IMPLEMENTASI PERINGKASAN KONTEN UTAMA HALAMAN WEB DENGAN ALGORITMA *HYBRID HIDDEN MARKOV MODEL EXTRACTION METHOD* DALAM PENERAPAN PADA *INFORMATION RETRIEVAL*

Alfian Akbar Gozali[1], Imelda Atastina[2]
[1,2]*Fakultas Informatika, Institut Teknologi Telkom*
Jl. Telekomunikasi, Dayeuhkolot, Bandung 40257
E-mail: panggil.aku.ian@gmail.com, imd@ittelkom.ac.id

ABSTRAK
Pencarian dokumen di Internet memiliki karakteristik khusus yang harus dipertimbangkan yaitu bandwidth atau kecepatan akses yang terbatas serta waktu pencarian relatif lebih lambat daripada pencarian di desktop. Karena itu perlu dilakukan indexing pada proses Information Retrieval agar dapat mempercepat dan mempermudah pencarian. Makin banyak term yang terindeks akan makin membutuhkan waktu ekstra untuk mencari sebuah term. Sehingga diperlukan metode khusus untuk memangkas jumlah term dalam indeks. Salah satunya dengan melakukan ekstraksi dokumen menggunakan algoritma Hybrid Hidden Markov Model. Metode yang dipakai dalam sistem ekstraksi ini adalah dengan melakukan pendekatan statistik yang dikombinasikan dengan pendekatan tata bahasa dan HMM Hedge sebagai model HMM.
Metode yang digunakan tersebut diharapkan dapat menyelesaikan masalah yang terjadi pada sistem Information Retrieval yang hanya menggunakan ekstraksi dokumen dengan algoritma Hidden Markov Model.

Kata kunci: hybrid hidden markov model, indexing, information retrieval, hmm hedge

1. PENDAHULUAN

1.1 Latar Belakang

Tidak ada yang dapat menangkal pertumbuhan jumlah website yang sangat cepat. Pertumbuhan website yang sangat cepat ini mempunyai konsekuensi yaitu pengguna menjadi lebih kesulitan untuk mencari website yang berhubungan dengan apa yang mereka butuhkan. Oleh karena itu pembangunan search engine untuk membuat pencarian tersebut menjadi lebih mudah mutlak diperlukan. Hal itu dapat terjadi karena search engine dapat melakukan indexing terhadap semua kata/frasa yang terdapat di dalam website. Namun cara ini membuat ironi yang lain yaitu jumlah website yang sedemikian besarnya akan berbanding lurus dengan jumlah kata yang terindeks menjadi luar biasa banyak. Lebih banyak kata yang terindeks berarti makin besar pula space hardisk yang diperlukan untuk menyimpannya. Saat ini ini masih menjadi masalah, bagaimana cara mengkompresi atau membuang kata yang terindeks secara efektif.

Terdapat beberapa metode yang digunakan untuk memecahkan masalah ini. Salah satunya adalah sistem peringkasan dengan metode Hidden Markov Model [21]. Hidden Markov Model (HMM) adalah sebuah model statistik dari sebuah sistem yang diasumsikan sebuah Markov Process dengan parameter yang tak diketahui, dan tantangannya adalah menentukan parameter-parameter tersembunyi (hidden) dari parameter-parameter yang dapat diamati[9].

Atas karakteristik itulah Hidden Markov Model dapat digunakan untuk mengekstraksi kalimat utama dari suatu paragraf di dalam suatu dokumen[15]. Dalam HMM, bagian yang dapat diamati disebut observed state sedangkan bagian yang tersembunyi disebut hidden state. HMM memungkinkan pemodelan sistem yang mengandung observed state dan hidden state yang saling terkait. Pada kasus POS tagging, observed state adalah urutan kata sedangkan hidden state adalah urutan tag[15]. Metode ini meringkas konten utama dari website terlebih dahulu sebelum dilakukan pengindeksan untuk search engine yang berkaitan. Ini karena Hidden Markov Model (HMM) dapat memprediksi pola yang paling optimal dari deretan (sekuens) state [9], yang dalam kasus ini adalah pola dari kalimat [15].

Dalam penelitian sebelumnya [21] dapat dilihat bahwa metode HMM untuk membuat ringkasan konten utama bisa mengurangi kata yang teriindeks secara signifikan yaitu hingga 56,98% dengan waktu eksekusi yang berkurang secara signifikan juga menjadi 71,95% dengan akurasi yang dapat ditolerir. Namun tidak dipungkiri hasil ringkasan yang diperoleh seringkali kehilangan makna aslinya. Selain itu pada saat digunakan dalam dalam proses pencarian, adakalanya sistem memberikan hasil akurasi sangat rendah. Hal ini diduga karena tidak adanya kata yang terkandung pada query yang diberikan, pada database hasil training sebelumnya.

1.2 Perumusan masalah

Mengacu pada latar belakang dan penelitian sebelumnya [21], terdapat beberapa masalah yang ingin diselesaikan, antara lain :

a. Bagaimana agar indeks pada database tetap diminimalisir, tanpa kehilangan makna kalimat pada dokumen aslinya sambil tetap menjaga tingkat akurasi dari sistem *Information Retrieval* yang layak ?

b. Bagaimana meningkatkan tingkat akurasi sistem IR, jika kata-kata pada query yang diberikan tidak terdaftar pada indeks sistem *Information Retrieval*?

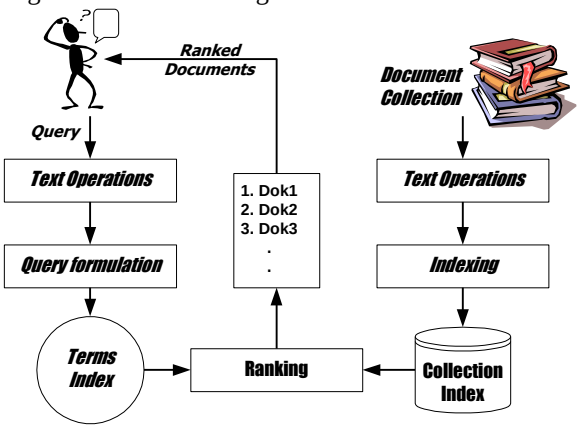
1.3 Tujuan

Tujuan dari penelitian ini adalah untuk mengusulkan sebuah metode peringkasan dengan Hidden Markov Model (HMM) baru yang dapat meminimalisir proses indexing namun masih dapat menghasilkan akurasi yang cukup walaupun kata tersebut tidak ada dalam index.

