Register : Jurnal Ilmiah Teknologi Sistem Informasi 3 (2) 94-104
ISSN 2502-3357 (*Online*)
ISSN 2503-0477 (*Print*)
Tersedia *online* di www.journal.unipdu.ac.id
Unipdu
Halaman jurnal di www.journal.unipdu.ac.id/index.php/register

Peringkasan dokumen berita Bahasa Indonesia menggunakan metode *Cross Latent Semantic Analysis*

Gamaria Mandar [a], *Gunawan Gunawan* [b]

[a,b] *Teknologi Informasi, Sekolah Tinggi Teknik Surabaya, Surabaya, Indonesia*

email: [a] gamariamandar@gmail.com, [b] gunawan@stts.edu

INFO ARTIKEL

Sejarah artikel:
Menerima 18 Mei 2018
Revisi 1 Juni 2018
Diterima 1 Juni 2018
Online 2 Juni 2018

Kata kunci:
Berita
Cross latent semantic analyisis
Latent semantic analysis
Peringkasan dokumen
RSS-Feed

Keywords
Cross latent semantic analysis
Document summarization
Latent semantic analysis
News
RSS-Feed

Style APA dalam mensitasi artikel ini:
Mandar, G & Gunawan, G. (2018). Peringkasan dokumen berita Bahasa Indonesia menggunakan metode Cross Latent Semantic Analysis. *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, 3(2), 94-104.

ABSTRAK

Peringkasan dokumen berita Bahasa Indonesia dapat membantu untuk menemukan ide-ide pokok atau informasi penting lain dari sebuah berita. Berita umumnya terdiri atas banyaknya paragraf menjadi sebab diperlukan sebuah sistem untuk mengekstrak informasi, sehingga mampu memberikan ide pokok atau informasi penting yang tepat kepada pembaca, tanpa harus membaca secara detail keseluruhan isi berita tersebut, di samping itu dapat dimanfaatkan guna keperluaan *Really Simple Syndication Feed* (RSS-Feed). Penelitian ini memaparkan peringkasan dokumen berita berbahasa Indonesia menggunakan metode *Cross Latent Semantic Analysis* (CLSA) dan *Latent Semantic Analysis* (LSA). Untuk menguji seberapa baik hasil ringkasan yang dilakukan CLSA penelitian ini menggunakan 240 artikel berita yang diambil dari halaman portal www.kompas.com dan dua pakar yang berlatar belakang bidang yang berbeda. Hasil ringkasan CLSA dengan *compression rate* 30% memperoleh nilai *F-Measure* 0,72%. Penelitian ini juga menemukan fakta bahwa CLSA lebih baik dari metode LSA yang merupakan cikal bakal dari metode CLSA, walaupun skor hasil *F-Measure* keduanya tidak berbeda jauh.

ABSTRACT

Summarizing news documents in Bahasa serves to find main ideas or any other important information from a piece of news. A system to extract the information from ones consisting of many paragraphs is then deemed necessary in order to present precise main ideas or important information to the readers without them having to read the entire passage of news documents, in addition to become useful for Really Simple Syndication Feed (RSS-Feed). This article discusses summarizing news documents in Bahasa using Cross Latent Semantic Analysis (CLSA). To test if the summary resulted from CLSA qualified, this study examines 240 news articles retrieved from www.kompas.com and employs two experts from different fields. The summary resulted from CLSA with a compression rate of 30% obtains an F-Measure of 0,72%. This study also evidently indicates that CLSA has better performance from Latent Semantic Analysis (LSA) which was the initial system for CLSA, despite both F-Measure percentages being only slightly different.

1. Pendahuluan

Perkembangan internet yang semakin cepat dan terus mengalami inovasi setiap saat, memacu pertumbuhan informasi dan memunculkan berbagai situs-situs berita *online* baik nasional maupun skala lokal di Indonesia. Tercatat pada sebuah riset 2016 oleh Indonesia Digital Association (IDA) terkait sumber yang digunakan konsumen dalam pencarian berita, sumber *online* memperoleh peringkat yang tertinggi (Viva, 2016).

Kemudahan ini menyebabkan informasi semakin banyak dan beragam yang tersedia secara *online*, berita biasanya terdiri dari teks yang panjang, hingga membutuhkan waktu untuk memahami

ISSN 2502-3357 (*Online*)
G. Mandar dan G. Gunawan/Register 3 (2) 94-104 ISSN 2503-0477 (*Print*)

inti (informasi penting atau ide pokok) dari berita tersebut, untuk itu diperlukan sebuah peringkasan teks otomatis yang dapat membantu mengekstrak informasi penting dari isi berita. Salah satu bidang yang mampu mengatasi masalah ini ialah *Text Summarization* (Peringkasan Teks).

Penelitian peringkasan dokumen pertama kali dimulai pada tahun 1958 yakni *The Automatic Creation of Literature Abstract* oleh Luhn, penelitian tersebut menghasilkan sebuah ringkasan dengan menghitung nilai frekuensi kata dan kalimat untuk menentukan hasil ringkasan yang terbaik. Di tahun yang sama Baxendle menemukan fakta bahwa 85% kalimat yang mengadung topik dari isi berita berada pada awal kalimat dan 7% pada akhir kalimat (Das & Martins, 2007). Keuntungan dari metode peringkasan dokumen ini adalah upaya untuk memaksimalkan kepadatan informasi yang diberikan kepada pembaca.

Umumnya peringkasan dokumen diklasifikasi menjadi dua yaitu peringkasan ekstraktif dan peringkasan abstraktif (Gunawan, Juandi, & Soewito, 2015). Peringkasan ekstraktif adalah ringkasan dengan memilih kalimat penting dari dokumen asli, dengan cara mengekstrak kalimat berdasarkan fitur statistik dan linguistik, sedangkan peringkasan abstraktif adalah memahami dokumen asli dan menghasilkan kalimat baru dari dokumen yang diringkas, metode ini lebih kompleks serupa dengan ringkasan yang dilakukan oleh manusia (Badry, Eldin, & Elzanfally, 2013).

