SKRIPSI

TEMU KEMBALI INFORMASI DENGAN EKSPANSI PERMINTAAN MENGGUNAKAN MATRIKS ASOSIASI

ARIF PRADANA

NPM: 201173088

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2017

UNDERGRADUATE THESIS

INFORMATION RETRIEVAL WITH QUERY EXPANSION USING ASSOCIATION MATRIX

ARIF PRADANA

NPM: 201173088

DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2017

LEMBAR PENGESAHAN

TEMU KEMBALI INFORMASI DENGAN EKSPANSI PERMINTAAN MENGGUNAKAN MATRIKS ASOSIASI

ARIF PRADANA

NPM: 201173088

Bandung, 20 Desember 2016

Menyetujui,

Pembimbing

Luciana Abednego, M.T.

Ketua Tim Penguji

Chandra Wijaya, M.T.

Anggota Tim Penguji

Dr. Veronica Sri Moertini

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

TEMU KEMBALI INFORMASI DENGAN EKSPANSI PERMINTAAN MENGGUNAKAN MATRIKS ASOSIASI

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 5 Januari 2017

Arif Pradana
NPM: 201173088

ABSTRAK

Suatu aplikasi temu kembali informasi diharapkan dapat membuat pengguna menemukan banyak informasi yang dibutuhkannya. Namun kenyataannya bahwa sumber informasi berupa dokumen pada suatu tempat penyimpanan yang dipakai suatu aplikasi adalah terbatas. Hal tersebut dapat menyebabkan informasi yang didapatkan oleh pengguna tidak mencukupi harapannya. Masalah lainnya adalah pengguna tidak mengetahui informasi apa saja yang disimpan dalam suatu tempat penyimpanan yang dipakai oleh suatu aplikasi. Akibatnya, sulit bagi pengguna untuk memasukkan kata kunci yang tepat sebagai permintaannya terhadap aplikasi tersebut agar mampu mendapatkan sejumlah informasi yang dibutuhkan. Oleh karena itu, tujuan penelitian ini adalah untuk membangun perangkat lunak yang mampu meningkatkan jumlah hasil informasi yang relevan terhadap permintaan pengguna.

Ekspansi permintan merupakan metode yang dipakai dalam menemukan kembali informasi untuk meningkatkan jumlah informasi yang diberikan kepada pengguna. Perangkat lunak akan melakukan ekspansi permintaan dengan menggunakan Matriks Asosiasi. Matriks Asosiasi berisi nilai dari setiap pasangan kandidat ekspansi yang menunjukkan seberapa besar pasangan tersebut dapat mewakili permintaan pengguna terhadap sumber informasi yang dipakai. Semua kandidat yang terpilih akan diseleksi. Hasil ekspansi merupakan pasangan kata yang relevan dengan topik yang dibutuhkan oleh pengguna yang telah dicocokan dengan sumber informasi yang dipakai perangkat lunak.

Berbagai pengujian dan eksperimen pada perangkat lunak telah dilakukan dengan 101 dokumen dan 21032 kata tidak berulang. Telah didapatkan hasil dari setiap eksperimen bahwa dengan hasil ekspansi permintaan menggunakan Matriks Asosiasi, perangkat lunak berhasil menemukan lebih banyak informasi yang relevan dengan permintaan pengguna. Pada setiap eksperimen juga dilakukan pengukuran performa setelah ekspansi dan didapat bahwa nilai *recall* setelah melakukan ekspansi meningkat dan tetap menjaga nilai *precision* tidak turun terlalu besar atau bahkan tetap.

Kata-kata kunci: Temu Kembali Informasi, Ekspansi Permintaan, Matriks Asosisasi

ABSTRACT

An information retrieval application is expected to make the users find a lot of information is needed. But the fact that information resources in the form of a document in a storage area which used by application is limited. This can lead to information obtained by the user insufficient hopes. The other problem is the users does not know the information what is stored in a storage area used by an application. As a result, it is difficult for the user to enter the keyword precisely as a requirement for the application to be able to get some information needed. Therefore, the purpose of this study was to build software that is able to increase the amount of information results that relevant to user query.

