

# Pemanfaatan Vector Space Model Untuk Peringkasan Berita

Rina Anggeriani, Fitria, Muhammad Hafid

**Abstraksi**— *Automatic summarization* adalah proses mengurangi dokumen teks dengan sebuah program komputer untuk membuat ringkasan yang mempertahankan bagian yang paling penting dari dokumen asli. Untuk mengatasi informasi yang berlebihan, dan jumlah data yang meningkat, sehingga perlu peringkasan otomatis. Konsep sederhana ringkasan adalah mengambil bagian penting dari keseluruhan isi dari artikel. Yang kemudian menyajikannya kembali dalam bentuk ringkasan. Langkah yang diambil pada penelitian ini nantinya berawal dari user (admin) memilih atau mencari teks dokumen yang akan diringkas dengan kata kunci pada judul berita sebagai *query*.

Sistem melakukan penyiapan teks (*text preprocessing*) dokumen yang terdiri dari tahap peringkasan mulai dari pemecahan kalimat, case folding, tokenisasi, stop word removal, stemming, pembobotan *tf/idf*, pembobotan *query* dengan kalimat menggunakan *vector space model* yang nantinya diperoleh hasil ringkasan. Hasil ringkasan diperoleh dari nilai *cosine >0* dan diurutkan dari nilai maksimal ke nilai minimal. Hasil pengujian dibandingkan dengan ringkasan manual yang menghasilkan rata-rata *precision 64%*, *recall 67%*, dan *f-measure 64,73%*.

**Kata Kunci**— metode *vector space model*, ringkasan berita.

## I. PENDAHULUAN

Ketersediaan informasi yang semakin berkembang pesat menandakan bahwa akses untuk memperoleh ringkasan yang logis menjadi hal yang vital. Dengan adanya ringkasan, diharapkan pembaca dapat dengan cepat dan mudah memahami makna sebuah teks tanpa harus membaca keseluruhan teks. Hal ini dapat menghemat waktu pembaca karena dapat menghindari pembacaan teks yang tidak relevan dengan informasi yang diharapkan oleh pembaca, terutama ketika sangat banyak informasi tersedia di internet.

Terdapat dua pendekatan pada peringkasan teks, yaitu ekstraksi (*shallower approaches*) dan abstraksi (*deeper approaches*). Pada umumnya, abstraksi dapat meringkas teks lebih kuat daripada ekstraksi, tetapi sistemnya lebih sulit dikembangkan karena mengaplikasikan teknologi *natural language generation* yang merupakan bahasan yang dikembangkan tersendiri.

Berdasarkan hal tersebut muncul ide untuk membuat sistem peringkasan teks otomatis berita bahasa Indonesia sehingga bisa memberikan pelayanan yang lebih baik kepada pembaca untuk mempermudah dalam memahami inti dari suatu berita. Kajian terhadap metode yang digunakan dalam peringkasan dokumen berdasarkan representasi kebutuhan informasi berupa kata kunci, yaitu judul berita sebagai bentuk *query* yang diterapkan

dengan pemanfaatan *Vector Space Model (VSM)* sebagai sistem pembobotan *query relevance* dan untuk *similarity* kalimat.

Langkah yang diambil pada penelitian ini nantinya berawal dari user (admin) memilih atau mencari teks dokumen berita yang akan diringkas dengan kata kunci pada judul sebagai *query*. Sistem melakukan penyiapan teks (*text preprocessing*) dokumen yang terdiri dari tahap pemecahan kalimat, case folding, tokenizing, filtering dan stemming. Hasil proses *text preprocessing* dilakukan pembobotan *tf-idf*, kemudian melakukan pembobotan *query relevance* dan *similarity* kalimat untuk memperoleh ringkasan.

## II. TINJAUAN PUSTAKA

Peringkasan teks adalah proses mengurangi dokumen teks dengan sebuah program komputer untuk membuat ringkasan yang mempertahankan poin yang paling penting dari dokumen asli. Untuk mengatasi informasi yang berlebihan, dan jumlah data yang meningkat, sehingga perlu dalam peringkasan otomatis. Otomatisasi ringkasan dapat dikenakan terhadap satu dokumen (*single document summarization*) atau beberapa dokumen (*multi document summarization*), satu bahasa (*monolingual*) atau beberapa bahasa (*translingual/multilingual*). Contoh penggunaan teknologi *summarization* adalah mesin pencari seperti Google.

Secara umum, ada dua pendekatan untuk peringkasan otomatis: *extractive summary* dan *abstractive summary*. *Extractive summary* bekerja dengan memilih kalimat atau kata yang ada dalam ringkasan diambil secara utuh dari dokumen aslinya. Cara *abstractive* umumnya membutuhkan pembangkit kalimat (*language generator*) yang lebih kompleks dan membutuhkan waktu lebih lama, sehingga dapat ditebak bahwa pendekatan pertama yaitu *extractive* lebih mudah bagi komputer [1]. Terdapat 3 metode dalam peringkasan sistem yaitu *Extraction-based summarization*, *Abstraction-based summarization*, *Maximum entropy-based summarization*.