Salah satu algoritma yang digunakan untuk menyelesaikan peringkasan ekstraktif adalah *Latent Semantic Analysis* (LSA). Peringkasan dokumen dengan LSA menggunakan *Singular Value Decomposition (SVD)* untuk menemukan kemiripan semantik kata dan kalimat pada sebuah dokumen. SVD adalah model hubungan antara kata dan kalimat (Ozsoy, Cicekli, & Alpaslan, 2010). Peringkasan dokumen mengunakan LSA pertama kali diusulkan Gong dan Liu. Penelitian ini menggunakan dua metode yaitu *Information Retrieval* (IR) dan LSA. IR melakukan pendekatan pada ukuran relevansi dengan mengambil kalimat nilai relevansi tertinggi untuk dijadikan ringkasan, sedangkan LSA mengidentifikasi kalimat yang memiliki nilai indeks dan *singular vector* terbesar untuk dijadikan sebuah ringkasan. Dengan menggunakan tiga pakar dari 243 artikel berita, hasil evaluasi dari kedua metode tersebut tidak jauh berbeda yaitu *F-Measure* IR 0,55% dan LSA 0,57% (Gong & Liu, 2001), namun menurut Steinberger dan Ježek (2004) menyebutkan ada dua kelemahan yang terdapat pada penelitian Gong: 1) Apabila menggunakan demensi yang lebih tinggi, maka jumlah ruang demensi akan berkurang, hal ini menyebabkan topik sebuah berita kurang signifikan untuk menghitung ringkasan, akan tetapi kekurangan ini menjadi keuntungan apabila mengetahui berapa banyak topik yang berbeda pada dokumen asli; dan 2) Kalimat yang memiliki nilai indeks besar, tetapi tidak terbesar, tidak dipilih meskipun isinya untuk ringkasan sangatlah cocok.

Untuk menghilangkan kekurangan tersebut, Steinberger dan Ježek (2004) menghitung nilai SVD dari setiap kata di matriks kalimat, sehingga mendapatkan tiga matriks, yakni matriks A, matriks V dan matriks U. Selanjutnya untuk setiap vektor matriks V^T dikalikan dengan nilai *singular* guna memperoleh panjang dari masing-masing dokumen kalimat. Panjang dokumen kalimat ternyata mampu meningkatkan akurasi *F-Measure* dari LSA pada penelitian sebelumnya yaitu 0,75% menjadi 0,78%. Penelitian yang sama dilakukan oleh Zeniarja, dkk (2013) pada dokumen Bahasa Indonesia yang mengabungkan metode fitur dan LSA sebagai *feature reducation* yang memperoleh hasil *F-Measure* 0,93%, di lain sisi peringkasan 10 dokumen berita kesehatan menggunakan LSA dengan *compression rate* 50% juga memperoleh *F-Measure* yang baik yakni 0,70% (Gotami, Indriati, & Dewi, 2018).

Perkembangan penelitian di bidang peringkasan hampir setiap saat mengalami perubahan yang pesat, begitu pula dengan metode yang digunakan dengan segala upaya untuk menghasilkan ringkasan yang lebih baik. Sama halnya dengan metode LSA, penelitian terdahulu Steinberger menjadi cikal bakal muncul *Cross Latent Semantic Analysis* (CLSA), Steinberger menjadikan hasil ringkasan tidak hanya berdasarkan pada kemiripan kata dan kalimat dalam suatu dokumen, melainkan panjang kalimat juga menentukan keberhasilan naiknya nilai akurasi *F-Measure* dari LSA pada penelitian sebelumnya, itulah yang dilakukan oleh Geetha dan Deepamala (2015), penelitian Geetha dan Deepamala (2015) berupaya menemukan cara untuk meningkatkan hasil akurasi *F-Measure* dengan melakukan perubahan pada

ISSN 2502-3357 (*Online*)
G. Mandar dan G. Gunawan/Register 3 (2) 94-104 ISSN 2503-0477 (*Print*)

beberapa proses SVD dan membuang dokumen kalimat yang dianggap tidak penting dengan cara mencari rata-rata dari setiap dokumen kalimat pada matriks V^T, kemudian membandingkan setiap nilai pada matriks V^T dengan rata-rata setiap dokumen kalimat matriks V^T, hasil penelitian Geetha dan Deepamala (2015) menemukan fakta baru, bahwa seleksi dokumen matriks V^T dengan nilai rata-rata dapat meningkatkan hasil akurasi *F-Measure* dari penelitian Steinberger sebelumnya, yaitu dari 78% menjadi 87%, berdasarkan akumulasi rata-rata dari berita kesenian 0,77%, ekonomi 0,87%, sastra 0,94% dan homeopathy 0,90%. Demikianpula implementasi CLSA untuk meringkas dokumen berita Bahasa Indonesia juga memperoleh hasil akurasi yang cukup baik yakni 69,6% dari 6 artikel berita (Winata & Rainarli, 2016).

Menurut Geetha selain dua kekurangan LSA yang dipaparkan Steinberger di atas, kelemahan lainny dari LSA adalah representasi tidak eksplisit dan dapat menurunkan proses kinerja untuk dokumen yang besar dan multibahasa, namun LSA mempunyai keunggulan yakni hasil ringkasan sangat baik mewakili hubungan konseptual antara kata-kata, kalimat dan paragaraf (Geetha & Deepamala, 2015).

Berdasarkan penelitian yang sudah dipaparkan sebelumnya, penelitian ini akan memaparkan hasil peringkasan dokumen berita Bahasa Indonesia dengan metode CLSA, yang bertujuan untuk menghasilkan sebuah sistem yang mampu memberikan ide pokok atau informasi penting, berupa sebuah ringkasan dengan hasil yang cepat dan tepat tanpa perlu membaca teks berita secara keseluruhan kepada *user* yang membutuhkan, selain itu dapat membantu di beberapa bidang yang proses pekerjaanya *real-time* seperti editor/redaktur di sebuah media cetak, hasil ringkasan juga dapat dimanfaatkan untuk keperluan *Really Simple Syndication Feed* (RSS Feed), penelitian ini juga berupaya menemukan perbandingan antara hasil ringkasan menggunakan CLSA dan LSA pada berita Bahasa Indonesia.

2. Metode Penelitian

Metode penelitian pada penelitian ini meliputi beberapa hal diantaranya peringkasan dokumen, *dataset*, *preprocessing*, pembobotan kata dengan algoritma TF-IDF, dan menentukan hasil ringkasan menggunakan CLSA dan LSA.

