Strategi Pemilihan Kalimat pada Peringkasan Multi Dokumen

Satrio Verdianto, Agus Zainal Arifin, dan Diana Purwitasari
Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember (ITS)
Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia
e-mail: agusza@cs.its.ac.id

Abstrak*—Ringkasan berita diartikan sebagai teks yang dihasilkan dari satu atau lebih kalimat yang menyampaikan informasi penting dari berita. Salah satu fase penting dalam peringkasan adalah pembobotan kalimat (*sentence scoring*). Dimana pada peringkasan berita, metode pembobotannya sebagian besar menggunakan fitur dari berita sendiri. Berdasarkan hasil dari penelitian [3] bahwa untuk pembobotan kalimat pada dokumen yang memiliki karakter teks pendek dan terstruktur seperti berita maka teknik pembobotan kalimat terbaik adalah dengan menggunakan kombinasi dari keempat fitur yaitu *word frequency*, TF-IDF, posisi kalimat, dan kemiripan kalimat terhadap judul *(Resemblance to the title).

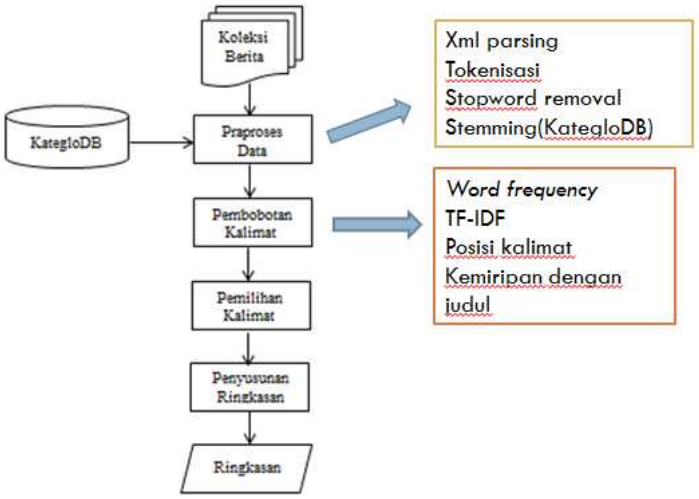
Pada penelitian ini kombinasi keempat fitur tersebut dibandingkan dengan kombinasi tiga fitur dan dua fitur dan dievaluasi menggunakan nilai ROUGE-N dan dievaluasi berdasarkan lama waktu eksekusi. Berdasarkan hasil uji coba didapatkan hasil bahwa yang paling optimal diantara keempat kombinasi fitur tersebut adalah kombinasi antara dua buah fitur yakni fitur posisi kalimat dan *word frequency dengan nilai ROUGE-N sebesar 0.679 dan lama waktu eksekusi 28.458 detik.*

Kata Kunci*— *kemiripan kalimat terhadap judul, pembobotan kalimat, posisi kalimat, ROUGE-N, TF-IDF, word frequency

I. PENDAHULUAN

KEBUTUHAN untuk mengakses informasi khususnya berita secara praktis menjadi masalah yang harus diselesaikan seiring berkembang-pesatnya berita yang dapat diakses secara *online*. Peringkasan berita secara otomatis adalah salah satu solusi untuk menjawab permasalahan diatas. Ringkasan berita dapat diartikan sebagai sebuah teks yang dihasilkan dari satu atau lebih kalimat yang mampu menyampaikan informasi penting dari sebuah berita. Dimana panjang dari sebuah ringkasan tidak lebih dari setengah panjang dokumen asli, dan biasanya lebih pendek [7]. Peringkasan multi dokumen berita merupakan sistem peringkasan yang melibatkan lebih dari satu berita sebagai input.