Tahapan peringkasan mulai dari pemecahan kalimat, case folding, tokenisasi, Stop Word Removal, stemming, pembobotan *tf/idf*, pembobotan *query* dengan kalimat menggunakan *vector space model* yang nantinya diperoleh hasil ringkasan.

### A. *Vector Space Model (Model Ruang Vektor)*

Model ruang vektor adalah salah satu model yang digunakan untuk mengukur derajat kemiripan antara suatu dokumen dengan suatu *query*. Dalam model ruang vektor, koleksi

dokumen dipresentasikan oleh matriks term document atau matriks term frequency. Sebagaimana digambarkan pada gambar 2.1 setiap sel dalam matriks bersesuaian dengan bobot yang diberikan dari suatu term dalam dokumen. yang ditentukan. Nilai nol berarti bahwa term tersebut tidak hadir di dalam dokumen. Keberhasilan dari model VSM ini ditentukan oleh skema pembobotan terhadap suatu term baik untuk cakupan lokal maupun global dan faktor normalisasi pembobotan local hanya berpedoman pada frekuensi munculnya term dalam suatu dokumen dan tidak melihat frekuensi kemunculan term tersebut di dalam dokumen lainnya.

$$\begin{bmatrix} & T_1 & T_2 & \dots & T_t \\ D_1 & W_{11} & W_{21} & \dots & W_{t1} \\ D_2 & W_{12} & W_{22} & \dots & W_{t2} \\ \dots & \dots & \dots & \dots & \dots \\ D_n & W_{1n} & W_{2n} & \dots & W_{tn} \end{bmatrix}$$

Gambar 2.1 Matriks Term Document

Mencari nilai cosinus sudut antara dua vector dari setiap bobot dokumen (WD) dan bobot dari kata kunci Hubungan antara kata.

$$\text{Cosine} \rightarrow \text{sim}(d_j, q) = \frac{\vec{d}_j \cdot \vec{q}}{|\vec{d}_j| \cdot |\vec{q}|} = \frac{\sum_{i=1}^t (W_{ij} \cdot W_{iq})}{\sqrt{\sum_{i=1}^t W_{ij}^2 \cdot \sum_{i=1}^t W_{iq}^2}}$$

Keterangan :

t = Term dalam Kalimat

D = Dokumen

Q = Kata Kunci

$W_{ij}$  = bobot term i dalam blok ij

$W_{iq}$  = bobot term i dalam blok iq

Setelah mendapatkan cosine tiap-tiap dokumen, maka hasil bobot dari kata kunci diurutkan. Bobot yang besar menjadi prioritas sebagai dokumen yang memiliki hubungan dengan kata kunci.

$$\text{Cosine}(D_i) = \text{sum}(kk^2, D_i) / [\text{sqrt}(kk^2) * \text{sqrt}(D_i^2)]$$

### B. Term Frequency dan Inverse Document Frequency

Pembobotan dapat diperoleh berdasarkan jumlah kemunculan suatu term dalam sebuah dokumen term frequency (tf) dan jumlah kemunculan term dalam koleksi dokumen inverse document frequency (idf). Nilai idf sebuah term(kata) dapat dihitung menggunakan persamaan sebagai berikut:

$$IDF = \text{Log} \frac{d}{df^i}$$

D adalah jumlah dokumen yang berisi term (t) dan dfi adalah jumlah kemunculan (frekuensi) term terhadap D. Adapun algoritma yang digunakan untuk menghitung bobot (W) masing-masing dokumen terhadap kata kunci (query), yaitu:

$$W_{d,t} = tf_{d,t} * IDF_t$$

Keterangan :

d = dokumen ke- d

t = term ke- t dari kata kunci

tf = term frekuensi/frekuensi kata

W = bobot dokumen ke- d terhadap term ke- t

Setelah bobot (W) masing-masing dokumen diketahui, maka dilakukan proses pengurutan (sorting) dimana semakin besar nilai W, semakin besar tingkat kesamaan (similarity) dokumen tersebut terhadap kata yang dicari, demikian pula sebaliknya.

## III. ANALISA DAN DESAIN SISTEM

### A. Penerapan Metode

Vector space model adalah salah satu model yang digunakan untuk mengukur kemiripan antara dokumen dengan query. Pada model ini, query dan dokumen dianggap sebagai vektor-vektor pada ruang n-dimensi, dimana n adalah jumlah dari seluruh term yang ada dalam leksikon. Leksikon adalah daftar semua term yang ada dalam indeks. Salah satu cara untuk mengatasi hal tersebut dalam model ruang vektor adalah dengan melakukan perluasan vektor.

Dalam pembobotan query relevance dengan menggunakan vector space model, dokumen dan query direpresentasikan dalam bentuk vector. Untuk setiap kata diberi bobot menggunakan skema tf atau tf-idf.

Perhitungan bobot query relevance merupakan bobot hasil perbandingan kemiripan (similaritas) antara query yang dimasukkan oleh user terhadap keseluruhan kalimat. Perhitungan bobot query relevance yaitu menghitung bobot kemiripan antara query dengan kalimat dalam dokumen, dengan menghitung cosinus sudut dari dua vector yaitu W, bobot dari tiap kalimat dan W (bobot) query.

