Jurnal SimanteC
Vol. 7, No. 2 Juni 2019 ISSN 2088-2130

RANCANGAN DAN IMPLEMENTASI APLIKASI PENCARIAN TEKS TERJEMAHAN AL QUR'AN BERBASIS MODEL RUANG VEKTOR

Husni[1], Bakhtiyar Arifin[2]
[1,2] Program Studi Informatika, Jurusan Teknik Informatika, Fakultas teknik
Universitas Trunojoyo Madura
Jl. Raya Telang, PO BOX 2, Kamal, Bangkalan – 69162
Email: [1]husni@truojoyo.ac.id

ABSTRAK

Al Qur'an adalah kitab suci agama Islam yang di dalamnya terkandung aturan dan pedoman hidup sehingga menjadi rujukan utama saat memerlukan solusi. Tidak banyak masyarakat Indonesia yang memahami Al Quran dalam bahasa Arab, sehingga menjadikan terjemahannya sebagai acuan dalam memahami kandungan kitab ini. Selain itu, susunan isi dari Al Quran yang menempatkan bahasan atau topik secara menyebar, membuat pencarian suatu topik mendapatkan jawaban di banyak tempat, tersebar di antara halaman pertama sampai terakhir yang berjumlah 6236 ayat. Paper ini melaporkan penelitian yang diakukan dengan menerapkan teknik penambangan teks terhadap teks terjemahan Al Quran bahasa Indonesia untuk membantu mempermudah pencarian referensi berupa ayat dan terjemahan berrdasarkan dalam permasalahan yang ingin diketahui solusinya. Secara garis besar, sistem yang dihasilkan menyerupai Search Engine spesfisik terjemahan Al Quran. Proses kerja dari sistem ini terbagi ke dalam dua bagian, yaitu preprocesing dan query processing. Preprocessing meliputi tokenisasi, penghapusan sop-word, stemming, pembauatan indeks dan pembobotan. Query processing lebih fokus pada perhitungan kemiriapan antara kata kunci dari pengguna dengan koleksi teks terjemahan menggunakan kemiripan kosinus. Sistem yang dihasilkan mampu memberikan recall sampai 100% meskipun dari sisi presisi masih di bawah 70% dan telah mampu mempermudah pencarian topik tertentu oleh pengguna.

Kata Kunci : *Kemiripan Kosinus, Penambangan Teks, Pencarian Ayat Al Qur'an, Preprocessing Teks.*

ABSTRACT

Al Qur'an is the holy book of Islam which contains rules and guidelines for life so that it becomes the main reference when needing a solution. There are not many Indonesians who understand the Qur'an in Arabic, so it makes the translation as a reference in understanding the contents of this book. In addition, the composition of the contents of the Quran that puts the discussion or topic in a scattered manner, making the search for a topic to get answers in many places, spread between the first page to the last, amounting to 6236 verses. This paper reports on research undertaken by applying text mining techniques to the Indonesian translation of the Al Quran text to help facilitate the search for references in the form of verses and translations based on the problem of which the solution is sought. Broadly speaking, the resulting system resembles a specific search engine translation of the Book. The work process of this system is divided into two parts, namely preprocesing and query processing. Preprocessing includes tokenization, sop-word removal, stemming, indexing and weighting. Query processing is more focused on calculating the similarity between the keywords of the user and the collection of translated texts using cosine similarity. The resulting system is able to provide up to 100% recall although in terms of precision it is still below 70% and has been able to facilitate the search for specific topics by users.

Keywords: *The Similarity Of Cosines, Mining The Text, Searching The Verses Of The Quran, Preprocessing Text.*

PENDAHULUAN

Sebagai seorang muslim menjalani kehidupan sesuai dengan aturan yang ada di Al Qur'an adalah suatu kewajiban. Dalam menghadapi suatu permasalahan sebaiknya kita harus berpegang pada Al Qur'an sebagai acuan penyelesaian. Kita diwajibkan mempelajari Al Qur'an dan mengamalkannya dalam kehidupan sehari – hari. Tetapi kadang kita mengalami kesulitan dalam mempelajari dan mencari ayat – ayat yang sesuai dengan permasalahan yang kita hadapi. kesulitan menemukan ayat – ayat Al Qur'an yang sesuai dengan permasalahan, dikarenakan bentuk Al Qur'an yang konvensional sulit dipelajari. Ayat – ayat dalam Al Qur'an disusun tidak berdasarkan sebuah permasalahan, sehingga sifatnya yang berpencar memakan waktu lama untuk pencarian ayat – ayat yang dibutuhkan untuk pemecahan suatu masalah [1].