Selain itu, dibutuhkan teknik pembobotan kalimat yang handal untuk dapat menghasilkan ringkasan berita yang baik. Berdasarkan hasil dari penelitian [3] bahwa untuk pembobotan kalimat pada dokumen yang memiliki karakter teks pendek dan terstruktur seperti berita maka teknik pembobotan kalimat terbaik adalah dengan menggunakan kombinasi dari keempat fitur yaitu *word frequency*, TF-IDF, posisi, dan *Resemblance to the title.*



Gambar 1. Diagram alir proses sistem secara umum

Sebelum memasuki proses pembobotan, koleksi dokumen berita melalui fase praproses data. Fase praproses data meliputi proses xml *parsing, tokenizing*, *stopword removal*, dan *stemming*. XML *parsing* adalah proses pengubahan data .xml ke bentuk *string* atau teks. *Tokenizing* adalah proses pemenggalan kata-kata sehingga setiap kata dapat berdiri sendiri. *Stopword removal* adalah proses menghapus kata kunci yang tidak layak untuk digunakan, seperti kata sambung, kata depan, kata ganti dls. Sedangkan *stemming* adalah proses untuk memperoleh kata dasar dari setiap kata. Dalam tugas akhir ini *stemming* dilakukan dengan memanfaatkan kategloDB. Proses *stemming* dilakukan dengan mengubah setiap kata ke bentuk dasarnya dengan merujuk ke kategloDB. Data hasil praproses disimpan ke dalam database.

II. TINJAUAN PUSTAKA

A. *Word Frequency*

Konsep dari *Word Frequency* (*WF*) adalah semakin sering suatu kata muncul dalam sebuah teks maka kata tersebut dianggap sebagai kata penting [3]. Sehingga untuk mendapatkan kata-kata penting dari sebuah dokumen dilakukan pembobotan kata dengan menghitung frekuensi kemunculan kata tersebut pada dokumen. Semakin besar frekuensi kemunculan sebuah kata maka skornya akan semakin tinggi. Langkah awal yang dilakukan adalah ekstraksi *term* dari dokumen kemudian memberikan bobot pada tiap

term tersebut berdasarkan jumlah kemunculan *term* pada dokumen. Kemudian meranking *term* berdasarkan bobot dan menyeleksi *term* yang memiliki bobot diatas nilai ambang (*threshold*). *Term* yang terseleksi akan menjadi *Word Frequency List* (*WFList*). *WFList* inilah yang nantinya digunakan sebagai fitur pada pembobotan kalimat dengan cara mengukur kemiripan antara kalimat terhadap *WFList*. Metode untuk mengukur kemiripan dapat menggunakan *cosine similarity* atau metode pengukur kemiripan yang lain.

B. TF-IDF

Term Frequency Inverse Document Frequency (TF-IDF) adalah konsep pembobotan *term* pada sebuah dokumen. Ketika TF-IDF diterapkan pada lingkup kalimat, maka sebuah kalimat akan diberlakukan sebagai dokumen. Konsep dari TF-IDF adalah jika ada "kata-kata yang spesifik" muncul pada kalimat tertentu maka kalimat tersebut relatif dianggap sebagai kalimat penting [2]. Metode ini melakukan perbandingan antara frekuensi kemunculan *term j* pada kalimat *i* (TF_{ij}) dengan frekuensi kalimat yang mengandung *term j* (DF_j). Bobot TF-IDF dari *term j* dapat dihitung dengan menggunakan persamaan 1, dimana $tf_{w_i doc_j}$ adalah frekuensi kata *w* ke-*i* pada dokumen ke-*j*. Konsep tersebut memberikan pengukuran terhadap pentingnya kata *w* ke-*i* pada dokumen tersebut. Sedangkan idf_{w_i} ditentukan melalui Persamaan 2, dimana *N* adalah jumlah dokumen, df_{w_i} adalah jumlah dari dokumen yang mengandung kata *w* ke-*i*.

