

Rancangan Dasar: VIKOR untuk Peringkasan Otomatis

Preliminary Design : VIKOR for Automatic Summarization

Nelly Indriani Widiastuti
Universitas Komputer Indonesia
Jl. Dipati Ukur no. 112-116
Email : nelly.indriani@email.unikom.ac.id

Abstrak - Peringkasan dokumen otomatis secara umum dilakukan dengan cara menentukan kalimat yang dinilai cocok digunakan sebagai hasil ringkasan. Kalimat-kalimat dalam dokumen diberi bobot dan kemudian diurutkan berdasarkan bobot yang terbesar hingga yang terkecil. Hasil peringkasan mengambil sebagian atau prosentasi tertentu. Cara ini serupa dengan teknik penentuan keputusan berdasarkan beberapa kriteria, salah satunya adalah VIKOR. Metode ini umum digunakan dalam berbagai bidang bisnis, teknologi, dan pendidikan. Metode ini berfokus pada pemeringkatan berdasarkan kedekatan dengan solusi ideal. Artikel ini mendeskripsikan rancangan dasar peringkasan dokumen dengan menggunakan VIKOR. Kriteria yang digunakan adalah fitur ekstraksi kalimat. Proses VIKOR telah dapat menghasilkan urutan kalimat yang dapat dianggap sebagai hasil ringkasan.

Kata kunci : VIKOR, peringkasan otomatis, perankingan, multi kriteria, keputusan

Abstract - Automatic document summaries are generally done by determining which sentences are considered appropriate to be used as a summary result. The sentences in the document are weighted and then sorted according to the weight of the largest to the smallest. The summary results take some or a certain percentage. This method is similar to decision-making techniques based on several criteria, one of which is VIKOR. This method is commonly used in many areas of business, technology, and education. This method focuses on ranking based on proximity to the ideal solution. This article describes the basic draft document design using VIKOR. The criteria used are sentence extraction features. The VIKOR process has been able to generate a sequence of sentences that can be considered as a summary result.

Keyword : VIKOR, automatic summarization, ranking, multicriteria, decision

I. PENDAHULUAN

Peringkasan otomatis umumnya dilakukan dengan cara ekstraktif. Cara ini tidak berfokus pada pemahaman terhadap konten dokumen, tetapi dengan cara mengekstrak bagian terpenting berdasarkan fitur linguistik dan statistik sebagai kata yang mengandung informasi[1][2]. Penelitian terkait peringkasan otomatis selama ini menggunakan algoritma klasifikasi. Algoritma yang populer dalam sistem peringkasan otomatis diantaranya SVM, KNN atau Naïve Bayes dll[3]. Cara kerja algoritma klasifikasi pada peringkasan otomatis umumnya dengan pengenali pola data yang terjadi pada dataset (data training). Hal ini menyebabkan algoritma klasifikasi selalu bergantung pada dataset yang digunakan.

Dalam meringkas secara ekstraktif, suatu dokumen memiliki beberapa faktor, atau lebih dikenal sebagai fitur ekstraksi yang perlu dipertimbangkan. Untuk menyelesaikan masalah ini dibutuhkan suatu pendekatan. Pendekatan yang digunakan dalam penelitian ini adalah metode VIKOR. Pendekatan ini membutuhkan penilaian untuk menentukan kinerja dari tiap alternatif yang berkaitan dengan tiap kriteria[4]. Opricovic pertama kali mengusulkan VIKOR sebagai pendekatan MCDM dalam bidang civil engineering[5]. Kemudian, dikembangkan kembali dengan

menambahkan pendekatan fuzzy untuk water resource planning[6]. VIKOR juga digunakan untuk memilih pelatih pribadi[7]. Pada beberapa penelitian VIKOR juga digabungkan dengan Fuzzy AHP untuk menentukan prioritas perawatan trotoar[8]. VIKOR juga telah mengalami perkembangan dengan menambahkan fuzzy. Hal tersebut ada dalam penelitian pada Fuzzy VIKOR menggunakan entropy measure[9], dan untuk evaluasi terhadap kesehatan internet[10]. Abbas Mardani dkk. telah melakukan review terhadap 176 artikel penelitian yang dipublikasikan dari tahun 2004 sampai 2015 dalam jurnal ranking 83 tertinggi. Hasil penelitian tersebut menyatakan VIKOR dan fuzzy VIKOR adalah metode peringkat pertama, akan tetapi VIKOR jarang digunakan dalam bidang science[11].

Berdasarkan hal tersebut, maka dalam penelitian ini VIKOR digunakan untuk memutuskan kalimat-kalimat dalam suatu dokumen, dianggap sesuai sebagai hasil peringkasan. Keputusan itu mempertimbangkan beberapa fitur yang telah digunakan pada penelitian sebelumnya.

