Peringkas Teks Otomatis Menggunakan Metode Latent Dirichlet Allocation (LDA)

Automatic Text Summarization using Latent Dirichlet Allocation (LDA)

Agit Fadillah Rihardi*[1], Surya Agustian[2], Eka Pandu Cynthia[3]
[1,2,3] Teknik Informatika, Fakultas Sains dan Teknologi, UIN Sultan Syarif Kasim, Riau
e-mail: ***[1]11751100389@students.uin-suska.ac.id**

Abstrak - *Saat ini pertumbuhan jumlah dokumen, artikel, tulisan berita, email dan bentuk teks lainnya yang tayang di internet, sangat cepat setiap harinya. Bagi pengguna yang membutuhkan informasi secara cepat dari berbagai dokumen tersebut, membaca keseluruhan isi teks dokumen akan sangat memboroskan waktu. Sistem peringkas teks otomatis membantu pengguna mendapatkan informasi secara cepat, tanpa mengabaikan inti dari informasi tersebut. Penilitian ini mengusulkan pemodelan topik sebagai metode untuk meringkas teks. Metode yang digunakan adalah Latent Dirichlet Allocation (LDA) yang membangkitkan topik dari keseluruhan isi teks. Ringkasan dibuat berdasarkan kalimat yang terpilih dari relevansi antara topik dengan kalimat-kalimat tersebut, yang diukur menggunakan jackard similarity. Performa sistem dievaluasi menggunakan ROUGE, dengan membandingkan ringkasan yang dihasilkan oleh sistem dengan ringkasan yang dibuat oleh manusia (gold standard). Dari optimasi yang dilakukan, pengujian untuk 150 artikel dengan kompresi 50%, memberikan hasil F1-score untuk ROUGE-1, ROUGE-2, dan ROUGE-L masing-masing sebesar 67.81%, 59.96%, dan 67.44%. Sedangkan untuk kompresi 30% mendapat F1-score untuk ROUGE-1, ROUGE-2, dan ROUGE-L masing-masing adalah 52.37%, 42.11%, dan 51.47%. Penilitian ini menghasilkan skor yang baik dan kompetitif dibandingkan dengan penilitian-penilitian lain yang terkait.*

Kata kunci – Jaccard Similarity, LDA, Pemodelan Topik, Peringkas Otomatis, ROUGE

Abstract *- Recently the number of documents, articles, news, e-mails and other forms of text that appear on the internet has been growing rapidly every day. For users who require prompt access to information from these diverse documents, reading the entire text of each document would be time-consuming and inefficient. Automatic text summarization systems help users find information quickly, without neglecting the essence of the information. This research proposes topic modeling as a method for summarizing documents. The method used is Latent Dirichlet Allocation (LDA) which generates the topic of the entire text. The summary is composed from the selected sentences, which are considered relevant by measuring the similarity between the topic words and the sentences in the text. System performance is measured with ROUGE scores, by comparing the summaries generated by system with the summaries composed by human (gold standard). The optimized system is tested on 150 articles at 50% compression rates, yielding F1-scores of 67.81%, 59.96%, and 67.44% for ROUGE-1, ROUGE-2, and ROUGE-L, respectively. Meanwhile, at 30% compression rates it yields F1-scores of 52.37%, 42.11%, and 51.47% for ROUGE-1, ROUGE-2, and ROUGE-L respectively. This research yields good and competitive scores compared to other related studies.*

Keywords – jaccard similarity, LDA, topic modelling, summarization, ROUGE.

I. PENDAHULUAN

Setiap harinya selalu ada informasi-informasi terbaru yang diterbitkan di internet, baik melalui portal media *mainstream*, media sosial, blog, website organisasi, forum dan sebagainya. Informasi yang berbentuk berita banyak yang isinya sama atau seragam, terlebih bila diterbitkan oleh media mainstream yang memiliki jaringan pemberitaan. Banyaknya berita yang terbit setiap harinya, menyebabkan kesulitan bagi pembaca untuk mendapatkan inti sari dari berita secara cepat. Metode peringkas teks otomatis memberikan solusi untuk menemukan informasi relevan dari dokumen berita, artikel, dan dokumen panjang lainnya secara efisien.

Peringkas teks otomatis menentukan kalimat paling informatif dari keseluruhan dokumen dan membuat sebuah ringkasan yang representatif [1]. Beradasarkan bagaimana hasil akhir sebuah ringkasan dibuat, metode peringkas terdiri dari *extractive summarization* dan *abstractive summarization* [2]. Metode *extractive summarization* membuat sebuah ringkasan membuat sebuah ringkasan dengan menggekstrak kalimat-kalimat signifikan dari sebuah atau beberapa dokumen tanpa mengubah isi dari kalimat-kalimat tersebut. Sedangkan metode *abstractive summarization* membuat sebuah ringkasan dengan me-*rephrasing* kalimat yang mempunyai informasi paling relevan atau membuat kalimat baru dengan konsep berdasarkan kalimat dari dokumen.