Query expansion is a method used in Information Retrieval to find back the information to increase the amount of information which given to users. The software will expand the query by using Association Matrix. Association matrix contains the value of each pair of expansion candidates that shows how big the value for the pair to represent user query against resources used. All shortlisted candidates will be selected. The result of the expansion is word pairs that are relevant to the topics needed by users which have been matched with the information resources that used by the software.

Various tests and experiments on the software has been done for 101 documents and 21032 words is not repetitive. The results has been obtained from each experiment that by using the results of the query expansion using Association Matrix, the software can find more number of relevant information based on user query. In each experiment was also carried out measurements of performance after expansion and found that the recall value after the expansion is increasing while the value of precision does not fall too big or even fixed.

Keywords: Information Retrieval, Query Expansion, Association Matrix

Dipersembahkan untuk kedua orang tua

KATA PENGANTAR

Segala puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa, karena kasih dan ridho-Nya, sehingga penulis diberi kesempatan meraih gelar sarjana teknik dari Program Studi Teknik Informatika, Fakultas Teknologi Informasi dan Sains, Universitas Katolik Parahyangan. Penulis akan selalu berusaha dan berdoa agar dengan menyandang gelar sarjana ini, penulis dapat membaktikan diri untuk kebaikan umat manusia.

Skripsi yang berjudul "Temu Kembali Informasi dengan Ekspansi *Query* menggunakan Matriks Asosiasi" ini dibuat sebagai tugas akhir dan persyaratan untuk menyandang gelar sarjana. Judul tersebut merupakan pengembangan dari topik "Temu Kembali Informasi dengan Ekspansi *Query*" yang disediakan oleh pembimbing penulis yaitu Ibu Luciana Abednego. Pemilihan topik ini didasari minat penulis dalam penelitian ilmu komputer.

Penulis mendapat berbagai dukungan dari banyak pihak terutama saat menempuh kuliah dan pembuatan skripsi ini sehingga penulis dapat menyelesaikannya. Oleh karena itu, penulis ingin mengucapkan terima kasih kepada pihak-pihak yang telah mendukung dan membantu penulis:

1. Ibu Luciana Abednego sebagai pembimbing penulis yang selalu memberi arahan dan dukungan serta sebagai dosen yang mengajar kuliah Temu Kembali Informasi.
2. Bapak Chandra sebagai dosen *reviewer* skripsi I dan penguji skripsi II yang memberikan saran dan kritik yang berguna pada skripsi ini.
3. Ibu Veronica Sri Moertini sebagai dosen penguji skripsi II yang telah memberikan saran dan kritik yang berguna pada skripsi ini.
4. Bapak Thomas Anung Basuki sebagai dosen yang telah mengajar kuliah Pengolahan Bahasa Alami.
5. Seluruh dosen yang telah mengajar penulis dengan baik.
6. Kedua orang tua yang selalu mendukung dan memberi doa.
7. Adik penulis yang selalu ada dan mendoakan.
8. Reinhart dan Andre sebagai teman yang selalu ada dan menyemangati.
9. Teman-teman khususnya selama menempuh kuliah yaitu Sudarsono, Anton, Lucas, Daniel, Adjie, Andi, Adit, Alex, Donny, Irvan dan lainnya yang belum disebutkan.

Penulis menyadari bahwa skripsi ini masih belum sempurna. Oleh karena itu, penulis memohon maaf jika terdapat kekurangan. Demi perbaikan dan kemajuan selanjutnya, saran dan kritik yang membangun akan penulis terima dengan senang hati. Penulis berharap tugas akhir ini dapat bermanfaat dan memberikan kontribusi pada penelitian atau pembelajaran selanjutnya.

Bandung, Januari 2017

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Information Retrieval merupakan aktivitas untuk menemukan kembali informasi yang tersimpan dari suatu *corpus*(koleksi teks berisi informasi)[1]. Dalam pencariannya, beberapa jenis data dapat ditemukan diantaranya teks, gambar, video, dan audio. Adapun tujuan dari *Infromation Retrieval* adalah untuk menemukan kembali informasi-informasi yang relevan terhadap kebutuhan pengguna. Salah satu contoh aplikasi yang sudah terkenal di bidang *Information Retrieval* adalah Google Search.

