# PERINGKASAN PROPOSAL SKRIPSI MENGGUNAKAN ALGORITMA *VECTOR SPACE MODEL*

**Latius Hermawan**
Jurusan Teknik Informatika, Fakultas Sains dan Teknologi
Jl. Bangau No. 60, Ilir Timur II, Palembang 30113
Email: tiuz.hermawan@ukmc.ac.id

**Abstrak**

*Peningkatan teknologi informasi telah memicu peningkatan dokumen teks digital secara massif termasuk dokumen, salah satunya proposal skripsi mahasiswa. Suatu artikel yang memiliki ukuran yang panjang, akan mengakibatkan pembaca akan sangat kesulitan bila harus membaca dan menyerap semua informasi dari artikel tersebut. Information Retrieval yang merupakan sub dari Data Mining yang mempelajari peringkasan dokumen akan menjadi dasar penelitian untuk menyelesaikan masalah peringkasan. Peringkasan dokumen dalam hal ini sangat dibutuhkan untuk mengekstraksi informasi dari sebuah dokumen yang dibaca. Waktu yang digunakan untuk membaca proposal bagi para dosen juga dapat mengganggu aktivitas yang mungkin sedang sibuk melakukan pekerjaan lainnya. Peringkasan otomatis akan membantu proses ekstraksi dalam penyusunan kalimat mengenai intisari dari dokumen serta menggabungkan menjadi suatu ringkasan. Peringkasan teks dilakukan dengan cara memberika bobot pada setiap kalimat dengan menggunakan algoritma Vector Space Model. Hasilnya adalah sistem meringkas dokumen dengan menghilangkan kalimat yang tidak memiliki bobot yang besar terhadap kalimat yang lain. Hasil dari kemampuan tingkat ringkasan oleh Vector Space Model dari semua data adalah 31% dari jumlah kata yang ada (jumlah kalimat setelah diringkas / jumlah kalimat sebelum diringkas * 100%) atau memotong 69% kata yang tidak memiliki bobot yang besar terhadap kalimat lain.*

***Kata kunci :*** *peringkasan, proposal skripsi, vector space model, information retrieval*

## 1. PENDAHULUAN

Peningkatan teknologi informasi telah memicu peningkatan dokumen teks digital secara massif termasuk dokumen berbahasa Indonesia. Penggalian informasi dari dokumen berupa ringkasan secara otomatis sangat dibutuhkan (Ridock, 2014). Suatu artikel yang memiliki ukuran yang panjang, akan mengakibatkan pembaca akan sangat kesulitan bila harus membaca dan menyerap semua informasi dari artikel tersebut . *Infromation retrieval* merupakan salah satu ilmu dalam bidang informatika yang mempelajari pengolahan data dokumen teks (Han, 2006).

Peringkasan dokumen dalam hal ini sangat dibutuhkan untuk mengekstraksi informasi dari sebuah dokumen yang dibaca. Information Retrieval yang merupakan sub dari Data Mining yang mempelajari peringkasan dokumen akan menjadi dasar penelitian untuk menyelesaikan masalah peringkasan. Peringkasan dokumen dalam penelitian ini adalah penulisan kembali sebuah dokumen dalam format yang lebih pendek dan merepresentasikan dokumen asli tanpa kehilangan informasi penting yang tersedia dalam dokumen asli (Binwahlan, 2011). Waktu yang digunakan untuk membaca proposal bagi para dosen juga dapat mengganggu aktivitas yang mungkin sedang sibuk melakukan pekerjaan lainnya. Salah satu pendekatan yang digunakan untuk menangani permasalahan ini adalah dengan menggunakan peringkasan kalimat secara otomatis yang dilakukan dengan bantuan komputer. Peringkasan otomatis akan membantu proses ekstraksi dalam penyusunan kalimat mengenai intisari dari kalimat serta menggabungkan menjadi suatu ringkasan (Jazek, 2008). Peringkasan teks dilakukan dengan cara memberika bobot pada setiap kalimat dengan menggunakan algoritma *Vector Space Model.*

Salah satu metode yang umumnya digunakan dalam bidang pencarian informasi adalah metode Vector Space Model (VSM). Inti dari metode VSM adalah dasar dari tiap dokumen atau query diwakilkan oleh kata-kata yang terdapat di dalamnya (pengindeksan). Vektor yang terdiri dari kata-kata tersebut dapat didefinisikan untuk menggambarkan setiap bagian dari dokumen dan query, maka dokumen tersebut dapat ditentukan berhubungan dengan permintaan atau tidak berdasarkan hasil perhitungan korelasi antara mereka. Dokumen yang memiliki relativitas yang lebih besar dengan pencarian tertentu dianggap lebih terkait (Hongdan, 2011).