II. METODE DESAIN

Artikel ini merupakan rancangan awal yang akan menunjukkan bahwa tahapan dalam VIKOR dapat

digunakan untuk meringkas suatu dokumen. Dalam artikel ini dideskripsikan setiap tahap VIKOR sebagai proses peringkasan. Proses ini dianggap sebagai proses dalam mengambil keputusan berdasarkan banyak kriteria. Kriteria tersebut adalah fitur ekstraksi peringkasan yang digunakan pada penelitian sebelumnya. Dalam akhir artikel ini ditunjukkan hasil perangkaian yang diperoleh dari metode VIKOR. VIKOR termasuk jenis metode pembuat keputusan multi kriteria. Metode ini berfokus pada pemeringkatan dan pemilihan dari serangkaian alternatif karena adanya kriteria yang bertentangan. Hal tersebut menggunakan cara indeks peringkat multikriteria berdasarkan ukuran "kedekatan" dengan solusi "ideal"[12].

Metode dalam merancang penggunaan VIKOR untuk peringkasan otomatis dilakukan dengan cara mensimulasikan suatu kasus untuk diproses sehingga menghasilkan urutan kalimat yang dianggap sesuai sebagai peringkasan berdasarkan metode VIKOR.

Kasus yang digunakan dalam artikel ini adalah latar belakang suatu karya ilmiah bidang ilmu komputer. Dokumen tersebut akan diproses melalui tahap pra proses, ekstraksi fitur, terakhir penentuan urutan kalimat calon hasil ringkasan.

A. Pra proses

Sebagai rancangan awal, dalam paper ini akan disimulasikan proses peringkasan dokumen secara otomatis menggunakan metode VIKOR. Sebelum melalui proses peringkasan dokumen ini harus melalui tahap pre processing yaitu segmentasi kalimat, *case folding*, dan *filtering(stopword)*.

Tabel 1 adalah dokumen yang dijadikan contoh dalam artikel ini. Dokumen tersebut telah melalui proses pra proses segmentasi kalimat, case folding dan filtering.

Tabel 1. Contoh dokumen hasil pra proses

No	Kalimat
S1	deteksi manusia pada suatu citra digital dapat dilakukan dengan menggunakan metode dari dalal dan triggs pada tahun yaitu histograms of oriented gradients hog
S2	metode ini menghasilkan akurasi deteksi sebesar dan diakui sebagai metode yang robust untuk pendeteksian objek
S3	penerapan metode hog untuk deteksi mobil menghasilkan akurasi sebesar
S4	metode ini diawali dengan mengkonversi citra red green blue rgb menjadi citra grayscale kemudian dilanjutkan dengan menghitung nilai gradien citra
S5	hasil tahap penghitungan gradien piksel ini berupa gambar hasil deteksi tepi menggunakan filter konvolusi seperti sobel mask central difference mask atau diagonal mask
S6	pada citra yang banyak terdapat noise hasil perhitungan gradien masih kurang

maksimal sehingga dapat mempengaruhi akurasi pendeteksian

Tabel 1. Contoh dokumen hasil pra proses (lanjutan)

No	Kalimat
S7	beberapa riset telah mengembangkan metode ini salah satunya dengan menambahkan fitur image segmentation pada metode hog yang dilakukan oleh socarras dkk
S8	mereka menggabungkan teknik segmentasi sebelum tahap penghitungan gradien pada metode hog dan menghasilkan perkembangan akurasi deteksi sebesar
S9	berdasarkan penelitian oleh frei dan chen dihasilkan sebuah algoritma deteksi tepi baru yang disebut algoritma freichen
S10	metode ini menunjukkan pendeteksian tepi yang terbaik dilakukan oleh detektor tepi yang sederhana diikuti dengan penipisan dan menghubungkan proses untuk mengoptimalkan garis tepi

B. Fitur Peringkasan

Secara umum VIKOR membutuhkan kriteria-kriteria penilaian yang digunakan untuk mengambil keputusan (C_j). Dalam kasus peringkasan otomatis, kriteria tersebut diganti dengan fitur-fitur yang fungsinya serupa dengan kriteria penilaian. Telah banyak penelitian yang khusus membahas fitur dalam peringkasan. Secara umum penggunaan fitur sangat bergantung pada kebutuhan. Dalam paper ini fitur yang digunakan adalah posisi kalimat (F_1), kalimat positif (F_2), Kalimat inti (F_3), Keterkaitan kalimat dengan judul (F_4), panjang relatif kalimat (F_5), keterkaitan antar kalimat (F_6).

Fitur yang digunakan dalam rancangan awal VIKOR untuk peringkasan dokumen adalah fitur yang digunakan dalam penelitian M. Abdel Fattah dan R. Fujji[13]. Secara umum fitur-fitur tersebut biasa digunakan, namun rumus yang digunakan berbeda-beda, maka dalam artikel ini rumus masing-masing fitur yang digunakan ditulis ulang.