$$tf\_idf_{w_i doc_j} = tf_{w_i doc_j} * idf_{w_i} \quad (1)$$

$$idf_{w_i} = \log(\frac{N}{df_{w_i}}) \quad (2)$$

C. Posisi Kalimat

Posisi kalimat merupakan salah satu fitur yang dapat digunakan untuk pembobotan kalimat. Dimana penilaiannya berdasarkan pada letak kalimat dalam sebuah dokumen. Sama seperti penelitian [10] yang menggunakan posisi sebagai salah fitur pembobotan kalimat. Dengan menggunakan aturan, kalimat yang posisinya berada diawal dokumen memiliki skor lebih besar dibanding kalimat yang posisinya diakhir. Penelitian tersebut mampu memberikan penjelasan ilmiah tentang alasan penggunaan aturan tersebut untuk pembobotan kalimat dengan mengutip pernyataan dari Baxendale bahwa kebanyakan kalimat yang muncul diawal paragraf merupakan *topic sentence*. Hal inilah yang menjadi dasar Jiang-ping (2012) untuk memberikan skor lebih besar pada kalimat yang muncul di awal dokumen. Namun sebenarnya, *topic sentence* kurang tepat digunakan sebagai alasan dikarenakan *topic sentence* berlaku untuk semua jenis tulisan *term*asuk berita *online*. Dan *topic sentence* bisa saja muncul disemua posisi dokumen, baik di awal, tengah, maupun akhir.

Dalam ilmu jurnalistik, ada beberapa teknik penulisan berita. Teknik yang paling banyak digunakan untuk berita *online* adalah "piramida terbalik". Pola "piramida terbalik" merupakan teknik penulisan berita yang dimulai atau diawali dari kalimat yang dianggap paling penting, setelah itu diikuti hal-hal yang kurang penting. Penelitian ini meyakini bahwa alasan ini merupakan alasan yang tepat untuk memberikan skor lebih besar pada kalimat yang ada di posisi awal dibanding dengan penggunaan alasan *topic sentence*.

D. Kemiripan Kalimat dengan Judul Berita

Judul berita merupakan satu komponen penting dalam penulisan berita. Dalam berita *online*, judul ditulis secara ringkas dan jelas. Sebuah judul minimal mengandung unsur S-P-O-K (Subyek – Predikat – Obyek – Keterangan) dan dapat diambil dari beberapa kata atau kutipan yang ada dalam isi berita. Hal inilah yang menjadi dasar penggunaan judul sebagai informasi untuk mengetahui kalimat penting dalam sebuah berita. Konsep dari teknik pembobotan kalimat berdasarkan kemiripan kalimat terhadap judul adalah bahwa bobot sebuah kalimat besar ketika nilai kemiripan antara judul dengan kalimat tinggi. Semakin besar bobot kalimat maka kalimat tersebut akan dianggap semakin penting. Hal ini sama seperti yang ada pada penelitian [2] bahwa kalimat yang mirip dengan judul dan kalimat yang mencakup kata-kata dalam judul yang akan dianggap sebagai kalimat penting.

III. PEMBOBOTAN KALIMAT DAN PEMILIHAN KALIMAT SEBAGAI RINGKASAN

A. Pembobotan Kalimat

Fase pembobotan kalimat merupakan proses perhitungan empat buah fitur untuk tiap kalimat. Keempat buah fitur tersebut ialah fitur posisi kalimat, fitur *word frequency,* fitur TF-IDF, dan fitur kemiripan kalimat dengan judul. Konsep dari pembobotan kalimat yang pertama adalah dengan menggunakan *WF*. Dalam hal ini *WFList* didapatkan dari sejumlah *term* dengan nilai *WF* memenuhi nilai ambang (*threshold*), *WFList* = {$WF_1,\ldots,WF_k$}. Pembobotan kalimat dihitung berdasarkan nilai kemiripan antara kalimat terhadap *WFList*, persamaan 3. Dalam penelitian ini digunakan *cosine similarity* untuk mengukur kemiripan antara kalimat dengan *WFList*, Bobot kalimat berdasarkan *WF* untuk selanjutnya disebut dengan w_1, dengan *S* adalah kalimat. Sehingga $w_1(s_i)$ adalah nilai kemiripan kalimat s_i terhadap *WFList*, dimana $S=\{s_1,\ldots,s_n\}$.