Peringkas dokumen dengan metode ekstraktif kebanyakan menggunakan teknik berbasis *unsupervised*. Pemodelan topik adalah salah satu pendekatan *unsupervised learning*. Pemodelan topik (*topic modelling*) merupakan sebuah algoritma yang bertujuan untuk membangkitkan topik dalam sekumpulan dokumen yang sangat besar dan tidak terstruktur tanpa memerlukan adanya anotasi atau pelabelan dalam dokumen tersebut [3]. Untuk mengekstraksi topik dari suatu dokumen, *Latent Dirichlet Allocation* (LDA) adalah salah satu metode yang cukup populer. LDA adalah metode pembelajaran tanpa pengawasan (*unsupervised*) dan model probabilitas generatif yang menggambarkan dokumen sebagai komposisi acak dari topik yang tersembunyi (laten), yang setiap topiknya dihitung dari distribusi kata [4].

Pada penilitian [5], peneliti mengajukan sebuah pendekatan ringkasan teks multi-dokumen berbasis *extractive topic modelling* untuk dokumen berita Malayalam. Lalu dengan mengadopsi *vector space model*, vektor topik dan vektor kalimat dari dokumen dihasilkan. Penilitian yang menggunakan konsep pemodelan topik lainnya [6], menggabungkan pemodelan topik dan semantic measure dalam *vector space model*.

Selain digunakan untuk meringkas dokumen, pemodelan topik dapat digunakan untuk klasifikasi dokumen. Penilitian [7] menggunakan metode LDA untuk mengklasifikasi dokumen laporan tugas akhir. Pada penilitian [8], LDA dapat diterapkan untuk memodelkan topik berita dan ringkasan dibuat melalui pengklasteran dengan metode k-means.

Pemodelan topik Latent Semantic Analysis (LSA) digunakan dalam [9] untuk tugas peringkas text otomatis. LSA digunakan bersamaan dengan kata kunci yang diekstrak dari nilai bobot TF.IDF setiap kalimat dalam dokumen. Similariy antara topik LSA dan kata kunci TF.IDF menjadi ukuran pemilihan kalimat yang akan diambil sebagai ringkasan.

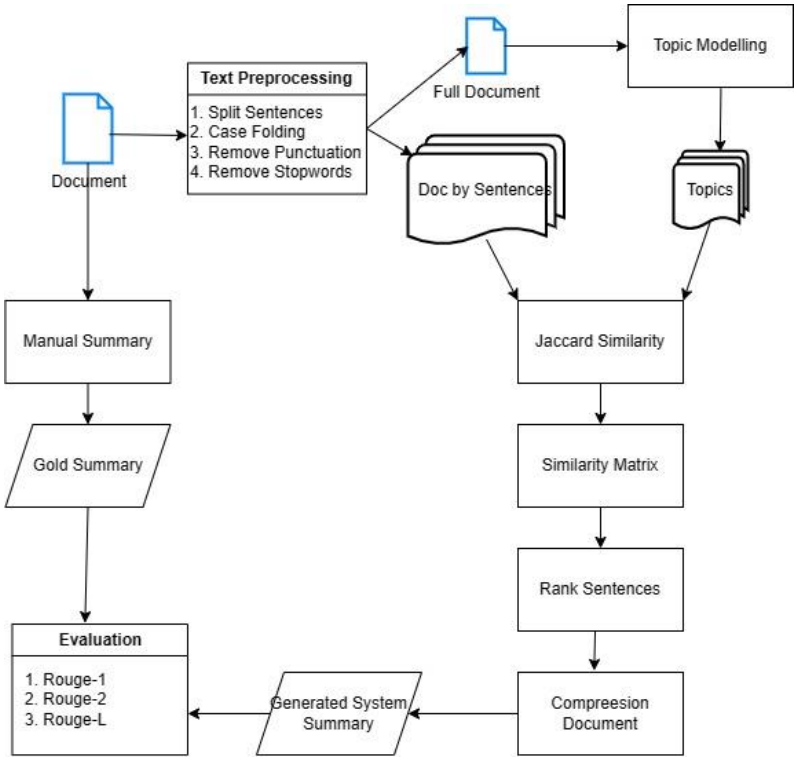
Penilitian ini mengusulkan penggunaan metode *Latent Dirichlet Allocation* (LDA) dalam melakukan pemodelan topik terhadap artikel-artikel berita. Setelah dilakukan pemrosesan teks dokumen, pemodelan topik dilakukan untuk mendapatkan topik-topik dari dokumen.

