ISSN : 2407 - 3911

PEMBOBOTAN KATA BERDASARKAN KLASTER PADA OPTIMISASI COVERAGE, DIVERSITY DAN COHERENCE UNTUK PERINGKASAN MULTI DOKUMEN

Ryfial Azhar[1], Muhammad Machmud[2], Hanif Affandi Hartanto[3]
Agus Zainal Arifin[4], Diana Purwitasari[5]
[1,2,3,4,5] Jurusan Teknik Informatika, Institut Teknologi Sepuluh Nopember
Kampus ITS Keputih, Sukolilo, Surabaya 60111, Jawa Timur, Indonesia
Email: [1]ryfial.azhar14@mhs.if.its.ac.id, [2]machmud14@mhs.if.its.ac.id, [3]hanf_aff@apps.ipb.ac.id,
[4]agusza@cs.its.ac.id, [5]diana@if.its.ac.id

Abstrak

Peringkasan yang baik dapat diperoleh dengan coverage, diversity dan coherence yang optimal. Namun, terkadang sub-sub topik yang terkandug dalam dokumen tidak terekstrak dengan baik, sehingga keterwakilan setiap sub-sub topik tersebut tidak ada dalam hasil peringkasan dokumen. Pada paper ini diusulkan metode baru pembobotan kata berdasarkan klaster pada optimisasi coverage, diversity dan coherence untuk peringkasan multi-dokumen. Metode optimasi yang digunakan ialah self-adaptive differential evolution (SaDE) dengan penambahan pembobotan kata berdasarkan hasil dari pembentukan cluster dengan metode Similarity Based Histogram Clustering (SHC). Metode SHC digunakan untuk mengklaster kalimat sehingga setiap sub-topik pada dokumen bisa terwakili dalam hasil peringkasan. Metode SaDE digunakan untuk mencari solusi hasil ringkasan yang memiliki tingkat coverage, diversity, dan coherence paling tinggi. Uji coba dilakukan pada 15 topik dataset Text Analysis Conference (TAC) 2008. Hasil uji coba menunjukkan bahwa metode yang diusulkan dapat menghasilkan ringkasan skor ROUGE-1 sebesar 0.6704, ROUGE-2 sebesar 0.2051, ROUGE-L sebesar 0.6271 dan ROUGE-SU sebesar 0.3951.

Kata kunci :

peringkasan multi dokumen, similarity based histogram clustering, coverage, diversity, coherence

Abstract

Good summary can be obtained with optimizing coverage, diversity, and coherence. Nevertheless, sometime sub-topics wich is contained in the document is not extracted well, so that the representation of each sub-topic is appear in docment summarizarion result. In this paper, we propose new of term weighting based on cluster in optimizing coverage, diversity, and coherence for multi-document summarization. Optimization method which is used is self-adaptive differential evolution (SaDE) with additional term weighting based on clustering result with Similarity Based Histogram Clustering (SHC). SHC is used to cluster sentence so that every sub-topic in the document can be represented in summarization result. SaDE is used to search summarization result solution which has high coverage, diversity, and coherence level. Experiment is done on 15 topics in Text Analysis Conference (TAC) 2008 dataset. Experimental results show that this proposed method can produce summarization score ROUGE-1 0.6704, ROUGE-2 0.2051, ROUGE-L 0.6271 and ROUGE-SU 0.3951.

Keywords:

multy-document summarization, similarity based histogram clustering, coverage, diversity, coherence.

I. PENDAHULUAN

Ketersediaan dokumen secara *online* menyediakan informasi yang tidak terbatas. Namun, hal ini menyebabkan kesulitan mencari informasi yang relevan dan sesuai dengan kebutuhan kita. Banyak informasi yang memiliki tema yang sama berada di beberapa dokumen yang berbeda namun dan mengakibatkan redundansi informasi. Berdasarkan masalah di atas maka dibutuhkan sistem untuk meringkas dokumen - dokumen ini.

Peringkasan dokumen secara otomatis ialah proses peringkasan dokumen yang mengekstrak informasi-informasi penting yang mewakili semua informasi yang ada pada dokumen asli secara relevan. Berdasarkan jumlah dokumen yang diproses, peringkasan dokumen dapat dikategorikan menjadi dua yaitu peringkasan dokumen tunggal dan multi-dokumen. Peringkasan dokumen tunggal ialah peringkasan yang hanya memproses satu dokumen untuk diringkas, sedangkan peringkasan multi-dokumen ialah peringkasan yang memproses lebih dari satu dokumen dengan topik yang sama untuk diringkas.