2. LANDASAN TEORI

2.1 *Information Retrieval*

Information Retrieval System (IR) atau sistem temu kembali dapat dikatakan merupakan sistem yang dapat digunakan untuk memperoleh kembali informasi yang diperlukan oleh seseorang. Pada dasarnya sistem IR dapat diterapkan pada berbagai sistem informasi selama hal tersebut berkaitan dengan proses mendapat kembali informasi yang dianggap relevan oleh seseorang, seperti sistem pencarian buku di perpustakaan. Namun yang paling sering menggunakan teknik IR ini adalah *search engine*. Sebagai sebuah sistem informasi sistem IR ini, sebenarnya terdiri dari beberapa bagian. Berikut adalah gambaran sistem IR:



Gambar 2.1. Sistem Information Retrieval [22]

Pada Gambar 2.1 tersebut terlihat bahwa user akan berinteraksi dengan sistem melalui *query*. Dimana *query* ini bisa dinyatakan dalam berbagai bentuk, mulai dari kata,kalimat bahkan artikel. Selain itu gambar di atas juga memperlihatkan bahwa terdapat dua buah alur operasi pada sistem IR. Alur pertama dimulai dari koleksi dokumen dan alur kedua dimulai dari *query* pengguna. Alur pertama yaitu pemrosesan terhadap koleksi dokumen menjadi basis data indeks tidak tergantung pada alur kedua. Sedangkan alur kedua tergantung dari keberadaan basis data indeks yang dihasilkan pada alur pertama.

Lebih rinci, dapat dikatakan sistem IR seperti yang tertera pada Gambar 2.1 adalah sebagai berikut:

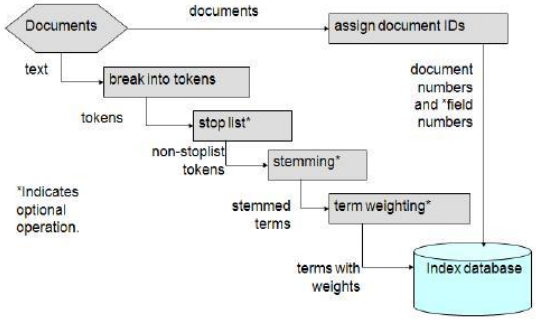
1. Text Operation, juga seringkali disebut tahapan *text preprocessing* yang pada umumnya meliputi tokenizing, *stopword removal* atau *stopword listing* dan *stemming*.
2. *Query formulation* (formulasi terhadap *query*) yaitu memberi bobot pada indeks kata-kata atau term-term *query*. Pada penelitian ini, proses formulasi terhadap query akan melibatkan bobot yang dihasilkan dari hasil training sistem IR yang diusulkan.
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. Dimana pada penelitian ini akan dilakukan setelah melakukan ekstraksi dokumen dengan Hybrid Hidden Markov Model.

2.2 Preprocessing Information Retrieval

Information Retrieval merupakan bagian dari *computer science* yang berhubungan dengan pengambilan informasi dari dokumen-dokumen yang didasarkan pada isi dan konteks dari dokumen-dokumen itu sendiri. Berdasarkan referensi[11] dijelaskan bahwa *Information Retrieval* merupakan suatu pencarian informasi (biasanya berupa dokumen) yang didasarkan pada suatu *query* (inputan *user*) yang diharapkan dapat memenuhi keinginan *user* dari kumpulan dokumen yang ada. Proses dalam *Information Retrieval* dapat digambarkan sebagai sebuah proses untuk mendapatkan *relevant documents* dari *documents collection* yang ada melalui pencarian *query* yang diinputkan user.

2.3 Indexing

Indexing merupakan sebuah proses untuk melakukan pengindeksan terhadap kumpulan dokumen yang akan disediakan sebagai informasi kepada pemakai. Data *term* yang ditemukan disimpan dalam sebuah *database* indeks untuk digunakan dalam pencarian selanjutnya. [14]



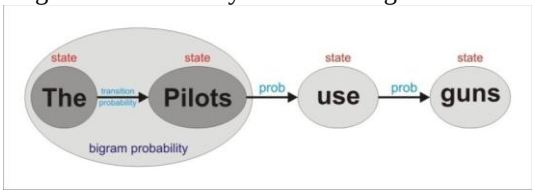
Gambar 2.2. Tahapan indexing[14]

Pada Gambar 2.2 diilustrasikan tahapan-tahapan umum dalam proses *indexing.* Beberapa tahapan di atas merupakan bagian dari tahap *preprocessing* terhadap dokumen. Proses *term weighting* (pembobotan *term* dari suatu dokumen) bersifat *optional.* Proses ini dapat dilakukan untuk meningkatkan nilai relevansi pencarian.

Term weighting dilakukan untuk mengidentifikasi mana kata-kata yang penting dan mana yang kurang penting. Identifikasi ini dilakukan untuk mempermudah dalam melakukan searching atau proses hilir dalam *Information Retrieval*(penelitian ini tidak menangani masalah searching). Proses *scoring* atau *weighting* yang dipakai dalam sistem ini adalah weighted *term frequency - inverted document frequency* (wf-idf).

2.4 *Hidden Markov Model*

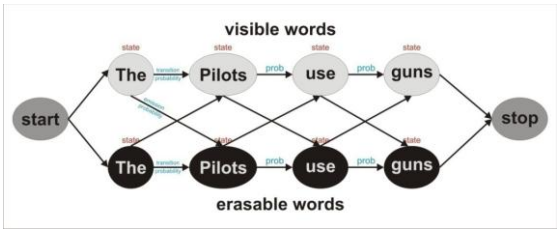
Dalam penelitian ini salah satu pokok permasalahan yang paling penting adalah bagaimana menemukan *pattern* yang tepat untuk suatu *bigram* atau dalam istilah bahasa dikenal juga dengan frasa. *Pattern* yang dimaksud adalah bagaimana probabilitas ketergantungan suatu kata itu dengan kata lain setelah dan sebelumnya. Kata-kata tersebut dapat dimodelkan dalam bentuk *state* dan hubungan antar kata dimodelkan dalam bentuk garis transisi atau gambaran umumnya adalah sebagai berikut:



Gambar 2.3. Markov Chain

Untuk pemodelan sistem seperti di atas, sistem yang paling cocok adalah *Markov Chain* karena definisi *Markov Chain* sendiri adalah sistem yang merupakan himpunan dari beberapa *state* yang berbeda[9]. Penelitian ini berfokus besarnya ketergantungan sebuah kata dengan kata lain sehingga jika ada sebuah frasa (pasangan kata) yang mempunyai ketergantungan yang rendah, bisa dihilangkan salah satu atau seluruh kata tersebut dalam kalimat. Karena pasangan kata yang mempuynai ketergantungan yang rendah biasanya bukan merupakan kata yang penting untuk membentuk kalimat. Proses penghilangan kata yang mempunyai nilai ketergantungan rendah tersebut biasa disebut juga dengan ekstraksi kalimat [12].

