Prosiding
ANNUAL RESEARCH SEMINAR 2016
6 Desember 2016, Vol 2 No. 1

ISBN : 979-587-626-0 | UNSRI http://ars.ilkom.unsri.ac.id

Rancang Bangun Sistem Peringkasan Teks Multi-Dokumen

Gilbert Christopher
Department of Informatics Engineering
Sriwijaya University
Palembang, Indonesia
gilbertchrist95@gmail.com

Novi Yusliani
Department of Informatics Engineering
Sriwijaya University
Palembang, Indonesia
novi_yusliani@unsri.ac.id

***Abstrak*—Seiring dengan bertumbuhnya jumlah dokumen digital yang sangat pesat, membuat pengguna dihadapkan dengan sejumlah besar informasi yang redundan. Oleh sebab itu, dibutuhkan suatu sistem yang dapat melakukan peringkasan teks multi-dokumen untuk mengekstrak kumpulan informasi relevan secara otomatis. Pada penelitian ini diusulkan sebuah rancangan peringkasan teks multi-dokumen dengan pendekatan *clustering* dan pemilihan kalimat. Proses *clustering* kalimat menggunakan metode *Latent Semantic Indexing* (LSI) dan *Similarity Based Histogram Clustering* (SHC). Penggunaan metode LSI dipilih karena mampu menangkap kemiripan hubungan *term-term* dalam suatu konteks kalimat serupa secara semantik dan metode SHC dilakukan untuk memperoleh *cluster-cluster* kalimat yang berkoherensi tinggi. Sedangkan proses pemilihan kalimat menggunakan metode *Sentences Information Density* (SID), yaitu proses pemilihan kalimat berbasis *positional text graph* dimana representasi kalimat dipilih berdasarkan tingkat kemiripan antapasangan dalam suatu *graph*. Kombinasi metode tersebut mampu menghasilkan sebuah peringkasan teks multi-dokumen yang mengandung *coverage, diversity* dan koherensi yang tinggi.**

Keywords—multi-document summarization; latent semantic indexing; similaritiy based histogram clustering; sentences information density; sentences clustering

I. LATAR BELAKANG

Seiring dengan perkembangan teknologi informasi-komunikasi yang ekponensial, menyebabkan ketersediaan dokumen digital tumbuh dengan jumlah yang pesat [1]. Hal ini tentu menyulitkan pengguna dalam mengekstrak informasi yang relevan dan sesuai dengan kebutuhan. Pengguna kini dihadapkan dengan sejumlah besar informasi yang bersifat redundan dengan konten yang secara kontekstual sama tapi dengan narasi yang berbeda [2]. Berdasarkan masalah tersebut maka dibutuhkan suatu sistem yang mampu mengekstrak informasi-informasi dari beberapa dokumen dan merepresentasikannya ke dalam suatu teks ringkasan yang efisien.

Peringkasan teks merupakan cara dalam menyingkat sejumlah besar dokumen ke dalam bentuk ringkasan dengan memilih informasi-informasi penting dan membuang informasi yang redundan dan/atau kurang penting. Berdasarkan jumlah sumbernya, teknik peringkasan teks dapat dikategorikan menjadi dua, yakni peringkasan dokumen tunggal dan multi-dokumen [1]–[3]. Peringkasan dokumen tunggal merupakan pemrosesan sebuah dokumen ke dalam bentuk yang lebih singkat, sedangkan peringkasan multi-dokumen merupakan pemrosesan kumpulan dokumen ke dalam sebuah ringkasan.

Penelitian mengenai peringkasan multi-dokumen merupakan kelanjutan dari peringkasan dokumen tunggal. Penelitian tersebut melakukan penggabungan dan pengintegrasian informasi dalam suatu kumpulan dokumen, melakukan sintesis pengetahuan, dan merepresentasikannya ke dalam bentuk yang lebih ringkas [1]. Kini para peneliti mulai melakukan penelitian peringkasan teks multi-dokumen untuk mencapai sebuah tujuan yaitu menghasilkan sebuah ringkasan yang memiliki *coverage* yang luas, tidak redundan dan koherensi yang tinggi.

