# Peringkasan Teks Berbahasa Indonesia dengan *Latent Dirichlet Allocation* dan *Maximum Marginal Relevance*

*Summarizing Indonesian Texts using Latent Dirichlet Allocation and Maximum Marginal Relevance*

Bima Hamdani Mawaridi[1], Muhammad Faisal[2], Hani Nurhayati[3]
[1,2,3]Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang
E-mail: [1]03bima.mawaridi@gmail.com, [2]mfaisal@ti.uin-malang.ac.id, [3]hani@ti.uin-malang.ac.id

**Abstrak**

Kemajuan teknologi membuat berita mudah ditemukan pada media online. Jumlah artikel berita yang tersedia semakin banyak dengan teks yang cukup panjang. Hal ini akan menyulitkan pembaca berita dalam mencari inti informasi dari berita sehingga diperlukan ringkasan teks untuk membantu pengguna memahami inti dari suatu teks tanpa perlu membaca seluruhnya. Metode yang digunakan untuk peringkasan teks yaitu *Maximum Marginal Relevance* (MMR) dengan menggabungkan dua faktor pemilihan, yaitu relevansi dan keragaman. Sering ditemukan saat ini bahwa judul berita dalam artikel online belum sepenuhnya mewakili isi berita atau disebut *clickbait*, untuk menghindari judul yang kurang sesuai, pada penelitian ini peringkasan didasarkan pada kata kunci yang dihasilkan dengan metode *Latent Dirichlet Allocation* (LDA). Hasil uji coba dengan 2500 data artikel berita menghasilkan nilai rata-rata ROUGE-1 terbaik sebesar 0.488 untuk tingkat kompresi 50% dan 0.462 untuk tingkat kompresi 30%. Nilai ROUGE-1 terendah yaitu 0.453 untuk tingkat kompresi 50% dan 0.435 untuk tingkat kompresi 30%. Hasil tersebut menunjukkan bahwa sistem dapat menghasilkan ringkasan yang cukup relevan dengan menggunakan kata kunci yang diekstrak dari konten berita.

Kata kunci: LDA, MMR, Pemodelan Topik, Peringkasan Teks

***Abstract***

*Advances in technology make news easy to find on online media. The number of news articles available is increasing with a fairly long text. This will make it difficult for news readers to find the core information from the news so that a text summary is needed to help users understand the essence of a text without the need to read it all. The method used for text summarization is Maximum Marginal Relevance (MMR) by combining two selection factors, namely relevance and diversity. It is often found today that news titles in online articles do not fully represent the content of the news or called clickbait, to avoid inappropriate titles, in this study the summary is based on keywords generated by the Latent Dirichlet Allocation (LDA) method. The test results with 2500 news article data produced the best average ROUGE-1 value of 0.488 for a compression level of 50% and 0.462 for a compression level of 30%. The lowest ROUGE-1 value is 0.453 for a compression level of 50% and 0.435 for a compression level of 30%. These results show that the system can produce quite relevant summaries using keywords extracted from news content.*

*Keywords: LDA, MMR, Topic Modeling, Text Summarization*

## 1. PENDAHULUAN

Berita merupakan kumpulan informasi yang telah terjadi atau sedang terjadi dan disebarluaskan melalui berbagai sarana penyiaran seperti internet, media cetak, siaran radio, dan dari mulut ke mulut [1]. Informasi yang ada pada sebuah artikel berita cukup beragam dan padat. Hal tersebut membuat pembaca seringkali kesulitan memahami poin utama dari teks karena banyaknya informasi yang ada pada artikel berita. Ringkasan dapat menyajikan informasi penting dari sumber dengan cara yang lebih singkat dan mudah dimengerti, sehingga pembaca dapat memahami intisari dari artikel tersebut tanpa harus membaca keseluruhan teks aslinya [2]. Artikel berita yang panjang jika dibaca secara keseluruhan maka akan memerlukan waktu yang lama, adanya sistem yang dapat meringkas secara otomatis diharapkan dapat membantu untuk mempersingkat waktu dan juga mendapatkan informasi yang tepat.

Sistem peringkasan jika dilihat dari inputnya dapat menggunakan *single* dokumen atau *multi* dokumen, jika dilihat dari keluarannya dapat dibedakan menjadi bentuk ekstraktif dan abstraktif. Sistem peringkasan otomatis dapat menggunakan berbagai pendekatan, salah satunya pendekatan berbasis kueri dimana hasil ringkasan didasarkan pada kueri atau topik tertentu [3]. Penelitian telah dilakukan menggunakan *Maximum Marginal Relevance* (MMR) untuk menghasilkan ringkasan dari artikel berita [4]. Penelitian tersebut menguji 30 artikel berita dengan topik COVID-19 menggunakan judul artikel sebagai *query*. Terdapat juga penelitian yang membandingkan MMR dengan *Textrank* untuk meringkas dua dokumen protokol etika kesehatan, hasilnya ringkasan yang menggunakan MMR mempunyai tingkat relevansi lebih tinggi daripada *Textrank* [5]. Metode MMR dapat mencegah redudansi dan dapat mengambil informasi yang relevan. Metode ini mampu memilih dokumen yang paling relevan dengan sebuah kueri tertentu dengan menggabungkan dua faktor yaitu relevansi dan keragaman [6].