Dalam ekstraksi kalimat itu sendiri menggunakan model diagram transisi yang sedikit berbeda dengan gambar di atas. Selain diagram transisi antar kata, kita juga memerlukan *state* diagram transisi antar kata yang bisa dihilangkan karena ketergantungannya yang rendah. Demikian sehingga diagramnya menjadi berikut:



Gambar 2.4. Hidden Markov Model

Dalam gambar di atas bisa dilihat bahwa terdapat dua lintasan *state* yaitu *visible words* dengan *erasable words*. Daftar kata pada lintasan *visible words* bisa dilihat secara langsung itu berarti *state-state* dalam *visible words* adalah suatu "*observed state*". Sebaliknya untuk lintasan *erasable words*, *state*-nya tidak dapat diihat secara langsung. *State* tersebut dapat ditempati hanya jika ditemukan kata yang tidak penting (mempunyai ketergantungan frasa yang rendah) dalam *observed state*. Karena sifatnya yang tidak dapat dilihat secara langsung tersebut, lintasan *erasable words* merupakan suatu "*hidden state*". Dari penjelasan tersebut, pemodelan pattern yang paling cocok adalah "*Hidden Markov Model*"[15].

Hidden Markov Model (HMM) adalah suatu rantai Markov dimana simbol keluaran atau fungsi peluang yang menggambarkan simbol keluaran berhubungan dengan *state* atau transisi antar *state* [9]. Observasi tiap *state* digambarkan secara terpisah dengan suatu fungsi probabilitas atau fungsi densitas (*probability density function*, *pdf*) yang didefinisikan sebagai peluang untuk menghasilkan simbol tertentu saat terjadi transisi antar *state*. Berbeda dengan *Observable Markov Model* (OMM), HMM terdiri dari serangkaian proses stokastik rangkap yang proses utamanya tidak dapat diobservasi secara langsung (*hidden*) tetapi hanya dapat diobservasi melalui set proses stokastik lain yang menghasilkan suatu deretan observasi.

2.5 *Topic Sentence Extraction*

Topic sentence extraction atau ekstraksi kalimat utama atau dikenal juga dengan istilah umum sebagai ekstraksi kalimat adalah pemilihan kata atau frasa yang penting dalam satu kalimat dan menyusun ulang kata tersebut menjadi kalimat yang lebih ringkas. Sebaliknya, ekstraksi kalimat juga dapat dipandang sebagai proses pembuangan kata atau frasa yang tidak penting[15].

Terdapat beberapa teknik yang dapat digunakan untuk melakukan ekstraksi kalimat. Jika ditilik dari ketergantungan bahasa yang digunakan, teknik tersebut dapat dibagi menjadi dua: teknik yang bergantung pada bahasa tertentu dan teknik yang tidak bergantung pada bahasa

2.5.1 Teknik yang Bergantung pada Bahasa

Teknik yang digunakan untuk melakukan ekstraksi kalimat jenis ini adalah teknik yang

biasanya menggunakan struktur atau tata bahasa. Salah satu contohnya adalah tata bahasa atau grammar dalam bahasa Inggris. Ada beberapa peneliti yang menggunakan teknik ini unutk melakukan peringkasan dokumen misalnya Knight et al. [8], Zajic et al. [19], dan Hongyan Jing [7].

2.5.2 Teknik yang Tidak Bergantung pada Bahasa

Berbeda dengan teknik sebelumnya yang memanfaatkan tata bahasa untuk melakukan ekstraksi kalimat, terdapat juga teknik yang tidak menggunakan tata bahasa untuk menghitung kelayakan suatu kata untuk dibuang. Teknik ini biasanya menggunakan kumpulan dokumen *training* berupa kumpulan dokumen yang berisi kalimat yang belum terekstraksi dengan kalimat yang telah terekstraksi. Dari kumpulan kalimat tersebut kemudian diidentifikasi "kekuatan" suatu pasangan kata. Jika suatu pasangan kata tersebut kuat atau mempunyai ketergantungan yang tinggi dengan kata sebelum atau sesudahnya. Dengan hanya mengenali ciri dari kumpulan *corpus*, teknik ini dapat menghasilkan kalimat terekstraksi [15].

Teknik ini mempunyai kelebihan yaitu lebih *robust* terhadap noise baik secara sintaksis maupun semantik dan tidak tergantung pada bahasa. Selain itu teknik ini hanya memerlukan sekali *training* untuk mencari nilai dari parameter-parameter yang dibutuhkan dalam ekstraksi. Namun teknik ini juga mempunyai kelemahan yaitu dapat menyalahi aturan tata bahasa[15]. Untuk menghasilkan hasil yang akurasinya tinggi, harus digunakan *corpus* yang besar [12].

Salah satu contoh lain pengimplementasian teknik ini dalam peringkasan dokumen adalah dengan menggunakan algoritma *Hidden Markov Model* (HMM). Beberapa peneliti yang menggunakan HMM antara lain Zajic et al. [19] Nguyen et al.[12], dan Banko et al. [2].

Source model menggunakan pendekatan kalkulasi probabilitas bigram untuk menentukan tingkat kepentingan suatu kata. *Generative model* untuk menentukan suatu kata layak dihapus atau dipertahankan. Sebuah model HMM bernama *HMM Hedge* yang khusus untuk meng-*generate headline* berita digunakan di tahap ini. Tahap terakhir atau tahap viterbi *decoding* adalah tahap untuk menemukan barisan kata yang paling optimal. Tentunya setelah menghilangkan kata-kata yang tidak penting.