Hasil peringkasan yang baik ialah peringkasan mengandung tiga faktor yaitu cakupan pembahasan (*coverage*) yang luas, tingkat keberagaman (*diversity*), dan keterhubungan antarkalimat (*coherence*) yang tinggi (Alguliev, Aliguliyev, & Isazade, 2013). Hasil ringkasan yang memiliki *coverage* yang tinggi merupakan ringkasan yang mengandung seluruh informasi dari dokumen asal. Ringkasan yang tinggi *diversity*-nya ialah ringkasan yang tidak mengandung informasi yang berulang (*redundant*). Keterkaitan masing - masing kalimat pada hasil ringkasan menandakan peringkasan yang memiliki *coherence* yang tinggi.

Penelitian mengenai peringkasan multi-dokumen yang memperhatikan *coverage*, *diversity* dan *coherence* telah dilakukan oleh penelitian sebelumnya (Umam, Putro, & Pratamasunu, 2015), dimana dalam penelitian ini sistem optimasi dengan menggunakan algoritma *Self-adaptive Differential Evolution* (SaDE) diterapkan untuk meningkatkan ketiga faktor tersebut. Selanjutnya, Algoritma pengurutan kalimat yang menggunakan pendekatan *topical closeness* juga diintegrasikan ke dalam tiap iterasi algoritma SaDE untuk meningkatkan koherensi antarkalimat hasil ringkasan.

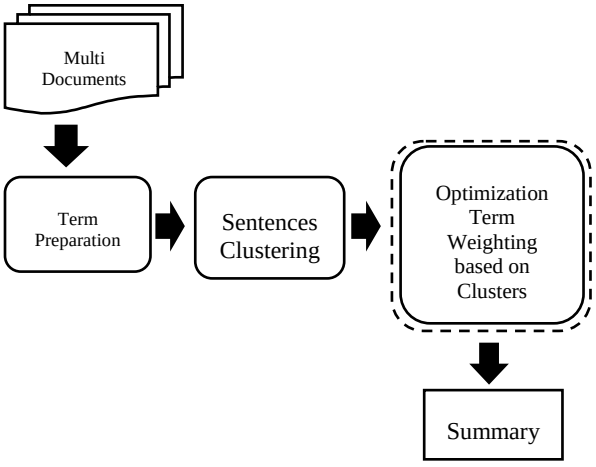
Penelitian lainnya mengenai peringkasan multi-dokumen dilakukan dengan metode *clustering* telah dilakukan dalam beberapa penelitian (Lukmana, Swanjaya, Kurniawardhani, Arifin, & Purwitasari, 2014), (Pasnur, Santika, & Syaifuddin, 2014), (Sarkar, 2009) (Suputra, Arifin, & Yuniarti, 2013) pengklasteran kalimat dilakukan untuk meningkatkan *coverage* pada hasil peringkasan. Selain itu, pengklasteran kalimat bertujuan untuk mengentahui sub-sub topik yang terkandung dalam dokumen.

Proses untuk menghasilkan peringkasan yang baik dapat diperoleh tidak hanya dengan *coverage*, *diversity* dan *coherence* yang optimal. Namun, terkadang sub-sub topik yang terkandung dalam dokumen tidak terekstrak dengan baik, sehingga keterwakilan setiap sub-sub topik tersebut tidak ada dalam hasil peringkasan dokumen.

Pada paper ini dusulkan metode pembobotan kata berdasarkan klaster pada optimisasi *coverage*, *diversity* dan *coherence* untuk peringkasan multi-dokumen.

II. METODE

Pada bagian ini dijelaskan secara detil tahapan yang dilakukan sehingga menghasilkan ringkasan. Sesuai Gambar 1, secara garis besar ada tiga tahapan utama yang digunakan dalam penelitian ini, yakni *term preparation*, *sentences clustering*, dan proses pembentukan ringkasan dengan optimisasi. Tahap *term preparation* berkaitan dengan tahap ekstraksi kalimat dan kata dari dokumen sampai dengan pembentukan document-term vector dan pembobotannya. *Sentences clustering* berperan dalam mengelompokkan kalimat dari semua dokumen ke dalam klaster-klaster sesuai dengan kemiripan sub topik masing-masing kalimat. Proses optimisasi dengan algoritma SaDE bertujuan untuk menghasilkan ringkasan berdasarkan hasil dari pengelompokkan kalimat pada proses klasterisasi. Hasil akhir dari proses ini ditentukan melalui nilai *fitness* tertinggi pada *coverage*, *diversity* dan *coherence* hasil peringkasan dengan pembobotan kata berdasarkan klaster-klaster yang telah terbentuk.



Gambar 1. *Flowchart* Metode usulan

II.1 Term Preparation

Term Preparation adalah tahapan untuk menyiapkan data yang digunakan pada proses utama. Tahapan ini bertujuan untuk menyiapkan data yang digunakan pada tahapan selanjutnya. Tahap *term preparation* ini terdiri dari proses pengambilan kalimat

dari dokumen asal, proses ekstraksi term, *stopword removal*, *stemming*, dan pembobotan. Pada tahapan ini dihasilkan matriks dokumen-term, dan matriks dokumen-*sentence* dan bobot masing-masing.