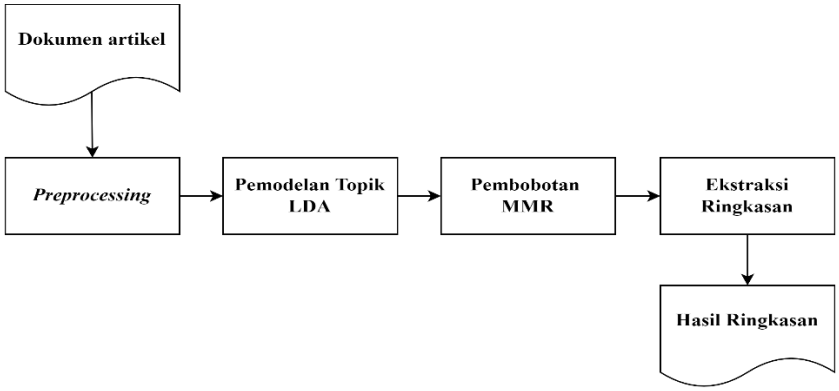
Pada umumnya peringkasan dokumen dengan metode MMR menggunakan judul sebagai kuerinya [7]. Namun yang sering terjadi saat ini, judul pada artikel berita belum sepenuhnya mewakili isi berita atau disebut *clickbait* sehingga pembaca perlu membaca keseluruhan berita untuk mengetahui konteks berita tersebut [8]. Oleh karena itu pada penelitian ini menghindari penggunaan judul sebagai kueri pada MMR. Kami mengusulkan penggunaan kata kunci yang diekstrak dari konten artikel berita menggunakan pemodelan topik. Suatu artikel akan diekstrak topik beserta kata kunci yang berkaitan dengan topik tersebut. Terdapat beberapa pendekatan dalam melakukan pemodelan topik, salah satunya menggunakan *Latent Dirichlet Allocation* (LDA). LDA pernah digunakan untuk pemodelan topik kemudian dilanjutkan untuk peringkasan teks menggunakan *Texrank* [9,10].

Penelitian ini berusaha mengkombinasikan metode LDA dengan metode MMR. Kombinasi tersebut diharapkan dapat memudahkan pembaca memahami artikel berita dengan cepat walaupun judulnya kurang sesuai dengan konten berita, sehingga pembaca dapat melakukan pekerjaan secara efektif dan tidak terkecoh dengan judul yang ada. Pada penelitian ini sistem peringkasan menggunakan input dokumen tunggal dan jenis peringkasan ekstraktif. Metode LDA akan digunakan untuk pemodelan topik pada artikel berita kemudian distribusi kata dari topik yang dihasilkan akan digunakan sebagai parameter *query* pada metode MMR.

## 2. METODE PENELITIAN

Tahapan pertama pada penelitian ini yaitu melakukan persiapan data. *Dataset* yang digunakan yaitu *Indonesian Text Summarization Datasets (IndoSum)*. *Preprocessing* data merupakan tahapan selanjutnya setelah persiapan data yang meliputi proses *segmentation, case folding, cleaning, stopword removal*, dan *stemming*. Dilakukannya *preprocessing* memungkinkan pemrosesan data pada tahap berikutnya akan berjalan dengan lebih efektif dan efisien. Tahap setelah *preprocessing* yaitu pemodelan topik menggunakan *Latent Dirichlet Allocation* yang dimana menghasilkan kata kunci dari topik yang diekstrak. Setelah itu dilakukan pembobotan pada setiap kalimat menggunakan *Maximum Marginal Relevance* hingga akhirnya dihasilkan sebuah ringkasan. Hasil ringkasan kemudian akan dievaluasi menggunakan metrik ROUGE-1

dengan membandingkan antara hasil ringkasan sistem dengan ringkasan manual yang dibuat oleh manusia.




Gambar 1 Alur Penelitian

*2.1 Data Penelitian*

Penelitian ini menggunakan *dataset* "*Indonesian Text Summarization (IndoSum)*". *IndoSum* merupakan kumpulan data besar untuk peringkasan teks dalam bahasa Indonesia yang dikumpulkan dari artikel berita online dan tersedia untuk umum [11]. *Dataset* ini merupakan kumpulan data yang disediakan oleh "Shortir", sebuah perusahaan agregator berita dan ringkasan bahasa Indonesia. Kumpulan data ini berisi sekitar 20 ribu artikel berita. *Dataset IndoSum* dipilih karena *dataset* tersebut mempunyai kumpulan artikel bahasa Indonesia dengan jumlah besar dan berbagai topik untuk memastikan bahwa sistem dapat menangani berbagai jenis teks berita dan topik. *Dataset IndoSum* mempunyai data ringkasan manual secara ekstraktif dan terdapat label ekstraktif untuk setiap kalimat pada setiap artikel berita sehingga sesuai dengan penelitian ini yang berfokus pada peringkasan teks secara ekstraktif. Setiap artikel mempunyai judul, label ekstraktif, kategori, sumber (misalnya CNN Indonesia, Kumparan), URL artikel asli, dan ringkasan yang dibuat secara manual oleh total 2 penutur asli bahasa Indonesia (*native speaker*). Pada penelitian ini digunakan 2500 data artikel yang diujicobakan pada sistem peringkasan teks.