Nguyen et al. [12] menggunakan *Example Based Sentence Reduction* [12] HMM. EBSR HMM adalah suatu teknik yang memanfaatkan HMM dimana teknik ini berusaha untuk mencari kemiripan antara pasangan kalimat satu dengan pasangan kalimat lain. Sedangkan penelitian yang dilakukan oleh Banko et al. [2] mengenalkan sebuah rumus yang dapat digunakan sebagai acuan melakukan *training* HMM. Rumus tersebut adalah:

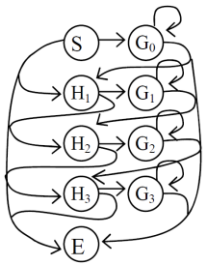
$$\begin{aligned}\arg\max_{H}(\quad & \alpha \cdot \sum_{i=1}^{n} log(P(w_i \in H \mid w_i \in D)) + \\ & \beta \cdot log(P(len(H) = n)) + \\ & \gamma \cdot \sum_{i=2}^{n} log(P(w_i \mid w_{i-1})))\end{aligned} \tag{1}$$

Dari beberapa studi literatur yang telah dilakukan, untuk mengerjakan penelitian ini digunakan teknik yang tidak bergantung kepada bahasa mengacu pada penelitian yang dilakukan oleh Banko et al. [2]. Karena kekurangan yang dimiliki oleh teknik ini yaitu kalimat yang dihasilkan dapat melanggar aturan sintaksis bahasa, tidak begitu berpengaruh. Hal ini karena tujuan akhir dari penelitian ni adalah melakukan pengindeksan. Dalam melakukan pengindeksan kita hanya memerlukan barisan kata yang akurat secara semantik untuk diberikan bobot dan diindeks bukan barisan kata yang benar secara sintaksis.

2.6 HMM-Hedge

Sebenarnya ada beberapa jenis cara untuk melakukan ekstraksi kalimat seperti telah dijelaskan sebelumnya. Namun untuk penelitian ini akan digunakan HMM-Hedge sebagai model *state*-nya. Hal ini karena HMM-Hedge lebih tahan terhadap kalimat yang memiliki noise tinggi[20]. Oleh karena itu penelitian yang ditujukan untuk mencari *headline* suatu berita biasanya menggunakan topologi jenis ini seperti yang telah dijelaskan di atas.

Model topologi HHM-Hedge[19] sendiri adalah sebagai berikut:



Gambar 2.5. HMM HEDGE [15]

Keterangan:
S : *start state*
H : *keepable words*
G : *erasable words*
E : end *state*

Contoh penerapan topologi tersebut untuk ekstrasi kalimat utama adalah sebagai berikut: misal diketahui kalimat yang akan diekstraksi adalah "*After months of debating following the Sept.11 terrorist hijacking the Transportation Department decided that airline pilots will not allowed to have guns in the cockpits".*

Langkah pembuangan *erasable words* adalah: mulai dari *state S*, mengeluarkan simbol *S*, lalu pindah ke *state* G_0. HMM akan tetap di *state* G_0 dan mengeluarkan kata "after", "month" … "airline". Kemudian pindah ke H_{pilot} dan mengeluarkan kata "pilot". Dilanjutkan G_{will} dan H_{not} dan seterusnya sampai $H_{cockpit}$ yang dilanjutkan ke *E*. Hasil dari emisi *H* berupa: "pilots" "not" "allowed" "to" "have" "guns" "in" "cockpits"[15]. Topologi HMM-Hedge digunakan sebagai acuan topologi untuk proses *decoding*.

2.7 Hybrid HMM

Hybrid HMM adalah proses menggabungkan fungsi translasi model pada dokumen corpus dengan POS Tagging-nya. Penghitungan dan formulasi model translasi pun sama (Gozali, 2010) hanya dataset diubah menjadi dataset yang berupa file sekuensial POS Tagging yang berkorespondensi dengan file corpus-nya.

POS Tagger yang dipakai untuk melakukan proses tagging adalah Stanford POS Tagger versi rilis tanggal 14 September 2011. Terdapat 35 jenis POS Tag yang dipakai.

Setelah dilakukan proses evaluasi dokumen corpus dan dokumen pos tag, maka rumus akhir yang digunakan dalam decoding menjadi seperti berikut:

$$C' = arg_H \max\left((1-\alpha)\sum_{i=1}^{n}\log\left(avg\left(P(S_i|C_i), P(ST_i|CT_i)\right)\right) + \alpha\sum_{i=1}^{n} avg\left(P(C_i|C_{i-1}), P(CT_i|CT_{i-1})\right)\right) \quad (2)$$

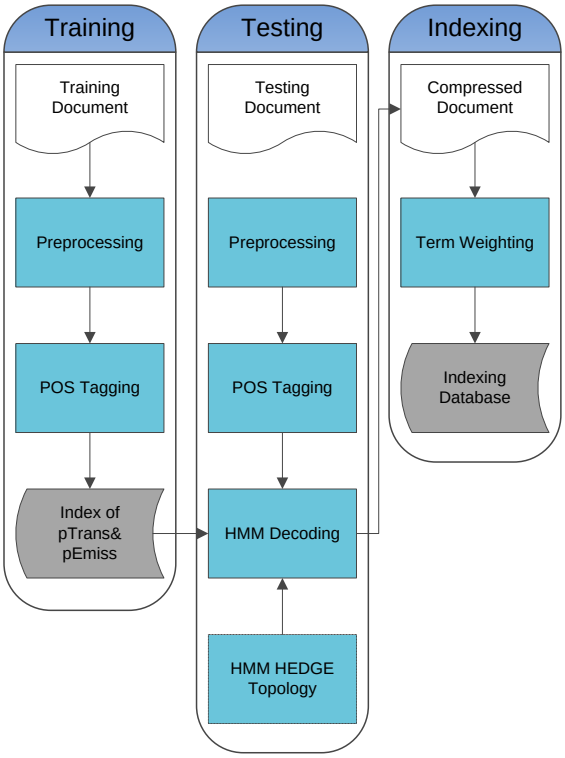
Persamaan akhir inilah yang akan menggabungkan dua jenis ekstraksi, yaitu ekstraksi dengan model bahasa dan ekstraksi dengan berbasis statistik.