II.2 Clustering

Proses *clustering* bertujuan untuk mengelompokkan kalimat-kalimat yang memiliki tingkat kemiripan tinggi ke dalam *cluster-cluster* tertentu. Pada tahap ini digunakan metode *Similarity Based Histogram Ratio Clustering* (SHC). Metode ini digunakan karena pendekatan dalam metode ini adalah *cluster similarity histogram* yang menjamin tiap *cluster* agar tetap *coherent*. Untuk menghitung tingkat kemiripan kalimat pada proses SHC ini menggunakan penelitian sebelumnya (Song & Park, 2004) yaitu dengan pendekatan kemiripan semantik dengan metode *Latent Semantic Indexing* (LSI). Metode LSI mampu mengidentifikasi hubungan semantik antar term berdasarkan pola dan hubungan antara istilah dan konsep-konsep yang terkandung dalam koleksi teks (Wahib, Pasnur, Santika, & Arifin, 2015).

Untuk menentukan kemiripan semantik dengan LSI dibutuhkan proses *Singular Value Decomposition* (SVD) terhadap matriks X yang memetakan term dan kalimat. Matriks X akan didekomposisi menjadi tiga buah matriks U, ∑, dan V, sesuai dengan Persamaan 1.

$$X = U.\sum.V^T. \quad \ldots [1]$$

Dengan U dan V merupakan matriks *singular* kiri dan kanan, sedangankan matriks ∑ merupakan matriks diagonal yang menunjukkan nilai *singular*.

Pengukuran nilai *similarity* yang digunakan adalah *cosine similarity*, yaitu perhitungan tingkat kemiripan berdasar pada besar sudut kosinus antara dua vektor sesuai Persamaan 2 :

$$sim(A,B) = \cos(\theta) = \frac{A.B}{\|A\|\|B\|} \quad \ldots [2]$$

Dengan θ adalah sudut antara vektor *A* dan vektor *B*.

Proses *clustering* kalimat adalah bagian yang penting dalam sistem peringkasan otomatis karena setiap topik dalam set dokumen harus diidentifikasi secara tepat untuk menemukan kemiripan dalam semua dokumen sehingga menjamin *coverage* yang baik (Sarkar, 2009). Jika kalimat-kalimat dikelompokkan ke dalam sejumlah *cluster* yang telah ditentukan, *cluster* mungkin tidak koheren karena beberapa kalimat bisa saja terpaksa menjadi salah satu anggota *cluster* meskipun seharusnya tidak. *Cluster-cluster* tidak koheren mungkin mengandung unit-unit teks yang terduplikasi pada *cluster* yang berbeda dan menyebabkan pemilihan kalimat menjadi redundan untuk ringkasan. Sebaliknya, jika *cluster* sangat ketat, sebagian besar *cluster* menjadi singletons. Dengan demikian, harus dipilih metode *clustering* yang menjamin koherensi *cluster*. Pada paper ini digunakan algoritma (*Similarity Based Histogram Ratio*) SHC yang diadopsi dari (Hammouda & Kamel, 2003). SHC dipilih karena SHC menggunakan pendekatan *cluster similarity histogram* yang berguna untuk menjaga agar cakupan *cluster*.

Setiap *cluster* yang terbentuk harus memiliki koherensi yang baik pula. Hal tersebut untuk mencegah adanya kalimat-kalimat yang menginformasikan hal yang sama pada saat pemilihan kalimat penyusun ringkasan. Kualitas dari suatu *similarity histogram* yang merepresentasikan koherensi *cluster* ditentukan dengan menghitung rasio *similarity* yang berada diatas *threshold* dengan total jumlah *similarity* yang ada. Rasio dari histogram yang tinggi mencerminkan koherensi yang tinggi pula (Sarkar, 2009).

Misalkan pada suatu *cluster* terdapat *k* buah kalimat, maka banyaknya similarity kalimat yang ada pada *cluster* tersebut sebanyak *k(k+1)/2*. Sim={sim_1, sim_2, sim_3, …, sim_p} adalah kumpulan dari pasangan similarity antar kalimat, dengan nilai *p=k(k+1)/2*. Similarity histogram dari *cluster* dinotasikan dengan $H=\{h_1, h_2, h_3, ..., h_{nb}\}$. Jumlah dari bin yang ada pada suatu histogram dinotasikan dengan nb sedangkan jumlah similarity kalimat yang ada pada bin ke-i dinotasikan dengan h_i. Fungsi untuk menghitung nilai h_i ditunjukkan pada Persamaan 3.

$$h_i = count\ (sim) \quad \ldots [3]$$

Untuk $sim_{li} \leq sim_j \leq sim_{ui}$, dengan sim_{li} ialah batas minimum *similarity* pada bin ke-i sedangkan sim_{ui} ialah batas maksimum *similarity* pada bin ke-i.