*2.2 Preprocessing*

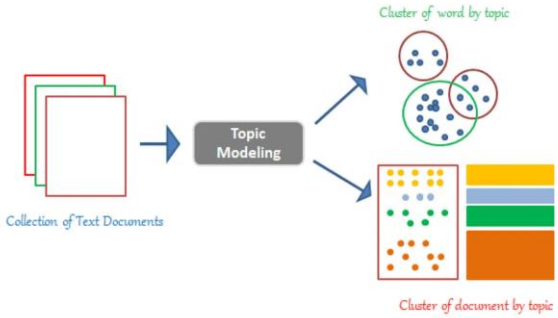
Tahap *Preprocessing* perlu diterapkan pada data yang diperoleh untuk menyempurnakan struktur data inputan. Tokenisasi sering kali menjadi bagian dari langkah *preprocessing* pada NLP, karena pada penelitian ini data yang digunakan adalah *dataset IndoSum*, tokenisasi sudah dilakukan pada *dataset* tersebut. Oleh karena itu proses tokenisasi tidak perlu dilakukan pada penelitian ini. Pada *dataset IndoSum*, setiap artikel dipisahkan berdasarkan paragraf, kemudian setiap paragraf berisi *list* kalimat, dan setiap kalimat berisi *list* kata atau *token*.

Tahap pertama *preprocessing* yaitu *segmentation* ini untuk memudahkan proses selanjutnya dengan menghiraukan pemisah paragraf sehingga setiap artikel langsung dipisah berdasarkan kalimat yang ada pada artikel tersebut. Tahapan kedua *preprocessing* yaitu *case folding*, pada tahap ini seluruh kata yang menggunakan huruf kapital akan diubah menjadi huruf kecil semua. Tahap ketiga yaitu *cleaning* data, pada tahap ini dilakukan penghapusan inputan tertentu, yaitu karakter selain huruf seperti tanda baca dan angka. Tahap keempat yaitu *stopword removal*, pada tahapan ini dilakukan proses mengidentifikasi serta menghapus kata-kata yang umum dan sering muncul dalam sebuah teks tetapi sering kali tidak memberikan informasi penting, contohnya seperti konjungsi. Tahap terakhir yaitu *stemming*, pada tahap ini dilakukan pengubahan kata-kata menjadi kata dasarnya.

*2.3 Pemodelan Topik LDA*

Proses ini mengidentifikasi topik yang terkandung di dalam setiap artikel dan kata kunci yang berkaitan dengan topik tersebut menggunakan metode LDA. *Latent Dirichlet Allocation*

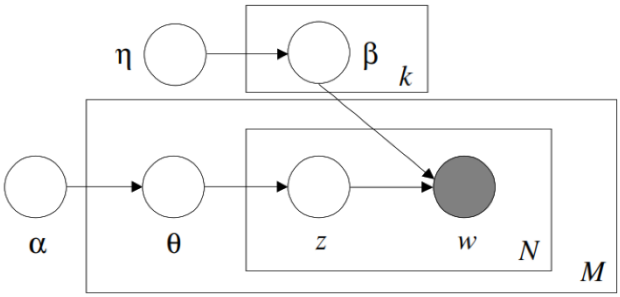
(LDA) pertama kali diperkenalkan oleh Blei dan Jordan, merupakan model probabilistik generatif untuk menemukan struktur semantik *corpus* menggunakan *hierarchical bayesian model* [12].




Gambar 2 Alur Kerja LDA

Cara kerja model LDA yang pertama yaitu menginisialisasi beberapa parameter, termasuk jumlah dokumen, topik, dan iterasi. Jumlah topik merupakan parameter yang paling penting dalam penggunaan metode LDA. Kedua, menetapkan kata untuk topik tertentu secara acak sesuai dengan distribusi *dirichlet*. Proses penentuan topik pada LDA melibatkan pencarian dua bentuk distribusi probabilitas yaitu distribusi probabilitas topik dalam sebuah dokumen dan distribusi probabilitas kata dalam sebuah topik. Langkah ketiga melakukan pengulangan setiap alur proses untuk setiap kata dalam korpus yang digunakan [13].

Pada LDA dasar, kata-kata yang tidak muncul dalam dokumen akan memiliki probabilitas nol sehingga dapat menimbulkan masalah pada model probabilistik. Menghindari distribusi posterior yang tidak valid dan terpusat pada nilai tertentu dari parameter maka digunakan *smooth* LDA. *Smooth* LDA memiliki *hyperparameter* $\eta$ pada *dirichlet* yang menyatakan *smoothing* kata dalam topik, serta *hyperparameter* $\alpha$ menyatakan *smoothing* topik dalam dokumen. Gambar 3 merupakan bentuk dari arsitektur model *smooth* LDA dalam sebuah diagram.




Gambar 3 Arsitektur model *smooth* LDA

**Keterangan:**

| | | |
|---|---|---|
| $\alpha$ | : | Parameter distribusi topik per dokumen |
| $\eta$ | : | Parameter distribusi kata per topik |
| $\theta$ | : | Distribusi probabilitas topik pada dokumen ke-d dengan parameter α |
| $\beta$ | : | Distribusi probabilitas kata-kata pada topik ke-k dengan parameter $\eta$ |
| $Z$ | : | Topik yang dihasilkan oleh distribusi multinomial dengan parameter $\theta$ |
| $W$ | : | Sampel kata dari distribusi multinomial dengan parameter $\beta$ dan $Z$ |

