JOINT

Journal of Informatics and Technology, Vol 1, No 3, Tahun 2012, p 31-51
http://ejournal-s1.undip.ac.id/index.php/joint

PENYARINGAN FRASA KUNCI SECARA OTOMATIS MENGGUNAKAN ALGORITMA KEA++ UNTUK PENCARIAN ARTIKEL ILMIAH BERBAHASA INDONESIA

Kuncara Adi Nugraha
Program Studi Teknik Informatika Jurusan Matematika FSM UNDIP Semarang
Email : kuncara.adinugraha@gmail.com

Nurdin Bahtiar, S.Si, M.T.
Dosen Program Studi Teknik Informatika Jurusan Matematika FSM UNDIP Semarang

Beta Noranita, S.Si, M.Kom
Dosen Program Studi Teknik Informatika Jurusan Matematika FSM UNDIP Semarang

ABSTRAK

Pencarian artikel ilmiah akan lebih mudah dengan adanya ikhtisar dari artikel ilmiah. Frasa kunci menyediakan ikhtisar ringkas mengenai artikel ilmiah. Meskipun frasa kunci sangat berguna, sebagian artikel ilmiah tidak memiliki frasa kunci. Penyaringan frasa kunci secara manual menghabiskan waktu dan cukup mahal. Penyaringan frasa kunci secara otomatis dapat menghemat waktu yang diperlukan untuk menentukan frasa kunci artikel ilmiah. Penelitian ini mengembangkan sistem pencarian artikel ilmiah berbahasa Indonesia yang memanfaatkan penyaringan frasa kunci artikel ilmiah berbahasa Indonesia menggunakan algoritma KEA++. Data yang digunakan dalam penelitian ini adalah 30 jurnal akuntansi berbahasa Indonesia dan Tesaurus akuntansi berbahasa Indonesia. Berdasarkan hasil pengujian, semakin banyak data pelatihan, akurasi penyaringan frasa kunci semakin baik. Hasil pengujian untuk penggunaan Tesaurus, nilai rata-rata *precision* dan *recall* yang diperoleh adalah 23.8% dan 37.4%. Berdasarkan penilaian dari responden untuk penilaian kinerja penyaringan frasa kunci, sistem memperoleh total skor 31 dari total skor memungkinkan 45 dan persentase frasa kunci yang sesuai adalah 51.75%.

Kata Kunci : penyaringan frasa kunci otomatis, frasa kunci, algoritma KEA++, artikel ilmiah berbahasa Indonesia, Tesaurus

ABSTRACT

Searching of scientific papers will be easier with a summary of scientific papers. Keyphrases provide a short summary of the scientific paper. Although keyphrases are very useful, several scientific papers don't have keyphrases. Manual keyphrase extraction is time-consuming and costly. Automatic keyphrase extraction can save the time needed to assign keyphrases. This research developed a searching system of Indonesian-language scientific papers that utilising the keyphrase extraction of Indonesia-language scientific papers using KEA++ algorithm. The data used in this research are 30 Indonesian-language journals of accounting and Indonesian-language thesaurus of accounting. Based

on the test results, the more training data, accuration of keyphrase extraction is increased. The test results for thesaurus usage, the average values of precision and recall are 23.8% and 37.4%. Based on assessment from respondents for system's keyphrase extraction, system reached 31 point from maximal point 45 and percentage of correct keyphrases is 51.75%.

Keywords : automatic keyphrase extraction, keyphrase, KEA++ algorithm, Indonesian-language scientific papers, thesaurus

1. Pendahuluan

Artikel ilmiah sering digunakan sebagai referensi untuk penyelesaian masalah atau penulisan artikel ilmiah yang lain. Dengan semakin melimpahnya jumlah artikel ilmiah yang tersedia di perpustakaan dan internet, setiap pembaca yang ingin mencari referensi akan lebih dipermudah jika setiap artikel ilmiah yang tersedia telah memiliki ikhtisar atau ringkasan mengenai isi artikel ilmiah tersebut. Salah satu bentuk ikhtisar tersebut adalah frasa kunci. Frasa kunci merepresentasikan ikhtisar ringkas dan tepat dari dokumen [12]. Tetapi, tidak semua artikel ilmiah memiliki frasa kunci.

Frasa kunci dalam sebuah artikel ilmiah, pada umumnya dipilih secara manual oleh penulis artikel ilmiah tersebut. Pengindeks profesional juga dapat menentukan frasa kunci dalam sebuah artikel ilmiah, tetapi proses ini sering terkendala oleh waktu. Oleh karena itu, diperlukan adanya penyaringan frasa kunci secara otomatis.

Penyaringan frasa kunci otomatis adalah identifikasi frasa yang paling penting dalam sebuah dokumen oleh komputer daripada manusia [10]. Dengan adanya proses penyaringan frasa kunci secara otomatis, waktu yang diperlukan untuk mencari frasa kunci dalam sebuah artikel ilmiah menjadi lebih ringkas. Dengan tersedianya frasa kunci, pembaca akan dipermudah dengan dapat memilah–milah artikel ilmiah sesuai kebutuhannya dengan melihat terlebih dahulu frasa kunci dari artikel tersebut.

Penyaringan frasa kunci dalam teks yang tersimpan dalam basis data atau dokumen teks termasuk dalam bidang *Text Mining.* Salah satu algoritma yang telah dikembangkan adalah algoritma *Keyphrase Extraction Algorithm for Controlled Indexing* (KEA++). Dalam penelitian tugas akhir ini, dikembangkan sistem pencarian artikel ilmiah berbahasa Indonesia yang memanfaatkan hasil dari

penyaringan frasa kunci secara otomatis dengan menggunakan algoritma KEA++ yang disesuaikan untuk artikel ilmiah berbahasa Indonesia.

2. Landasan Pustaka

2.1 Artikel Ilmiah

Artikel ilmiah adalah laporan yang ditulis dan dipublikasikan mendeskripsikan tentang hasil penelitian yang orisinal. Artikel ilmiah, meskipun memenuhi semua kategori penulisan yang baik, tidak dipublikasikan secara valid jika dipublikasikan di tempat yang salah. [3]

Artikel ilmiah termasuk jurnal harus memiliki bab-bab tertentu untuk dapat dikategorikan sebagai artikel ilmiah. Setiap artikel ilmiah harus memiliki, secara urut, Abstrak, Pendahuluan, Materi dan Metodologi, Hasil Penelitian, Pembahasan, dan Daftar Pustaka [3].

