SISTEM PERINGKASAN TEKS BERITA BERBAHASA INDONESIA MENGGUNAKAN *LATENT DIRICHLET ALLOCATION* DAN *MAXIMUM MARGINAL RELEVANCE*

SKRIPSI

Oleh:
BIMA HAMDANI MAWARIDI
NIM. 200605110011

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024

SISTEM PERINGKASAN TEKS BERITA BERBAHASA INDONESIA MENGGUNAKAN *LATENT DIRICHLET ALLOCATION* DAN *MAXIMUM MARGINAL RELEVANCE*

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh:
BIMA HAMDANI MAWARIDI
NIM. 200605110011

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2024

HALAMAN PERSETUJUAN

SISTEM PERINGKASAN TEKS BERITA BERBAHASA INDONESIA MENGGUNAKAN *LATENT DIRICHLET ALLOCATION* DAN *MAXIMUM MARGINAL RELEVANCE*

SKRIPSI

Oleh:
BIMA HAMDANI MAWARIDI
NIM. 200605110011

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 31 Mei 2024

Pembimbing I	Pembimbing II
Dr. Muhammad Faisal, M.T	Hani Nurhayati, M.T
NIP. 19740510 200501 1 007	NIP. 19780625 200801 2 006

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Fachrul Kurniawan, M.MT, IPM
NIP. 19771020 200912 1 001

HALAMAN PENGESAHAN

SISTEM PERINGKASAN TEKS BERITA BERBAHASA INDONESIA MENGGUNAKAN *LATENT DIRICHLET ALLOCATION* DAN *MAXIMUM MARGINAL RELEVANCE*

SKRIPSI

Oleh:
BIMA HAMDANI MAWARIDI
NIM. 200605110011

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 7 Juni 2024

Susunan Dewan Penguji

Ketua Penguji	: Dr. Zainal Abidin, M.Kom NIP. 19760613 200501 1 004	( )
Anggota Penguji I	: Shoffin Nahwa Utama, M.T NIP. 19860703 202012 1 003	( )
Anggota Penguji II	: Dr. Muhammad Faisal, M.T NIP. 19740510 200501 1 007	( )
Anggota Penguji III	: Hani Nurhayati, M.T NIP. 19780625 200801 2 006	( )

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Fachrul Kurniawan, M.MT, IPM
NIP. 19771020 200912 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama	: Bima Hamdani Mawaridi
NIM	: 200605110011
Fakultas / Program Studi	: Sains dan Teknologi / Teknik Informatika
Judul Skripsi	: Sistem Peringkasan Teks Berita Berbahasa Indonesia Menggunakan *Latent Dirichlet Allocation* dan *Maximum Marginal Relevance*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 31 Mei 2024
Yang membuat pernyataan,

Bima Hamdani Mawaridi
NIM. 200605110011

HALAMAN MOTTO

"Tidak ada kesuksesan tanpa kerja keras. Tidak ada keberhasilan tanpa kebersamaan. Tidak ada kemudahan tanpa doa"

(Ridwan Kamil)

HALAMAN PERSEMBAHAN

Puji Syukur atas kehadirat Allah Subhanahu wa ta'ala, karena berkat rahmat dan petunjuk-Nya penulis dapat menyelesaikan skripsi ini.

Penulis mempersembahkan karya ini kepada kedua orang tua penulis, kedua adik perempuan penulis, dosen, teman, dan sahabat yang telah menemani masa perkuliahan serta semua pihak yang telah membantu dalam menyelesaikan skripsi ini.

KATA PENGANTAR

Assalamualaikum Wr. Wb

Puji dan syukur penulis panjatkan kehadirat Tuhan Yang Maha Esa, Allah subhanahu wa ta'ala yang telah memberikan Taufik dan Hidayah-Nya kepada penulis sehingga dapat penyelesaikan skripsi ini dengan baik. Banyak pihak yang terlibat dalam penulisan skripsi ini yang telah memberikan dukungan baik moril maupun materil. Untuk itu dalam kesempatan kali ini penulis ingin mengucapkan banyak terimakasih kepada:

1. Prof. Dr. H. M. Zainuddin, MA, selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang beserta jajarannya.
2. Prof. Dr. Hj. Sri Harini, M.Si, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang beserta jajarannya.
3. Dr. Fachrul Kurniawan M.MT, IPM selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Dr. Muhammad Faisal, M.T selaku Dosen Pembimbing I yang telah dengan sabar memberikan arahan baik dalam penulisan hingga program yang dibuat dalam menyelesaikan skripsi ini.
5. Hani Nurhayati, M.T selaku Dosen Pembimbing II yang telah memberikan bimbingan, arahan serta bantuan dalam terwujudnya karya tulis skripsi ini.
6. Dr. Zainal Abidin, M.Kom selaku ketua penguji, Shoffin Nahwa Utama, M.T dan Ahmad Fahmi Karami, M.Kom selaku penguji I yang telah meluangkan waktunya untuk menguji dan dengan sabar memberi arahan serta saran dalam menyelesaikan skripsi ini.
7. Fatchurrohman, M.Kom selaku Dosen Wali yang telah memberikan arahan dalam proses perkuliahan.
8. Khadijah Fahmi Hayati Holle, M.Kom dan Tri Mukti Lestari, M.Kom yang telah mengajarkan *Natural Language Processing* dan *Information Retrieval* sehingga muncul ide untuk melakukan penelitian di bidang tersebut.
9. Segenap Dosen, Laboran dan jajaran pada Program Studi Teknik Informatika yang telah memberikan bimbingan dan bantuan selama studi.

10. Nia Faricha, S.Si selaku Admin Program Studi Teknik Informatika yang dengan sabar membantu, memberikan arahan informasi terkait perkuliahan.
11. Kedua orangtua penulis Ibu Endarti dan Ayah Akhmad Sahidin yang selalu memberi dukungan dan doa serta selalu memberikan yang terbaik untuk kelancaran putranya dalam pendidikan.
12. Kedua adik penulis Nadhira Nova Sabrina dan Aisyanda Kayla Salsabillah yang selalu memberi semangat dan menghibur di tengah kegiatan padat kuliah.
13. Sahabat penulis "*enter new subject*" yang beranggotakan Zidan, Rizka, Vera terimakasih atas segala bantuan dan semangat yang diberikan dari awal perkuliahan yang masih *online* sampai terselesaikannya skripsi ini.
14. Teman Himatif Encoder, Dema Fakultas Saintek, Komunitas GDSC, Weboender, dan ISC yang telah membantu mengembangkan *softskill* maupun *hardskill* selama masa perkuliahan.
15. Bapak Machsun Zain, Pak Hafidz, Pak Bas, Pak Jaz dan seluruh keluarga Kantor Kemenag Kota Batu yang sudah memberikan pengalaman selama PKL.
16. Teman-teman KKM 42 DEBUTAN yang tetap memberikan *support* dan doa walaupun pengabdian di desa Bunut Wetan, Pakis sudah selesai.
17. Seluruh keluarga besar Saudara Teknik Informatika UIN Malang terkhusus Angkatan 2020 "INTEGER", terimakasih telah memberikan *support*, motivasi dan bantuannya kepada penulis.
18. Seluruh pihak yang telah terlibat secara langsung maupun tidak langsung dalam proses penyusunan skripsi sejauh ini.

Penulis menyadari dalam penulisan skripsi ini tidak luput dari kesalahan yang jauh dari kata sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun sehingga skripsi ini dapat lebih dikembangkan.

Malang, 31 Mei 2024

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

ABSTRAK

Mawaridi, Bima Hamdani. 2024. **Sistem Peringkasan Teks Berita Berbahasa Indonesia Menggunakan *Latent Dirichlet Allocation* dan *Maximum Marginal Relevance*.** Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Muhammad Faisal, M.T (II) Hani Nurhayati, M.T.

.

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

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.

ABSTRACT

Mawaridi, Bima Hamdani. 2024. **Indonesian News Text Summarization System Using Latent Dirichlet Allocation and Maximum Marginal Relevance**. Undergraduate Thesis. Department of Informatics Engineering Faculty of Science and Technology Maulana Malik Ibrahim State Islamic University Malang. Supervisor: (I) Dr. Muhammad Faisal, M.T (II) Hani Nurhayati, M.T.

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

البحث مستخلص

ماوريدي، بيما حمداني. ٢٠٢٤ .**نظام تلخيص نصوص الأخبار التحدث بالإندونيسية باستخدام *Latent Dirichlet Allocation* و *Maximum Marginal Relevance***. الأطروحة. برنامج دراسة هندسة المعلوماتية، كلية العلوم والتكنولوجيا، الجامعة الإسلامية الحكومية، مولانا مالك إبراهيم مالانج. المشرف : (١) الدكتور محمد فيصل الماجستير (٢) هاني نورهاياتي الماجستير

الكلمات الرئيسية : LDA، MMR، نمذجة المواضيع، تلخيص النصوص

التقدم في التكنولوجيا يجعل من السهل العثور على الأخبار على وسائل الإعلام على الإنترنت. يتزايد عدد المقالات ،الإخبارية المتاحة بنصوص طويلة إلى حد ما. وهذا سيجعل من الصعب على قراء الأخبار العثور على المعلومات الأساسية من الأخبار مما يجعل من الصعب على قراء الأخبار العثور على المعلومات الأساسية من الأخبار، لذا هناك حاجة إلى ملخص نصي لمساعدة المستخدمين على فهم جوهر النص دون الحاجة إلى قراءته كله. الطريقة المستخدمة لتلخيص النص هي *Maximum Marginal Relevance* (MMR) من خلال الجمع بين عاملي اختيار وهما الملاءمة والتنوع. غالبًا ما يتبين اليوم أن عناوين الأخبار في *clickbait*، المقالات على الإنترنت لا تمثل محتوى الخبر بشكل كامل أو تسمى لتجنب العناوين غير المناسبة، في هذه الدراسة يعتمد الملخص على الكلمات المفتاحية التي تم إنشاؤها بواسطة طريقة *Latent Dirichlet Allocation* (LDA). أسفرت نتائج الاختبار باستخدام بيانات 2500 مقال إخباري عن أفضل قيمة ROUGE-1 بمتوسط 0.488 لمستوى ضغط 50% و0.462 لمستوى ضغط 30%. أدنى قيمة ROUGE-1 هي 0.453 لمستوى ضغط 50% و0.435 لمستوى ضغط 30%. تُظهر هذه النتائج أن النظام يمكن أن ينتج ملخصًا ذا صلة إلى حد ما باستخدام الكلمات الرئيسية المستخرجة من محتوى الأخبار.

BAB I

PENDAHULUAN

1.1 Latar Belakang

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. Media online merupakan sumber berita utama bagi masyarakat Indonesia berdasarkan survei dari *Reuters Institute* dengan presentase 84% pada tahun 2023. Kemajuan teknologi internet yang sangat pesat berdampak pada cara penyebaran berita online di internet dengan jumlahnya yang semakin meningkat.

Berita online saat ini tersedia dalam bentuk artikel yang mudah ditemukan di mana saja karena ditempatkan pada internet serta dapat dilihat kapan saja. Banyaknya berita yang masuk dari berbagai arah membuat para pembaca memiliki banyak pilihan untuk mengakses media pemberitaan yang mereka percaya. Adanya artikel berita memungkinkan seseorang untuk tetap *update* dengan peristiwa dan informasi terbaru dari berbagai bidang seperti teknologi, kesehatan, olahraga, dan lain-lain.

Informasi yang ada pada sebuah artikel berita cukup beragam dan padat. Pembaca mungkin 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. Ketika pembaca merupakan seseorang yang mempunyai kegiatan padat seperti mahasiswa atau pekerja kantoran, membaca keseluruhan teks yang panjang dapat membuat pekerjaan lainnya tertunda. Oleh karena itu mendapatkan inti informasi dengan cepat bermanfaat dalam membuat suatu keputusan atau menyelesaikan tugas dengan cepat pula. Dengan demikian, waktu dapat dimanfaatkan secara maksimal untuk kegiatan atau keputusan lainnya.

Allah tidak menginginkan hamba-Nya menjalani waktu tanpa produktivitas, karena waktu bagi seorang mukmin merupakan sebuah perputaran yang tidak pernah putus, tidak pernah kosong dengan aktivitas yang membawa manfaat, dan dilakukan dengan bersungguh-sungguh (Murniyetti, 2016). Dalam Al-Qur'an surah Al-Insyirah Ayat 7 Allah subhanahu wa ta'ala berfirman:

فَإِذَا فَرَغْتَ فَانْصَبْ ۝٧

Maka apabila kamu telah selesai (dari sesuatu urusan), kerjakanlah dengan sungguh-sungguh (urusan) yang lain, (QS. Al-Insyirah/94: 7)

Dalam tafsir Ibnu Katsir dijelaskan bahwa Allah subhanahu wa ta'ala berfirman apabila kamu telah merampungkan urusan-urusan duniamu dan kesibukannya serta telah kamu selesaikan semua yang berkaitan dengannya, maka bulatkanlah tekadmu untuk ibadah dan bangkitlah kamu kepada-Nya dalam keadaan bersemangat. Curahkanlah hatimu dan ikhlaskanlah niatmu dalam beribadah dan berharap kepada-Nya. Hal tersebut menunjukkan bahwa setiap orang dianjurkan untuk produktif dengan segera merampungkan suatu urusan sehingga tidak ada waktu yang sia-sia dan bisa digunakan untuk beribadah ketika urusan duniawi sudah selesai. 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.

Natural Language Processing (NLP) merupakan cabang dari ilmu komputer yang berfokus pada pemrosesan bahasa. NLP bertujuan untuk membuat komputer dapat memahami, memproses, dan mengolah bahasa alami seperti yang digunakan oleh manusia (Sivarethinamohan *et al*., 2021). NLP mencakup tugas seperti pengenalan kalimat, pengenalan kata, dan pengambilan keputusan. Penerapan NLP ini dapat ditemukan pada mesin penerjemah, pendeteksi email spam, peringkasan teks, dan *chatbot*. Kemampuan NLP menganalisis bahasa dapat diaplikasikan pada sistem peringkasan otomatis.

Tujuan dari sistem peringkasan otomatis adalah mengkaji berbagai pendekatan atau algoritma untuk meringkas teks. Jika dilihat dari inputnya, tipe peringkasan teks otomatis dapat menggunakan *single* dokumen atau *multi* dokumen, jika dilihat dari keluarannya dapat dibedakan menjadi bentuk ekstraktif dan abstraktif (Belwal *et al*., 2023). Sistem peringkasan otomatis dapat menggunakan metode berbasis kueri dimana hasil ringkasan didasarkan pada kueri atau topik tertentu (Abdi *et al*., 2018).

Metode *Maximum Marginal Relevance* (MMR) merupakan salah satu teknik yang telah dikembangkan untuk melakukan peringkasan teks. Tujuan utama dari MMR adalah untuk memilih himpunan kalimat yang relevan dengan kueri dan beragam satu sama lain (Saraswati *et al*., 2018). Penelitian pernah dilakukan oleh (Purbawa *et al*., 2021) dengan melakukan peringkasan menggunakan dua metode

yaitu MMR dan *TextRank*. Penelitian tersebut menggunakan dua dokumen protokol etika kesehatan sebagai bahan uji coba untuk sistem peringkasan. Hasil dari ringkasan sistem yang menggunakan metode MMR memiliki nilai evaluasi lebih tinggi daripada menggunakan metode peringkasan lain yaitu *TextRank*. Pada penelitian ini, sebelum dilakukan peringkasan menggunakan MMR akan dilakukan terlebih dahulu pemodelan topik untuk mendapatkan kueri topik dari setiap artikel berita.

Pada umumnya peringkasan dengan metode MMR menggunakan judul sebagai kuerinya (Firman A *et al*., 2022). Namun yang sering terjadi saat ini, judul berita belum sepenuhnya mewakili isi berita atau disebut *clickbait* sehingga pembaca perlu membaca keseluruhan berita untuk mengetahui konteks berita tersebut. Dampak *clickbait* dalam judul berita yaitu dapat menimbulkan adanya *hoax*, terpotongnya berita, serta masyarakat merasakan keresahan dengan ketidaksesuaian antara judul dan konten berita (Almajid & Wirawanda, 2023). Menurut Tomy Tresnady, seorang jurnalis pembuat konten untuk website dan media sosial, menyatakan bahwa jurnalis sengaja memanfaatkan *clickbait* seperti menggunakan manipulasi gambar atau sekedar judul artikel yang mereka tulis untuk menarik minat pembaca (Vanessa & Ibrahim, 2023). Oleh karena itu pada penelitian ini menghindari penggunaan judul sebagai kueri MMR tetapi menggunakan kata kunci yang diekstrak dari teks artikel.

Kata kunci suatu teks bisa didapatkan secara otomatis, salah satunya menggunakan metode *Latent Dirichlet Allocation* (LDA). LDA merupakan salah satu metode pemodelan topik yang bertujuan untuk menemukan topik secara

otomatis dalam kumpulan data atau *corpus*. LDA pernah digunakan oleh (Atikah *et al*., 2022) sebagai pemodelan topik yang diteruskan membentuk suatu ringkasan. Metode LDA berperan untuk mengidentifikasi topik pada teks kemudian akan dilakukan peringkasan berdasarkan topik yang didapatkan. Penelitian tersebut menggunakan metode *TextRank* sebagai pendekatan untuk peringkasan teks yang diujicobakan pada 1300 data twitter. Pada penelitian ini, LDA digunakan untuk pemodelan topik pada data artikel berita berbahasa Indonesia dan dalam metode peringkasannya tidak menggunakan *TextRank* tetapi menggunakan metode *Maximum Marginal Relevance* (MMR).

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 *single* dokumen 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.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, maka rumusan masalah penelitian ini yaitu berapa nilai relevansi dari hasil peringkasan teks artikel berita berbahasa Indonesia menggunakan kombinasi metode *Latent Dirichlet Allocation* (LDA) dan *Maximum Marginal Relevance* (MMR)?

1.3 Batasan Masalah

1. Penelitian ini berfokus pada peringkasan ekstraktif dengan inputan *single* dokumen.
2. Penelitian ini menggunakan *Indonesian Text Summarization (IndoSum) dataset*.

1.4 Tujuan Penelitian

Membangun sebuah sistem peringkasan teks menggunakan kombinasi metode *Latent Dirichlet Allocation* (LDA) dan *Maximum Marginal Relevance* (MMR) yang dievaluasi menggunakan ROUGE.

1.5 Manfaat Penelitian

Sistem peringkasan teks dapat dimanfaatkan oleh jurnalis, akademisi, ataupun pekerjaan lain yang membutuhkan peringkasan teks. Sistem akan membantu pengguna menghemat waktu dengan memberikan ringkasan singkat dari teks yang lebih panjang. Sistem juga dapat membantu pengguna memahami inti dari suatu teks tanpa perlu membaca seluruhnya. Dalam analisis informasi, sistem peringkasan teks dapat membantu analis untuk dengan cepat menemukan informasi yang relevan dari suatu sumber.

BAB II

STUDI PUSTAKA

2.1 Penelitian Terkait

Penelitian pernah dilakukan oleh (Savanti *et al*., 2018) dengan judul "Peringkasan Teks Otomatis Secara Ekstraktif Pada Artikel Berita Kesehatan Berbahasa Indonesia Dengan Menggunakan Metode *Latent Semantic Analysis*". Data yang diujikan adalah dokumen artikel berita kesehatan yang didapat dari website berita online kompas.com yang akan diringkas oleh sistem sebanyak 10 dokumen. Pengujian pada *compression rate* 50% mendapatkan nilai akurasi, *precision*, *recall* dan *f1-score* secara berturut 0.668, 0.743, 0.700 dan 0.690 sedangkan pada *compression rate* 40% sebesar 0.696, 0.605, 0.642 dan 0.663. Perbedaan penelitian terletak pada metode dan data yang digunakan, pada penelitian rujukan menggunakan *Latent Semantic Analysis* sedangkan pada penelitian ini menggunakan metode MMR dan LDA. Data yang digunakan pada penelitian rujukan yaitu data berita kesehatan dari kompas.com sedangkan pada penelitian ini menggunakan artikel dari *dataset IndoSum*.

Penelitian pernah dilakukan oleh (Gunawan *et al*., 2023) dengan judul "*Maximum Marginal Relevance and Vector Space Model for Summarizing Students' Final Project Abstracts"*. Pada penelitian tersebut menggunakan abstrak projek akhir yang diambil dari 200 tugas akhir mahasiswa dan dokumen skripsi sebagai objek yang diujicobakan pada sistem peringkasan. Tahapan dalam penelitian tersebut yaitu *preprocessing*, menghitung bobot kalimat menggunakan

TF-IDF, menghitung bobot relevansi kueri menggunakan *Vector Space Model*, menghitung kesamaan kalimat menggunakan *cosine similarity* serta yang terakhir MMR untuk ekstraksi ringkasan. Hasil percobaan dengan membandingkan ringkasan sistem dengan ringkasan manusia menghasilkan nilai rata-rata presisi sebesar 88%, *recall* sebesar 61%, dan *f1-score* sebesar 70%. Perbedaan penelitian terletak pada objek yang diringkas pada penelitian rujukan adalah abstrak tugas akhir sedangkan pada penelitian ini yaitu artikel berita. Penelitian rujukan hanya menggunakan MMR sedangkan pada penelitian ini menggunakan kombinasi MMR dan LDA.

Penelitian pernah dilakukan oleh (Andika *et al*., 2021) dengan judul "Pemodelan Topik Berita pada Portal Berita Online Berbahasa Indonesia menggunakan *Latent Dirichlet Allocation* (LDA)". Penelitian tersebut berfokus pada pemodelan topik menggunakan data dari portal berita detik.com sebanyak 68.537 artikel. Data melalui proses *preprocessing, feature extraction*, pemodelan topik LDA, dan menghitung nilai koherensi model. Pemodelan topik yang menerapkan LDA mendapatkan hasil terbaik dengan membentuk lima buah topik dengan menggunakan pengukuran cv menunjukkan hasil koheren sebesar 0.67. Penelitian rujukan sebatas melakukan pemodelan topik tanpa dilanjutkan peringkasan sedangkan penelitian ini setelah pemodelan topik dilanjutkan melakukan peringkasan. Data artikel pada penelitian rujukan berasal dari situs detik.com sedangkan penelitian ini menggunakan *dataset IndoSum* yang berasal dari beberapa situs penyedia berita.