Paper ini akan melaporkan hasil penelitian yang telah dilakukan berupa suatu sistem yang dapat mengenali, mencari dan mengelompokkan permasalahan yang ada sehingga sistem dapat menampilkan ayat – ayat Al Qur'an dan terjemahannya sebagai referensi dan solusi dari permasalahan yang ditanyakan oleh pengguna. Sistem menggunakan sederet teknik yang terdapat di dalam Penambangan Teks (*text mining*). Pembangunan sistem ini memerlukan beberapa method seperti parsing, steaming, dan morphing hingga dapat mengenali kebutuhan infomasi pengguna dengan lebih baik. Penerapan konsep text preprocessing dan perhitungan kemiripan menggunakan cosine similarity menjadi titik-titik yang sangat penting di dalam sistem ini.

Bagian selanjutnya dari paper ini akan menjabarkan mengenai teori yang mendasari seperti penambangan teks, preprocessing, model ruang vektor, kemiripan kosinus. Pembahasan mengenai sistem yang dihasilkan akan mengikutinya dan ditutup dengan suatu kesimpulan.

PENAMBANGAN TEKS

Penambangan Teks dapat diartikan sebagai proses menambang data berupa teks dimana sumber data biasanya diperoleh dari dokumen dan tujuannya adalah untuk mencari kata-kata (*term*) yang dapat mewakili isi dari dokumen sehingga dapat dilakukan analisis keterhubungan antar dokumen.

Tahapan-tahapan dalam teks mining adalah sebagai berikut[2] :

1. *Tokenizing*
2. *Filtering*
3. *Stemming*
4. *Tagging* (pada penelitian ini tidak dilakukan proses tagging karena tidak melakukan penanganan untuk bahsa inggris)
5. *Analyzing* (merupakan tahap penentuan seberapa jauh keterhubungan antar kata-kata antar dokumen, yang akan dijelaskan dalam sub bab berikutnya)

Stemming

Stemming merupakan suatu proses untuk menemukan kata dasar dari sebuah kata. Dengan menghilangkan semua imbuhan (affixes) baik yang terdiri dari Prefiks (prefixes), sisipan (infixes), akhiran (suffixes) dan confixes (kombinasi dari Prefiks dan akhiran) pada kata turunan. Stemming digunakan untuk mengganti bentuk dari suatu kata menjadi kata dasar dari kata tersebut yang sesuai dengan struktur morfologi Bahasa Indonesia yang baik dan benar. Metode stemming memerlukan input berupa term yang terdapat dalam dokumen. Sedangkan outputnya berupa stem. Algoritma ini didahului pembacaan tiap kata dari file sampel.

Menurut Arifin dan Setiono [3], proses stemming adalah sebagai berikut:

a. Pemeriksaan semua kemungkinan bentuk kata. Setiap kata diasumsikan memiliki 2 Prefiks (prefiks) dan 3 akhiran (sufiks). Sehingga bentuknya menjadi:

   Prefiks 1 + ***Prefiks 2*** + ***Kata dasar*** + ***Sufiks 3*** + ***Sufiks 2*** + ***Sufiks 1***

   Jika kata tersebut tidak memiliki imbuhan sebanyak imbuhan diatas, maka imbuhan yang kosong diberikan tanda x untuk prefiks dan diberi tanda xx untuk sufiks
b. Pemotongan dilakukan berurutan : Prefiks 1, Prefiks 2, Sufiks 1, Sufiks 2, Sufiks 3 (jika ada), dan Kata Dasar.
c. Setiap tahap pemotongan diikuti dengan pemeriksaan di database (berisi daftar kata dasar) apakah hasil pemotongan itu sudah berada dalam bentuk dasar. Jika pemeriksaan berhasil maka proses dinyatakan selesai dan tidak perlu melanjutkan proses pemotongan imbuhan selanjutnya.
d. Namun jika sampai pada pemotongan Sufiks 3 masih belum juga ditemukan di kamus, maka dilakukan proses kombinasi: Kata Dasar yang dihasilkan dikombinasikan dengan