Salah satu bentuk dari ikhtisar dari artikel ilmiah adalah kata kunci dan frasa kunci. Frasa kunci dapat diartikan sebagai rangkaian satu kata atau lebih yang dianggap sangat sesuai dengan dokumen, sementara kata kunci adalah satu kata yang sesuai dengan dokumen. Kombinasi yang berubah-ubah dari sebuah kata kunci belum tentu menggantikan sebuah frasa kunci dan komponen dari sebuah frasa kunci belum tentu mewakili kata kunci [6].

2.2 *Text Mining*

Text Mining merupakan salah satu bidang dalam *Data Mining* yang mempelajari tentang *mining* informasi yang tersimpan dalam teks dokumen atau basis data yang bertipe teks.

Pengguna informasi memerlukan *tools* yang dapat digunakan untuk membandingkan dokumen yang berbeda, mengatur skala kepentingan dan keterkaitan dokumen, atau menemukan pola dan kecenderungan terhadap berbagai dokumen. Oleh karena itu, *Text Mining* menjadi populer dan tema penting dalam *Data Mining.* [9]

2.2.1 Penyaringan Frasa Kunci Otomatis

Penyaringan frasa kunci otomatis (*automatic keyphrase extraction*) adalah identifikasi frasa yang paling penting dalam sebuah dokumen oleh komputer

daripada manusia [10]. Penyaringan frasa kunci otomatis merupakan pekerjaan klasifikasi terhadap frasa-frasa yang ada dalam dokumen yang ada di dalamnya. Algoritma penyaringan frasa kunci otomatis dapat mengklasifikasikan sebuah frasa merupakan frasa kunci atau bukan.

2.2.2 *Keyphrase Extraction Algorithm for Controlled Indexing* (KEA++)

Algoritma KEA++ merupakan algoritma yang dikembangkan oleh Olena Medelyan. Algoritma ini disebut KEA++ karena memperbaiki algoritma sebelumnya yaitu algoritma KEA, dan bekerja berdasarkan pada pembelajaran mesin serta bekerja pada dua tahap, yakni Identifikasi Kandidat yang mengidentifikasi istilah dalam Tesaurus yang terkait dengan isi dokumen dan *filtering* yang menggunakan model pembelajaran untuk mengidentifikasi istilah yang paling signifikan berdasarkan *features* [12].

2.2.2.1 Identifikasi Kandidat

Proses Identifikasi Kandidat adalah proses identifikasi kandidat frasa kunci yang terkait dengan isi dokumen dan memasukkannya ke daftar kandidat. Penjelasan langkah-langkah dalam proses Identifikasi Kandidat adalah sebagai berikut:

1) *Tokenization*

   Proses *tokenization* adalah proses memecah teks menjadi *token* berdasarkan spasi dan tanda baca yang ada dalam teks. Dalam proses ini, tanda baca dan pembatas paragraf dihilangkan dari teks.

2) Membuat n-gram dari token-token yang telah dihasilkan

   Token yang diperoleh dari proses *tokenization* yang letaknya berurutan disusun menjadi *n-gram*. *N-gram* merupakan rangkaian dari *n* huruf atau *n* kata. Setiap *n-gram* yang dihasilkan dihitung jumlah kemunculannya dalam dokumen tersebut kecuali *n-gram* yang diawali atau diakhiri dengan *stopwords. Stopwords* merupakan kata-kata yang tidak memiliki relevansi dalam pencarian tetapi sering muncul dalam dokumen.

3) Membuat *pseudo-phrase* dari *n-gram* yang telah dihasilkan dan Tesaurus

   Pseudo-phrase diperoleh dengan cara menghapus *stopwords*, *stemming*, dan mengurutkan sesuai abjad. Proses menghapus *stopwords* adalah sebuah proses

untuk menghilangkan kata yang 'tidak relevan' pada hasil *tokenization* sebuah dokumen teks dengan cara membandingkannya dengan *Stoplist* yang ada [2].

Proses *stemming* merupakan proses untuk mengembalikan kata-kata yang berimbuhan ke kata dasarnya. Algoritma yang digunakan untuk penelitian ini adalah algoritma *Porter Stemmer for* Bahasa Indonesia yang dikembangkan oleh Fadillah Z. Tala. Proses ini dilaksanakan untuk memperoleh tingkat kecocokan kata serta *conflation* yang lebih tinggi.

4) Membandingkan setiap *pseudo-phrase* yang diperoleh dengan *pseudo-phrase* dalam Tesaurus

Proses yang dilakukan selanjutnya setelah diperoleh *pseudo-phrase* adalah membandingkan setiap *pseudo-phrase* dengan istilah yang tersimpan dalam Tesaurus yang telah diubah ke bentuk *pseudo-phrase*. Proses ini dilakukan untuk membuang *n-gram* yang tidak memiliki arti signifikan. Setiap istilah mendapat jumlah kemunculan yang merupakan penjumlahan dari jumlah kemunculan setiap *n-gram* yang dipetakan ke *descriptor* tersebut [11]. Kandidat yang diperoleh berasal dari istilah dalam Tesaurus yang muncul dalam teks beserta jumlah kemunculannya.

5) *Semantic conflation*

Setiap istilah *non-descriptor* dalam Tesaurus yang teridentifikasi dalam teks digantikan dengan *descriptor*-nya [11]. Hal ini dilakukan untuk mencegah istilah yang memiliki definisi yang sama muncul sebagai kandidat. Sebagai contoh, frasa Sistem Informasi Manajemen dan SIM. Jumlah kemunculan dari *descriptor* yang terkait dijumlahkan dengan jumlah kemunculan dari semua *non-descriptor* yang terasosiasi. Proses ini menghasilkan daftar istilah yang menjadi kandidat yang terkait dengan isi dokumen dan jumlah kemunculannya.

6) Menentukan nilai *feature* dari masing-masing kandidat

Ada empat *feature* yang digunakan dalam algoritma KEA++, yaitu:

a) TF x IDF

Skor *Terms Frequency* (TF) x *Inverse Document Frequency* (IDF) membandingkan frekuensi dari kemunculan frasa dalam sebuah dokumen dengan frekuensi kemunculan kata secara umum. Frasa kandidat yang memiliki nilai TFxIDF tinggi lebih berpeluang menjadi frasa kunci [4].