Pembobotan kalimat kedua (w_2) pada penelitian ini menggunakan pendekatan TF-IDF. Setelah didapatkan bobot tiap *term* $TFIDF_{ij}$ dengan menggunakan persamaan 1, langkah selanjutnya adalah menghitung bobot kalimat berdasarkan bobot TF-IDF yang selanjutnya disebut dengan w_2 menggunakan persamaan 4. w_2 merupakan hasil penjumlahan dari seluruh bobot *term j* yang muncul pada kalimat *i* (s_i).

Pembobotan kalimat ketiga (w_3) menggunakan fitur posisi. w_3 dihitung dengan menggunakan persamaan 5 yang mengadopsi dari penelitian (Mei & Chen, 2012). Dengan aturan, kalimat yang posisinya berada diawal dokumen memiliki skor lebih besar dibanding kalimat yang posisinya diakhir.

Pembobotan kalimat keempat (w_4) melibatkan judul berita (*Title*). Penghitungan w_4 menggunakan persamaan 6 yang mengadopsi dari [2] yaitu dengan cara membagi antara jumlah *term* judul yang muncul pada kalimat (*Ntw*) dengan jumlah seluruh *term* yang ada pada judul (*T*).

$$w_1(s_i) = Sim(s_i, WFList) \quad (3)$$

$$w_2(s_i) = \sum TFIDF_{ij} \quad (4)$$

$$w_3(s_i) = \text{[illegible]} \quad (5)$$

$$w_4(s_i) = \frac{Ntw}{T} \quad (6)$$

Setelah didapatkan bobot w_1 sampai w_4 langkah berikutnya adalah menghitung total bobot kalimat *i* dengan menggunakan persamaan 7. Bobot kalimat yang didapat berdasarkan w_1 sampai w_4 seluruhnya dijumlahkan. Seluruh kalimat akan dihitung bobotnya, hasil dari persamaan 7 inilah yang akan menjadi total bobot kalimat *i*.

$$score(s_i) = w_1(s_i) + w_2(s_i) + w_3(s_i) + w_4(s_i) \quad (7)$$

B. Pemilihan Kalimat sebagai Ringkasan

Fase pemilihan kalimat dan penyusunan ringkasan dilakukan dengan melakukan pengurutan bobot kalimat secara *descending* (terbesar ke terkecil). Kemudian beberapa kalimat dengan bobot terbesar diambil sebagai ringkasan.

IV. UJI COBA DAN EVALUASI

Uji coba dilakukan dengan mengukur performa hasil ringkasan dengan menggunakan kombinasi empat fitur berita yaitu posisi kalimat (p), *word frequency* (w), TF-IDF (t), dan judul berita (j). Nantinya kombinasi 4 fitur akan dibandingkan dengan kombinasi 3 fitur dan kombinasi 2 fitur. Untuk mengukur performansi hasil ringkasan digunakan metode evaluasi ROUGE-N yaitu ROUGE-1 dan evaluasi berdasarkan waktu eksekusi.

A. Dataset

Pengujian pada sistem peringkasan dalam penelitian ini dilakukan dengan membandingkan hasil ringkasan sistem dengan hasil ringkasan manusia dengan menggunakan ROUGE-N. Pengujian dilakukan terhadap 15 kelompok dokumen berita berformat .xml yang dikelompokkan berdasarkan topik dimana masing-masing kelompok memiliki jumlah dokumen berita yang dijelaskan pada Tabel 1. Ringkasan yang dihasilkan terdiri dari 10 buah kalimat untuk masing-masing topik.

