IMPLEMENTASI CROSS METHOD LATENT SEMANTIC ANALYSIS UNTUK MERINGKAS DOKUMEN BERITA BERBAHASA INDONESIA

Fernando Winata[1], Ednawati Rainarli[2]
[1,2]Teknik Informatika, Teknik dan Ilmu Komputer, Universitas Komputer Indonesia
Jalan Dipatiukur No. 112-116, Coblong, Bandung, Jawa Barat 40132, Indonesia
E-mail : winata.nando@gmail.com[1], ednawati.rainarli@email.unikom.ac.id[2]

Abstrak

Penelitian mengenai peringkasan teks secara otomatis sampai saat ini masih terus dilakukan dengan harapan hasil ringkasan yang dihasilkan oleh mesin dapat mendekati ringkasan yang dihasilkan oleh manusia. Salah satu metode yang digunakan untuk menghasilkan ringkasan dengan bantuan mesin adalah metode Latent Semantic Analysis (LSA) yang menerapkan konsep Singular Value Decomposition untuk pemilihan ringkasan yang dihasilkan, tetapi dari beberapa hasil pengujian yang telah dilakukan terhadap metode ini dapat diketahui bahwa tingkat akurasi dari ringkasan yang dihasilkan masih dapat ditingkatkan kembali. Cross Method Latent Semantic Analysis (CMLSA) merupakan pengembangan dari metode LSA yang dianggap dapat menghasilkan ringkasan dengan tingkat akurasi yang lebih tinggi dari metode LSA. Untuk menguji seberapa baik ringkasan yang dihasilkan maka dibuatlah simulator peringkas teks otomatis dengan menggunakan CMLSA sebagai metode peringkasan sedangkan teks yag digunakan sebagai masukan adalah teks berita yang diambil dari portal berita viva.co.id. Dari hasil penelitian ini dapat diketahui bahwa rata – rata nilai performansi dari ringkasan yang dihasilkan dengan menggunakan metoede pengujian Precision, Recall dan F-Measure adalah nilai Precision sebesar 72,25%, nilai Recall sebesar 66,7% dan nilai F-Measure sebesar 69,6%, hasil ringkasan dengan menggunakan metode ini dengan akurasi ringkasan sebesar 69,6% dianggap cukup baik.

Kata Kunci: *cross method latent semantic analysis, TF-IDF, automatic text summarization, natural language processing.*

Abstract

The research on extracting summary from document automaticly still being researched with the expectation that the extracted summary using machine help can be close to the summary extracted by human. One of the method that used to extracting summary is called Latent Semantic Analysis (LSA), this method used the concept of Singular Value Decomposition as its base on extracting summary, but from the result of researchs that has been done to this method it is known that this method still can be further upgraded to make the accuracy of the extracted summary better. Cross Method Latent Semantic Analysis (CMLSA) is one of the upgraded version of LSA with the ability to extract summary better than using LSA. Automatic text summarization simulator is being builded for testing the accuracy of the extracted summary with the use of CMLSA for the method of extracting summary and the text that used as an input is news article that obtained from viva.co.id website. The result from this research are the accuracy of extracted summary using Precision, Recall and F-Measure testing method are Precison method value is 72,5%, Recall method value is 66,7% and F-Measure method value is 69,6%. From these result, we can know that the extracted summary using CMLSA can be considered good.

Keywords: *cross method latent semantic analysis, TF-IDF, automatic text summarization, natural language processing.*

1. PENDAHULUAN

Peringkas teks otmatis merupakan sebuah teknik untuk mengambil ringkasan dengan menggunakan bantuan mesin. Penggunaan peringkas teks otomatis diharapkan dapat membantu manusia untuk mendapatkan ringkasan atau informasi inti dari sebuah dokumen dengan akurat dan cepat. Pada tahun 2001 Yi Gong dan Xi Liu [1] mengenalkan sebuah metode baru yang dapat diterapkan pada peringkas teks otomatis dengan menggunakan konsep *Singular Value Decomposition* (SVD) yang diberi nama *Latent Semantic Analysis* (LSA). Beberapa penelitian yang berkaitan dengan peringkas teks otomatis terutama penggunaan LSA sebagai metode penghasil ringkasan telah banyak dilakukan sebelumnya.

Pada penelitian yang dilakukan Steinberg dan kawan – kawan [2] diketahui bahwa terdapat kekurangan dari metode LSA yaitu pada jumlah dimensi dari matriks yang dihasilkan dengan konsep SVD yang dapat mempengaruhi ringkasan yang dihasilkan, sehingga pada penelitian ini dilakukan perbaikan pada masalah yang ditemukan untuk menghasilkan ringkasan yang lebih baik. Penelitian yang dilakukan oleh Murray dan kawan – kawan [3] yang melakukan proses modifikasi pada tahapan reduksi matriks serta modifikasi pada tahap pengambilan ringkasan yang menghasilkan ringkasan yang lebih akurat dibanding dengan metode LSA sebelumnya[4].