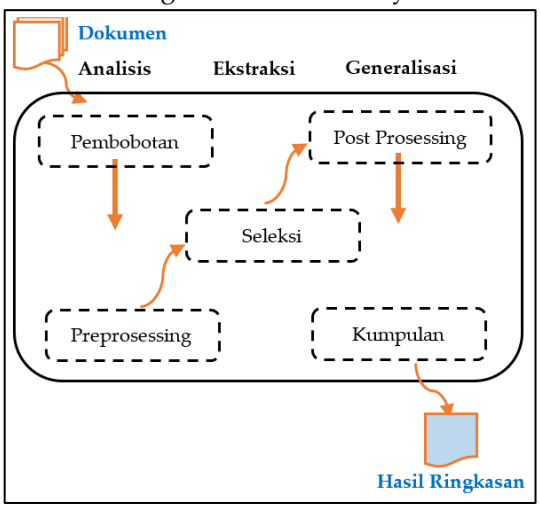
2.1 Peringkasan dokumen

Peringkasan dokumen adalah salah satu bidang *Natural Language Processing* (NLP) yang dapat mengekstrak informasi penting dari teks asli untuk menghasilkan sebuah ringkasan. Sedangkan menurut Najibullah dan Mingyan (2015), peringkasan dokumen adalah proses penyajian kembali dokumen dalam bentuk yang lebih singkat tanpa membuang informasi penting yang terdapat pada dokumen tersebut.

Tujuan dari peringkasan dokumen ialah untuk memperoleh informasi yang penting dari sebuah dokumen (teks) yang akan disajikan kepada pembaca, karena peringkasan teks otomatis mampu menghilangkan kata, kalimat yang dianggap tidak relevan atau redundan dengan tetap menjaga inti makna dari dokumen (Zeniarja, Salam, Luthfiarta, Handoko, & Jamhari, 2013). Selain itu mempermudah pembaca agar lebih cepat menangkap ide pokok atau isi yang dianggap penting pada dokumen atau teks tanpa membaca dokumen secara keseluruhan. Di bidang lain peringkasan dokumen juga sangat bermanfaat sebagai penunjang kegiatan sistem manejemen surat menyurat suatu organisasai (Najibullah & Mingyan, 2015). Menurut Manual Joan ringkasan memiliki dua fungsi utama yaitu fungsi langsung dan tidak langsung. Fungsi langsung memberikan gambaran secara umum berupa informasi penting dari sebuah dokumen (Essential Information), memberikan pembahruan terbaru kepada pembaca (Update Summary), dan menghilangkan noise pada bahasa (Eliminate Language Barriers) atau sebagai fasilitas *Information Retrieval* (IR). Sedangkan fungsi tak langsung adalah memungkinkan dokumen diklasifikasikan dan di indeks serta mengekstrak kata kunci (Torres-Moreno, 2014).

ISSN 2502-3357 (*Online*)
G. Mandar dan G. Gunawan/Register 3 (2) 94-104 ISSN 2503-0477 (*Print*)

Ringkasan dikategorikan menjadi beberapa golongan diantaranya berdasarkan fungsi, jenis dan lain-lainnya. Adapun beberapa hal yang perlu diketahui terkait peringkasan berdasarkan jenis. Hal ini sangat penting untuk menetukan hasil ringkasan. Salah satunya adalah ringkasan ekstrak.

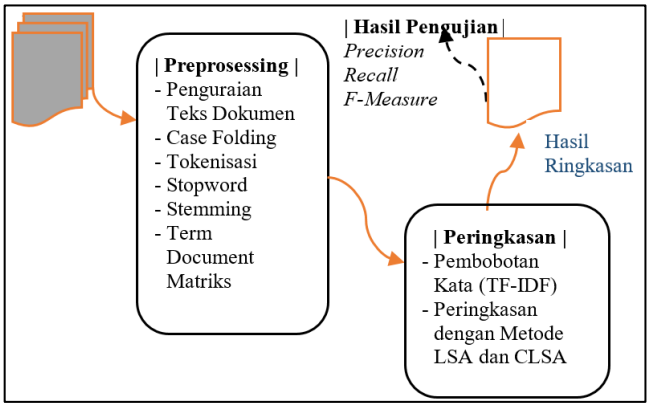


Gambar 1. Arsitektur umum ringkasan ekstrasi (Torres-Moreno, 2014)

Ringkasan ekstrak adalah ringkasan yang dihasilkan dari pemilihan unit teks. Misalkan kata, kalimat, segmen kalimat, atau paragraf yang memiliki informasi penting dari sebuah dokumen dengan mengumpulkan fragmen yang telah diekstrak dari dokumen sumber. Tujuan dari ekstrak adalah memberikan gambaran umum tentang konten dari teks asli. Pada ringkasan ekstrak umumnya terdiri dari tiga proses yaitu, tahap analisis, tahap ekstraksi dan tahap generasi seperti pada Gambar 1. Sedangkan menurut Radev dkk, hasil ringkasan ekstraksi dibagi menjadi tiga level, yaitu *surface-level*, *intermediate-level*, dan *deep parsing techiques* (Torres-Moreno, 2014):

Surface-level adalah level yang hasil ringkasan tidak menyelediki kedalaman linguistik dari sebuah teks, melainkan menggunakan elemen linguistik tertentu untuk mengidentifikasi segmen dokumen yang relevan. Teknik ini menggunakan kata dari bobot kalimat dan berdasarkan ide pada kata yang terdapat di judul berita. *Intermediate-level* merupakan proses yang menggunakan informasi linguistik yang lebih baik dari pada *surface-level*, seperti teknik *Lexical Chain Recognition*, yakni kata diurutkan dan dihubungkan secara leksikal semantik. Pada tahap ini biasanya menghasilkan ringkasan dengan empat proses yaitu: 1) Teks asli diekstrak menjadi beberapa segmen; 2) Melakukan konstruksi leksikal; 3) Mendefenisikan bobot dari setiap segmen tersebut; dan 4) Menentukan hasil ringkasan berdasarkan urutan skor nilai tertinggi. *Deep Parsing Techiques* hasil ringkasan ini merupakan tahap ekstraksi yang paling mendalam, di mana menggunakan teknik linguistik yang mendalam dengan mengeksploitasi stuktur teks diskursif.

2.2 *Arsitektur sistem*



Gambar 2. Arsitektur sistem

ISSN 2502-3357 (*Online*)
G. Mandar dan G. Gunawan/Register 3 (2) 94-104 ISSN 2503-0477 (*Print*)

Arsitektur peringkasan dokumen Bahasa Indonesia yang digunakan untuk mengambarkan sistem kerja pada penelitian ini terdiri dari tiga tahapan yaitu, *preprocessing* dan metode pengujian seperti pada Gambar 2.