Untuk menentukan kalimat yang akan diekstrak menjadi ringkasan, dilakukan pengukuran relevansi antara topik yang dihasilkan LDA dan kalimat pada dokumen asli. Kualitas ringkasan yang dihasilkan akan dievaluasi menggunakan *ROUGE*, dengan membandingkan output ringkasan mesin dengan ringkasan yang dibuat oleh manusia (*gold standard*).

II. METODE

Penilitian ini dimulai dengan tahap pengumpulan dan analisa dataset. Dataset yang digunakan adalah artikel berita yang dikumpulkan pada penilitian sebelumnya [10][11]. Dataset terdiri dari 300 dokumen berita yang dikumpulkan dari berbagai portal berita *online*.

Tahapan penelitian selanjutnya untuk menghasilkan ringkasan dari sebuah dokumen, dapat dilihat pada Gambar 1 berikut.



Gambar 1. Tahapan Penilitian

2.1 *Text Preprocessing*

Text Preprocessing dilakukan untuk mengolah teks agar sesuai dengan kebutuhan untuk proses selanjutnya. Dokumen diolah dengan beberapa perlakuan seperti pembentukan token, menyaring kata-kata yang tidak signifikan, tanda baca yang tidak perlu, mengubah kata ke dalam bentuk akar kata, dan seterusnya. Proses ini akan menghasilkan teks yang dianggap bersih untuk menjalani proses selanjutnya, misalnya pembentukan *bag of words*, atau pelatihan untuk membentuk *word embeddings*, dan sebagainya. Hasil tersebut nantinya digunakan sebagai bahan analisis dan inputan tugas-tugas pemrosessan teks [12].

Tahapan-tahapan dalam *text preprocessing* dalam penelitian ini meliputi *sentences splitting* (memisahkan kalimat), *remove punctuation* (penghapusan tanda baca), *case folding* (mengubah teks menjadi huruf kecil), dan *stopwords removal* (penghapusan kata-kata yang tidak signifikan). Tabel 1 berikut ini menunjukkan contoh tahap *preprocessing* dokumen dari dataset dengan judul "Ancaman Pidana Laporan Perkosaan Palsu Gadis Muda Bukittinggi."

Tabel 1. Hasil Text Preprocessing

No.	*Text Preprocessing*	
	Kalimat Awal	**Hasil *Preprocessing***
1.	Perempuan muda berinisisal N, 23 tahun, warga Bukittinggi yang diduga membuat keterangan palsu terkait percobaan pemerkosaan oleh seorang driver ojek online terhadap dirinya bisa berpotensi dipidana.	perempuan muda berinisisal n 23 warga bukittinggi diduga keterangan palsu terkait percobaan pemerkosaan driver ojek online berpotensi dipidana
2.	"Bisa berpotensi (dipidana) jika itu memang keterangan palsu," ujar Kasat Reskrim Polresta Padang Kompol Rico Fernanda kepada Padangkita.com dihubungi melalui telepon selulernya, Selasa (28/9/2021).	berpotensi dipidana keterangan palsu kasat reskrim polresta padang kompol rico fernanda padangkitacom dihubungi telepon selulernya selasa 2892021
3.	Menurut Rico, karena melibatkan ojek online kasus ini akan berdampak besar.	rico melibatkan ojek online berdampak
4.	Para driver ojek online akan sangat marah karena namanya dibawa-bawa padahal bukan lah driver ojek online pelakunya.	driver ojek online marah namanya dibawabawa driver ojek online pelakunya
5.	"Sebab, perbuatan yang dilakukan oleh perempuan muda tersebut bisa menimbulkan ketakutan bagi para perempuan lainnya yang ingin naik ojek online untuk bepergian," imbuhnya.	perbuatan perempuan muda menimbulkan ketakutan perempuan ojek online bepergian imbuhnya
6.	Sebelumnya, N mengaku menjadi korban percobaan pemerkosaan dari seorang driver ojek online yang ia pesan untuk pergi ke Pasar Raya Padang.	n mengaku korban percobaan pemerkosaan driver ojek online pesan pergi pasar raya padang
7.	Kejadiannya pada Senin kemarin di kawasan Transmart Padang.	kejadiannya senin kemarin kawasan transmart padang
8.	Dari hasil pemeriksaan polisi, hal tersebut hanyalah rekayasa semata dari N agar tidak ketahuan selingkuh oleh pacarnya.	hasil pemeriksaan polisi tersebut rekayasa n ketahuan selingkuh pacarnya
9.	Ternyata, driver ojek online yang disebut N merupakan selingkuhannya.	driver ojek online n selingkuhannya', 'polisi mendalami
10.	Kini polisi masih mendalami kasus ini.	polisi mendalami
11.	Penyelidik telah memintai keterangan dari terduga pelaku dan N sendiri.	penyelidik memintai keterangan terduga pelaku n
12.	Saat ini, terduga pelaku masih di amankan polisi di Mapolresta Padang untuk penyelidikan lebih lanjut.	terduga pelaku amankan polisi mapolresta padang penyelidikan