Sedangkan penelitian yang terkait untuk penerapan LSA pada artikel berbahasa Indonesia juga telah banyak dilakukan. Penelitian Junta Zeniarja dan kawan – kawan [5] berhasil meningkatkan tingkat akurasi yang dihasilkan dari multi document dengan cara menggabungkan algoritma *latent semantic analysis* dengan algoritma *clustering*. Pada penelitian Adiwijaya dan kawan – kawan [6] dapat membuktikan bahwa metode *latent semantic analysis* mempunyai tingkat akurasi ringkasan yang lebih tinggi dibanding dengan metode *relevance measure.*

Dari beberapa penelitian yang telah dilakukan, baik untuk pengujian maupun pengembangan dari metode LSA dapat diketahui bahwa metode ini masih dapat dikembangkan untuk menghasilkan ringkasan dengan tingkat akurasi yang lebih baik. Salah satu hasil pengembangan dari metode LSA yang dianggap dapat menghasilkan ringkasan yang lebih akurat dibandingkan dengan metode LSA dan hasil pengembangan metode LSA yang lain adalah *Cross Method Latent Semantic Analysis* (CMLSA) [4]. CMLSA merupakan pengembangan dari perbaikan metode LSA Steinberg dan kawan – kawan pada tahun 2004 yang menambahkan perbaikan pada tahap ekstraksi kalimat ringkasan yang dilakukan untuk meningkatkan tingkat akurasi ringkasan yang dihasilkan.

Peringkas teks otomatis diharapkan dapat menghasilkan ringkasan mendekati ringkasan yang dihasilkan oleh manusia dengan akurat dan cepat. Penerapan *cross method latent semantic analysis* pada peringkas teks otomatis diharapkan dapat menghasilkan ringkasan yang lebih akurat dibanding dengan ringkasan yang dihasilkan metode - metode *Latent Semantic Analysis* terdahulu [4].

2. METODE PENELITIAN

Pada bab ini akan membahas tentang teori apa saja yang dapat mendukung implementasi *Cross Metod Latent*

Semantic Analysis pada peringkas teks otomatis

2.1 Peringkas Teks Otomatis

Ringkasan adalah sebuah teks yang dihasilkan dari sebuah dokumen atau lebih yang menyatakan informasi penting dari dokumen asli, dan sebuah ringkasan memiliki ukuran yang relatif lebih pendek dari dokumen asli [7]. Tujuan utama dari ringkasan adalah menghasilkan sebuah intisari atau informasi inti yang terdapat dalam dokumen asli dalam bentuk yang lebih kecil agar pembaca dapat mendapatkan informasi penting yang terdapat dalam dokumen dengan lebih cepat. Proses peringkasan teks secara otomatis sendiri merupakan proses peringkasan pada suatu dokumen dengan menggunakan bantuan komputer.

Menurut Andre F.T dan kawan – kawan [7], terdapat dua buah pendekatan yang dilakukan untuk menghasilkan sebuah ringkasan secara otomatis yaitu :

a. Ekstraksi
   Pada teknik ekstraksi, sistem menyalin unit-unit teks yang dianggap paling penting dari teks sumber menjadi ringkasan. Unit-unit teks yang disalin dapat berupa klausa utama, kalimat utama, atau paragraf utama tanpa ada penambahkan kalimat-kalimat baru yang tidak terdapat pada dokumen aslinya.
b. Abstraksi
   Teknik abstraksi menggunakan metode linguistik untuk memeriksa dan menafsirkan teks dokumen menjadi ringkasan. Ringkasan teks tersebut dihasilkan dengan cara menambahkan kalimat-kalimat baru yang merepresentasikan intisari teks sumber ke dalam bentuk yang berbeda dengan kalimat-kalimat yang ada pada teks sumber.

Pada penelitian ini teknik peringkasan teks yang digunakan adalah teknik peringkasan secara ekstraksi dengan menggunakan mesin atau komputer sebagai alat bantu untuk menghasilkan ringkasan

2.2 Preprocessing

Preprocessing merupakan tahapan awal untuk menghasilkan sebuah ringkasan. Teks masukan yang akan di ringkas terlebih dahulu harus melalui tahap untuk membuang berbagai macam jenis *noise* atau kata – kata yang di anggap tidak penting dalam ringkasan yang masih terdapat pada teks masukan [8].

Dalam penerapan *preprocessing* terdapat beberapa tahap yang harus dilalui dimulai dari tahapan tokenisasi, penghilangan *stop words* sampai dengan tahapan *stemming,* selain tahapan itu juga biasanya di tambahkan beberapa tahapan lain untuk kasus tertentu seperi *case folding* dan penghilangan kata yang jarang dimunculkan atau kata dengan frekuensi kemunculan yang kecil [9].

Proses *preprocessing* yang diterapkan pada penelitian ini meliputi beberapa macam tahap yaitu:

a. Pemecahan Kalimat
   Pada tahapan ini teks masukan dipecah menjadi beberapa kalimat berdasarkan delimiter atau pemisah yang sudah ditetapkan. Pemisah tersebut adalah tanda titik(.),tanda seru(!) dan tanda Tanya(?).
b. *Case Folding*
   Pada tahapan ini dilakukan proses penyamaan *case* atau besar kecil dari setiap huruf yang terdapat pada teks masukan yang telah di pecah menjadi beberapa kalimat. Selain itu pada tahap ini juga di lakukakan proses pembuangan pada tanda baca,simbol dan angka yang terdapat pada data masukan.