Garis batas pada gambar adalah "*plates*" yang melambangkan replika. *Plates* luar melambangkan dokumen, sedangkan *plates* dalam melambangkan pilihan berulang topik dan kata-kata dalam dokumen. *Edge* adalah tautan yang menghubungkan *node*, sedangkan *node* adalah variabel acak. Variabel yang diarsir menunjukkan variabel yang teramati dan yang tidak diarsir merupakan variabel tersembunyi (*laten)*. Variabel *M* merupakan jumlah seluruh dokumen, *N* jumlah seluruh kata pada dokumen tertentu, dan *k* ialah jumlah topik yang ingin diekstrak. Parameter *dirichlet* diatur menjadi simetris untuk perataan kata dalam

topik dengan $\eta = \frac{1}{V}$ dan topik dalam dokumen dengan $\alpha = \frac{1}{K}$. *K* adalah jumlah seluruh topik, *V* adalah jumlah kosa kata dalam *corpus* [14].

*2.4 Pembobotan TF-IDF*

Pembobotan *Term Frequency–Inverse Document Frequency* (TF-IDF) adalah metrik statistik yang digunakan untuk mengukur seberapa penting suatu istilah dalam suatu dokumen relatif terhadap seluruh kumpulan dokumen (korpus). Metode ini menggabungkan dua konsep utama yaitu frekuensi kemunculan kata dalam sebuah dokumen (TF) dan kebalikan dari frekuensi kata tersebut di seluruh kumpulan dokumen (IDF) [15]. Nilai IDF dapat dihitung menggunakan persamaan 1 [16] :

$$IDF_t = \log(\frac{N}{n_t}) + 1 \quad (1)$$

N merupakan jumlah semua dokumen, $n_t$ merupakan jumlah dokumen yang mengandung kata tertentu. Bobot akhir TF-IDF kata dapat dihitung dengan mengalikan nilai TF dengan IDF sesuai persamaan 2:

$$W_{t,d} = TF_{t,d} * IDF_t \quad (2)$$

$W_{t,d}$ adalah bobot dari *t (term)* dalam satu dokumen, $TF_{t,d}$ merupakan frekuensi kemunculan *t* (*term*) dalam dokumen *d* dan $IDF_t$ merupakan *Inverse Document Frequency* dimana IDF didapatkan melalui persamaan 1.

*2.5 Cosine Similarity*

*Cosine similarity* digunakan untuk mengetahui nilai *similarity* diantara dua dokumen seperti d1 dan d2 yang masing-masing dokumen dinyatakan dalam suatu vektor. Nilai *cosine similarity* dapat dihitung menggunakan persamaan 3 [6] :

$$CosSim\ (d_i, q_i) = \frac{q_i\ .\ d_i}{|q_i||d_i|} = \frac{\sum_{j=1}^{t}(q_{ij}\ .\ d_{ij})}{\sqrt{\sum_{j=1}^{t}(q_{ij})^2\ .\ \sum_{j=1}^{t}(d_{ij})^2}} \quad (3)$$

$d_i$ dan $q_i$ merupakan dua vektor kalimat yang ingin diukur tingkat kemiripannya. Nilai *cosine similarity* dihitung dengan membagi antara *dot product* kedua vektor dengan *cross product* vektor. Variabel $q_{ij}$ merupakan bobot istilah *j* pada dokumen $q_i$ dan $d_{ij}$ bobot istilah *j* pada dokumen $d_i$.

*2.6 Peringkasan MMR*

Setelah kata kunci dari masing-masing diketahui melalui proses pemodelan topik menggunakan LDA, proses selanjutnya yaitu memilih kalimat untuk dijadikan ringkasan menggunakan metode MMR. *Maximum Marginal Relevance* (MMR) merupakan salah satu metode ekstraksi ringkasan *single document* atau *multi document* yang diusulkan oleh Carbonell dan Goldstein pada tahun 1998 [17].

MMR memberi peringkat pada kalimat berdasarkan kombinasi matriks *cosine similarity* sebagai tanggapan dari kueri yang diberikan. Perhitungan MMR dilakukan dengan membandingkan hasil relevansi kueri dan hasil kesamaan kalimat. Jika suatu dokumen memiliki bobot kesamaan maksimum dengan kueri dan relevan dengan isi dokumen, maka dokumen tersebut dianggap memiliki *marginal relevance* yang tinggi. Skor dari MMR dapat dihitung dengan persamaan 4.

$$MMR = \text{Argmax}_{D_i \in \text{R}\setminus\text{S}}[\lambda * Sim_1(D_i, \text{Q}) - (1-\lambda)\text{Max}_{D_j \in \text{S}}\ (Sim_2(D_i, D_j))] \quad (4)$$

*R* adalah kumpulan kalimat dalam satu dokumen yang akan diekstrak menjadi ringkasan. *S* adalah kumpulan kalimat di *R* yang sudah terpilih sebagai ringkasan. *R\S* adalah selisih

himpunan, yaitu himpunan kalimat di *R* yang belum terpilih menjadi ringkasan. $D_i$ merupakan kalimat ke-i pada suatu dokumen yang akan dipilih sebagai ringkasan, sedangkan $D_j$ merupakan kalimat ke-j yang sudah dipilih sebagai ringkasan. $Sim_1$ merupakan nilai kemiripan antara kalimat yang belum terpilih sebagai ringkaasan dengan *query*, sedangkan $Sim_2$ merupakan nilai kemiripan antara kalimat yang belum terpilih sebagai ringkasan dengan kalimat yang sudah dipilih sebagai ringkasan. Relevansi kalimat diatur dan redundansi dikurangi dengan nilai parameter $\lambda$. Kisaran nilai parameter $\lambda$ adalah *range* [0,1], mulai dari 0 sampai 1. Nilai MMR yang dihitung biasanya relevan dengan dokumen asli untuk parameter $\lambda = 1$, sedangkan nilai MMR akan relevan dengan kalimat yang diekstrak sebelumnya ketika $\lambda = 0$.