TF x IDF untuk frasa *P* dalam dokumen *D* dihitung dengan persamaan 2.1 [8].

$$TF \times IDF = \frac{freq(P,D)}{size(D)} \times -\log_2 \frac{df(P)}{N} \quad \text{.....(2.1) [11]}$$

Keterangan :

freq (P,D) = jumlah kemunculan frasa *P* dalam dokumen *D*
size(D) = jumlah frasa dalam dokumen *D*
df(P) = jumlah dokumen yang memiliki frasa *P* di *global corpus*
N = ukuran dari *global corpus.*

b) Jarak kemunculan kata dari awal dokumen

Feature ini dihitung dari jumlah kata yang ada sebelum kemunculan sebuah frasa dibagi dengan jumlah kata sebuah dokumen. Kandidat yang memiliki nilai sangat tinggi atau sangat rendah cenderung merupakan frasa yang benar, karena muncul di bagian pembukaan dokumen seperti judul dan abstrak atau di bagian akhir seperti kesimpulan dan daftar pustaka [12].

c) Panjang frasa dalam kata

Penggunaan panjang frasa dalam kata sebagai *feature* meningkatkan kemungkinan kandidat yang terdiri dari dua kata sebagai frasa kunci [11].

d) *Node Degree*

Feature ini menunjukkan jumlah dari *links* dalam Tesaurus yang menghubungkan sebuah frasa terhadap frasa kandidat lain. Sebuah frasa yang memiliki nilai *node degree* besar cenderung lebih berarti [12].

2.2.2.2 Filtering fase Pembuatan Model

Dalam proses pembuatan model, diperlukan dokumen pelatihan yang telah dilengkapi dengan frasa kunci. Untuk setiap dokumen pelatihan, proses Identifikasi Kandidat dilakukan untuk memperoleh kandidat beserta *features*-nya.

Penjelasan langkah-langkah proses pembuatan model adalah sebagai berikut :

1) Menandai setiap kandidat dengan label *index term*

Setelah *feature* dari masing-masing kandidat dihitung, setiap kandidat ditandai apakah merupakan *index term* atau tidak dengan menggunakan frasa kunci yang telah disediakan oleh penulis artikel ilmiah. Kelas *index term* merupakan kelas biner yang memiliki nilai *true* atau *false* yang menandakan apakah kandidat merupakan frasa kunci atau tidak.

2) Diskretisasi data

Proses diskretisasi dilaksanakan untuk mengubah nilai dari masing-masing *feature* dari bentuk *real* ke bentuk *nominal* yang digunakan dalam fase pelatihan. KEA++ menggunakan metode diskretisasi *supervised* yaitu *entropy-based* dengan menggunakan *Minimum Description Length* (MDL) *stopping criterion*. Dari proses diskretisasi, dihasilkan sebuah tabel diskretisasi. Tabel diskretisasi digunakan untuk menyederhanakan data yang berbentuk *numeric* menjadi *nominal* dengan cara membagi kisaran nilai atribut berdasarkan interval tertentu. Dengan cara mengubah nilai-nilai dari atribut *continuous* dengan sejumlah *interval label*, dapat mengurangi dan menyederhanakan data [9].

3) *Training* dengan menggunakan klasifikasi Naïve Bayes

Proses fase pelatihan menggunakan klasifikasi *Naive Bayes* dan menggunakan kelas *index term*. Dari proses ini dihasilkan probabilitas nilai *features* menjadi frasa kunci dan probabilitas prior frasa kunci

Dari fase diskretisasi dan pelatihan diperoleh model yang digunakan selanjutnya dalam proses pengujian. Model yang dihasilkan berupa tabel diskretisasi, probabilitas nilai *features* menjadi frasa kunci, dan probabilitas prior frasa kunci.

2.2.2.3 Filtering fase Identifikasi Kandidat

Untuk menentukan frasa kunci dari sebuah dokumen, langkah yang digunakan adalah Identifikasi Kandidat dan kemudian dihitung probabilitasnya dengan menggunakan model yang telah dihasilkan dari fase pelatihan. Jika dimisalkan hanya dua nilai *features* yang digunakan yaitu, TFIDF(*t*) dan jarak kemunculan dari awal dokumen (f), nilai dari *P[yes]* atau *P[no]* diperoleh dari perhitungan persamaan 2.2.

$$P[yes] = \frac{Y}{Y+N} P_{TFIDF}[t|yes] P_{1stOccur}[f \quad \text{.....(2.2) [11]}$$

Keterangan :

= Probabilitas prior untuk frasa kandidat memiliki *index term* dengan nilai *true*

$P_{TFIDF}[t$ = probabilitas kelas dari *TFxIDF t* yang bernilai *yes.*

$P_{1stOccur}[f$ = probabilitas kelas dari *First Occurence f* yang bernilai *yes*

Probabilitas dari sebuah frasa adalah frasa kunci diperoleh dengan menghitung dengan menggunakan persamaan 2.3.

$$p = \frac{P[yes]}{(P[yes] + P[no])} \quad \text{..... (2.3) [11]}$$

Frasa kandidat diranking berdasarkan nilai probabilitas *p* dan nilai TFxIDF sebelum proses diskretisasi digunakan sebagai *tiebreaker* jika dua frasa memiliki probabilitas yang sama (biasanya karena diskretisasi). Dari daftar inilah diambil beberapa frasa yang memiliki ranking tertinggi sebagai frasa kunci.

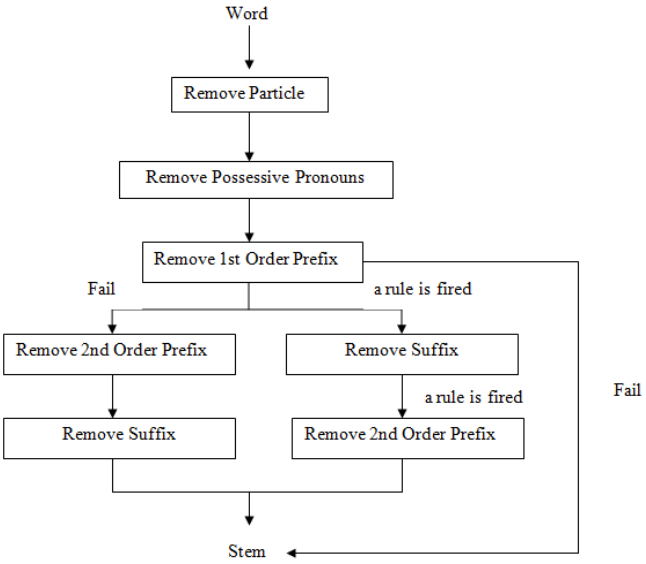
2.2.3 Algoritma *Porter Stemmer for* Bahasa Indonesia

Algoritma *Porter Stemmer for* Bahasa Indonesia merupakan algoritma untuk proses *stemming* yang dikembangkan oleh Fadillah Z. Tala berdasarkan Algoritma *English Porter Stemmer*. Pengembangan yang dilakukan terdiri atas perubahan aturan dan kondisi ukuran kata. Desain dari Algoritma *Porter Stemmer for* Bahasa Indonesia dapat dilihat pada Gambar 2.1.