Dalam paper ini diusulkan sebuah rancangan peringkasan multi-dokumen berdasarkan pendekatan *clustering* dan pemilihan kalimat yang diharapkan mampu menghasilkan sebuah ringkasan yang mencakup sebanyak mungkin informasi-informasi penting, keberagaman yang baik, dan koherensi antarkalimat yang tinggi. Paper ini disusun sebagai berikut; Bagian 2 menjelaskan tentang analisis prapengolahan teks. Peng*clusteran* kalimat dideskripsikan di Bagian 3. Pengurutan c*luster* dan pemilihan kalimat dideskripsikan secara berurut pada Bagian 4 dan 5. Arsitektur peringkasan

ISBN : 979-587-626-0 | UNSRI http://ars.ilkom.unsri.ac.id

yang diusulkan dideskripsikan pada Bagian 6. Terakhir Bagian 7 berisi kesimpulan dari keseluruhan isi paper ini.

II. PRAPENGOLAHAN TEKS

Prapengolahan teks merupakan tahap awal dalam pemrosesan dokumen sebelum dilakukannya proses peringkasan teks. Proses ini dilakukan untuk mengubah bentuk dokumen yang sebelumnya tidak terstuktur ke dalam bentuk yang terstruktur [4]. Hal ini dilakukan untuk menghilangkan *noise* pada saat pengambilan infromasi. Adapun tahap prapengolahan teks ini terdiri dari proses segmentasi kalimat, *case folding*, *tokenizing*, *filtering* dan *stemming*. Segmentasi kalimat merupakan proses pemecahan teks dokumen menjadi beberapa kalimat berdasarkan tanda baca titik (.), tanda seru (!) dan tanda tanya (?). Kalimat-kalimat yang dipecah tersebut selanjutnya dilakukan proses *case folding*, yaitu proses penyeragaman bentuk huruf (‘a’-‘z’ atau ‘A’-‘Z’) dan membuang semua karakter selain huruf. Kalimat-kalimat tersebut kemudian dilakukan perombakan menjadi sebuah kata-kata tunggal (*token*) melalui proses *tokenizing*. Kumpulan *token* tersebut kemudian dilakukan proses *filtering*, yakni proses pembuangan *term-term* yang tidak bermakna atau kurang berarti seperti dia, karena, bahwa, atau, pada, dan lain sebagainya. Proses terakhir yaitu *stemming*. *Stemming* merupakan proses mereduksi suatu kata ke dalam bentuk akar katanya berdasarkan aturan-aturan morfologi suatu bahasa tertentu [5]. Pada tahap prapengolahan teks ini akan dihasilkan sebuah matriks yang berisi frekuensi kemunculan masing-masing *term* didalam kalimat.

III. CLUSTERING KALIMAT

Clustering kalimat merupakan tahapan yang penting dalam proses peringkasan teks otomatis multi-dokumen berdasarkan pengelompokan. Hal ini dikarenakan masing-masing kalimat dalam set dokumen harus diidentifikasi *cluster* mana yang tepat untuk menjamin cakupan pembahasaan (*coverage*) yang baik [6]. Oleh sebab itu, diperlukan suatu pemilihan metode yang mampu menjamin kalimat-kalimat dalam *cluster* tersebut memiliki koherensi yang tinggi dan *cluster* yang terbentuk bersifat tidak kaku. Hal ini dikarenakan jika kalimat-kalimat dalam suatu set dokumen tersebut dikelompokkan ke dalam *cluster* yang jumlahnya tetap, maka ada kemungkinan bahwa kalimat tersebut terpaksa masuk menjadi anggota *cluster* [2]. Hal ini tentu menyebabkan *cluster* menjadi tidak koheren dan pemilihan kalimat menjadi redundan untuk ringkasan. Pada paper ini diusulkan metode *Similarity Based Histogram Clustering* (SHC) yang diadopsi dari [7]. Metode ini menggunakan pendekatan *cluster similarity histogram* dalam menjamin koherensi sebuah *cluster*. Dalam menghitung tingkat kemiripan suatu kalimat terhadap yang lain pada proses SHC ini digunakan pendekatan semantik menggunakan *Latent Semantic Indexing* (LSI) yang mampu membangun hubungan antara istilah yang satu dengan yang lain dalam konteks kalimat yang serupa [8].

2.27. *Latent Semantic Indexing*

Term-term dalam kalimat yang diperoleh dari tahap prapengolahan teks menghasilkan sebuah matriks A yang berisi frekuensi kehadiran masing-masing *term* dalam kalimat tertentu. Pada dasarnya, kalimat dalam konteks yang satu memiliki makna yang semantik terhadap yang lainnya namun direpresentasikan dalam kata yang berbeda. Hal ini disebabkan oleh sifat teks, dimana konsep yang sama dapat diwakili kata-kata yang berbeda dan memiliki arti yang sama [10]. Dengan menggunakan LSI, hubungan *term-term* dalam konteks kalimat yang satu dengan yang lainnya secara semantik dapat diukur.