2.2 Pemodelan Topik

Pemodelan topik untuk ringkasan dimaksudkan untuk mencari kalimat-kalimat yang paling informatif sesuai dengan topik dokumen. Metode yang digunakan dalam penilitian ini adalah *Latent Dirichlet Allocation* (LDA). LDA merupakan model probabilistik yang dihasilkan

dari sekumpulan data diskrit, contohnya pada korpus dokumen yang berbentuk teks [13]. Ide mendasar dari LDA adalah setiap dokumen diwakili sebagai komposisi atas topik-topik yang tersembunyi (laten), dimana setiap topik mempunyai ciri-ciri yang dihitung berdasarkan distribusi probabilitas kata-kata di dalamnya. Kata-kata yang memiliki probabilitas tertinggi di setiap topik dipilih sebagai representasi topik yang dihasilkan oleh algoritma.

Pada tahapan proses pemodelan topik, LDA akan menemukan topik laten yang tersembunyi dalam dokumen teks. Penemuan topik laten ini dilakukan dengan mengelompokkan kata-kata yang memiliki hubungan semantik berdasarkan perhitungan distribusi probabilitas. Dalam penelitian ini, membentuk model eksperimen dilakukan pada input parameter. Parameter yang dimasukkan adalah jumlah topik dan jumlah kata dalam suatu topik. Eksperimen terhadap dokumen di atas dengan parameter jumlah topik tiga dan jumlah kata sepuluh akan menghasilkan pemodelan topik sebagai berikut.

Tabel 2. Hasil Pemodelan Topik

No.	Pemodelan Topik	
	Topik	**Kata-kata**
1.	Topik 1	online, ojek, driver, perempuan, padang, n, muda, pemerkosaan, percobaan, terkait
2.	Topik 2	n, polisi, keterangan, padang, rico, terduga, pelaku, dipidana, berpotensi, palsu
3.	Topik 3	mendalami, polisi, pelaku, terduga, rico, melibatkan, berdampak, selingkuhannya, n, padang

2.3 Pengukuran Relevansi

Setelah melakukan pemodelan topik dan mendapatkan hasilnya, maka topik-topik yang dihasilkan akan diukur *similarity* (kemiripannya) dengan kalimat-kalimat dalam dokumen. Pengukuran ini dilakukan menggunakan metode *Jaccard Similarity*. *Jaccard similarity* menghitung nilai kemiripan antara suatu himpunan data dan himpunan data yang lain. Semakin tinggi nilai *similarity*-nya, maka semakin mirip dua himpunan yang dibandingkan [14]. *Jaccard similarity* menghitung nilai kemiripan antara suatu kalimat awal dokumen dan topik-topik yang dihasilkan LDA.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \quad \text{.....(1)}$$

Perhitungan menggunakan *jaccard similarity* dilakukan dengan formula pada persamaan (1). Dimana A adalah kata-kata pada himpunan A (topik) dan B adalah list kata-kata pada himpunan B (kalimat-kalimat dalam dokumen). Simbol ∩ adalah irisan (*intersection*), yaitu daftar kata-kata yang sama dari kedua himpunan. Sedangkan simbol ∪ adalah gabungan (*union*) kata-kata uniq dari kedua himpunan. Simbol nilai mutlak menandakan yang dihitung adalah jumlahnya.

Contoh perhitungan *jaccard similarity* dari kalimat 1 pada Tabel 1 dan topik 1 pada Tabel 2 yang dibangkitkan oleh metode LDA, dapat dihitung dengan cara sebagai berikut.

Kalimat 1:

perempuan muda berinisisal **n** 23 warga bukittinggi diduga keterangan palsu **terkait percobaan pemerkosaan driver ojek online** berpotensi dipidana

Topik 1:

online, **ojek**, **driver**, **perempuan**, padang, **n**, **muda**, **pemerkosaan**, **percobaan**, **terkait**

Kata yang tercetak tebal adalah kata yang beririsan, jumlahnya adalah 9 kata. Sedangkan jumlah kata uniq yang gabungan antara Kalimat 1 dan Topik 1 adalah 19. Maka hasil perhitungan kemiripan antara Kalimat 1 dan Topik 1 adalah:

Kata yang beririsan/jumlah seluruh kata = 9/19 = **0.47**

2.4 Ekstraksi Ringkasan

Proses ekstraksi ringkasan adalah proses untuk mengambil kalimat-kalimat yang mempunyai relevensi terhadap topik berdasarkan skor *similarity* yang tertinggi. Proses ini meliputi:

1) *Ranking Sentences*

Proses ini melakukan pengurutan nilai *similarity* antara kalimat topik hasil LDA dan kalimat dokumen. Skor *similarity* akan diurutkan dari skor yang tertinggi. Kalimat dengan skor *similarity* tinggi dapat diekstrak ke dalam ringkasan.