Tabel 1. Dataset berita

NO	TOPIK BERITA	JUMLAH BERITA
1	BLBI	4
2	LG G4	3
3	Kunjungan Mark Zuckerberg	4
4	Internet Indonesia lambat	2
5	Intel	3
6	Prosesor baru Intel	3
7	Smartphone 4G Intel	3
8	Kunjungan Jokowi	2
9	Pidato Presiden dan pemberian penghargaan	3
10	Saran SBY	3
11	Proyek LRT	3
12	Jokowi ke Arab Saudi	2
13	Jokowi ancam copot menteri	2
14	Ulang Tahun Jokowi	5
15	Unikom dan UPI di LIMA Badminton	3
	TOTAL	45

B. Evaluasi berdasarkan Nilai ROUGE-1

Dari Tabel 2 dapat dilihat bahwa kombinasi empat fitur yakni posisi kalimat (*p*), *word frequency (w),* TF-IDF (*t*), dan judul berita (*j*) memiliki total nilai ROUGE-1 terbesar kedua setelah kombinasi tiga fitur yakni posisi kalimat, *word frequency,* dan TF-IDF. Perbedaan ditunjukkan pada dataset ringkasan ke-4 yang disebabkan oleh struktur xml yang kurang tepat sehingga mengakibatkan sistem dengan kombinasi *pwtj* mengambil kalimat yang bukan bagian dari berita, yakni kalimat *"Baca juga"*

Tabel 2. Nilai ROUGE-1 antara ringkasan sistem (kombinasi 2, 3, dan 4 fitur) dengan ringkasan *groundtruth*

RINGKASAN GROUNDTRUTH	NILAI ROUGE-1 TIAP KOMBINASI FITUR (p=posisi, w=word frequency, t=tfidf, j=judul)										
	pwtj	pwt	pwj	ptj	wtj	pw	pt	pj	wt	wj	tj
1	0.83221	0.83221	0.83221	0.67542	0.72477	0.83221	0.67542	0.38254	0.72477	0.75829	0.48445
2	0.38267	0.38267	0.34409	0.33566	0.27206	0.34409	0.33566	0.44898	0.27206	0.29213	0.38225
3	0.78801	0.78801	0.79911	0.80973	0.71548	0.79911	0.80973	0.50679	0.71548	0.55895	0.60938
4	0.6962	0.79654	0.62472	0.70782	0.43441	0.72517	0.75162	0.64615	0.49443	0.39912	0.51089
5	0.6501	0.6501	0.62128	0.60285	0.51402	0.62128	0.60285	0.39443	0.51402	0.47826	0.56513
6	0.61191	0.61191	0.61191	0.68526	0.569	0.61191	0.68526	0.57328	0.569	0.53719	0.58027
7	0.76082	0.76082	0.76082	0.7713	0.52008	0.76082	0.7713	0.61157	0.52008	0.52008	0.54656
8	0.504	0.504	0.504	0.52713	0.43103	0.504	0.52713	0.40304	0.43103	0.51012	0.58451
9	0.67269	0.67269	0.67269	0.79175	0.53215	0.67269	0.79175	0.40429	0.53215	0.53881	0.38968
10	0.59726	0.59726	0.59726	0.49874	0.46991	0.59726	0.49874	0.33939	0.46991	0.40491	0.36317
11	0.78286	0.78286	0.78286	0.7181	0.53786	0.78286	0.7181	0.40816	0.53786	0.53786	0.50867
12	0.94944	0.94944	0.81346	0.8	0.6513	0.81346	0.8	0.68599	0.6513	0.63128	0.67005
13	0.60047	0.60047	0.64732	0.67317	0.63514	0.64732	0.67317	0.66667	0.63514	0.63514	0.64286
14	0.79741	0.79741	0.8341	0.8341	0.62737	0.8341	0.8341	0.447	0.62737	0.61123	0.41942
15	0.58364	0.58364	0.6457	0.65376	0.60571	0.6457	0.65376	0.47692	0.60571	0.59398	0.49737
TOTAL	10.20969	10.31003	10.09153	10.08479	8.24029	10.19198	10.12859	7.3952	8.30031	8.00735	7.75466
RATA-RATA	0.680646	0.687335	0.672769	0.672319	0.549353	0.679465	0.675239	0.493013	0.553354	0.533823	0.516977
URUTAN	2	1	5	6	8	3	4	11	7	9	10