Latent Semantic Indexing (LSI) adalah sebuah metode untuk menggali dan merepresentasikan konteks tersembunyi dalam suatu dokumen [8] [9]. LSI menganut suatu asas bahwa kata-kata yang digunakan dalam konteks yang sama cenderung memiliki makna yang semantik. Oleh sebab itu, penentuan tingkat kemiripan antarkalimat dalam pembentukan *cluster* menggunakan prinsip LSI dapat menghubungkan antara istilah yang satu dengan yang lain dalam konteks kalimat yang serupa.

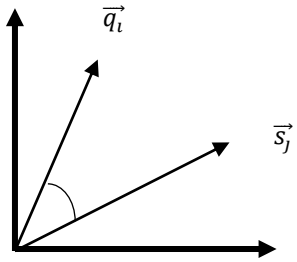
Metode LSI menggunakan *Singular Value Decomposition* (SVD) dalam menemukan relasi semantik antara kata dan kalimat [11]. Setelah matriks A antara *term* dan kalimat dibentuk, maka SVD akan mendekomposisi matriks A menjadi tiga buah matriks T, Σ, dan S, sesuai dengan (1). Proses ini dimaksudkan sebagai proses pengurangan *noise*, sehingga diperoleh matriks *Latent Semantic* yang bebas dari *noise dimension*.

$$A = T \cdot \sum \cdot S^T \qquad (1)$$

Matriks T dan S merupakan matriks *singular* vektor kiri dan kanan, dan matriks Σ merupakan matriks diagonal yang berisi nilai *singular*.

Pengukuran jarak kemiripan antarkalimat dalam matriks tersebut menggunakan pendekatan *cosine similarity. Cosine similarity* adalah perhitungan tingkat kemiripan berdasarkan besar sudut kosinus antara dua vektor, yaitu vektor kalimat pertama yang merupakan *centroid* hasil kali antara vektor *term* (T) terhadap matriks *singular value* (∑) dan vektor kalimat kedua yang merupakan vektor hasil kali antara matriks *singular value* (∑) terhadap vektor kalimat (S). Berdasarkan nilai kosinus tersebut, maka nilai *similarity* memiliki rentang nilai antar 0 sampai 1. Semakin besar nilai *similarity*, maka semakin relevan pula pasangan kalimat tersebut.

ISBN : 979-587-626-0 | UNSRI http://ars.ilkom.unsri.ac.id



Gambar 1. Vektor dalam *Cosine Similarity*

Perhitungan nilai *Cosine Similarity* dapat dituliskan sebagai

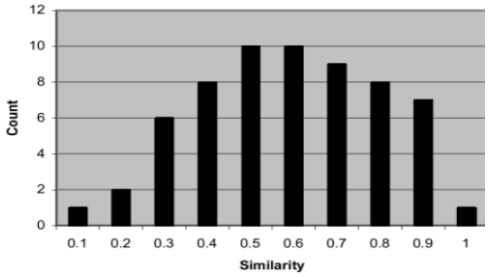
$$\cos(\vec{q_i}, \vec{s_j}) = \frac{\vec{q_i}.\vec{s_j}}{|\vec{q_i}||\vec{s_j}|} \quad \text{dimana}$$

$\vec{q_i}$ = vektor kalimat ke-i
$\vec{s_j}$ = vektor kalimat ke-j
$\overrightarrow{|q_i|}$ = panjang kalimat vektor q_i
$\overrightarrow{|s_j|}$ = panjang kalimat vektor s_j (2)

2.28. *Similarity Based Histogram Clustering*

Similarity Based Histogram Clustering (SHC) adalah metode representasi statistik yang berasal dari himpunan pasangan kalimat dalam suatu *cluster* [7]. Metode yang diadopsi dari [7] ini terbukti lebih baik dibandingkan *Single-Pass Clustering, K-Nearest Neightbor, dan Hierarchical Agglomertative Clustering* dalam hal penentuan *clustering* kalimat. Hal ini dikarenakan metode SHC mampu menjaga setiap *cluster* sebisa mungkin berada dalam kondisi koheren dan mempertahankan derajat koherensi dalam suatu *cluster*.

Salah satu cara dalam SHC untuk menjaga setiap *cluster* dalam kondisi koheren yang tinggi adalah dengan mempertahankan derajat kemiripan antar anggota yang cenderung ke kanan pada histogram kemiripan dalam *cluster* seperti Gambar 2.