Histogram ratio (HR) dari suatu *cluster* dapat dihitung dengan Persamaan 4.

$$HR = \frac{\sum_{i=1}^{n_b} h_i}{\sum_{j=1}^{n_b} h_j} \quad \ldots [4]$$

$$T = \lfloor S_T * n_b \rfloor \quad \ldots [5]$$

ST adalah *similarity threshold*. Persamaan 5 menunjukkan jumlah bin yang sesuai dengan similarity *threshold* yang dinotasikan dengan T.

Penambahan anggota baru yang buruk pada suatu *cluster* pada setiap tahap akan berpengaruh terhadap kualitas *cluster*, dalam hal ini akan menurunkan nilai HR *cluster* tersebut. Untuk mengantisipasi hal tersebut maka harus ditetapkan HRmin, yakni *histogram ratio minimum* yang digunakan sebagai batas minimum pada penambahan anggota *cluster* baru. Detil algoritma SHC diperlihatkan pada *pseudocode* pada Gambar 2.

```
1  N <- Empty List {Cluster List}
2  for each sentence s do
3      for each cluster c in N do
4          HRold = HRC
5          Simulate adding s to c
6          HRnew = HRC
7          if(HRnew >= HRold) OR
8              ((HRnew >= HRmin) AND
9              (HRold - HRnew < ε )) then
10             Add s to c
11             exit
12         end if
13     end for
14     if s was not added to any cluster then
15         Create a new cluster c
16         Add s to c
17         Add c to N
18     end if
19 end for
```

Gambar 2. *Pseudocode Clustering* Kalimat

Metode SHC berjalan secara bertahap sesuai Gambar 2 HRold adalah nilai *histogram ratio* pada suatu *cluster* sebelum penambahan kalimat baru kedalam *cluster* tersebut, sedangkan HRnew adalah *histogram ratio* yang diperoleh setelah sebuah kalimat dijadikan anggota dari suatu *cluster* yang diujikan. HRmin adalah nilai minimum dari *histogram ratio* pada *cluster* tertentu. Parameter ε (epsilon) digunakan sebagai *threshold* selisih antara HRold dengan HRnew. SHC akan menguji setiap kalimat dimana *cluster silmilarity histogram* dari setiap *cluster* dihitung sebelum dan sesudah melakukan simulasi penambahan kalimat baru pada suatu *cluster*.

Pada saat membentuk *cluster* baru nilai *histogram ratio* disimpan pada HRC yang dihitung menggunakan Persamaan 3. Pada proses pembentukan *cluster*, HRold dan HRnew dibandingkan. Jika nilai HRnew lebih atau sama dengan nilai HRold, maka kalimat tersebut ditambahkan ke dalam *cluster* uji, atau jika nilai HRnew lebih rendah dari nilai HRold namun nilai HRnew masih berada diatas HRmin dan selisih antara HRold dengan HRnew tidak lebih dari ε maka kalimat juga ditambahkan sedangkan selain itu kalimat tidak ditambahkan. Jika suatu kalimat tidak mendapatkan *cluster* setelah diuji dengan semua *cluster*, maka sebuah *cluster* baru dibentuk dan kalimat tersebut menjadi anggotanya.

II.3 Optimisasi

Proses optimisasi digunakan untuk membentuk ringkasan berdasarkan dokumen yang telah terkelompok. Optimisasi ini bertujuan untuk menentukan nilai *coverage*, *diversity*, dan *coherence* dengan tetap mengacu pada dokumen hasil *clustering* (dokumen *cluster*). Peringkasan yang baik bukan hanya memiliki *coverage*, *diversity* dan *coherence* yang optimal. Peringkasan yang baik juga harus mampu mewakili sub-sub topik yang terdapat dalam semua dokumen asal. Apabila sub-sub topik yang terkandung dalam masing-masing dokumen tidak terekstrak dengan baik, maka keterwakilan setiap sub-sub topik tersebut tidak terdapat dalam hasil peringkasan.

Untuk menjaga agar keterwakilan sub-sub topik pada semua dokumen asal terwakili, pada paper ini diusulkan metode pembobotan *Term Frequency Inverse Class Frequency* (TF-ICF). Pembobotan TF-ICF suatu *term* pada suatu dokumen *cluster doc* tergantung pada dua hal, jumlah *term* pada dokumen *cluster doc*, dan banyaknya dokumen *cluster* lain yang mengandung *term* tersebut. Semakin banyak jumlah *term* pada dokumen *cluster doc* maka semakin besar nilai bobot *term* tersebut, sebaliknya semakin sedikit jumlah *term* pada dokumen *cluster doc* maka semakin kecil nilai bobot *term* tersebut. Semakin banyak dokumen *cluster* yang mengandung *term* tersebut, maka semakin kecil nilai bobot *term* tersebut, Sebaliknya semakin sedikit dokumen *cluster* yang mengandung *term* tersebut, maka semakin besar nilai bobot *term* tersebut