*2.7 Evaluasi*

Evaluasi hasil ringkasan pada penelitian ini menggunakan skor ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*). ROUGE merupakan metrik yang digunakan untuk mengevaluasi sistem peringkasan teks dan model terjemahan. ROUGE menghitung kata-kata yang tumpang tindih antara ringkasan sistem dan ringkasan referensi serta bobotnya masing-masing [18]. Unit yang sesuai, seperti *n-gram* (n-kata), urutan kata, dan pasangan kata antara ringkasan sistem dan ringkasan referensi dihitung untuk melakukan pengukuran nilai ROUGE. ROUGE-N mengukur jumlah *n-gram* yang cocok antara teks yang dihasilkan sistem dan ringkasan referensi yang dilakukan manusia. N menunjukkan jumlah *n-gram* yang bisa berupa 1 atau lebih. Pada penelitian ini menggunakan ROUGE-1 untuk mengetahui kualitas hasil ringkasan sistem. Nilai *recall, precision,* dan *f-score* ROUGE dapat dihitung melalui persamaan berikut:

$$Recall = \frac{\sum_{s \in sys} \sum_{gramN \in s} Count_{match}(gram_N)}{\sum_{s \in ref} \sum_{gramN \in s} Count(gram_N)} \quad (5)$$

Nilai *recall* dihitung dengan persamaan 5 di mana *s* adalah kalimat atau *sentence* yang ada pada ringkasan, *ref* merupakan ringkasan referensi, $Count(gram_N)$ merupakan jumlah N-gram yang ada pada ringkasan referensi. $Count_{match}(gram_N)$ adalah jumlah maksimum N-gram yang muncul dalam ringkasan sistem dan ringkasan referensi.

$$Precision = \frac{\sum_{s \in sys} \sum_{gramN \in s} Count_{match}(gram_N)}{\sum_{s \in sys} \sum_{gramN \in s} Count(gram_N)} \quad (6)$$

Nilai *precision* dihitung dengan persamaan 6 di mana *sys* merupakan ringkasan sistem, $Count(gram_N)$ merupakan jumlah N-gram yang ada pada ringkasan sistem. $Count_{match}(gram_N)$ adalah jumlah maksimum N-gram yang muncul dalam ringkasan sistem dan ringkasan referensi.

$$F\text{-}Score = \frac{(1+\beta^2)(Precision * Recall)}{(\beta^2 * Precision + Recall)} \quad (7)$$

Variabel $\beta$ adalah parameter yang menentukan bobot relatif dari *precision* dan *recall*, biasanya diatur ke 1 untuk menyeimbangkan *precision* dan *recall*, sehingga *F-Score* menjadi *F1-Score. F1-score* merupakan hasil kombinasi antara nilai *recall* dan *precision* untuk mengukur kinerja suatu sistem.

## 3. HASIL DAN PEMBAHASAN

Uji coba dilakukan pada 2500 data artikel pertama pada file *train.03 jsonl* yang merupakan bagian dari *dataset IndoSum*. Skenario uji coba dilakukan dari artikel ke-1 sampai dengan artikel ke-2500 untuk mengukur nilai ROUGE-1 pada masing-masing hasil ringkasan sistem. Sebelum dilakukan peringkasan setiap artikel melalui proses pemodelan topik dengan metode *Latent Dirichlet Allocation* (LDA) yang menghasilkan satu topik dan sepuluh kata kunci. Proses pemodelan topik dengan metode LDA menggunakan nilai *alpha* = 1/K dan *eta* = 1/V. Sepuluh kata kunci dari hasil pemodelan topik LDA digunakan sebagai *query* untuk menentukan

ringkasan menggunakan metode *Maximum Marginal Relevance*. Proses peringkasan dengan metode MMR menggunakan tiga variasi nilai *lambda* yaitu 0.5, 0.7, dan 0.9. Panjang ringkasan sistem yang dihasilkan sebesar 50% dan 30% dari keseluruhan kalimat pada teks asli.

Tabel 1 menampilkan hasil ekstraksi kata kunci pada setiap artikel yang dihasilkan dari pemodelan topik menggunakan LDA. Masing-masing artikel memiliki sepuluh kata kunci yang mewakili artikel tersebut.