c. *Tokenizing*
Pada tahapan ini teks masukan hasil dari proses *case folding* dipecah kembali menjadi beberapa kalimat berdasarkan spasi yang terdapat pada kalimat hasil *case folding*.

d. *Stop Words Removal*
Pada tahapan ini dilakukan proses penghapusan *stop words* yang masih terdapat pada teks hasil dari proses *tokenizing*. *Stop words* merupakan sebuah kata yang dianggap tidak terlalu penting dalam proses peringkasan atau kata yang tidak mempunyai arti dalam sebuah dokumen[10]. Contoh dari *stop words* dalam bahasa Indonesia adalah : dan,atau,jika dan sebagainya. Tujuan utama dilakukan tahap ini adalah untuk mengurangi dimensi dari teks masukan sehingga proses peringkasan dapat berjalan dengan lebih mudah [8].

e. *Stemming*
Stemming merupakan proses yang menyediakan pemetaan varian morfologi yang berbeda dari suatu kataker akar katanya dengan cara [10]. proses ini melakukan pemetaan dari penguraian berbagai bentuk kata baik itu *prefix*, *sufix*, maupun gabungan antara *prefix* dan *sufix* (*confix*), menjadi bentuk kata dasarnya. Pada penelitian ini algoritma stemming yang digunakan adalah algoritma stemming Nazief dan Adriani.

2.3 Algoritma Term Frequency – Inverse Document Frequency (TF-IDF)

Metode *Term Frequency-Inverse Document Frequency* (TF-IDF) adalah cara pemberian bobot hubungan suatu kata (*term*) terhadap dokumen. Untuk dokumen tunggal tiap kalimat dianggap sebagai dokumen. Metode ini menggabungkan dua konsep untuk perhitungan bobot, yaitu *Term frequency* (TF) merupakan frekuensi kemunculan kata (*t*) pada kalimat (*d*). *Document frequency* (DF) adalah banyaknya kalimat dimana suatu kata (*t*) muncul.

Frekuensi kemunculan kata di dalam dokumen yang diberikan menunjukkan seberapa penting kata itu di dalam dokumen tersebut. Frekuensi dokumen yang mengandung kata tersebut menunjukkan seberapa umum kata tersebut[4]. Bobot kata semakin besar jika sering muncul dalam suatu dokumen dan semakin kecil jika muncul dalam banyak dokumen. Pada aloritma TF-IDF digunakan rumus untuk menghitung bobot (W) masing masing dokumen terhadap kata kunci dengan persamaan :

$$W = tf * IDF \quad (1)$$

Keterangan:

d : dokumen ke-d
t : kata ke-t dari kata kunci
W : bobot dokumen ke-d terhadap kata ke-t
tf : banyaknya kata yang dicari pada sebuah dokumen
IDF : Inversed Document Frequency

Untuk nilai IDF dapat di cari dengan persamaan :

IDF =**Error! Reference source not found.** (2)

Keterangan:

N : total dokumen
df : banyak dokumen yang mengandung kata yang dicari

2.4 Latent Semantic Analysis

Latent Semantic Analysis (LSA) adalah suatu metode untuk mengekstrak sebuah tulisan dalam suatu dokumen dan kemudian mengaplikasikannya dalam perhitungan matematis. Penilaian dengan metode LSA lebih kepada kata-

kata yang ada dalam tulisan tanpa memperhatikan urutan kata dan tata bahasa dalam tulisan tersebut, sehingga suatu kalimat yang dinilai adalah berdasarkan kata-kata kunci yang ada pada kalimat tersebut [1].

Metode ini terinspirasi dari penggunaan latent semantic indexing yang mengimplementaasikan metode singular value decomposition (SVD) untuk menghasilkan sebuah ringkasan. Terdapat tiga tahapan utama dalam proses menghasilkan ringkasan menggunakan metode ini, yaitu :

a. Pembuatan matriks
b. Singular value decomposition
c. Ekstraksi kalimat ringkasan

2.4.1 Pembuatan Matriks

Pada tahap ini dibuat sebuah matriks berdasarkan dengan teks masukan yang akan di ringkas. Matriks yang dibuat berupa matriks dengan kata dari artikel berita dijadikan sebagai baris dan kalimat dalam artikel berita dijadikan sebagai kolom, untuk nilai atau isi dari matriks tersebut di ambil berdasarkan nilai dari bobot setiap kata yang terdapat pada artikel yang di dapatkan dari tahapan pembobotan kata menggunakan Algoritma *Term Frequency – Inverse Document Frequency* (TF-IDF).

2.4.2 Singular Value Decomposition

Setelah matriks di ciptakan maka tahap selanjutnya adalah mengubah matriks tersebut menjadi lebih kecil dengan menggunakan metode *singular value decomposition* (SVD). Suatu proses dekomposisi akan memfaktorkan sebuah matriks menjadi lebih dari satu matriks. Demikian halnya dengan Dekomposisi Nilai Singular (*Singular Value Decomposition*) atau yang lebih dikenal sebagai SVD, adalah salah satu teknik dekomposisi berkaitan dengan nilai singular (*singular value*) suatu matriks yang merupakan salah satu karakteristik matriks tersebut [11]. Dekomposisi nilai singular matriks riil A mxn adalah faktorisasi.

$$A_{mxn} = USV^T \quad (3)$$

Dengan U matriks orthogonal m x m, V matriks orthogonal n x n dan S matriks diagonal m x n bernilai riil tak negatif yang disebut nilai-nilai singular. Dengan kata lain S = diag ($\sigma 1, \sigma 2, \ldots, \sigma n$) terurut sehingga $\sigma 1 \geq \sigma 2 \geq \ldots \geq \sigma n$ [11].