   1. Kata Dasar
   2. Kata Dasar + Sufiks 3

e. imbuhan-imbuhannya dalam 12 konfigurasi berikut:

3. Kata Dasar + Sufiks n 3 + Sufiks 2
4. Kata Dasar + Sufiks 3 + Sufiks 2 + Sufiks 1
5. Prefiks 1 + Prefiks 2 +Kata Dasar
6. Prefiks 1 + Prefiks 2 +Kata Dasar + Sufiks 3
7. Prefiks 1 + Prefiks 2 +Kata Dasar + Sufiks 3 + Sufiks 2
8. Prefiks 1 + Prefiks 2 +Kata Dasar + Sufiks 3 + Sufiks 1
9. Prefiks 2 + Kata Dasar
10. Prefiks 2 + Kata Dasar + Sufiks 3
11. Prefiks 2 + Kata Dasar + Sufiks 3 + Sufiks 2
12. Prefiks 2 + Kata Dasar + Sufiks 3 + Sufiks 2 + Sufiks 1

Sebenarnya kombinasi 1, 2, 3, 4, 8, dan 12 sudah diperiksa pada tahap sebelumnya, karena kombinasi ini adalah hasil pemotongan bertahap tersebut. Dengan demikian, kombinasi yang masih perlu dilakukan tinggal 6 yakni pada kombinasi-kombinasi yang belum dilakukan (5, 6, 7, 9, 10, dan 11). Tentunya bila hasil pemeriksaan suatu kombinasi adalah 'ada', maka pemeriksaan pada kombinasi lainnya sudah tidak diperlukan lagi. Pemeriksaan 12 kombinasi ini diperlukan, karena adanya fenomena overstemming pada algoritma pemotongan imbuhan. Kelemahan ini berakibat pada pemotongan bagian kata yang sebenarnya adalah milik Kata Dasar itu sendiri yang kebetulan mirip dengan salah satu jenis imbuhan yang ada. Dengan 12 kombinasi itu, pemotongan yang sudah terlanjur tersebut dapat dikembalikan sesuai posisinya.

Pembobotan

Metode Tf-Idf merupakan suatu cara untuk memberikan bobot hubungan suatu kata (*term*) terhadap dokumen. Metode ini menggabungkan dua konsep untuk perhitungan bobot yaitu, frekuensi kemunculan sebuah kata didalam sebuah dokumen tertentu dan inverse frekuensi dokumen yang mengandung kata tersebut. Frekuensi kemunculan kata didalam dokumen yang diberikan menunjukkan seberapa penting kata tersebut didalam dokumen tersebut. Frekuensi dokumen yang mengandung kata tersebut menunjukkan seberapa umum kata tersebut. Sehingga bobot hubungan antara sebuah kata dan sebuah dokumen akan tinggi apabila frekuensi kata tersebut tinggi didalam dokumen dan frekuensi keseluruhan dokumen yang mengandung kata tersebut yang rendah pada kumpulan dokumen (database).

Rumus umum untuk Tf-Idf :

$$w_{ij} = tf \times idf$$

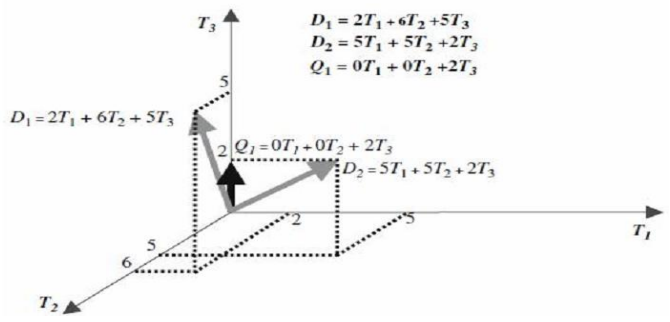
$$w_{ij} = tf_{ij} \times \log \frac{N}{n}$$

Keterangan :
W*ij* = bobot kata/term t*j* terhadap dokumen d*i*
Tf*ij* = jumlah kemunculan kata/term t*j* dalam dokumen d*i*
N = jumlah semua dokumen yang ada dalam database
n = jumlah dokumen yang mengandung kata/term t*j*

Model Ruang Vektor

Model ruang vektor (*vector space model*, VSM) adalah suatu model yang digunakan untuk mengukur kemiripan antara suatu dokumen dengan suatu kueri [10]. Kueri dandokumen diangggap sebagai vector-vektor dalam ruang n-dimensi, dimana t adalah jumlah seluruh *term* dalam tabel index. Selanjutnya akan dihitung nilai cosines sudut dari dua vector, yaitu W dari tiap dokumen dan W dari kata kunci.