ISBN 978-602-99334-9-9

Diharapkan dengan penerapan metode ini mampu memberikan hasil peringkasan yang mampu membantu para dosen untuk mendapatkan informasi inti dari proposal skripsi yang akan diujikan kepada para mahasiswanya.

## 2. METODOLOGI

### 2.1. Algoritma Vector Space Model

*Vector Space Model* adalah suatu model aljabar untuk mewakili dokumen teks sebagai suatu vektor pengenal, contohnya indeks kata. VSM biasanya digunakan dalam penyaringan informasi, temu balik informasi, pengindeksan, dan perankingan relevansi (Hongdan, 2011). Pemikiran dasar dari metode VSM ini adalah merepresentasikan setiap kata independen dan setiap dokumen dinyatakan dalam sebuah vektor sehingga kompleksitas hubungan kata-kata menjadi sederhana dan dapat dihitung. Dalam VSM, setiap dokumen terdiri dari *term* (T1, T2, ..., Tn) dan setiap *term* Ti memiliki bobot Wi. *Term* (T1, T2, ..., Tn) dianggap sebagai salah satu elemen vektor dalam sistem koordinat N-dimensi (Guo, 2008). TF-IDF merupakan sebuah skema pembobotan yang sering digunakan dalam VSM bersama dengan *cosine similarity* untuk menentukan kesamaan antara dua buah dokumen. TF-IDF mempertimbangkan frekuensi katakata yang berbeda dalam semua dokumen dan mampu membedakan dokumen. Dalam VSM, setiap vector disusun oleh *term* dan bobot yang mewakili dokumen. Kesamaan dokumen dapat dinyatakan dengan sudut atau jarak antara vektor, semakin kecil sudut atau jarak berarti semakin mirip dua dokumen tersebut. TF merupakan *Term Frequency* dan IDF adalah *Inverse Document Frequency*. Rumusnya adalah sebagai seperti disajikan dalam persamaan (1) dan (2) (Chuang, 1997).

$$W_{t,d} = TF_{t,d} * IDF_t \tag{1}$$

Keterangan :

$W_{t,d}$ = bobot dari *t (term)* dalam satu dokumen

$TF_{t,d}$ = frekuensi kemunculan *t (term)* dalam dokumen *d*

$IDF_t$ = *Inverse document frequency*, dimana

$$IDF_t = \log\left(\frac{N}{n_t}\right) \tag{2}$$

Keterangan :

$N$ = jumlah semua dokumen

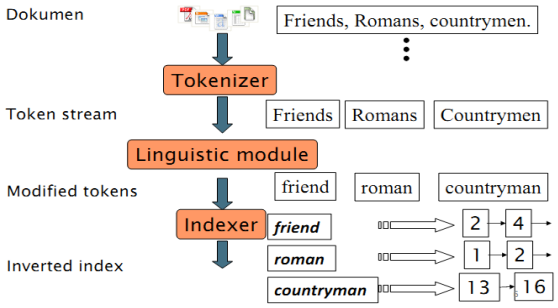
$n_t$ = jumlah dokumen yang mengandung *term t*

IDF mencerminkan penyebaran *term t* dalam keseluruhan dokumen sehingga dapat memperlihatkan perbedaan *term t* dalam tiap dokumen. TF mencerminkan penyebaran *term t* dalam sebuah dokumen. TF-IDF dapat membuat pengecualian bagi kata-kata yang berfrekuensi tinggi tetapi sedikit memiliki persamaan, sehingga TFIDF merupakan algoritma yang efektif untuk perhitungan bobot *term t*. Setelah pembobotan tiap *term* dilakukan, diperlukan perhitungan untuk melakukan perankingan untuk mengukur kemiripan antara vektor *query* dan vector dokumen yang akan dibandingkan. Salah satu metode yang biasa digunakan dalam perhitungan kemiripan adalah pengukuran *cosine*, yang menentukan sudut antara vektor dokumen dan vektor *query* dan didefinisikan dalam persamaan (3).