Tabel 1 Hasil Ekstraksi Kata Kunci Dengan LDA

| **Dokumen ke-** | **Kata Kunci** |
|---|---|
| 1 | aksi, abu, ormas, gelar, bakal, usaha, rencana, organisasi, selesai, cepat |
| 2 | herkis, arema, hilang, main, poin, waspada, cedera, pilar, bekap, latih |
| 3 | bank, pt, indonesia, kredit, kartu, data, uang, menteri, informasi, djp |
| 4 | negara, sahabat, phu, batas, kawasan, bangsa, mangindaan, soal, zee, temu |
| 5 | nikmat, suka, tunjuk, emerson, indonesia, dunia, wayang, amerika, serikat, asal |
| … | … |
| 2498 | kuarter, cavaliers, warriors, poin, main, curry, laku, cetak, unggul, double |
| 2499 | album, lagu, band, sih, big, bilang, udah, gihon, vokal, enggak |
| 2500 | android, go, ponsel, smartphone, bakal, kabar, murah, google, pabrikan, sebut |

Skenario ke-1 menggunakan nilai parameter *lambda (λ)* pada *Maximum Marginal Relevance* (MMR) sebesar 0.5. Setiap artikel mempunyai ringkasan dengan tingkat kompresi 30% dan 50%. Tabel 2 menampilkan statistik kata dan kalimat pada hasil ringkasan yang diperoleh dari peringkasan manual oleh manusia dan peringkasan otomatis oleh sistem.

Tabel 2 Statistik Jumlah Kata dan Kalimat Pada Skenario 1

| **Nomor Dokumen** | ***Compression rate*** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | **50%** | | | | | **30%** | | | | |
| | **Kata Unik Rsistem** | **Kata Unik Rmanusia** | **Kata Unik overlap** | **Jumlah Kalimat Rsistem** | **Jumlah Kalimat Rmanusia** | **Kata Unik Rsistem** | **Kata Unik Rmanusia** | **Kata Unik overlap** | **Jumlah Kalimat Rsistem** | **Jumlah Kalimat Rmanusia** |
| 1 | 86 | 57 | 53 | 7 | 4 | 56 | 57 | 32 | 5 | 4 |
| 2 | 69 | 52 | 48 | 6 | 5 | 50 | 52 | 38 | 4 | 5 |
| 3 | 149 | 49 | 46 | 14 | 4 | 107 | 49 | 43 | 10 | 4 |
| 4 | 110 | 55 | 55 | 12 | 4 | 60 | 55 | 18 | 8 | 4 |
| 5 | 130 | 54 | 39 | 10 | 3 | 86 | 54 | 38 | 7 | 3 |
| … | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 2496 | 26 | 49 | 10 | 4 | 4 | 18 | 49 | 9 | 3 | 4 |
| 2497 | 122 | 53 | 39 | 13 | 5 | 74 | 53 | 35 | 8 | 5 |
| 2498 | 91 | 43 | 22 | 11 | 5 | 67 | 43 | 12 | 7 | 5 |
| 2499 | 117 | 45 | 45 | 10 | 4 | 76 | 45 | 45 | 7 | 4 |
| 2500 | 55 | 44 | 25 | 6 | 4 | 44 | 44 | 25 | 5 | 4 |

Pada skenario ke-2, menggunakan nilai parameter *lambda (λ)* pada MMR sebesar 0.7. Tabel 3 menampilkan statistik kata dan kalimat pada hasil ringkasan yang diperoleh dari peringkasan manual dan peringkasan sistem.

Tabel 3 Statistik Jumlah Kata dan Kalimat Pada Skenario 2

| **Nomor Dokumen** | ***Compression rate*** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | **50%** | | | | | **30%** | | | | |
| | **Kata Unik Rsistem** | **Kata Unik Rmanusia** | **Kata Unik overlap** | **Jumlah Kalimat Rsistem** | **Jumlah Kalimat Rmanusia** | **Kata Unik Rsistem** | **Kata Unik Rmanusia** | **Kata Unik overlap** | **Jumlah Kalimat Rsistem** | **Jumlah Kalimat Rmanusia** |
| 1 | 78 | 57 | 44 | 7 | 4 | 64 | 57 | 43 | 5 | 4 |
| 2 | 72 | 52 | 37 | 6 | 5 | 41 | 52 | 34 | 4 | 5 |
| 3 | 147 | 49 | 37 | 14 | 4 | 95 | 49 | 34 | 10 | 4 |
| 4 | 91 | 55 | 42 | 12 | 4 | 66 | 55 | 28 | 8 | 4 |
| 5 | 143 | 54 | 40 | 11 | 3 | 91 | 54 | 38 | 7 | 3 |
| … | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 2496 | 26 | 49 | 10 | 4 | 4 | 18 | 49 | 9 | 3 | 4 |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 2497 | 92 | 53 | 19 | 13 | 5 | 55 | 53 | 17 | 8 | 5 |
| 2498 | 104 | 43 | 30 | 11 | 5 | 67 | 43 | 12 | 7 | 5 |
| 2499 | 125 | 45 | 45 | 10 | 4 | 89 | 45 | 45 | 7 | 4 |
| 2500 | 78 | 44 | 25 | 7 | 4 | 39 | 44 | 13 | 5 | 4 |

Pada skenario ke-3, menggunakan nilai parameter *lambda (λ)* dalam *Maximum Marginal Relevance* (MMR) sebesar 0.9. Tabel 4 menampilkan statistik kata dan kalimat dari hasil peringkasan manual oleh manusia dan peringkasan oleh sistem untuk artikel dengan tingkat kompresi 30% dan 50%.

