنْدَ اللّٰهِ اَتْقٰىكُمْ ۗ اِنَّ اللّٰهَ عَلِيْمٌ خَبِيْرٌ ﴿١٣﴾

Artinya :

"Hai manusia, sesungguhnya Kami menciptakan kamu dari seorang laki-laki dan seorang perempuan dan menjadikan kamu berbangsa-bangsa dan bersuku-suku supaya kamu saling kenal-mengenal. Sesungguhnya orang yang paling mulia di antara kamu di sisi Allah ialah orang yang paling taqwa di antara kamu. Sesungguhnya Allah Maha Mengetahui lagi Maha Mengenal."

Ayat di atas menjelaskan tentang keberagaman, dan setiap keberagaman tersebut pasti memiliki ciri-ciri agar mudah di kenal. Seperti halnya pada penelitian ini yang menggunakan metode untuk mencari persamaan antara *query* dengan dokumen artikel dari ciri-ciri yang di miliki oleh setiap artikel dokumen dan ciri-ciri yang dimiliki oleh *query*. *Similarity* dokumen artikel di nilai dari frekuensi kata yang terdapat pada suatu dokumen artikel, semakin banyak frekuensi kata yang dikandung maka semakin tinggi nilai *similarity*-nya dan dokumen artikel dinyatakan sebagai dokumen artikel yang mirip.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

BAB 5

PENUTUP

Pada penilitian ini telah dilakukan perangkingan dokumen artikel kesehatan dengan menggunakan metode TF.IDF dan metode TF.IDF.LSI. Pada bab ini akan di jabarkan kesimpulan dari hasil uji coba penelitian dan saran untuk penelitian selanjutnya.

5.1 Kesimpulan

Berdasarkan uji coba berupa input 20 *query* atau kata kunci yang telah di lakukan pada sistem yang dibangun dalam penelitian ini, nilai *Mean Average Precision* (MAP) dari metode TF.IDF.LSI adalah 82.4 % untuk *rank-k*=15 dan 86 % untuk *rank-k*=10, dimana nilai tersebut lebih tinggi sebasar 3.4 % pada *rank-k* 15 dan 3.2 % dari *rank-k*=10 dibandingkan dengan nilai pada metode TF.IDF untuk *rank-k*=15 yang memiliki nilai 79 % dan *rank-k*=10 adalah 82.8 %. Maka dapat diambil kesimpulan bahwa *rank-k*=10 memiliki nilai lebih tinggi daripada *rank-k=15* berdasarkan MAP. Yang mana dengan *rank-k*=10 perbandingan antara TF.IDF dan TF.IDF.LSI menghasilkan bahwa TF.IDF.LSI memilki nilai presisi yang lebih tinggi. Pada hasil uji coba TF.IDF.LSI lebih optimal pada jenis *query* yang panjang dan jenis dokumen yang relative panjang atau memiliki jumlah kata yang banyak.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

5.2 Saran

Berikut ini merupakan beberapa saran untuk penelitian di masa akan datang. Saran-saran ini didasarkan pada hasil perancangan, implementasi dan pengujian pada sistem. Saran-saran tersebut antara lain :

1. Data training dokumen artikel yang lebih banyak dan bervariasi.
2. Jika semakin banyak dokumen artikel yang diproses maka otomatis akan menambah lama waktu komputasinya. Yang mana untuk penelitian selanjutnya adalah pengembangan waktu komputasi yang cepat

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

DAFTAR PUSTAKA

Anand, R., & Jeffrey, D. U. (2011). Mining of massive datasets. *2011-01-03]*. *Http://Infolab. Stanford*. Retrieved from http://scholar.google.com/scholar?q=related:o4rvxS-5BtwJ:scholar.google.com/&hl=en&num=20&as_sdt=0,5

Baeza, R. Y., & Ribeiro, B. N. (1999). *Model Information Retrieval*. *ACM press, A Division of the Association for Computing* (Vol. 40). New York. http://doi.org/10.1080/14735789709366603

Bunyamin, H., & Negara, C. P. (2008). Aplikasi Information Retrieval (IR) CATA Dengan Metode Generalized *Vector* Space Model. *Jurnal Informatika*, *4*(1), 29–38.