Contoh dari VSM tiga dimensi untuk dua dokumen D1 dan D2, satu kueri pengguna Q1, dan tiga term T1, T2 dan T3 diperlihatkan pada gambar

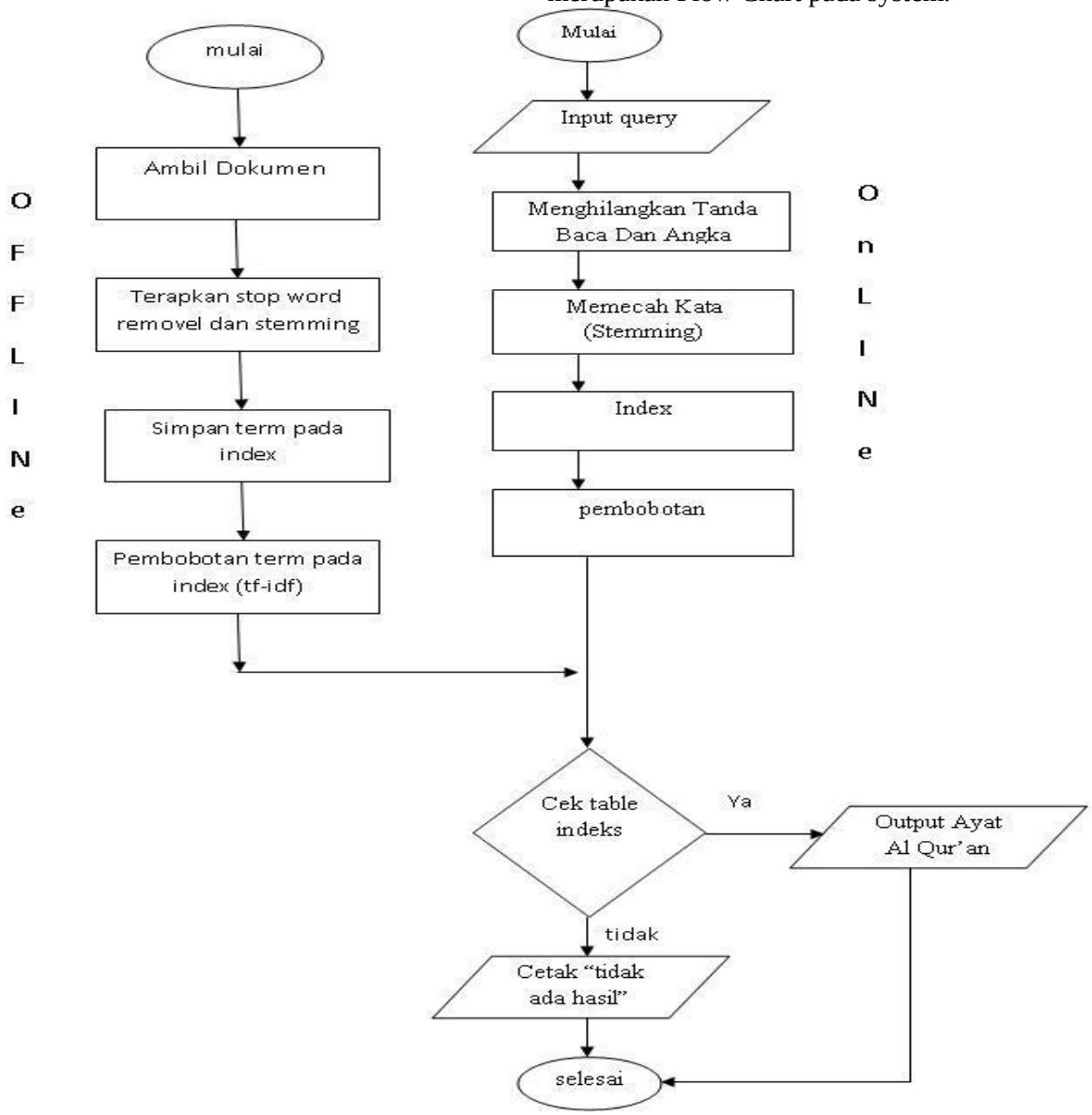


1.