Gambar 2. Histogram Kemiripan dalam *Cluster*

Kualitas dari koherensi *cluster* yang direpresentasikan melalui kemiripan dalam suatu histogram tersebut ditentukan dengan menghitung rasio histogram. Perhitungan rasio histogram tersebut dapat dituliskan sebagai

$$HR = \frac{\sum_{i=T}^{nb} h_i}{\sum_{i=1}^{nb} h_i} \quad \text{dimana}$$

T = Batas ambang kemiripan x jumlah bin histogram
h_i = jumlah kemiripan pada bin ke-i
(3)

Berikut merupakan langkah-langkah dari metode SHC.

```
Algoritma SHC
DEKLARASI
    L : cluster list  {cluster kosong}
    s  : array [1..n] of string
         {kumpulan kalimat peringkasan}
    c  : list
DESKRIPSI
    FOR each k on the s
        FOR each c on the list L
            RH_lama = RH_c
            Simulate insert k to c
            RH_baru = RH_c
            IF (RH_baru ≥ RH_lama) OR ((RH_baru ≥ RH_min)
                AND ((RH_lama – RH_baru < ε)) THEN
                Insert k to c
            END IF
        END FOR
        IF k was not inserted to c THEN
            Create new c
            Insert k to c
            Add c to L
        END IF
    END FOR
```

Gambar 3. *Pseudocode clustering* Kalimat Menggunakan Metode SHC

Berdasarkan *pseudocode* metode SHC pada Gambar 3, SHC melakukan pengujian masing-masing kalimat terhadap *cluster* c pada *cluster list* L. Nilai rasio histogram diukur pada saat sebelum dan sesudah penambahan kalimat pada *cluster* c. Apabila rasio histogram sesudah penambahan kalimat lebih besar dari nilai rasio sebelumnya atau nilai rasio histogram sesudah penambahan lebih besar dari rasio histogram minimum dan selisih nilai antara rasio histogram sebelum dan sesudah penambahan kalimat lebih kecil dari epsilon(ε), maka kalimat tersebut ditambahkan. Proses tersebut dilakukan terus menerus sebanyak jumlah *cluster* yang terbentuk. Apabila setelah dilakukan pengujian dan kalimat k tersebut tidak berada di

ISBN : 979-587-626-0 | UNSRI http://ars.ilkom.unsri.ac.id

cluster manapun, maka *cluster* baru akan dibuat dalam *list* L dan diisi dengan kalimat k tersebut.

IV. PENGURUTAN CLUSTER

Karena metode yang digunakan dalam *clustering* kalimat merupakan metode *unsupervised* dan tidak ada kepastian khusus berapa jumlah *cluster* yang terbentuk, maka sangat penting untuk menentukan *cluster* mana yang akan dijadikan perwakilan ringkasan. Salah satu metode termudah dalam menentukannya yaitu mengasumsikan bahwa *cluster* yang memiliki jumlah kalimat lebih banyak merupakan *cluster* yang penting dan dijadikan perwakilan ringkasan [6]. Akan tetapi, metode tersebut memiliki kekurangan karena beberapa kelompok *cluster* mungkin saja memiliki jumlah kalimat yang sama dan sejumlah kalimat dalam suatu *cluster* kurang informatif, yang terkesan hanya memperbanyak kalimat dalam suatu *cluster*.

Dalam paper ini, diusulkan suatu pendekatan pengurutan *cluster* yang mampu mengatasi masalah diatas, yaitu pengurutan menggunakan *cluster importance*. *Cluster importance* ini merupakan suatu pembobotan *cluster* berdasarkan jumlah bobot suatu *term* dalam *cluster* yang memiliki frekuensi kemunculan diatas batas ambang/ *threshold* (Θ). Perhitungan *cluster importance* dalam suatu *cluster* dapat dituliskan dalam (4).

$$weigth(c_j) = \sum_{t \in c_j} \log(1 + count(t)) \qquad (4)$$

Dimana *count(t)* merupakan jumlah *term* t pada *cluster* ke-j yang lebih dari *threshold* dan *weight*(c_j) merupakan bobot yang menyatakan *information richness* dari suatu *cluster* ke-j. Ketika semua bobot *cluster* telah dihitung, selanjutnya dilakukan proses pengurutan berdasarkan bobot terbesar hingga terkecil. Nilai bobot *cluster* terbesar akan dijadikan kandidat *cluster* pertama dan begitu seterusnya sampai bobot yang terkecil.