$$Similarity\left(\vec{d_j}, \vec{q}\right) = \frac{\vec{d_j} \cdot \vec{q}}{|\vec{d_j}| . |\vec{q}|} = \frac{\sum_{i=1}^{t}(w_{ij} . w_{iq})}{\sqrt{\sum_{i=1}^{t} w^2{}_{ij} \cdot \sum_{i=1}^{t} w^2{}_{iq}}} \tag{3}$$

dimana $w_{q,\,t}$ adalah bobot dari *term t*, penyebut dalam persamaan ini disebut faktor normalisasi yang berfungsi untuk menghilangkan pengaruh panjang dokumen (Chuang, 1997). Normalisasi ini diperlukan karena dimana dokumen panjang akan cenderung memiliki nilai lebih besar karena memiliki frekuensi kemunculan kata yang besar pula. Proses perankingan dari dokumen dapat dianggap sebagai proses pemilihan (vektor) dokumen yang dekat dengan (vektor) *query*, kedekatan ini diindikasikan dengan sudut yang dibentuk.

## 2.2. Tahapan Peringkasan




**Gambar 1. Tahapan Peringkasan Dokumen**

*Text preprocessing* adalah tahapan untuk mempersiapkan teks menjadi data yang akan diolah di tahapan berikutnya. Inputan awal pada proses ini adalah berupa dokumen. *Text preprocessing* pada penelitian ini terdiri dari beberapa tahapan, yaitu: proses pemecahan kalimat, proses *tokenizing* kata, proses *filtering*, dan proses *stemming.*

**a. *Preprocessing***

Memecah dokumen menjadi kalimat-kalimat merupakan langkah awal tahapan *text preprocessing*. Pemecahan kalimat yaitu proses memecah string teks dokumen yang panjang menjadi kumpulan kalimat-kalimat. Dalam memecah dokumen menjadi kalimat-kalimat menggunakan fungsi split(), dengan tanda titik ".", tanda tanya "?" dan tanda tanya "!" sebagai delimiter untuk memotong *string* dokumen.

**b. *Tokenizing***

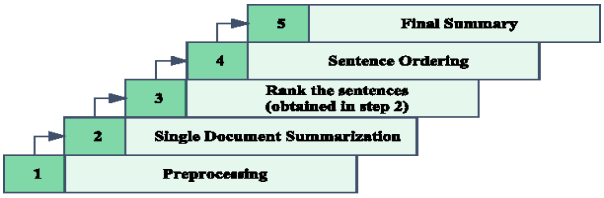
Proses *tokenizing* adalah proses pemotongan *string* masukan berdasarkan tiap kata yang menyusunnya. Pada prinsipnya proses ini adalah memisahkan setiap kata yang menyusun suatu dokumen. Pada umumnnya setiap kata terindentifikasi atau terpisahkan dengan kata yang lain oleh karakter spasi, sehingga proses *tokenizing* mengandalkan karakter spasi pada dokumen untuk melakukan pemisahan kata.

**c. *Filtering***

*Filtering* merupakan proses penghilangan *stopword*. *Stopword* adalah kata-kata yang sering kali muncul dalam dokumen namun artinya tidak deskriptif dan tidak memiliki keterkaitan dengan tema tertentu.

**d. *Stemming***

*Stemming* merupakan proses mencari akar (*root*) kata dari tiap token kata yaitu dengan pengembalian suatu kata berimbuhan ke bentuk dasarnya (*stem*). Pada penelitian ini menggunakan *stemming* untuk bahasa indonesia (Tala, 2003). Peringkasan dokumen pada Gambar 2.