Penelitian pernah dilakukan oleh (Issam *et al*., 2020) dengan judul "*Topic Modeling Based Extractive Text Summarization"*. Penelitian tersebut menggunakan metode pemodelan topik yaitu *Latent Dirichlet Allocation* dan metode peringkasan *TextRank*. Sebelum meringkas dilakukan dahulu pemodelan topik dan pengelompokan topik menggunakan metode LDA. Hasil dari evaluasi menggunakan ROUGE yaitu ROUGE-1 sebesar 27,08; ROUGE-2 sebesar 6,89; dan ROUGE-L sebesar 25,43. Penelitian rujukan menggunakan metode peringkasan *TextRank* sedangkan pada penelitian ini menggunakan metode MMR. *Dataset* yang digunakan pada penelitian rujukan yaitu *WikiHow datasets* yang berisi artikel bahasa Inggris dari situs *WikiHow* sedangkan penelitian yang diusulkan menggunakan *dataset IndoSum* yang berisi artikel bahasa Indonesia.

Penelitian pernah dilakukan oleh (Halimah *et al*., 2022) "Peringkasan Teks Otomatis (*Automated Text Summarization*) pada Artikel Berbahasa Indonesia Menggunakan Algoritma *Lexrank*". Penelitian tersebut menggunakan 300 data artikel dari internet. Sistem akan melakukan *preprocessing* terlebih dahulu kemudian pembentukan graf dan perangkingan menggunakan *Lexrank*. Tingkat kompresi 50% menghasilkan nilai rata-rata *f1-score* masing-masing sebesar 67,53% pada ROUGE-1, variasi ROUGE-2 sebesar 59,10%, dan ROUGE-L sebesar 67,05%. Perbedaan penelitian terletak pada metode peringkasan yang digunakan, penelitian rujukan menggunakan metode *Lexrank* sedangkan pada penelitian ini menggunakan metode MMR. Data penelitian rujukan menggunakan 300 artikel berita dari internet, sedangkan data uji penelitian ini menggunakan *dataset IndoSum*.

Penelitian pernah dilakukan oleh (Juna & Hayaty, 2023) dengan judul *"The Observed Preprocessing Strategies for Doing Automatic Text Summarizing"*. Penelitian tersebut berfokus pada perbandingan variasi *preprocessing* dalam menghasilkan ringkasan menggunakan metode *Bidirectional Encoder Representations from Transformers* (BERT). Penelitian menggunakan data berita berbahasa Indonesia sebanyak 3762 artikel. Dilakukan 16 percobaan untuk menilai keakuratan hasil ringkasan sebelum dan sesudah penerapan metode *preprocessing*. Perbedaan masing-masing skenario terdapat pada bagian *preprocessing*. Hasil terbaik didapat dengan menggabungkan proses *data cleaning* dan *case folding* yang menghasilkan nilai ROUGE-1 sebesar 0,78, ROUGE-2 sebesar 0,60, dan ROUGE-L sebesar 0,68. Penelitian rujukan berfokus pada perbandingan variasi *preprocessing* untuk menghasilkan ringkasan menggunakan metode BERT, sedangkan penelitian ini menggunakan kombinasi metode LDA sebagai pemodelan topik dan MMR sebagai metode peringkasan.

Tabel 2.1 Penelitian terkait

No	Peneliti	Judul	Metode	Hasil	Perbedaan Penelitian
1.	Nurina Savanti Widya Gotami, Indriati, dan Ratih Kartika Dewi	Peringkasan Teks Otomatis Secara Ekstraktif Pada Artikel Berita Kesehatan Berbahasa Indonesia Dengan Menggunakan Metode *Latent Semantic Analysis*	*Latent Semantic Analysis*	*Compression rate* 50% nilai rata-rata *accuracy*, *precision, recall* dan *f1-score*, secara berturut 0.668, 0.743, 0.700 dan 0.690, *compression rate* 40% sebesar 0.696, 0.605, 0.642, dan 0.663.	1. Menggunakan metode *Latent Semantic Analysis*. 2. Menggunakan data berita kesehatan dari kompas.com sebanyak 10 dokumen.
2	Gunawan, Fitria, Esther Irawati Setiawan, dan Kimiya Fujisawa	*Maximum Marginal Relevance and Vector Space Model for Summarizing Students' Final Project Abstracts*	*Maximum Marginal Relevance*	Rata-rata *precision* sebesar 88%, *recall* sebesar 61%, dan *f1-score* sebesar 70%	1. Menggunakan MMR tanpa pemodelan topik. 2. Objeknya 200 abstrak tugas akhir.

Lanjutan Tabel 2.1

No	Peneliti	Judul	Metode	Hasil	Perbedaan Penelitian
3.	Muhammad Audika Nugraha dan Lulu Chaerani Munggaran	Pemodelan Topik Berita pada Portal Berita Online Berbahasa Indonesia menggunakan *Latent Dirichlet Allocation* (LDA)	*Latent Dirichlet Allocation*	Hasil terbaik LDA dengan membentuk 5 topik dengan pengukuran cv menunjukkan hasil koheren sebesar 0.67.	1. Penelitian sebatas pemodelan topik tanpa dilanjutkan peringkasan. 2. Data dari portal berita detik.com sebanyak 68.537 artikel.
4.	Kalliath Abdul Rasheed Issam, Shivam Patel, Subalalitha C. N.	*Topic Modeling Based Extractive Text Summarization*	*Latent Dirichlet Allocation* dan *TextRank*	Hasil ringkasan yang di evaluasi dengan ROUGE yaitu ROUGE-1 27,08; ROUGE-2 6,89; dan ROUGE-L 25,43.	1. Metode peringkasan menggunakan *TextRank.* 2. Menggunakan *WikiHow datasets* yang berisi artikel bahasa Inggris.
5.	Halimah, Surya Agustian, dan Siti Ramadhan	Peringkasan Teks Otomatis (*Automated Text Summarization*) pada Artikel Berbahasa Indonesia Menggunakan Algoritma *Lexrank*	*Lexrank*	Tingkat kompresi 50% menghasilkan nilai rata-rata *f1-score* masing-masing sebesar 67,53%, 59,10%, dan 67,05% pada ROUGE-1, ROUGE-2, dan ROUGE-L.	1. Metode peringkasan menggunakan *Lexrank.* 2. Menggunakan 300 artikel berita dari internet.
6.	Muhammad Farhan Juna dan Mardhiya Hayaty	*The Observed Preprocessing Strategies for Doing Automatic Text Summarizing*	*Bidirectional Encoder Representations from Transformers* (BERT)	Nilai rata-rata terbaik ROUGE-1 sebesar 0,78, ROUGE-2 sebesar 0,60, dan ROUGE-L sebesar 0,68.	1. Berfokus pada perbandingan variasi *preprocessing* untuk menghasilkan ringkasan menggunakan metode BERT.
-	Usulan Penelitian	Sistem Peringkasan Teks Berita Berbahasa Indonesia Menggunakan *Latent Dirichlet Allocation* dan *Maximum Marginal Relevance*	*Latent Dirichlet Allocation* dan *Maximum Marginal Relevance*	Penelitian yang akan dilakukan	1. Menggunakan kombinasi metode *Latent Dirichlet Allocation* dan *Maximum Marginal Relevance* 2. Menggunakan *dataset* artikel *Indonesian Text Summarization (IndoSum)*

Tabel 2.1 menjelaskan penelitian sebelumnya yang telah menggunakan beberapa metode yaitu *Latent Semantic Analysis (LSA), Maximum Marginal Relevance (MMR), Lexrank, TextRank,* dan BERT untuk peringkasan teks. Terdapat juga penelitian tentang pemodelan topik menggunakan metode *Latent Dirichlet Allocation* sebagai penunjang sistem peringkasan teks dan beberapa penelitian tersebut objeknya yaitu tentang artikel berita. Keberhasilan dari metode LDA dalam pemodelan topik membuat peneliti tertarik mengkombinasikan metode LDA dengan metode MMR untuk menghasilkan sebuah ringkasan yang baik. Hasil dari peringkasan ini dievaluasi dengan ROUGE *(Recall-Oriented Underresearch for Gisting Evaluation)* untuk mengetahui performa sistem peringkasan ini. Terdapat dua pembaruan pada penelitian ini, pertama penggunaan *dataset* dari *Indonesian Text Summarization (IndoSum)* dan kedua menggunakan metode MMR untuk melakukan peringkasan teks bahasa Indonesia yang dikombinasikan dengan metode LDA sebagai pemodelan topik.

2.2 *Natural Language Processing*

Natural Language Processing (NLP) merupakan bidang kecerdasan buatan yang berfokus pada kinerja komputer dalam memahami dan menafsirkan bahasa manusia. NLP melibatkan analisis, pemahaman, dan menghasilkan bahasa manusia tertulis atau lisan. Teknik NLP digunakan dalam berbagai aplikasi seperti mesin terjemahan, analisis sentimen, *chatbot*, dan pengenalan suara. Perkembangan NLP ini dimulai dari penggunaan kartu pons (*punched cards*) yang digunakan untuk berkomunikasi dengan komputer sekitar 70 tahun yang lalu (Sivarethinamohan *et al*., 2021).

Sistem NLP yang umum digunakan terbagi dalam dua kategori. Pertama, sistem berbasis aturan (*rules-based* system) adalah algoritma yang telah ada sejak awal pemrosesan bahasa alami dan masih digunakan. Cabang NLP tersebut menggunakan aturan linguistik yang diterapkan secara terstruktur. Jenis yang kedua yaitu *machine learning-based system*, teknik statistik digunakan oleh sistem pembelajaran mesin untuk memproses model NLP. Sistem ini menggunakan data pelatihan yang telah disediakan sebelumnya untuk menjalankan tugas secara otomatis. Saat data baru ditambahkan, selama pemrosesan data algoritma akan disesuaikan.

2.2.1 Sistem Peringkasan Teks

Salah satu aplikasi *Natural Language Processing* (NLP) yang dapat mengekstraksi informasi penting dari teks sumber dan menghasilkan ringkasan adalah sistem peringkas teks. Peringkasan teks merupakan suatu proses penyajian kembali dokumen dalam format ringkas tanpa kehilangan informasi penting apa pun yang dikandungnya (Hernawan *et al*., 2022). Tujuan peringkasan teks otomatis adalah menggunakan teks yang lebih pendek dari dokumen asli untuk mewakili informasi utama dokumen (Mao *et al*., 2021).

Ada dua metode yang umum digunakan dalam proses peringkasan teks yaitu metode ekstraktif dan abstraktif. Teknik abstraktif akan menghasilkan kata baru yang tidak terdapat dalam teks aslinya kemudian menggabungkan dan menyusun kata-kata baru tersebut dengan kata aslinya untuk menyusun suatu kalimat baru, kalimat baru inilah yang akan menjadi hasil ringkasan. Teknik ekstraktif adalah

teknik yang mengambil setiap kata atau kalimat dari teks sumber sebagai ringkasan, jadi tidak ada kata atau kalimat yang berubah.

Jika dilihat dari inputnya, tipe peringkasan teks otomatis dapat menggunakan *single* dokumen atau *multi* dokumen. Peringkasan *single* dokumen merupakan proses menghasilkan ringkasan singkat yang menggambarkan informasi kunci dari dokumen tunggal. Meringkas teks yang masukannya terdiri dari beberapa dokumen dari satu atau lebih sumber yang berkaitan dengan topik dokumen dan memiliki gagasan utama yang sama bisa juga berbeda dikenal sebagai peringkasan teks multi-dokumen (Roul, 2021).

2.2.2 Teks *Preprocessing*

Preprocessing merupakan tahapan umum yang digunakan dalam membuat setiap sistem untuk *text processing* pada NLP. *Preprocessing* teks bertujuan untuk membersihkan, memformat, dan menyiapkan teks agar memenuhi persyaratan analisis atau pemodelan yang akan dilakukan. Teks *preprocessing* juga berfungsi untuk meminimalisir kesalahan dalam mengambil atribut agar dapat menghasilkan bentuk data yang lebih baik (Juna & Hayaty, 2023). *Case folding* atau mengubah huruf menjadi huruf kecil, memfilter dengan menghilangkan kata-kata *stopwords*, tokenisasi yaitu proses yang membagi sekelompok karakter teks menjadi satuan kata dan *stemming* yang mengubah kata-kata dalam dokumen menjadi bentuk paling dasar sesuai pedoman yang telah ditentukan adalah di antara prosedur yang dilakukan selama tahap *preprocessing* teks. Serangkaian proses tersebut akan mempengaruhi hasil analisis teks yang dilakukan pada sistem NLP.

2.2.3 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) (Nurkholis *et al*., 2022). *Term Frequency* (TF) adalah jumlah kemunculan suatu istilah dalam sebuah dokumen. Semakin sering sebuah istilah muncul dalam dokumen, semakin besar kemungkinan bahwa istilah tersebut penting bagi dokumen tersebut. Sedangkan ketika sebuah istilah muncul di banyak dokumen, kemungkinan istilah tersebut kurang penting untuk setiap dokumen. Sederhananya, pendekatan TF-IDF menghitung seberapa sering sebuah kata muncul pada dokumen dan seberapa penting kata tersebut dalam keseluruhan korpus. Nilai IDF dapat dihitung menggunakan persamaan 2.1 (Halimah *et al*., 2022) :

$$IDF_t = \log(\frac{N}{n_t}) + 1 \tag{2.1}$$

Keterangan:
N : Jumlah semua dokumen
n_t : Jumlah dokumen yang mengandung *term*

Bobot akhir TF-IDF kata dapat dihitung dengan mengalikan nilai TF dengan IDF sesuai persamaan 2.2:

$$W_{t,d} = TF_{t,d} * IDF_t \tag{2.2}$$

Keterangan:

$W_{t,d}$	:	Bobot dari *t* (*term*) dalam satu dokumen
$TF_{t,d}$	:	Frekuensi kemunculan *t* (*term*) dalam dokumen *d*
IDF_t	:	*Inverse Document Frequency*, dimana IDF didapatkan melalui persamaan 2.1

2.2.4 *Cosine Similarity*

Derajat kemiripan dua hal dapat ditentukan dengan menggunakan metode *cosine similarity*. Secara umum, ukuran kesamaan ruang vektor berfungsi sebagai dasar perhitungan metode *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 2.3 (Nurdiana *et al*., 2016) :

$$CosSim\ (d_i, q_i) = \frac{q_i\ .\ d_i}{|q_i||d_i|} = \frac{\sum_{j=1}^{t}(q_{ij}\ .\ d_{ij})}{\sqrt{\sum_{j=1}^{t}(q_{ij})^2\ .\ \sum_{j=1}^{t}(d_{ij})^2}} \qquad (2.3)$$

Keterangan:

d_i	: Vektor d_i yang akan dibandingkan kemiripannya
q_i	: Vektor q_i yang akan dibandingkan kemiripannya
$q_i\ .\ d_i$	: *Dot product* antara vektor d_i dan vektor q_i
$\|\ d_i\|$	: Panjang vektor d_i
$\|\ q_i\|$	: Panjang vektor q_i
$\|\ q_i\|\ \|\ d_i\|$	: *Cross product* antara $\|\ d_i\|$ dan $\|\ q_i\|$
q_{ij}	: Bobot istilah *j* pada dokumen *qi*
d_{ij}	: Bobot istilah *j* pada dokumen *di*

d_i dan q_i merupakan dua vektor kalimat yang ingin diukur tingkat kemiripannya. Nilai *cosine* diperoleh dari pembagian antara *dot product* dengan *cross product* vektor. Dalam konteks pemrosesan teks, vektor biasanya merepresentasikan dokumen dalam bentuk vektor ruang berdimensi tinggi, seperti model vektor kata *word embeddings* atau TF-IDF.

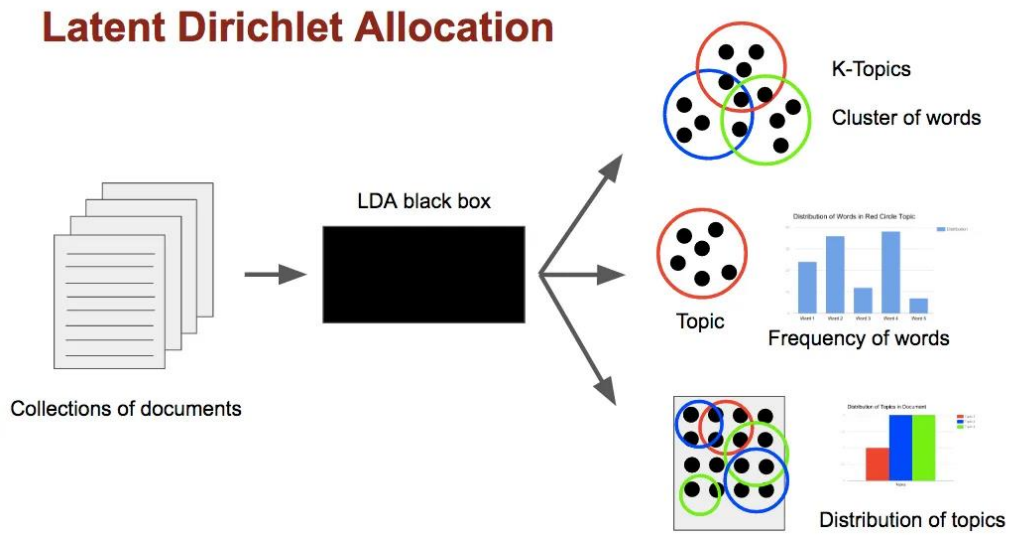
2.3 Berita

Berita adalah cara tercepat untuk mengetahui peristiwa, kejadian, fakta, atau pemikiran terkini yang dapat dipercaya dan dapat menarik perhatian publik terhadap informasi penting yang terdapat di media. Haris Sumadiria mendefinisikan berita merupakan sebuah laporan tercepat mengenai fakta atau ide terbaru yang benar, menarik, atau penting bagi sebagian besar khalayak (A.S. Haris Sumadiria, 2014). Berita memiliki ciri khas tertentu, seperti faktual, aktual, dan dipilih oleh wartawan untuk dimuat. Hal ini menunjukkan bahwa berita harus memenuhi standar kebenaran, kebaruan, dan kepentingan bagi khalayak. Berita dapat disebarluaskan melalui berbagai media, termasuk media online di internet dan media tradisional seperti radio, televisi, dan surat kabar. Memberi informasi kepada pembaca, pendengar, atau pemirsa tentang peristiwa yang terjadi di sekitar mereka atau dalam bidang tertentu merupakan tujuan utama dari berita.

2.4 *Latent Dirichlet Allocation (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* (Pinto & Chahed, 2014). LDA merupakan teknik yang paling banyak digunakan untuk pemodelan topik. Dengan asumsi bahwa kata dan teks merupakan campuran acak dari topik tersembunyi (*laten*), LDA memandang setiap topik sebagai sebaran kata yang dihasilkan oleh model probabilistik generatif (Oktaviana *et al*., 2022). Setiap topik mempunyai distribusi probabilitas kata yang terpisah. Model probabilistik generatif merupakan suatu jenis model statistik yang dirancang untuk memodelkan

distribusi probabilitas dari suatu dataset. Model ini dapat menghasilkan data yang serupa dengan data pelatihan dan dapat digunakan untuk generasi data baru (Liu *et al*., 2024). Gambar 2.1 merupakan alur kerja dari metode LDA untuk pemodelan topik.

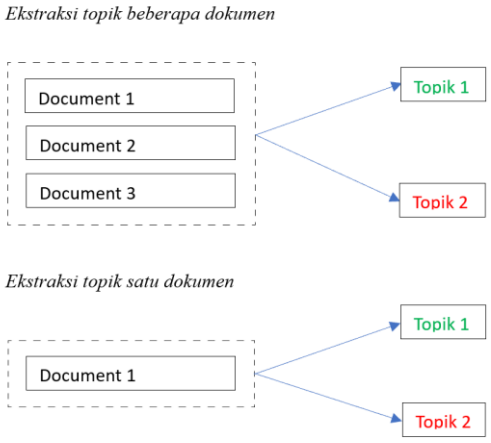


Gambar 2.1 Alur kerja metode LDA
sumber : Hidayatullah *et al.*, 2019

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*. Distribusi *dirichlet* merupakan distribusi probabilitas multivariat yang digunakan untuk memodelkan distribusi probabilitas pada sejumlah variabel acak yang jumlahnya tetap atau normalisasi. Langkah ketiga melakukan pengulangan setiap alur proses untuk setiap kata dalam korpus yang digunakan (Hidayatullah *et al*., 2019).

Corpus merupakan kumpulan M dokumen yang dilambangkan dengan **D** = {w_1,w_2,...,w_M}. Suatu dokumen adalah barisan N kata yang dilambangkan dengan **w** = (w_1,w_2,...,w_N), dimana w_n adalah kata ke-n dalam barisan tersebut. Kata adalah unit dasar yang merupakan bagian dari kosakata, diindeks oleh {1,...,V}. Pada pemodelan topik, *corpus* adalah kumpulan dokumen yang direpresentasikan sebagai *Document Term Matrix* (DTM) atau kadang disebut *document word matrix* (Rusdhi & Sari, 2022).

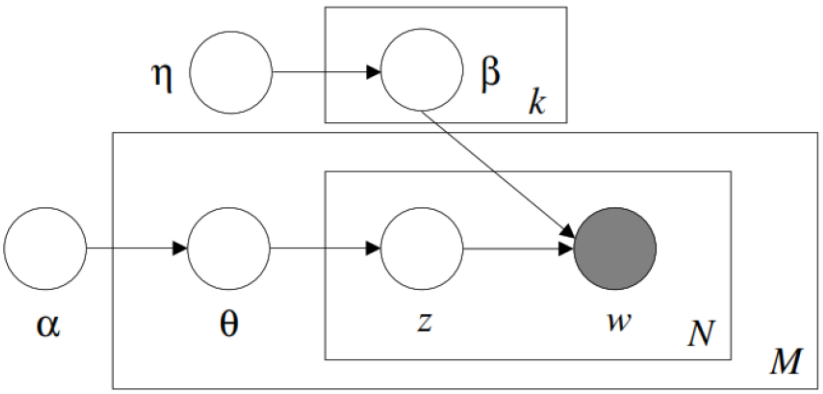
LDA dapat diterapkan untuk mengidentifikasi distribusi umum topik tersembunyi dalam kumpulan dokumen serta distribusi topik masing-masing dokumen (Sanjaya, 2021). Gambar 2.2 merupakan ilustrasi ekstraksi topik yang dilakukan oleh LDA.