Matriks A_{mxn} dapat dinyatakan sebagai dekomposisi matriks yaitu matriks U, S dan V . Matriks S merupakan matriks diagonal dengan elemen diagonalnya berupa nilai-nilai singular matriks A, sedangkan matriks U dan V merupakan matriks-matriks yang kolom-kolomnya berupa vektor singular kiri dan vektor singular kanan dari matriks A untuk nilai singular yang bersesuaian [11].

Menentukan SVD meliputi langkah-langkah menentukan nilai eigen dan vektor eigen dari matriks AA^T atau A^TA. Vektor eigen dari A^TA membentuk kolom V, sedangkan vektor eigen dari AA^T membentuk kolom U. Nilai singular dalam S adalah akar pangkat dua dari nilai-nilai eigen matriks AA^T atau A^TA. Nilai singular adalah elemen-elemen diagonal dari S dan disusun dengan urutan menurun [11].

2.4.3 Ekstraksi Kalimat Ringkasan

Pada proses ini dilakukan tahap pemilihan kalimat yang akan di jadikan sebagai ringkasan. Kalimat yang dipilih diambil dari kalimat yang terdapat pada matriks V^T. Kemudian dilakukan pemilihan kalimat yang akan di jadikan sebagai ringkasan berdasarkan kalimat yang mengandung bobot kata terbesar. Proses pemilihan diulang sebanyak jumlah kalimat yang terdapat dalam

matriks.Contoh dari proses ekstraksi menggunakan metode ini adalah sebagai berikut.

2.5 Cross Method Latent Semantic Analysis

Cross method latent semantic analysis merupakan sebuah pegembangan dari metode *latent semantic analysis* yang telah ada sebelumnya. Metode ini dapat menghasilkan ringkasan dari teks masukan yang lebih akurat dibandingkan dengan metode *latent semantic analysis* yang sebelumnya [4]. Tahapan dari metode ini sebenarnya sama seperti metode *Latent Semantic Analysis* sebelumnya yaitu dimulai dari tahap pembuatan matriks, *Singular Value Decomposition* dan ekstraksi ringkasan.

Yang menjadi perbedaan metode ini dengan metode *latent semantic analysis* terdapat pada saat tahap ekstraksi ringkasan. Metode ini menggunakan nilai rata – rata (*average*) dan panjang (*length*) yang di ambil dari matriks V^T dan matriks S. Nilai *average* diambil dari nilai rata – rata dari setiap bobot kata yang terdapat baris matriks V^T. setelah ditemukan nilai rata – rata dari setiap kata yang terdapat pada matriks V^T, langkah selanjutnya adalah mencocokan nilai rata-rata yang didapatkan pada setiap baris dengan nilai dari setiap kata yang terdapat pada baris tersebut. Jika ternyata nilai dari kata tersebut lebih rendah dari nilai rata-rata yang didapat, maka nilai kata tersebut di ubah menjadi nol. Tetapi jika tidak maka nilai dari kata tersebut tetap.

Setelah tahapan pencarian dan pencocokan nilai rata – rata telah dilakukan, tahap selanjutnya adalah menghitung *length* dari setiap baris dari matriks V^T dengan rumus :

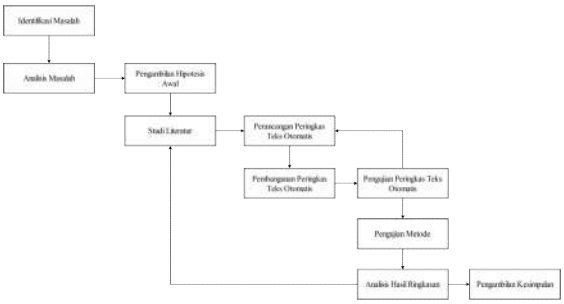
$$length = \sqrt{\sum_{j=1}^{n} V_{ij}^2 * S_{jj}^2} \qquad (4)$$

Dimana :
i : baris matriks
j : kolom matriks

Baris - baris pada matriks atau kalimat - kalimat yang mempunyai nilai *length* yang tinggi akan di jadikan sebagai ringkasan.

2.6 Tahapan Penelitian

Tahapan penelitian yang dilakukan pada penelitian ini dapat dilihat pada gambar 1 berikut:



Gambar 1: Tahapan Penelitian

Penjelasan metode penelitian yang digunakan pada gambar 1 adalah sebagai berikut :

a. Identifikasi Masalah
   Berdasarkan latar belakang yang telah disebutkan, permasalahan yang dapat di identifikasi adalah kebutuhan masyarakat akan peringkas teks otomatis yang dapat menghasilkan ringkasan dengan cepat dan akurat. Serta penggunaan metode pada peringkas teks otomatis yang dapat menghasilkan ringkasan dengan tingkat akurasi tinggi.
b. Analisis Masalah
   Dari permasalahan yang telah di identifikasi didapatkan bahwa salah satu metode yang digunakan untuk menghasilkan ringkasan pada peringkas teks otomatis untuk artikel berita berbahasa Indonesia adalah metode *Latent Semantic*