nilai tertinggi — nilai terkecil

Selanjutnya posisi ketiga adalah kombinasi dua fitur yakni posisi kalimat dan *word frequency*. Perbedaan nilai ROUGE-1 diantara ketiganya pun dapat dikatakan sangat kecil yakni sekitar 0.01. Hal ini menunjukkan bahwa kombinasi empat fitur bukanlah yang terbaik dalam menghasilkan ringkasan yang baik. Selain itu, hasil ini juga menunjukkan bahwa fitur posisi kalimat dan *word frequency* mengambil peranan penting dalam menghasilkan ringkasan yang baik. Sebaliknya fitur kemiripan dengan judul berita tidak peranan penting karena dengan atau tanpa fitur tersebut nilai ROUGE-1 tidak menunjukkan perbedaan yakni antara *pwtj* dan *pwt*. Bahkan pada kombinasi tiga fitur dan dua fitur dapat dilihat bahwa penggunaan fitur tersebut menghasilkan nilai ROUGE-1 yang lebih kecil dibandingkan menggunakan fitur lain.

C. Evaluasi berdasarkan Waktu Eksekusi

Selain itu, uji coba juga dilakukan dengan mengukur waktu eksekusi masing-masing kombinasi untuk seluruh berita.

Tabel 3. Lama waktu eksekusi program (dalam satuan detik) tiap kombinasi fitur untuk tiap topik berita

TOPIK BERITA	LAMA WAKTU EKSEKUSI (detik) TIAP KOMBINASI FITUR (p=posisi, w=word frequency, t=tfidf, j=judul)										
	pwtj	pwt	pwj	ptj	wtj	pw	pt	pj	wt	wj	tj
1	58.611	62.921	57.114	65.169	55.764	51.504	53.929	48.504	57.308	50.092	59.809
2	35.333	27.453	25.107	27.915	28.19	27.215	28.351	24.042	27.05	24.85	28.928
3	24.832	23.303	24.433	25.115	22.905	22.072	22.541	20.436	23.568	23.302	22.804
4	30.244	28.185	27.469	30.018	30.157	26.623	25.838	25.919	28.423	24.546	27.204
5	26.384	42.805	23.453	23.793	24.976	22.045	25.556	20.907	27.579	22.081	22.85
6	29.062	26.493	22.803	23.23	23.583	22.857	23.988	22.531	25.338	28.893	23.768
7	21.614	20.307	20.012	23.18	19.509	18.195	19.299	16.858	19.22	18.693	21.051
8	16.783	16.732	16.198	18.825	17.13	17.571	16.481	14.941	17.05	16.998	16.683
9	41.057	37.637	39.223	41.214	37.898	36.606	38.398	33.63	40.22	35.419	38.617
10	37.729	36.186	32.094	34.658	41.536	30.604	34.026	31.587	36.052	30.829	34.674
11	28.073	32.132	26.505	29.66	29.832	28.818	28.748	24.206	27.796	25.93	30.958
12	20.549	19.825	18.304	18.996	18.84	17.824	18.566	17.159	19.12	16.86	19.942
13	13.461	12.948	11.94	12.484	11.809	11.427	12.323	10.421	10.707	10.853	12.143
14	83.463	78.305	75.619	88.96	83.715	78.32	86.671	72.324	80.06	77.115	78.665
15	17.998	16.9	16.025	23.267	16.465	15.184	17.27	13.946	15.428	14.728	15.895
TOTAL	485.193	482.132	436.299	486.484	462.309	426.865	451.985	397.411	454.919	421.189	453.991
RATA-RATA	32.3462	32.14213	29.0866	32.43227	30.8206	28.45767	30.13233	26.49407	30.32793	28.07927	30.26607
URUTAN	10	9	4	11	8	3	5	1	7	2	6