3. DESAIN SISTEM

Pada bagian ini akan dijelaskan lebih rinci tahapan-tahapan yang akan dilakukan agar sistem IR yang baru dapat meminimalisir term yang terindeks dalam index collection. Untuk mencapai tujuan ini, maka dilakukan tahap peringkasan. Peringkasan yang akan diterapkan adalah sistem peringkas otomatis hibrid. Disebut sebagai peringkas Hybrid karena pada sistem peringkasan ini peneliti berusaha mengkombinasikan pendekatan dengan teknik yang tidak bergantung pada bahasa tertentu dan pendekatan dengan teknik yang bergantung pada tata bahasa tertenu.

Untuk teknik yang tidak bergantung pada bahasa, digunakan Hidden Markov Model, sedangkan untuk mengakomodasi tata bahasa maka sebelum dilakukan peringkasan tiap term ditentukan dahulu jenis katanya. Dengan melibatkan jenis kata dari tiap term, diharapkan makna kata yang tersaji dalam kalimat tidak hilang sehingga hasil ringkasan yang diperoleh menjadi lebih enak untuk dibaca.

Lebih lanjut dalam pemodelannya, hidden state yang dimaksud pada Hidden Markov Model yang digunakan adalah jenis kata tiap term atau POS Tag dari tiap term. Adapun usulan tahapan preprocessing ini dapat digambarkan pada gambar 3.1.



Gambar 3.1. Desain Sistem Hybrid HMM

Secara umum preprocessing ini terbagi menjadi 3 bagian besar yaitu :

1. Tahap Evaluation/ Training, yaitu tahap dimana dilakukan pelatihan atau pemodelan dengan menggunakan Model Translasi dan Model Bahasa seperti yang telah dipaparkan pada landasan teori Bab 2. Hasil akhir dari tahap evaluation atau training ini adalah parameter pTrans dan pEmission untuk setiap term.
2. Tahap Decoding/Testing, yaitu tahap peringkasan dokumen menggunakan parameter pTrans dan pEmiss yang telah dihasilkan dari tahap evaluation. Dimana pTrans dan pEmiss ini menjadi masukan pada HMM Decoding dengan menggunakan topologi HMM Hedge. Hasil akhir dari tahap decoding ini adalah ringkasan dokumen.
3. Tahap Indexing, dimana ringkasan dokumen kembali mengalami tokenisasi, stoplisting dan Term Weighting untuk kemudian dimasukkan ke dalam database, sehingga dihasilkan indexing beserta ID Dokumen, agar pada saat pencarian dapat dilakukan dengan lebih mudah.

3.1 Implementasi

Pada penelitian ini dibangun sebuah aplikasi yang mengimplementasikan proses ekstraksi kalimat utama suatu dokumen dengan algoritma *Hidden Markov Model*. Hasil dari proses tersebut akan digunakan sebagai input untuk proses *indexing* yang merupakan bagian hulu dari *Information Retrieval system*. Dalam penelitian ini juga dilakukan analisis terhadap kinerja sistem ekstraksi dokumen tersebut. Baik dari sisi sistem ekstraksi itu sendiri maupun dari implementasinya pada proses *indexing Information Retrieval*.

Implementasi Penelitian ini menggunakan perangkat lunak sebagai berikut:

1. Sistem operasi Windows 7 Ultimate
2. Notepad++ 5.7
3. XAMPP 1.6.2 for Windows dengan PHP 5.2.2 dan MySQL 5.0.41
4. Library untuk POS Tagger: Stanford-postagger-2011-09-14.jar

3.2 Implementasi Perangkat Keras

Implementasi Penelitian ini menggunakan perangkat keras sebagai berikut:

1. Prosesor Intel® Core™ 2 Duo CPU T6600 @2.20 GHz
2. RAM 1 GB (954 MB) dan Hardisk 160 GB

3.3 Data Set

Sistem yang dibangun pada penelitian ini akan menerima inputan berupa direktori tempat kumpulan dokumen halaman *web* berbahasa Inggris berada bentuk *file* yang berekstensi *.html. Spesifikasi dokumen berupa halaman *web* yang mempunyai bagian yang merupakan artikel utama. Salah satu cirinya adalah ditemukannya *tag* <p> dalam dokumen tersebut.

Data set yang digunakan sebagai inputan pada sistem penelitian ini adalah sekumpulan halaman *web* (*.htm/*.html) berita berbahasa Inggris yang diambil dari *website* media berita internasional yaitu www.tribalfootball.com, www.fifa.com, dan www.nytimes.com yang diambil pada bulan Agustus 2010. Dengan perincian sebagai berikut:

Table 1. Tabel Data Set

Situs	Jenis	Kategori	Jml
www.footballtribal.com	*Training* set	Sepak bola (liga)	200
www.footballtribal.com	*Testing* set	Sepak bola (world cup)	50
www.fifa.com	*Testing* set	Sepak bola (liga)	50
www.nytimes.com	*Testing* set	Berita dan politik dunia	50

Selain itu juga ada direktori tempat kumpulan hasil ringkasan kalimat yang dibuat manual oleh manusia (*corpus*). *File* hasil ringkasan berupa *file* *.txt yang akan digunakan untuk input proses *evaluation*/*training* dan pengukuran tingkat akurasi peringkasan dokumen. Jumlah dan spesifikasi *corpus* ini sama dengan *corpus* yang digunakan untuk *training* dan *testing*. Spesifikasi orang yang membuat *corpus* ini (*expert judgement*) memiliki TOEFL *score* lebih dari 450 (standar kelulusan mahasiswa ITTelkom). Sedangkan spesifikasi tambahan adalah sebagai berikut:

Table 2. Spesifikasi Expert Judgement

Pekerjaan	Pendidikan	Suka Sepakbola	Jumlah
Kepala sekolah	S2	Ya	1
Guru bahasa inggris	S1	Ya	1
Guru bahasa inggris	S1	Tidak	1
Mahasiswa	S1	Ya	3
Mahasiswa	S1	Tidak	1
Alumni	S1	Ya	3
Alumni	S1	Tidak	1

Keterangan: untuk profil lebih lengkap ada pada lampiran C.

Untuk mendukung peningkatan akurasi dan sebagai dokemen testing, pada penelitian ini juga ditambahkan dataset tambahan, yaitu dataset *medpost* yang didapatkan dari *alias-i* (www.alias-i.com). Corpus berupa kalimat yang diambil dalam jurnal kedokteran dan medis. Corpus ini terdiri dari 6.700 kalimat dengan 183.562 kata. Peringkas corpus ini diserahkan kepada lima orang mahasiswa dengan jenjang pendidikan D3 Teknik Informatika.