Tabel 4 Statistik Jumlah Kata dan Kalimat Pada Skenario 3

| **Nomor Dokumen** | ***Compression rate*** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | **50%** | | | | | **30%** | | | | |
| | **Kata Unik Rsistem** | **Kata Unik Rmanusia** | **Kata Unik overlap** | **Jumlah Kalimat Rsistem** | **Jumlah Kalimat Rmanusia** | **Kata Unik Rsistem** | **Kata Unik Rmanusia** | **Kata Unik overlap** | **Jumlah Kalimat Rsistem** | **Jumlah Kalimat Rmanusia** |
| 1 | 74 | 57 | 44 | 7 | 4 | 64 | 57 | 43 | 5 | 4 |
| 2 | 69 | 52 | 48 | 6 | 5 | 41 | 52 | 34 | 4 | 5 |
| 3 | 103 | 49 | 35 | 14 | 4 | 80 | 49 | 35 | 10 | 4 |
| 4 | 105 | 55 | 55 | 12 | 4 | 66 | 55 | 28 | 8 | 4 |
| 5 | 143 | 54 | 40 | 11 | 3 | 91 | 54 | 38 | 7 | 3 |
| … | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 2496 | 26 | 49 | 10 | 4 | 4 | 18 | 49 | 9 | 3 | 4 |
| 2497 | 93 | 53 | 19 | 13 | 5 | 59 | 53 | 18 | 8 | 5 |
| 2498 | 97 | 43 | 26 | 11 | 5 | 69 | 43 | 11 | 7 | 5 |
| 2499 | 128 | 45 | 45 | 10 | 4 | 89 | 45 | 45 | 7 | 4 |
| 2500 | 78 | 44 | 25 | 7 | 4 | 39 | 44 | 13 | 5 | 4 |

Berdasarkan hasil keseluruhan evaluasi dari 2500 artikel yang diujicobakan, Tabel 5 menampilkan rata-rata hasil akhir sistem ketika dibangun menggunakan dua tingkat kompresi berbeda dan tiga variasi nilai *lambda*.

Tabel 5 Rata-rata hasil evaluasi ROUGE-1

| **Skenario** | ***Compression rate*** | | | | | |
|---|---|---|---|---|---|---|
| | **50%** | | | **30%** | | |
| | **Rata-rata *Recall*** | **Rata-rata *Precision*** | **Rata-rata *F1-score*** | **Rata-rata *Recall*** | **Rata-rata *Precision*** | **Rata-rata *F1-score*** |
| 1 (λ = 0.5) | 0.680 | 0.352 | 0.453 | 0.515 | 0.397 | 0.435 |
| 2 (λ = 0.7) | 0.723 | 0.374 | 0.482 | 0.541 | 0.418 | 0.458 |
| 3 (λ = 0.9) | 0.726 | 0.380 | 0.488 | 0.542 | 0.425 | 0.462 |

Pada tingkat kompresi 30% hasil terbaik diperoleh dari skenario 3 yaitu MMR dengan nilai *lambda* sebesar 0,9 berdasarkan nilai rata-rata setiap skenario pengujian. Skenario 3 memperoleh rata-rata *recall* 0.542, *precision* 0.425, dan *f1-score* 0.462 untuk *compression rate* 30% sedangkan pada *compression rate* 50% nilai rata-rata *recall* 0.726, *precision* 0.380, dan *f1-score* 0.488. Berdasarkan hasil rata-rata pada Tabel 5, nilai *recall* tinggi dikarenakan hasil ringkasan manusia pada penelitian ini lebih sedikit dari ringkasan sistem, sehingga tidak terjadi keseimbangan antara nilai *recall* dan *precision*. Tingkat kompresi 30% hasil ringkasan sistem jumlahnya tidak berbeda jauh dengan ringkasan manual sehingga nilai *recall* dan *precision* tidak terlalu berbeda.

## 4. KESIMPULAN DAN SARAN

Berdasarkan pengujian sistem peringkasan teks berita berbahasa Indonesia menggunakan *smooth Latent Dirichlet Allocation* (LDA) dan *Maximum Marginal Relevance* (MMR) dengan 2500 data artikel berita menghasilkan nilai rata-rata ROUGE-1 terbaik pada skenario *lambda*=0.9 sebesar 0.488 untuk tingkat kompresi 50% dan 0.462 untuk tingkat kompresi 30%. Nilai ROUGE-

1 terendah didapatkan oleh skenario *lambda*=0.5 yaitu 0.453 untuk tingkat kompresi 50% dan 0.435 untuk tingkat kompresi 30%. Berdasarkan hasil percobaan menunjukkan bahwa sistem dapat menghasilkan ringkasan yang cukup relevan dengan menggunakan kata kunci yang diekstrak dari konten berita untuk menghindari judul yang berpotensi *clickbait* atau kurang sesuai dengan isi berita. Sistem peringkasan ini dapat menyampaikan informasi penting dan esensial secara ringkas dengan waktu yang lebih singkat dari pada meringkas secara manual.

Saran untuk penelitian selanjutnya yaitu menggunakan data ringkasan manual yang lebih panjang, dikarenakan ringkasan yang terlalu sedikit berpotensi kehilangan informasi penting yang ada dalam artikel berita. Kemudian untuk pemodelan topik dapat menambah kamus *stopword* agar model LDA dapat menghasilkan kata kunci yang lebih spesifik serta mencoba beberapa variasi nilai *Alpha* dan *Eta* untuk mengetahui variasi nilai yang menghasilkan kata kunci lebih baik.

## DAFTAR PUSTAKA


[1] D. Samsudin, "POLA PENGEMBANGAN PARAGRAF PEMBUKA DALAM BERITA UTAMA KORAN DI CIREBON DAN DI BOGOR JAWA BARAT (The Development Pattern of Opening Paragraph in the Newspaper Headlines in the City of Bogor and Cirebon West Java)," *Sirok Bastra*, vol. 7, no. 1, 2019, doi: 10.37671/sb.v7i1.153.