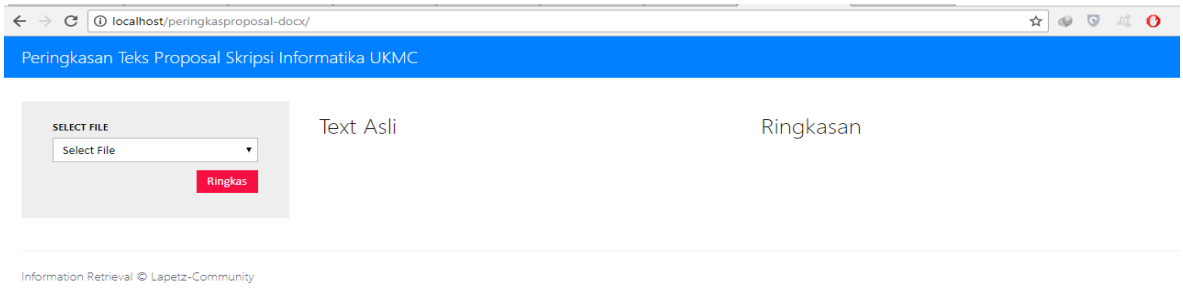



**Gambar 2. Peringkasan Dokumen**

## 2.3. Tahapan Perancangan

*User Interface* pada sistem ini hanya terdapat satu halaman tunggal yang isinya langsung ke bagian peringkasan dokumen. Tujuan halaman yang dibuat tunggal agar memudahkan pengerjaan sistem dan juga memudahkan pengguna. Halaman dirancang dengan HTML dan CSS menggunakan bahasa pemrograman PHP. Halaman peringkasan tertampil pada Gambar 3.

ISBN 978-602-99334-9-9



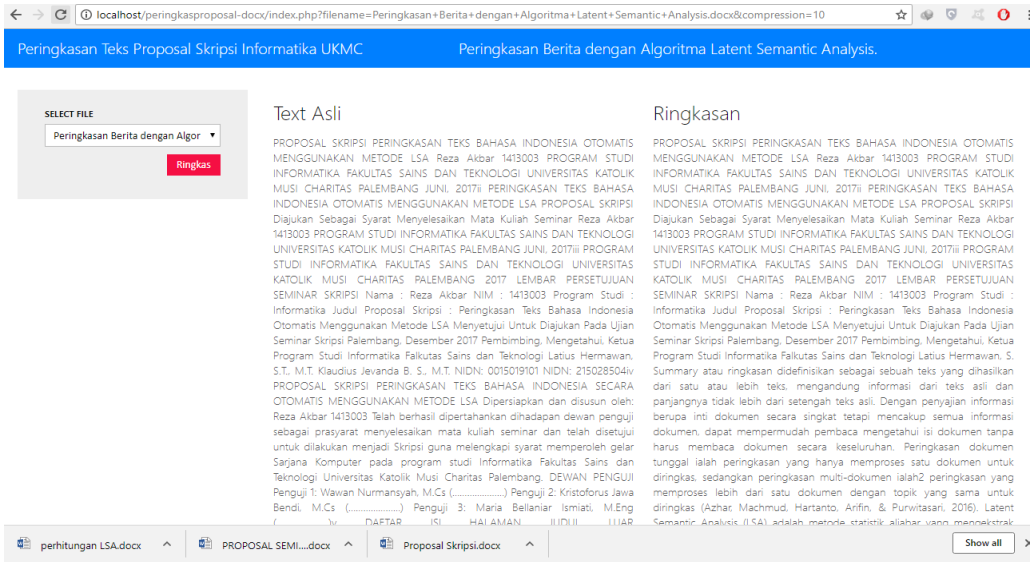

**Gambar 3. Tampilan Halaman Peringkasan Dokumen**

## 3. HASIL DAN PEMBAHASAN

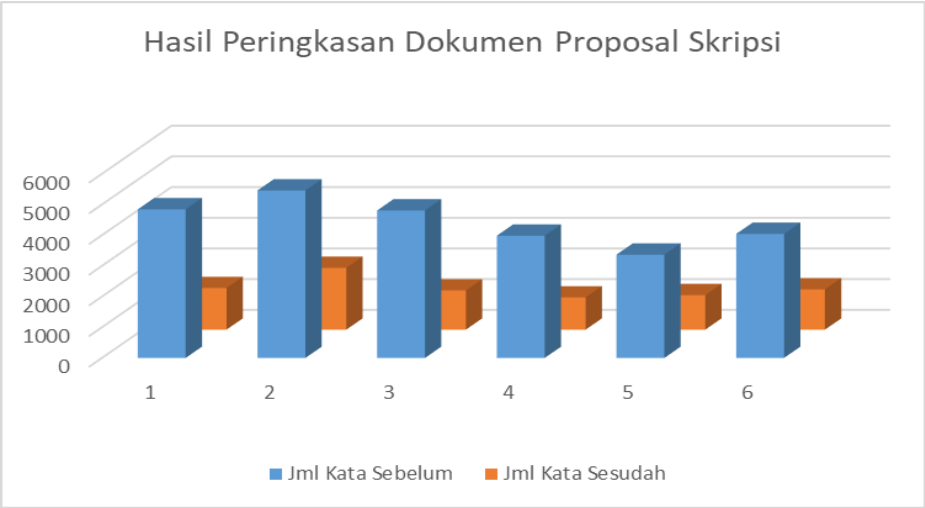
Pengujian dilakukan dengan meringkas dokumen proposal mahasiswa informatika yang sudah dipilih sebagai percobaan dari peringkasan dokumen teks. Dimana proposal skripsi yang ada di*load* ke aplikasi untuk menjadi daftar dokumen yang akan menjadi data percobaan. Jumlah data yang digunakan terdapat 5 dokumen teks *(*.doc / *.docx)* yang akan digunakan sebagai percobaan. Berikut tampilan hasil dari percobaan seperti disajikan dalam tabel 1.