2.3 Data

Dataset yang digunakan pada penelitian ini adalah berupa dokumen tunggal yang terdiri dari teks berita dengan panjang minimal 10 kalimat sebanyak 240 artikel, diambil dari halaman www.kompas.com pada tanggal 31 Januari sampai tanggal 15 Maret 2018, masing-masing 60 artikel berita untuk tiap-tiap kategori berita Nasional, Internasional, Regional dan Olahraga. Penelitian ini menggunakan dua pakar dengan latar belakang bidang yang berbeda yaitu pakar pertama berasal dari akademik dan pakar kedua adalah editor di salah satu media koran lokal, serta penelitian ini juga membatasi *compression rate* ringkasan sebesar 30%.

2.4 Preprocessing

Preprocessing merupakan tahap awal pada pemrosesan peringkasan berita. *Preprocessing* memiliki tujuan menghasilkan *Term Index* (kata) sehingga dapat menentukan bobot *term* dengan menggunakan algoritma TF-IDF (Gotami, Indriati, & Dewi, 2018). Adapun langkah-langkah *preprocessing* yang digunakan pada penelitian ini sebagai berikut:

1. Menguraikan teks berita menjadi beberapa kumpulan dokumen kalimat.
2. Menyeragamkan teks dengan cara mengubah semua huruf teks menjadi huruf kecil dan menghilangkan beberapa karakter huruf yang dianggap *delimeter*, proses ini disebut dengan istilah *Case Folding*.
3. Setelah melewati proses *case folding*, langkah selanjutnya adalah melakukan proses tokenisasi dengan cara memisahkan *string* kata dari dokumen kalimat.
4. *Stopword* adalah adalah proses untuk mengahapus kata-kata yang dianggap tidak penting, hal ini berfungsi agar dapat memaksimalkan informasi yang penting pada teks berita.
5. Langkah terakhir dari *preprocessing* adalah melakukan *stemming* yaitu dengan mengubah kata menjadi kata dasar. Tujuan *stemming* sendiri adalah mereduksi kata dari hasil token sebelumnya untuk memperoleh *index* kemunculan kata yang baik pada setiap dokumen kalimat (Asian, 2007). Algoritma *stemming* yang digunakan pada penelitian ini adalah *stemming* Bahasa Indonesia dari Nazief dan Adriyani.

2.5 Pembobotan kata mengunakan algoritma TF-IDF

Pembobotan kata adalah proses untuk menghitung bobot suatu kata pada dokumen yang dilihat dari banyaknya frekuensi kemunculkan kata pada sebuah teks berita atau dokumen kalimat. Algoritma yang digunakan pada penelitian ini adalah menggunakan *Term Frequency-Inverse Document* (TF-IDF) secara matematis dapat ditulis pada Rumus 1 dan Rumus 2 (Winata & Rainarli, 2016),

$$IDF = Log(\frac{D}{DF}) \quad (1)$$

$$W = TF \times IDF \quad (2)$$

Inverse Document Frequency (*IDF*) adalah hubungan antara banyak dokumen yang memiliki kata (*Document Frequency*) dengan jumlah dokumen kalimat (*D*), sedangkan *W* adalah nilai bobot dari setiap kata pada sebuah dokumen, hasil dari pembobotan kata adalah sebagai *dataset* untuk membentuk matriks A_{mn}. Rumus matriks A_{mn} dapat dilihat pada Rumus 3.

2.6 Peringkasan menggunakan metode LSA

Latent Semantic Analysis (LSA) menurut bahasa terbagi atas beberapa kata yang penting yaitu *latent* dan *semantic*, *latent* yang memiliki arti tersembunyi atau sesuatu yang masih belum terlihat, sedangkan *semantic* berasal dari bahasa yunani "*semanticos*" yang berarti memberi tanda, penting atau cabang linguistik yang mempelajari arti dan makna dari suatu bahasa, kode atau jenis representasi lainnya. Dari pengertian dapat ditarik kesimpulan bahwa, LSA adalah menguraikan atau menganalisa makna

ISSN 2502-3357 (*Online*)
G. Mandar dan G. Gunawan/Register 3 (2) 94-104 ISSN 2503-0477 (*Print*)

yang masih tersembunyi dari suatu bahasa, kode atau jenis representasi lainya, guna memperoleh informasi yang penting. Sedangkan menurut Rasha, LSA adalah metode yang didasarkan pada perhitungan untuk mengekstrak dan mewakili makna kontektual dari kata dan kesamaan kalimat (Badry, Eldin, & Elzanfally, 2013). Kesamaan kata dan kalimat diperoleh dengan cara menggunakan *Singular Value Decomposition* (SVD), di mana SVD mempunyai kapasitas untuk mereduksi *noise*, sehingga dapat meningkatkan hasil akurasi pada ringkasan (Zeniarja, Salam, Luthfiarta, Handoko, & Jamhari, 2013).

Konsep LSA direlisasikan dengan menggunakan dua fitur utama yaitu matriks dan SVD, struktur bahasa dalam hal ini ialah, kalimat atau kata diubah menjadi sebuah matriks, sedangkan SVD bertugas untuk mengolah komponen matriks kata dan kalimat guna menemukan hubungan kesamaan antara kata dan kalimat. Teori Aljabar Liner SVD membagi matriks *A* menjadi tiga bagian yaitu matriks orthogonal U, matriks diagonal S dan matriks *orthogonal transpose* V secara matematis dapat ditulis dengan Rumus 3.

$$A = USV^T \tag{3}$$

A adalah matriks dokumen yang mewakili kalimat atau kata yang dikenal dengan matriks A_{mn}, U mendiskripsikan matriks *orthogonal* $m \times m$ yang dikenal dengan istilah *left singular vector*, di mana U dihasilkan dari perkalian antara $U = A.V.S^{-1}$. *Right Singular Vektor* (V) merupakan matriks *orthogonal* $n \times n$ yang diperoleh dari *eigenvector* matriks A^TA, sedangkan matriks diagonal S dihasilkan dari *eigenvalue* matriks A^TA yang diakarkan. Adapun langkah-langkah LSA sebagai berikut (Geetha & Deepamala, 2015):

1. Membentuk matriks A_{mn}.
2. Membuat matriks V dan *eigenvalue,* di mana matriks V adalah hasil dari *eigenvector* matriks A^TA.
3. Membentuk matriks S dengan cara mengurutkan nilai tertinggi *eigenvalue* kemudian diakarkan.
4. Menghitung *length* pada setiap nilai matriks V^T dengan menggunakan Rumus 4,

$$\mathrm{S_k} = \sqrt{\sum_{i=1}^{n} (V^T)_{k1}^2 . S_1^2} \tag{4}$$

5. Menentukan hasil ringkasan berdasarkan skor tertinggi dari dokumen kalimat.

Di mana S_k adalah panjang vektor k pada kalimat yang dimodifikasi oleh laten vektor. n adalah jumlah ruang demensi baru. Hasil dari *length* terbesar pada setiap dokumen kalimat akan jadikan ringkasan.