Gambar 2.2 Ilustrasi ekstraksi topik LDA

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 menerapkan metode

inferensi variasional pada model yang diperluas dengan menyertakan *smoothing dirichlet* pada parameter multinomial (Blei et al., 2003). *Hyperparameter* yang disesuaikan (*smoothing)* dan inferensi variasional menghasilkan distribusi posterior yang lebih lembut, mencerminkan ketidakpastian yang lebih realistis dalam data. *Smooth* LDA memiliki *hyperparameter* η pada *dirichlet* yang menyatakan *smoothing* kata dalam topik, serta *hyperparameter* α menyatakan *smoothing* topik dalam dokumen. Gambar 2.3 merupakan bentuk dari arsitektur model *smooth* LDA dalam sebuah diagram.



Gambar 2.3 Arsitektur model smooth LDA
sumber : Blei et al., 2003

Keterangan:

α	:	Parameter distribusi topik per dokumen
η	:	Parameter distribusi kata per topik
θ	:	Distribusi probabilitas topik pada dokumen ke-d dengan parameter α
β	:	Distribusi probabilitas kata-kata pada topik ke-k dengan parameter η
Z	:	Topik yang dihasilkan oleh distribusi multinomial dengan parameter θ
W	:	Sampel kata dari distribusi multinomial dengan parameter β 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.

Sebuah dokumen akan diberikan dalam bentuk (w_1 , w_2, ..., w_n) dengan jumlah k topik yang diinginkan. Distribusi topik di dalam dokumen ditentukan menggunakan parameter α. Nilai α yang lebih besar pada suatu dokumen menunjukkan cakupan topik yang lebih luas yang disebutkan di dalamnya. Untuk menentukan distribusi kata dalam suatu topik, digunakan parameter η. Topik memiliki kata-kata yang lebih khusus ketika nilai η lebih kecil, yang menunjukkan lebih sedikit kata dalam topik tersebut. Sebaliknya, topik dengan nilai η yang lebih tinggi mempunyai lebih banyak kata di dalamnya. 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* (Syed & Spruit, 2017). θ merupakan distribusi topik yang tersembunyi pada setiap dokumen dan β merupakan probabilitas setiap kata yang diberi topik. Proses pengelompokan metode LDA merupakan distribusi gabungan atas variabel *laten* dan acak yang diamati (W, Z, β, θ), hal tersebut dapat diperoleh pada persamaan 2.4 (Syed & Spruit, 2017):

$$p(\beta_K, \theta_D, z_D, w_D \mid \alpha, \eta) = \prod_{k=1}^{K} p(\beta_k|\eta) \prod_{d=1}^{D} p(\theta_d|\alpha) \prod_{n=1}^{N} p\left(Z_{d,n}|\theta_d\right) p\left(w_{d,n}|z_{d,n}, \beta_{d,k}\right) \quad (2.4)$$

Pada persamaan tersebut ada 4 bagian :

1. $\prod_{k=1}^{K} p(\beta_k|\eta)$
2. $\prod_{d=1}^{D} p(\theta_d|\alpha)$
3. $p\left(Z_{d,n}\,|\,\theta_d\right)$
4. $p(w_{d,n}|z_{d,n},\beta_{d,k})$

Bagian 1 dan 2 adalah *dirichlet distribution* sedangkan bagian 3 dan 4 adalah *multinomial distribution*. Bagian 1 merupakan probabilitas dari suatu topik yang memuat kata tertentu. Bagian 2 menghitung probabilitas dari suatu dokumen terhadap topik tertentu. Bagian 3 menghitung probabilitas dari topik berdasarkan nilai distribusi topik dari dokumen. Bagian 4 menghitung probabilitas suatu kata untuk diasosiasikan ke dalam topik tertentu.

Kata-kata (w) dalam LDA merupakan variabel yang diamati, distribusi topik dan kata tersembunyi, serta α dan η adalah *hyperparameter*. Oleh karena itu, kita perlu menyimpulkan distribusi dan hyperparameternya. Secara umum, masalah ini sulit diselesaikan. Ada beberapa teknik umum untuk inferensi dan estimasi parameter salah satunya menggunakan *Gibbs Sampling*. *Gibbs Sampling* merupakan anggota dari *Markov Chain Monte Carlo* (MCMC) *framework* yang bekerja dengan melakukan pengambilan sampel dari distribusi gabungan ketika hanya distribusi topik dan kata bersyarat yang dapat dihitung secara efisien (Pinto & Chahed, 2014).

2.5 *Maximum Marginal Relevance* (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 (Arie, 2021). MMR meringkas dokumen dengan menentukan derajat kemiripan antar bagian teks. Metode MMR dapat mencegah redudansi dan dapat mengambil informasi yang relevan (Susanto *et al*., 2021). Metode ini mampu memilih dokumen yang paling relevan dengan sebuah kueri tertentu dengan menggabungkan dua faktor yaitu relevansi dan keragaman.

Faktor relevansi digunakan untuk mengukur tingkat kerelevansian dokumen tersebut dengan kueri, sedangkan faktor keragaman digunakan untuk mengukur tingkat perbedaan dengan dokumen-dokumen lain yang sudah dipilih. Dengan menggabungkan konsep relevansi dan keberagaman, MMR dapat menghasilkan ringkasan yang menggambarkan informasi penting dengan menghindari pengulangan yang tidak perlu. Sistem peringkasan teks menggunakan MMR dapat menghasilkan ringkasan yang tidak hanya menangkap informasi yang paling relevan tetapi juga memastikan representasi yang beragam dari teks (Yuliska & Syaliman, 2020).

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. Metode ini memastikan bahwa kalimat yang dipilih tidak hanya relevan dengan topik utama tetapi juga menambah informasi baru yang belum tercakup oleh kalimat-kalimat sebelumnya dalam ringkasan. Skor dari MMR dapat dihitung dengan persamaan 2.5 (Goldstein & Carbonell, 1998):

$$\text{MMR} = \underset{D_i \in R \setminus S}{\text{Argmax}} \left[\lambda * \text{Sim}_1(D_i, Q) - (1-\lambda) \underset{D_j \in S}{\text{Max}}(\text{Sim}_2(D_i, D_j))\right] \quad (2.5)$$

Keterangan:

λ	:	Parameter yang memengaruhi tingkat relevansi
D	:	Kalimat
D_i	:	Kalimat ke-i pada suatu dokumen yang belum terpilih menjadi ringkasan
D_j	:	Kalimat ke-j yang sudah terpilih menjadi ringkasan
Q	:	Kueri
$Sim_1(D_i, Q)$	:	Nilai *similarity* antara kalimat ke-i dengan kueri
$Sim_2(D_i, D_j)$	:	Nilai *similarity* antara kalimat ke-i dengan kalimat yang sudah terpilih menjadi ringkasan

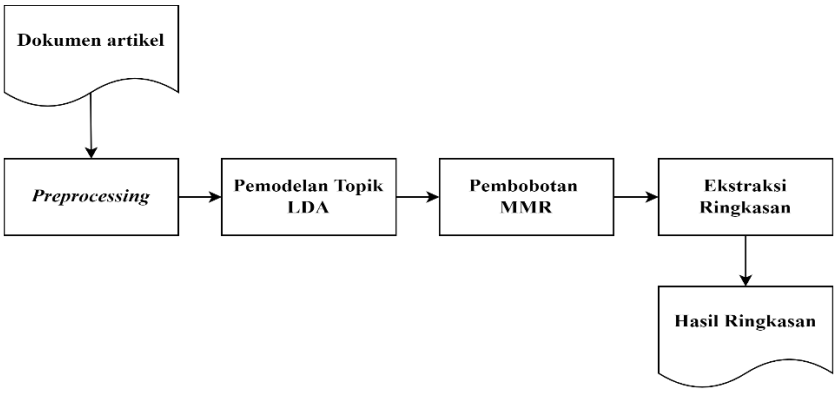
R adalah kumpulan kalimat dalam satu dokumen yang akan diekstrak menjadi ringkasan. *S* adalah kumpulan kalimat di *R* yang sudah terpilih sebagai ringkasan. *R\S* adalah selisih himpunan, yaitu himpunan kalimat di *R* yang belum terpilih menjadi ringkasan. D_i merupakan kalimat ke-i pada suatu dokumen 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 λ. Kisaran nilai parameter λ 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 $\lambda = 0$ (Mustaqhfiri *et al*., 2011). Oleh karena itu, untuk mendapatkan ringkasan yang masuk akal, nilai λ harus dioptimalkan antara 0 dan 1. Saat meringkas teks pendek seperti artikel, nilai parameter $\lambda = 0{,}7$ akan menghasilkan hasil ringkasan yang baik (Saraswati *et al*., 2018).

BAB III

DESAIN DAN IMPLEMENTASI

3.1 Desain Sistem

Tahapan ini meliputi alur dari sistem yang dimulai dari input data sampai menghasilkan output yang sesuai dengan tujuan penelitian. *Preprocessing* data merupakan tahapan awal dari penelitian ini 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. Gambar 3.1 merupakan desain sistem yang diimplementasikan pada penelitian ini.



Gambar 3.1 Desain sistem

3.2 Data Penelitian

Penelitian ini menggunakan *dataset* "*Indonesian Text Summarization (IndoSum)*" dengan *update* terbaru tahun 2022 (Kemal Kurniawan & Samuel Louvan). *IndoSum* merupakan kumpulan data besar untuk peringkasan teks dalam bahasa Indonesia yang dikumpulkan dari artikel berita online dan tersedia untuk umum. *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. Setiap artikel mempunyai judul, kategori, sumber (misalnya CNN Indonesia, Kumparan), URL artikel asli, dan ringkasan yang dibuat secara manual oleh total 2 penutur asli bahasa Indonesia (*native speaker)* (Kurniawan & Louvan, 2018).

Dataset ini pernah digunakan pada penelitian yang dilakukan oleh (Nyoman Purnama & Ni Nengah Widya Utami, 2023). Penelitian tersebut menggunakan 500 data paragraf dari *dataset IndoSum*. Pengujian pada penelitian tersebut menggunakan 3 skenario. Pengujian menggunakan ROUGE-1 memperoleh hasil terbaik pada skenario 2 yaitu menerapkan *stemming* tanpa penghapusan *stopwords* dengan nilai ROUGE-1 sebesar 0,17568.

Dataset Indosum menggunakan 5-*fold cross-validation* untuk membagi data menjadi 5 kumpulan data *training, development,* dan *testing.* Tiap *fold* (kumpulan data) terdiri dari *3 file .jsonl* yaitu *file train, dev, dan test.* Salah satu *file* dalam *dataset* yaitu *train.03.jsonl* dengan jumlah berita sebanyak 14290 dan 7 fitur. Salah satu fiturnya yaitu *paragraphs* berisi sebuah paragraf yang terdapat *list* kalimat dan sebuah kalimat diberikan *list* kata. Untuk fitur *summary* rata-rata kalimat ringkasan

manual adalah 3.48 kalimat setiap artikel. Penjelasan dari fitur-fitur pada *dataset* dapat dilihat pada Tabel 3.1.

Tabel 3.1 Deskripsi data

Fitur	Penjelasan
category	Kategori artikel tersebut
gold_labels	Label ekstraktif dari kalimat-kalimat dalam artikel tersebut
id	Pengenal unik untuk tiap artikel
paragraphs	Daftar paragraf dari artikel asli
source	Sumber artikel
source_url	Url dari artikel asli
summary	Ringkasan manual artikel yang disajikan dalam daftar kalimat

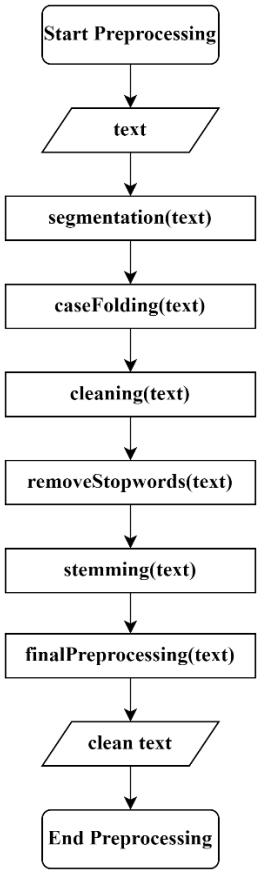
Dari 7 fitur pada Tabel 3.1 terdapat 2 fitur utama yang digunakan, yaitu *paragraphs* dan *summary*. Contoh struktur dari *dataset IndoSum* yang digunakan dapat dilihat pada Tabel 3.2.

Tabel 3.2 Contoh dataset

category	***gold_labels***	***id***	***paragraphs***	***Source***	***source_url***	***Summary***
tajuk utama	[[True, True], [True, True], [False, False, False], [False], [False], [False], [False], [False], [False, False], [False], [False], [False, False], [False, False], [False, False], …]	1490864409-ditjen-pajak-buka-lagi-ancaman-kore k-transaksi-kar	[[['Jakarta',',','CNN','Indonesia','-','-','Direktorat','Jenderal','Pajak','Kementerian','Keuangan','(','DJP','Kemenkeu',')','kembali','meminta','industri','perbankan','untuk','mempersiapkan','data','transaksi','kartu','kredit','nasabah','.'], ['Wacana','tersebut','sempat','tenggelam',',','ketika','pemerintah','memulai','program','amnesti','pajak','1','Juli','2016','lalu','.']],[['Sebelumnya','Kementerian','Keuangan','menunda','pemberlakuan','Peraturan','Menteri','Keuangan','(','PMK',')',…],…]	cnn indonesia	http://www.cnnindonesia.com/ekonomi/20170329131820-78-203397/ditjen-pajak-buka-lagi-ancaman-korek-transaksi-kartu-kredit/	[['Direktorat','Jenderal','Pajak','Kementerian','Keuangan','(','DJP','Kemenkeu',')','kembali','meminta','industri','perbankan','untuk','mempersiapkan','data','transaksi','kartu','kredit','nasabah','.'],['Wacana','tersebut','sempat','tenggelam',',','ketika','pemerintah','memulai','program','amnesti','pajak','1','Juli','2016','lalu','.'],['Hal','tersebut','sesuai','dengan','Peraturan','Menteri','Keuangan','(','PMK',')','Nomor','39/PMK.03','/','2016',',','yang','meminta','perbankan','nasional','menunda','pelaporan','transaksi','kartu','kredit','hingga','program','amnesti','pajak','berakhir','31','Maret','2017','ini','.']]

3.3 *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*. Terdapat 6 tahapan *preprocessing* yang akan diterapkan pada penelitian ini. Gambar 3.2 merupakan alur dari proses *preprocessing*.



Gambar 3.2 *Flowchart preprocessing*

3.3.1 *Segmentation*

Teks asli pada *dataset* setiap artikel dipisahkan berdasarkan paragraf. Tahap *segmentation* ini untuk memudahkan proses selanjutnya dengan menghiraukan pemisah paragraf sehingga setiap artikel langsung dipisah berdasarkan kalimat yang ada pada artikel tersebut. Tahap ini lah proses *segmentation* dilakukan untuk menggabung seluruh paragraf pada setiap artikel dan memisahkan setiap kalimat. Seluruh paragraf pada satu artikel akan disatukan ke dalam satu *list*. Hasilnya setiap artikel berisi *list* kalimat dan setiap kalimat berisi *list* kata atau *token*. Tabel 3.3 menunjukkan contoh dari proses *segmentation*.

Tabel 3.3 Contoh proses *segmentation*

Sebelum *segmentation*	[[['Jakarta', ',', 'CNN', 'Indonesia', '-', '-', 'Direktorat', 'Jenderal', 'Pajak', 'Kementerian', 'Keuangan', '(', 'DJP', 'Kemenkeu', ')', 'kembali', 'meminta', 'industri', 'perbankan', 'untuk', 'mempersiapkan', 'data', 'transaksi', 'kartu', 'kredit', 'nasabah', '.'], ['Wacana', 'tersebut', 'sempat', 'tenggelam', ',', 'ketika', 'pemerintah', 'memulai', 'program', 'amnesti', 'pajak', '1', 'Juli', '2016', 'lalu', '.']],…]
Setelah *segmentation*	[['Jakarta', ',', 'CNN', 'Indonesia', '-', '-', 'Direktorat', 'Jenderal', 'Pajak', 'Kementerian', 'Keuangan', '(', 'DJP', 'Kemenkeu', ')', 'kembali', 'meminta', 'industri', 'perbankan', 'untuk', 'mempersiapkan', 'data', 'transaksi', 'kartu', 'kredit', 'nasabah', '.'], ['Wacana', 'tersebut', 'sempat', 'tenggelam', ',', 'ketika', 'pemerintah', 'memulai', 'program', 'amnesti', 'pajak', '1', 'Juli', '2016', 'lalu', '.'],…]

3.3.2 *Case Folding*

Tahapan kedua *preprocessing* yaitu *case folding*, pada tahap ini seluruh kata yang menggunakan huruf kapital akan diubah menjadi huruf kecil semua. Proses ini bertujuan untuk menghindari perbedaan makna ketika suatu kata mengandung huruf kapital. Hasil *segmentation* yang berupa *list* kalimat dan tiap kalimat terdapat *list token* akan dirubah seluruh tokennya menjadi huruf kecil. Hasil dari proses ini yaitu *list token* yang sudah menjadi huruf kecil seluruhnya. Contoh proses *case folding* dapat dilihat pada Tabel 3.4.

Tabel 3.4 Contoh proses *case folding*

Sebelum *case folding*	[['Jakarta', ',', 'CNN', 'Indonesia', '-', '-', 'Direktorat', 'Jenderal', 'Pajak', 'Kementerian', 'Keuangan', '(', 'DJP', 'Kemenkeu', ')', 'kembali', 'meminta', 'industri', 'perbankan', 'untuk', 'mempersiapkan', 'data', 'transaksi', 'kartu', 'kredit', 'nasabah', '.'], ['Wacana', 'tersebut', 'sempat', 'tenggelam', ',', 'ketika', 'pemerintah', 'memulai', 'program', 'amnesti', 'pajak', '1', 'Juli', '2016', 'lalu', '.'],…]
Setelah *case folding*	[['jakarta', ',', 'cnn', 'indonesia', '-', '-', 'direktorat', 'jenderal', 'pajak', 'kementerian', 'keuangan', '(', 'djp', 'kemenkeu', ')', 'kembali', 'meminta', 'industri', 'perbankan', 'untuk', 'mempersiapkan', 'data', 'transaksi', 'kartu', 'kredit', 'nasabah', '.'], ['wacana', 'tersebut', 'sempat', 'tenggelam', ',', 'ketika', 'pemerintah', 'memulai', 'program', 'amnesti', 'pajak', '1', 'juli', '2016', 'lalu', '.'],…]

3.3.3 *Cleaning*

Tahap ketiga yaitu *cleaning* data, pada tahap ini dilakukan penghapusan inputan tertentu, yaitu karakter selain huruf seperti tanda baca dan angka. Tujuan pembersihan data adalah untuk menghilangkan data apa pun yang tidak diperlukan untuk pengoperasian sistem, sehingga hanya menyisakan data teks sebagai inputan. Tabel 3.5 menunjukkan contoh dari proses *cleaning data*.

Tabel 3.5 Contoh proses *cleaning data*

Sebelum *cleaning data*	[['jakarta', ',', 'cnn', 'indonesia', '-', '-', 'direktorat', 'jenderal', 'pajak', 'kementerian', 'keuangan', '(', 'djp', 'kemenkeu', ')', 'kembali', 'meminta', 'industri', 'perbankan', 'untuk', 'mempersiapkan', 'data', 'transaksi', 'kartu', 'kredit', 'nasabah', '.'], ['wacana', 'tersebut', 'sempat', 'tenggelam', ',', 'ketika', 'pemerintah', 'memulai', 'program', 'amnesti', 'pajak', '1', 'juli', '2016', 'lalu', '.'],…]
Setelah *cleaning data*	[['jakarta', '', 'cnn', 'indonesia', '', '', 'direktorat', 'jenderal', 'pajak', 'kementerian', 'keuangan', '', 'djp', 'kemenkeu', '', 'kembali', 'meminta', 'industri', 'perbankan', 'untuk', 'mempersiapkan', 'data', 'transaksi', 'kartu', 'kredit', 'nasabah', ''], ['wacana', 'tersebut', 'sempat', 'tenggelam', '', 'ketika', 'pemerintah', 'memulai', 'program', 'amnesti', 'pajak', '', 'juli', '', 'lalu', ''],…]

3.3.4 *Stopword Removal*

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. Tujuan utama

penghapusan *stopword* yaitu membuat teks lebih bersih dan meningkatkan kualitas analisis teks atau pemodelan yang dilakukan. Menghapus *stopword* bermanfaat agar fokus analisis dapat ditempatkan pada kata-kata yang lebih relevan atau bermakna. Contoh dari *stopword* yaitu konjungsi atau kata hubung. Setiap *token* akan dicek ada pada *list stopword* atau tidak, jika ada pada *list stopword* maka *token* tersebut akan dihapus dan tidak disertakan pada proses selanjutnya. Jika tidak ada pada *list stopword* maka akan lanjut ke proses berikutnya. Tabel 3.6 merupakan contoh proses dari *stopword removal*.

Tabel 3.6 Contoh proses *stopword removal*

Sebelum *stopword removal*	[['jakarta', '', 'cnn', 'indonesia', '', '', 'direktorat', 'jenderal', 'pajak', 'kementerian', 'keuangan', '', 'djp', 'kemenkeu', '', 'kembali', 'meminta', 'industri', 'perbankan', 'untuk', 'mempersiapkan', 'data', 'transaksi', 'kartu', 'kredit', 'nasabah', ''], ['wacana', 'tersebut', 'sempat', 'tenggelam', '', 'ketika', 'pemerintah', 'memulai', 'program', 'amnesti', 'pajak', '', 'juli', '', 'lalu', ''],…]
Setelah *stopword removal*	[['jakarta', '', '', 'indonesia', '', '', 'direktorat', 'jenderal', 'pajak', 'kementerian', 'keuangan', '', 'djp', 'kemenkeu', '', '', 'meminta', 'industri', 'perbankan', '', 'mempersiapkan', 'data', 'transaksi', 'kartu', 'kredit', 'nasabah', ''], ['wacana', 'tersebut', 'sempat', 'tenggelam', '', '', 'pemerintah', 'memulai', 'program', 'amnesti', 'pajak', '', 'juli', '', '', ''],…]

3.3.5 *Stemming*

Tahap selanjutnya yaitu *stemming*, pada tahap ini dilakukan pengubahan kata-kata menjadi kata dasarnya. Tujuan dari *stemming* adalah menghilangkan infleksi atau afiksasi sehingga kata-kata dengan akar yang sama dapat direpresentasikan dengan cara yang seragam (Rofiqi *et al*., 2019). Setiap *token* akan dicek pada kamus kata dasar, apabila *token* tidak terdapat pada kamus kata dasar maka *token* merupakan kata imbuhan sehingga akan dilakukan *stemming* dengan menghilangkan akhiran (-lah, -kah, -ku, -mu, -nya, -tah atau -pun). Kemudian menghilangkan kata imbuhan turunan (-i, -kan, -an), setelah itu menghapus

imbuhan awal (be-, di-, ke-, me, pe-, se- dan te-) (Mustikasari *et al*., 2021). Tabel 3.7 merupakan contoh dari proses *stemming* yang dilakukan.