Pada sekumpulan data berita sebanyak 150 data berita pada database akan diambil 1 berita yang akan digunakan sebagai contoh untuk perhitungan query relevance menggunakan vector space model. Query yang diberikan pada penelitian ini berupa kata kunci dari judul berita. Contoh dokumen dapat dilihat pada Tabel I.

Tahapan-tahapan praproses tersebut sama dengan tahapan praproses data untuk peringkasan berita. Hasil tokenisasi, penghapusan stopwords dan stemming dokumen, untuk membentuk sebuah matrik antara term dengan masing-masing dokumen, ditunjukkan pada tabel II.

Selanjutnya melakukan Perhitungan nilai W dengan menggunakan tf-idf. Kata kunci pada data berita yang digunakan untuk contoh perhitungan yaitu “dana, silpa, deficit, dan habis” diambil dari kode 9,12,15,dan 16. Perhitungan dilakukan untuk semua kolom yaitu kk, D1 s.d D19. Diperoleh nilai bobot W untuk kata kunci dan masing-masing dokumen sebagai berikut:

KK	= 1,2 + 1,2 + 1,3 + 1,3	= 5
D1	= 0 + 1,20 + 1,26 + 0	= 2,5
D2	= 0 + 0 + 0 + 0	= 0
D3	= 0 + 1,20 + 0 + 0	= 1,20
D4	= 1,22 + 1,20 + 0 + 0	= 3,7
D5	= 0 + 1,20 + 1,26 + 0	= 2,5
D6	= 0 + 1,20 + 1,26 + 0	= 2,5
D7	= 1,22 + 0 + 0 + 1,26	= 2,48
D8	= 0 + 0 + 0 + 0	= 0
D9	= 0 + 0 + 0 + 0	= 0
D10	= 0 + 0 + 0 + 0	= 0
D11	= 1,22 + 0 + 0 + 0	= 1,22
D12	= 0 + 0 + 0 + 0	= 0
D13	= 1,22 + 0 + 0 + 1,26	= 2,48
D14	= 0 + 0 + 0 + 0	= 0
D15	= 0 + 0 + 0 + 0	= 0
D16	= 0 + 0 + 0 + 0	= 0
D17	= 0 + 0 + 0 + 0	= 0
D18	= 0 + 0 + 0 + 0	= 0

$$D19 = 0 + 0 + 0 + 1,26 = 1,26$$

TABEL I  
 TABEL PEMECAHAN KALIMAT

Dokumen	Pemecahan Kalimat
1	Pemerintah Kota Tarakan memang tidak jadi mengutang untuk menutupi kewajiban yang belum dibayar di tahun anggaran 2015, tetapi dana Sisa Lebih Penggunaan Anggaran (Silpa) yang dua tahun terakhir berhasil menutupi defisit, kini sudah habis
2	Badan Pemeriksa Keuangan (BPK) altara memaparkan, dalam dua tahun belakangan Pemkot Tarakan sebenarnya mengalami defisit anggaran.
3	Namun berhasil ditutupi dengan Silpa.
4	Menurut data yang dipaparkan Kepala Kantor BPK perwakilan Kalimantan Utara (Kaltara), Ade Iwan Rusmana, pada tahun 2012 Pemkot Tarakan memiliki Rp 859.30 miliar, namun karena defisit tahun 2013 sebesar Rp 323 miliar, sehingga dana silpa tersisa Rp 531,15 miliar.
5	Kemudian berkurang lagi akibat defisit tahun 2014 sebesar Rp 325.89 sehingga silpa di akhir tahun 2014 tersisa Rp 205.27 miliar.
6	Angka silpa tersebut yang kemudian dipertanyakan mengapa tidak digunakan.
7	Sekretaris Kota (sekkot) Tarakan dr. Khairul yang juga menjabat ketua tim penyusun anggaran daerah (TPAD) menjelaskan, dana Rp 205.27 miliar telah masuk di struktur anggaran 2015 dan telah habis digunakan.
8	“Itukan sudah masuk di tahun anggaran 2015.
9	Sudah kami pakai.
10	Ini setelah kami hitung sampai 31 Desember 2015 pemkot masih memiliki kewajiban yang harus dibayar sebesar Rp 95 miliar.
11	Walaupun sekarang tidak punya uang cash, tapi kami masih punya uang dari dana bagi hasil (DBH) yang belum ditransfer dari pemerintah pusat dan provinsi,” jelas mantan Kepala Dinas Kesehatan itu.
12	Nilai DBH tersebut terdiri dari Rp 44 miliar dari pemerintah pusat dan Rp 53 miliar pemprov, dengan total Rp 97 miliar.
13	Diakui Khairul, dana yang dimiliki pemkot sempat mencapai hampir Rp 1 triliun di tahun 2012 lalu, sekarang sudah habis dan hanya tersisa Rp 2 miliar jika DBH yang sejumlah Rp 97 miliar tersebut telah ditransfer dan dikurangi Rp 95 miliar kekurangan kewajiban bayar pemkot tahun 2015.
14	“Sudah dipakai, sisa Rp 2 miliar.
15	Tapi itu justru bagus.
16	Instruksi dari pemerintah pusat untuk jangan menyimpan uang.
17	Gunakan untuk pembangunan dan program-program prioritas.
18	Kalau simpanan kita tinggi (dari silpa,Red.) maka dipertanyakan bagaimana dengan pembangunan kita.
19	Kalau seluruh anggaran sudah digunakan dan habis, berarti serapannya bagus,” jelasnya.
Kata kunci	Defisit Dua Tahun , Dana Silpa Habis