2) Kompresi Dokumen

Kompresi dokumen adalah proses untuk mengkompres atau mereduksi dokumen menjadi sekian persen. *Compression rate* yang dilakukan untuk membentuk ringkasan adalah 30% dan 50%. Proses kompresi ini akan membuang kalimat yang mempunyai skor *similarity* yang rendah.

3) Get Unique Senteces

Setelah dilakukan kompresi dokumen, maka masih ada peluang kalimat yang redundan. Hal ini diakibatkan karena kalimat yang redundan ini mendapatkan skor *similarity* yang tinggi. Maka dilakukan pembuangan kalimat yang redundan dan mengambil hanya satu kalimat agar kalimat yang muncul di ringkasan tidak berulang-ulang.

2.5 Evaluasi

Metode yang dikembangkan untuk ringkasan otomatis ini perlu dievaluasi, seberapa baik untuk menjawab permasalahan penelitian. Caranya adalah dengan mengukur kualitas hasil ringkasan yang dibangkitkan oleh sistem menggunakan *ROUGE scores*. *ROUGE* (*Recall-Oriented Understudy for Gisting Evaluation*) mengukur kualitas sebuah ringkasan dengan membandingkannya dengan ringkasan manual yang dibuat oleh manusia (*gold standard*) [15]. *ROUGE scores* menampilkan komponen nilai *precision*, *recall*, dan *F1-score* untuk *ROUGE-N*, dengan *N* biasanya adalah {1, 2 dan L}. Indeks variabel *N* memiliki arti jumlah *N* kata berurutan yang sama dari kedua set yang dibandingkan (ringkasan sistem dan ringkasan manual). *N*=1 mengukur *word-1-gram* (*unigram*) dan *N*=2 mengukur *word-2-gram* (*bigram*) yang terdalat di dalam kedua dokumen. Sedangkan *L* adalah *Longest Common Subsequent* (LCS), yaitu kata-kata berurutan yang terpanjang yang terdapat di kedua set yang dibandingkan. Persamaan untuk menghitung *ROUGE scores* adalah sebagai berikut.

$$ROUGE\text{-}1\ recall = \frac{jumlah\ unigram\ kata\ yang\ sama}{total\ kata\ di\ ringkasan\ manual} \quad \text{.....(2)}$$

$$ROUGE\text{-}1\ precission = \frac{jumlah\ unigram\ kata\ yang\ sama}{total\ kata\ di\ ringkasan\ sistem} \quad \text{.....(3)}$$

$$ROUGE\text{-}2\ recall = \frac{jumlah\ bigram\ kata\ yang\ sama}{total\ kata\ di\ ringkasan\ manual} \quad \text{.....(4)}$$

$$ROUGE\text{-}2\ precission = \frac{jumlah\ bigram\ kata\ yang\ sama}{total\ kata\ di\ ringkasan\ sistem} \quad \text{.....(5)}$$

$$ROUGE\text{-}L\ recall = \frac{LCS(longest\ common\ subsequent)}{total\ kata\ di\ ringkasan\ manual} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(6)$$

$$ROUGE\text{-}L\ precission = \frac{LCS(longest\ common\ subsequent)}{total\ kata\ di\ ringkasan\ sistem} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(7)$$

$$f1\text{-}scores = 2 * \frac{precission*recall}{precission+recall} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(8)$$

III. HASIL DAN ANALISA

3.1 Pemilihan Parameter Pemodelan Topik

Dari 300 artikel pada dataset, dibagi menjadi 150 dokumen sebagai data untuk pengembangan sistem (data "*training*") dan 150 dokumen lainnya sebagai data pengujian (data *testing*). Istilah data training meminjam terminologi pada penelitian pembelajaran mesin, karena secara kosep metode penelitian ini tidak menerapkan pembelajaran mesin. Karena pelaksanaan eksperimen menyerupai pendekatan yang banyak dipakai pada penelitian *machine learning*, maka istilah data "*training*" dipakai. Data *training* digunakan untuk melakukan optimasi metode ringkasan yang diusulkan, sehingga cara pemilihan topik yang menghasilkan score evaluasi terbaik, akan dipilih sebagai model sistem peringkas teks otomatis. Model optimal dipilih berdasarkan nilai *F1-score* tertinggi untuk *ROUGE-L* dari hasil ringkasan untuk data *training*. Kemudian, model ini diterapkan untuk menghasilkan ringkasan otomatis pada data *testing*.