Ada lima aturan mengenai imbuhan yang terdapat pada algoritma *Porter Stemmer for* Bahasa Indonesia**.** Aturan ini diperoleh dari proses perubahan kata yang terdapat pada bahasa Indonesia.

Kendala yang dihadapi adalah apabila terjadi ambiguitas kata hasil dari proses *stemming*. Tidak semua kata yang mengalami proses *stemming* sesuai dengan kata dasarnya. Misal, kata mengubah bisa saja berasal dari kata dasar kubah atau ubah.

Secara umum, terdapat satu huruf vokal dalam setiap suku kata. Jumlah minimum untuk setiap kata dari hasil *stemming* adalah dua suku kata. Ukuran kata yang didesain di sini tidak dapat menangkap semua ukuran aktual dari kata-kata di bahasa Indonesia [14]. Kendala yang dihadapi adalah jika terdapat diftong atau dua huruf vokal berurutan yang dianggap sebagai tidak terpisah, yaitu *au*, *ai* dan *oi*. Untuk diftong *ai* dan *oi*, jika terdapat di akhir kata, maka akan mengalami proses *stemming* karena dianggap sebagai kata berimbuhan akhiran –i.



Gambar 2.1. Desain Algoritma *Porter Stemmer for* Bahasa Indonesia [14]

2.2.4 Klasifikasi Naïve Bayes

Klasifikasi Naive Bayes dalam algoritma KEA++ digunakan untuk membuat model yang digunakan dalam proses pengujian. KEA++ menggunakan Klasifikasi Naive Bayes karena sederhana dan menghasilkan model yang bagus [12].

Klasifikasi Naive Bayes dikembangkan berdasarkan teorema Bayes. Secara teori, Klasifikasi Naive Bayes memiliki tingkat kesalahan paling rendah dibandingkan dengan *classifier* yang lain [9]. Keuntungan dari Klasifikasi Naive Bayes adalah hanya memerlukan data pelatihan yang kecil untuk mengestimasi parameter.

2.2.5 Metode Validasi *k-fold Cross Validation*

Metode *k-fold Cross Validation* merupakan salah satu metode yang digunakan untuk melakukan pengujian akurasi dari sebuah model klasifikasi. Data yang digunakan dibagi menjadi *k* bagian, D_1, D_2, D_3,…D_k, dengan jumlah data yang sama untuk masing-masing bagian. Pelatihan dan pengujian dilaksanakan sebanyak *k* kali [9].

2.2.6 *Precision, Recall,* and *F-score*

Precision dan *Recall* merupakan salah satu metode yang digunakan untuk mengetahui akurasi sistem dalam pemrosesan *text mining* dan *text retrieval*.

Precision merupakan persentase jumlah frasa kunci yang relevan terhadap keseluruhan frasa kunci yang telah disaring. *Recall* merupakan persentase jumlah frasa kunci yang relevan terhadap jumlah frasa kunci yang dipilih secara manual oleh penulis artikel ilmiah.

Nilai dari *precision* dan *recall* berada di antara 0 dan 1. Nilai 0 menunjukkan bahwa sistem gagal dalam proses *text mining* dan nilai 1 menunjukkan bahwa sistem optimal dalam proses *text mining*. Semakin banyak frasa kunci yang digunakan, maka nilai *precision* akan menurun tetapi nilai *recall* akan meningkat dan sebaliknya. Untuk menghitung rata-rata harmonis antara *precision* dan *recall*, digunakan *F-score*. Rata-rata harmonis mencegah sistem untuk mengorbankan salah satu nilai secara drastis [9].

2.3 Tesaurus

ISO 2788:1986 mendefinisikan Tesaurus sebagai perbendaharaan kata dari *controlled indexing language,* terorganisasi secara formal bertujuan untuk menunjukkan secara eksplisit hubungan yang ada di antara konsep [13]. Dari perspektif fungsional, Tesaurus digunakan untuk mengatasi ambiguitas dari kata–kata terkait dengan proses perolehan informasi. Tesaurus dibuat berdasarkan instrumen kebahasaan dan berkembang menjadi sistem yang berfokus pada organisasi informasi dan representasi dari isi sebuah *corpus* dokumen, biasanya digunakan untuk *term extraction* [13].

Salah satu bentuk format yang digunakan untuk merepresentasikan tesaurus adalah SKOS. *Simple Knowledge Organization System* (SKOS) merupakan RDF *vocabulary* yang digunakan untuk merepresentasikan sistem organisasi pengetahuan yang bersifat semi-formal, seperti tesaurus, taksonomi, skema klasifikasi dan daftar yang berisi tentang subjek tertentu [8].

3. Analisis dan Perancangan

3.1 Data yang Digunakan

Data yang digunakan dalam sistem ini ada 3 jenis, yaitu :

1) *Stoplist* Bahasa Indonesia

Stoplist merupakan daftar yang berisi sekumpulan kata yang tidak relevan, tetapi sering sekali muncul dalam sebuah dokumen. *Stoplist* yang digunakan dalam penelitian ini adalah *stoplist* yang disusun oleh Fadillah Z. Tala.