2.5 *Peringkasan menggunakan metode CLSA*

Cross Latent Semantic Analysis (CLSA) merupakan pengembangan dari algoritma terdahulu yaitu LSA. Secara bahasa *cross* memiliki arti memotong, menyembrangi atau menyilang. Maka dapat ditarik kesimpulan bahwa, CLSA menurut bahasa adalah suatu proses silang pada LSA dengan mempercepat atau menggubah beberapa proses pada LSA. Cikal bakal CLSA pertama kali diusulkan oleh Steinberger dan Jezek (2004), Steinberger dan Jezek menilai kesamaan pada topik dan signifikasi kata, akan tetapi pada penelitian Steinberger dan Jezek CLSA belum sama sekali dikenal, namun beberapa proses di LSA dalam hal ini pemanfaatan SVD mengalami perubahan yang berbeda seperti proses peringkasan yang tidak hanya dilihat dari kemiripan antar dokumen kalimat dengan judul berita, melainkan panjang dari sebuah dokumen kalimat juga menjadi faktor yang penting dalam menentukan hasil peringkasan yang lebih baik, penelitian Geetha dan Deepmala (2015) menawarakan pembanding antara LSA pada peringkasan yang buat oleh Steinberger dengan hasil eksperimenya yaitu CLSA. Pada dasarnya Geetha menambah beberapa perubahan di penelitian Steinberger untuk menemukan CLSA. Berikut langkah-langkah CLSA dalam peringkasan dokumen (Geetha & Deepmala, 2015).

1. Membentuk matriks A_{mn}
2. Menemukan *eigenvector* (matriks V) dan *eigenvalue* dari matriks A^TA.

ISSN 2502-3357 (*Online*)
G. Mandar dan G. Gunawan/Register 3 (2) 94-104 ISSN 2503-0477 (*Print*)

3. Mencari nilai *singular* (matriks *S*), dengan cara mengurutkan nilai yang paling tertinggi dan diakarkan.
4. Melakukan *Transpose* pada *eigevector* untuk membentuk matriks V^T.
5. Menghitung nilai rata-rata dari matriks V^T, seperti pada Tabel 1.
6. Melakukan seleksi pada setiap nilai matriks V^T, apabila nilai tersebut lebih kecil dari nilai rata-rata pada setiap dokumen kalimat, maka nilai pada matriks V^T diubah menjadi 0 dan membentuk matriks *V* yang baru seperti pada Tabel 2.
7. Menghitung nilai *length* pada setiap matriks V^T dengan menggunakan Rumus 4 untuk memperoleh skor dari tiap-tiap dokumen kalimat.
8. Menentukan hasil ringkasan berdasarkan skor tertinggi dari dokumen kalimat.

Tabel 1. Menghitung nilai rata-rata dari matriks VT

	Matriks V^T				**Rata-Rata**
d0	2.5752-01	3.6490-01	~~8.2269-03~~	→ 2.0619-01	0,1526
d1	~~1.0455-01~~	~~-2.3108-01~~	1.9422-01	~~9.0445-02~~	0,1864
d2	-2.5470-02	~~-2.6107-01~~	~~-4.7228-01~~	4.7610-03	-0,0831
d3	2.2282-01	~~-2.1538-01~~	1.9306-01	~~-3.0245-01~~	0,0157
d4	~~-8.9783-02~~	~~6.3092-02~~	~~-1.0721-02~~	~~1.1087-01~~	0,0691
⇩					
d12	~~8.3042-03~~	~~-3.8975-01~~	1.2999-01	1.2366-02	0,0089

Tabel 2. Matriks VT setelah diseleksi

	Matriks V^T			
	VT_0	VT_1	VT_2	VT_{12}
d0	2.5752-01	3.6490-01	0	→ 2.0619-01
d1	0	0	1.9422-01	0
d2	-2.5470-02	0	0	4.7610-03
d3	2.2282-01	0	1.9306-01	0
d4	0	0	0	0
⇩				
d12	0	0	1.2999-01	1.2366-02

Keterangan:

Nilai yang dicoret pada Tabel 1 adalah nilai yang lebih kecil dari rata-ratanya, seperti dokumen kalimat *d0* = 8.2269-03 lebih kecil dari hasil rata-rata *d0* yaitu 0,1526 maka nilai V^T tersebut diubah menjadi 0 (lihat Tabel 1 dan Tabel 2).

2.6 *Metode pengujian*

Metode pengujian hasil ringkasan dikategorikan menjadi dua yaitu metode evaluasi instrik dan metode evaluasi ekstrinsik. Instrinsik adalah evaluasi berdasarkan hasil analisis secara langsung pada ringkasan, sedangkan ekstrinsik adalah evaluasi kualitas hasil ringkasan dilandaskan pada efek apakah hasil dari ringkasan dapat membantu pada kasus yang diberikan (Winata & Rainarli, 2016).

Berdasarkan paparan sebelumnya, maka pengujian hasil ringkasan pada penelitian ini menggunakan metode evaluasi instrik yaitu metode *Precision*, *Recall* dan *F-Measure* untuk memperoleh hasil akurasi antara ringkasan pakar dengan sistem. *Recall* ialah kemampuan untuk mengambil peringkat teratas yang sebagian besar relevan (benar), *precision* adalah berapa banyak dokumen yang berhasil diambil oleh sistem, sedangkan untuk mengukur kualitas *recall* dan *precision* menggunakan *F-Measure*, seperti pada Rumus 5, Rumus 6, dan Rumus 7 (Mustaqhfiri, Abidin, & Kusumawati, 2011).

$$Precision = \frac{\text{kalimat ringkasan sistem} \cap \text{ringkasan pakar (TP)}}{\sum \text{kalimat ringkasan sistem}(TP+FP)} \quad (5)$$

$$Recall = \frac{\text{kalimat ringkasan sistem} \cap \text{ringkasan pakar(TP)}}{\sum \text{kalimat ringkasan pakar (TP+FN)}} \quad (6)$$

$$\text{F} - Measure = \frac{2*Precision*Recall}{Recall+Precision} \quad (7)$$

ISSN 2502-3357 (*Online*)
G. Mandar dan G. Gunawan/Register 3 (2) 94-104 ISSN 2503-0477 (*Print*)

Dari Rumus 5, Rumus 6, dan Rumus 7 diketahui bahwa, ada tiga komponen yang penting yaitu *True Positive* (TP) adalah jumlah dokumen kalimat yang dipilih oleh pakar, *False Positive* (FP) merupakan jumlah dokumen kalimat yang dipilih oleh sistem benar tetapi menurut pakar salah dan *False Negative* (FN) adalah jumlah dokumen kalimat yang benar menurut pakar tetapi salah menurut sistem.