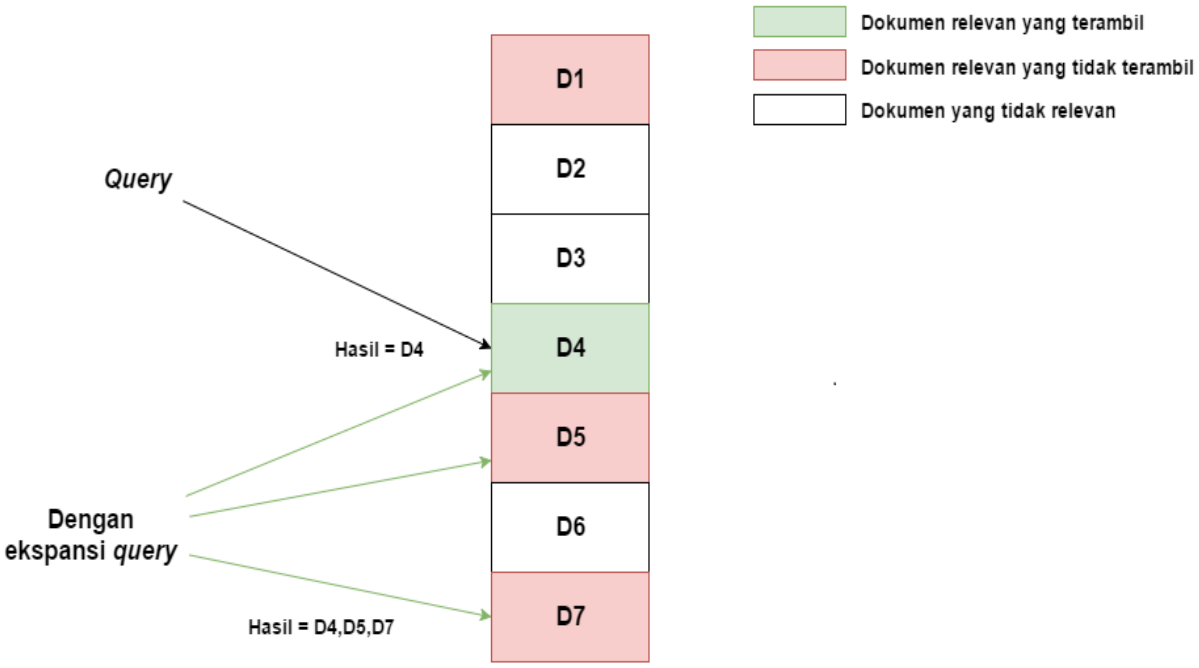
Saat ini melakukan pencarian untuk menemukan informasi menggunakan jaringan internet telah menjadi salah satu kegiatan yang sering dilakukan. Pengguna akan senang jika setelah melakukan pencarian ditemukan suatu hasil yang mengandung informasi mengenai kebutuhannya. Namun, tidak jarang pengguna mendapatkan banyak hasil yang tidak relevan dengan kebutuhannya. Dengan kata lain, hanya terdapat beberapa saja yang relevan dan bahkan mungkin tidak ada. Hal seperti itu membuat pengguna tidak puas dengan suatu aplikasi *Information Retrieval*.

Salah satu masalah lain dalam *Information Retrieval* yaitu berkaitan dengan kata kunci pencarian yang mewakili permintaan kebutuhan pengguna yang biasa disebut sebagai *query*. Kenyataannya pengguna sulit untuk memasukkan *query* yang benar-benar tepat dan mewakili kebutuhan informasinya. Hal ini disebabkan karena pengguna tidak mengetahui isi *corpus* dari suatu sistem *Information Retrieval* dan bagaimana operasi yang dilakukan dalam suatu sistem *Information Retrieval* terhadap *query* yang telah dimasukkan.

Terdapat beberapa kemungkinan dari hasil pencarian untuk menemukan informasi berupa dokumen pada suatu sistem *Information Retrieval* yaitu antara lain adalah terlalu banyak dokumen yang dikembalikan, terlalu sedikit hasil yang dikembalikan dan bahkan tidak ada dokumen yang dikembalikan. Semua kemungkinan tersebut, harus dioptimalkan agar pengguna puas terhadap suatu aplikasi *Information Retrieval*. Jika terlalu banyak yang ditemukan, maka pengguna harus dapat dengan mudah mencari dokumen yang relevan diantara semua hasil dokumen yang telah ditemukan. Untuk kasus dengan hasil penemuan dokumen yang sedikit, diperlukan suatu cara untuk menambah jumlah hasil penemuan dokumen yang relevan. Sebuah *corpus* pada suatu sistem *Information Retrieval* bisa saja tidak mengandung kata kunci pada *query* yang pengguna masukkan, tetapi bukan tidak mungkin bahwa beberapa dokumen pada suatu *corpus* tersebut mengandung informasi yang mirip dengan yang pengguna butuhkan.