V. PEMILIHAN KALIMAT

Pemilihan representatif kalimat yang dijadikan kandidat ringkasan dalam suatu *cluster* merupakan tahapan penting dalam menghasilkan ringkasan yang *coverage* dan *diversity*. Terdapat beberapa solusi tradisional dalam memilih representatif kalimat dalam suatu *cluster,* diantaranya memilih secara acak atau memilih kalimat yang terpanjang. Akan tetapi, kedua solusi tersebut belum mampu menghasilkan representatif kalimat yang baik [6]. Hal ini dikarenakan apabila memilih kalimat secara acak atau secara jumlah kata terbanyak, maka kalimat yang dipilih tersebut belum tentu menghasilkan suatu representasi informasi secara keseluruhan dalam suatu *cluster*.

Pada paper ini diusulkan sebuah metode pemilihan representatif kalimat berdasarkan *positional text graph* dalam menentukan kalimat-kalimat yang informatif. Metode tersebut adalah *Sentence Information Density* (SID) yang diadopsi dari [12]. Dalam paper ini nilai SID dihitung berdasarkan nilai kemiripan pasangan-pasangan kalimat dalam suatu *cluster* yang membentuk sebuah *graph* [13].

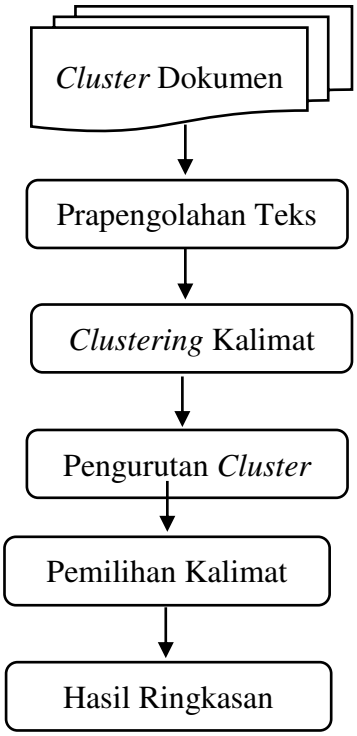
Masing-masing *cluster* yang telah diurutkan dalam proses sebelumnya dibentuk sebuah *graph*. *Graph* digambarkan sebagai G = (V,E) dimana V merupakan himpunan simpul (*vertex*) pada *graph* yang berisi kalimat-kalimat dalam suatu *cluster*, dan E merupakan himpuanan setiap busur (*edge*) pada *graph* yang berisi nilai kemiripan antarkalimat. Apabila nilai kemiripan antar pasangan kalimat tersebut memenuhi nilai batas ambang/*threshold* α, maka *edge* dibentuk dengan nilai bobot yang berasal dari nilai kemiripan antarkalimat. Ketika *graph* terbentuk, nilai SID dihitung menggunakan (5).

$$F_{sim}(k_i) = \frac{W_{k_i}}{max_{j \in \{1,2,..n\}} W_{kj}} \qquad (5)$$

Dimana $F_{sim}(k_i)$ adalah nilai bobot representasi kalimat ke-i dalam suatu *cluster*, W_{k_i} adalah jumlah semua nilai kemiripan yang datang dari kalimat k ke-i pada sebuah *cluster*, dan W_{kj} adalah jumlah semua nilai kemiripan maksimum diantara semua pasangan kalimat ke-j yang ada pada *cluster*. kalimat yang ada pada masing-masing *cluster* yang memiliki nilai SID yang terbesar akan dijadikan kandidat sebuah kalimat dalam ringkasan.

ISBN : 979-587-626-0 | UNSRI http://ars.ilkom.unsri.ac.id

VI. ARSITEKTUR PERINGKASAN



Gambar 4. *Framework* Peringkasan Teks Multi-Dokumen yang Diusulkan

Pada paper ini diusulkan sebuah skema perancangan peringkasan teks multi-dokumen secara ektraksi seperti Gambar 4. Dokumen yang telah di *cluster* berdasarkan topik terlebih dahulu dilakukan proses prapengolahan teks meliputi segmentasi kalimat, *case folding*, *tokenizing*, *filtering*, dan *stemming*. Hal ini dilakukan untuk mengubah dokumen-dokumen masukan ke dalam bentuk matriks yang berisi bobot *term* terhadap kalimat. Matriks tersebut kemudian dijadikan input dalam proses selanjutnya, yaitu proses *clustering* kalimat yang meliputi LSI dan SHC. LSI dilakukan untuk memperoleh tingkat kemiripan antarpasangan kalimat berdasarkan relasi semantik. Selanjutnya, dilakukan proses pengelompokkan kalimat menggunakan metode SHC, yaitu suatu proses pengelompokkan kalimat ke dalam masing-masing *cluster* koheren berdasarkan rasio histogram.