Tabel 3.7 Contoh proses *stemming*

Sebelum *stemming*	[['jakarta', '', '', 'indonesia', '', '', 'direktorat', 'jenderal', 'pajak', 'kementerian', 'keuangan', '', 'djp', 'kemenkeu', '', '', 'meminta', 'industri', 'perbankan', '', 'mempersiapkan', 'data', 'transaksi', 'kartu', 'kredit', 'nasabah', ''], ['wacana', 'tersebut', 'sempat', 'tenggelam', '', '', 'pemerintah', 'memulai', 'program', 'amnesti', 'pajak', '', 'juli', '', '', ''],…]
Setelah *stemming*	[['jakarta', '', '', 'indonesia', '', '', 'direktorat', 'jenderal', 'pajak', 'menteri', 'uang', '', 'djp', 'kemenkeu', '', '', 'minta', 'industri', 'perban', '', 'siap', 'data', 'transaksi', 'kartu', 'kredit', 'nasabah', ''], ['wacana', '', 'sempat', 'tenggelam', '', '', 'perintah', 'mulai', 'program', 'amnesti', 'pajak', '', 'juli', '', '', ''],…]

3.3.6 *Final Preprocessing*

Setelah melalui proses *stemming* pada tahap terakhir yaitu *final preprocessing list token* akan dibentuk menjadi suatu kalimat. Pertama dilakukan penghapusan terhadap *string* dan *list* kosong sisa proses-proses sebelumnya. Selanjutnya membentuk *list token* menjadi suatu kalimat untuk melakukan proses pemodelan topik. Contoh hasil dari proses *final preprocessing* dapat dilihat pada Tabel 3.8.

Tabel 3.8 Contoh proses *final preprocessing*

Sebelum *final preprocessing*	[['jakarta', '', '', '', '', 'direktorat', 'jenderal', 'pajak', 'menteri', 'uang', '', 'djp', 'kemenkeu', '', '', 'minta', 'industri', 'perban', '', 'siap', 'data', 'transaksi', 'kartu', 'kredit', 'nasabah', ''], ['wacana', '', 'sempat', 'tenggelam', '', '', 'perintah', 'mulai', 'program', 'amnesti', 'pajak', '', 'juli', '', '', ''],…]
Setelah *final preprocessing*	['jakarta direktorat jenderal pajak menteri uang djp kemenkeu minta industri perban siap data transaksi kartu kredit nasabah ', 'wacana sempat tenggelam perintah mulai program amnesti pajak juli ', …]

3.4 Pemodelan Topik LDA

Proses ini mengidentifikasi topik yang terkandung di dalam setiap artikel dan kata kunci yang berkaitan dengan topik tersebut menggunakan metode LDA. Pertama, yaitu membuat representasi *corpus* dokumen, representasi *corpus* yang

akan digunakan oleh model LDA ini merupakan vektor TF-IDF. Kedua, membentuk *dictionary* yaitu menetapkan setiap kata ke id unik. Selanjutnya menentukan jumlah topik dan berapa kata kunci yang diekstrak sehingga nantinya akan dihasilkan distribusi topik dan bobot kata yang berkaitan dengan topik melalui proses pada metode LDA.

3.4.1 Pembentukan representasi *Corpus*

Corpus berisi sekumpulan kata yang ada pada setiap artikel, *corpus* yang akan digunakan untuk proses pemodelan topik menggunakan LDA direpresentasikan dengan pembobotan TF-IDF. Tabel 3.9 menunjukkan contoh satu dokumen artikel yang mempunyai tiga kalimat.

Tabel 3.9 Contoh kalimat pada satu artikel untuk LDA

Kalimat ke -	**Kalimat yang sudah di *preprocessing***
1	jakarta direktorat jenderal pajak menteri uang djp kemenkeu minta industri perban siap data transaksi kartu kredit nasabah
2	wacana sempat tenggelam perintah mulai program amnesti pajak juli
3	tunda laku hingga program amnesti pajak akhir maret

Contoh perhitungan TF-IDF dapat dilihat pada Tabel 3.10.

Tabel 3.10 Perhitungan TF-IDF pada LDA

no	***term***	***TF***			***N***	***DF***	***IDF***	***TFIDF = TF x IDF***		
		s1	***s2***	***s3***				***s1***	***s2***	***s3***
1	akhir	0	0	1	3	1	2.099	0	0	2.099
2	amnesti	0	1	1		2	1.406	0	1.405	1.405
3	data	1	0	0		1	2.099	2.099	0	0
4	direktorat	1	0	0		1	2.099	2.099	0	0
5	djp	1	0	0		1	2.099	2.099	0	0
…	…	…	…	…		…	…	…	…	…
26	tenggelam	0	1	0		1	2.099	0	2.099	0
27	transaksi	1	0	0		1	2.099	2.099	0	0
28	tunda	0	0	1		1	2.099	0	0	2.099
29	uang	1	0	0		1	2.099	2.099	0	0
30	wacana	0	1	0		1	2.099	0	2.099	0

Kolom *TF* merupakan *Term Frequency*, yaitu banyaknya suatu kata pada setiap kalimat. Misalkan kata "akhir" muncul satu kali pada *s3*, artinya pada kalimat ketiga mengandung kata "akhir" sebanyak satu. *N* merupakan jumlah seluruh kalimat yang ada pada satu artikel. Kolom *DF* merupakan *Document Frequency*, yaitu banyaknya kalimat yang mengandung kata tertentu. Contohnya kata "amnesti" muncul pada *s2* dan *s3* maka nilai *DF* yaitu 2. Nilai *IDF* merupakan hasil perhitungan menggunakan persamaan 2.1 yaitu menghitung *log* dari jumlah dokumen (N) dibagi nilai *DF* kemudian ditambah 1. Perhitungan *log* menggunakan logaritma natural, yaitu dapat ditulis *eloga*. Hasil akhir nilai pembobotan TF-IDF diperoleh dengan mengalikan nilai *TF* dengan *IDF*.

3.4.2 Pembentukan *Dictionary*

Pembentukan *dictionary* bertujuan untuk menetapkan setiap kata yang ada pada *vocabulary* ke id unik. Contoh *dictionary* dapat dilihat pada Tabel 3.11.

Tabel 3.11 Contoh *dictionary*

id	kata	id	kata
0	akhir	15	menteri
1	amnesti	16	minta
2	data	17	mulai
3	direktorat	18	nasabah
4	djp	19	pajak
5	hingga	20	perban
6	industri	21	perintah
7	jakarta	22	program
8	jenderal	23	sempat
9	juli	24	siap
10	kartu	25	tenggelam
11	kemenkeu	26	transaksi
12	kredit	27	tunda
13	laku	28	uang
14	maret	29	wacana

Bisa diamati pada Tabel 3.11 setiap kata mempunyai id uniknya masing-masing, seperti kata “akhir” mempunyai id 0. Dengan menggunakan *dictionary*, pencarian atau pencocokan *string* langsung dapat diganti menggunakan id dari kata-kata yang ingin dicari dikarenakan bilangan bulat lebih efisien dalam proses komputasi.

3.4.3 Penentuan Topik dan Kata Kunci

Langkah pertama LDA dengan algoritma *gibbs sampling* yaitu menentukan jumlah topik yang selanjutnya dilakukan penugasan masing-masing kata pada setiap topik. 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.

3.4.3.1 Penentuan Topik Secara Acak

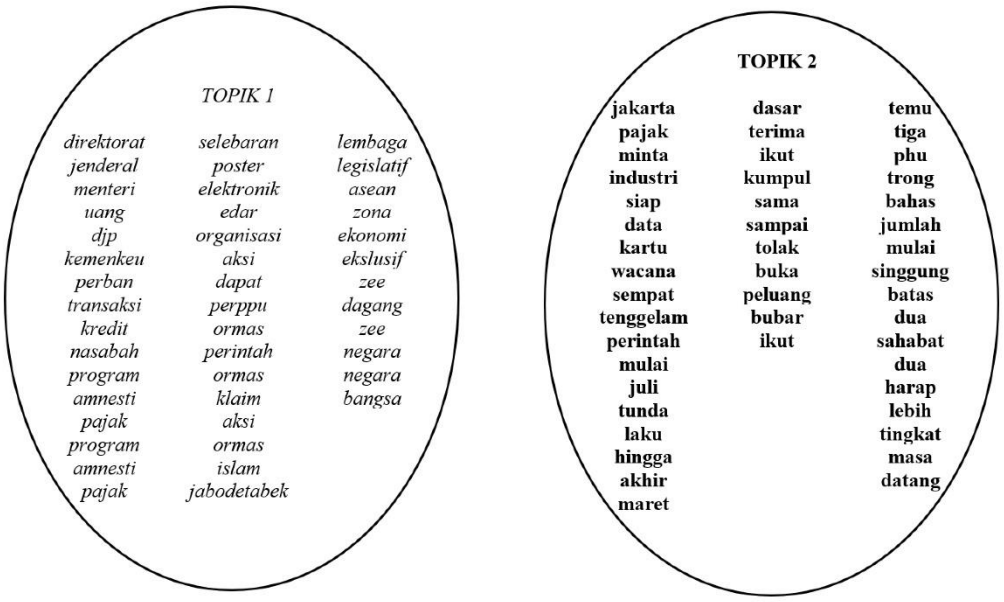
Vocabulary merupakan kata unik yang ada pada setiap dokumen. Jumlah topik yang diekstrak setiap artikel atau nilai K diatur samadengan 2. Berdasarkan penelitian oleh (Rusdhi & Sari, 2022) berapapun jumlah iterasinya, dua topik yang dihasilkan tidak tumpang tindih, menandakan keduanya berbeda atau tidak mengandung kata yang sama pada 10 kata yang mempunyai probabilitas tertinggi. Jumlah K=2 dapat memberikan hasil yang baik, di mana data memiliki struktur yang jelas terbagi menjadi dua tema utama. Tabel 3.12 merupakan contoh data seluruh kata pada dokumen artikel yang sudah diinisiasi topik secara acak dan juga *vocabulary* setiap dokumen. Angka yang ditulis pada setiap kata contohnya kata (jakarta^{0}) hanya untuk memudahkan perhitungan jumlah kata pada setiap artikel

maupun *vocabulary* yang digunakan. Angka 1 pada kata “jakarta” menunjukkan bahwa kata “jakarta” menempati urutan pertama pada daftar kata dalam artikel.

Tabel 3.12 Contoh data kata dan *vocabulary* setiap dokumen

Nomor Dokumen	***word***	***vocabulary***
1	**jakarta**[1], *direktorat*[2], *jenderal*[3], **pajak**[4], *menteri*[5], *uang*[6], *djp*[7], *kemenkeu*[8], **minta**[9], **industri**[10], *perban*[11], **siap**[12], **data**[13], *transaksi*[14], **kartu**[15], *kredit*[16], *nasabah*[17], **wacana**[18], **sempat**[19], **tenggelam**[20], **perintah**[21], **mulai**[22], *program*[23], *amnesti*[24], *pajak*[25], **juli**[26], **tunda**[27], **laku**[28], **hingga**[29], *program*[30], *amnesti*[31], *pajak*[32], **akhir**[33], **maret**[34]	akhir[1], amnesti[2], data[3], direktorat[4], djp[5], hingga[6], industri[7], jakarta[8], jenderal[9], juli[10], kartu[11], 'kemenkeu[12], kredit[13], laku[14], maret[15], menteri[16], minta[17], mulai[18], nasabah[19], pajak[20], perban[21], perintah[22], program[23], sempat[24], siap[25], tenggelam[26], transaksi[27], tunda[28], uang[29], wacana[30]
2	**dasar**[1], *selebaran*[2], *poster*[3], *elektronik*[4], *edar*[5], **terima**[6], *organisasi*[7], **ikut**[8], *aksi*[9], **kumpul**[10], **sama**[11], **sampai**[12], *dapat*[13], **tolak**[14], *perppu*[15], *ormas*[16], **buka**[17], **peluang**[18], *perintah*[19], **bubar**[20], *ormas*[21], *klaim*[22], *aksi*[23], **ikut**[24], *ormas*[25], *islam*[26], *jabodetabek*[27]	aksi[1], bubar[2], buka[3], dapat[4], dasar[5], edar[6], elektronik[7], ikut[8], islam[9], jabodetabek[10], klaim[11], kumpul[12], organisasi[13], ormas[14], peluang[15], perintah[16], perppu[17], poster[18], sama[19], sampai[20], selebaran[21], terima[22], tolak[23]
3	**temu**[1], **tiga**[2], *lembaga*[3], *legislatif*[4], **phu**[5], **trong**[6], **bahas**[7], **jumlah**[8], **mulai**[9], *asean*[10], *zona*[11], *ekonomi*[12], *ekslusif*[13], *zee*[14], *dagang*[15], **singgung**[16], **batas**[17], *zee*[18], **dua**[19], *negara*[20], **sahabat**[21], **dua**[22], *negara*[23], *bangsa*[24], **harap**[25], **lebih**[26], **tingkat**[27], **masa**[28], **datang**[29]	asean[1], bahas[2], bangsa[3], batas[4], dagang[5], datang[6], dua[7], ekonomi[8], ekslusif[9], harap[10], jumlah[11], lebih[12], legislatif[13], lembaga[14], masa[15], mulai[16], negara[17], phu[18], sahabat[19], singgung[20], temu[21], tiga[22], tingkat[23], trong[24], zee[25], zona[26]

Pada tahap ini, dilakukan inisiasi topik secara acak pada setiap kata dalam setiap dokumen. Kata-kata yang termasuk dalam *topik pertama* diberi tanda cetak *miring*, sedangkan kata-kata yang termasuk dalam **topik kedua** diberi tanda cetak **tebal**. Proses inisiasi ini bertujuan untuk memberikan penugasan awal kata-kata ke dalam topik-topik tertentu. Gambar 3.3 merupakan ilustrasi dari penugasan kata untuk setiap topik pada masing-masing dokumen.



Gambar 3.3 Ilustrasi penugasan kata setiap topik

Jumlah topik yang diekstrak sebanyak dua topik pada setiap artikel, maka nilai *Alpha* (α) yang mempengaruhi distribusi topik setiap dokumen dan nilai *Eta* (η) yang mempengaruhi distribusi kata pada suatu topik dapat diamati pada Tabel 3.13.

Tabel 3.13 Nilai parameter *alpha* dan *eta* setiap dokumen

Nomor Dokumen	**Jumlah kata pada *vocabulary* (V)**	**Nilai *Alpha* (α)**	**Nilai *Eta* (η)**
1	30	$\alpha = \frac{1}{K} = \frac{1}{2} = 0.5$	$\eta = \frac{1}{V} = \frac{1}{30} = 0.033$
2	23		$\eta = \frac{1}{V} = \frac{1}{23} = 0.043$
3	26		$\eta = \frac{1}{V} = \frac{1}{26} = 0.038$

3.4.3.2 Peluang Topik Pada Suatu Dokumen

Setelah penentuan topik secara acak, langkah selanjutnya adalah melakukan melakukan perhitungan peluang topik pada suatu dokumen $p\left(Z_{d,n}|\theta_d\right)$ menggunakan persamaan 3.1.

$$p\left(Z_{d,n}\,|\theta_d\right) = \frac{n_{dk} + \alpha}{N_d - 1 + K\alpha} \tag{3.1}$$

Keterangan:

n_{dk}	:	Jumlah kata dalam dokumen ke-d yang masuk topik ke-k
N_d	:	Jumlah seluruh kata di dalam dokumen ke-d
K	:	Jumlah topik yang diekstrak
α	:	Parameter yang menentukan distribusi topik dalam dokumen

Perhitungan dilakukan untuk setiap topik dan setiap dokumennya yang masing-masing kata sudah ditentukan secara acak masuk ke topik berapa. Distribusi topik setiap dokumen dapat dilihat pada Tabel 3.14.

Tabel 3.14 Distribusi topik setiap dokumen

Nomor Dokumen	**Jumlah Kata pada Topik ke-1 (*k1*)**	**Jumlah Kata pada Topik ke-2 (*k2*)**	**Jumlah seluruh Kata (N_d)**
1	16	18	34
2	16	11	27
3	12	17	29

Pada Tabel 3.14 dokumen ke-1 jumlah kata yang diinisiasi masuk topik ke-1 sebanyak 16 kata dan topik ke-2 sebanyak 18 kata, jumlah seluruh kata pada dokumen ke-1 sebanyak 34 kata. Dokumen ke-2 terdapat 16 kata pada topik ke-1 dan 11 kata pada topik ke-2 serta 27 untuk jumlah keseluruhan kata dalam dokumen. Pada dokumen ke-3 terdapat 12 kata pada topik ke-1, 17 kata pada topik ke-2, dan keseluruhan kata pada dokumen sebanyak 29 kata. Contoh perhitungan peluang topik pada suatu dokumen dapat dilihat sebagai berikut.

$$p\left(Z_{1,1}\,|\theta_1\right) = \frac{n_{11} + \alpha}{N_1 - 1 + K\alpha} = \frac{16 + 0.5}{34 - 1 + (2(0.5))} = 0.485$$

$$p\left(Z_{1,2}\,|\theta_1\right) = \frac{n_{12} + \alpha}{N_1 - 1 + K\alpha} = \frac{18 + 0.5}{34 - 1 + (2(0.5))} = 0.544$$

$$p\left(Z_{2,1}\middle|\theta_2\right) = \frac{n_{21} + \alpha}{N_2 - 1 + K\alpha} = \frac{16 + 0.5}{27 - 1 + (2(0.5))} = 0.611$$

$$p\left(Z_{2,2}\middle|\theta_2\right) = \frac{n_{22} + \alpha}{N_2 - 1 + K\alpha} = \frac{11 + 0.5}{27 - 1 + (2(0.5))} = 0.426$$

$$p\left(Z_{3,1}\middle|\theta_3\right) = \frac{n_{31} + \alpha}{N_3 - 1 + K\alpha} = \frac{12 + 0.5}{29 - 1 + (2(0.5))} = 0.431$$

$$p\left(Z_{3,2}\middle|\theta_3\right) = \frac{n_{32} + \alpha}{N_3 - 1 + K\alpha} = \frac{17 + 0.5}{29 - 1 + (2(0.5))} = 0.603$$

Peluang topik ke-1 pada dokumen ke-1 sebesar 0.485 dan peluang topik ke-2 pada dokumen ke-1 sebesar 0.544. Peluang topik ke-1 pada dokumen ke-2 sebesar 0.611 dan peluang topik ke-2 pada dokumen ke-2 sebesar 0.426. Peluang topik ke-1 pada dokumen ke-3 sebesar 0.431 dan peluang topik ke-2 pada dokumen ke-3 sebesar 0.603.

3.4.3.3 Peluang Setiap Kata Pada Suatu Topik

Langkah selanjutnya yaitu menghitung peluang setiap kata pada suatu topik $p(w_n|z_n, \beta)$. Peluang setiap kata pada suatu topik dapat dihitung menggunakan persamaan 3.2.

$$p(w_n|z_n, \beta) = \frac{w_{v,k} + \eta}{\sum_{v \in V} w_{v,k} + V\eta} \tag{3.2}$$

Keterangan:

$w_{v,k}$	:	Jumlah kosakata tertentu pada topik ke-k
V	:	Jumlah seluruh kosakata pada *vocabulary*
$\sum_{v \in V} w_{v,k}$	:	Jumlah seluruh kosakata pada topik ke-k
η	:	Parameter yang menentukan distribusi kata dalam topik

Contoh perhitungan peluang setiap kosakata dalam setiap topik dapat ditulis sebagai berikut.

Dokumen ke-1

$$p(w_1|z_1,\beta)_{(akhir)} = \frac{w_{1.1} + \eta}{\sum_{1 \in V} w_{1.1} + V\eta} = \frac{0 + 0.033}{16 + (30(0.033))} = 0.002$$

$$p(w_1|z_2,\beta)_{(akhir)} = \frac{w_{1.2} + \eta}{\sum_{1 \in V} w_{1.2} + V\eta} = \frac{1 + 0.033}{18 + (30(0.033))} = 0.054$$

$$\vdots$$

$$p(w_8|z_1,\beta)_{(jakarta)} = \frac{w_{8.1} + \eta}{\sum_{8 \in V} w_{8.1} + V\eta} = \frac{0 + 0.033}{16 + (30(0.033))} = 0.002$$

$$\vdots$$

$$p(w_{15}|z_2,\beta)_{(maret)} = \frac{w_{15.2} + \eta}{\sum_{15 \in V} w_{15.2} + V\eta} = \frac{1 + 0.033}{18 + (30(0.033))} = 0.054$$

$$\vdots$$

$$p(w_{30}|z_2,\beta)_{(wacana)} = \frac{w_{30.2} + \eta}{\sum_{30 \in V} w_{30.2} + V\eta} = \frac{1 + 0.033}{18 + (30(0.033))} = 0.054$$

Dokumen ke-2

$$p(w_1|z_1,\beta)_{(aksi)} = \frac{w_{1.1} + \eta}{\sum_{1 \in V} w_{1.1} + V\eta} = \frac{2 + 0.043}{16 + (23(0.043\ ))} = 0.12$$

$$\vdots$$

$$p(w_{23}|z_2,\beta)_{(tolak)} = \frac{w_{23.2} + \eta}{\sum_{23 \in V} w_{23.2} + V\eta} = \frac{1 + 0.043}{11 + (23(0.043\ ))} = 0.087$$

Dokumen ke-3

$$p(w_1|z_1,\beta)_{(asean)} = \frac{w_{1.1} + \eta}{\sum_{1 \in V} w_{1.1} + V\eta} = \frac{1 + 0.038}{12 + (26(0.038\ ))} = 0.08$$

$$\vdots$$

$$p(w_{26}|z_2,\beta)_{(zona)} = \frac{w_{26.2} + \eta}{\sum_{26 \in V} w_{26.2} + V\eta} = \frac{0 + 0.038}{17 + (26(0.038\ ))} = 0.002$$

Perhitungan probabilitas kata dalam sebuah topik $p(w_n|z_n,\beta)$ dilakukan untuk seluruh kosakata dan setiap topiknya, dimana terdapat 30 kosakata dalam *vocabulary* dokumen ke-1, 23 kosakata pada *vocabulary* dokumen ke-2, dan 26 kosakata pada *vocabulary* dokumen ke-3.