waktu tercepat — waktu terlama

Dari Tabel 3 dapat dilihat bahwa urutan rata-rata waktu eksekusi seluruh kombinasi terhadap 15 topik berita dari yang tercepat hingga yang terlambat ialah sebagai berikut.

1. pj
2. wj
3. pw

4. pwj
5. pt
6. tj
7. wt
8. wtj
9. pwt
10. pwtj
11. ptj

Pengukuran berdasarkan waktu eksekusi saja tentunya tidak dapat dijadikan landasan mutlak untuk mengukur performa suatu metode. Maka dari itu, untuk mengetahui kombinasi yang paling optimal untuk digunakan dalam proses pemilihan kalimat, analisis berdasarkan waktu eksekusi akan dipadukan dengan analisis berdasarkan nilai ROUGE-1.

D. Analisis

Tabel 4 menunjukkan urutan kombinasi berdasarkan nilai ROUGE-1 dan waktu eksekusi. Nilai ROUGE-1 diurutkan secara *descending* (terbesar - terkecil) sedangkan waktu eksekusi diurutkan secara *ascending* (tercepat – terlambat).

Dari tabel tersebut dapat diambil disimpulkan bahwa kombinasi dua fitur yakni posisi kalimat dan *word frequency* merupakan kombinasi yang optimal untuk mendapatkan ringkasan yang baik dengan waktu yang cukup cepat.

Kombinasi dua fitur yakni posisi kalimat dan *word frequency* merupakan kombinasi yang optimal disebabkan oleh hal-hal berikut.

1. Sebagian besar berita cenderung menyampaikan ide pokoknya pada awal-awal kalimat sedangkan kalimat-kalimat selanjutnya merupakan penjelas atau bahkan informasi-informasi lain di luar pokok bahasan. Sehingga dengan menggunakan fitur posisi kalimat, kita dapat mengambil intisari dari berita tersebut. Selain itu, perhitungan skor posisi kalimat juga sangat sederhana (persamaan 5) sehingga tidak memakan waktu eksekusi program.
2. Kalimat-kalimat berita yang dapat dijadikan sebagai ringkasan secara umum mengandung kata-kata yang sering muncul pada kumpulan dokumen.
3. Fitur TF-IDF sebenarnya merupakan fitur yang cukup penting untuk menghasilkan ringkasan yang baik. Hal ini dapat dilihat dari nilai ROUGE-1 yang tinggi ketika menggunakan fitur TF-IDF. Namun, jika dilihat berdasarkan waktu, penggunaan fitur TF-IDF cukup memakan waktu eksekusi karena dalam prosesnya fitur ini harus menghitung bobot TF-IDF tiap kata di tiap dokumen.

Untuk fitur kemiripan dengan judul berita, berdasarkan hasil uji coba, dapat disimpulkan bahwa fitur tersebut tidak terlalu memegang peranan penting karena dengan atau tanpa fitur tersebut nilai ROUGE-1 tidak menunjukkan perbedaan yakni antara kombinasi 4 fitur dan kombinasi fitur posisi kalimat, *word frequency, dan* TF-IDF. Bahkan pada kombinasi tiga fitur dan dua fitur dapat dilihat bahwa penggunaan fitur tersebut menghasilkan nilai ROUGE-1 yang lebih kecil dibandingkan menggunakan fitur lain.