1. Posisi Kalimat (F_1)

Fitur ini digunakan karena diasumsikan kalimat pertama adalah kalimat yang paling penting. Kalimat pertama dan terakhir akan memiliki nilai tertinggi dibanding kalimat berikut lainnya[14]. Pada persamaan (1).

$$Score_{F_1}(S_i) = \max\left(\frac{1}{i}, \frac{1}{n-i+1}\right) \quad (1)$$

Dimana n adalah jumlah kalimat, i adalah posisi kalimat, S_i adalah kalimat ke- i dan $Score_{F_1}$ adalah nilai untuk fitur F_1 . Misalnya dalam suatu dokumen memiliki 10 kalimat. Dengan menggunakan persamaan (1) diperoleh hasil seperti pada **Tabel 2**. $Score_{F_1}(S_i)$ diperoleh dengan mengambil nilai tertinggi diantara satu per posisi kalimat terhadap kalimat pertama dan kalimat terakhir.

Tabel 2. Nilai fitur 1 (F1)

kalimat	$Score_{F1}(S_i)$
S1	1,0
S2	0,5
S3	0,33
S4	0,25
S5	0,2
S6	0,2
S7	0,25
S8	0,33
S9	0,5
S10	1

2. Kalimat Positif (F₂)

Kalimat positif atau *positive keyword* adalah kata yang sering muncul pada sebuah paragraf. Penilaian didasarkan pada persamaan (2):

$$Score_{F2}(S_i) = \sum_{i=1}^n \begin{cases} 0,1 & i \text{ kata kunci} \\ 0 & i \neq \text{ kata kunci} \end{cases} \quad (2)$$

Dimana S_i adalah jumlah kata terbanyak (keyword) dalam suatu kalimat. F_2 adalah fitur kata kunci yang diekstrak dari 10 kata terbanyak yang muncul dalam sebuah dokumen. Setelah kata kunci didapatkan, setiap token yang sama dengan kata kunci akan diberi nilai 0,1 dan yang tidak sama akan diberi nilai 0. Misalnya dalam suatu dokumen terdapat kata kunci seperti pada **Tabel 3**.

Tabel 3. Kata kunci

No	Kata/term	N
1	deteksi	8
2	metode	7
3	hasil	6
4	hog	4
5	akurasi	4
6	hitung	4
7	gradien	4
8	citra	3
9	tepi	3

Dokumen yang digunakan dalam analisis ini memiliki 10 kalimat (tabel 1) maka setiap kalimat akan dihitung jumlah bobot kata kunci sesuai dengan persamaan (2). Setiap kata pada S_1 dibandingkan dengan kata yang ada pada tabel 3. Dalam S_1 terdapat 4 kata yang sama yaitu {deteksi, citra, metode, hog}. Berdasarkan hal tersebut, maka S_1 memiliki nilai F_2 sebesar 0,4. Hasil penghitungan bobot dapat dilihat pada **Tabel 4**. Hal ini juga dilakukan pada semua kalimat dalam dokumen.

3. Kalimat Inti (F₃)

Kalimat ini adalah istilah untuk kata yang tumpang tindih antara kalimat satu dengan kalimat lain dalam dokumen. Fitur kalimat inti dapat dilihat pada persamaan (3). $Score_{F3}(S_i)$ diperoleh dengan menghitung jumlah kata yang sama di kalimat ke- i dengan kalimat ke- j . j adalah indeks untuk setiap kalimat lain selain i .

$$Score_{F3}(S_i) = \frac{|Kata \text{ di } S_i \cap kata \text{ di } S_j|}{|Kata \text{ di } S_i \cup kata \text{ di } S_j|} \quad (3)$$

Tabel 4. Contoh penghitungan F2

kalimat	Token	bobot	\sum bobot
S1	deteksi	0,1	0,4
	manusia	0	
	citra	0,1	
	digital	0	
	metode	0,1	
	dalal	0	
	triggs	0	
	histograms	0	
	oriented	0	
	gradients	0	
	hog	0,1	

Nilai inti S_1 terhadap kalimat lain seperti pada **Tabel 5**.

Tabel 6. Hasil penghitungan F3 untuk S_1

Kalimat	Bobot
S2	0,1905
S3	0,1579
S4	0,1154
S5	0,0333
S6	0,0952
S7	0,125
S8	0,12
S9	0,4545
S10	0,12

Hasil maksimum perhitungan pada **Tabel 6** adalah nilai yang digunakan sebagai nilai $Score_{F3}(S_1)$ yaitu 0,4545. Hal ini juga dilakukan pada kalimat lain dalam dokumen. Fitur F3 untuk kalimat lain dapat dilihat pada **Tabel 7**.