Penelitian tentang bagaimana cara untuk menambah hasil dokumen relevan yang didapatk-

an dari suatu pencarian dalam suatu sistem *Information Retrieval* telah banyak dilakukan. *Query expansion* atau ekspansi *query* telah lama diusulkan sebagai suatu metode atau cara untuk meningkatkan efektivitas pencarian informasi untuk menambah hasil penemuan dokumen yang relevan. Ekspansi *query* adalah sebuah metode melengkapi istilah atau frase tambahan pada *query*(permintaan pengguna) aslinya untuk meningkatkan kinerja pengambilan. Metode ini dapat membuat proses pencarian menjadi lebih efektif karena mesin pencarian dapat mencari lebih jauh dengan menggunakan bantuan kata atau frasa lain yang mirip dan tetap mewakili permintaan yang pengguna masukkan. Lebih jelasnya ilustrasi ekspansi *query* dapat dilihat pada Gambar 1.1.



Gambar 1.1: Ilustrasi Ekspansi *Query*

Pada Gambar 1.1 terdapat 1 hasil dokumen relevan sebagai jawaban dari *query* yang dimasukkan oleh pengguna yaitu D_4. Dapat dilihat bahwa terdapat dokumen lainnya yang relevan terhadap *query* dan tidak terambil dengan *query* aslinya yang ditandakan dengan warna merah muda. Ekspansi *query* dapat dilakukan pada *Information Retrieval* agar dokumen relevan lain dapat dikeluarkan sebagai hasil yang diberikan kepada pengguna. Hasilnya bahwa dengan ekspansi *query* pada Gambar 1.1 membuat dokumen D_5 dan D_7 menjadi terambil sebagai hasil yang diberikan kepada pengguna.

Matriks Asosiasi merupakan salah satu cara untuk melakukan metode ekspansi *query*. Dengan cara ini, perangkat lunak akan mendapatkan *query* baru sebagai hasil perluasan *query* aslinya yang mewakili dokumen yang memiliki kemiripan tinggi terhadap *query* yang dimasukkan pengguna. Prinsip kerjanya adalah mencari nilai kecocokan dari setiap pasangan kata yang telah terpilih sebagai kata yang akan menjadi kandidat hasil ekspansi. Setiap kata yang menjadi kandidat tersebut merupakan kata yang berpotensi menjadi *query* baru yang mewakili *query* pengguna terhadap *corpus* yang dipakai. Lebih jelasnya contoh Matriks Asosiasi dapat dilihat pada Gambar 1.2.

$$
\begin{array}{c} \\ T_1 \\ T_3 \\ T_{205} \\ T_{409} \\ T_{705} \\ T_{901} \end{array}
\begin{array}{c}
\begin{array}{cccccc} T_1 & T_3 & T_{205} & T_{409} & T_{705} & T_{901} \end{array} \\
\left[\begin{array}{cccccc}
C_{1,1} & C_{1,3} & C_{1,205} & C_{1,409} & C_{1,705} & C_{1,901} \\
C_{3,1} & C_{3,3} & C_{3,205} & C_{3,409} & C_{3,705} & C_{3,901} \\
C_{205,1} & C_{205,3} & C_{205,205} & C_{205,409} & C_{205,705} & C_{205,901} \\
C_{409,1} & C_{409,3} & C_{409,205} & C_{409,409} & C_{409,705} & C_{409,901} \\
C_{705,1} & C_{705,3} & C_{705,205} & C_{705,409} & C_{705,705} & C_{705,901} \\
C_{901,1} & C_{901,3} & C_{901,205} & C_{901,409} & C_{901,705} & C_{901,901}
\end{array} \right]
\end{array}
$$