Tabel 4. Urutan kombinasi berdasarkan ROUGE-1 dan waktu eksekusi

Urutan	Nilai ROUGE-1		Waktu eksekusi (detik)	
	Kombinasi	Rata-rata	Kombinasi	Rata-rata
1	pwt	0.687	pj	26.494
2	pwtj	0.681	wj	28.079
3	pw	0.679	pw	28.458
4	pt	0.675	pwj	29.087
5	pwj	0.673	pt	30.132
6	ptj	0.672	tj	30.266
7	wt	0.553	wt	30.328
8	wtj	0.549	wtj	30.821
9	wj	0.534	pwt	32.142
10	tj	0.517	pwtj	32.346
11	pj	0.493	ptj	32.432

fitur:
p =posisi
w =word frequency
t =tfidf
j =judul

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan uji coba, didapatkan kesimpulan bahwa diantara empat kombinasi fitur yakni fitur posisi kalimat, *word frequency,* TF-IDF, dan judul berita, kombinasi yang paling optimal berdasarkan nilai ROUGE-1 dan waktu eksekusi adalah kombinasi fitur posisi kalimat dan *word frequency dengan nilai ROUGE-1 sebesar 0.679 dan lama waktu eksekusi 28.458 detik.*

B. Saran

Adapun saran untuk pengembangan lebih lanjut dari proses peringkasan multi-dokumen dalam Tugas Akhir ini ialah dilakukan pengembangan lebih lanjut agar tingkat akurasi yang dihasilkan bisa lebih baik yaitu dengan cara mencari tahu nilai parameter-parameter yang optimal contohnya parameter *threshold* jumlah kata yang dimasukkan ke dalam *WFList*.

DAFTAR PUSTAKA

[1] Fachrurrozi, M., Yusliani, N., & Yoanita, R. U. (2013). Frequent Term based Text Summarization for Bahasa Indonesia. *International Conference on Innovations in Engineering and Technology (ICIET'2013)* . Bangkok (Thailand).

[2] Ferreira, R., Cabral, L. d., Lins, R. D., e Silva, G. P., & Freitas, F. (2013). Assessing sentence scoring techniques for extractive text summarization. *Expert Systems with Applications, 40*, 5755–5764.

[3] Ferreira, R., Freitas, F., Cabral, L. d., Lins, R. D., Lima, R., Franc¸a, G., . . . Favaro, L. (2014). A Context Based Text Summarization System. *11th IAPR International Workshop on Document Analysis Systems.* IEEE.

[4] Holi, M. H. (2006). Integrating tf-idf Weighting With Fuzzy View based Search. *Proceedings of the ECAI Workshop on Text-Based Information Retrieval (TIR-06).* Riva del Garda, Italy.

[5] Karel J., J. S. (2008). Automatic Text Summarization (The State of The Art 2007 and New Challenges). *Znalosti* (hal. 1-12). Ústav informatiky a softvérového inziinierstva: FIIT STU Bratislava.

[6] Lin, C. Y. (2004). ROUGE: a Package for Automatic Evaluation of Summaries. *In Proceedings of Workshop on Text Summarization Brances Out* (hal. 74-81). Barcelona: Association for Computational Linguistics.

[7] Radev, D. R., Hovy, E. H., & McKeown, K. (2002). Introduction to the Special Issue on Summarization. *Computational Linguistics, 28*(4), 399-408.

[8] Salton, G., & Buckley, C. (1988). TERM-WEIGHTING APPROACHES IN AUTOMATIC TEXT RETRIEVAL. *Information Processing & Management, 24*, 513-523.

[9] Kavita-Ganesan (2016). *ROUGE 2.0 Documentation - Java Package for Evaluation of Summarization Tasks* [Online].Tersedia:http://kavita-ganesan.com/content/rouge-2.0-documentation [18 Juli 2016]

[10] Mei, J.-P., & Chen, L. (2012). SumCR: A new subtopic-based extractive approach for text summarization. *Knowl Inf Syst (2012), 31*, 527–545.