Tabel 7. Contoh Nilai F3

Kalimat	$Score_{F3}(S_i)$
S1	0,45
S2	0,31
S3	0,31
S4	0,22
S5	0,16
S6	0,23
S7	0,19
S8	0,26
S9	0,45
S10	0,27

4. Keterkaitan kalimat dengan judul (F₄)

Fitur ini menghitung kata yang tumpang tindih antara kalimat dengan judul dokumen. Fitur ini dihitung seperti pada persamaan (4). F_4 dihitung dengan menghitung jumlah kata yang sama antara kalimat S_i dengan kalimat judul, dibagi dengan jumlah semua kata di S_i dengan judul.

$$Score_{F4}(S_i) = \frac{|Kata \text{ di } S_i \cap kata \text{ di judul}|}{|Kata \text{ di } S_i \cup kata \text{ di judul}|} \quad (4)$$

Dokumen yang digunakan dalam artikel ini memiliki token judul seperti pada **Tabel 8**.

Tabel 8. Token Judul

No	Token
1	Implementasi
2	Metode
3	histograms
4	Of
5	Oriented
6	gradients
7	optimal
8	algoritma
9	frei-chen
10	deteksi
11	citra
12	manusia

Hasil penghitungan F4 untuk dokumen tersebut dapat dilihat pada **Tabel 9**.

Tabel 9. Hasil penghitungan F4

Kalimat	Bobot
S1	0,3333
S2	0,1905
S3	0,1053
S4	0,1538
S5	0,0333
S6	0,0952
S7	0,0833
S8	0,08
S9	0,1364
S10	0,16

5. Panjang relatif kalimat (F5)

Panjang kalimat relatif bertujuan untuk mengeliminasi kalimat yang terlalu pendek. Jenis kalimat tersebut biasanya tidak digunakan dalam peringkasan. Fitur ini dihitung pada persamaan (5) [15][16]. F5 dihitung dengan membagi jumlah kata dalam S_i dengan jumlah kata terbanyak dalam suatu dokumen. **Tabel 10**. adalah hasil perhitungan fitur panjang kalimat.

$$Score_{F5}(S_i) = \left| \frac{panjang(S_i)}{\max panjang(S_i)} \right| \quad (5)$$

Tabel 10. Hasil Perhitungan F5

kalimat	Panjang (Si)	Score _{F5} (S _i)
S1	11	0,6
S2	9	0,5
S3	7	0,4
S4	14	0,8
S5	18	1
S6	9	0,5
S7	12	0,7
S8	13	0,7
S9	10	0,5
S10	13	0,7
Max Si	18	

6. Keterkaitan antar kalimat (F6)

Fitur 6 (F6) mempertimbangkan kata yang tumpang tindih antar kalimat dengan kalimat lain dalam paragraf yang sama. Hal ini dapat dihitung dengan menggunakan persamaan (6) [17].

$$Score_{F6}(S_i) = \frac{\sum Sim(S_i, S_j)}{\max(\sum Sim(S_i, S_j))} \quad (6)$$

dimana,

$Sim(S_i, S_j)$ = kemiripan antara kalimat ke-i dan kalimat ke-j

$\sum Sim(S_i, S_j)$ = jumlah kemiripan antara kalimat ke-i dan kalimat ke-j

Fungsi $sim(S_i, S_j)$ diperoleh dari rumus jarak antar kalimat seperti pada rumus (7).

$$S_i, S_j = \frac{\sum_{t=1}^n W_{it} \times W_{jt}}{\sqrt{\sum_{t=1}^n W_i^2} \times \sqrt{\sum_{t=1}^n W_j^2}} \quad (7)$$

dimana,

$\sum_{t=1}^n W_{it} \times W_{jt}$ = jumlah perhitungan skalar dari pembobotan kalimat

$\sqrt{\sum_{t=1}^n W_i^2} \times \sqrt{\sum_{t=1}^n W_j^2}$ = jumlah perhitungan vektor dari pembobotan kalimat.

Contoh hasil nilai F6 dapat dilihat pada **Tabel 11**.

Setiap kalimat dalam dokumen memiliki nilai F6.