Darmawan, H. A., & Wurijanto, T. (2010). *Rancang Bangun Aplikasi Search Engine Tafsir Al- Qur'an Menggunakan Teknik Text Mining Dengan Algoritma VSM (Vector Space Model)*. STIKOM Surabaya, Surabaya.

Feldman, R. & Sanger, J. (2007). The Text Mining Handbook. New York: Cambridge University Press.

Harjanto, D. S., Endah, N. S., & Bahtiar, N. (2012). Sistem Temu Kembali pada Dokumen Teks Menggunakan Metode Term Frequency Inverse Document Frequency (TF.IDF).

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Harrag F., A. Hamdi-Cherif, E. El-Qawasmeh. *Vector* space model for Arabic information retrieval - application to Hadith indexing. Proceedings of the First IEEE Conference on the Applications of Digital Information and Web Technologies. ICADWIT. 2008

Holle, K. F. ., Arifin, A. Z., & Purwitasari, D. (2015). Preference Based Term Weighting For Arabic Fiqh Document Rangking, *1*, 45–52.

Imbar, R. V., Ayub, M., Rehatta, A., & Adelia. (2014). Implementasi *Cosine Similarity* dan Algoritma Smith-Waterman untuk Mendeteksi Kemiripan Teks. *Jurnal Informatika*, 31–42.

Indranandita, A., Santoso, B., & Rachmat, A. (2008). Sistem Klasifikasi dan Pencarian Jurnal dengan Menggunakan Metode Naive Bayes dan *Vector* Space Model. *Jurnal Informatika*, *4*(2), 10.

Maarif, A. A. (2015). *Penerapan Algoritma TF-IDF untuk Pencarian Karya Ilmiah*. Universitas Dian Nuswantoro, Semarang.

Manning, C. D. P. R. a. H. S., 2008. Introduction to Information Retrieval. Cambridge: Cambridge University Press.

Parwita, W. G. S. (2015). Hybrid Recommendation System Memanfaatkan Penggalian Frequent Itemset dan Perbandingan Keyword, *9*(2), 19–21.

Salim, M. A., & Anistyasari, Y. (2017). *Pengembangan Aplikasi Penilaian Ujian Essay Berbasis Online Menggunakan Algoritma Nazief dan Adriani dengan*

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Metode Cosine Similarity. Universitas Negei Surabaya, Surabaya.

Sari, Y. A., & Puspaningrum, E. Y. (2015). Pencarian Semantik Dokumen Berita Menggunakan Essential Dimension of Latent Semantic Indexing dengan Memakai Reduksi Fitur Document Frequency dan Information Gain Thresholding. *Seminar Nasional Teknologi Informasi Dan Multimedia*, (July), 27–32.

T. Wicaksono. (2005). Text Mining Untuk Pencarian Dokumen Bahasa Inggris Menggunakan Suffix Tree Clustering. *Eepis Final Project*.

Triana, A., Saptono, R., & Sulistyo, M. E. (2014). Pemanfaatan Metode *Vector* Space Model Dan *Cosine Similarity* Pada Fitur Deteksi Hama Dan Penyakit Tanaman Padi. *Jurnal ITSMART*, 1–6.

Wahib, A., Pasnur, Santika, P. P., & Arifin, A. Z. (2015). Perangkingan Dokumen Berbahasa Arab Menggunakan Latent Semantic Indexing. *Jurnal Buana Informatika*, *6*(2), 83–92. Retrieved from https://ojs.uajy.ac.id/index.php/jbi/article/view/411

Wahyuni, R. T., & dkk. (2017). Penerapan Algoritma *Cosine Similarity* dan Pembobotan TF-IDF pada Sistem Klasifikasi Dokumen Skripsi. *Jurnal Teknik Elektro*, *9*(1).

Wardhana, S. R., Yunianto, D. R., Arifin, A. Z., & Purwitasari, D. (2015). Pembobotan Kata Berbasis Preferensi Dan Hubungan Semantik Pada Dokumen Fiqih Berbahasa Arab. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *2*(2), 132–137.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG

Wibisono, S., & Utomo, M. S. (2013). *Perancangan Aplikasi Web Scraping Untuk Koleksi Konten Resep Masakan Tradisional Jawa Berbasis XML*. Universitas Stikubank, Semarang.

CENTRAL LIBRARY OF MAULANA MALIK IBRAHIM STATE ISLAMIC UNIVERSITY OF MALANG