3.4.3.4 Pelabelan Ulang Kata

Agar pelabelan topik yang didapatkan konstan dilakukan iterasi untuk menentukan kata tersebut masuk ke topik berapa, perhitungan peluang setiap kata pada dokumen-d dan topik ke-k *p(w, z, θ)* dapat diperoleh dari *p(w, z, θ)* = $p(\beta_k|\eta)\, p(\theta_d|\alpha)$ contohnya sebagai berikut.

Dokumen ke-1

$$p\,(w_1, z_1, \theta_1)_{\text{(jakarta)}} = p(\beta_1|\eta)\, p(\theta_1|\alpha) = 0.002\ x\ 0.485 = 0.00094$$
$$\vdots$$
$$p\,(w_{33}, z_1, \theta_1)_{\text{(akhir)}} = p(\beta_1|\eta)\, p(\theta_1|\alpha) = 0.002\ x\ 0.485 = 0.00094$$

$$p\,(w_{33}, z_2, \theta_1)_{\text{(akhir)}} = p(\beta_2|\eta)\, p(\theta_1|\alpha) = 0.054\ x\ 0.544 = 0.02959$$
$$\vdots$$
$$p\,(w_{34}, z_2, \theta_1)_{\text{(maret)}} = p(\beta_k|\eta)p(\theta_1|\alpha) = 0.054 \text{ x } 0.544 = 0.02959$$

Dokumen ke-2

$$p\,(w_1, z_1, \theta_2)_{\text{(dasar)}} = p(\beta_1|\eta)\, p(\theta_1|\alpha) = 0.002\ x\ 0.589 = 0.001$$
$$\vdots$$
$$p\,(w_{27}, z_2, \theta_2)_{\text{(jabodetabek)}} = p(\beta_k|\eta)p(\theta_1|\alpha) = 0.003 \text{ x } 0.446 = 0.001$$

Dokumen ke-3

$$p\,(w_1, z_1, \theta_3)_{\text{(temu)}} = p(\beta_1|\eta)\, p(\theta_1|\alpha) = 0.003\ x\ 0.431 = 0.001$$
$$\vdots$$
$$p\,(w_{29}, z_2, \theta_3)_{\text{(datang)}} = p(\beta_k|\eta)p(\theta_1|\alpha) = 0.057 \text{ x } 0.603 = 0.034$$

Perhitungan di atas untuk mengetahui peluang setiap kata masuk ke dalam topik pertama atau kedua. Jumlah seluruh kata pada dokumen ke-1 yaitu 34, maka akan dilakukan perhitungan peluang sebanyak 34x pada setiap topik. Dari peluang ini *p (w, z, θ)* = $p(\beta_k|\eta)\, p(\theta_d|\alpha)$ menjadi acuan suatu kata masuk ke dalam topik yang mana. Contohnya untuk kata ke-33 pada dokumen ke-1 yaitu “akhir”

memperoleh peluang (0.00094) pada topik pertama dan (0.029) pada topik kedua. Berdasarkan peluang tersebut maka kata ke-33 masuk kedalam topik kedua karena memiliki peluang yang lebih tinggi.

Tabel 3.15 Contoh penentuan kata kunci LDA

Nomor Dokumen	Nomor Topik	Kata yang berkaitan dengan topik
1	1	0.12*”amnesti” + 0.12*”pajak” + 0.12*”program” + 0.061*”direktorat” + 0.061*”djp” + 0.061*”jenderal” + 0.061*”kemenkeu” + 0.061*”kredit” + 0.061*”menteri” + 0.061*”nasabah”
	2	0.054*”akhir” + 0.054*”data” + 0.054*”hingga” + 0.054*”industri” + 0.054*”jakarta” + 0.054*”juli” + 0.054*”kartu” + 0.054*”laku” + 0.054*”maret” + 0.054*”minta”
2	1	0.179*”ormas” + 0.120*”aksi” + 0.061*”dapat” + 0.061*”edar” + 0.061*”elektronik” + 0.061*”islam” + 0.061*”jabodetabek” + 0.061*”klaim” + 0.061*”organisasi” + 0.061*”perintah”
	2	0.170*”ikut” + 0.087*”bubar” + 0.087*”buka” + 0.087*”dasar” + 0.087*”kumpul” + 0.087*”peluang” + 0.087*”sama” + 0.087*”sampai” + 0.087*”terima” + 0.087*”tolak”
3	1	0.157*”negara” + 0.157*”zee” + 0.08*”asean” + 0.08*”bangsa” + 0.08*”dagang” + 0.08*”ekonomi” + 0.08*”eksklusif” + 0.08*”legislatif” + 0.08*”lembaga” + 0.08*”zona”
	2	0.113*”dua” + 0.058*”bahas” + 0.058*”batas” + 0.058*”datang” + 0.058*”harap” + 0.058*”jumlah” + 0.058*”lebih” + 0.058*”masa” + 0.058*”mulai” + 0.058*”phu”

Setelah dilakukan pelabelan ulang topik pada setiap kata, maka distribusi kata setiap topik akan berubah sesuai dengan banyak iterasi yang dilakukan. Pada Tabel 3.15 dihasilkan dua topik dan juga sepuluh kata kunci dengan probabilitas tertinggi yang berkaitan dengan topik tersebut. Bobot tersebut menunjukkan seberapa penting atau berkontribusi kata tersebut terhadap topik tertentu. Pada dokumen pertama topik pertama, kata “amnesti” memiliki peluang 0.12 dan dijumlahkan dengan kata “pajak” dengan peluang 0.12, sampai dengan kata ke-10 pada topik pertama. Pada proses ini dihasilkan 1 topik oleh LDA pada setiap dokumen untuk diambil kata kuncinya sebagai kueri pada peringkasan MMR.

3.5 Penerapan MMR

Pada tahap ini setiap kalimat pada artikel akan mendapatkan skor MMR masing-masing, setelah itu dipilih beberapa kalimat yang memiliki skor MMR tertinggi untuk diambil sebagai ringkasan. Kata kunci dari topik yang dihasilkan proses pemodelan topik LDA akan digunakan sebagai kueri MMR. Seluruh kalimat digunakan sebagai parameter kalimat non-kandidat ringkasan pada metode MMR. Menghitung skor MMR perlu mengetahui nilai *similarity* antara kalimat kandidat dengan kueri dan nilai *similarity* antara kalimat kandidat dengan kalimat non-kandidat. Proses pertama dalam MMR yaitu pembobotan kata menggunakan TF-IDF dan juga menghitung kemiripan antar kalimat menggunakan *cosine similarity*.

3.5.1 Pembobotan Kata

Dalam proses ini akan diketahui tingkat pentingnya suatu kata pada kalimat. Pembobotan akan menggunakan adalah *term frequency–inverse document frequency* (TF-IDF). Pada tahap ini akan terbentuk sebuah vektor yang mewakili tiap kalimat yang akan dijadikan ringkasan. Tabel 3.16 merupakan contoh hasil perhitungan TF-IDF.

Tabel 3.16 Contoh pembobotan TF-IDF

No	***term***	***TF-IDF***								
		Q	***s1***	***s2***	***s3***	***s4***	***s5***	***s6***	***s7***	***s8***
1	akhir	0.000	0.000	0.000	0.000	2.609	0.000	0.000	0.000	0.000
2	amnesti	0.000	0.000	2.204	0.000	2.204	0.000	0.000	0.000	0.000
3	atur	0.000	0.000	0.000	2.609	0.000	0.000	0.000	0.000	0.000
4	bank	1.916	0.000	0.000	0.000	0.000	0.000	0.000	1.916	3.833
5	data	1.693	1.693	0.000	0.000	0.000	1.693	0.000	0.000	5.079
6	direktorat	0.000	2.609	0.000	0.000	0.000	0.000	0.000	0.000	0.000
7	direktur	0.000	0.000	0.000	0.000	0.000	0.000	2.204	2.204	0.000

Lanjutan Tabel 3.16

No	*term*	*TF-IDF*								
		Q	***s1***	***s2***	***s3***	***s4***	***s5***	***s6***	***s7***	***s8***
8	ditandatangani	0.000	0.000	0.000	0.000	0.000	0.000	2.609	0.000	0.000
9	djp	1.693	1.693	0.000	1.693	0.000	0.000	1.693	0.000	0.000
10	format	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	2.609
…	…	…		…	…					
49	teknologi	0.000	0.000	0.000	0.000	0.000	0.000	2.609	0.000	0.000
50	tenggelam	0.000	0.000	2.609	0.000	0.000	0.000	0.000	0.000	0.000
51	transaksi	0.000	2.204	0.000	2.204	0.000	0.000	0.000	0.000	0.000
52	tuang	0.000	0.000	0.000	0.000	0.000	2.609	0.000	0.000	0.000
53	tuju	0.000	0.000	0.000	0.000	0.000	0.000	0.000	2.609	0.000
54	tunda	0.000	0.000	0.000	2.204	2.204	0.000	0.000	0.000	0.000
55	tutur	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	2.609
56	uang	1.693	1.693	0.000	3.386	0.000	1.693	0.000	0.000	0.000
57	utama	0.000	0.000	0.000	0.000	0.000	0.000	0.000	2.609	0.000
58	wacana	0.000	0.000	2.609	0.000	0.000	0.000	0.000	0.000	0.000

Pada tabel 3.16, *Q* merupakan *query* yaitu kata kunci dari topik yang telah didapatkan dari pemodelan topik LDA pada tahap sebelumnya. Sedangkan *d1* sampai *d8* merupakan kalimat kandidat ringkasan. Kolom *TF* merupakan banyaknya suatu kata pada tiap dokumen. Kolom *N* merupakan jumlah seluruh dokumen dan *DF* adalah banyaknya dokumen yang mengandung kata tertentu. Kolom *IDF* dapat dihitung menggunakan persamaan 2.1 yaitu menghitung log dari jumlah dokumen (*N*) dibagi nilai *DF* kemudian ditambah 1. Nilai akhir pembobotan diperoleh dengan persamaan 2.2 yaitu mengalikan antara nilai *TF* dan *IDF*.

3.5.2 *Cosine Similarity*

Pada tahap ini akan menghitung nilai kesamaan antara dua dokumen yang masing-masing dokumen dinyatakan dalam vektor. Setelah bobot kata diperoleh, selanjutnya yaitu mencari nilai *cosine similarity* dari dokumen yang ada. Perhitungan *cosine similarity* dalam MMR dibagi menjadi 2:

1. Perhitungan relevansi antara dokumen dan *query* (kata kunci topik)

   Menghitung *cosinus* sudut dari dua vektor, yaitu bobot TF-IDF dari setiap dokumen atau kalimat dengan bobot TF-IDF dari *query*.

2. Perhitungan *similarity* antara dokumen kandidat dan non-kandidat

   Menghitung *cosinus* sudut vektor bobot TF-IDF suatu kalimat dengan bobot TF-IDF kalimat lainnya.

Proses menghitung *cosine similarity* dibagi menjadi beberapa tahap, tahap pertama yaitu menghitung perkalian skalar (*dot product*) antara vektor *s1* dengan vektor *Q* serta perkalian vektor *s1* dengan vektor *s2* sampai *s8* kemudian dicari totalnya. Contoh perhitungan dapat dilihat pada Tabel 3.17.

Tabel 3.17 Perkalian skalar dua vektor

	Hasil Perkalian Skalar							
	s1 . Q	***s1 . s2***	***s1.s3***	***s1.s4***	***s1.s5***	***s1.s6***	***s1.s7***	***s1.s8***
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	2.866	0	0	0	2.866	0	0	8.600
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	2.866	0	2.866	0	0	2.866	0	0
	0	0	0	0	0	0	0	0
	…	…	…	…				
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	4.857	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	2.866	0	5.733	0	2.866	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
Total	18.0149	2.8668	36.2595	2.8668	15.9535	2.8667	3.6811	30.6692

Langkah selanjutnya yaitu menghitung panjang setiap dokumen, termasuk *d1*. Dengan cara mengkuadratkan bobot setiap *term* dalam setiap dokumen, kemudian menjumlahkan nilai kuadrat tersebut dan menghitung akarnya. Tabel 3.18 merupakan contoh perhitungan kuadrat vektor.

Tabel 3.18 Perhitungan kuadrat vektor

Hasil Kuadrat Vektor								
Q^2	$s1^2$	$s2^2$	$s3^2$	$s4^2$	$s5^2$	$s6^2$	$s7^2$	$s8^2$
0.000	0.000	0.000	0.000	6.809	0.000	0.000	0.000	0.000
0.000	0.000	4.857	0.000	4.857	0.000	0.000	0.000	0.000
0.000	0.000	0.000	6.809	0.000	0.000	0.000	0.000	0.000
3.672	0.000	0.000	0.000	0.000	0.000	0.000	3.672	14.689
2.867	2.867	0.000	0.000	0.000	2.867	0.000	0.000	25.801
0.000	6.809	0.000	0.000	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	4.857	4.857	0.000
0.000	0.000	0.000	0.000	0.000	0.000	6.809	0.000	0.000
…	…	…	…	…	…	…	…	…
0.000	0.000	0.000	0.000	0.000	0.000	6.809	0.000	0.000
0.000	0.000	6.809	0.000	0.000	0.000	0.000	0.000	0.000
0.000	4.857	0.000	4.857	0.000	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	6.809	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	6.809	0.000
0.000	0.000	0.000	4.857	4.857	0.000	0.000	0.000	0.000
0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	6.809
2.867	2.867	0.000	11.467	0.000	2.867	0.000	0.000	0.000

Setelah bobot setiap *term* dikuadratkan selanjutnya menghitung jumlah nilai kuadrat tersebut dan menghitung akarnya. Contoh hasil perhitungan dapat dilihat pada Tabel 3.19.

Tabel 3.19 Penjumlahan dan hasil akar vektor

	Q	***s1***	***s2***	***s3***	***s4***	***s5***	***s6***	***s7***	***s8***
Hasil penjumlahan kuadrat vektor	35.31	73.98	53.44	109.96	40.77	44.91	42.72	39.22	199.83
Akar dari jumlah kuadrat vektor	5.941	8.601	7.310	10.486	6.385	6.701	6.536	6.262	14.136

Proses terakhir yaitu menerapkan rumus *cosine* yang perhitungannya dapat dilihat pada Tabel 3.20.

Tabel 3.20 Proses *cosine similarity*

	Perhitungan rumus *cosine*	**Nilai *cosine similarity***
Sim1(s1,Q)	18.0149 / (8.6010 * 5.9418)	0.3525
Sim2(s1,s2)	2.86675 / (8.6010 * 7.31)	0.0456
…	…	…
Sim2(s1,s8)	30.6692 / (8.6010 * 14.1363)	0.2522

Contoh hasil *similarity* antara *s1* dengan *Q*, hasil penjumlahan perkalian skalar antara vektor *s1* dan vektor *Q* yaitu 18.0149 dibagi dengan akar vektor *s1* yaitu 8.6010 dikali dengan akar vektor *Q* yaitu 5.9418, maka hasil akhir nilai *cosine similarity* antara *s1* dengan *Q* yaitu 0.3525. Tabel 3.21 menunjukkan bobot kemiripan antara kueri dengan kalimat dalam dokumen.

Tabel 3.21 Bobot *query relevance*

	s1	***s2***	***s3***	***s4***	***s5***	***s6***	***s7***	***s8***
Q	0.3525	0.066	0.2891	0.0756	0.3084	0.0738	0.1976	0.3651

Tabel 3.22 menunjukkan bobot kemiripan antara kalimat dengan kalimat lainnya dalam dokumen.

Tabel 3.22 Bobot *similarity* antar kalimat

	s1	***s2***	***s3***	***s4***	***s5***	***s6***	***s7***	***s8***
s1	1	0.0456	0.402	0.0522	0.2768	0.051	0.0683	0.2522
s2	0.0456	1	0	0.2696	0	0	0	0
s3	0.402	0	1	0.1451	0.2847	0.0418	0.0561	0.1241
s4	0.0522	0.2696	0.1451	1	0	0.1164	0	0
s5	0.2768	0	0.2847	0	1	0	0.1752	0.3237
s6	0.051	0	0.0418	0.1164	0	1	0.1187	0.1051
s7	0.0683	0	0.0561	0	0.1752	0.1187	1	0.5103
s8	0.2522	0	0.1241	0	0.3237	0.1051	0.5103	1

3.5.3 MMR *Score*

Prinsip perhitungan metode MMR adalah mengambil kalimat dengan nilai tertinggi dari setiap perhitungan iterasi. Iterasi akan berhenti, ketika mencapai batas ringkasan yang diinginkan. Iterasi pertama memilih kalimat berdasarkan relevansi yaitu $MMR(s_i) = \lambda \cdot Sim_1(s_i, q)$ karena belum ada dokumen yang telah dipilih. Contoh perhitungan dengan nilai *lambda* 0.7 sebagai berikut.

$$MMR(s_1) = 0.7 \cdot 0.3525 = 0.2467$$

$$MMR(s_2) = 0.7 \cdot 0.066 = 0.0462$$

$$MMR(s_3) = 0.7 \cdot 0.2891 = 0.2024$$

$$MMR(s_4) = 0.7 \cdot 0.0756 = 0.0529$$

$$MMR(s_5) = 0.7 \cdot 0.3084 = 0.2159$$

$$MMR(s_6) = 0.7 \cdot 0.0738 = 0.0517$$

$$MMR(s_7) = 0.7 \cdot 0.1976 = 0.1383$$

$$MMR(s_8) = 0.7 \cdot 0.3651 = 0.2556$$

Menggunakan *argmax*, dipilih dokumen *s8* karena memiliki nilai MMR tertinggi yaitu 0.2556. Iterasi ke-2 nilai kebaruan akan diperhitungkan, maka $MMR(s_i) = \lambda \cdot Sim_1(s_i, q) - (1-\lambda) \max Sim_2(s_i, s_j)$. Pada iterasi kedua s_j yaitu *s8* karena *s8* sudah terpilih sebagai ringkasan pada iterasi sebelumnya. Berikut contoh perhitungan pada iterasi kedua.

$$MMR(s_1) = 0.7 \cdot 0.3525 - (1 - 0.7) \cdot 0.2522 = 0.1710$$

$$MMR(s_2) = 0.7 \cdot 0.066 - (1 - 0.7) \cdot 0 = 0.0462$$

$$MMR(s_3) = 0.7 \cdot 0.2891 - (1 - 0.7) \cdot 0.1241 = 0.1651$$

$$MMR(s_4) = 0.7 \cdot 0.0756 - (1 - 0.7) \cdot 0 = 0.0529$$

$$MMR(s_5) = 0.7 \cdot 0.3084 - (1 - 0.7) \cdot 0.3237 = 0.1187$$

$$MMR(s_6) = 0.7 \cdot 0.0738 - (1 - 0.7) \cdot 0.1051 = 0.0201$$

$$MMR(s_7) = 0.7 \cdot 0.1976 - (1 - 0.7) \cdot 0.5103 = -0.0147$$

Tabel 3.23 merupakan contoh iterasi MMR dengan ringkasan kalimat yang akan diambil sebanyak 3 kalimat.

Tabel 3.23 Contoh iterasi MMR

	s1	***s2***	***s3***	***s4***	***s5***	***s6***	***s7***	***s8***
iterasi1	0.2467	0.0462	0.2024	0.0529	0.2159	0.0517	0.1383	0.2556
iterasi2	0.1710	0.0462	0.1651	0.0529	0.1187	0.0201	-0.0147	-
iterasi3	-	0.0325	0.0817	0.0372	0.1187	0.0201	-0.0147	-

Hasil perhitungan pada iterasi ke-1 memperoleh nilai *maximum* MMR sebesar 0.2556 pada *s8* atau pada kalimat 8. Oleh karena itu, kalimat ke-8 akan dipilih sebagai ringkasan dan tidak diikut sertakan mencari skor MMRnya pada iterasi ke-2. Dari hasil perhitungan pada iterasi ke-2, diperoleh nilai maximum MMR sebesar 0.1710 pada *s1* atau pada kalimat ke-1. Oleh karena itu, kalimat ke-1 akan dipilih sebagai ringkasan setelah kalimat ke-8. Pada iterasi ke-3 nilai maksimum didapatkan oleh *s5* atau kalimat ke-5 dengan nilai 0.1187. Dikarenakan hanya ingin mengambil tiga kalimat teratas maka iterasi akan berhenti pada iterasi ke-3.

3.6 Ekstraksi Ringkasan

Tahap terakhir yaitu ekstraksi hasil ringkasan, Setiap kalimat sudah mempunyai nilai MMR masing-masing berdasarkan hasil iterasi. Proses ekstraksi kalimat menjadi ringkasan yaitu memilih kalimat dengan urutan skor MMR tertinggi. Berdasarkan hasil iterasi pada Tabel 3.23 maka urutan kalimat yang dijadikan ringkasan yaitu kalimat ke-8, kalimat ke-1, dan terakhir kalimat ke-5. Contoh hasil ekstraksi ringkasan dapat dilihat pada Tabel 3.24.

Tabel 3.24 Contoh hasil ekstraksi ringkasan

Nomor Kalimat	Kalimat Ringkasan Terpilih	Skor MMR max
s8	Dengan ini kami meminta kepada Bank / Lembaga Penyelenggara Kartu Kredit untuk mempersiapkan data kartu kredit sesuai dengan format data yang telah disepakati dalam Kamus Data dan Informasi Kartu Kredit dari Bank / Lembaga Penyelenggara Kartu Kredit	0.2556
s1	Jakarta , CNN Indonesia - - Direktorat Jenderal Pajak Kementerian Keuangan (DJP Kemenkeu) kembali meminta industri perbankan untuk mempersiapkan data transaksi kartu kredit nasabah.	0.1710
s5	Sementara , instruksi untuk menyiapkan kembali data kartu kredit tertuang dalam Surat Kementerian Keuangan Nomor S - 119/PJ .	0.1187

3.7 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 (Zamzam *et al*., 2020). 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. ROUGE-1 membandingkan *unigrams* sedangkan pada ROUGE-2 membandingkan banyaknya *bigrams*. Contohnya:

Unigrams : (saya),(suka),(bermain),(sepakbola)

Bigrams : (saya suka),(suka bermain),(bermain sepakbola)

ROUGE-N merupakan pengukuran yang berhubungan dengan *recall* karena penyebut persamaannya adalah jumlah total dari *n-gram* yang muncul pada sisi ringkasan referensi (Lin, 2004). Nilai *recall*, *precision*, dan *f-score* dapat dihitung melalui persamaan berikut:

$$Recall \quad = \frac{\sum_{s \in sys} \sum_{gramN \in s} Count_{match}(gram_N)}{\sum_{s \in ref} \sum_{gramN \in s} Count(gram_N)} \tag{3.3}$$

Nilai *recall* dihitung dengan persamaan 3.3 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.

$$Precision = \frac{\sum_{s \in sys} \sum_{gramN \in s} Count_{match}(gram_N)}{\sum_{s \in sys} \sum_{gramN \in s} Count(gram_N)} \tag{3.4}$$

Nilai *precision* dihitung dengan persamaan 3.4 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.

$$F\text{-}Score \quad = \frac{(1+\beta^2)(Precision * Recall)}{(\beta^2 * Precision + Recall)} \tag{3.5}$$

Variabel β 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.