Analysis (LSA). Tetapi metode LSA yang diterapkan pada bebrapa penelitian untuk menghasilkan ringkasan pada dokumen berbahasa Indonesia menggunakan metode LSA yang lama sedangkan metode LSA telah mengalami beberapa tahap pengembangan. Sehingga ringkasan yang dihasilkan oleh metode LSA yang lama memiliki tingkat akurasi yang lebih rendah jika dibandingkan dengan metode LSA yang telah dikembangkan.

c. Pengambilan Hipotesis Awal
Dari hasil analisis masalah di atas dapat diambil sebuah hipotesis awal yaitu, untuk menghasilkan sebuah peringkas teks otomatis metode yang dapat digunakan adalah *Cross Method Latent Semantic Analysis* yang dapat menghasilkan sebuah ringkasan yang lebih akurat dibandingkan dengan metode *latent semantic analysis* biasa.

d. Studi Literatur
Pada tahap ini dilakukan studi terhadap literatur – literatur yang berkaitan dengan penelitian yang dilakukan seperti literature mengenai peringkas teks otomatis, proses pengolahan teks, dan *cross method latent semantic analysis*, serta literature – literatur lain yang mendukung penelitian.

e. Perancangan Simulator
Pada tahap ini dilakukan proses perancangan peringkas teks otomatis. Proses perancangan terdiri dari proses analisis kebutuhan fungsional dan non fungsional dari peringkas teks otomatis yang akan dibangun, analisis data masukan, perancangan tampilan antar muka peringkas teks otomatis dan perancangan jaringan semantik.

f. Pembangunan Peringkas Teks Otomatis
Pada tahap ini dilakaukan pembangunan dari peringkas teks otomatis untuk teks berbahasa Indonesia. Dimulai dari tahap pembangunan *interface* peringkas teks otomatis, tahapan *preprocessing*(pemecahan kalimat, *Case Folding, Tokenizing, Stop Words Removal* dan *Stemming*), pembobotan kata menggunakan metode *Term Frequency – Inverse Document Frequency* (TF-IDF) dan implementasi C*ross Method Latent Semantic Analysis* untuk menghasilkan ringkasan.

g. Pengujian Peringkas Teks Otomatis
Pada tahapan ini, akan dilakukan proses pengujian terhadap peringkas teks otomatis yang telah di bangun untuk melihat apakah peringkas teks otomatis dapat berjalan dengan semestinya. Jika peringkas teks otomatis yang dibangun memiliki kekurangan maka akan kembali ke tahapan perancangan peringkas teks otomatis untuk melakukan analisis kesalahan yang mungkin terjadi pada saat pembangunan peringkas teks otomatis. Jika peringkas teks otomatis sudah dapat berjalan dengan semestinya maka akan masuk ke dalam tahapan pengujian metode.

h. Pengujian Metode
Dalam tahapan ini akan dilakukan pengujian terhadap *Cross Method Latent Semantic Analysis* untuk mengetahui hasil dari implementasi metode ini dalam peringkas teks otomatis. Jika dari tahapan ini memberikan hasil yang kurang baik maka akan kembali ke tahapan studi literatur untuk menganalisis kesalahan yang mungkin terjadi pada saat penerapan metode.

i. Analisis Hasil Ringkasan
Pada proses ini dilakukan analisis pada hasil ringkasan yang didapatkan dengan menggunakan metode *Latent Semantic Analysis*. Untuk proses analisis digunakan

metode *Precision,Recall* serta metode *F-Measure* untuk mengetahui tingkat ke akuratan ringkasan yang dihasilkan dari *Cross Method Latent Semantic Analysis.*

j. Pengambilan Kesimpulan
Pada tahap ini dilakukan pengambilan kesimpulan yang didapat dari hasil ringkasan yang dihasilkan oleh peringkas teks otomatis untuk teks berita berbahasa Indonesia.

2.7 Pengumpulan Data dan Analisa Data

Pada penelitian ini terdapat tiga jenis data yang digunakan dalam penelitian, yaitu data berita, data hasil ringkasan peringkas teks otomatis dan data hasil ringkasan manual atau manusia.

Data berita merupakan berita - berita yang akan digunakan sebagai masukan atau data yang akan di ringkas secara otomatis menggunakan bantuan mesin (peringkas teks otomatis) dan secara manual oleh manusia. Data berita yang digunakan merupakan data berita bertema politik yang diambil dari situs berita *online* viva.co.id. Dalam penelitian ini digunakan enam buah berita bertema politik sebagai data berita.

Data ringkasan peringkas teks otomatis merupakan data dari hasil ringkasan yang dihasilkan oleh mesin dengan menggunakan Algoritma *Cross Method Latent Semantic Analysis* dengan data yang di ringkas merupakan data berita politik dari situs viva.co.id yang telah di ambil sebelumnya. Jumlah dari kalimat yang dapat dijadikan ringkasan oleh mesin di batasi, yaitu hanya 35% kalimat dari keseluruhan isi berita pada setiap berita yang akan di ringkas. Hasil dari data ringkasan adalah kalimat - kalimat yang dipilih sebagai ringkasan pada setiap berita oleh mesin serta jumlah dari kalimat ringkasan yang dihasilkan oleh mesin pada setiap berita yang di ringkas.