Notasi pembobotan TF-ICF yang digunakan pada paper ini sesuai dengan Persamaan 6 dan Persamaan 7. $W_{term,doc}$ menyatakan bobot dari *term* pada dokumen *cluster* asal *doc*, $tf_{term,doc}$ menyatakan banyaknya jumlah *term* pada dokumen *cluster* tertentu *doc*, icf_{term} menyatakan *Inverse Class Frequency term*, yang menunjukkan pentingnya suatu *term* pada semua dokumen *cluster* asal dibandingkan dengan dokumen *cluster* tertentu *doc*.

$$icf_{term,doc} = \log\left(1 + \frac{N}{N_{term,doc}}\right) \quad \dots [6]$$

$$W_{term,doc} = tf_{term,doc} \times icf_{term,doc} \quad \dots [7]$$

Pembobotan TF-ICF ini kemudian digunakan pada proses optimisasi *coverage*, *diversity*, dan *coherence*. Nilai *coverage* hasil peringkasan menunjukkan perbandingan cakupan isi ringkasan terhadap isi dalam dokumen *cluster* asal. Hal ini dapat dihitung dengan menghitung *similarity* antara *center* isi dalam dokumen *cluster* asal dengan *center* kandidat ringkasan, dimana *center* suatu dokumen adalah rata-rata bobot semua term dalam dokumen (Alguliev, Aliguliyev, & Isazade, 2013). Nilai *coverage* juga turut mempertimbakan *similarity* antara dokumen *cluster* asal dengan tiap kalimat hasil peringkasan sesuai persamaan 8.

$$f_{coverage} = \sum_{i=1}^{n} sim(O_i, O_{sum}) x \sum_{i=1}^{n} \sum_{j=1}^{m} sim(O_i, sen_j) \quad \dots [8]$$

$sim(O_i, O_{sum})$ menunjukkan similarity antara *center* dokumen *cluster* ke-i dengan *center* dari *summary* (hasil peringkasan). Perhitungan ini dilakukan untuk dokumen *cluster* ke-i sampai dengan *n* dokumen *cluster* yang telah terbentuk. Sedangkan $sim(O_i, sen_j)$ menunjukkan *similarity* antara *center* dokumen *cluster* ke-i dengan *sentences* (kalimat) hasil peringkasan, mulai dari kalimat ke-*j* sampai dengan *m* kalimat hasil ringkasan.

Nilai *diversity* ringkasan mencerminkan keragaman kalimat hasil peringkasan. Untuk menentukan *diversity* ringkasan dapat dilakukan dengan menghitung total *similarity* antara setiap kalimat dalam ringkasan, sesuai persamaan 9. Jika ringkasan memiliki total nilai kesamaan kalimat yang tinggi, maka ringkasan tersebut memiliki keanekaragaman yang rendah. Sebaliknya, jika ringkasan memiliki total nilai kesamaan kalimat yang rendah, maka ringkasan tersebut memiliki keragaman antara kalimat yang tinggi (Alguliev, Aliguliyev, & Isazade, 2013).

$$f_{diversity} = \sum_{j=1}^{m-1} \sum_{k=j+1}^{m} sim(sen_j, sen_k) \quad \dots [9]$$

$sim(sen_j, sen_k)$ menunjukkan similarity antara *sentences* (kalimat) hasil peringkasan, mulai dari kalimat ke-*j* sampai dengan *m* kalimat hasil ringkasan. Proses ini dilakukan pada semua pasangan kalimat untuk mengetahui total kemiripan antar kalimat hasil peringkasan.

Nilai *coherence* mencerminkan tingkat *coherences* dalam kalimat dalam hasil ringkasan. Nilai *coherence* menunjukkan bahwa konektivitas antar kalimat dalam ringkasan yang cukup baik. Dengan tingkat *coherences* yang baik akan memudahkan pembaca untuk memahami informasi dalam ringkasan. Salah satu indikasi tingkat *coherences* yang baik adalah jika kalimat yang berdekatan membahas konten atau topik yang serupa, atau dengan kata lain memiliki *simirality* yang tinggi sesuai persamaan 10.

$$f_{coherences} = \sum_{j=1}^{m-1} sim(sen_m, sen_{m+1}) \quad \dots [10]$$

$sim(sen_m, sen_{m+1})$ menunjukkan similarity antara *sentences* (kalimat) hasil peringkasan, mulai dari kalimat ke-*j* sampai dengan *m* kalimat hasil ringkasan. Proses ini dilakukan pada kalimat yang berurutan untuk mengetahui total kemiripan antar kalimat hasil peringkasan.

Metode optimasi yang digunakan ialah *self-adaptive differential evolution* (SaDE). Alur proses optimasi untuk peringkasan muti dokumen terdiri dari : inisialisasi populasi, binerisasi populasi, pengurutan, penghitungan nilai fitnes, mutasi, crossover dan seleksi, binerisasi trial vektor, pengurutan, dan pemeriksaan *stopping criterion* yang ditetapkan (Umam, Putro, & Pratamasunu, 2015).