Parameter pemodelan topik yang digunakan untuk mencari model terbaik pada penelitian ini antara lain; jumlah set topik (*topic-set*) dari dokumen (**dua**, **tiga**, dan **lima** topik), dengan jumlah kata (*word-list*) pada masing-masing topik adalah **lima** dan **sepuluh** kata.

3.2 Pemodelan Topik untuk Ringkasan dengan tingkat kompresi 50%

Hasil eksperimen untuk pencarian model optimal peringkas teks otomatis dengan tingkat kompresi (CR, *compression rate*) 50%, ditunjukkan pada Tabel 3. Seluruh skor yang dihitung adalah rata-rata dari hasil pengukuran *ROUGE score* pada 150 artikel data *training*. Parameter optimal yang dipilih adalah yang menghasilkan dengan nilai *F1-score* pada *ROUGE-L* yang tertinggi, yaitu topic set=3 dan word list=10.

Tabel 3. Eksperimen untuk *Compression Rate* 50%

Parameter LDA		Rouge-1			Rouge-2			Rouge-L		
Topic set	**Word list**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
2	5	63.29	73.19	67.26	55.91	63.79	59.04	62.98	72.81	66.92
2	10	63.78	73.33	67.75	56.48	64.03	59.58	63.33	72.83	67.28
3	5	62.86	72.24	66.73	54.81	62.56	57.96	62.35	71.63	66.18
3	**10**	**64.08**	**72.91**	**67.76**	**56.16**	**63.37**	**59.12**	**63.66**	**72.41**	**67.30**
5	5	62.10	70.13	65.24	52.80	60.05	55.61	61.46	69.40	64.57
5	10	63.30	70.65	66.35	54.59	61.08	57.26	62.66	69.91	65.66

3.3 Evaluasi Ringkasan pada Data *Testing* dengan tingkat kompresi 50%

Evaluasi pada data uji *(testing)* dari parameter yang diperoleh sebagaimana pada Tabel 3, diterapkan untuk meringkas artikel-artikel pada data *testing*. Skor rata-rata yang didapat untuk masing-masing *ROUGE*-1, *ROUGE*-2, dan *ROUGE*-L adalah 67.81%, 59.96%, dan 67.44%. Tabel 4 memperlihatkan cuplikan sejumlah artikel dan hasil perhitungan *ROUGE score* dari ringkasan yang dihasilkan oleh sistem.

Tabel 4. *ROUGE score* untuk Ringkasan Data *Testing* dengan *Compression Rate* 50%

No.	Artikel ID	Rouge-1(%)			Rouge-2(%)			Rouge-L(%)		
		R*	P*	F1*	R	P	F1	R	P	F1
1	Doc-151	66.56	83	73.87	59.35	76.24	66.75	66.56	83	73.87
2	Doc-152	75.22	88.54	81.34	70.47	84.68	76.92	75.22	88.54	81.34
…	…	…	…	…	…	…	…	…	…	…
149	Doc-299	55.06	55.06	55.06	39.84	44.55	42.06	53.93	53.93	53.93
150	Doc-300	76.19	75	75.59	68.06	69.01	68.53	76.19	75	75.59
	Rata-rata	**64.83**	**72.04**	**67.81**	**57.34**	**63.81**	**59.96**	**64.47**	**71.64**	**67.44**

**R=recall, P=precision, F1=F1-score*

3.4 Pemodelan Topik untuk Ringkasan dengan tingkat kompresi 30%

Eksperimen untuk optimasi model menggunakan data *training* untuk *CR*=30% memberikan hasil *ROUGE-score* rata-rata sebagaimana Tabel 5 di bawah ini. Parameter LDA optimal yang dipilih adalah *topic-set*=2 dan *word-list*=10.

Tabel 5. Eksperimen untuk *Compression Rate* 50%

Parameter LDA		Rouge-1			Rouge-2			Rouge-L		
Topic set	Word list	R	P	F1	R	P	F1	R	P	F1
2	5	47.96	58.61	52.00	37.59	45.46	40.53	47.03	57.38	50.95
2	10	**48.19**	**59.57**	**52.57**	**38.20**	**46.81**	**41.49**	**47.36**	**58.54**	**51.66**
3	5	47.56	56.50	50.96	36.80	43.76	39.42	46.61	55.30	49.91
3	10	47.99	57.06	51.41	37.52	44.47	40.10	46.96	55.80	50.29
5	5	47.02	54.05	49.64	35.90	41.85	38.11	45.89	52.70	48.44
5	10	47.50	54.83	50.23	36.24	42.51	38.64	46.43	53.54	49.08

3.5 Evaluasi Ringkasan pada Data *Testing* dengan tingkat kompresi 30%

Hasil evaluasi pada data *testing* dengan parameter LDA optimal sebagaimana pada Tabel 5, diterapkan untuk membangkitkan ringkasan secara otomatis dari data testing, dengan *CR*=30%. Nilai rata-rata *F1-score* untuk *ROUGE*-1, *ROUGE*-2, dan *ROUGE*-L masing-masing sebesar 52.37%, 42.11%, dan 51.47%.