Kalimat ringkasan sistem :

"Menunda pemberlakuan Peraturan Menteri Keuangan"

Kalimat ringkasan referensi :

"Hal tersebut sesuai dengan Peraturan Menteri Keuangan"

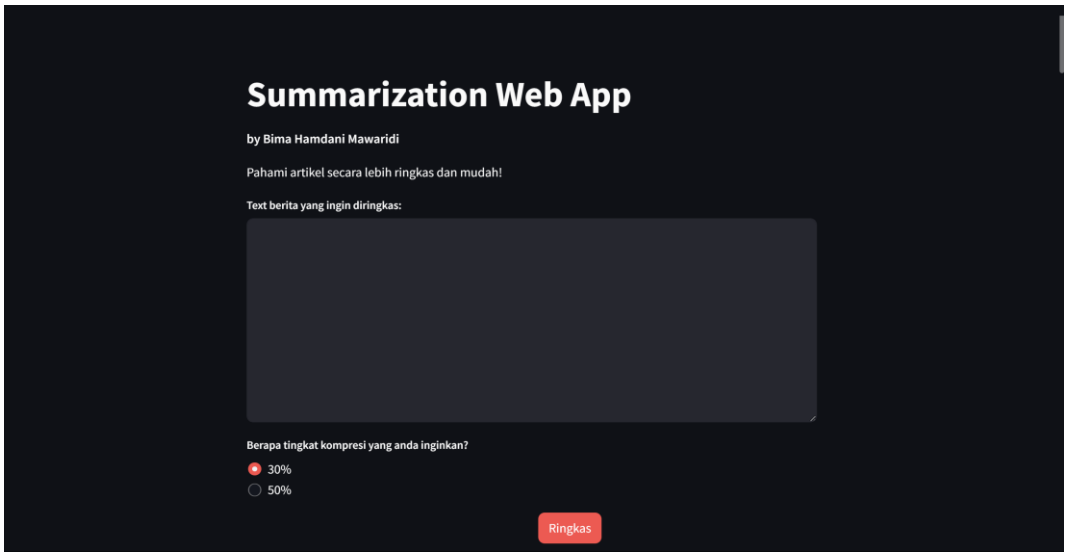
Tabel 3.25 Contoh perhitungan ROUGE-1

Kata unik ringkasan sistem	**Kata unik ringkasan referensi**	**Kata unik overlap**	***recall***	***precision***	***f1-score***
5	7	3	3/7 = 0,42	3/5 = 0,6	(2 x 0,6 x 0,42) / (0,6 + 0,42) = 0,49

Tabel 3.25 merupakan contoh menghitung nilai ROUGE-1. ROUGE-1 memperhitungkan kesamaan *unigram*, jumlah kesamaan *unigram* pada contoh kalimat yaitu sebanyak 3 kata antara lain "peraturan", "menteri", dan "keuangan". Nilai *recall* didapat dari jumlah kata yang sama antara ringkasan sistem dengan ringkasan referensi yaitu 3, kemudian dibagi dengan jumlah kata pada ringkasan referensi yaitu 7. Maka didapat nilai *recall* sebesar 0,42. Sedangkan nilai *precision* didapat dari jumlah kata yang sama antara hasil ringkasan oleh sistem dengan ringkasan referensi yaitu 3 dibagi dengan jumlah seluruh kata pada ringkasan oleh sistem yaitu 5, maka didapat nilai *precision* sebesar 0,6. Untuk nilai *f1-score* merupakan kombinasi nilai *recall* dan *precision* yang didapat dari persamaan 3.5, sehingga menghasilkan nilai *f1-score* sebesar 0.49. *Recall* mengukur berapa banyak elemen penting dari teks referensi yang berhasil ditangkap oleh sistem peringkasan atau terjemahan. *Precision* dalam konteks ROUGE mengukur seberapa banyak *n-gram* yang dihasilkan oleh sistem yang benar-benar relevan dengan teks referensi.

3.8 Implementasi Sistem

Implementasi *user interface* sistem peringkasan teks berita berbahasa Indonesia dapat dilihat pada Gambar 3.4.



Gambar 3.4 Implementasi Sistem

Gambar 3.4 merupakan tampilan sistem untuk *user*. Pertama *user* memasukkan teks artikel asli yang akan diringkas, kemudian memilih tingkat kompresi antara 30% atau 50%. Setelah diklik *button* “ringkas” akan ditampilkan kata kunci dari teks sebanyak sepuluh dan hasil ringkasan sesuai dengan tingkat kompresi yang dipilih.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Skenario Uji Coba

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.

4.2 Hasil Uji Coba

Berdasarkan skenario uji coba yang telah dijelaskan pada sub bab 4.1 pengujian dilakukan untuk membandingkan ringkasan sistem dengan ringkasan manusia. Penelitian ini menggunakan evaluasi ROUGE-1 dalam mendapatkan nilai *recall*, *precision*, dan *f1-score*. Tabel 4.1 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 4.1 Hasil ekstraksi kata kunci dengan LDA

Nomor Dokumen	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
6	betis, main, barca, buat, messi, suarez, pertama, bola, babak, rakitic
7	skor, vietnam, indonesia, gol, gagal, raih, hingga, main, unggul, ubah
8	menit, imbang, mil, skor, menang, sempat, akhir, laga, peringkat, perdana
9	kasper, museum, lukis, inap, sendiri, night, menang, tidur, watch, undi
10	larang, cadar, terap, hak, adil, belgia, sama, kena, atur, warga
…	…
2491	bandara, jember, perintah, budi, kembang, notohadinegoro, bangun, kerja, panjang, kabupaten
2492	pihak, tugas, kabur, serah, tahan, muji, tim, keluarga, bentuk, lkpa
2493	united, rooney, akhir, gol, masuk, inggris, laga, sukses, everton, liga
2494	negara, embun, level, masuk, selatan, jalan, korea, aman, as, tingkat
2495	main, kroos, heynckes, prestasi, bayern, munich, champions, jerman, biasa, strategi
2496	pisang, hitam, kulit, pie, rasa, makan, takut, matang, paling, warna
2497	film, cinta, hidup, landa, masalah, pasang, saling, bangsa, akhir, hari
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

Header pada tabel statistik hasil uji coba menggunakan istilah yang lebih singkat. Berikut keterangan istilah yang digunakan pada statistik hasil uji coba:

W Sistem : Jumlah kata unik pada hasil ringkasan sistem
W Manusia : Jumlah kata unik pada ringkasan manusia
W Overlap : Jumlah kata yang sama pada ringkasan sistem dan ringkasan manusia
S Sistem : Jumlah kalimat pada ringkasan sistem
S Manusia : Jumlah kalimat pada ringkasan manusia

4.2.1 Percobaan Skenario 1 MMR Kueri LDA (*Lambda* = 0.5)

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 4.2 menampilkan statistik kata dan kalimat pada hasil ringkasan yang diperoleh dari peringkasan manual oleh manusia dan peringkasan otomatis oleh sistem.

Tabel 4.2 Statistik jumlah kata dan kalimat pada skenario 1

Nomor Dokumen	***Compression rate***									
	50%					**30%**				
	W sistem	**W manusia**	**W overlap**	**S Sistem**	**S manusia**	**W sistem**	**W manusia**	**W overlap**	**S Sistem**	**S manusia**
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

Evaluasi menggunakan metrik ROUGE-1 untuk menilai kualitas ringkasan yang dihasilkan oleh sistem. Evaluasi ini melibatkan perhitungan nilai *recall*, *precision*, dan *f1-score* pada setiap dokumen artikel yang telah diringkas. Tabel 4.3 merupakan hasil evaluasi ringkasan untuk skenario ke-1.

Tabel 4.3 Hasil perhitungan ROUGE-1 pada skenario 1

Nomor Dokumen	***Compression rate***					
	50%			**30%**		
	Recall	***Precision***	***F1-Score***	***Recall***	***Precision***	***F1-Score***
1	0.929825	0.616279	0.741259	0.561404	0.571429	0.566372
2	0.923077	0.695652	0.793388	0.730769	0.760000	0.745098
3	0.938776	0.308725	0.464646	0.877551	0.401869	0.551282
4	1.000000	0.500000	0.666667	0.327273	0.300000	0.313043
5	0.722222	0.300000	0.423913	0.703704	0.441860	0.542857
…	...	...	...	...	...	...
2496	0.204082	0.384615	0.266667	0.183673	0.500000	0.268657
2497	0.735849	0.319672	0.445714	0.660377	0.472973	0.551181
2498	0.511628	0.241758	0.328358	0.279070	0.179104	0.218182
2499	1.000000	0.384615	0.555556	1.000000	0.592105	0.743802
2500	0.568182	0.454545	0.505051	0.568182	0.568182	0.568182

Nilai evaluasi didapatkan dari perbandingan ringkasan sistem dengan ringkasan manual. Gambar 4.1 menunjukkan ringkasan manual oleh manusia pada artikel ke-4, Gambar 4.2 dan Gambar 4.3 merupakan hasil ringkasan sistem dengan tingkat kompresi 30% dan 50%. Pada dokumen artikel ke-1 menggunakan *query* [aksi, abu, ormas, gelar, bakal, usaha, rencana, organisasi, selesai, cepat]. Gambar 4.4 memperlihatkan teks asli dari dokumen artikel ke-1 sebelum diringkas.

Aliansi Organisasi Massa (Ormas) Islam se - Jabodetabek menyatakan siap menggelar aksi hari ini untuk menggugat Peraturan Pemerintah Pengganti Undang-Undang (Perppu) No.2 tahun 2017 tentang Ormas. "Insya Allah jadi pukul 13.00 WIB di Monas, patung kuda," kata Koordinator Aksi Abu Zidan. Berdasarkan selebaran poster elektronik beredar, setidaknya ada 27 organisasi yang akan ikut beraksi. Mereka berkumpul untuk menyampaikan pendapat menolak Perppu Ormas.

Gambar 4.1 Ringkasan manual artikel-1

Abu mengatakan aksi rencananya bakal digelar sampai pukul 16:00 WIB. Namun ia mengusahakan aksi selesai lebih cepat yakni sekitar pukul 15:45 WIB. Jakarta, CNN Indonesia - - Aliansi Organisasi Massa (Ormas) Islam se - Jabodetabek menyatakan siap menggelar aksi hari ini untuk menggugat Peraturan Pemerintah Pengganti Undang-Undang (Perppu) No.2 tahun 2017 tentang Organisasi Kemasyarakatan (Ormas). Ia juga selalu berkomunikasi dengan kepolisian untuk teknis pelaksanaan aksi.

Gambar 4.2 Ringkasan sistem *compression rate* 30% artikel-1

Abu mengatakan aksi rencananya bakal digelar sampai pukul 16:00 WIB. Namun ia mengusahakan aksi selesai lebih cepat yakni sekitar pukul 15:45 WIB. Jakarta, CNN Indonesia - - Aliansi Organisasi Massa (Ormas) Islam se - Jabodetabek menyatakan siap menggelar aksi hari ini untuk menggugat Peraturan Pemerintah Pengganti Undang-Undang (Perppu) No.2 tahun 2017 tentang Organisasi Kemasyarakatan (Ormas). Ia juga selalu berkomunikasi dengan kepolisian untuk teknis pelaksanaan aksi. "Insya Allah jadi pukul 13.00 WIB di Monas, patung kuda," kata Koordinator Aksi Abu Zidan lewat pesan singkat, Selasa (18/7). Berdasarkan selebaran poster elektronik beredar yang juga diterima CNNIndonesia.com, setidaknya ada 27 organisasi yang akan ikut beraksi.

Gambar 4.3 Ringkasan sistem *compression rate* 50% artikel-1

Jakarta, CNN Indonesia - - Aliansi Organisasi Massa (Ormas) Islam se - Jabodetabek menyatakan siap menggelar aksi hari ini untuk menggugat Peraturan Pemerintah Pengganti Undang-Undang (Perppu) No.2 tahun 2017 tentang Organisasi Kemasyarakatan (Ormas). "Insya Allah jadi pukul 13.00 WIB di Monas, patung kuda," kata Koordinator Aksi Abu Zidan lewat pesan singkat, Selasa (18/7). Berdasarkan selebaran poster elektronik beredar yang juga diterima CNNIndonesia.com, setidaknya ada 27 organisasi yang akan ikut beraksi. Mereka akan berkumpul bersama untuk menyampaikan pendapat menolak Perppu Ormas yang membuka peluang pemerintah membubarkan Ormas tersebut. Abu menjelaskan aksi ini bukan hanya dilakukan oleh Hizbut Tahrir Indonesia (HTI). Ia mengklaim aksi tersebut diikuti Ormas Islam se - Jabodetabek." Di antara 27 ormas tersebut di antaranya ada Badan Koordinasi Lembaga Dakwah Kampus (BKLDK), Pesantren mahasiswa UI, Ma'had UIN, Pelajar Islam Indonesia (PII), dan Forum Mubaligh Bekasi." Abu mengatakan aksi rencananya bakal digelar sampai pukul 16:00 WIB. Namun ia mengusahakan aksi selesai lebih cepat yakni sekitar pukul 15:45 WIB." Kemungkinan salat Ashar tidak di lokasi aksi, tapi di masjid-masjid terdekat," kata Abu. "Abu mengaku sudah mendapat izin dari kepolisian sejak Jum'at (14/7) lalu." Ia juga selalu berkomunikasi dengan kepolisian untuk teknis pelaksanaan aksi. (kid / gil)

Gambar 4.4 Teks asli artikel-1

4.2.2 Percobaan Skenario 2 MMR Kueri LDA (*Lambda* = 0.7)

Pada skenario ke-2, menggunakan nilai parameter *lambda (λ)* pada MMR sebesar 0.7. Tabel 4.4 menampilkan statistik kata dan kalimat pada hasil ringkasan yang diperoleh dari peringkasan manual dan peringkasan sistem.

Tabel 4.4 Statistik jumlah kata dan kalimat pada skenario 2

Nomor Dokumen	***Compression rate***									
	50%					**30%**				
	W sistem	**W manusia**	**W overlap**	**S Sistem**	**S manusia**	**W sistem**	**W manusia**	**W overlap**	**S Sistem**	**S manusia**
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

Statistik kata menentukan nilai evaluasi ROUGE-1 dengan menghitung *recall*, *precision,* dan *f1-score* pada setiap dokumen artikel. Hasil evaluasi ringkasan untuk skenario 2 ditunjukkan pada Tabel 4.5.

Tabel 4.5 Hasil perhitungan ROUGE-1 pada skenario 2

Nomor Dokumen	***Compression rate***					
	50%			**30%**		
	Recall	***Precision***	***F1-Score***	***Recall***	***Precision***	***F1-Score***
1	0.771930	0.564103	0.651852	0.754386	0.671875	0.710744
2	0.711538	0.513889	0.596774	0.653846	0.829268	0.731183
3	0.755102	0.251701	0.377551	0.693878	0.357895	0.472222
4	0.763636	0.461538	0.575342	0.509091	0.424242	0.462810
5	0.740741	0.279720	0.406091	0.703704	0.417582	0.524138
…	...	...	...	...	...	...
2496	0.204082	0.384615	0.266667	0.183673	0.500000	0.268657
2497	0.358491	0.206522	0.262069	0.320755	0.309091	0.314815
2498	0.697674	0.288462	0.408163	0.279070	0.179104	0.218182
2499	1.000000	0.360000	0.529412	1.000000	0.505618	0.671642
2500	0.568182	0.320513	0.409836	0.295455	0.333333	0.313253

Contoh dari hasil ringkasan manusia dan sistem pada artikel ke-3 dengan *query* [bank, pt, indonesia, kredit, kartu, data, uang, menteri, informasi, djp] dapat dilihat pada Gambar 4.5, Gambar 4.6, dan Gambar 4.7.

Direktorat Jenderal Pajak Kementerian Keuangan (DJP Kemenkeu) kembali meminta industri perbankan untuk mempersiapkan data transaksi kartu kredit nasabah. Wacana tersebut sempat tenggelam, ketika pemerintah memulai program amnesti pajak 1 Juli 2016 lalu. Hal tersebut sesuai dengan Peraturan Menteri Keuangan (PMK) Nomor 39/PMK.03/2016, yang meminta perbankan nasional menunda pelaporan transaksi kartu kredit hingga program amnesti pajak berakhir 31 Maret 2017 ini.

Gambar 4.5 Ringkasan manual artikel-3

Pan Indonesia Bank, Ltd. Tbk. Sebelumnya Kementerian Keuangan menunda pemberlakuan Peraturan Menteri Keuangan (PMK) Nomor 39/PMK.03/2016, yang meminta perbankan nasional melaporkan transaksi kartu kredit nasabahnya kepada DJP. "Dengan ini kami meminta kepada Bank / Lembaga Penyelenggara Kartu Kredit untuk mempersiapkan data kartu kredit sesuai dengan format data yang telah disepakati dalam Kamus Data dan Informasi Kartu Kredit dari Bank/Lembaga Penyelenggara Kartu Kredit," tutur Lusiani seperti dikutip dalam surat tersebut, Rabu (29/3). Sementara, instruksi untuk menyiapkan kembali data kartu kredit tertuang dalam Surat Kementerian Keuangan Nomor S - 119/PJ. Jakarta, CNN Indonesia - - Direktorat Jenderal Pajak Kementerian Keuangan (DJP Kemenkeu) kembali meminta industri perbankan untuk mempersiapkan data transaksi kartu kredit nasabah. "Informasi teknis mengenai jatuh tempo dan cara penyampaian data tersebut akan kami informasikan lebih lanjut," ujarnya. Data yang diminta DJP terdiri dari data pokok pemegang kartu periode Juni 2016 sampai dengan Maret 2017 untuk seluruh pemegang kartu kredit dan data transaksi kartu kredit periode data Juni 2016 sampai dengan Maret 2017 untuk seluruh pemegang kartu kredit. 10/2017 yang ditandangani oleh Direktur Teknologi Informasi dan Perpajakan DJP Lusiani pada 23 Maret 2017 silam.

Gambar 4.6 Ringkasan sistem *compression rate* 30% artikel-3

Pan Indonesia Bank, Ltd. Tbk. Sebelumnya Kementerian Keuangan menunda pemberlakuan Peraturan Menteri Keuangan (PMK) Nomor 39/PMK.03/2016, yang meminta perbankan nasional melaporkan transaksi kartu kredit nasabahnya kepada DJP. "Dengan ini kami meminta kepada Bank/Lembaga Penyelenggara Kartu Kredit untuk mempersiapkan data kartu kredit sesuai dengan format data yang telah disepakati dalam Kamus Data dan Informasi Kartu Kredit dari Bank/Lembaga Penyelenggara Kartu Kredit," tutur Lusiani seperti dikutip dalam surat tersebut, Rabu (29/3). Sementara, instruksi untuk menyiapkan kembali data kartu kredit tertuang dalam Surat Kementerian Keuangan Nomor S - 119/PJ. Jakarta, CNN Indonesia - - Direktorat Jenderal Pajak Kementerian Keuangan (DJP Kemenkeu) kembali meminta industri perbankan untuk mempersiapkan data transaksi kartu kredit nasabah. "Informasi teknis mengenai jatuh tempo dan cara penyampaian data tersebut akan kami informasikan lebih lanjut, "ujarnya. Data yang diminta DJP terdiri dari data pokok pemegang kartu periode Juni 2016 sampai dengan Maret 2017 untuk seluruh pemegang kartu kredit dan data transaksi kartu kredit periode data Juni 2016 sampai dengan Maret 2017 untuk seluruh pemegang kartu kredit. 10/2017 yang ditandangani oleh Direktur Teknologi Informasi dan Perpajakan DJP Lusiani pada 23 Maret 2017 silam. Secara terpisah, Direktur Penyuluhan, Pelayanan, dan Hubungan Masyarakat (Humas) DJP Hestu Yoga Saksama mengungkapkan, sesuai PMK Nomor 39 tahun 2016, pengumpulan data transaksi kartu kredit akan dilakukan secara rutin setiap akhir bulan. Yoga menekankan , data kartu kredit hanya untuk kepentingan perpajakan yang dijaga kerahasiaannya. Berikut daftar bank yang wajib menyerahkan data transaksi kartu kredit nasabahnya : " Ke depan kan memang data keuangan dan perbankan akan semakin terbuka bagi otoritas perpajakan, misalnya melalui Automatic Exchange of Information, " tutur pria yang akrab disapa Yoga ini. Sesuai pasal 34 UU KUP, pegawai pajak yang membocorkan data wajib pajak bakal terkena ancaman 1 tahun penjara.