Gambar 1.2: Contoh Matriks Asosiasi yang Berisi Nilai Seluruh Kandidat Ekpansi

Pada Gambar 1.2 terdapat 6 *term* sebagai kandidat ekspansi dari total *term* yang ada pada suatu corpus 1000 *term*. *Term* dalam Information *Retrieval* merupakan kata yang digunakan dalam melakukan setiap proses *Information Retrieval*. Setiap pasangan *term* yang menjadi kandidat ekspansi akan dihitung nilai C yang disebut sebagai *correlation factor*. Nilai C kemudian akan dilakukan normalisasi menjadi nilai yang disebut *normalized association score*. Nilai *normalized association score* menandakan seberapa besar pasangan tersebut mewakili *query* terhadap *corpus* yang dimiliki. Cara menghitung nilai C dan *normalized association score* dapat dilihat pada Persamaan 1.1 dan 1.2.

$$C_{ij} = \sum_{d_k \in D} F_{ik} * F_{jk} \tag{1.1}$$

$$S_{ij} = \frac{C_{ij}}{C_{jj} + C_{ii} - C_{ij}} \tag{1.2}$$

Nilai C_{ij} menyatakan *correlation factor term* i terhadap *term* j, S_{ij} menyatakan *normalized association score term i* terhadap *term j*, dan F_{ik} menyatakan frekuensi kemunculan *term i* pada dokumen k. Beberapa nilai S_{ij} tertinggi merupakan pasangan *term* yang akan menjadi *query* baru yang ditambahkan kepada *query* awal.

Ekspansi *query* menggunakan Matriks Asosiasi bertujuan untuk melakukan seleksi terhadap setiap pasangan kandidat yang telah terpilih untuk menjadi *query* baru yang nantinya ditambahkan kepada *query* aslinya. *Term* kandidat ekspansi *query* yang dimaksud disini adalah *term* terpilih yang mempunyai similaritas tinggi terhadap *query*. Perhitungan untuk melakukan seleksi dilakukan dengan mempertimbangkan isi *corpus* pada suatu sistem. Jika salah satu atau kedua *term* dari suatu pasangan kandidat *term* tidak relevan dengan topik yang dimaksud oleh *query*, maka pasangan tersebut tidak akan lolos seleksi. Suatu pasangan *term* bisa saja memiliki relasi atau kemiripan dengan *query* yang mewakili permintaan kebutuhan pengguna, tetapi jika *term* tersebut tidak terdapat pada *corpus* maka pasangan *term* tersebut tidak akan lolos seleksi.

Perangkat lunak yang dibuat pada skripsi ini bertujuan untuk membuat suatu sistem IR dengan metode ekspansi *query* untuk menambah jumlah dokumen relevan yang dikembalikan kepada pengguna dengan menggunakan Matriks Asosiasi. Hasil ekspansi merupakan pasangan-pasangan *term* yang mewakili *query* terhadap *corpus* yang digunakan. Setiap pasangan *term* hasil ekspansi berpotensi untuk menghasilkan dokumen yang relevan terhadap *query* yang sebelumnya dokumen tersebut belum terambil menggunakan *query* asli.

Terdapat cara lain yang sering digunakan untuk melakukan ekspansi *query* yaitu menggunakan *term* sinonim atau *term* yang mempunyai relasi dari *term* yang ada pada *query*. *Term* terse-

but bisa didapatkan pada layanan Thesaurus yang sudah ada. Thesaurus adalah karya referensi yang berisi daftar kata-kata yang dikelompokkan bersama menurut kesamaan makna (mengandung sinonim dan antonim), berbeda dengan kamus, yang memberikan definisi untuk kata-kata dan umumnya dalam urutan abjad. Dalam *Information Retrieval*, Thesaurus merupakan sebuah alat untuk membantu memperluas atau menambah suatu kosa kata kedalam *query* [3]. Dengan mempertimbangkan hal tersebut, perangkat lunak ini juga menggunakan layanan Thesaurus yang telah ada untuk mendapat *term* sinonim atau *term* yang mempunyai relasi dengan *term query* untuk dimasukkan sebagai kandidat ekspansi *query* yang selanjutnya akan diseleksi menggunakan Matriks Asosiasi.