Tabel 6. *ROUGE score* untuk Ringkasan Data *Testing* dengan *Compression Rate* 50%

No.	Artikel ID	Rouge-1(%)			Rouge-2(%)			Rouge-L(%)		
		R	P	F	R	P	F	R	P	F
1	Doc-151	58.62	82.93	68.69	52.23	77.19	62.3	58.62	82.93	68.69
2	Doc-152	72.97	67.5	70.13	64.77	58.16	61.29	71.62	66.25	68.83
…	…	…	…	…	…	…	…	…	…	…
149	Doc-299	61.11	73.33	66.67	52.81	60.26	56.29	59.72	71.67	65.15
150	Doc-300	60	52.5	56	47.06	38.1	42.11	57.14	50	53.33
	Rata-rata	**48.62**	**58.62**	**52.37**	**39.26**	**47.07**	**42.11**	**47.80**	**57.58**	**51.47**

3.6 Analisa Hasil dan Perbandingan dengan Penelitian terkait

Dari hasil yang diperoleh, metode yang diusulkan cukup kompetitif dan secara umum memiliki performa yang baik, bila dibandingkan dengan hasil-hasil yang diperoleh pada penelitian peringkas teks otomatis. Diperlukan sebuah *benchmark* dataset agar metode ini

dapat dibandingkan secara *apple-to-apple.*

Tabel 7 adalah *ROUGE score* dari skema pengujian yang sama antara penelitian [10, 11] dengan metode yang diusulkan dalam penelitian ini. Terlihat bahwa untuk dataset yang sama, nilai *F1-score* penelitian ini dapat melampaui hasil algoritma TextRank [10], dan mendekati ringkasan dari metode LexRank [11] untuk CR=30% untuk ketiga *ROUGE score*. Sedangkan untuk CR=50%, *F1-score* ringkasan yang diperoleh dari metode LDA melampaui metode *LexRank* [11], namun agak sedikit di bawah *TextRank* [10]. Ketiga metode tidak terpaut jauh, hanya selisih dalam rentang nilai 1% untuk *F1-score*. Namun untuk nilai *Precision*, metode LDA mengungguli *TextRank* dan *LexRank* untuk ketiga *ROUGE score*.

Dari kedua skema pengujian terhadap data testing (CR 30% dan 50%), metode LDA secara konsisten berada di pertengahan bila performa ketiga metode dalam Tabel 7 di-ranking. Sedangkan metode lainnya tidak konsisten, untuk CR 30% metode *LexRank* terbaik, sedangkan untuk CR 50% metode *TextRank* yang lebih baik dari segi *F1-score*-nya.

Tabel 7. Perbandingan beberapa metode penelitian

No.	CR	Metode	Rouge-1(%)			Rouge-2(%)			Rouge-L(%)		
			R	P	F	R	P	F	R	P	F
1	30%	TextRank [10]	43,28	48,02	45,00	30,24	34,05	31,62	41,91	46,52	43,59
2		LexRank [11]	**53,35**	**60,12**	**55,82**	**44,33**	**48,09**	**45,51**	**52,35**	**58,96**	**54,76**
3		LDA	48.62	58.62	52.37	39.26	47.07	42.11	47.80	57.58	51.47
4	50%	TextRank [10]	**72,32**	66,15	**68,76**	**65,08**	57,29	**60,60**	**71,81**	65,72	**68,29**
5		LexRank [11]	67,03	69,09	67,53	59,74	59,58	59,14	66,53	68,62	67,05
6		LDA	64.83	**72.04**	67.81	57.34	**63.81**	59.96	64.47	**71.64**	67.44