**Tabel 1. Percobaan Peringkasan Dokumen Proposal Skripsi**

| No. | Judul | Jml Kata Sebelum | Jml Kata Sesudah |
|---|---|---|---|
| 1 | Pencarian ATM Terdekat Menggunakan Algoritma Flloyd Warshall.doc | *4843* | *1356 28%* |
| 2 | Peringkasan Berita Menggunakan Algoritma Latent Semantic Analysis.docx | 5457 | 2011 37% |
| 3 | Penerapan Algoritma Collision Detection Pada Game RTS (*Real Time Strategy*) Untuk Mengatur Pergerakan NPC.docx | 4808 | 1278 27% |
| 4 | Implementasi Aloritma *Elgamal* Untuk *Enkripsi* Dan *Dekripsi* Pada File *Word* Berbasis *Web.doc* | 3983 | 1049 26% |
| 5 | Aplikasi Pengumpulan Tugas Kuliah Bagi Mahasiswa Informatika Dan Sistem Informasi .docx | 3367 | 1125 33% |
| 6 | Analisis Laporan Skripsi Dengan Metoda *Systematic Literature Review.doc* | 4042 | 1312 32% |




**Gambar 4. Website Aplikasi Peringkasan Dokumen**




**Gambar 5. Hasil dari Percobaan Peringkasan Dokumen**

## 4. KESIMPULAN

Dari hasil tabel 1 dapat dilihat bahwa sistem meringkas dokumen dengan menghilangkan kalimat yang tidak memiliki bobot yang besar terhadap kalimat yang lain, sehingga bukan bagian dari intisari dari informasi aslinya. Hasil dari kemampuan tingkat ringkasan oleh *Vector Space Model* dari semua data adalah 31% dari jumlah kata yang ada (jumlah kalimat setelah diringkas / jumlah kalimat sebelum diringkas * 100%) atau memotong 69% kata yang tidak memiliki bobot yang besar terhadap kalimat lain. Peneliti berharap bahwa penelitian selanjutnya dapat ditingkatkan dengan adanya fasilitas untuk menentukan sendiri tingkat kompresinya sesuai dengan keinginan pembaca ataupun dengan menambahkan metode lain untuk memperbaiki hasil dari ringkasan dokumen ini.

## DAFTAR PUSTAKA

Binwahlan, M. S. (2011). *Fuzzy Swarm Diversity Based Text Summarization*. Johor Bahru: Universiti Teknologi Malaysia.

Fadillah Z Tala.(2003). *A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia. Institute or Logic, Language and Computation*, Universiteit van Amsterdam, The Netherlands.

Guo, Q. (2008). *The Similarity Computing of Document based on VSM*. IEEE.

Han, J., Kamber, M., dan Pei, J.(2006). *Data Mining: Concepts and Techniques*. Morgan Kaufmann.

Hongdan, et al. (2011). *A Document-Based Information Retrieval Model Vector Space*. IEEE. 65-68.

Jazek K. (2008). *Automatic Text Summarization (The State of The Art 2007 and new challenges).* Znalosti 2008, page 1-12

L. L. D., Chuang, H., Seamons, K. (1997). *Document Ranking and the Vector-Space Model.* IEEE. 67-79.

Ridock A. (2014). Peringkasan Dokumen Bahasa Indonesia Berbasis *Non-Negative Matrix Factorization (NMF).* Jurnal Teknologi Informasi dan Ilmu Komputer *(JTIIK) Vol. 1, No. 1, April 2014, hlm. 39-4*

ISBN 978-602-99334-9-9