Untuk membangun perangkat lunak *Information Retrieval* pada skripsi ini terdapat beberapa pemodelan dalam *Information Retrieval* yang dipakai yaitu *Boolean Model* dan *Vector Space Model(VSM)*. Keduanya mempunyai kelebihan tersendiri. Pemodelan *Information Retrieval* dengan *Boolean Model* memungkinkan suatu aplikasi *Information Retrieval* dapat menjawab permintaan pengguna dengan melakakukan operasi menggunakan operator *boolean* yaitu "OR","AND, dan"NOT". Dengan menggunakan model ini, hasil dari operasi *boolean* merupakan jawaban permintaan pengguna. Misalnya, *query* adalah "much money", jika operator yang digunakan adalah "AND" maka jawaban dari pemintaan pengguna adalah semua dokumen yang mengandung kata "much" dan kata "money".

Pemodelan *Information Retrieval* dengan menggunakan *Vector Space Model* dilakukan dengan cara mengumpamakan setiap *term*, dokumen dan *query* sebagai vektor. Dengan menentukan arah vektor dokumen dan *query* maka perhitungan similaritas antara *query* dan dokumen bisa dilakukan dengan cara menghitung kosinus sudut antara kedua vektor. Perangkat lunak yang dibuat pada skripsi ini dapat melakukan pengurutan peringkat dokumen berdasarkan kemiripan dokumen terhadap *query*.

Dalam *Information Retrieval*, isi dokumen(teks) dan *query* melibatkan bahasa manusia. Oleh karena itu diperlukan bidang ilmu lain yang berkaitan dengan interaksi manusia dan komputer untuk mendukung perangkat lunak IR agar lebih efektif dan berjalan dengan baik. Bidang ilmu yang dipakai adalah *Natural Language Processing*(NLP). NLP adalah adalah bidang ilmu komputer, Kecerdasan Buatan dan Komputasi Linguistik yang berkaitan dengan interaksi antara komputer dan bahasa manusia (alami) [4]. Linguistik yang dimaksud disini antara lain adalah mengenai bagaimana struktur dari sebuah kata, penggunaan kata, dan pengartian kata. Terdapat 3 proses dalam NLP yang dipakai dalam perangkat lunak skripsi ini yaitu *tokenization*, *stop word Removal*, dan *stemming*. *Tokenization* merupakan proses untuk memisahkan setiap kata dalam suatu teks. Hasilnya berupa token-token dari setiap kata. Proses kedua yaitu *stop word Removal* merupakan proses yang bertujuan untuk menghilangkan kata-kata yang sering muncul pada suatu bahasa. Hal ini dilakukan karena *stop word* merupakan kata yang tidak diperlukan dalam *Information Retrieval* dan dapat menyebabkan hasil yang salah jika tetap digunakan. Proses ketiga NLP yang dipakai adalah *stemming* merupakan proses yang bertujuan untuk mereduksi segala perubahan perbentukan kata dari suatu kata dasar. Dengan dilakukannya *stemming*, memori yang dipakai bisa lebih sedikit dan suatu kata bisa mewakili perubahan bentuk katanya yang merupakan relasi dari suatu kata dasar.

Dengan demikian, perangkat lunak *Information Retrieval* yang dibuat pada skirpsi ini akan

dimodelkan dengan *Boolean Model* dan *Vector Space Model*. Metode ekspansi *query* akan dilakukan untuk menambah jumlah dokumen relevan yang sebelumnya tidak terambil menggunakan *query* asli. Ekspansi *query* dilakukan dengan menggunakan Matriks Asosiasi. Mengetahui bahwa sinonim atau relasi dari sebuah kata merupakan sesuatu yang bisa dipakai untuk ekspansi *query*, maka Thesaurus digunakan dalam mendapatkan perluasan *query* dengan cara menambah kata yang didapat dari layanan Thesaurus sebagai kandidat ekspansi dan akan dimasukkan kedalam Matriks Asoasiasi. Dalam perangkat lunak pada skripsi ini juga akan dipakai beberapa proses dari NLP untuk melakukan normalisasi kata sehingga menjadi *term* yang dipakai dalam *Information Retrieval*.