Data ringkasan manual merupakan hasil ringkasan yang diperoleh secara manual oleh manusia, pada penelitian ini data ringkasan manual dihasilkan oleh sepuluh orang dari berbagai latar belakang, hal ini dilakukan dengan harapan bahwa hasil ringkasan yan g dihasilkan beragam. Data dihasilkan dengan meggunakan bantuan kuesioner dimana kuesioner yang digunakan berisi berita yang akan di ringkas yang telah dipotong per kalimat dalam sebuah tabel sehingga untuk menentukan kalimat mana yang dipilih sebagai ringkasan pengambil ringkasan hanya perlu mencoret pada nomor kalimat yang dianggap memliki informasi penting dalam berita tersebut. Jumlah kalimat yang dapat dipilih sebagai ringkasan tidak ditentukan sehingga ada kemungkinan jumlah kalimat yang dipilih oleh peringkas satu dengan peringkas lainnya berbeda, hal ini dilakukan karena peneliti memiliki anggapan bahwa hasil ringkasan secara manual yang dihasilkan oleh setiap orang berbeda - beda. Hasil dari ringkasan manual merupakan kalimat - kalimat yang dipilih sebagai ringkasan dan jumlah kalimat yang dipilih sebagai ringkasan pada setiap berita yang di ringkas.

2.8 Metode Pengujian Hasil Ringkasan

Hasil ringkasan yang dihasilkan oleh peringkas teks otomatis selanjutnya harus melalui tahapan pengujian dan evaluasi untuk mengetahui tingkat akurasi dan ketepatan hasil ringkasan yang dihasilkan. Proses evaluasi hasil ringkasan dapat dikategorikan menjadi dua yaitu metode evaluasi intrinsik dan metode evaluasi ekstrinsik [12].

Pada proses evaluasi secara ekstrinsik, kualitas dari hasil ringkasan dilandaskan pada efek apakah hasil dari ringkasan dapat membantu pada kasus yang diberikan. Sedangkan pada proses evaluasi secara intrinsik, kualitas dari hasil ringkasan berdasarkan dari hasil analisis yang dilakukan pada ringkasan secara langsung. Pada kasus evaluasi intrinsik hasil ringkasan.

dibandingkan dengan dokumen asli, dari situ akan di analisis seberapa banyak ide utama pada dokumen asli yang terdapat pada hasil ringkasan dengan menyamakan hasil ringkasan dengan hasil ringkasan abstrak atau hasil ringkasan manual yang dilakukan oleh manusia [13].

Pada penelitian ini proses evaluasi yang akan digunakan adalah proses evaluasi secara intrinsik dengan menggunakan metode *precision*, *recall* dan *f -measure* . Nilai *f - measure* dihasikan berdasarkan nilai *precision* dan *recall*. Metode evaluasi ini merupakan metode evaluasi yang sering digunakan dalam proses evaluasi hasil ringkasan. Dalam metode intrinsik, *precision* dan *recall* digunakan untuk mengukur kualitas ringkasan sistem dengan cara membandingkan ringkasan sistem dengan ringkasan manual (buatan manusia). *Precision* adalah tingkat ketepatan hasil ringkasan yang dihasilkan peringkas teks otomatis sedangkan *recall* adalah tingkat keberhasilan ringkasan yang dihasilkan peringkas teks otomatis.Untuk menghitung nilai *precision* dan nilai *recall* digunakan persamaan berikut [13]:

$$Precision = \frac{\sum kalimat\ relevan}{\sum kalimat\ ringkasan\ sistem} \quad (5)$$

$$Recall = \frac{\sum kalimat\ relevan}{\sum kalimat\ ringkasan\ manual} \quad (6)$$

Kalimat relevan adalah kalimat – kalimat ringkasan yang dihasilkan oleh peringkas teks otomatis yang sama dengan kalimat – kalimat ringkasan yang dihasilkan secara manual oleh manusia. Untuk mencari kalimat relevan digunakan persamaan:

Kalimat relevan = **Error! Reference source not found.** (7)

Dimana :
KRS : Kalimat Ringkasan Sistem
KRM : Kalimat Ringkasan Manual

Setelah diketahui nilai *precision* dan nilai *recall*, tahapan selanjutnya adalah menghitung nilai dari *f – measure* yang merupakan nilai yang digunakan untuk mengukur nilai akurasi dari ringkasan yang dihasilkan dengan menggunakan nilai yang dihasilkan pada perhitungan *precision* dan *recall* sebelumnya, sehingga *f-measure* bisa juga disebut sebagai gabungan atau kombinasi nilai *precision* dan *recall* [15]. *F-measure* dapat dicari dengan menggunakan persamaan berikut:

$$F - Measure = 2 * \frac{Precision * Recal}{(Presicion + Recall)} \quad (8)$$

3.HASIL DAN PEMBAHASAN

3.1 Skenario Pengujian

Pada bagian ini akan dipaparkan tindakan yang akan dilakukan selama proses pengujian terhadap peringkas teks otomatis berlangsung. Pada penelitian ini, akan dilakukan proses pengujian tingkat akurasi dari ringkasan yang dihasilkan oleh manusia atau manual dan hasil ringkasan dari.. Hasil dari ringkasan manual diperoleh dari sepuluh orang penguji dari berbagai macam kalangan, hal ini dilakukan untuk mendapatkan hasil ringkasan manual yang beragam dengan catatan penguji yang dipilih dianggap dapat memberikan atau mengambil informasi

inti dari berita yang berikan dengan baik.