3. Hasil dan Pembahasan

Pada hasil dan pembahasan penelitian ini akan memaparkan berdasarkan dua evaluasi, diantaranya berdasarkan pengujian *F-Measure*, evaluasi berdasarkan proses *preprocessing* dan kinerja metode.

3.1 Evaluasi-1 hasil berdasarkan pengujian F-Measure

Hasil peringkasan CLSA menggunakan 240 artikel berita dengan *compression rate* 30% oleh dua pakar dapat dilihat pada Tabel 3 dan Tabel 4. Pakar-1 mengahasilkan *F-Measure* CLSA 0,7255 sedangkan untuk LSA 0,6862. Pakar-1 mendominasikan hampir dokumen kalimat pertama di sebuah berita yang menjadi salah satu pilihan untuk dijadikan ringkasan. Nilai terbaik *precision* dan *recall* CLSA pada Pakar-1 adalah kategori berita regional 0,7271 dan 0,7284, namun berbeda pada Pakar-2, berita regional pada metode CLSA justru merupakan *precision* dan *recall* yang terendah yakni 0,6729 dan 0,6675. Sedangkan untuk LSA pada Pakar-1, *precision* dan *recall* kedua-duanya terdapat pada berita olahraga, hal ini serupa dengan yang dihasilkan pada LSA Pakar-2, berita olahraga menjadi *precision* yang terbaik, tapi tidak untuk *recall*, karena *recall* tertinggi terdapat pada berita internasional, sebaliknya berita internasional pada LSA Pakar-1 menjadi yang paling terendah baik *precision* maupun *recall*. Tabel 3 ini menujukan hasil yang sama antara kedua pakar, yaitu berita olahraga pada LSA Pakar-1 dan CLSA Pakar-2 menjadi yang tertinggi.

Tabel 3. Hasil pengujian Precision, Recall dan F-Measure Pakar-1

Kategori Berita	**CLSA**			**LSA**		
	Precision	*Recall*	*F-Measure*	*Precision*	*Recall*	*F-Measure*
Nasional	0,7210	0,7210	0,7210	0,6709	0,6698	0,6703
Internasional	0,7230	0,7209	0,7219	0,6608	0,6591	0,6599
Regional	0,7271	0,7284	0,7278	0,6880	0,6904	0,6892
Olahraga	0,7194	0,7194	0,7194	0,7241	0,7265	0,7253
Rata-Rata	**0,7227**	**0,7225**	**0,7255**	**0,6860**	**0,6864**	**0,6862**

Pakar-2 *F-Measure* dari CLSA dan LSA memperoleh nilai yaitu 0,7253 dan 0,7148. Tabel 4 menampilkan Pakar-2 akurasi *F-Measure* terbaik jatuh pada CLSA, di mana *precision* dan *recall* terbaik dari CLSA berada pada kategori berita olahraga, sedangkan untuk LSA, *precision* terbaik jatuh pada berita olahraga dan terendah pada berita regional, dan untuk *recall* berita internasional menjadi yang terbaik.

Tabel 4. Hasil pengujian Precision, Recall dan F-Measure Pakar-2

Kategori Berita	**CLSA**			**LSA**		
	Precision	*Recall*	*F-Measure*	*Precision*	*Recall*	*F-Measure*
Nasional	0,7258	0,7244	0,7251	0,6752	0,6728	0,6739
Internasional	0,7214	0,7235	0,7224	0,7316	0,7737	0,7327
Regional	0,6729	0,6675	0,6752	0,6869	0,6942	0,6906
Olahraga	0,7274	0,7250	0,7262	0,7638	0,7624	0,7631
Rata-Rata	**0,7254**	**0,7253**	**0,7253**	**0,7147**	**0,7148**	**0,7148**

Tabel 5. Rata-rata Precision, Recall dan F-Measure CLSA dan LSA

Kategori Berita	**CLSA**			**LSA**		
	Precision	*Recall*	*F-Measure*	*Precision*	*Recall*	*F-Measure*
Pakar-1	0,7222	0,7225	0,7255	0,6860	0,6864	0,6862
Pakar-2	0,7254	0,7253	0,7253	0,7147	0,7148	0,7148
Rata-Rata	**0,7240**	**0,7240**	**0,7240**	**0,7003**	**0,7006**	**0,7005**

ISSN 2502-3357 (*Online*)
G. Mandar dan G. Gunawan/Register 3 (2) 94-104 ISSN 2503-0477 (*Print*)

Dari kedua hasil pengujian ringkasan Pakar-1 dan Pakar-2, setelah dirata-ratakan memperoleh akurasi *F-Measure* CLSA 0,7240 sedangkan LSA 0,7005. Terlihat bahwa meskipun CLSA memiliki akurasi yang lebih tinggi dari LSA, namun hasil tersebut tidak terlalu jauh berbeda seperti pada Tabel 5. Selain itu, pada penelitian ini juga menemukan bahwa umumnya hasil ringkasan CLSA jauh lebih pendek dari pada LSA seperti pada Gambar 3.