2) Artikel Ilmiah berbahasa Indonesia

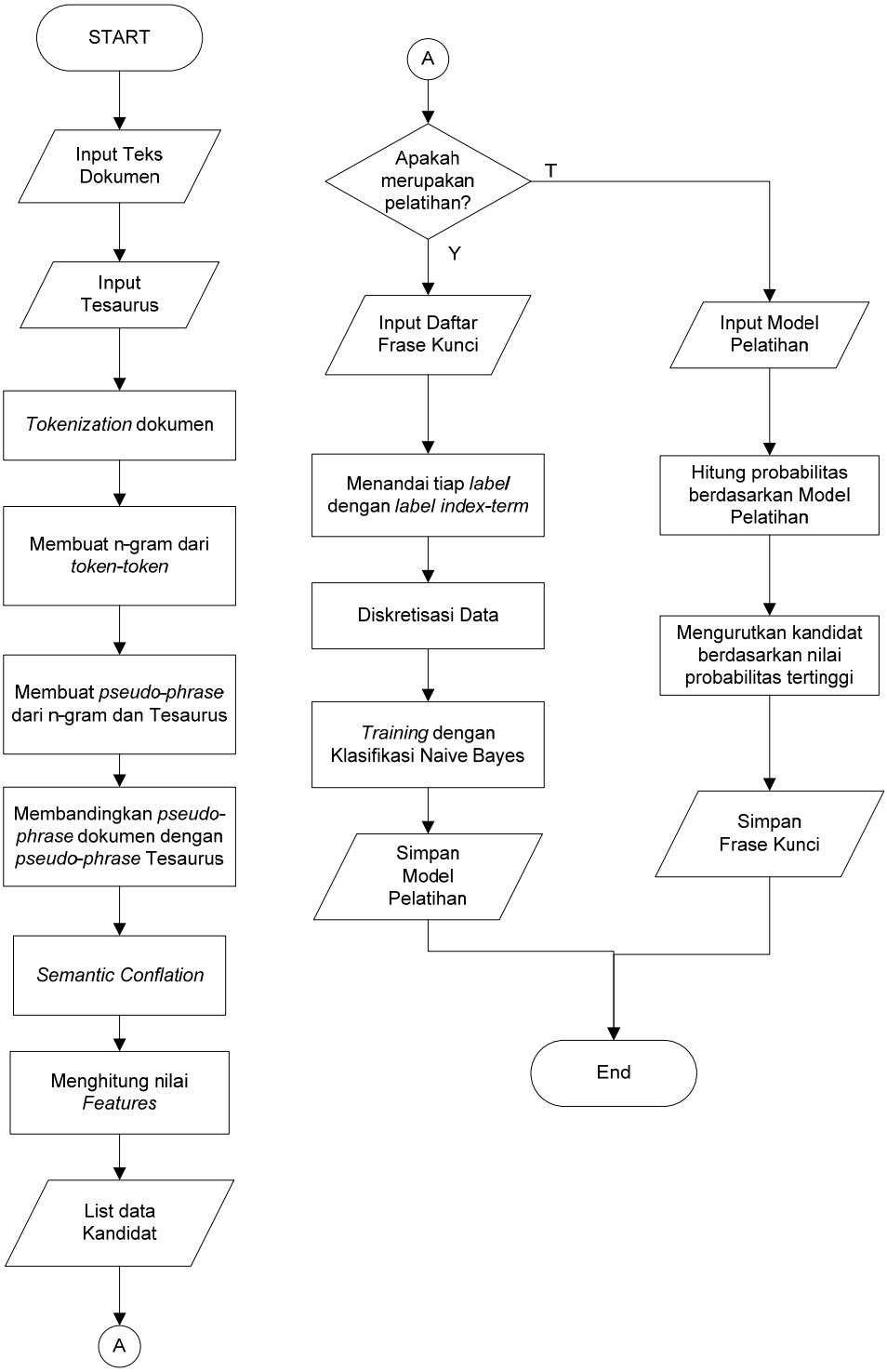
Artikel ilmiah yang digunakan adalah 30 jurnal ilmiah bidang Akuntansi. Jurnal yang digunakan diunduh dari perpustakaan digital jurnal dari universitas-universitas. Indeks yang digunakan untuk masing-masing jurnal berasal dari setiap kata kunci yang disediakan dalam jurnal tersebut.

3) Tesaurus berbahasa Indonesia

Dalam pelaksanaan penelitian ini, ada kendala dalam penyediaan Tesaurus Bahasa Indonesia dari bidang-bidang ilmiah. Tesaurus yang tersedia dalam bahasa Indonesia adalah Tesaurus Bahasa Indonesia secara umum dan tidak dalam bidang yang spesifik. Sebagai pengganti, penelitian ini menggunakan kumpulan istilah Akuntansi yang diperoleh dari Kamus Terbaru Ekonomi dan Bisnis Edisi Lengkap yang disusun oleh drs. Waluyo Hadi dan Dini Hastuti, S.E. jumlah konsep yang digunakan dalam penelitian ini adalah sebanyak 498 konsep yang terdiri atas 972 kata yang dipilih dari kata-kata dalam buku referensi yang muncul dalam jurnal yang digunakan. Kekurangannya adalah tidak adanya relasi antar istilah dan hanya berisi daftar istilah beserta sinonimnya dalam bahasa Indonesia dan bahasa Inggris.

3.2 *Flowchart* Algoritma KEA++

Algoritma KEA++ secara umum terdiri dari dua tahap utama, yaitu Identifikasi Kandidat dan *filtering*. Diagram alir untuk algoritma KEA++ dapat dilihat pada Gambar 3.1.



Gambar 3.1. Diagram Alir Algoritma KEA++

3.3 Perancangan Basis Data

Dalam perancangan sistem pencarian artikel ilmiah berbahasa Indonesia, diperlukan basis data untuk menyimpan data mengenai data artikel ilmiah dan data pengguna. Tabel 3.1. menunjukkan mengenai tabel-tabel yang diperlukan dalam sistem pencarian artikel ilmiah berbahasa Indonesia.

Tabel 3.2. menunjukkan mengenai rancangan struktur tabel yang digunakan untuk menyimpan data artikel ilmiah. Dalam sistem pencarian artikel ilmiah, data artikel ilmiah yang disimpan adalah nama berkas, alamat penyimpanan berkas dan frasa kunci. Kolom alamat penyimpanan berkas digunakan untuk menyimpan lokasi menyimpan berkas dalam media penyimpanan. Kolom frasa kunci digunakan untuk menyimpan hasil dari proses penyaringan frasa kunci artikel ilmiah.

Tabel 3.3. menunjukkan rancangan struktur tabel yang digunakan untuk menyimpan data pengguna. Tabel ini berguna untuk menyimpan data pengguna yang berhak mengakses menu administrator.