4. HASIL DAN ANALISA PENELITIAN

Ini adalah tahap dimana sistem akan diuji untuk kemudian didapatkan hasilnya dan dianalisa.

4.1 Variabel Pengujian

Variabel pengujian yang akan digunakan adalah sebagai berikut:

1. Extraction Accuracy with Rouge-2
2. Average Execution Time
4. Number of Indexed Term

4.2 Skenario Pengujian

Pengujian sistem menggunakan dataset yang telah dijelaskan sebelumnya akan diuji dan dianalisis hasilnya. Hal-hal yang akan dilakukan dalam pengujian sistem ini antara lain:

1. Analisis kecepatan ekstraksi dan akurasi dalam penggunaan POS *tag* dalam *preprocessing*
2. Analisis akurasi sistem berdasarkan parameter α dalam proses ekstraksi
3. Analisis pengaruh penerapan proses ekstraksi dalam *indexing* pada kecepatan dan jumlah *term* yang terindeks
4. Analisis pengaruh beberapa jenis *corpus* pada kecepatan dan akurasi ekstraksi

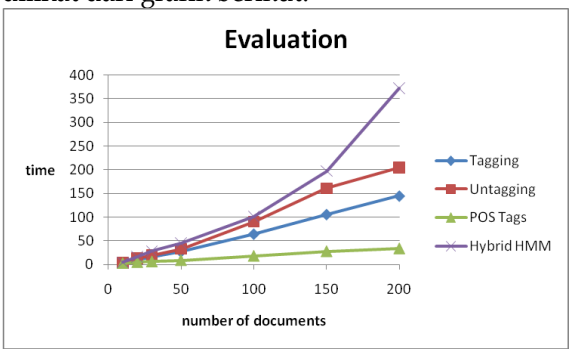
 ISSN: 1907-5022

4.3 Hasil dan Analisa Pengujian

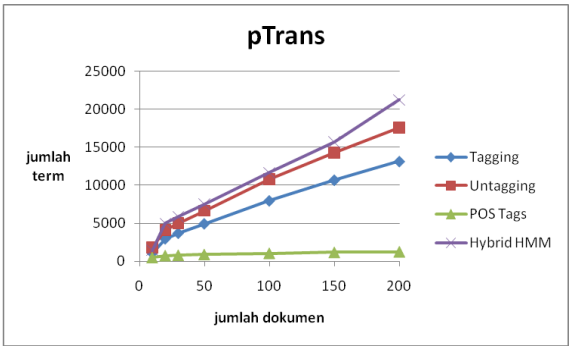
Pada bab ini akan dipaparkan hasil pengujian empat skenario di atas dan analisa terhadap hasil pengujian tersebut.

4.3.1 Skenario 1 – Pengujian POS *Tagging*

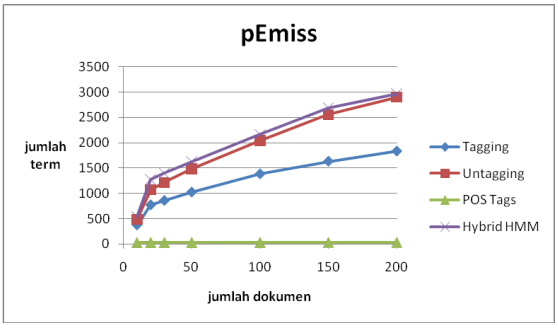
Berdasarkan data hasil pengujian proses pemberian *tag* pada entitas nama dan numerik maka dapat dilakukan perbandingan antara waktu, jumlah *term* yang terindeks, dan tingkat akurasi antara sebelum dengan sesudah proses *tagging* dilakukan. Perbandingan tersebut dapat dilihat dari grafik berikut:



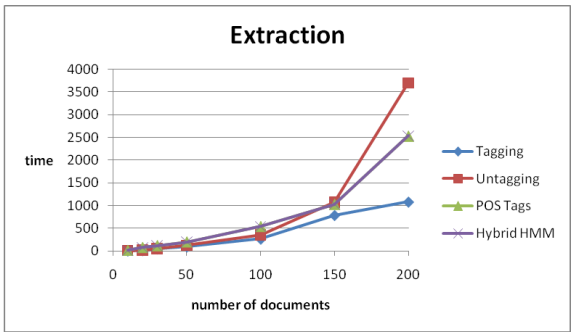
Gambar 4.1. Perbandingan waktu evaluasi sebelum dan sesudah tagging



Gambar 4-2 Perbandingan jumlah term pada tabel pTrans sebelum dan sesudah tagging

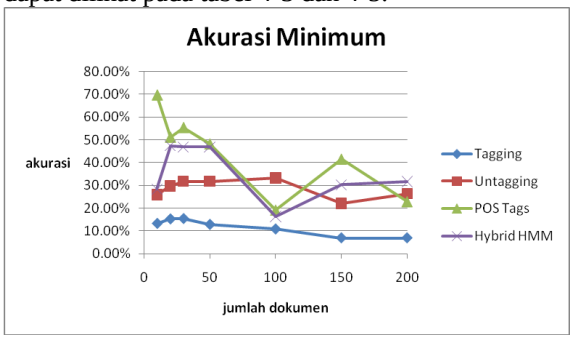


Gambar 4-3 Perbandingan jumlah term pada tabel pEmiss sebelum dan sesudah tagging

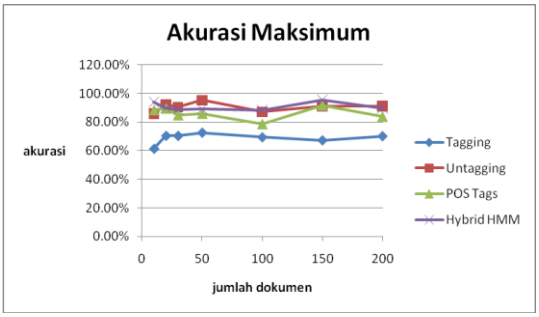


Gambar 4-4 Perbandingan waktu ekstraksi sebelum dan sesudah tagging

Dari grafik 4-1, 4-2, 4-3, dan 4-4 dapat dilihat bahwa preprosesing dokumen dengan menerapkan *tagging* pada entitas nama dan numerik dapat memangkas waktu proses *evaluation* dan *extraction*. Di samping itu dapat dilihat pula bahwa jumlah *term* pTrans dan pEmiss yang terindeks juga relatif lebih sedikit. Hal ini dapat terjadi karena jumlah *term* unik akan berkurang secara signifikan dan tergantikan oleh *tag* nama dan numerik seperti dapat dilihat pada tabel 4-3 dan 4-5.