3.2 Evaluasi-2 hasil berdasarkan preprocessing dan kinerja metode

Pada tahap ini dilakukan evaluasi pengujian dari sisi *preprocessing* dan kinerja metode yang digunakan. Tahap *preprocessing* adalah tahap yang paling penting dan berpengaruh pada hasil ringkasan, seperti pada proses penguraian teks berita dan *stopword*. Penguraian teks akan menghasilkan dokumen-dokumen kalimat yang akan diproses, dan dipilih dokumen kalimat mana saja yang diekstrak menjadi sebuah ringkasan berdasarkan skor *length* tertinggi, akan tetapi proses ini menjadi sensitif pada sistem peringkasan, dengan menggunakan *expression operations*, setiap dari teks yang dikatakan sebagai suatu kalimat adalah yang diakhiri dengan titik dan spasi, hal ini menjadi sangat sensitif apabila menemukan penelitian teks berita yang salah, misalkan setiap akhir kalimat tidak diakhiri dengan tanda titik atau spasi, maka teks tersebut tidak dapat diuraikan menjadi dokumen kalimat, sebaliknya pun demikian, misalkan "*….Kompas. Com- Propinsi papua menjadi…..*" dari contoh kalimat ini seharusnya jika diuraikan akan menjadi satu dokumen kalimat, akan tetapi penelitian yang salah pada kata "*kompas*" dan "*com*" yang seharusnya tidak terdapat spasi antara keduanya, maka yang diuraikan dari sistem adalah dokumen kalimat yang seharusnya satu menjadi dua, hal ini tentu akan menambah dokumen kalimat yang tidak sempurna dan mempengaruhi hasil ringkasan. Sama halnya dengan *stopword*, semakin banyak kata yang tersimpan dalam kamus *stopword*, maka semakin banyak pula kata dalam dokumen kalimat yang tidak terpakai dan ini tentu mempengaruhi bobot dokumen kalimat.

Tabel 6. Perubahan nilai pada vektor V^T

Matriks VT	Panjang Vektor	Jumlah *Cell* yang diubah menjadi 0
Vektor-d1	13	9
Vektor-d3	13	6
Vektor-d3	13	6
Vektor-d4	13	6
Vektor-d5	13	10
Vektor-d6	13	9
Vektor-d7	13	6
Vektor-d8	13	6
Vektor-d9	13	10
Vektor-d10	13	8
Vektor-d11	13	5
Vektor-d12	13	7
Vektor-d13	13	7

Tabel 7. Waktu interval proses perhitungan *length*

Berita	CLSA	LSA
Berita-1	0,000700 detik	0,002592 detik
Berita-2	0,002770 detik	0,000847 detik
Berita-3	0,001027 detik	0,003708 detik
Berita-4	0,001173 detik	0,002815 detik
Berita-5	0,000667 detik	0,001970 detik
Berita-6	0,000712 detik	0,003158 detik
Berita-7	0,000559 detik	0,002864 detik
Berita-8	0,001097 detik	0,001354 detik
Berita-9	0,001086 detik	0,000692 detik
Berita-10	0,000664 detik	0,000658 detik

Sedangkan pada evaluasi kinerja metode, antara CLSA dan LSA terdapat interval waktu proses perhitungan nilai *length* yang berbeda di antara keduanya, sebagaimana dijelasakan sebelumnya, bahwa CLSA melakukan seleksi pada matriks V^T menggunakan nilai rata-rata dari masing-masing

ISSN 2502-3357 (*Online*)
G. Mandar dan G. Gunawan/Register 3 (2) 94-104 ISSN 2503-0477 (*Print*)

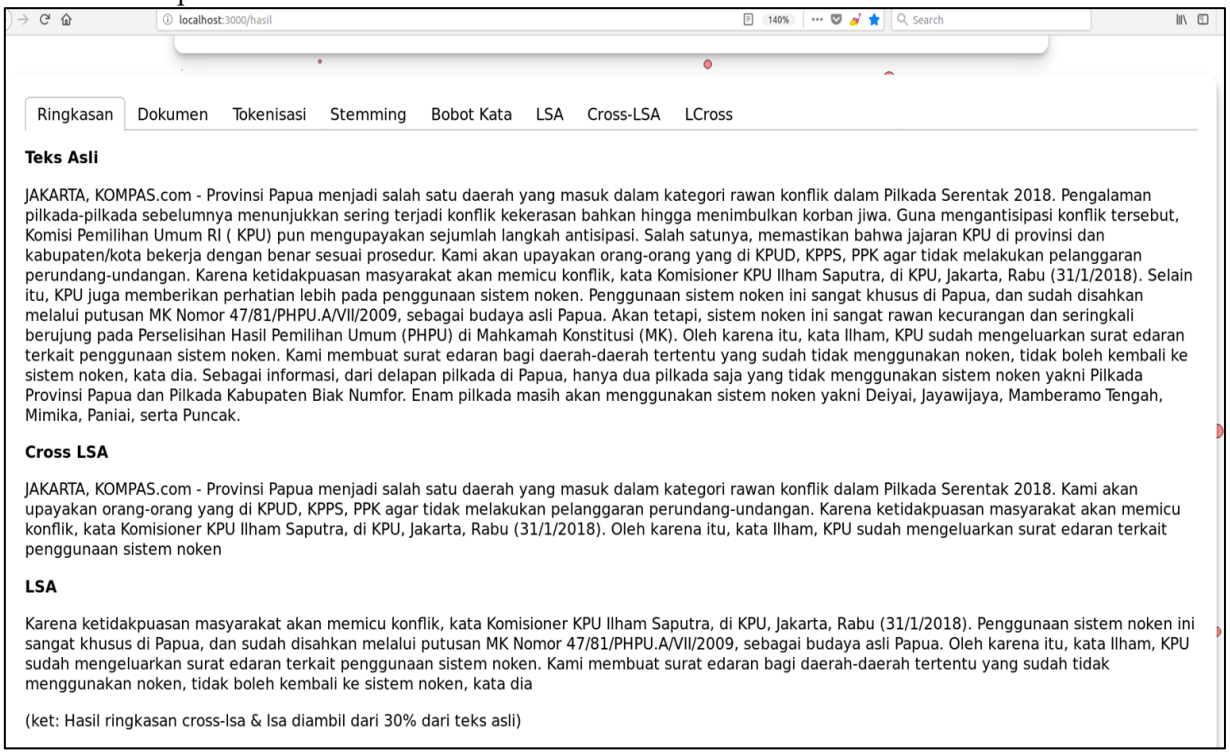
dokumen kalimat, hal ini menyebabkan banyaknya nilai *cell* pada matriks V^T berkurang, meskipun demensi tetap sama, seperti pada salah satu contoh berita yang memiliki matriks V^T 13 x 13, setelah dilakukan proses seleksi, *vektor-d1* yang awalnya terdiri panjang 13 *cell*, 9 diantara nilainya diubah menjadi 0, karena memiliki nilai yang lebih kecil dari rata-rata dokumen kalimat tersebut seperti pada Tabel 6.