Gambar 1. Vektor Space Model

METODE

Metode yang digunakan dalam penelitian ini dimulai dengan pembobotan dokumen menggunakan metode tf/idf. Proses dimulai dengan ekstraksi yang bertujuan untuk mendapatkan term-term dari tiap dokumen. Term dokumen akan diproses melalui tokenizing, filtering dan stemming untuk mendapatkan integrasi antar term frequensi (tf), dan inverse dokumen frequensi (idf). Vector space model digunakan sebagai metode pencaian. Kemiripan dokumen akan diperoleh dari pembandingan bobot kueri dengan bobot kata pada dokumen. Dokumen dengan bobot kemiripan tertinggi berada pada urutan teratas. Berikut merupakan Flow Chart pada system.



Gambar 2. *Flow Chart* dari sistem Pencarian Terjemahan Teks Al Quran

Secara garis besar, terdapat dua proses yaitu proses yang dilakukan secara *online* dan *offline.* Kedua proses ini dapat pula dinamakan *text preprocessing* dan *query processing*.

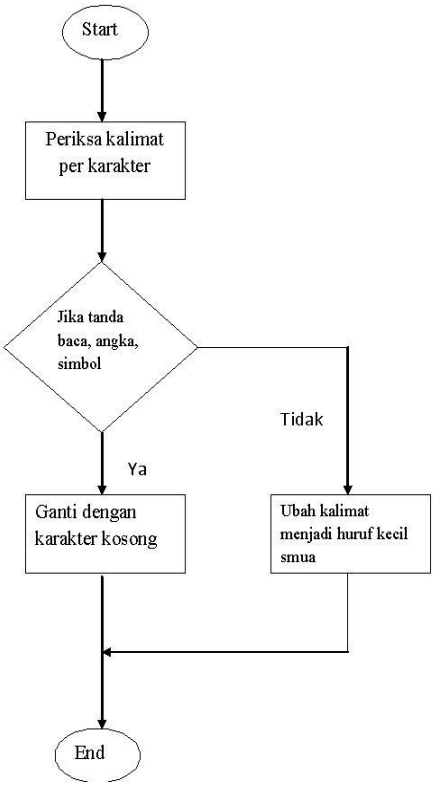
Proses secara *offline* sebagai berikut :

1. Pengambilan data dari database.
2. Preprocessing, yaitu penerapan *Stopword Removel* dan stemming pada data.
3. Data hasil preprocessing di simpan dalam tabel indek.
4. Dilakukan pembobotan untuk setiap hasil pada tabel indek.

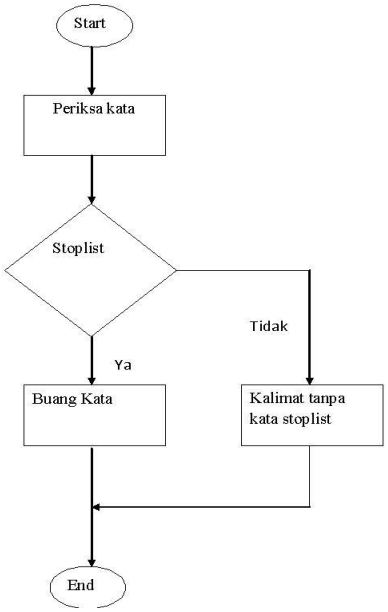
Sedangkan proses secara *online* adalah sebagai berikut :

1. Dimulai dengan peng-*inputan keyword.*
2. Menghilangkan tanda baca pada *keyword.*
3. Penerapan stemming pada *keyword.*
4. Didapat indek dari *keyword*
5. Dan pemberian bobot untuk *keyword.*
6. Pencarian dilakukan dengan membandingkan indek *keyword* dengan data dalam tabel indek dengan perhitungan Cosine Similarity.
7. Jika sesuai akan ditampilkan hasil jika tidak pesan tidak ada hasil.
   Selesai.