Data yang digunakan dalam penelitian ini adalah Artikel berita atau dokumen sumber berupa artikel berita bertema politik yang bersumber dari portal berita viva.co.id dengan jumlah artikel berita yang digunakan berjumlah enam buah dokumen. Jumlah ringkasan yang diambil baik ringkasan manual maupun ringkasan peringkas teks otomatis sebesar 35% dari banyak kalimat di setiap dokumen uji tetapi untuk ringkasan manual jika terdapat kalimat yang memiliki jumlah frekuensi kemiripan yang sama, maka seluruh kalimat dengan frekuensi yang sama tersebut dipilih sebagai hasil ringkasan manual. Hasil ringkasan manual untuk setiap dokumen uji yang diambil dari sepuluh orang penguji merupakan kalimat dengan tingkat frekuensi pemilihan kalimat sebagai ringkasan tertinggi di setiap dokumen uji.

3.2 Hasil Ringkasan Manual

Hasil ringkasan dari enam buah dokumen artikel berita politik berbahasa Indonesia yang di dapatkan dari sepuluh orang penguji yang berbeda dapat dilihat pada tabel 1 di bawah :

Tabel 1: Hasil Ringkasan Manual

Dokumen	Jumlah Kalimat	Jumlah Ringkasan Manual	Indeks Kalimat
1	12	4	2,3,6,8
2	12	4	1,2,7, 12
3	10	3	1,3,7
4	7	2	1,2
5	11	4	1,2,4,11
6	9	3	1,2,8

3.3. Hasil Ringkasan Peringkas Teks Otomatis

Hasil ringkasan dari 6 buah dokumen artikel berita politik berbahasa Indonesia yang dihasilkan oleh peringkas teks otomatis dapat dilihat pada tabel 2 dibawah :

Tabel 2: Hasil Ringkasan Peringkas Teks Otomatis

Dokumen	Jumlah Kalimat	Jumlah Ringkasan Peringkas Teks Otomatis	Indeks Kalimat
1	12	3	1,2,3
2	12	3	1,2,3
3	10	3	1,2,7
4	7	2	1,2
5	11	3	1,2,6
6	9	3	1,2,4

3.4 Evaluasi Pengujian

Evaluasi pengujian peringkas teks otomatis dilakukan dengan cara membandingkan hasil ringkasan manual dengan hasil ringkasan sistem. Dalam proses evaluasi ini juga dicari nilai kalimat relevan dari setiap ringkasan yang dihasilkan dari setiap dokumen dengan menggunakan persamaan (7). Uji evaluasi peringkas teks otomatis dapat dilihat pada Tabel 3.

Tabel 3: Uji Evaluasi Peringkas Teks Otomatis

Dokumen	Jumlah Kalimat	KRM	∑KRM	KRS	∑KRS	KR	∑KR
1	12	2,3,6,8	4	1,2,3	3	2,3	2
2	12	1,2,7	3	1,2,3	3	1,2	2
3	10	1,3,7	3	1,2,7	3	1,7	2
4	7	1,2	2	1,2	2	1,2	2
5	11	1,2,4,11	4	1,2,6	3	1,2	2
6	9	1,2,8	3	1,2,4	3	1,2	2

Keterangan :
KRM : Kalimat ringkasan manual yang dihasilkan oleh manusia
∑KRM : Jumlah kalimat ringkasan manual
KK : Kalimat ringkasan sistem
∑KRS : Jumlah kalimat ringkasan sistem
KR : Kalimat relevan
∑KR : Jumlah kalimat relevan

Setelah diketahui hasil dari pengujian evaluasi pada peringkas teks otomatis , selanjutnya adalah tahapan pengujian peringkas teks otomatis yang dilakukan terhadap ketepatan, keberhasilan dan gabungan keduanya, dengan menggunakan perhitungan *recall* menggunakan persamaan (5), *precision* menggunakan persamaan (6), dan *f-measure* menggunakan persamaan (8).

Hasil evaluasi pengujian peringkas teks otomatis dalam notasi perhitungan *recall*, *precision*, dan *fmeasure* dapat dilihat pada Tabel 4.

Tabel 4: Hasil Pengujian Metode Precison, Recall dan F-Measure

Dokumen	Precision	Recall	F-Measure
1	66,7%	50%	57,15%
2	66,7%	66,7%	67,7%
3	66,7%	66,7%	67,7%
4	100%	100%	100%
5	66,7%	50%	57,15%
6	66,7%	66,7%	67,7%
Rata - rata	72,25%	66,7%	69,6%

Dari tahapan pengujian yang telah dilakukan terhadap peringkas teks otomatis yang dibangun dengan menggunakan metode pengujian *Precision , Recall* dan *F-Measure*, dapat diketahui bahwa penggunaan metode cross method latent semantic analysis untuk menghasilkan ringkasan dari artikel politik berbahasa Indonesia memiliki rata – rata nilai performansi *Precision* sebesar 72,25%, *Recall* sebesar 66,7% dan *F-Measure* sebesar 69,6%. Nilai akurasi ini dianggap sudah dapat menghasilkan sebuah ringkasan dengan cukup baik.