Tabel 11 Hasil Perhitungan F6

Kalimat	$\frac{\sum Sim(S_i, S_j)}{\max(\sum Sim(S_i, S_j))}$	Score _{F6} (S _i)
S1	1.0679	0.8016
	1.3321	
S2	1.3321	1
	1.3321	
S3	1.0707	0.8038
	1.3321	
S4	1.2311	0.9242
	1.3321	
S5	1.1729	0.8805
	1.3321	
S6	1.1331	0.8305
	1.3321	
S7	1.1331	0.8305
	1.3321	
S8	1.2331	0,9256
	1.3321	
S9	1.1315	0,8494
	1.3321	
S10	1.1251	0,8446
	1.3321	

III. HASIL DAN PEMBAHASAN

VIKOR berasal dari bahasa Serbia yaitu singkatan dari ViseKriterijumska Optimizacija I Kompromisno Resenje yang artinya pengambilan keputusan berdasarkan multi kriteria[5]. VIKOR berfokus pada pemeringkatan dan pemilahan satu set alternatif terhadap berbagai kriteria keputusan, atau mungkin bertentangan dan tidak dapat disepakati, dengan asumsi bahwa kompromi dapat diterima untuk menyelesaikan konflik. VIKOR bergantung pada fungsi agregat yang mewakili kedekatan dengan ideal, memperkenalkan indeks peringkat berdasarkan ukuran kedekatan dengan

solusi ideal dan metode ini menggunakan normalisasi linier untuk mengeliminasi satuan fungsi kriteria[12].

Pada bagian ini akan dibahas rancangan peringkasan dokumen teks menggunakan proses-proses dalam VIKOR. Tahapan dalam VIKOR telah banyak mengalami perkembangan. Dalam artikel ini sumber yang digunakan adalah hasil penelitian yang dilakukan oleh Alireza Anvari, Norzima Zulkifli & Omid Arghish. Mulai dari membuat matriks keputusan, normalisasi matriks keputusan, menentukan solusi yang dan buruk, menghitung ukuran utility dan regret, menghitung indeks VIKOR, dan terakhir merangking urutan keputusan[18].

A. Matriks Keputusan

Matriks keputusan adalah matriks yang berisi nilai atau bobot masing-masing kriteria yang digunakan untuk menentukan keputusan. Pada penelitian yang dilakukan oleh Eko Pramulanto dkk. matriks keputusan adalah bobot cara bayar, jumlah pelanggan, proteksi dan lama kontrak untuk memilih produk asuransi[19]. Pada penelitian yang menentukan *leanness manufacturing system* menggunakan atribut *cost*, *lead time*, *defect*, dan *value*[18]. Dalam penelitian yang dilakukan oleh opricovic dkk, menggunakan *investment cost*, *water supplay discharge*, *social impact*, dan *impact on monastery* yang dinilai oleh ahli[12].

Dalam artikel ini nilai kriteria yang digunakan adalah nilai fitur ekstraksi kalimat yang telah dihitung pada bagian metode. Pada **Tabel 12.** adalah contoh nilai fitur yang digunakan dalam pembahasan ini.

Tabel 12. Bobot Fitur

kali- mat	F_1	F_2	F_3	F_4	F_5	F_6
S_1	1,0	0,4	0,45	0,33	0,6	0,80
S_2	0,5	0,4	0,31	0,19	0,5	1
S_3	0,33	0,5	0,31	0,10	0,4	0,80
S_4	0,25	0,6	0,22	0,15	0,8	0,92
S_5	0,2	0,6	0,16	0,03	1	0,88
S_6	0,2	0,5	0,23	0,09	0,5	0,83
S_7	0,25	0,3	0,19	0,08	0,7	0,83
S_8	0,33	0,7	0,26	0,08	0,7	0,92
S_9	0,5	0,3	0,45	0,13	0,5	0,84
S_{10}	1	0,5	0,27	0,16	0,7	0,84

Dimana kolom akan dinyatakan sebagai fitur-fitur peringkasan yang telah dibahas pada bagian metode dalam artikel ini yaitu (F_1, F_2, \dots, F_6), dan baris menyatakan kalimat-kalimat dalam dokumen contoh pada **Tabel 1** yaitu (S_1, S_2, \dots, S_{10}). Berdasarkan tabel 12 maka matriks keputusan dapat dinyatakan sebagai berikut:

$$x = \begin{bmatrix} 1 & 0,4 & 0,5 & 0,3 & 0,6 & 0,8 \\ 0,5 & 0,4 & 0,3 & 0,2 & 0,5 & 1 \\ 0,3 & 0,5 & 0,3 & 0,1 & 0,4 & 0,8 \\ 0,3 & 0,6 & 0,2 & 0,2 & 0,8 & 0,9 \\ 0,3 & 0,6 & 0,2 & 0 & 1 & 0,9 \\ 0,2 & 0,5 & 0,2 & 0,1 & 0,5 & 0,8 \\ 0,2 & 0,3 & 0,2 & 0,1 & 0,7 & 0,8 \\ 0,3 & 0,7 & 0,3 & 0,1 & 0,7 & 0,9 \\ 0,5 & 0,3 & 0,5 & 0,1 & 0,5 & 0,8 \\ 1 & 0,5 & 0,3 & 0,2 & 0,7 & 0,8 \end{bmatrix}$$

B. Normalisasi Matriks

Umumnya kriteria yang digunakan dalam mengambil keputusan, tidak selalu dalam range yang sama. Dalam penentuan penerima beasiswa, yang menjadi kriteria indeks prestasi (1-4), semester (1-14), daya listerik (450-1300)[20]. Opricovic dkk melakukan normalisasi menggunakan logika fuzzy[12]. Persamaan (8) adalah fungsi untuk normalisasi matriks

$$f_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m (x_{ij})^2}} \quad i=1,2,\dots,m; j=1,2,\dots,n \quad (8)$$

Dimana x_{ij} adalah elemen matrik pada baris i kolom j , m adalah jumlah fitur yaitu 6. Untuk selanjutnya matriks kriteria akan disajikan dalam bentuk tabel. Pada **Tabel 13** adalah hasil normalisasi.