Pada proses penghitungan nilai *fitness* dilakukan untuk setiap solusi peringkasan sesuai persamaan 11. Evaluasi dilakukan untuk setiap kalimat asli yang telah dikodekan ke bentuk biner. Berdasarkan tujuan pada paper ini, perhitungan nilai *fitness* dilakukan dengan mempertimbangkan tiga

faktor kualitas ringkasan ini, yaitu *coverage*, *diversity* dan *coherence*. Solusi terbaik dan terburuk di tiap generasi dapat ditentukan dengan menghitung nilai *fitness* masing-masing.

$$fitness = \frac{f_{coverage}}{f_{diversity}} x f_{coherences} \quad \dots [11]$$

III. UJI COBA DAN EVALUASI

Pada penelitian ini, data *testing* yang digunakan yaitu dataset Text Analysis Conference (TAC) 2008 dari *National Institute of Standards and Technology* (NIST). Percobaan ini dilakukan dengan memilih 15 topik pada data *testing*, dimana setiap topik terdiri dari 10 dokumen. Kemudian data *testing* tersebut dilakukan peringkasan dengan menggunakan metode yang diusulkan. Gambar 3 adalah contoh dari dokumen TAC 2008.

Percobaan dilakukan dengan menggunakan Matlab R2013a dan berjalan di *platform* Microsoft Windows. pengujian metode usulan dengan cara membandingkan hasil *summarization* antara metode usulan dengan metode CoDiCo. Pada proses klusterisasi kalimat digunakan kombinasi parameter uji optimal yang telah digunakan pada penelitian (Pasnur, Santika, & Syaifuddin, 2014), dengan menetapkan batas nilai minimum (HR_{min}), batas selisih maksimum antara HR_{old} dengan HR_{new} (ε), batas *similarity bin* pada perhitungan *histogram ratio* (S_T) dengan nilai berturut-turut 0.1, 0.2, 1. Selain itu, kedua algoritma yang digunakan yakni metode usulan dan CoDiCo akan diuji, untuk melihat seberapa berpengaruhnnya hasil klusterisasi kalimat terhadap metode usulan yang dibandingkan dengan metode CoDiCo. Skenario yang digunakan adalah melibatkan *threshold* T_{sim} pada nilai *similarity* kalimat untuk mengevaluasi dampak dari *similarity* antara kalimat peringkasan dengan solusi kalimat peringkasan yang optimal. Pada penelitian ini akan digunakan dua nilai *threshold*, yaitu 0,7 dan 0,9.

Metode usulan dan metode CoDiCo menggunakan empat parameter yang telah ditentukan pada proses inisialisasi yaitu ukuran populasi (P), generasi maksimum (t_{max}), batas bawah (u_{min}), dan batas atas (u_{max}) dengan nilai 20, 500, -5, dan 5. Nilai-nilai parameter tersebut ditetapkan berdasarkan pilihan heuristik. Setelah beberapa dokumen peringkasan diolah dengan menggunakan metode usulan, akan didapatkan hasil peringkasan sebanyak topik yang dipilih. Sehingga akan terbentuk 15 ringkasan untuk masing-masing 15 topik yang dipilih untuk setiap metode. Gambar 4 menunjukkan ilustrasi hasil pembentukan ringkasan multi dokumen.

```
<DOC id="AFP_ENG_20050601.0580" type="story" >
<HEADLINE>
Airbus announces delay in delivering new superjumbo A380
by Daphne Benoit
ATTENTION - UPDATES, ADDS details, EADS share value ///
</HEADLINE>
<DATELINE>
PARIS, June 1
</DATELINE>
<TEXT>
<P>
Airbus said Wednesday it was up to six
months behind schedule in delivering its new superjumbo A380
aircraft to airlines due to production problems, a delay that could
entail financial penalties.

</P>
<P>
The European aircraft maker said that A380 deliveries to
customers would be pushed back by two to six months after
```

Gambar 3. Contoh Dokumen TAC 2008

```
1  "Every state has its autonomy with a self government and constitution with little
   guidelines from the federal government," he said.
2  Prior to Wednesday's talks, Ibrahim Mohammed Ibrahim, spokesmen for the Sudanese
   government, reaffirmed the willingness of his government to talk the political issue with
   the rebels as the issue of power and wealth sharing contained in the federal system of
   Sudan had been developing since 1994.
3  17, concentrated on humanitarian and security issues but "took a lot of time and ended in a
   deadlock.".
4  "We don't have time to go there without knowing why we're going there," said U.K.-based SLA
   spokesman Abdul Latif.
5  Clashes flared up in February 2003 between marauding militia, known as Janjaweed, and local
   black Africans over scarce resources in the barren western Sudanese region of Darfur.
6  The two non-Arab rebel movements took up arms in February 2003 against government
   installations, saying they wanted a bigger share of power and Sudan's resources.
7  The other rebel group in Darfur, the Justice and Equality Movement, said it had not been
   asked to join the talks in Libya.
8  The Egyptian diplomat said the summit in Libya would focus on finding ways to revive the
   stalled peace talks.
9  "It's better to focus on, immediately, the political issue we are looking for ...
10 The disagreement on the sensitive security issue has led to the collapse of the first round
   of peace talks in Abuja a month ago.
```