Gambar 4.7 Ringkasan sistem *compression rate* 50% artikel-3

Jakarta, CNN Indonesia - - Direktorat Jenderal Pajak Kementerian Keuangan (DJP Kemenkeu) kembali meminta industri perbankan untuk mempersiapkan data transaksi kartu kredit nasabah. Wacana tersebut sempat tenggelam, ketika pemerintah memulai program amnesti pajak 1 Juli 2016 lalu. Sebelumnya Kementerian Keuangan menunda pemberlakuan Peraturan Menteri Keuangan (PMK) Nomor 39/PMK.03 / 2016, yang meminta perbankan nasional melaporkan transaksi kartu kredit nasabahnya kepada DJP. Penundaan tersebut berlaku hingga program amnesti pajak berakhir 31 Maret 2017 ini. Sementara, instruksi untuk menyiapkan kembali data kartu kredit tertuang dalam Surat Kementerian Keuangan Nomor S - 119/PJ.10/2017 yang ditandangani oleh Direktur Teknologi Informasi dan Perpajakan DJP Lusiani pada 23 Maret 2017 silam. Surat itu ditujukan kepada Direktur Utama dari 22 Bank/Lembaga Penyelenggara Kartu Kredit. "Dengan ini kami meminta kepada Bank/Lembaga Penyelenggara Kartu Kredit untuk mempersiapkan data kartu kredit sesuai dengan format data yang telah disepakati dalam Kamus Data dan Informasi Kartu Kredit dari Bank/Lembaga Penyelenggara Kartu Kredit, "tutur Lusiani seperti dikutip dalam surat tersebut, Rabu (29/3). Data yang diminta DJP terdiri dari data pokok pemegang kartu periode Juni 2016 sampai dengan Maret 2017 untuk seluruh pemegang kartu kredit dan data transaksi kartu kredit periode data Juni 2016 sampai dengan Maret 2017 untuk seluruh pemegang kartu kredit. " Informasi teknis mengenai jatuh tempo dan cara penyampaian data tersebut akan kami informasikan lebih lanjut" ujarnya. Secara terpisah, Direktur Penyuluhan, Pelayanan, dan Hubungan Masyarakat (Humas) DJP Hestu Yoga Saksama mengungkapkan, sesuai PMK Nomor 39 tahun 2016, pengumpulan data transaksi kartu kredit akan dilakukan secara rutin setiap akhir bulan. "Ke depan kan memang data keuangan dan perbankan akan semakin terbuka bagi otoritas perpajakan, misalnya melalui Automatic Exchange of Information, "tutur pria yang akrab disapa Yoga ini. Yoga menekankan, data kartu kredit hanya untuk kepentingan perpajakan yang dijaga kerahasiaannya. Sesuai pasal 34 UU KUP, pegawai pajak yang membocorkan data wajib pajak bakal terkena ancaman 1 tahun penjara. "Kami minta masyarakat tidak perlu khawatir sepanjang sudah ikut amnesti dan sudah membayar pajak serta melapor Surat Pemberitahuan Pajak Tahunan dengan benar" ujarnya. Berikut daftar bank yang wajib menyerahkan data transaksi kartu kredit nasabahnya : 1. Pan Indonesia Bank, Ltd. Tbk., 2. PT Bank ANZ Indonesia, 3. PT Bank Bukopin, Tbk., 4. PT Bank Central Asia, Tbk., 5. PT Bank CIMB Niaga, Tbk., 6. PT Bank Danamon Indonesia, Tbk., 7. PT Bank MNC Internasional, 8. PT Bank ICBC Indonesia, 9. PT Bank Maybank Indonesia, Tbk. 10. PT Bank Mandiri (Persero) Tbk.

Gambar 4.8 Teks asli artikel-3

4.2.3 Percobaan Skenario 3 MMR Kueri LDA (*Lambda* = 0.9)

Pada skenario ke-3, menggunakan nilai parameter *lambda (λ)* dalam *Maximum Marginal Relevance* (MMR) sebesar 0.9.

Tabel 4.6 Statistik jumlah kata dan kalimat pada skenario 3

Nomor Dokumen	*Compression rate*									
	50%					30%				
	W sistem	W manusia	W overlap	S Sistem	S manusia	W sistem	W manusia	W overlap	S Sistem	S manusia
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

Tabel 4.6 menampilkan statistik kata dan kalimat dari hasil peringkasan manual oleh manusia dan peringkasan oleh sistem untuk artikel dengan tingkat kompresi 30% dan 50%. Evaluasi menggunakan metrik ROUGE-1 untuk menilai kualitas ringkasan yang dihasilkan oleh sistem. Evaluasi ini melibatkan perhitungan nilai *recall, precision,* dan *f1-score* pada setiap dokumen artikel yang telah diringkas. Hasil evaluasi ringkasan untuk skenario ke-3 ditunjukkan pada Tabel 4.7.

Tabel 4.7 Hasil perhitungan ROUGE-1 pada skenario 3

Dokumen ke-	***Compression rate***					
	50%			**30%**		
	Recall	***Precision***	***F1-Score***	***Recall***	***Precision***	***F1-Score***
1	0.771930	0.594595	0.671756	0.754386	0.671875	0.710744
2	0.923077	0.695652	0.793388	0.653846	0.829268	0.731183
3	0.714286	0.339806	0.460526	0.714286	0.437500	0.542636
4	1.000000	0.523810	0.687500	0.509091	0.424242	0.462810
5	0.740741	0.279720	0.406091	0.703704	0.417582	0.524138
…	...	...	...	...	...	...
2496	0.204082	0.384615	0.266667	0.183673	0.500000	0.268657
2497	0.358491	0.204301	0.260274	0.339623	0.305085	0.321429
2498	0.604651	0.268041	0.371429	0.255814	0.159420	0.196429
2499	1.000000	0.351562	0.520231	1.000000	0.505618	0.671642
2500	0.568182	0.320513	0.409836	0.295455	0.333333	0.313253

Gambar 4.9 dan Gambar 4.10 adalah ringkasan dari sistem dengan tingkat kompresi 50% dan 30% pada artikel ke-5 menggunakan *query* [nikmat, suka, tunjuk, emerson, indonesia, dunia, wayang, amerika, serikat, asal]. Gambar 4.11 menunjukkan hasil peringkasan yang dilakukan oleh manusia. Gambar 4.12 merupakan teks artikel asli sebelum diringkas.

Emerson adalah pecinta budaya asal Michigan, Amerika Serikat yang bertugas menerjemahkan pertunjukkan tersebut secara simultan ke dalam bahasa Inggris. Saya sangat menikmatinya. Para tamu undangan semakin karena pesinden yang tampil berasal dari sejumlah negara diantaranya, Hiromi Kano (Jepang), Dora Hyorfi (Hungaria), serta Agnes Feroso (Amerika Serikat). Terlebih ada juga Kitsie Emerson, yang merupakan lulusan PhD Wayang Studies dari Universitas Leiden di Belanda. Jakarta, CNN Indonesia - - Pertunjukkan wayang Indonesia bertema "The Revelation of God through the holy spirit : received by Romo Resi Brotonirmoyo " yang digelar di Borobudur International Golf & Country Club, Lembah Gunung Tidar, beberapa waktu lalu, sukses memikat duta besar dari belasan negara sahabat.

Gambar 4.9 Ringkasan sistem *compression rate* 30% artikel-5

Emerson adalah pecinta budaya asal Michigan, Amerika Serikat yang bertugas menerjemahkan pertunjukkan tersebut secara simultan ke dalam bahasa Inggris. Saya sangat menikmatinya. Para tamu undangan semakin karena pesinden yang tampil berasal dari sejumlah negara diantaranya , Hiromi Kano (Jepang), Dora Hyorfi (Hungaria), serta Agnes Feroso (Amerika Serikat). Terlebih ada juga Kitsie Emerson, yang merupakan lulusan PhD Wayang Studies dari Universitas Leiden di Belanda. Jakarta, CNN Indonesia - - Pertunjukkan wayang Indonesia bertema “The Revelation of God through the holy spirit : received by Romo Resi Brotonirmoyo ” yang digelar di Borobudur International Golf & Country Club, Lembah Gunung Tidar, beberapa waktu lalu, sukses memikat duta besar dari belasan negara sahabat. Jirg Kinnen, Kepala Divisi Pers dan Kebudayaan Kedutaan Jerman terlihat menikmati pertunjukkan tersebut. Darmono, pendiri Jababeka sekaligus salah satu pendiri THF, menyebutkan pertunjukkan itu merupakan bentuk kontribusi yayasan untuk lebih mengenalkan budaya Indonesia pada dunia. Pertunjukkan juga diiringi pemain saksofon kelas dunia Harry Wisnu, serta penyanyi Rebecca Quento dan Gracia. “Borobudur itu ibaratnya Mekkah bagi umat Buddha, ” kata dia, sembari menambahkan, Indonesia juga bisa jadi slaah satu destinasi spiritualitas dunia, seperti Roma, India, Jerusalem, ataupun Mekkah .

Gambar 4.10 Ringkasan sistem *compression rate* 50% artikel-5

Pertunjukkan wayang Indonesia bertema" The Revelation of God through the holy spirit : received by Romo Resi Brotonirmoyo " yang digelar di Borobudur International Golf & Country Club, Lembah Gunung Tidar, sukses memikat duta besar dari belasan negara sahabat. Beberapa duta besar yang hadir berasal dari Australia, Filipina, Hungaria, Jerman, Kroasia, Laos, Lebanon, Mongolia, Oman, Panama, Republik Rakyat China, Serbia, dan Venezuela.

Gambar 4.11 Ringkasan manual artikel-5

Jakarta, Pertunjukkan wayang Indonesia bertema “ The Revelation of God through the holy spirit : received by Romo Resi Brotonirmoyo” yang digelar di Borobudur International Golf & Country Club, Lembah Gunung Tidar, beberapa waktu lalu, sukses memikat duta besar dari belasan negara sahabat. Perhelatan yang digagas Tidar Heritage Foundation (THF) itu memang bertujuan memperkenalkan budaya Indonesia di mata dunia. Beberapa duta besar yang hadir berasal dari Australia, Filipina, Hungaria, Jerman, Kroasia, Laos, Lebanon, Mongolia, Oman, Panama, Republik Rakyat China, Serbia, dan Venezuela. Para tamu undangan semakin karena pesinden yang tampil berasal dari sejumlah negara diantaranya, Hiromi Kano (Jepang), Dora Hyorfi (Hungaria), serta Agnes Feroso (Amerika Serikat). Pertunjukkan juga diiringi pemain saksofon kelas dunia Harry Wisnu, serta penyanyi Rebecca Quento dan Gracia. Terlebih ada juga Kitsie Emerson, yang merupakan lulusan PhD Wayang Studies dari Universitas Leiden di Belanda. Emerson adalah pecinta budaya asal Michigan, Amerika Serikat yang bertugas menerjemahkan pertunjukkan tersebut secara simultan ke dalam bahasa Inggris. Jirg Kinnen, Kepala Divisi Pers dan Kebudayaan Kedutaan Jerman terlihat menikmati pertunjukkan tersebut. “ Saya suka, Saya sangat menikmatinya. Ada panduan yang sangat membantu untuk memahami jalan ceritanya, ” ujar dia. Hal serupa disampaikan Steven Barraclough dari Kedutaan Australia. Selain itu, hadir juga sejumlah tokoh seperti Cosmas Batubara, Fadel Mohammad, Mardiyanto, Didik Nini Thowok dan Ibu Jero Wacik. Adapun, Kementrian Pariwisata diwakili Deputi Bidang Pengembangan Pemasaran Pariwisata Nusantara, Esthy Reko Astuty dan Sekretaris Tim Percepatan Borobudur Watie Murani. S.D. Darmono, pendiri Jababeka sekaligus salah satu pendiri THF, menyebutkan pertunjukkan itu merupakan bentuk kontribusi yayasan untuk lebih mengenalkan budaya Indonesia pada dunia . “ THF ingin berkontribusi pada negara, bangsa bahkan kepada dunia melalui promosi perdamaian dan harmoni , salah satunya lewat pertunjukkan budaya, ” ujarnya . Selain itu, langkah lain yang juga dilakukan THF adalah mendorong pemerintah segera merevitalisasi Candi Borobudur. “ Borobudur itu ibaratnya Mekkah bagi umat Buddha, ” kata dia, sembari menambahkan, Indonesia juga bisa jadi slaah satu destinasi spiritualitas dunia, seperti Roma, India, Jerusalem, ataupun Mekkah.

Gambar 4.12 Teks asli artikel-5

4.3 Pembahasan

Nilai *recall* yang tinggi berarti ringkasan sistem lebih relevan dengan ringkasan manusia. Semua ringkasan manusia masuk ke dalam ringkasan sistem jika nilai *recall* mendekati nilai maksimum atau sama dengan 1. Sebaliknya, ringkasan sistem secara keseluruhan masuk ke dalam ringkasan manusia jika nilai *precision* mencapai nilai maksimum 1. Kombinasi nilai *recall* dan *precision* yaitu *f1-score* memberikan gambaran keseluruhan tentang kemampuan sistem dalam menangkap dan menyajikan informasi yang sesuai serta relevan dalam ringkasannya.

Tingkat kompresi 50% menghasilkan nilai *f1-score* tertinggi pada skenario 1 yaitu 0.924, skenario 2 menghasilkan nilai tertinggi sebesar 0.915, dan skenario 3 nilai *f1-score* tertinggi dengan nilai 0.938. Nilai *f1-score* terendah untuk skenario 1 yaitu 0.090, skenario 2 mendapatkan nilai *f1-score* terendah 0.075, dan skenario 3 mendapatkan nilai *f1-score* terendah sebesar 0.067. Tingkat kompresi 30% menghasilkan nilai *f1-score* tertinggi untuk skenario 1 yaitu 0.999, skenario 2 sebesar 0.999. Skenario 3 mendapatkan nilai *f1-score* tertinggi sebesar 0.888. Nilai terendah *f1-score* pada skenario 1, 2, dan 3 ada yang mendapatkan nilai 0, hal tersebut karena memang jumlah kalimat asli artikel sangat sedikit, yaitu pada artikel nomor 2041 yang teks aslinya hanya terdiri dari 6 kalimat.

Berdasarkan hasil keseluruhan evaluasi dari 2500 artikel yang diujicobakan, Tabel 4.8 menampilkan rata-rata hasil akhir sistem ketika dibangun menggunakan dua tingkat kompresi berbeda dan tiga variasi nilai *lambda*.

Tabel 4.8 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 4.8, 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.

Pada penelitian yang dilakukan oleh (Alfhi Saputra, 2021) membuat sistem peringkasan teks menggunakan metode *Long Short-Term Memory* (LSTM) dengan dataset *IndoSum* menghasilkan nilai rata-rata ROUGE-1 terbaik sebesar 0.13846. Penelitian juga dilakukan oleh (Nyoman Purnama & Ni Nengah Widya Utami, 2023) dengan metode *Text to Text Transfer Transformer* (T5). Pengujian menggunakan ROUGE-1 memperoleh hasil terbaik dengan nilai ROUGE-1 sebesar 0,17568.

Berdasarkan percobaan yang dilakukan bahwa sistem mampu merangkum berita secara otomatis tanpa menggunakan judul artikel melainkan menggunakan

kata kunci artikel. Hasil peringkasan menggunakan MMR dan LDA juga lebih baik dari penelitian yang menggunakan metode LSTM dan T5 dengan *dataset* yang sama. Dibandingkan dengan peringkasan manual yang membutuhkan waktu lebih lama, cara ini selain lebih efisien juga menghemat waktu. Tersedianya sistem peringkasan berita ini memudahkan dan mempercepat proses pemahaman informasi dari dokumen tanpa menggunakan judul yang berpotensi tidak sesuai dengan isi konten berita. Hal ini sesuai dengan tujuan sistem peringkasan, yaitu membantu memperoleh informasi yang relevan dan penting.

4.4 Integrasi Islam

Sistem peringkasan teks otomatis ini sejalan dengan dua konsep muamalah, yaitu muamalah dengan Allah (muamalah ma'a Allah) dan muamalah dengan sesama manusia (muamalah ma'a an-nas). Mu'amalah kepada Allah merupakan segala sesuatu yang berhubungan antara makhluk hidup dengan Allah. Sedangkan muamalah ma'a an-nas merupakan hal-hal yang berhubungan dengan sesama manusia. Konsep muamalah ma'a Allah dalam penelitian ini terdapat pada aspek diperbolehkannya meringkas sesuatu dan dianjurkan menyampaikan suatu penjelasan secara ringkas. Konsep muamalah ma'a an-nas terdapat pada aspek saling membantu dan berbuat kebaikan dengan sesama manusia.

4.4.1 Muamalah Ma'a Allah

Banyak artikel berita disajikan dengan judul yang seringkali kurang sesuai dengan konten serta terdapat teks yang cukup panjang. Sistem peringkas secara otomatis dapat membantu untuk mempersingkat waktu dalam memperoleh

informasi. Sistem peringkasan teks berita otomatis dapat menyaring materi utama dari artikel berita yang panjang tanpa menghapus informasi penting. Menyampaikan suatu perkataan dianjurkan langsung kepada intinya atau tidak melebihkan-lebihkan. Hal tersebut sebagaimana firman Allah subhanahu wa ta'ala dalam surah Al-Ahzab ayat 70 dalam Al-Qur'an yang berbunyi:

يٰٓاَيُّهَا الَّذِيْنَ اٰمَنُوا اتَّقُوا اللّٰهَ وَقُوْلُوْا قَوْلًا سَدِيْدًاۙ ﴿٧٠﴾

"Wahai orang-orang yang beriman, bertakwalah kamu kepada Allah dan ucapkanlah perkataan yang benar." (QS. Al-Ahzab/33:70)

Imam Ibnu Katsir membahas surah Al-Ahzab ayat 70 dalam kitab tafsir Al-Qur'an Al-'Azhim, dalam ayat tersebut terdapat kata *"Qaulan Sadidan"* berarti perkataan yang lurus tidak berbelit-belit, jujur, dan sesuai dengan kebenaran. Beliau menjelaskan pentingnya kejujuran dan kejelasan dalam berbicara tanpa menyimpang atau berbelit-belit. Dalam tafsir lainnya yaitu tafsir Al-Azhar oleh Abdul Malik Karim Amrullah dituliskan bahwa *"Di antara sikap hidup karena iman dan taqwa adalah jika berkata-kata pilihlah kata-kata yang tepat. Dalam kata yang tepat itu terkandunglah perkataan yang benar. Jangan berbelit-belit. Jangan yang dimaksud lain, tetapi yang dipakai lain pula,"*. Berdasarkan kedua tafsir tersebut dianjurkan berbicara langsung ke intinya agar pesan yang disampaikan dapat diterima dengan baik dan tidak menimbulkan kebingungan. Suatu artikel berita seringkali mempunyai teks yang panjang sehingga pembaca butuh waktu lebih untuk memahami inti berita. Dengan sistem peringkasan teks, berita dapat disajikan secara ringkas tanpa menghilangkan informasi penting dan langsung ke intinya agar tidak menimbulkan kebingungan bagi pembaca.

Menyampaikan sesuatu secara ringkas dan jelas juga dianjurkan melalui hadits yang diriwayatkan oleh Imam Ahmad bin Hanbal.

حَدَّثَنَا قُرَيْشُ بْنُ إِبْرَاهِيمَ قَالَ حَدَّثَنَا عَبْدُ الرَّحْمَنِ بْنُ عَبْدِ الْمَلِكِ بْنِ أَبْجَرَ عَنْ أَبِيهِ عَنْ وَاصِلِ بْنِ حَيَّانَ قَالَ قَالَ أَبُو وَائِلٍ خَطَبَنَا عَمَّارٌ فَأَبْلَغَ وَأَوْجَزَ فَلَمَّا نَزَلَ قُلْنَا يَا أَبَا الْيَقْظَانِ لَقَدْ أَبْلَغْتَ وَأَوْجَزْتَ فَلَوْ كُنْتَ تَنَفَّسْتَ قَالَ إِنِّي سَمِعْتُ رَسُولَ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ يَقُولُ إِنَّ طُولَ صَلَاةِ الرَّجُلِ وَقِصَرَ خُطْبَتِهِ مَئِنَّةٌ مِنْ فِقْهِهِ فَأَطِيلُوا الصَّلَاةَ وَأَقْصِرُوا الْخُطْبَةَ فَإِنَّ مِنْ الْبَيَانِ لَسِحْرًا

Telah menceritakan kepada kami [Quraisy bin Ibrahim] telah menceritakan kepada kami [Abdurrahman bin Abdul Malik bin Abjar] dari [Bapaknya] dari [Washil bin Hayyan] ia berkata, [Abu Wa`il] berkata, "Ammar pernah berkhutbah di hadapan kami dengan khutbah yang singkat menyentuh hati. Saat ia turun, maka kami berkata, "Wahai Abu Yaqzhan, sungguh kamu telah menyampaikan khutbah yang menyentuh namun ringkas, sekiranya kamu mau memangjangkan (khutbah)." Ammar lalu berkata, "Aku mendengar Rasulullah shallallahu 'alaihi wasallam bersabda: "Sungguh, panjangnya shalat dan ringkasnya khutbah yang disampaikan oleh seseorang adalah tanda kefakihannya. Maka panjangkanlah shalat dan ringkaskanlah khutbah. Dan sungguh di antara indahnya penjelasan adalah bagian dari sihir." (H.R. Ahmad no 17598)

Dalam hadits tersebut Rasulullah shallallahu 'alaihi wasallam menyampaikan agar memanjangkan shalat dan meringkas khutbah yang disampaikan. Hal tersebut merupakan suatu tanda berarti orang yang paham terhadap aturan atau syariat Islam (fakih). Dapat dimaknai bahwa suatu penjelasan atau informasi yang disampaikan tidak perlu panjang karena dalam hadits tersebut menyampaikan juga bahwa di antara indahnya penjelasan adalah bagian dari sihir. Dalam hal ini termasuk juga artikel berita yang terdapat teks panjang di dalamnya. Informasi yang berlebihan ini dapat membuat pembaca kesulitan memahami inti berita. Adanya sistem peringkasan teks ini sebagai pelaksanaan amalan yang disampaikan oleh Rasulullah shallallahu 'alaihi wasallam yaitu mendapatkan penjelasan yang secara ringkas namun tidak mengurangi informasi yang penting di dalamnya.

4.4.2 Muamalah Ma'a An-Nas

Memberikan bantuan kepada mereka yang membutuhkan merupakan tindakan yang mengamalkan prinsip muamalah ma'a an-nnas. Dalam situasi dimana sistem yang dikembangkan dapat membantu pengguna mendapatkan informasi dari berita dengan cepat. Saling membantu merupakan salah satu cara manusia menunjukkan kepatuhan kepada Allah subhanahu wa ta'ala.

Sebuah hadits tentang membantu sesama juga diriwayatkan oleh imam Ahmad bin Hanbal.