Tabel 13. Hasil normalisasi

	F_1	F_2	F_3	F_4	F_5	F_6
S_1	0,64	0,13	0,15	0,07	0,26	0,50
S_2	0,19	0,13	0,08	0,03	0,20	0,84
S_3	0,10	0,23	0,10	0,01	0,14	0,62
S_4	0,04	0,26	0,04	0,03	0,48	0,74
S_5	0,03	0,24	0,02	0,06	0,69	0,75
S_6	0,04	0,23	0,05	0,01	0,23	0,65
S_7	0,05	0,08	0,03	0,01	0,43	0,71
S_8	0,08	0,36	0,06	0,01	0,36	0,73
S_9	0,20	0,08	0,17	0,01	0,21	0,65
S_{10}	0,63	0,20	0,05	0,03	0,34	0,56

C. Solusi positif ideal dan solusi negatif ideal

Tahap ini adalah menentukan solusi ideal positif, dengan mencari nilai maksimum f_{ij}^* yang terbaik dan f_{ij}^- yang (tiap alternatif memiliki kriteria masing-masing). Dalam penentuan penerima beasiswa, penentuan A^* dan A^- ditentukan dengan nilai 1 untuk kriteria yang mengharuskan semakin tinggi semakin baik. Misalnya kriteria IPK. Selain itu kriteria lain akan bernilai -1 karena semakin kecil semakin baik. Contohnya rekening listerik, daya dan semester[20]. Dalam artikel ini menggunakan persamaan (9) dan (10).

$$A^* = \{ \max f_{ij} | i = 1, 2, \dots, m \} = \{ f_1^*, f_2^*, f_3^*, \dots, f_n^* \} \quad (9)$$

$$A^- = \{ \min f_{ij} | i = 1, 2, \dots, m \} = \{ f_1^-, f_2^-, f_3^-, \dots, f_n^- \} \quad (10)$$

f_{ij} dalam artikel ini adalah nilai fitur.

Pada **Tabel 14**, A^* diperoleh dengan cara mencari nilai maksimum dari nilai F_1 sd F_6 untuk setiap kalimat (S_1 sd S_{10}). Contohnya berdasarkan tabel 13, kolom F_1 menunjukkan nilai terbesar dari $S_1 - S_{10}$ adalah 0,64. Sedangkan A^- adalah nilai terkecil dalam kolom F_1 yaitu 0,03.

Tabel 14. Nilai solusi ideal dan solusi buruk

	F_1	F_2	F_3	F_4	F_5	F_6
A^*	0,64	0,36	0,17	0,07	0,69	0,84
A^-	0,03	0,08	0,02	0,01	0,14	0,50

D. Utility dan Regret Measure

Menghitung *utility measure* dan *regret measure* seperti pada persamaan (11) dan (12). Menghitung S_i menekankan utilitas kelompok maksimum, sedangkan menghitung R_i menekankan pemilihan minimum di antara penyesalan individu maksimum[18].

- i. Nilai R_i (*Regret measure*)

$$R_i = \text{Max} \left(W_j \frac{(f_{ij}^+ - f_{ij})}{(f_{ij}^+ - f_{ij}^-)} \right) \quad (11)$$

- ii. Nilai S_i (*Utility measure*)

$$S_i = \sum_{j=1}^n \left(W_j \frac{(f_{ij}^+ - f_{ij})}{(f_{ij}^+ - f_{ij}^-)} \right) \quad (12)$$

Dalam tahap ini dibutuhkan pemberian bobot oleh pakar (W_j) yang biasa memutuskan keputusan terkait. Dalam artikel ini bobot untuk keenam fitur atau kriteria yang digunakan secara berturut-turut adalah F_1 sd F_6 (0,2 ; 0,1 ; 0,1 ; 0,3 ; 0,1 ; 0,1 ; 0,2}. f_{ij}^+ atau A^* adalah nilai solusi ideal positif, f_{ij} nilai pada tabel 13 sedangkan f_{ij}^- atau A^- adalah nilai solusi ideal negatif. Hasil penghitungan nilai utility dan Regret dapat dilihat pada **Tabel 15**.