Tabel 3.1. Daftar tabel pada Sistem Pencarian Artikel Ilmiah

No.	**Nama Tabel**	**Field**	***Primary Key***	**Deskripsi**
1	Teks	id alamatfile namafile frasakunci	Id	Berisi informasi tentang artikel ilmiah yang digunakan oleh sistem untuk proses pencarian dan hasil dari proses penyaringan frasa kunci.
2	Pengguna	id username password	Idberkas	Berisi daftar pengguna yang dapat mengakses menu administrator

Tabel 3.2. Struktur tabel Teks

No.	**Field**	**Tipe Data**	**Null**	**Keterangan**
1	Id	Integer	No	Nomor id berkas, *auto- increment*
2	alamatfile	Text	No	Alamat penyimpanan artikel ilmiah
3	namafile	Text	No	Nama artikel ilmiah
4	frasakunci	Text	No	Frasa kunci artikel ilmiah

Tabel 3.3 Struktur tabel Pengguna

No.	**Field**	**Tipe Data**	**Null**	**Keterangan**
1	id	Integer	No	Nomor id berkas, *auto- increment*
2	*username*	Text	No	*Username* dari pengguna
3	*password*	Text	No	*Password* dari pengguna

4. Implementasi dan Pengujian

Pengujian akurasi dari penyaringan frasa kunci algoritma KEA++ dilakukan dengan metode *k-fold cross validation* dan *precision-recall*. Dalam pengujian, digunakan 30 dokumen pengujian yang dibagi dalam 3 *fold*. Pembagian dokumen ke dalam *fold-fold* dilakukan secara acak. Masing-masing *fold* digunakan sebanyak 1 kali sebagai *fold* pengujian dan 2 kali sebagai pelatihan.

Jumlah frasa kunci yang disaring adalah 7 untuk masing-masing *fold*. Jumlah ini dipilih karena merupakan jumlah frasa kunci terbanyak pada data jurnal yang digunakan. Panjang maksimal frasa untuk setiap pengujian adalah 5 karena merupakan jumlah frasa yang terpanjang yang tersedia dalam Tesaurus dan panjang minimal frasa adalah 1 sesuai dengan panjang terpendek frasa yang tersedia dalam Tesaurus.

Indikator yang digunakan dalam pengujian ini adalah *precision*, *recall* dan *F-score*. Nilai rata-rata maksimal dari *precision* dalam pengujian tidak akan melebihi 63.33%. Hal ini disebabkan karena nilai 63.33% adalah jumlah rata-rata frasa kunci yang tersedia dalam jurnal dibagi dengan jumlah total frasa kunci yang dihasilkan dalam pengujian.

Ada dua jenis pengujian yang dilaksanakan untuk sistem pencarian artikel ilmiah berbahasa Indonesia. Pengujian yang pertama adalah mengenai jumlah dokumen yang digunakan untuk pelatihan. Pengujian kedua adalah mengenai efek penggunaan Tesaurus.

4.1 Perubahan Jumlah Artikel Ilmiah pada Direktori Pelatihan

Salah satu faktor yang mempengaruhi hasil dari pelatihan adalah model yang digunakan. Model dihasilkan dari proses *filtering* dari pengolahan sejumlah jurnal pada direktori pelatihan. Dalam pengujian ini, dievaluasi pengaruh jumlah dokumen dalam direktori pelatihan untuk kinerja penyaringan frasa kunci artikel ilmiah berbahasa Indonesia menggunakan algoritma KEA++.

Dalam masing-masing *fold*, terdapat 20 dokumen pelatihan dan 10 dokumen pengujian. Pada pengujian untuk mengetahui pengaruh jumlah dokumen pelatihan, proses dibagi menjadi 3 tahap untuk pembuatan model. Tahap pertama adalah 10 dokumen pelatihan yang diambil secara acak dalam masing-masing *fold* yang digunakan dalam proses pembuatan model. Tahap kedua 15 dokumen diambil secara

acak dalam masing-masing *fold* dan tahap ketiga adalah 20 dokumen pelatihan. Setelah model dari masing-masing tahap diperoleh, model digunakan untuk menyaring frasa kunci dari 10 dokumen pengujian.

Setelah pengujian untuk masing-masing model dilaksanakan, jumlah frasa kunci yang sesuai dengan frasa kunci yang ditentukan oleh penulis dihitung untuk masing-masing *fold*. Yang dimaksud dengan frasa kunci yang sesuai adalah frasa kunci yang sama persis atau merupakan satu konsep dalam Tesaurus dengan frasa kunci yang dipilih oleh penulis. Tabel 4.1. menunjukkan contoh hasil penyaringan frasa kunci. Dalam tabel tersebut, dapat dilihat bahwa jumlah frasa kunci yang dihasilkan oleh sistem yang sama dengan frasa kunci yang dihasilkan oleh penulis jurnal adalah sebanyak 3 frasa kunci.

Tabel 4.1. Perbandingan Hasil Penyaringan Frasa Kunci

"Pengaruh Kemajuan Teknologi Informasi Terhadap Perkembangan Akuntansi"	
Frasa Kunci dari Jurnal	**Hasil Penyaringan Frasa Kunci**
- teknologi informasi - **akuntansi** - **sistem informasi akuntansi** - **audit**	- **akuntansi** - **sistem informasi akuntansi** - **audit** - komputer - sistem informasi - teknologi - akuntan

Nilai *precision*(P), *recall*(R) dan *F-score*(F) untuk pengujian ini secara lengkap ditunjukkan pada tabel 4.2. Dari hasil pengujian, nilai rata-rata dari *precision, recall* dan *F-score* mengalami peningkatan seiring dengan bertambahnya jumlah data pelatihan. Dapat disimpulkan bahwa semakin banyak dokumen yang digunakan untuk pelatihan, maka akurasi dari penyaringan frasa kunci semakin meningkat. Karena keterbatasan jumlah data, pengujian hanya dilakukan sampai maksimal 20 dokumen.

Tabel 4.2. Nilai *P, R* dan *F* untuk jumlah dokumen pelatihan yang berbeda

Fold	**10 dokumen**			**15 dokumen**			**20 dokumen**		
	P(%)	**R(%)**	**F(%)**	**P(%)**	**R(%)**	**F(%)**	**P(%)**	**R(%)**	**F(%)**
Fold 1	18.6	30.9	23.2	18.6	30.9	23.2	20	33.3	25
Fold 2	27.1	40.4	32.5	28.6	42.5	34.2	28.5	42.5	34
Fold 3	24.3	38.6	29.8	22.9	36.4	28.1	22.9	36.4	28.1
Rata-rata	23.3	36.6	28.5	23.4	36.6	28.5	**23.8**	**37.4**	**29**

4.2 Evaluasi Penggunaan Tesaurus

KEA++ merupakan algoritma yang memerlukan Tesaurus dalam proses penyaringan frasa kunci. Dalam pengujian ini, dievaluasi bagaimana pengaruh Tesaurus dalam kinerja penyaringan frasa kunci artikel ilmiah berbahasa Indonesia menggunakan algoritma KEA++ dibandingkan dengan tidak menggunakan Tesaurus.