Dari Tabel 6 dapat dilihat, bahwa banyak nilai pada matriks V^T yang dijadikan nol, dengan demikian hal ini akan berpengaruh baik pada proses peringkasan teks berita yang memiliki teks panjang, karena nilai dimensi pada matriks V^T setelah diseleksi jauh lebih kecil dari pada matriks V^T (tanpa seleksi) yang digunakan pada LSA. Hal ini dapat dilihat dari interval waktu proses perhitungan nilai *length* pada Tabel 7, dari 10 artikel berita hanya satu berita, di mana proses intervalnya lebih cepat LSA yaitu pada artikel berita 9, sisanya CLSA memproses lebih cepat daripda LSA.

4. Kesimpulan

Kesimpulan dari hasil peringkasan dokumen berita Bahasa Indonesia menggunakan CLSA dengan *compression rate* 30%, baik pengujian berdasarkan hasil sistem dan pakar, maupun kedua metode yaitu CLSA dan LSA, dapat disimpulkan sebagai berikut:

1. Peringkasan dokumen berita dengan 240 *dataset* menggunakan metode CLSA mampu meningkatkan hasil akurasi *F-Measure* dari metode LSA sebelumnya, walaupun peningkatan akurasi antara kedua metode tersebut tidak terlalu jauh yaitu dari 70% menjadi 72%.
2. Tahap *preprocessing* sangat berpengaruh pada kedua hasil ringkasan baik itu CLSA maupun LSA, akan tetapi bukan faktor yang mempengaruhi nilai akurasi antara CLSA dan LSA, sebab demensi dan nilai matriks A_{mn} yang digunakan keduanya sama.
3. Interval waktu proses perhitungan nilai *length* masing-masing dokumen kalimat, proses CLSA lebih cepat dari LSA.



Gambar 3. Sistem peringkasan dokumen berita Berbahasa Indonesia menggunakan CLSA

Selain hasil akurasi, ringkasan yang dihasilkan CLSA umumnya jauh lebih pendek dibandingkan LSA seperti pada Gambar 3. Namun keungulan metode CLSA pada hasil akurasi *F-Measure* belum dapat diibuktikan secara signifikan, dikarena hasil kedua metode tersebut tidak terlalu jauh berbeda. maka pada proses penelitian selanjutnya dapat ditambahkan data atau pakar sebagai pembanding hasil uji LSA dan CLSA. Penelitian ini belum mengevaluasi apakah algoritma *stemming* yang lain dapat mempengaruhi hasil akurasi, sehingga penelitian selanjutnya disarankan menggunakan algoritma

ISSN 2502-3357 (*Online*)
G. Mandar dan G. Gunawan/Register 3 (2) 94-104 ISSN 2503-0477 (*Print*)

stemming dan *compression rate* yang berbeda dari penelitian ini. Untuk menghasilkan ringkasan yang lebih baik lagi, maka penelitian selanjutnya diharapkan dapat mengembangkan hasil peringkasan CLSA ini dengan menambahakan kesamaan *similarity* antara judul berita dan dokumen kalimat yang dipilih oleh CLSA menggunakan metode *Cosine Similarity* atau metode lainnya yang sesuai.

4. Referensi

Asian, J. (2007). *Effective Techniques for Indonesian Text Retrieval.* Melbourne: RMIT University.

Badry, R. M., Eldin, A. S., & Elzanfally, D. S. (2013). Text Summarization within the Latent Semantic Analysis Framework: Comparative Study. *International Journal of Computer Applications, 81*(11), 40-45.

Das, D., & Martins, A. F. (2007). A Survey on Automatic Text Summarization. *Literature Survey for the Language and Statistics II course at CMU*, 192-195.

Geetha, J. K., & Deepamala, N. (2015). Kannada text summarization using Latent Semantic Analysis. *International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (pp. 1508-1512). Pune: IEEE.

Gong, Y., & Liu, X. (2001). Generic text summarization using relevance measure and latent semantic analysis. *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 19-25). New Orleans: ACM.

Gotami, N. S., Indriati, I., & Dewi, R. K. (2018). Peringkasan Teks Otomatis Secara Ekstraktif Pada Artikel Berita Kesehatan Berbahasa Indonesia Dengan Menggunakan Metode Latent Semantic Analysis. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 2*(9), 2821-2828.

Gunawan, F. E., Juandi, A. V., & Soewito, B. (2015). An automatic text summarization using text features and singular value decomposition for popular articles in Indonesia language. *2015 International Seminar on Intelligent Technology and Its Applications (ISITIA)* (pp. 27-32). Surabaya: IEEE. doi:10,1109/ISITIA.2015.7219948

Mustaqhfiri, M., Abidin, Z., & Kusumawati, R. (2011). Peringkasan teks otomatis berita berbahasa Indonesia menggunakan metode Maximum Marginal Relevance. *MATICS, 4*(4), 134-147.

Najibullah, A., & Mingyan, W. (2015). Otomatisasi peringkasan dokumen sebagai pendukung sistem manajemen surat. *Register: Jurnal Ilmiah Teknologi Sistem Informasi, 1*(1), 1-6.

Ozsoy, M. G., Cicekli, I., & Alpaslan, F. N. (2010). Text summarization of Turkish texts using latent semantic analysis. *Proceedings of the 23rd International Conference on Computational Linguistics* (pp. 869-876). Beijing: ACM.

Steinberger, J., & Ježek, K. (2004). Using Latent Semantic Analysis in Text Summarization and Summary Evaluation. *Proc. ISIM '04*, (pp. 93–100).

Torres-Moreno, J.-M. (2014). *Automatic text summarization* (Vol. 5). Hoboken: Wiley-ISTE.

Viva, T. (2016, Maret 16). *Riset: Konsumsi Berita Online Kalahkan Televisi*. Retrieved from Viva: https://www.viva.co.id/digital/digilife/748454-riset-konsumsi-berita-online-kalahkan-televisi

Winata, F., & Rainarli, E. (2016). Implementasi Cross method Latent Semantic Analysis untuk meringkas dokumen berita Berbahasa Indonesia. *Techno.Com, 15*(4), 266-277.

Zeniarja, J., Salam, A., Luthfiarta, A., Handoko, L. B., & Jamhari, M. (2013). Integrasi peringkas dokumen otomatis dengan penggabungan metode fitur dan metode Latent Semantic Analysis (LSA) sebagai Feature Reduction. *Seminar Nasional Teknologi Informasi & Komunikasi Terapan 2013 (SEMANTIK 2013)* (pp. 191-197). Semarang: Universitas Dian Nuswantoro.