Dalam melakukan pengurutan *cluster*, *cluster importance* melakukan pengurutan berdasarkan nilai penjumlahan bobot *term-term* yang merupakan kata *frequent*. Untuk melakukan pemilihan kalimat, diajukan sebuah metode perhitungan berbasis *positional text graph* menggunakan *Sentences Information Density* (SID). Kalimat dengan SID terbesar akan dijadikan kandidat kalimat ringkasan dalam suatu *cluster*.

VII. KESIMPULAN

Peringkasan teks multi-dokumen merupakan perpanjangan dari peringkasan dokumen tunggal, dimana peringkasan tersebut lebih sulit karena harus mengintegrasikan informasi dalam suatu kumpulan dokumen dan merepresentasikannya ke dalam bentuk yang lebih ringkas. Peringkasan teks multi-dokumen menggunakan metode LSI dan SHC mampu menjaga koherensi dalam *cluster* kalimat sehingga menghasilkan ringkasan dengan cakupan yang luas dan memiliki koherensi yang tinggi. Selain itu, penggunakan metode SID juga membantu dalam memilih kalimat mana yang akan dijadikan perwakilan *cluster.*

REFERENSI

[1] R. M. Alguliev, R. M. Aliguliyev, and N. R. Isazade, "Multiple documents summarization based on evolutionary optimization algorithm," *Expert Syst. Appl.*, vol. 40, no. 5, pp. 1675–1689, 2013.

[2] R. Azhar, M. Machmud, H. A. Hartanto, and A. Z. Arifin, "Pembobotan Kata Berdasarkan Klaster pada Optimisasi Coverage , Diversity dan Coherence untuk Peringkasan Multi Dokumen," *Inst. Teknol. Sepuluh Nop.*, vol. 2, 2016.

[3] Nidhi and V. Gupta, "Recent Trends in Text Classification Techniques," *Int. J. Comput. Appl.*, vol. 35, no. 6, pp. 45–51, 2011.

[4] R. Feldman and J. Sanger, *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*, vol. 1. Cambridge: Cambridge University Press, 2007.

[5] D. Keke, R. Chikita, and A. D. Prayogo, "Sistem temu balik informasi Algoritma Nazief Adriani," Gadjah Mada, 2012.

[6] K. Sarkar, "Sentence Clustering-based Summarization of Multiple Text Documents," *Tech. – Int. J. Comput. Sci. Commun. Technol.*, vol. 2, no. 1, pp. 325–335, 2009.

[7] K. M. Hammouda and M. S. Kamel, "Incremental document clustering using cluster similarity histograms," *Proc. - IEEE/WIC Int. Conf. Web Intell. WI 2003*, pp. 597–601, 2003.

[8] W. Song and S. C. Park, "Genetic algorithm for text

ISBN : 979-587-626-0 | UNSRI http://ars.ilkom.unsri.ac.id

clustering based on latent semantic indexing," *Comput. Math. with Appl.*, vol. 57, no. 11–12, pp. 1901–1907, 2009.

[9] A. Thomo, "Latent semantic analysis Tutorial," *Victoria, Canda.*, pp. 1–7, 2009.

[10] S. Zelikovitz and H. Hirsh, "Using LSI for text classification in the presence of background text," *Proc. tenth Int. Conf. Inf. Knowl. Manag. - CIKM'01*, p. 113, 2001.

[11] M. G. Ozsoy, F. N. Alpaslan, and I. Cicekli, "Text summarization using Latent Semantic Analysis," *J. Inf. Sci.*, vol. 37, no. August, pp. 405–417, 2011.

[12] T. He, F. Li, W. Shao, J. Chen, and L. Ma, "A new feature-fusion sentence selecting strategy for query-focused multi-document summarization," *Proc. - ALPIT 2008, 7th Int. Conf. Adv. Lang. Process. Web Inf. Technol.*, pp. 81–86, 2008.

[13] I. P. Gede, H. Suputra, A. Z. Arifin, A. Yuniarti, and K. Its, "Pendekatan Postionla Text Graph untuk Pemilihan Kalimat Representatif Cluster pada Peringkasan Multi-Dokumen," vol. 6, no. 2, pp. 18–24, 2013.