Tabel 15. Nilai Utility dan Regret

	<i>utility</i>	<i>regret</i>
	S_i	R_i
S_1	0,38	0,20
S_2	0,56	0,18
S_3	0,79	0,29
S_4	0,60	0,19
S_5	0,42	0,20
S_6	0,81	0,29
S_7	0,80	0,29
S_8	0,68	0,30
S_9	0,73	0,29
S_{10}	0,57	0,20

E. Indeks VIKOR

Setiap alternatif keputusan ditentukan masing-masing indeks VIKOR. Dalam artikel ini alternatif keputusan adalah kalimat-kalimat dalam dokumen. Indeks VIKOR dihitung menggunakan persamaan (13).

$$Q_i = \left[V \frac{(S_i - S^-)}{(S^+ - S^-)} \right] + \left[(1 - V) \frac{R_i - R^-}{R^+ - R^-} \right] \quad (13)$$

Dimana V atau dalam sumber lain adalah $\alpha \in \{0,1\}$ umumnya dipilih 0,5[18]. **Tabel 16** adalah hasil penghitungan persamaan (13).

Tabel 16. Nilai Indeks VIKOR

Qi	
S_1	0,96
S_2	0,96
S_3	0,94
S_4	0,88
S_5	0,86
S_6	0,32
S_7	0,30
S_8	0,21
S_9	0,14
S_{10}	0,10

F. Pemeringkatan Bobot VIKOR

Berdasarkan nilai (S_i, R_i, Q_i) pemeringkatan bobot VIKOR dilakukan secara menurun terhadap Q_i . Dalam beberapa penelitian, keputusan dapat memiliki alternatif adalah jika dua kondisi pada persamaan (14) dan (15) berikut dipenuhi.

$$(Q(A_2) - Q(A_1)) \geq \left(\frac{1}{(n-1)} \right) \quad (14)$$

$$(Q(A_m) - Q(A_1)) < \left(\frac{1}{(n-1)} \right) \quad (15)$$

Indeks VIKOR yang terbaik adalah alternatif keputusan yang memiliki indeks VIKOR terkecil.

Tabel 17. Peringkat VIKOR

	Qi	rangking
S_6	0,96	1
S_7	0,96	2
S_3	0,94	3
S_9	0,88	4
S_8	0,86	5
S_4	0,32	6
S_{10}	0,30	7
S_2	0,21	8
S_5	0,14	9
S_1	0,10	10

Dalam penelitian ini keputusan alternatif tidak digunakan karena kebutuhan dalam peringkasan adalah urutan nilai ekstraksi kalimat yang terbaik. Berdasarkan **Tabel 17.** dan kompresi 50% maka hasil ringkasan dapat dilihat pada **Tabel 18.**

Tabel 18. Hasil ringkasan menggunakan VIKOR

No	Kalimat	Rang king
S ₆	Pada citra yang banyak terdapat noise, hasil perhitungan gradien masih kurang maksimal sehingga dapat mempengaruhi akurasi pendeteksian	1
S ₇	Beberapa riset telah mengembangkan metode ini, salah satunya dengan menambahkan fitur Image Segmentation pada metode HOG yang dilakukan oleh Socarras dkk [3]	2
S ₃	Penerapan metode HOG untuk deteksi mobil menghasilkan akurasi sebesar 99,10% [2]	3
S ₉	Berdasarkan penelitian oleh Frei dan Chen [4], dihasilkan sebuah algoritma deteksi tepi baru yang disebut algoritma Frei-Chen	4
S ₈	Mereka menggabungkan teknik segmentasi sebelum tahap penghitungan gradien pada metode HOG dan menghasilkan perkembangan akurasi deteksi sebesar 4,47%	5

IV. KESIMPULAN

Rancangan dasar penggunaan VIKOR dalam sistem peringkasan dokumen secara otomatis telah disampaikan dalam artikel ini. Kebutuhan peringkasan otomatis terhadap pemeringkatan kalimat-kalimat dalam dokumen dapat dipenuhi oleh cara VIKOR dalam menentukan keputusan dengan banyak kriteria. Hasil peringkasan telah diperoleh, sehingga menunjukkan bahwa VIKOR dapat digunakan sebagai salah satu cara untuk membuat sistem peringkasan otomatis.

Pengembangan selanjutnya adalah mengukur kinerja VIKOR dalam bidang teks mining khususnya peringkasan. Hal ini dilakukan dengan cara membandingkan VIKOR dengan metode lain untuk mengukur seberapa baik akurasi peringkasan yang dihasilkan.

DAFTAR PUSTAKA

[1] F. Kyoomarsi, H. Khosravi, E. Eslami, P. K. Dehkordy, and A. Tajoddin, "Optimizing Text Summarization Based on Fuzzy Logic," *Seventh*

IEEE/ACIS Int. Conf. Comput. Inf. Sci. (icis 2008), vol. 5, no. January 2016, pp. 347–352, 2008.