TABLE II  
 TABEL MATRIKS TERM DAN DOKUMEN

No	Term	kk	D1	D2	D3	D4	..	..	D19
1	perin	0	1	0	0	0	..	..	0
2	kota	0	1	0	0	0	..	..	0
3	tara	0	1	1	0	1	..	..	0
4	kutang	0	1	0	0	0	..	..	0
5	tutup	0	2	0	0	0	..	..	0
6	wajib	0	1	0	0	0	..	..	0
7	bayar	0	1	0	0	0	..	..	0
8	anggar	0	2	1	0	0	..	..	1
9	dana	1	0	0	0	1	..	..	0
10	sisa	0	1	0	0	1	..	..	0
11	guna	0	1	0	0	0	..	..	0
12	silpa	1	1	0	1	1	..	..	0
13	akhir	0	1	0	0	0	..	..	0
14	hasil	0	1	0	2	0	..	..	0
15	defisit	1	1	0	0	1	..	..	0
16	habis	1	0	0	0	0	..	..	..
..	..	..	..	..	..	..	..	..	..
82	sipared	0	0	0	0	0	..	..	0
83	rap	0	0	0	0	0	..	..	1

Dari nilai bobot W yang diperoleh ditemukan nilai bobot W yang sama antara dokumen D1 dan kata kunci D5. Dikarenakan nilai bobot W dari dokumen ada yang sama, maka tidak dapat dilakukan sorting dokumen. Untuk mengatasi permasalahan tersebut, digunakan vector space model untuk menghitung query relevance antara query dengan dokumen. Langkah berikutnya adalah menghitung akar (Sqrt) dari jumlah  $KK^2$  dan semua dokumen yang telah dikuadratkan.

Sqrt ( $KK^2$ ) = 6,1	= 2,74
Sqrt ( $D1^2$ ) = 32,58	= 5,71
Sqrt ( $D2^2$ ) = 15,58	= 4,07
Sqrt ( $D3^2$ ) = 7,74	= 2,8
Sqrt ( $D4^2$ ) = 44,13	= 6,6
Sqrt ( $D5^2$ ) = 9,31	= 3,1
Sqrt ( $D6^2$ ) = 1,44	= 1,22
Sqrt ( $D7^2$ ) = 36,98	= 6,08
Sqrt ( $D8^2$ ) = 4,80	= 2,19
Sqrt ( $D9^2$ ) = 1,68	= 1,29
Sqrt ( $D10^2$ ) = 13,36	= 3,65
Sqrt ( $D11^2$ ) = 20,49	= 4,52
Sqrt ( $D12^2$ ) = 51,05	= 7,14
Sqrt ( $D13^2$ ) = 44,83	= 6,69
Sqrt ( $D14^2$ ) = 4,45	= 2,10
Sqrt ( $D15^2$ ) = 1,68	= 1,29
Sqrt ( $D16^2$ ) = 7,11	= 2,66
Sqrt ( $D17^2$ ) = 6,94	= 2,63
Sqrt ( $D18^2$ ) = 7,21	= 2,68
Sqrt ( $D19^2$ ) = 8,38	= 2,89

Setelah menghitung dan mendapatkan nilai Sqrt dari  $KK^2$  dan semua dokumen yang telah dikuadratkan, langkah selanjutnya menghitung nilai cosinus antara vector query dengan tiap dokumen. Adapun proses perhitungannya dilakukan dengan menggunakan persamaan 2.2 pada bab 2.

Berikut adalah perhitungan dari nilai query relevance terhadap masing-masing dokumen.