Gambar 4-5 Perbandingan akurasi minimum sebelum dan sesudah tagging



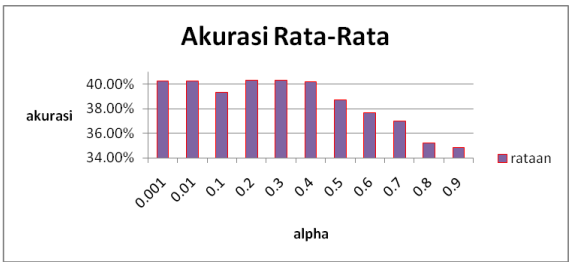
Gambar 4-6 Perbandingan akurasi maximum sebelum dan sesudah tagging

Dilihat grafik 4-5 dan 4-6 dapat ditarik kesimpulan bahwa proses *tagging* akan menurunkan tingkat akurasi baik untuk batas bawah maupun batas atas. Ini karena pada proses *tagging*, semua kata yang dianggap entitas nama dan numerik adalah sama walaupun dalam kenyataannya berbeda sebagaimana dijelaskan bab 3.2.2.

Berdasarkan hasil analisa, maka untuk pengujian selanjutnya akan digunakan *tagging* untuk preprosesingnya. Alasannya, sistem yang ditangani adalah sistem *indexing* yang mengutamakan kecepatan dan kehematan tempat(space). Selain itu, seiring dengan bertambahnya *corpus* maka tingkat akurasi bisa menjadi lebih tinggi[12].

4.3.2 Analisis Skenario 2 – Pengujian parameter *alpha*(α)

Berdasarkan pengujian terhadap perubahan parameter *alpha* yang merupakan parameter yang berpengaruh dalam penentuan jalur antar *state* pada proses *decoding* maka dapat dihasilkan perbandingan menurut tingkat akurasinya. Grafik perbandingan tersebut dapat disajikan sebagai berikut:



Gambar 4-7 Perbandingan nilai rata-rata variasi parameter alpha

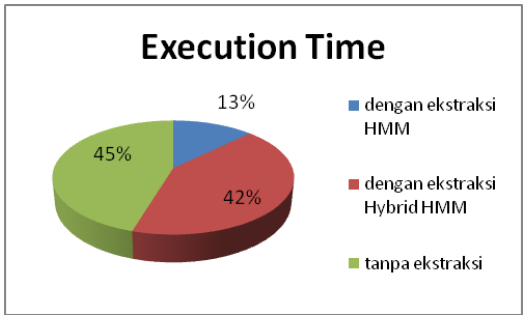
Jika dianalisis dari grafik 4-7 bisa diambil kesimpulan bahwa akurasi dari proses ekstraksi sangat dipengaruhi oleh parameter *alpha*. Hal ini terjadi karena parameter *alpha* adalah pengendali jalur *state* pada saat proses *decoding*. Makin besar nilai *alpha* maka akan semakin berpengaruh nilai dari probabilitas transisi dan emisi dibandingkan dengan probabilitas suatu *term* itu sendiri, dan sebaliknya.

Karena akurasi mencapai puncak tertingginya saat *alpha* sama dengan 0,2 dan 0,3 maka perlu ditentukan mana salah satu parameter yang akan diambil. Dan untuk pengujian selanjutnya digunakan 0,2.

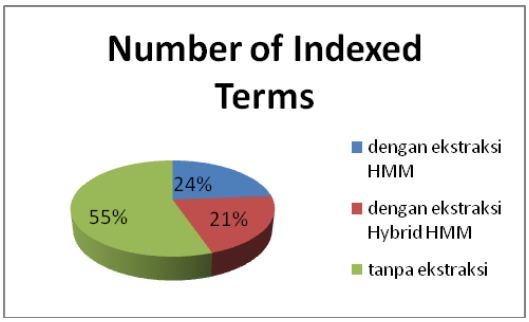
4.3.3 Analisis Skenario 3 – Pengujian pengaruh ekstraksi dokumen pada *indexing*

Berdasarkan hasil yang telah didapatkan pada pengujian skenario 3 maka dapat diambil kesimpulan bahwa dengan menerapkan proses ekstraksi dokumen sebelum dilakukan proses *indexing* akan membuat proses *indexing* menjadi lebih optimal. Hal ini dapat dilihat dari pengurangan waktu eksekusi dan jumlah *term* yang terindeks secara signifikan. Bahkan hingga memangkas 71.95% waktu eksekusi dan 56.98% jumlah *term* yang terindeks. Pengurangan *term* ini terjadi karena kata-kata yang dianggap kurang penting akan otomatis dihilangkan dalam rangka ekstraksi dokumen seperti dijelaskan bab 2.2.2.

Perbandingan performansi sistem dengan dan tanpa ekstraksi dapat dilihat dari grafik 4-10 dan 4-11 berikut:



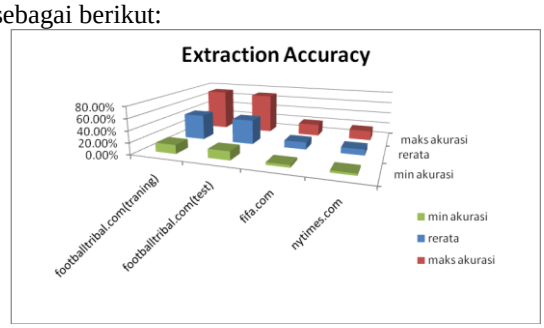
Gambar 4-8 Perbandingan execution time indexing dengan dan tanpa ekstraksi



Gambar 4-9 Perbandingan jumlah term pada tabel indexing dengan dan tanpa ekstraksi

4.3.4 Analisis Skenario 4 – Pengujian pengaruh jenis *corpus*

Berdasarkan hasil yang diperoleh saat pengujian skenario 3 maka dapat dilakukan analisis yang direpresentasikan pada grafik perbandingan sebagai berikut:



Gambar 4-10 Perbandingan nilai maks, min, dan average tiga jenis corpus

Dari grafik di atas dapat dianalisis ternyata jenis *corpus* cukup berpengaruh terhadap kinerja dari sistem ekstraksi. Semakin jauh karakteristik jenis *corpus* dengan karakteristik *training* data set maka akan semakin jauh pula akurasinya dari batas yang sapat ditoleransi. *Corpus* yang diambil dari footballtribal.com memiliki akurasi yang tinggi karena dari situs itu juga *training* set diambil. *Corpus* dari *fifa.com* memiliki tingkat akurasi

yang lebih tinggi daripada *NYTimes.com* karena karakteristik situs *fifa.com* mendekati *footballtribal.com,* yaitu membahas masalah sepak bola. Berbeda dengan NYTimes.com yang membahas masalah berita dan politik.