Dalam pengujian untuk setiap *fold*, digunakan 20 dokumen pelatihan dan 10 dokumen pengujian. Masing-masing *fold* diproses sebanyak dua kali yaitu dengan menggunakan Tesaurus dan tidak menggunakan Tesaurus. Setelah pengujian dilaksanakan, jumlah frasa kunci yang sesuai dengan frasa kunci yang ditentukan oleh penulis dihitung.

Nilai *precision*(P), *recall*(R) dan *F-score*(F) untuk pengujian ini ditunjukkan pada tabel 4.3. Dari hasil pengujian, nilai rata-rata dari *precision, recall* dan *F-score* dari *fold* yang diuji dengan menggunakan Tesaurus jauh lebih tinggi daripada yang tidak menggunakan tesaurus. Dapat disimpulkan bahwa akurasi dari penyaringan frasa kunci menggunakan Tesaurus lebih baik daripada tanpa Tesaurus. Penyaringan frasa tanpa menggunakan Tesaurus dapat menyaring frasa yang tidak tersedia dalam Tesaurus. Tetapi, kekurangannya adalah banyak frasa yang tidak signifikan yang muncul dalam hasil penyaringan. Tabel 4.4. menunjukkan contoh perbandingan frasa kunci yang dihasilkan dari penyaringan frasa kunci tanpa menggunakan Tesaurus dan dengan menggunakan Tesaurus. Istilah **teknologi informasi** yang tidak tersedia dalam Tesaurus dapat disaring oleh penyaringan tanpa Tesaurus, tetapi jumlah frasa kunci yang sesuai jauh lebih sedikit.

Tabel 4.3 Perbandingan nilai P, R dan F dari penggunaan Tesaurus dan tanpa Tesaurus

Fold	**tanpa Tesaurus**			**dengan Tesaurus**		
	P(%)	**R(%)**	**F(%)**	**P(%)**	**R(%)**	**F(%)**
Fold 1	17.1	28.6	21.4	22,9	38,1	28,6
Fold 2	11,4	17	13,7	25,7	38,3	30,1
Fold 3	15,7	25	19,3	20	31,8	24,6
Rata-rata	16,7	26,7	20,6	**22,9**	**36,1**	**27.8**

Tabel 4.4. Perbandingan hasil penyaringan frasa kunci tanpa menggunakan Tesaurus dan dengan menggunakan Tesaurus

"Pengaruh Kemajuan Teknologi Informasi Terhadap Perkembangan Akuntansi"		
Frasa Kunci dari Jurnal	**Penyaringan tanpa Tesaurus**	**Penyaringan dengan Tesaurus**
- teknologi informasi - akuntansi - sistem informasi akuntansi - audit	- teknologi informasi - akuntansi - informasi - teknologi - kemajuan - perkembangan - perkembangan akuntansi	- akuntansi - sistem informasi akuntansi - audit - komputer - sistem informasi - teknologi - akuntan

4.3 Evaluasi oleh Responden

Evaluasi yang melibatkan responden bertujuan untuk mengetahui apakah hasil dari penyaringan frasa kunci artikel ilmiah berbahasa Indonesia dapat diterima oleh manusia. Pengujian ini dilakukan karena tujuan awal dari penyaringan frasa kunci adalah bertujuan untuk membantu manusia dalam menentukan frasa kunci dari sebuah artikel ilmiah.

Evaluasi untuk mengukur tingkat kepuasan responden hanya dilakukan dengan menggunakan 1 *fold* saja karena keterbatasan waktu dari responden. Untuk evaluasi responden, jumlah dokumen pelatihan yang digunakan adalah 20 dan jumlah frasa kunci untuk masing-masing artikel ilmiah dalam pengujian sebanyak 7 frasa kunci. *Fold* yang digunakan adalah *fold* yang memiliki nilai *f-score* yang tertinggi yaitu *fold* 2. Jumlah responden yang berpartisipasi sebanyak 9 orang.

Hasil dari evaluasi oleh responden dapat dilihat pada Tabel 4.5.

Tabel 4.5. Hasil Evaluasi oleh Responden

Pekerjaan Responden	**Jumlah frasa kunci yang sesuai per artikel ilmiah**										**Nilai**
	1	2	3	4	5	6	7	8	9	10	
Karyawan	4	4	6	4	1	1	4	4	3	6	3
Karyawan	4	6	4	3	3	1	2	3	3	2	3
Karyawan	3	5	4	4	2	2	3	2	2	4	3
Staf Keuangan	5	4	5	4	5	3	4	6	6	5	3
Staf Keuangan	5	5	6	5	4	2	4	6	4	5	4
Mahasiswa	6	6	6	5	4	5	5	5	6	5	4
Mahasiswa	3	3	3	2	2	2	3	3	3	3	4
Mahasiswa	5	3	4	2	2	2	3	3	3	3	4
Mahasiswa	3	3	3	2	2	1	2	2	2	2	3
Total	38	39	41	31	25	19	29	35	33	36	31

Dari penilaian responden, jumlah total frasa kunci yang sesuai dengan artikel ilmiah yang terkait adalah sebanyak 326 frasa kunci dari jumlah total sebanyak 630 frasa kunci (70 frasa kunci x 9 responden). Persentase jumlah frasa kunci yang sesuai adalah sebanyak 51.75%. Karena jumlah skor dari penilaian responden adalah 31 dari total skor 45, maka hasil dari penyaringan frasa kunci artikel ilmiah berbahasa Indonesia dinilai baik.

4.4 Kendala yang Dihadapi

Ada beberapa hal yang menjadi penyebab rendahnya akurasi penyaringan frasa kunci algoritma KEA++, antara lain :

1) Tesaurus atau buku referensi yang digunakan tidak lengkap

Faktor ini merupakan faktor utama yang menyebabkan rendahnya akurasi penyaringan frasa kunci algoritma KEA++. Dari 133 frasa kunci yang tersedia dalam jurnal, 54 frasa kunci tidak terdapat dalam Tesaurus atau buku referensi yang digunakan. Banyak frasa yang tidak tersedia yang disebabkan oleh perbedaan dalam mengartikan frasa yang berasal dari bahasa Inggris. Hal ini menyebabkan frekuensi kemunculan dari frasa yang memiliki arti yang sama menjadi berkurang.