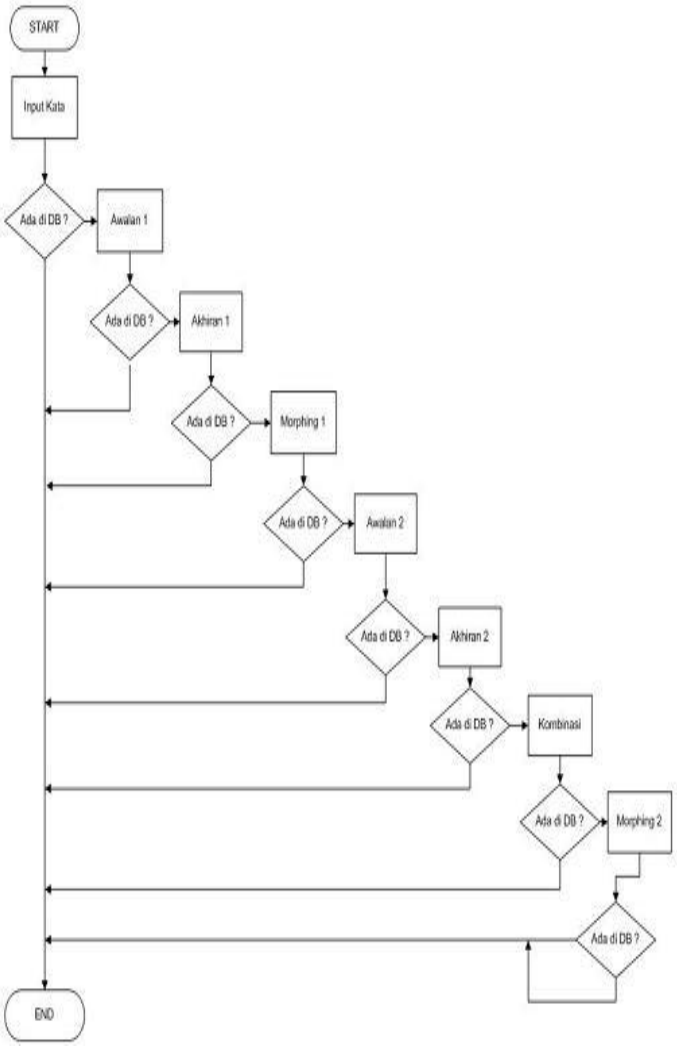
Untuk memperjelas kinerja system berikut merupakan *flow chart* dari proses *Penambangan Teks*.