Akurasi dari ringkasan yang dihasilkan dengan menggunakan *cross method latent semantic analysis* ini masih dapat di tingkatkan lagi dengan menggunakan daftar kata dasar dan *stop words* bahasa Indonesia yang lebih lengkap pada saat tahap *preprocessing.*

4. KESIMPULAN

Kesimpulan yang didapat dari hasil penelitian untuk implementasi *Cross Method Latent Semantic Analysis* untuk menghasilkan ringkasan pada peringkas teks otomatis pada artikel berita politik berbahasa Indonesia dengan *compression rate* sebesar 35% yang diuji dengan menggunakan data hasil ringkasan manual dan data hasil ringkasan sistem dengan menggunakan metode *Precision, Recall* dan *F-Measure* adalah :

1. Nilai akurasi rata-rata *precision* dari ringkasan yang dihasilkan sebesar 72,25%
2. Nilai rata-rata *recall* dari ringkasan yang dihasilkan sebesar 66,7%
3. Nilai rata-rata *f-measures* dari ringkasan yang dihasilkan sebesar 69,6%.

Dari nilai akurasi *Precision , Recall* dan *F-Measure* yang telah di dapatkan, dapat diketahui bahwa implementasi *cross method latent semantic analysis* pada peringkas teks otomatis untuk meringkas artikel berita politik berbahasa Indonesia dapat menghasilkan ringkasan dengan baik.

5.SARAN

Dari hasil penelitian pada ringkasan yang dihasilkan dengan *Cross Method Latent Semantic Analysis* yang di dapatkan dengan menggunakan metode pengujian intrinsik *F-Measure* diketahui bahwa tingkat akurasi dari ringkasan yang dihasilkan dengan menggunakan metode ini adalah sebesar 69,6%. Adapun saran yang dapat diberikan untuk meningkatkan tingkat akurasi dari ringkasan yang dihasilkan adalah sebagai berikut :

1. Penggunaan daftar *stop words* yang lebih lengkap untuk menghilangkan *noise* secara lebih sempurna yang

masih terdapat pada dokumen masukan.
2. Penggunaan daftar kata dasar bahasa Indonesia yang lebih lengkap untuk menyempurnakan hasil yang didapatkan pada tahapan *preprocessing stemming.*

DAFTAR PUSTAKA

[1] Y Gong and X Liu, "Generic Text Summarization Using Relevance Measure and Latent Semantic," *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Developement in Information Retrieval*, pp. 19 - 25, 2001.

[2] Karel Jezek Josef Steinberger, "Using Latent Semantic Analysis in Text Summarization and Summary Evaluation," *Proceedings of ISIM*, pp. 93-100, 2004.

[3] Steve Renals, Jean Carletta Gabriel Murray, "Extractive Summarization of Meeting Recordings," 2005.

[4] Makbule Gulcin Ozsoy, Ilyas Cicekli, and Ferda Nur Alp, "Text Summarization of Turkish Texts Using Latent Semantic Analysis," *Proceedings of the 23rd international conference on computational linguistics*, pp. 869 - 876, 2010.

[5] Junta Zeniarja, Abu Salam Ardytha Luthfiarta, "Algoritma Latent Semantic Analysis (LSA) Pada Peringkas Dokumen Otomatis Untuk Proses Clustering Dokumen," *Semantik*, vol. 3, no. 1, pp. 61-68, 2013.

[6] Adiwijawa, Moch Arif Bijaksana Agung Triwibowo, "Penggunaan Metode Relevance Measure Dan Latent Semantic Analysis (LSA) Dalam Membuat Ikhtisar Dokumen Berita," 2010.

[7] Andre F.T. Martins Dipanjan Das, "A Survey on Automatic Text Summarization," *Literature Survey for the Language and Statistics II course at CMU*, vol. 4, pp. 192 - 195, November 2007.

[8] J. Ilamathi, Nithya Vijayarani, "Preprocessing Techniques for Text Mining," *International Journal of Computer Science and Communication Network*, vol. 5, no. 1, pp. 7-16, 2015.

[9] Vikram Singh and Balwinder Saini, "An Effective Pre-Processing Algorithm For Information Retrieval Systems," *International Journal of Database Management Systems*, vol. 6, no. 6, p. 13, 2014.

[10] Fadillah Z Tala, "A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia," 2003.

[11] Gregoria Ariyanti, "Dekomposisi Nilai Singular dan Aplikasinya," *Prosiding Seminar Nasional Matematika dan Pendidikan Matematika (2010):" Peningkatan Kontribusi Penelitian dan Pembelajaran Matematika dalam Upaya Pembentukan Karakter Bangsa"*, 2010.

[12] Manabu Okumura Takahiro Fukusima, "Text Summarization Challenge Text Summarization Evaluation in Japan," *North American Association for Computational Linguistics (NAACL2001), Workshop on Automatic Summarization*, pp. 51 - 59, 2001.

[13] Karel Jezek Josef Steinberg, "Evaluation Measures for Text Summarization," *Computing and Informatics*, vol. 28, no. 2, pp. 251 - 275, 2009.