2) *Pseudophrase* yang sama

Dalam pembentukan *pseudophrase* ada beberapa frasa yang memiliki *pesudophrase* yang sama. Frasa pelaporan keuangan dan laporan keuangan

memiliki *pseudophrase* yang sama yaitu lapor uang. Frekuensi untuk kemunculan laporan keuangan akan dijumlahkan ke pelaporan keuangan karena frasa pelaporan keuangan muncul lebih awal dalam Tesaurus yang digunakan. Kesamaan frasa ini menyebabkan beberapa frasa menjadi hilang dari daftar kandidat dan digantikan dengan frasa lain yang tidak signifikan.

3) *Header* atau *footer* halaman yang ikut terkonversi

Nomor halaman dan *header* dari masing-masing halaman turut mengganggu dalam proses penyusunan *n-gram* kandidat.

5. Penutup

5.1 Kesimpulan

Kesimpulan yang dapat diambil dari pembuatan tugas akhir ini adalah sebagai berikut :

1) Telah dihasilkan sistem pencarian artikel ilmiah yang memanfaatkan penyaringan frasa kunci menggunakan algoritma KEA++.
2) Berdasarkan pengujian yang dilakukan, semakin banyak jumlah data pelatihan yang digunakan maka semakin tinggi pula akurasi dari proses penyaringan frasa kunci artikel ilmiah berbahasa Indonesia.
3) Berdasarkan pengujian yang dilakukan, penggunaan Tesaurus dapat meningkatkan akurasi dalam proses penyaringan frasa kunci artikel ilmiah berbahasa Indonesia.
4) Berdasarkan evaluasi dari responden terhadap 10 dokumen yang disaring frasa kuncinya, persentase jumlah frasa kunci yang sesuai adalah sebanyak 51.75%.
5) Berdasarkan evaluasi dari responden terhadap 10 dokumen yang disaring frasa kuncinya, sistem memperoleh skor 31 dari total skor 45 dan tergolong baik.

5.2 Saran

Saran-saran dari penulis untuk pengembangan lebih lanjut penelitian ini adalah sebagai berikut :

1) Proses penyaringan frasa kunci artikel ilmiah berbahasa Indonesia dengan menggunakan algoritma KEA++ belum diujicobakan untuk kumpulan artikel ilmiah yang memiliki jumlah cukup besar (ratusan atau ribuan). Perlu dilaksanakan penelitian lebih lanjut mengenai bagaimana kinerja proses

penyaringan frasa kunci apabila artikel ilmiah yang digunakan jumlahnya cukup besar.

2) Tesaurus yang tidak tersedia relasi antar konsep menyebabkan salah satu *feature* dalam algoritma KEA++, yaitu *node degree* tidak dapat diimplementasikan. Pada penelitian selanjutnya, diharapkan dapat digunakan Tesaurus yang memiliki relasi antar konsep sehingga proses penyaringan frasa kunci dengan algoritma KEA++ dapat berjalan dengan maksimal.
3) Pada penelitian selanjutnya, diharapkan dapat digunakan Tesaurus yang lebih lengkap konsep dan istilah-istilahnya sehingga dapat meningkatkan akurasi dari proses penyaringan frasa kunci artikel ilmiah berbahasa Indonesia.
4) Penyaringan frasa kunci otomatis dalam sistem Pencarian Artikel Ilmiah dapat dikembangkan lebih lanjut untuk penyaringan frasa kunci secara semiotomatis.
5) Dapat dilaksanakan penelitian lebih lanjut mengenai pemanfaatan Tesaurus yang digunakan dalam proses penyaringan frasa kunci untuk membantu proses pencarian dengan menggunakan sinonim atau relasi antar kata yang tersedia dalam Tesaurus.
6) Sistem dapat dikembangkan lebih lanjut dengan melengkapi sistem dengan manajemen data artikel ilmiah.

6. Daftar Pustaka

[1] Axmark D, dkk, 2001, *"mySQL Manual"*, mySQL AB

[2] Budhi GS et al., 2006, *"Algoritma Porter Stemmer For Bahasa Indonesia untuk Pre-Processing Text Mining Berbasis Metode Market Basket Analysis"*, UK Petra

[3] Dillard GT, *"The Scientific Paper"*, diakses dari bioweb.wku.edu pada 28 April 2011 pukul 18.15

[4] Frank E and Medelyan O, *"Keyphrase Extraction Algorithm Description"*, Diakses dari www.nzdl.org pada 17 Oktober 2011 pukul 07.32 WIB

[5] Frank E and Witten I, 2005, *"Data Mining : Practical Machine Learning Tools and Techniques 2nd Ed"*, Morgan Kaufmann, San Francisco.

[6] Hammouda KM et al., 2006, *"CorePhrase : Keyphrase Extraction for Document Clustering"*, University of Waterloo, Waterloo.

[7] Hariyanto B, 2007, *"Esensi-Esensi Bahasa Pemrograman Java"*, Informatika, Bandung.

[8] Isaac A and Summers E, 2009, *"SKOS Simple Knowledge Organization System"*, diakses dari www.w3.org pada 7 Maret 2011 pukul 08.05 WIB

[9] Jiawei H and Kamber M, 2006, *"Data Mining Concept and Techniques Second Edition"*, Elsevier Inc, San Francisco.

[10] Lui YJ, 2007, *"Extraction Significant Phrases from Text"*, International Journal of Electrical and Computer Engineering, vol 2, no 2, pp 101-109

[11] Medelyan O, 2005, *"Automatic Keyphrase Indexing with a Domain-Specific Thesaurus"*, Universitat Freiburg, Freiburg.

[12] Medelyan O et al., 2006, *"Thesaurus Based Automatic Keyphrase Indexing"*, JCDL '06, Chapel Hill.

[13] Pastor-Sanchez JA and al. e, 2009, *"Advantage of Thesaurus Representation using the Simple Knowledge Organisation System (SKOS) Compared with Proposed Alternatives"*, Information Research, vol 14, no 4

[14] Tala F, 2003, *"A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia"*, Institute of Logic, Language and Computation Universiteit Van Amsterdam, Amsterdam.