Gambar 4. Contoh Hasil Ringkasan

Hasil pengujian kemudian dievaluasi dengan menggunakan *Recall-Oriented Understudy of Gisting* (ROUGE) (Lin, 2004). Metode ROUGE yang digunakan untuk mengevaluasi percobaan ini adalah ROUGE-1, ROUGE-2, ROUGE-L dan ROUGE-SU. ROUGE-1 dan ROUGE-2 adalah varian dari ROUGE-N yang menganggap n-gram *recall* antara hasil ringkasan dari calon ringkasan dan referensi ringkasan untuk n yang diwakilkan dengan 1 dan 2. ROUGE-L adalah metode ROUGE yang menganggap subsequence terpanjang antara calon ringkasan dan referensi ringkasan. Sedangkan ROUGE-SU menganggap nilai unigram pada calon ringkasan dan referensi ringkasan sebagai satu unit yang dihitung. Pada tahap evaluasi ini, mengacu pada metode CoDiCo di setiap nilai *threshold* yang digunakan.

IV. HASIL DAN PEMBAHASAN

Hasil uji coba metode usulan dengan paratemer yang telah ditentukan sebelumnya dan *threshold* pada optimisasi sebesar 0.9, menunjukkan bahwa nilai ROUGE-1 tertinggi adalah 0.7893 untuk topik D0814CA. Untuk nilai ROUGE-2 dan ROUGE-SU tertinggi adalah 0.3822 dan 0.5677 untuk topik D0816CB. Sedangkan ROUGE-L tertinggi adalah 0.7521 untuk topik D0822DA. Hal tersebut dapat dilihat pada Tabel 1.

Tabel 1. Hasil Metode Usulan pada Setiap Topik

TOPIK	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-SU
D0801AB	0.7670	0.1638	0.7415	0.5171
D0814CA	**0.7893**	0.2353	0.7438	0.5576
D0814CB	0.7030	0.1869	0.6287	0.4147
D0816CB	0.7773	**0.3822**	0.6900	**0.5677**
D0821DB	0.7409	0.1852	0.6891	0.3675
D0822DA	0.7607	0.3130	**0.7521**	0.5196
D0827EA	0.5959	0.2199	0.5551	0.2329
D0827EB	0.2548	0.0275	0.2471	0.0510
D0829FA	0.7559	0.2080	0.7244	0.4530
D0829FB	0.5444	0.1429	0.5037	0.2198
D0831FB	0.7521	0.1609	0.7051	0.4969
D0842GA	0.7384	0.1717	0.6920	0.4987
D0846HA	0.3673	0.0856	0.3496	0.1109
D0846HB	0.6786	0.2727	0.6250	0.3854
D0847HA	0.7778	0.3024	0.7064	0.4829

Telah dilakukan percobaan untuk mengetahui apakan dengan penambahan pembobotan kalimat berdasarkan klaster pada optimasi *coverage*, *diversity*

dan *coherence* untuk peringkasan multi-dokumen dapat meningkatkan kualitas hasil peringkasan dengan tetap mengacu pada metode sebelumnya yaitu CoDiCo (Umam, Putro, & Pratamasunu, 2015). Berdasarkan percobaan yang telah dilakukan pada Tabel 2.

Tabel 2. Perbandingan Nilai Rouge Setiap Metode

METODE	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-SU
CODICO (th=0.7)	0.6063	0.1795	0.5709	0.2972
CODICO (th=0.9)	0.5506	0.1599	0.5167	0.2630
METODE USULAN (th=0.7)	0.2252	0.0276	0.2181	0.0691
METODE USULAN (th =0.9)	0.6704	0.2051	0.6271	0.3951

Metode usulan dengan menggunakan *threshold* 0.9 menujukan hasil evaluasi menggunakan ROUGE-1, ROUGE-2, ROUGE-L dan ROUGE-SU yang lebih tinggi dibandingkan metode sebelumnya yaitu sebesar 0.6704, 0.2051, 0.6271 dan 0.3951. Pada *threshold* 0.8, metode usulan memperoleh hasil yang kurang memuaskan dan jauh dibandingkan hasil dari metode sebelumnya. Hal ini menujukkan bahwa dengan menggunakan pembobotan berdasarkan kluster pada peringkasan mengakibatkan ketidak stabilan hasil yang diperoleh jika menggunakan *threshold* yang tidak sesuai. Namun, dengan *threshold* yang tepat, metode ini dapat meningkatkan akurasi dengan sangat baik.