- [2] N. Bhatia and A. Jaiswal, "Trends in Extractive and Abstractive Techniques in Text Summarization," *Int. J. Comput. Appl.*, vol. 117, no. 6, pp. 21–24, 2015.
- [3] R. Jindal, "Techniques for text classification: Literature review and current trends," *Webology*, vol. 12, no. 2, pp. 1–28, 2015.
- [4] M. S. Kuo, G. S. Liang, and W. C. Huang, "Extensions of the multicriteria analysis with pairwise comparison under a fuzzy environment," *Int. J. Approx. Reason.*, vol. 43, no. 3, pp. 268–285, 2006.
- [5] S. Opricovic, *Multicriteria Optimization of Civil Engineering Systems*. Belgrade, 1998.
- [6] S. Opricovic, "Fuzzy VIKOR with an application to water resources planning," *Expert Syst. Appl.*, vol. 38, no. 10, pp. 12983–12990, 2011.
- [7] M. F. El-santawy, "A VIKOR Method for Solving Personnel Training," *Int. J. Comput. Sci.*, vol. 1, no. 2, pp. 9–12, 2012.
- [8] P. Babashamsi, A. Golzadfar, N. I. M. Yusoff, H. Ceylan, and N. G. M. Nor, "Integrated fuzzy analytic hierarchy process and VIKOR method in the prioritization of pavement maintenance activities," *Int. J. Pavement Res. Technol.*, vol. 9, no. 2, pp. 112–120, 2016.
- [9] A. Shemshadi, H. Shirazi, M. Toreihi, and M. J. Tarokh, "A fuzzy VIKOR method for supplier selection based on entropy measure for objective weighting," *Expert Syst. Appl.*, vol. 38, no. 10, pp. 12160–12167, 2011.
- [10] E. Afful-Dadzie, S. Nabareseh, and Z. K. Oplatková, "Fuzzy Vikor Approach: Evaluating Quality of Internet Health Information," *2014 Fed. Conf. Comput. Sci. Inf. Syst. FedCSIS 2014*, vol. 2, no. Mcdm, pp. 183–190, 2014.
- [11] A. Mardani, E. K. Zavadskas, K. Govindan, A. A. Senin, and A. Jusoh, "VIKOR technique: A systematic review of the state of the art literature on methodologies and applications," *Sustain.*, vol. 8, no. 1, pp. 1–38, 2016.
- [12] S. Opricovic, "Multicriteria optimization of civil engineering systems," *Fac. Civ. Eng. Belgrade*, vol. 2, pp. 5–21, 1998.
- [13] M. Abdelfattah and R. Fuji, "Automatic text summarization," *World Acad. Sci. Eng. Technol.*, vol. 37, no. 17, pp. 192–195, 2008.
- [14] K. Nandhini and S. R. Balasundaram, "Extracting easy to understand summary using differential evolution algorithm," *Swarm Evol. Comput.*, vol. 16, no. Swarm and Evolutionary Computation, pp. 19–27, 2014.
- [15] Y. S. Ahda, "PERINGKASAN DOKUMEN BAHASA INDONESIA," Bogor, 2015.
- [16] H. P. Luhn, "The automatic creation of literature abstracts," *IBM*, vol. 2, p. 159–165.,

- 1958.
- [17] I. Mani, *Automatic Summarization*. John Benjamins, 2001.
- [18] A. Anvari, N. Zulkifli, and O. Arghish, "Application of a modified VIKOR method for decision-making problems in lean tool selection," *Int. J. Adv. Manuf. Technol.*, vol. 71, no. 5–8, pp. 829–841, 2014.
- [19] E. C. Pramulanto, M. Imrona, E. Darwiyanto, F. Informatika, and U. Telkom, "Aplikasi Pendukung Keputusan untuk Pemilihan Produk Asuransi dengan Metode Entropy dan Vikor pada AJB Bumiputera 1912 Jepara," 1912.
- [20] S. P. Lengkong *et al.*, "Implementasi Metode VIKOR untuk Seleksi Penerima Beasiswa," in *PROCEEDINGS OF THE 7TH NATIONAL CONFERENCE ON INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING*, 2015, pp. 107–112.

BIODATA PENULIS



Nelly Indriani Widiastuti S.Si., M.T. bekerja sebagai dosen tetap di program studi Teknik Informatika. Matakuliah yang diampu adalah Kecerdasan Buatan, Matematika Diskrit, dan Logika Matematika.

Sebagai koordinator kelompok keilmuan Computer Science, penelitian yang dilakukan terkait dengan Machine Learning, Text Mining, Information Extraction, dan Artificial Intelligence.