$$\begin{aligned} \text{Cosine}(D1) &= \frac{\text{sum}(\text{kk}^2.D1)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D1^2)]} \\ &= 3,0133/[2,47*5,71] \\ &= 6,97 \\ \text{Cosine}(D2) &= \frac{\text{sum}(\text{kk}^2.D2)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D2^2)]} \\ &= 0/[2,47*4,07] \\ &= 0 \\ \text{Cosine}(D3) &= \frac{\text{sum}(\text{kk}^2.D3)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D3^2)]} \\ &= 1,4366/[2,47*2,8] \\ &= 1,62 \\ \text{Cosine}(D4) &= \frac{\text{sum}(\text{kk}^2.D4)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D4^2)]} \\ &= 4,5126/[2,47*6,6] \\ &= 12,057 \\ \text{Cosine}(D5) &= \frac{\text{sum}(\text{kk}^2.D5)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D5^2)]} \\ &= 3,0133/[2,47*3,1] \\ &= 3,871 \\ \text{Cosine}(D6) &= \frac{\text{sum}(\text{kk}^2.D6)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D6^2)]} \\ &= 1,4366/[2,47*1,2] \\ &= 0,69 \\ \text{Cosine}(D7) &= \frac{\text{sum}(\text{kk}^2.D7)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 3,076/[2,47*6,1] \\ &= 7,59 \\ \text{Cosine}(D8) &= \frac{\text{sum}(\text{kk}^2.D8)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 0/[2,47*2,2] \\ &= 0 \\ \text{Cosine}(D9) &= \frac{\text{sum}(\text{kk}^2.D8)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 0/[2,47*1,3] \\ &= 0 \\ \text{Cosine}(D10) &= \frac{\text{sum}(\text{kk}^2.D8)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 0/[2,47*3,7] \\ &= 0 \\ \text{Cosine}(D11) &= \frac{\text{sum}(\text{kk}^2.D8)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 1,4993/[2,47*4,5] \\ &= 2,73 \\ \text{Cosine}(D12) &= \frac{\text{sum}(\text{kk}^2.D8)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 0/[2,47*7,1] \\ &= 0 \\ \text{Cosine}(D13) &= \frac{\text{sum}(\text{kk}^2.D8)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 3,076/[2,47*6,7] \\ &= 8,34 \\ \text{Cosine}(D14) &= \frac{\text{sum}(\text{kk}^2.D8)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 0/[2,47*2,1] \\ &= 0 \\ \text{Cosine}(D15) &= \frac{\text{sum}(\text{kk}^2.D8)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 1,2962/[2,47*1,3] \\ &= 0,68 \\ \text{Cosine}(D16) &= \frac{\text{sum}(\text{kk}^2.D8)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 0/[2,47*2,7] \\ &= 0 \\ \text{Cosine}(D17) &= \frac{\text{sum}(\text{kk}^2.D8)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 0/[2,47*2,6] \\ &= 0 \\ \text{Cosine}(D18) &= \frac{\text{sum}(\text{kk}^2.D8)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 0/[2,47*2,7] \\ &= 0 \\ \text{Cosine}(D19) &= \frac{\text{sum}(\text{kk}^2.D8)}{[\text{sqrt}(\text{kk}^2)*\text{sqrt}(D7^2)]} \\ &= 1,5767/[2,47*2,8] \\ &= 1,78 \end{aligned}$$

Sesuai hasil perhitungan diatas maka nilai cosinus setiap dokumen telah didapat. Nilai tersebut diranking dari nilai cosine >0 dan diurutkan dari nilai maksimal kenilai minimal.

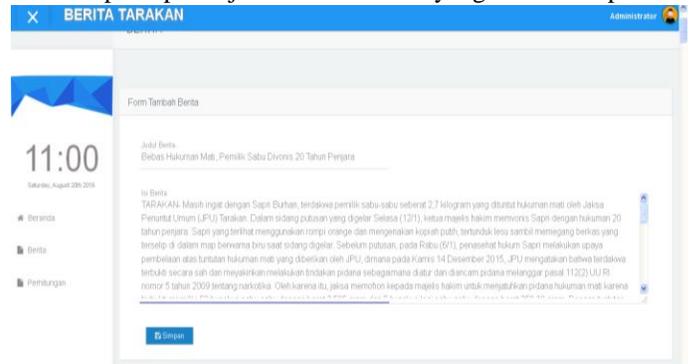
**B. Uji Coba**

Tampilan berita yang telah diringkas oleh admin, tampil di beranda admin terlihat pada gambar 3.1.



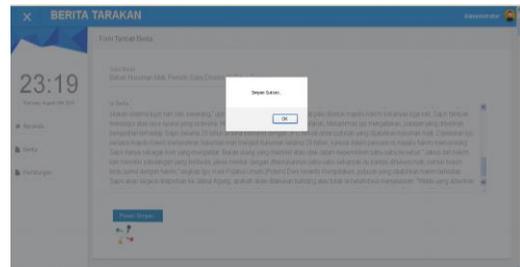
Gambar 3.1 Operasi Program Form Admin

Setelah itu, masuk ke form Berita dimana pada form ini adalah tempat inputan judul dan isi berita yang akan disimpan.



Gambar 3.2 Operasi Program Form Berita

Setelah judul dan isi berita di input maka admin melakukan proses simpan berita. Jika proses simpan berhasil maka akan muncul pesan yang menandakan bahwa proses simpan telah berhasil.



Gambar 3.3 Operasi Program Form Berita Tersimpan

Setelah memperoleh ringkasan sistem dan ringkasan manual dilakukan perhitungan akurasi menggunakan precision, recall, dan f-measure. Precision adalah menghitung jumlah ringkasan sistem yang ada pada ringkasan manual dibagi dengan jumlah ringkasan sistem, Recall adalah menghitung jumlah ringkasan sistem yang ada pada ringkasan manual dibagi dengan jumlah ringkasan manual. F-Measure digunakan untuk mengukur kombinasi antara precision dan recall yaitu antara hasil dari ringkasan sistem dan ringkasan manual.