Gambar 3. *Flow chart tokenizing*



Gambar 4. *Flow chart filtering*



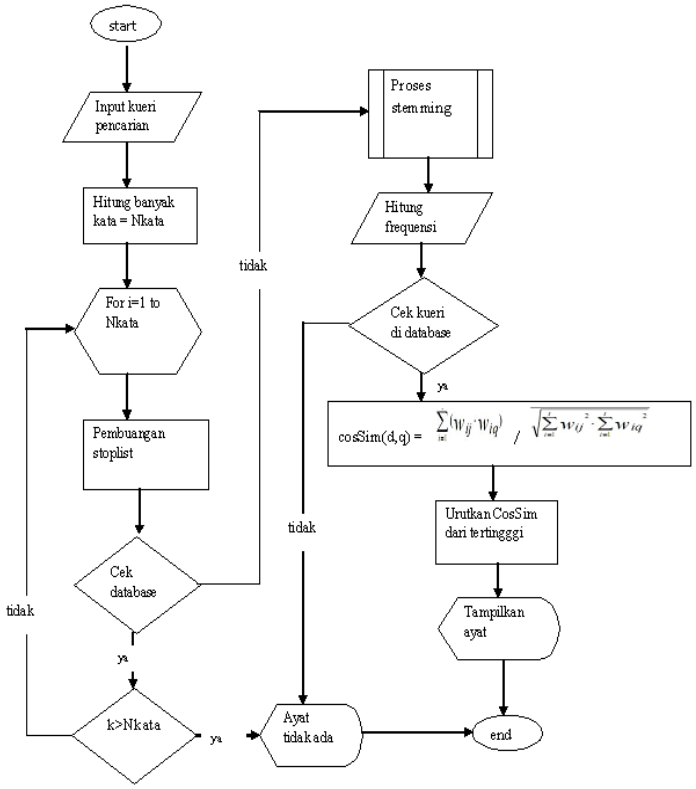
Gambar 5. *Flow chart Stemming*

Gambar 6. *Flow chart query processing: Cosine Similarity*

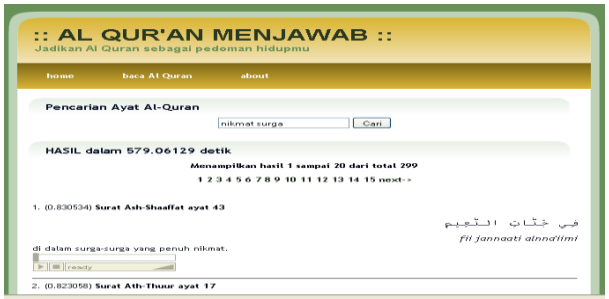
HASIL DAN PEMBAHASAN

Untuk memastikan bahwa sistem berjalan sebagaimana mestinya dilakukan uji coba pencarian ayat, dengan scenario sebagai berikut.

1. Pencarian menggunakan satu kata berimbuhan. Skenario ini dilakukan untuk mengetahui keberhasilan dari proses *stemming* pada sistem apakah bisa mengenali kata berimbuhan ataukah tidak



2. Pencarian dengan menggunakan dua kata mengandung imbuhan.
3. Pencarian dengan tiga kata.
4. Pencarian dengan kalimat terstruktur.
5. Uji coba juga dilakukan untuk mengetahui perbedaan kecepatan pencarian setelah data pencarian masuk dalam tabel cache dan sebelum data pencarian masuk dalam tabel cache.
6. Uji coba untuk mengetahui relevansi dari hasil yang ditampilkan oleh sistem.



Gambar 7. Tampilan sistem

Pada gambar 7 dapat dilihat *screenshot* dari pencarian dengan *keyword* ‘nikmat surga’.

Berikut merupakan hasil dari uji coba yang ditampilkan dalam tabel,dapat dilihat pada tabel 1.

Tabel 1. Hasil uji coba

No	Kueri	Hasil (ayat)	Waktu (detik)
1	Iman	609	0,19766
2	Pohon zaqqum	47	0,02508
3	Tuhan semesta alam	1044	1,77743
4	Kehancuran dunia sat kiamat	348	1,65742
5	Orang beriman takut masuk neraka	963	1,6987
6	Kenikmatan surga yang dijanjikan oleh Allah	2393	2,11268
7	Malaikat diciptakan dari cahaya setan api	468	1,8529
8	Syarat melaksanakan ibadah haji orang yang mampu harta	170	1,85374
9	Malaikat surga neraka langit bumi pohon zaqqum ibli api	1075	2,32583
10	Hokum islam makanan halal haram puasa zakat warisan sholat hujan	388	2,57408

Dari uji coba diatas dapat diketahui bahwa sistem mampu bekerja sesuai dengan harapan pengguna dan dapat terlihat jelas bahwa semakin panjang *keyword* dan semakin banyak ayat yang didapatkan maka akan membutuhkan waktu pencarian yang lebih lama. Hal itu disebabkan karena proses perhitungan kemiripan dilakukan pada banyak dokumen dan kata dari *keyword*.

Uji coba juga dilakukan untuk mengetahui relevansi dari hasil yang ditampilkan oleh sistem. Pencarian dengan *keyword* ‘cara haji’. Jika yang dimaksud dari pencarian tersebut diatas adalah ayat-ayat yang berhubungan dengan tata cara haji maka terdapat 8 ayat yang relevan terhadap *keyword* yang dimaksudkan maka didapatkan hasil dari pencarian dengan *keyword* cara haji sebanyak 12 ayat. Terdapat 8 ayat yang relevan, Sehingga bisa dihitung nilai recall dan *precision*-nya

Recall = 8/8 * 100% = 100%

Precision = 8/12 * 100% = 66.66 %

SIMPULAN

Penelitian yang menerapkan teknik dalam penambangan teks, terutama teks preprocessing dan kemiripan kosinus ini, dapat dikatakan bahwa proses stemming pada sistem cukup berhasil dengan

bukti sistem dapat menentukan masalah dari *keyword* yang ada. Dengan melakukan proses indeksing dengan pembobotan TF/IDF dalam suatu model *Vector* space diperoleh pengembalian terjemahan ayat Al Quran yang sesuai dengan *keyword* yang ada dimana *recall* sangat baik meskipun presisi masing belum memuaskan. Dari sisi waktu, sistem ini mampu menampilkan hasil pencarian dalam waktu yang cukup singkat.