1.2 Rumusan masalah

Rumusan masalah pada penelitian ini adalah

1. Bagaimana cara melakukan ekspansi *query* dengan menggunakan Matriks Asosiasi?
2. Bagaimana Thesaurus bisa digunakan pada Matriks Asosiasi untuk melakukan ekspansi *query*?
3. Bagaimana membangun perangkat lunak untuk penelitian ini?

1.3 Tujuan

Penelitian ini dilakukan dengan tujuan sebagai berikut :

1. Mempelajari langkah-langkah menggunakan Matriks Asosiasi untuk melakukan ekspansi *query* dan mengimplementasikannya.
2. Mempelajari Thesaurus dan mempelajari cara mengkombinasikan Thesaurus dengan Matriks Asosiasi untuk melakukan ekspansi *query*.
3. Mempelajari cara membangun perangkat lunak yang baik untuk penelitian ini.

1.4 Batasan Masalah

Terdapat batasan masalah pada penelitian ini antara lain :

1. Perangkat lunak ini hanya bekerja dalam bahasa bahasa Inggris.
2. Jumlah kata yang ada pada *corpus*(koleksi teks) jauh lebih sedikit dari suatu sistem *Information Retrieval* yang sudah ada seperti Google Search. Hal ini dikarenakan keterbatasan memori. Penelitian akan dilakukan dengan jumlah kata pada *corpus* setidaknya berisi 20000 kata.
3. Masalah yang berkaitan dengan semantik atau tentang masalah pemaknaan suatu kata akan diabaikan. Dalam perangkat lunak dilakukan *stemming* dimana suatu kata dasar akan

mewakili beberapa perubahan bentuk katanya. Suatu kata yang mewakili perubahannya memiliki makna yang berbeda dengan perubahan bentuk katanya, tetapi masih tergolong pada satu topik yang sama.

1.5 Metode Penelitian

Metode penelitian pada penelitian ini adalah sebagai berikut :

1. Melakukan studi mengenai *Information Retrieval*.
2. Mempelajari cara menggunakan Matriks Asosiasi untuk melakukan ekspansi *query*.
3. Melakukan analisis semua teori yang dipakai
4. Melakukan perancangan perangkat lunak dan melakukan implementasi dalam bahasa pemrograman Java.
5. Melakukan pengujian perangkat lunak.
6. Melakukan eksperimen.
7. Menarik kesimpulan dari hasil pengujian dan eksperimen.
8. Membuat dokumentasi skripsi.

1.6 Sistematika Pembahasan

Sistematika pembahasan pada penelitian ini adalah sebagai berikut :

1. **Pendahuluan**
   Membahas latar belakang diperlukannya ekspansi *query* dalam *Information Retrieval*, menjelaskan Matriks Asosiasi sebagai cara yang dipakai untuk ekspansi *query* dan membahas tujuan perangkat lunak dibuat. Selain itu dibahas juga rumusan masalah, tujuan penelitian, batasan masalah dan metode penelitian.
2. **Dasar Teori**
   Memuat landasan teori yang dipakai yang digunakan pada penelitian ini yaitu antara lain *Information Retrieval, Query expansion, Natural Language Processing*, dan *Lucene*.
3. **Analisis**
   Berisi analisis bagaimana melakukan proses yang telah dibahas pada bagian teori.
4. **Perancangan**
   Berisi perancangan antarmuka, diagram aktivitas, diagram kelas, dan *pseudocode* untuk perangkat lunak.
5. **Implementasi dan Pengujian**
   Berisi implementasi kode program dan pengujian. Pengujian dibagi menjadi 2 bagian yaitu pengujian fungsional dan eksperimental. Pengujian fungsional bertujuan untuk menguji

perangkat lunak berjalan dengan benar. Dokumen yang dipakai untuk pengujian fungsional adalah dokumen yang dibuat sendiri sedemikian rupa untuk membuktikan semua teori yang dipakai dapat memenuhi tujuan skripsi ini. Pengujian eksperimental dilakukan dengan jumlah dokumen dan *term* yang lebih banyak. Dokumen pada pengujian eksperimental menggunakan dokumen yang sudah ada di internet. Keterangan mengenai sumber dokumen yang dipakai dalam pengujian dapat dilihat pada Lampiran B.

6. **Kesimpulan dan Saran**

   . Berisi kesimpulan setelah melakukan pengujian dan saran-saran untuk mengembangkan perangkat lunak ini.