IV. KESIMPULAN

Metode *Latent Dirichlet Allocation,* dapat diterapkan untuk tugas peringkas teks otomatis dengan performa yang kompetitif. Dari proses pencarian model optimal menggunakan data *training* sebanyak 150 artikel berita, menunjukkan bahwa parameter pemodelan LDA terbaik untuk ringkasan dengan tingkat kompresi 50%, adalah 3 *topic-set* dan 10 *word-list* untuk masing-masing topik. Untuk ringkasan dengan tingkat kompresi 30%, parameter optimalnya adalah 2 *topic-set* dan 10 *word-list*. Penerapan model optimal LDA pada peringkasan data *testing*, menghasilkan *ROUGE score* yang kompetitif dibandingkan metode TextRank dan LexRank. Untuk ringkasan dengan tingkat kompresi dokumen 50%, rata-rata *F1-score* adalah 67.81%, 59.96%, 67.44% pada *ROUGE*-1 *ROUGE*-2, dan *ROUGE*-L. Sedangkan untuk ringkasan dengan tingkat kompresi dokumen 30%, rata-rata *F1-score* yang dicapai 52.37% pada *ROUGE*-1, 42.11% pada *ROUGE*-2, dan 51.47% pada *ROUGE*-L. Tingkat kompresi dokumen juga berpengaruh pada terhadap hasil kualitas ringkasan pada metode yang digunakan. Semakin besar tingkat kompresi dokumen, maka hasil kualitas ringkasan yang dicapai semakin tinggi, karena peluang kalimat yang terpilih sebagai ringkasan semakin besar.

DAFTAR PUSTAKA

[1] R. Deepa, J. Konshi, A. Haritha, and K. Shobini, "Automatic Text Summarization System," 2019. [Online]. Available: http://www.ripublication.com

[2] A. P. Widyassari *et al.*, "Review of automatic text summarization techniques & methods," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 4. King Saud bin Abdulaziz University, pp. 1029–1046, Apr. 01, 2022. doi: 10.1016/j.jksuci.2020.05.006.

[3] Y. S. Maulidia and N. F. Siti, "Peringkasan Teks Otomatis pada Modul Pembelajaran Berbahasa Indonesia Menggunakan Metode Cross Latent Semantic Analysis (CLSA)," *Jurnal Edukasi dan Penelitian Informatika*, vol. 7, no. 2, pp. 153–159, 2021

[4] D. M. Blei, "Probabilistic topic models," in *Communications of the ACM*, Apr. 2012, pp. 77–84. doi: 10.1145/2133806.2133826.

[5] M. Kondath, D. P. Suseelan, and S. M. Idicula, "Extractive summarization of Malayalam documents using latent Dirichlet allocation: An experience," *Journal of Intelligent Systems*, vol. 31, no. 1, pp. 393–406, Jan. 2022, doi: 10.1515/jisys-2022-0027.

[6] R. C. Belwal, S. Rai, and A. Gupta, "Text summarization using topic-based vector space model and semantic measure," *Inf Process Manag*, vol. 58, no. 3, May 2021, doi: 10.1016/j.ipm.2021.102536.

[7] U. T. Setijohatmo, S. Rachmat, T. Susilawati, and Y. Rahman, "Analisis Metoda Latent Dirichlet Allocation untuk Klasifikasi Dokumen Laporan Tugas Akhir Berdasarkan Pemodelan Topik," 2020.

[8] R. Siringoringo, R. Perangin-Angin, and Jamaluddin, "Pemodelan Topik Berita Menggunakan Latent Dirichlet Allocation dan K-Means Clustering," *Jurnal Informatika Kaputama (JIK)*, vol. 4, no. 2, pp. 216-222, 2020.

[9] H. Gupta and M. Patel, "Method of Text Summarization Using LSA and Sentence Based Topic Modelling with Bert," in *Proceedings - International Conference on Artificial Intelligence and Smart Systems, ICAIS 2021*, Institute of Electrical and Electronics Engineers Inc., Mar. 2021, pp. 511–517. doi: 10.1109/ICAIS50930.2021.9395976.

[10] F. Husniah, S. Agustian, and I. Afrianty, "Peringkasan Teks Otomatis Artikel Berbahasa Indonesia Menggunakan Algoritma Textrank," *Teknoka* 7, 2022.

[11] Halimah, Surya Agustian, and Siti Ramadhani, "Peringkasan teks otomatis (automated text summarization) pada artikel berbahasa indonesia menggunakan algoritma lexrank," *Jurnal CoSciTech (Computer Science and Information Technology)*, vol. 3, no. 3, pp. 371–381, Dec. 2022, doi: 10.37859/coscitech.v3i3.4300.

[12] M. Anandarajan, C. Hill, and T. Nolan, *Practical Text Analytics: Maximizing the Value of Text Data*. in Advances in Analytics and Data Science. Springer International Publishing, 2018. [Online]. Available: https://books.google.co.id/books?id=cwZ0DwAAQBAJ

[13] D. M. Blei, A. Y. Ng, and J. B. Edu, "Latent Dirichlet Allocation Michael I. Jordan," 2003.

[14] S. Kadagadkai, M. Patil, A. Nagathan, A. Harish, and A. MV, "Summarization tool for multimedia data," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 2–7, Jun. 2022, doi: 10.1016/j.gltp.2022.04.001.

[15] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of summaries," in *Proceedings of the ACL Workshop: Text Summarization Braches Out 2004*, Jun. 2004, p. 10.