Dengan menambahkan pembobotan kalimat berdasarkan klaster, hal ini mengakibatkan sub-sub topik dalam dokumen terekstrak dengan baik sehingga keterwakilan sub-sub topik tersebut berada dalam hasil peringkasan dokumen. Hal ini dapat mengakibatkan hasil peringkasan yang lebih baik dari pada metode sebelumnya karena tidak ada redundansi kalimat karena tiap kalimat mewakili tiap sub topik yang ada.

Dengan metode pengklasteran *Similarity Based Histogram Ratio Clustering* (SHC), pada awal proses akan menjamin keterwakilan sub-sub topik dalam dokumen yang akan diringkas hadir dalam peringkasan dan akan mengurangi kalimat yang *redundant*. Hal ini berkaitan dengan beberapa penelitian sebelumnya yang menggunakan pengklasteran SHC (Lukmana, Swanjaya, Kurniawardhani, Arifin, & Purwitasari, 2014), (Sarkar, 2009), (Pasnur, Santika, & Syaifuddin, 2014).

V. KESIMPULAN

Teknik *clustering* dengan *Similarity Based Histogram Ratio Clustering* (SHC) dapat diimplementasikan sebagai salah tahapan persiapan untuk peringkasan multi dokumen. Implementasi tersebut dilakukan dengan memanfaatkan prinsip kerja *LSI* dan *SHC* untuk pembentukan *cluster* kalimat secara semantik. Selanjutnya hasil clustering digunakan sebagai dasar pada proses pembentukan ringkasan dokumen dengan optimisasi menggunakan *self-adaptive differential evolution (SaDE)* dan pembobotan dengan TF-ICF.

Hasil pengujian metode yang diusulkan mampu mencapai nilai *ROUGE-1* sebesar 0.6704, *ROUGE-2* sebesar 0.2051, *ROUGE-L* sebesar 0.6271 dan *ROUGE-SU* sebesar 0.3951. Nilai *ROUGE-1* metode yang diusulkan lebih tinggi 21% dari metode *CODICO*. Nilai *ROUGE-2* metode yang diusulkan lebih tinggi 28% dari metode *CODICO*. Nilai *ROUGE-L* metode yang diusulkan lebih tinggi 21% dari metode *CODICO*. Sedangkan nilai *ROUGE-SU* metode yang diusulkan lebih tinggi 50% dari metode *CODICO*. Dengan demikian metode yang diusulkan layak untuk diimplementasikan pada peringkasan multi dokumen.

REFERENSI

Alguliev, R. M., Aliguliyev, R. M., & Isazade, N. R. 2013. *Multiple documents summarization based on evolutionary optimization algorithm. Expert Systems with Applic tions*, 40, 1675-1689.

Hammouda, K., & Kamel, M. 2003. *Incremental Document Clustering Using Cluster Similarity Histograms. IEEE/WIC International Conference on Web Intelligence* (WI'03). IEEE.

Lin, C.-Y. 2004. *ROUGE: A Package for Automatic Evaluation Summaries. in Proceedings of*

the workshop on text summarization branches out, (hal. 74-81). Spain.

Lukmana, I., Swanjaya, D., Kurniawardhani, A., Arifin, A. Z., & Purwitasari, D. 2014. *Multi-Document Summarization Based On Sentence Clustering Improved Using Topic Words*. Jurnal Ilmiah Teknologi Informasi.

Pasnur, Santika, P., & Syaifuddin, G. 2014. *Semantic Clustering* Dan Pemilihan Kalimat Representatif Untuk Peringkasan Multi Dokumen. Jurnal Teknologi Informasi dan Ilmu Komputer, 91-97.

Sarkar, K. 2009. *Sentence Clustering-based Summarization of Multiple Text Documents*. *International Journal of Computing Science and Communication Technologies*.

Song, W., & Park, S. C. 2004. *Genetic Algorithm for Text Clustering Based on Latent Semantic Indexing*. *Computers and Mathematics with Applications*, 1901-190.

Suputra, I. H., Arifin, A., & Yuniarti, A. 2013. Pendekatan *Positional Text Graph* Untuk Pemilihan Kalimat Representatif *Cluster* Pada Peringkasan Multi-Dokumen. Jurnal Ilmiah Ilmu Komputer.

Umam, K., Putro, F. W., & Pratamasunu, G. Q. 2015. *Coverage, Diversity, and Coherence Optimization For Multi-Document Summarization*. Jurnal Ilmu Komputer dan Informasi.

Wahib, A., Pasnur, Santika, P., & Arifin, A. 2015. Perangkingan Dokumen Berbahasa Arab Menggunakan *Latent Semantic Indexing*. Jurnal Buana Informatika