5. KESIMPULAN DAN SARAN

Dari hasil pengujian dan analisis yang telah dilakukan pada bab sebelumnya dalam penelitian ini, maka didapatkan kesimpulan:

1. Penggunaan Hybrid HMM dalam melakukan preprocessing dapat menurunkan waktu eksekusi dan jumlah term yang perindeks namun jika tidak diimbangi dengan jumlah training dataset yang besar dan akurat, akan menurunkan tingkat akurasi sistem ke level yang tidak dapat ditoleransi.
2. Nilai parameter alpha untuk parameter decoding berpengaruh dalam menentukan akurasi dari hasil ekstraksi dan nilai alpha yang paling optimum untuk jenis corpus yang dibahas dalam penelitian ini adalah 0,2 dan 0,3.
3. Penggunaan sistem ekstraksi dokumen dengan Hybrid HMM dapat mereduksi waktu eksekusi dan jumlah term dengan cukup signifikan.
4. Spesifikasi corpus yang digunakan sebagai training dataset sangat berpengaruh terhadap tingkat akurasi testing dataset yang menjadi input sistem. Semakin jauh karakteristik testing dataset dari training dataset, akurasinya akan semakin rendah. Sebaliknya, semakin dekat karakteristik testing dataset dengan training dataset, akurasinya akan semakin tinggi.

Saran untuk penelitian selanjutnya adalah sebagai berikut:

1. Perlu dikaji lagi penggunaan fungsi rata-rata dalam model fungsi Hybrid HMM. Fungsi rata-rata dapat diganti menjadi fungsi yang lebih generik dengan menggunakan variabel untuk menentukan tingkat kepentingan antara model bahasa dengan model statistik
2. Dibuat post-processing untuk menguji kinerja sistem indexing ini agar dapat diketahui tingkat kepuasan pengguna terhadap sistem yang telah dibuat
3. Perlu dilakukan riset lebih lanjut untuk identifikasi sekuens *term* yang layak dikenali sebagai kalimat

PUSTAKA

[1] B. H. Juang; L. R. Rabiner. *Hidden Markov Models for Speech Recognition*. Technometrics, Vol. 33, No. 3. (Aug. 1991):251-272

[2] Banko, M. Mittal, V. O. Witbrock, M. J., "*Headline Generation Based on Testing Translation*", Annual Meeting- Association For Computational Linguistics,Vol 38; Part 1, 2000: 318 – 325

[3] Broder, Andrei. *A Taxonomy of Web Search*. IBM Research, 2002

[4] Doran, W., Stokes, N., Newman, E., Dunnion, J., Carthy,J., Toolan, F., "*News Story Gising at University College Dublin*", Document Understanding Conference, DUC 2004.

[5] *Hidden Markov Model* – Wikipedia, the free encyclopedia, http://en.wikipedia.org/wiki/*Hidden* Markov *Model* diunduh 20 Oktober 2009

[6] J. Beal, Matthew; Ghahramani, Zoubin; Edward Rasmussen, Carl. *The Infinite Hidden Markov Model*. Gatsby Computational Neuroscience Unit University College London, 2008.

[7] Jing H, *Sentence Reduction For Automatic Text Summarization*-Proceedings of the 6th Applied Natural Language Processing, 2000.

[8] Knight, K., and Marcu, D, *Statistics Based Summarization: Step One: Sentence Compression*. Proceedings of the 17th National Conference of theAmerican Association for Artificial Intelligence AAAI2000, Austin, Texas, July 30-August 3, 2000.

[9] L. Rabiner. *A tutorial on Hidden Markov Models and selected applications in speech recognition*. Proc. of IEEE, 77(2):257-286, 1989.

[10] C.Y. Lin, *Recall-Oriented Understudy for Gisting Evaluation*, 2003.

[11] Marlow, Kit. 2003. *Information Retrieval Methods*, http://www.seas.upenn.edu/~zives/03s/cis650/ir.pdf diunduh 20 Oktober 2009.

[12] M Le Nguyen, S Horiguchi, A Shimazu, BT Ho, *Example-Based sentence Reduction* using the *Hidden Markov Model*, ACM Transactions on Asian Language Information *Process*sing, 2004.

[13] SF Chen, J Goodman, *An Empirical Study of Smoothing Techniques for Language Modeling*, Proceedings of the 34th annual meeting on Association for Computational Linguistik, 1996

[14] Waiyamai, Kitsana. *Introduction to Text Mining*. Dept of Computer Engineering, Faculty of Engineering, Kasetsart University, Bangkok, Thailand.

[15] Wibisono, Yudi, *Penggunaan Hidden Markov Model untuk Kompresi Kalimat*, Tesis Institut Teknologi Bandung, 2008.

[16] Witbrock, M.J., Mittal, V.O, "*Ultra-Summarization: A Testing Approach to Generating Highly Condensed Non-Extractive Summaries (poster abstract)*",1999.

[17] *Research and Development in Information Retrieval*, pages 315-316

[18] Wright, Jan 1998. *An Overview Of Indexing Methods* http://www.stcsig.org/idx/articles/methods.pdf diunduh 20 Oktober 2009.
[19] Zajik D, Dorr B, *Automatic Headline Generation for Newspaper Stories*, 2002.
[20] Zajic,D.etal., *Multi-candidate Reduction*: *Sentence Compression as a tool*, *Information Process*sing *and Management*, 2007
[21] Gozali, Alfian Akbar, dkk. *Analysis of Hidden Markov Model Method Implementation in Documents Topic Sentence Extraction for Information Retrieval*. International Conference on Telecommunication, ISSN: 1858-2982, October, 2010
[22] Mandala, Rila. *Bahan Kuliah Sistem Temu Balik Informasi: Pengantar Sistem Temu Balik Informasi*. Institut Teknologi Bandung. Departement Informatika. 2004