TABLE III  
 TABEL RINGKASAN SISTEM DAN RINGKASAN MANUAL

Kode	Ringkasan Sistem	Ringkasan Manual
1	D1,D2,D3,D4, D5,D6,D7,D11,D13,D19	D1,D3, D6,D13,D19
2	D1,D2,D3,D5, D7,D12,D14	D1,D2,D3,D4, D5,D7,D9,12
3	D1,D3,D4,D9, D16,D17,D19,D20	D2,D3,D5,D8, D11,D12,D16,D22, D24,D25
4	D1,D3,D8,D17, D19,D20,D21	D1,D3,D4,D9,D13, D17,D19,D20
5	D1,D3,D4,D5, D6,D10,D12	D1,D3,D4,D5,D6,D11
6	D1,D2,D3,D5, D6,D8,D9,D15	D1,D2,D3,D4,D9, D13,D15
7	D1,D3,D4,D7, D10,D11	D1,D2,D4,D7,D9, D10,D11
8	D1,D2,D3,D7,D13	D1,D2,D3,D7,D12
9	D1,D2,D3,D5,D8,D10,D15	D1,D3,D4,D6,D8,D10
10	D1,D2,D3,D5,D7,D8,D10	D4,D5,D7,D8

Dilakukan perhitungan untuk mencari nilai precision, Recall, F-Measure untuk berita.

**Precision**

$$= \frac{\# \text{kalimat ringkasan sistem} \cap \text{ringkasan manual}}{\sum \text{kalimat ringkasan sistem}}$$

$$\text{Precision 1} = \frac{5 \cap 5}{10} = 50\%$$

$$\text{Precision 2} = \frac{6 \cap 8}{7} = 86\%$$

$$\text{Precision 3} = \frac{2 \cap 10}{8} = 25\%$$

$$\text{Precision 4} = \frac{5 \cap 8}{7} = 71\%$$

$$\text{Precision 5} = \frac{5 \cap 6}{7} = 83\%$$

$$\text{Precision 6} = \frac{5 \cap 7}{8} = 63\%$$

$$\text{Precision 7} = \frac{5 \cap 7}{6} = 83\%$$

$$\text{Precision 8} = \frac{4 \cap 5}{5} = 80\%$$

$$\text{Precision 9} = \frac{4 \cap 6}{7} = 57\%$$

$$\text{Precision 10} = \frac{3 \cap 5}{7} = 43\%$$

Jadi nilai precision dari Dokumen 1, 2, 3, 4, 5, 6, 7, 8, 9 dan 10 adalah 50%, 86%, 25%, 71%, 83%, 63%, 83%, 80%, 57%, dan 43% Selanjutnya dengan contoh yang sama pada tabel 4.1 dilakukan perhitungan untuk nilai recall.

**Recall**

$$= \frac{\# \text{kalimat ringkasan sistem} \cap \text{ringkasan manual}}{\sum \text{kalimat ringkasan manual}}$$

$$\text{Recall 1} = \frac{10 \cap 5}{5} = 100\%$$

$$\text{Recall 2} = \frac{7 \cap 6}{8} = 75\%$$

$$\text{Recall 3} = \frac{8 \cap 2}{10} = 20\%$$

$$\text{Recall 4} = \frac{7 \cap 5}{8} = 63\%$$

$$\text{Recall 5} = \frac{7 \cap 5}{7} = 71\%$$

$$\text{Recall 6} = \frac{8 \cap 5}{7} = 71\%$$

$$\text{Recall 7} = \frac{6 \cap 5}{7} = 71\%$$

$$\text{Recall 8} = \frac{5 \cap 4}{5} = 80\%$$

$$\text{Recall 9} = \frac{7 \cap 4}{6} = 67\%$$

$$\text{Recall 10} = \frac{7 \cap 3}{5} = 60\%$$

Nilai untuk recall untuk dokumen 1, 2, 3, 4, 5, 6, 7, 8, 9, dan 10 adalah 100%, 75%, 20%, 63%, 71%, 71%, 71%, 80%, 67%, dan 60%. Kemudian dilakukan perhitungan untuk mencari nilai F-Measure.

$$f - \text{measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Recall} + \text{Precision}}$$

$$F - \text{Measure 1} = \frac{2 * 50\% * 100\%}{50\% + 100\%} = 66,67\%$$

$$F - \text{Measure 2} = \frac{2 * 86\% * 75\%}{86\% + 75\%} = 80,12\%$$

$$F - \text{Measure 3} = \frac{2 * 25\% * 20\%}{25\% + 20\%} = 22,22\%$$

$$F - Measure 4 = \frac{2 * 71\% * 63\%}{71\% + 63\%} = 66,67\%$$

$$F - Measure 5 = \frac{2 * 83\% * 71\%}{83\% + 71\%} = 76,53\%$$

$$F - Measure 6 = \frac{2 * 63\% * 71\%}{63\% + 71\%} = 66,67\%$$

$$F - Measure 7 = \frac{2 * 83\% * 71\%}{83\% + 71\%} = 76,53\%$$

$$F - Measure 8 = \frac{2 * 80\% * 80\%}{80\% + 80\%} = 80\%$$

$$F - Measure 9 = \frac{2 * 57\% * 67\%}{57\% + 67\%} = 61,6\%$$

$$F - Measure 10 = \frac{2 * 43\% * 60\%}{43\% + 60\%} = 50,1\%$$

Dari hasil perhitungan f-measure diperoleh nilai dokumen 1 adalah 66%. Tabel 4.2 hasil perolehan nilai dari precision, recall dan f-measure.