DAFTAR PUSTAKA

[1] Anonim, "Isi Kandungan Alquran : Aqidah, Ibadah, Akhlak, Hukum, Sejarah & Dorongan Untuk Berfikir - Garis Besar / Inti Sari Al-Quran", 2006.<URL:http://organisasi.org/isi_kandungan_alquran_aqidah_ibadah_akhlak_hukum_sejarah_dorongan_untuk_berfikir_garis_besar_inti_sari_al_quran> diakses 02 September 2010

[2] Christanty, M. H. Text Mining, <URL:http://lecturer.eepis-its.edu/~iwanarif/kuliah/dm/6Text%20Mining.pdf > diakses 16 Juli 2011

[3] Arifin, A. Z., dan Setiono A. N., "Klasifikasi Dokumen Berita Kejadian Berbahasa Indonesia dengan Algoritma Single Pass Clustering". Jurusan Teknik Informatika, Fakultas Teknologi Informasi Institut Teknologi Sepuluh Nopember (ITS), 2002 <URL:http://www.its.ac.id/personal/files/.../667-agusza- ITIAKlasifikasiEvent.pdf > diakses 02 September 2010

[4] Rizki, Membangun Web Service Aplikasi Kamus Indonesia - Inggris.2007 <URL:http:// www.eepis-its.edu/uploadta/datata/ kamus_berbasis_web.pdf > diakses 02 September 2010

[5] Agusta, Ledy, "perbandingan algoritma stemming porter dengan algoritma nazief & adriani untuk stemming dokumen teks bahasa indonesia". Universitas Kristen Satya wacana, November, 2009. <URL:http://yudiagusta.files.wordpress.com/…/196-201-knsi09-036-perbandingan-algoritma-stemming-porter-dengan-algoritma-nazied-adriani-untuk-stemming-dokumen-teks-bahasa-indonesia > diakses 02 September 2010

[6] Sukamto, Rosa Ariani, "Penguraian Bahasa Indonesia Menggunakan Pengurai Collins". Institut Tehnologi Bandung, 2009. <URLhttp://www.gangsir.com/download/Tesis-Rosa-23507024.pdf > diakses 02 September 2010

[7] Asian J., Williams H. E. dan Tahaghogi, S.M.M., "Stemming Indonesian", Melbourne, RMIT University. 2005.

[8] Lanin, I dan Hardiyanto,R. 2009. Bahasa dan terjemahan Indonesia. <URLhttp://www.bahtera.org/kateglo/?mod=dictionary&action=view&phrase=kamus>, diakses 27 Januari 2011.

[9] Wibisono, Y. 2008. Stop words Untuk Bahasa Indonesia. <URLhttp://yudiwbs.wordpress.com/2008/07/23/stop-word-untuk-bahasa-indonesia/>, diakses 27 januari 2011.

[10] Harjono, K. D. Perluasan Vektor Pada Metode Search Vektor Space. Integral Vol. 10 No.2, Juli 2005 Jurusan Ilmu Komputer, Universitas Katolik Parahyangan, Bandung. <URLhttp://home.unpar.ac.id/~integral/Volume%2010/integral%2010%20No.%202/Perluasan%20Vektor.pdf> diakses 30 Juni 2011

[11] Halim, L. Jalan Pintas Menjadi Master PHP. Yogyakarta : lokomedia.2009

[12] Intan, R dan Defeng, A. "Hard: Subject-Based Search Engine Menggunakan Tf-Idf Dan Jaccard's Coefficient". Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Kristen Petra Surabaya. <URL:http://puslit2.petra.ac.id/ejournal/index.php/ind/article/viewPDFInterstitial/16502/16494>, diakses 20 juni 2011.

[13] Husni. "Unified Messaging System Information Retrieval & Klasifikasi Teks". <URL:http://komputasi.files.wordpress.com/2010/01/umsirclassification.pdf> diakses 30 Juni 2011