[2] R. C. Belwal, S. Rai, and A. Gupta, "Extractive text summarization using clustering-based topic modeling," *Soft Comput.*, vol. 27, no. 7, pp. 3965–3982, 2023, doi: 10.1007/s00500-022-07534-6.

[3] I. R. Musyaffanto, G. Budi Herwanto, and M. Riasetiawan, "Automatic extractive text summarization for indonesian news articles using maximal marginal relevance and non-negative matrix factorization," *Proc. - 2019 5th Int. Conf. Sci. Technol. ICST 2019*, no. July 2019, 2019, doi: 10.1109/ICST47872.2019.9166376.

[4] Y. Ananda Kresna, I. Cholissodin, and Indriati, "Peringkasan Teks Menggunakan Metode Maximum Marginal Relevance terhadap Artikel Berita terkait COVID-19," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 9, pp. 3901–3907, 2021, [Online]. Available: http://j-ptiik.ub.ac.id

[5] D. P. Purbawa, Malikhah, R. N. E. Anggraini, and R. Sarno, "Automatic Text Summarization using Maximum Marginal Relevance for Health Ethics Protocol Document in Bahasa," *Proc. 2021 13th Int. Conf. Inf. Commun. Technol. Syst. ICTS 2021*, no. January 2022, pp. 324–329, 2021, doi: 10.1109/ICTS52701.2021.9607951.

[6] E. Susanto, V. C. Mawardi, and M. D. Lauro, "Aplikasi Clustering Berita Dengan Metode K Means Dan Peringkas Berita Dengan Metode Maximum Marginal Relevance," *J. Ilmu Komput. dan Sist. Inf.*, vol. 9, no. 1, p. 62, 2021, doi: 10.24912/jiksi.v9i1.11560.

[7] D. Firman A, I. Fahrur Rozi, and I. Kusumaning Putri, "Peringkasan Teks Otomatis pada Portal Berita Olahraga menggunakan metode Maximum Marginal Relevance.," *J. Inform. Polinema*, vol. 8, no. 3, pp. 21–30, 2022, doi: 10.33795/jip.v8i3.519.

[8] V. Vanessa and A. L. Ibrahim, "Clickbait as a Potential Threat in the Development of Cybercrime in Indonesia," *J. USM Law Rev.*, vol. 7, no. 1, pp. 1–17, 2023.

[9] L. Atikah, N. A. Hasanah, and A. Z. Arifin, "Topic Modelling Using VSM-LDA For Document Summarization," *Ultim. J. Tek. Inform.*, vol. 14, no. 2, pp. 91–95, 2022, doi: 10.31937/ti.v14i2.2854.

[10] K. A. R. Issam, S. Patel*, and S. C. N., "Topic Modeling Based Extractive Text Summarization," *Int. J. Innov. Technol. Explor. Eng.*, vol. 9, no. 6, pp. 1710–1719, 2020, doi: 10.35940/ijitee.f4611.049620.

[11] M. Alfhi Saputra, "Peringkas Teks Otomatis Bahasa Indonesia secara Abstraktif Menggunakan Metode Long Short-Term Memory," *e-Proceeding Eng. Vol.8, No.2 April 2021* |, vol. 8, no. 2, pp. 3474–3488, 2021.

[12] B. H. Puspita, M. Muhajir, and H. Aliady, "Topic Modeling Using Latent Dirichlet Allocation (LDA) and Sentiment Analysis for Marketing Planning Tiket.com," vol. 474,

no. Isstec 2019, pp. 16–22, 2020, doi: 10.2991/assehr.k.201010.004.

[13] A. F. Hidayatullah, S. K. Aditya, Karimah, and S. T. Gardini, "Topic modeling of weather and climate condition on twitter using latent dirichlet allocation (LDA)," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 482, no. 1, 2019, doi: 10.1088/1757-899X/482/1/012033.

[14] A. Eka Prasetiyanto, Kusrini, and A. D. Hartanto, "Analisis Review Siswa Selama Pembelajaran pada Masa Pandemi Menggunakan Metode Topic Modelling LDA," *Stain. (SEMINAR Nas. Teknol. & SAINS)*, vol. 1, no. 1, pp. 241–246, 2022, [Online]. Available: https://proceeding.unpkediri.ac.id/index.php/stains/article/view/1538

[15] A. Nurkholis, D. Alita, and A. Munandar, "Comparison of Kernel Support Vector Machine Multi-Class in PPKM Sentiment Analysis on Twitter," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 2, pp. 227–233, 2022, doi: 10.29207/resti.v6i2.3906.

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

[17] Arie Atwa Magriyanti, "Maximum Marginal Relevance Berbasis Boolean Model Pada Peringkasan Artikel Berita Pendek," *J. Ilm. Tek. Inform. dan Komun.*, vol. 1, no. 3, pp. 77–88, 2021, doi: 10.55606/juitik.v1i3.132.

[18] M. A. Zamzam, C. Crysdian, and K. F. H. Holle, "Sistem Automatic Text Summarization Menggunakan Algoritma Textrank," *Matics*, vol. 12, no. 2, pp. 111–116, 2020, doi: 10.18860/mat.v12i2.8372.