TABLE 3.4  
 TABEL RINGKASAN SISTEM DAN RINGKASAN MANUAL

Kode Abstraksi	Precision	Recall	F-Measure
1	50%	100%	66,67%
2	86%	75%	80,12%
3	25%	20%	22,22%
4	71%	63%	66,67%
5	83%	71%	76,53%
6	63%	71%	66,67%
7	83%	71%	76,53%
8	80%	80%	80%
9	57%	67%	61,6%
10	60%	60%	50,1%

Tabel 3.4 menunjukkan hasil perhitungan yang diperoleh dari perhitungan precision, recall dan f-measure. Tampilan aplikasi untuk menghitung precision, recall, dan f-measure dapat dilihat pada gambar 3.4.

Pada gambar 3.4 user memasukkan kode berita, kemudian memasukkan jumlah overlap dari ringkasan sistem dan ringkasan manual yaitu jumlah ringkasan sistem yang ada pada ringkasan manual atau jumlah ringkasan manual yang ada pada ringkasan sistem. Selanjutnya memasukkan jumlah ringkasan sistem yaitu ringkasan yang dilakukan oleh sistem, dan memasukkan jumlah ringkasan manual yaitu ringkasan yang

dilakukan oleh peringkasan manual. Pada penelitian ini, digunakan 10 dokumen data berita untuk menghitung nilai precision, recall dan f-measure. Untuk mendapatkan jumlah dokumen yang dihasilkan sesuai dengan kata kunci.

Gambar 3.4 Tampilan Form Uji coba Perhitungan Precision, Recall, F-Measure

Gambar 3.5 menggambarkan nama, nama adalah nama pengguna yang mengisi ringkasan manual, jabatan adalah jabatan yang dimiliki oleh pengguna yang mengisi ringkasan sistem, kata kunci adalah query yang dipakai dalam peringkasan, D1 s.d D14 merupakan dokumen hasil pemecahan kalimat, kolom kosong dibagian bawah merupakan kolom isian ringkasan manual.

Kode	: 01
Nama	: Andi Tenri Puji, S.kom
Pekerjaan	: Dosen
Kata kunci	: Defisi, Dana, Silfa, Habis
D1	= Pemerintah Kota Tarakan memang tidak jadi mengutang untuk menutupi kewajiban yang belum dibayar di tahun anggaran 2015, tetapi dana Sisa Lebih Penggunaan Anggaran (Silpa) yang dua tahun terakhir berhasil menutupi defisit, kini sudah habis.
D2	= Badan Pemeriksa Keuangan (BPK) altara memaparkan, dalam dua tahun belakangan Pemkot Tarakan sebenarnya mengalami defisit anggaran.
D3	=Namun berhasil ditutupi dengan Silpa.
D4	= Menurut data yang dipaparkan Kepala Kantor BPK perwakilan Kalimantan Utara (Kaltara), Ade Iwan Rusmana, pada tahun 2012 Pemkot Tarakan memiliki Rp 859.30 miliar, namun karena defisit tahun 2013 sebesar Rp 323 miliar, sehingga dana silpa tersisa Rp 531.15 miliar.
D5	= Kemudian berkurang lagi akibat defisit tahun 2014 sebesar Rp 325.89 sehingga silpa di akhir tahun 2014 tersisa Rp 205.27 miliar.
D6	= Angka silpa tersebut yang kemudian dipertanyakan mengapa tidak digunakan.
D7	= Sekretaris Kota (sekkot) Tarakan dr. Khairul yang juga menjabat ketua tim penyusun anggaran daerah (TPAD) menjelaskan, dana Rp 205.27 miliar telah masuk di struktur anggaran 2015 dan telah habis digunakan.
D8	= "Itukan sudah masuk di tahun anggaran 2015.
D9	= Sudah kami pakai.
D10	= Ini setelah kami hitung sampai 31 Desember 2015 Pemkot masih memiliki kewajiban yang harus dibayar sebesar Rp 95 miliar.
D11	= "Walaupun sekarang tidak punya uang cash, tapi kami masih punya uang dari dana bagi hasil (DBH) yang belum ditransfer dari pemerintah pusat dan provinsi," jelas mantan Kepala Dinas Kesehatan itu.
D12	= Nilai DBH tersebut terdiri dari Rp 44 miliar dari pemerintah pusat dan Rp 53 miliar Pemprov, dengan total Rp 97 miliar.
D13	= Diakui Khairul, dana yang dimiliki Pemkot sempat mencapai hampir Rp 1 triliun di tahun 2012 lalu, sekarang sudah habis dan hanya tersisa Rp 2 miliar jika DBH yang sejumlah Rp 97 miliar tersebut telah ditransfer dan dikurangi Rp 95 miliar kewenangan kewajiban bayar Pemkot tahun 2015.
D14	= "Sudah dipakai, sisa Rp 2 miliar.
D15	= Tapi itu justru bagus.
D16	= Instruksi dari pemerintah pusat untuk jangan menyimpan uang.
D17	= Gunakan untuk pembangunan dan program-program prioritas.
D18	= Kalau simpanan kita tinggi (dari silpa, Red.) maka dipertanyakan bagaimana dengan pembangunan kita.
D19	= Kalau seluruh anggaran sudah digunakan dan habis, berarti serapannya bagus," jelasnya.
Ringkasan Manual	:

Gambar 3.5 Tampilan Uji Coba Ringkasan Manual

Peringkasan manual mengisikan hasil ringkasan pada kolom dibagian bawah dengan sesuai dengan dokumen yang menjadi hasil ringkasan. Contoh yang diambil pada gambar 4.8 (misal : D1,D4,D5) dan diisi di bagian kolom bawah yang telah disediakan. Untuk melakukan ringkasan manual sama dengan peringkasan sistem, dengan mengukur kemiripan dari query

(kata kunci) dengan dokumen-dokumen, kemudian mengukur kemiripan antara dokumen dengan dokumen lainnya. Bentuk ringkasan manual untuk mengukur kinerja efektifitas dari suatu ringkasan sistem ditunjukkan pada gambar 4.8.

Peringkasan berita pada penelitian ini menggunakan data yang diambil dari Berita Radar Tarakan dengan jumlah 150 data berita yang kemudian diolah untuk menghasilkan sebuah ringkasan. Dengan memanfaatkan vector space model sebagai model untuk pembobotan query relevance, cosine similarity untuk memperoleh hasil ringkasan sistem.

Hasil peringkasan dipengaruhi oleh kata kunci pada berita dan dipengaruhi persamaan antara kalimat di dalam dokumen berita hasil pemecahan kalimat.

Hasil ringkasan diambil dari satu atau dua kalimat asli (bisa lebih dari dua) merupakan kelemahan karena maknanya bisa terjadi tidak integrated dikarenakan hasil ringkasan diambil secara berurut dari nilai tertinggi ke nilai terendah, banyaknya jumlah dokumen akan mempengaruhi waktu peringkasan, karena semakin banyak dokumen maka semakin lama waktu yang dibutuhkan sistem untuk meringkas..

Hasil uji coba yang dilakukan menghasilkan rata-rata precision 64,1%, recall 67,8%, dan f-measure 64,73% digunakan 10 dokumen data berita untuk menghitung nilai precision, recall dan f-measure.

#### IV. KESIMPULAN

Dari pembahasan pada bab-bab sebelumnya maka dapat disimpulkan:

1. Dokumen dengan nilai  $>0$  dan diurut dari nilai maksimal ke nilai minimal akan diambil sebagai ringkasan, kalimat yang diambil sebagai ringkasan merupakan kalimat yang mempresentasikan kesamaan pada kalimat dalam dokumen dengan query dan kesamaan antar kalimat dalam dokumen. Pemodelan vector space model digunakan untuk query relevance, dan cosine similarity untuk similarity kalimat.
2. Hasil uji coba yang dilakukan menghasilkan rata-rata precision 64,1%, recall 67,8%, dan f-measure 64,73% berdasarkan perbandingan ringkasan sistem dengan ringkasan manual. Data uji coba diambil dari tugas akhir dan skripsi mahasiswa sebanyak 10 dokumen berita. Dan digunakan sebagai data dalam peringkasan sistem dan peringkasan manual, kemudian data hasil ringkasan sistem dibandingkan dengan ringkasan manual untuk memperoleh hasil akurasi.
3. Hasil ringkasan diambil dari satu atau dua kalimat asli (bisa lebih dari 2) merupakan kelemahan karena maknanya bisa menjadi tidak *integrated*.
4. kelemahan karena semakin banyak kalimat pada dokumen semakin lama waktu meringkas.

#### REFERENSI

- [1] Martin Hassel, Evaluation of Automatic Text Summarization. Stockholm, Sweden, 2004
- [2] Eduard Hovy, Text Summarization Chapter 32, Information Sciences Institute of the University of Southern California.
- [3] Robert M. Losee, Journal of the American Society for Information Science and Technology. North Carolina, 2001
- [4] B. Zamand, E.Winarko., Analisis Fitur Kalimat untuk Peringkasan Teks Otomatis pada Bahasa Indonesia, IJCCS, 2011.

- [5] Jiwa Malem Marsya, Analisa dan Evaluasi Afiks Stemming untuk Bahasa Indonesia, FMIPA Universitas Syiah Kuala, 2011.
- [6] Muhamad Iqbal Aziz, Development Program Application to the Measurement of Documents Resemblance Text Mining, tf-idf, and Vector Space Model Algorithm, Yogyakarta : Faculty of Industrial Engineering. Gunadarma University, 2010.

**Penulis I**, Rina Anggeriani, memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Sistem Informasi STMIK PPKIA Tarakanita Rahmawati Tarakan, lulus tahun 2016

**Penulis II**, Fitria, memperoleh gelar Magister Komputer (M.Kom), Sekolah Tinggi Teknik Surabaya. Saat ini menjadi Dosen di STMIK PPKIA Tarakanita Rahmawati

**Penulis III**, Muhammad Hafid., memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Sistem Informasi STMIK PPKIA Tarakanita Rahmawati Tarakan, lulus tahun 2014. Saat ini menjadi Dosen di STMIK PPKIA Tarakanita Rahmawati