حَدَّثَنَا هَاشِمٌ حَدَّثَنِي عَبْدُ الْحَمِيدِ حَدَّثَنِي شَهْرٌ حَدَّثَنِي أَبُو ظَبْيَةَ قَالَ إِنَّ شُرَحْبِيلَ بْنَ السِّمْطِ دَعَا عَمْرَو بْنَ عَبَسَةَ السُّلَمِيَّ فَقَالَ يَا ابْنَ عَبَسَةَ هَلْ أَنْتَ مُحَدِّثِي حَدِيثًا سَمِعْتَهُ أَنْتَ مِنْ رَسُولِ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ لَيْسَ فِيهِ تَزَيُّدٌ وَلَا كَذِبٌ وَلَا تُحَدِّثْنِيهِ عَنْ آخَرَ سَمِعَهُ مِنْهُ غَيْرُكَ قَالَ نَعَمْ سَمِعْتُ رَسُولَ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ يَقُولُ إِنَّ اللَّهَ عَزَّ وَجَلَّ يَقُولُ قَدْ حَقَّتْ مَحَبَّتِي لِلَّذِينَ يَتَحَابُّونَ مِنْ أَجْلِي وَحَقَّتْ مَحَبَّتِي لِلَّذِينَ يَتَصَافُّونَ مِنْ أَجْلِي وَحَقَّتْ مَحَبَّتِي لِلَّذِينَ يَتَزَاوَرُونَ مِنْ أَجْلِي وَحَقَّتْ مَحَبَّتِي لِلَّذِينَ يَتَبَاذَلُونَ مِنْ أَجْلِي وَحَقَّتْ مَحَبَّتِي لِلَّذِينَ يَتَنَاصَرُونَ مِنْ أَجْلِي

Telah menceritakan kepada kami [Hasyim] telah menceritakan kepadaku [Abdul Hamid] telah menceritakan kepadaku [Syahr] telah menceritakan kepadaku [Abu Dzabyah] ia berkata; Bahwasanya Syurahbil bin As Simth memanggil Amru bin Abasah As Sulami dan bertanya, "Wahai Ibnu Abasah, apakah Anda mau menceritakan kepadaku suatu hadits yang telah Anda dengar dari Rasulullah shallallahu 'alaihi wasallam tanpa ada tambahan ataupun kedustaan? Dan janganlah Anda ceritakan kepadaku suatu hadits dari sahabat lain yang ia mendengarnya dari beliau selain Anda. [Amru bin Abasah] berkata; Baiklah, aku telah mendengar Rasulullah shallallahu 'alaihi wasallam bersabda: "Allah Ta'ala telah berfirman: 'Sungguh telah berhak mendapatkan kecintaan-Ku orang-orang yang saling mencintai karena Aku, dan sungguh telah berhak mendapatkan kecintaan-Ku orang-orang yang saling merapatkan barisan karena Aku, dan sungguh telah berhak mendapatkan kecintaan-Ku orang-orang yang saling mengunjungi karena Aku, dan sungguh telah berhak mendapatkan kecintaan-Ku orang-orang yang saling berkorban (untuk membantu yang lain) karena Aku, dan sungguh telah berhak mendapatkan kecintaan-Ku orang-orang yang saling menolong karena Aku.'" (H.R. Ahmad no 18621)

Hadits di atas mengimbau seluruh umat islam untuk saling membantu antara sesama manusia. Saling membantu antara saudara atau saudari muslim dalam

kesulitan atau kebutuhan merupakan tindakan yang mendapatkan kecintaan dari Allah subhanahu wa ta'ala. Pengembangan sistem peringkasan teks ini menjadi contoh tolong-menolong dalam ketakwaan karena dilakukan dengan tujuan kebaikan. Sistem ini memudahkan para pembaca, akademisi, jurnalis dan pekerjaan lain untuk memperoleh informasi tanpa membutuhkan waktu yang lama dan tidak kehilangan informasi yang penting. Teknik meringkas teks ini dimaksudkan untuk membantu orang lain dalam ikhtiarnya sebagai sarana mentaati ajaran Allah subhanahu wa ta'ala dalam ketakwaan dan kebaikan agar mendapat kasih sayang-Nya. Berbuat baik ke sesama manusia juga sesuai dengan firman Allah subhanahu wa ta'ala dalam surah Al-Qashash ayat 77 yang berbunyi:

وَابْتَغِ فِيْمَآ اٰتٰىكَ اللّٰهُ الدَّارَ الْاٰخِرَةَ وَلَا تَنْسَ نَصِيْبَكَ مِنَ الدُّنْيَا وَاَحْسِنْ كَمَآ اَحْسَنَ اللّٰهُ اِلَيْكَ وَلَا تَبْغِ الْفَسَادَ فِى الْاَرْضِۗ اِنَّ اللّٰهَ لَا يُحِبُّ الْمُفْسِدِيْنَ ٧٧

"Dan, carilah pada apa yang telah dianugerahkan Allah kepadamu (pahala) negeri akhirat, tetapi janganlah kamu lupakan bagianmu di dunia. Berbuat baiklah (kepada orang lain) sebagaimana Allah telah berbuat baik kepadamu dan janganlah kamu berbuat kerusakan di bumi. Sesungguhnya Allah tidak menyukai orang-orang yang berbuat kerusakan." (QS. Al-Qashash/28:77)

Menurut tafsir Ibnu Katsir dalam kitab Al-Qur'an Al-'Azhim sebagai umat muslim dianjurkan mencari kebahagiaan akhirat dengan tidak melupakan kenikmatan duniawi yaitu apa yang dihalalkan oleh Allah subhanahu wa ta'ala. Sebagai manusia mempunyai kewajiban terhadap Tuhan, diri sendiri, keluarga, dan orang-orang yang bertamu, maka setiap kewajiban itu harus dilaksanakan dan berbuat baik kepada orang lain sebagaimana Allah telah berbuat baik ke umatnya. Dengan sistem peringkasan ini ketika menemui teks artikel yang panjang tidak akan membutuhkan waktu yang lama untuk mengetahui inti sari dari artikel berita

tersebut. Sehingga waktu dapat dimanfaatkan untuk memenuhi kewajiban kepada Allah subhanahu wa ta’ala yaitu mencari kebahagiaan akhirat. Nilai berikutnya yang dapat diambil yaitu sistem peringkasan ini diharapkan membantu pembaca berita dalam memahami suatu informasi dari teks yang panjang sehingga termasuk kebaikan kepada sesama.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

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.

Keberhasilan implementasi sistem yang dikembangkan menjadi bentuk pengamalan muamalah ma'a Allah, sejalan dengan Al-Qur'an surah Al-Ahzab ayat 70 dan hadits riwayat Ahmad no. 17598 serta muamalah ma’a an-nnas yang sesuai dengan surah Al-Qashash ayat 77 dan hadits yang diriwayatkan oleh Imam Ahmad bin Hanbal no. 18621.

5.2 Saran

Peneliti menyadari terdapat beberapa kekurangan di dalam penelitian ini, untuk mendapatkan hasil terbaik dalam membuat sistem peringkasan teks otomatis dalam bahasa Indonesia dapat menerapkan beberapa saran. Berikut beberapa saran untuk peneliti selanjutnya:

1. Menggunakan data ringkasan manual yang lebih panjang, dikarenakan ringkasan yang terlalu sedikit berpotensi kehilangan informasi penting yang ada dalam artikel berita.
2. Menambah kamus *stopword* agar model LDA dapat menghasilkan kata kunci yang lebih spesifik.
3. Menggunakan beberapa variasi nilai *Alpha* dan *Eta* pada pemodelan topik LDA untuk mengetahui variasi nilai yang menghasilkan kata kunci lebih baik.

DAFTAR PUSTAKA

A.S. Haris Sumadiria. (2014). *Jurnalistik Indonesia : Menulis Berita dan Feature Panduan Praktis Jurnalistik Profesional*. Simbiosa Rekatama Media.

Abdi, A., Shamsuddin, S. M., & Aliguliyev, R. M. (2018). QMOS: Query-based multi-documents opinion-oriented summarization. *Information Processing and Management*, *54*(2), 318–338. https://doi.org/10.1016/j.ipm.2017.12.002

Alfhi Saputra, M. (2021). Peringkas Teks Otomatis Bahasa Indonesia secara Abstraktif Menggunakan Metode Long Short-Term Memory. *E-Proceeding of Engineering : Vol.8, No.2 April 2021 |*, *8*(2), 3474–3488.

Almajid, A. D., & Wirawanda, Y. (2023). *Persepsi Wartawan Kota Surakarta Mengenai Penggunaan Clickbait Pada Judul Berita Di Akun Media Sosial*. 1–25.

Andika, M., Chaerani, L., Data, K. K., Allocation, L. D., Online, M., & Topik, P. (2021). Pemodelan Topik Berita pada Portal Berita Online Berbahasa Indonesia Menggunakan Latent Dirichlet Allocation (LDA). *Jurnal Ilmiah Komputasi*, *20*(2), 173–180. https://doi.org/10.32409/jikstik.20.2.2719

Arie Atwa Magriyanti. (2021). Maximum Marginal Relevance Berbasis Boolean Model Pada Peringkasan Artikel Berita Pendek. *Jurnal Ilmiah Teknik Informatika Dan Komunikasi*, *1*(3), 77–88. https://doi.org/10.55606/juitik.v1i3.132

Atikah, L., Hasanah, N. A., & Arifin, A. Z. (2022). Topic Modelling Using VSM-LDA For Document Summarization. *Ultimatics : Jurnal Teknik Informatika*, *14*(2), 91–95. https://doi.org/10.31937/ti.v14i2.2854

Belwal, R. C., Rai, S., & Gupta, A. (2023). Extractive text summarization using clustering-based topic modeling. *Soft Computing*, *27*(7), 3965–3982. https://doi.org/10.1007/s00500-022-07534-6

Blei, D. M., Ng, A. Y., & Jordan, M. T. (2003). Latent dirichlet allocation. *Advances in Neural Information Processing Systems*, *3*, 993–1022.

Firman A, D., Fahrur Rozi, I., & Kusumaning Putri, I. (2022). Peringkasan Teks Otomatis pada Portal Berita Olahraga menggunakan metode Maximum Marginal Relevance. *Jurnal Informatika Polinema*, *8*(3), 21–30. https://doi.org/10.33795/jip.v8i3.519

Goldstein, J., & Carbonell, J. (1998). Summarization: (1) Using Mmr for Diversity-Based Reranking and (2) Evaluating Summaries. *TIPSTER TEXT PROGRAM PHASE III: Proceedings of a Workshop*, *1*, 181–195.

Gunawan, G., Fitria, F., Setiawan, E. I., & Fujisawa, K. (2023). Maximum Marginal Relevance and Vector Space Model for Summarizing Students' Final Project

Abstracts. *Knowledge Engineering and Data Science*, *6*(1), 57. https://doi.org/10.17977/um018v6i12023p57-68

Halimah, Surya Agustian, & Siti Ramadhani. (2022). Peringkasan teks otomatis (automated text summarization) pada artikel berbahasa indonesia menggunakan algoritma lexrank. *Jurnal CoSciTech (Computer Science and Information Technology)*, *3*(3), 371–381. https://doi.org/10.37859/coscitech.v3i3.4300

Hernawan, Y. F., Adikara, P. P., & Wihandika, R. C. (2022). Peringkasan Artikel Berbahasa Indonesia Menggunakan TextRank dengan Pembobotan BM25. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *9*(1), 61–68. https://doi.org/10.25126/jtiik.2022913765

Hidayatullah, A. F., Aditya, S. K., Karimah, & Gardini, S. T. (2019). Topic modeling of weather and climate condition on twitter using latent dirichlet allocation (LDA). *IOP Conference Series: Materials Science and Engineering*, *482*(1). https://doi.org/10.1088/1757-899X/482/1/012033

Issam, K. A. R., Patel*, S., & N., S. C. (2020). Topic Modeling Based Extractive Text Summarization. *International Journal of Innovative Technology and Exploring Engineering*, *9*(6), 1710–1719. https://doi.org/10.35940/ijitee.f4611.049620

Juna, M. F., & Hayaty, M. (2023). The observed preprocessing strategies for doing automatic text summarizing. *Computer Science and Information Technologies*, *4*(2), 119–126. https://doi.org/10.11591/csit.v4i2.p119-126

Kurniawan, K., & Louvan, S. (2018). IndoSum: A New Benchmark Dataset for Indonesian Text Summarization. *Proceedings of the 2018 International Conference on Asian Language Processing, IALP 2018*, 215–220. https://doi.org/10.1109/IALP.2018.8629109

Lin, C. Y. (2004). Rouge: A package for automatic evaluation of summaries. *Proceedings of the Workshop on Text Summarization Branches out (WAS 2004)*, *1*, 25–26. papers2://publication/uuid/5DDA0BB8-E59F-44C1-88E6-2AD316DAEF85

Liu, H., Zhang, T., Li, F., Yu, M., & Yu, G. (2024). A probabilistic generative model for tracking multi-knowledge concept mastery probability. *Frontiers of Computer Science*, *18*(3). https://doi.org/10.1007/s11704-023-3008-x

Mao, X., Huang, S., Shen, L., Li, R., & Yang, H. (2021). Single document summarization using the information from documents with the same topic. *Knowledge-Based Systems*, *228*, 107265. https://doi.org/10.1016/j.knosys.2021.107265

Murniyetti. (2016). Waktu dalam Perspektif Al-Qur'an. *Jurnal Ulunnuha*, *6*(1), 93–101.

Mustaqhfiri, M., Abidin, Z., Kusumawati, R. (2011). Peringkasan Teks Otomatis

Berita Menggunakan Metode Maximum Marginal Relevance. *Matics: Jurnal Ilmu Komputer Dan Teknologi Informasi*, *4*(1), 23–32. https://doi.org/https://doi.org/10.18860/mat.v0i0.1578

Mustikasari, D., Widaningrum, I., Arifin, R., & Putri, W. H. E. (2021). Comparison of Effectiveness of Stemming Algorithms in Indonesian Documents. *Proceedings of the 2nd Borobudur International Symposium on Science and Technology (BIS-STE 2020)*, *203*, 154–158. https://doi.org/10.2991/aer.k.210810.025

Nurdiana, O., Jumadi, J., & Nursantika, D. (2016). Perbandingan Metode Cosine Similarity Dengan Metode Jaccard Similarity Pada Aplikasi Pencarian Terjemah Al-Qur'an Dalam Bahasa Indonesia. *Jurnal Online Informatika*, *1*(1), 59. https://doi.org/10.15575/join.v1i1.12

Nurkholis, A., Alita, D., & Munandar, A. (2022). Comparison of Kernel Support Vector Machine Multi-Class in PPKM Sentiment Analysis on Twitter. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *6*(2), 227–233. https://doi.org/10.29207/resti.v6i2.3906

Nyoman Purnama, & Ni Nengah Widya Utami. (2023). *IMPLEMENTASI PERINGKAS DOKUMEN BERBAHASA INDONESIA MENGGUNAKAN METODE TEXT TO TEXT TRANSFER TRANSFORMER (T5) I Nyoman Purnama 1) , Ni Nengah Widya Utami 2) Program Studi Sistem Informasi 1) , Sistem Informasi Akutansi 2)*. 381–391.

Oktaviana, A. K. N., ER, N. A. S., Mahendra, I. B. M., Astawa, I. G. S., Wibawa, I. G. A., & Mogi, I. K. A. (2022). Pemodelan Topik Artikel Berita Menggunakan Structural Topic Model dan Latent Dirichlet Allocation. *JELIKU (Jurnal Elektronik Ilmu Komputer Udayana)*, *11*(3), 469. https://doi.org/10.24843/jlk.2023.v11.i03.p02

Pinto, J. C. L., & Chahed, T. (2014). Modeling multi-topic information diffusion in social networks using latent dirichlet allocation and hawkes processes. *Proceedings - 10th International Conference on Signal-Image Technology and Internet-Based Systems, SITIS 2014*, 339–346. https://doi.org/10.1109/SITIS.2014.24

Purbawa, D. P., Malikhah, Anggraini, R. N. E., & Sarno, R. (2021). Automatic Text Summarization using Maximum Marginal Relevance for Health Ethics Protocol Document in Bahasa. *Proceedings of 2021 13th International Conference on Information and Communication Technology and System, ICTS 2021, January 2022*, 324–329. https://doi.org/10.1109/ICTS52701.2021.9607951

Rofiqi, M. A., Fauzan, A. C., Agustin, A. P., & Saputra, A. A. (2019). Implementasi Term-Frequency Inverse Document Frequency (TF-IDF) Untuk Mencari Relevansi Dokumen Berdasarkan Query. *ILKOMNIKA: Journal of Computer Science and Applied Informatics*, *1*(2), 58–64.

https://doi.org/10.28926/ilkomnika.v1i2.18

Roul, R. K. (2021). Topic modeling combined with classification technique for extractive multi-document text summarization. *Soft Computing*, *25*(2), 1113–1127. https://doi.org/10.1007/s00500-020-05207-w

Rusdhi, V. F., & Sari, I. (2022). Identifikasi Topik Artikel Berita Menggunakan Topic Modelling Dengan Latent Dirichlet Allocation. *Jurnal Ilmiah Informatika Komputer*, *27*(2), 169–176. https://doi.org/10.35760/ik.2022.v27i2.6829

Sanjaya, N. A. (2021). lmplementasi Latent Dirichlet Allocation (LDA) untuk Klasterisasi Cerita Berbahasa Bali. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *8*(1), 127–134. https://doi.org/10.25126/jtiik.202183556

Saraswati, N. F., Indriati, & Perdana, R. S. (2018). Peringkasan Teks Otomatis Menggunakan Metode Maximum Marginal Relevance Pada Hasil Pencarian Sistem Temu Kembali Informasi Untuk Artikel Berbahasa Indonesia. In *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya* (Vol. 2, Issue 11, pp. 5494–5502). https://doi.org/10.1016/s1010-6030(01)00380-x

Savanti, N., Gotami, W., & Dewi, R. K. (2018). Peringkasan Teks Otomatis Secara Ekstraktif Pada Artikel Berita Kesehatan Berbahasa Indonesia Dengan Menggunakan Metode Latent Semantic Analysis. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, *2*(9), 2821–2828.

Sivarethinamohan, R., Sujatha, S., & Biswas, P. (2021). Envisioning the potential of Natural Language Processing (NLP) in Health Care Management. *Proceedings of the 7th International Engineering Conference "Research and Innovation Amid Global Pandemic", IEC 2021*, 189–193. https://doi.org/10.1109/IEC52205.2021.9476131

Susanto, E., Mawardi, V. C., & Lauro, M. D. (2021). Aplikasi Clustering Berita Dengan Metode K Means Dan Peringkas Berita Dengan Metode Maximum Marginal Relevance. *Jurnal Ilmu Komputer Dan Sistem Informasi*, *9*(1), 62. https://doi.org/10.24912/jiksi.v9i1.11560

Syed, S., & Spruit, M. (2017). Full-Text or abstract? Examining topic coherence scores using latent dirichlet allocation. *Proceedings - 2017 International Conference on Data Science and Advanced Analytics, DSAA 2017*, *2018-Janua*, 165–174. https://doi.org/10.1109/DSAA.2017.61

Vanessa, V., & Ibrahim, A. L. (2023). Clickbait as a Potential Threat in the Development of Cybercrime in Indonesia. *Jurnal USM Law Review*, *7*(1), 1–17.

Yuliska, Y., & Syaliman, K. U. (2020). Literatur Review Terhadap Metode, Aplikasi dan Dataset Peringkasan Dokumen Teks Otomatis untuk Teks

Berbahasa Indonesia. In *IT Journal Research and Development* (Vol. 5, Issue 1, pp. 19–31). https://doi.org/10.25299/itjrd.2020.vol5(1).4688

Zamzam, M. A., Crysdian, C., & Holle, K. F. H. (2020). Sistem Automatic Text Summarization Menggunakan Algoritma Textrank. *Matics*, *12*(2), 111–116. https://doi.org/10.18860/mat.v12i2.8372

LAMPIRAN

LAMPIRAN-LAMPIRAN

Lampiran I Peluang setiap kosakata dokumen ke-1 terhadap setiap topik

No.	Vocabulary	Jumlah kosakata ke-v pada topik ke-1 (w_{vk})	Jumlah seluruh kosakata pada topik ke-1	Jumlah kosakata ke-v pada topik ke-2 (w_{vk})	Jumlah seluruh kosakata pada topik ke-2	Jumlah kata dalam Vocabulary	Nilai Eta	$p(w_n \mid z_1, \beta)$	$p(w_n \mid z_2, \beta)$
1	akhir	0	16	1	18	30	0.033	0.002	0.054
2	amnesti	2		0				0.120	0.002
3	data	0		1				0.002	0.054
4	direktorat	1		0				0.061	0.002
5	djp	1		0				0.061	0.002
6	hingga	0		1				0.002	0.054
7	industri	0		1				0.002	0.054
8	jakarta	0		1				0.002	0.054
9	jenderal	1		0				0.061	0.002
10	juli	0		1				0.002	0.054
11	kartu	0		1				0.002	0.054
12	kemenkeu	1		0				0.061	0.002
13	kredit	1		0				0.061	0.002
14	laku	0		1				0.002	0.054
15	maret	0		1				0.002	0.054
16	menteri	1		0				0.061	0.002

No.	Vocabulary	Jumlah kosakata ke-v pada topik ke-1 (w_{vk})	Jumlah seluruh kosakata pada topik ke-1	Jumlah kosakata ke-v pada topik ke-2 (w_{vk})	Jumlah seluruh kosakata pada topik ke-2	Jumlah kata dalam Vocabulary	Nilai Eta	$p(w_n \mid z_1, \beta)$	$p(w_n \mid z_2, \beta)$
17	minta	0	16	1	18	30	0.033	0.002	0.054
18	mulai	0		1				0.002	0.054
19	nasabah	1		0				0.061	0.002
20	pajak	2		1				0.120	0.054
21	perban	1		0				0.061	0.002
22	perintah	0		1				0.002	0.054
23	program	2		0				0.120	0.002
24	sempat	0		1				0.002	0.054
25	siap	0		1				0.002	0.054
26	tenggelam	0		1				0.002	0.054
27	transaksi	1		0				0.061	0.002
28	tunda	0		1				0.002	0.054
29	uang	1		0				0.061	0.002
30	wacana	0		1				0.002	0.054

Lampiran II Peluang setiap kosakata pada dokumen ke-2 terhadap setiap topik

No.	Vocabulary	Jumlah kosakata ke-v pada topik ke-1 (w_{vk})	Jumlah seluruh kosakata pada topik ke-1	Jumlah kosakata ke-v pada topik ke-2 (w_{vk})	Jumlah seluruh kosakata pada topik ke-2	Jumlah kata dalam Vocabulary	Nilai Eta	$p(w_n \mid z_1, \beta)$	$p(w_n \mid z_2, \beta)$
1	aksi	2	16	0	11	23	0.043	0.120	0.004
2	bubar	0		1				0.003	0.087
3	buka	0		1				0.003	0.087
4	dapat	1		0				0.061	0.004
5	dasar	0		1				0.003	0.087
6	edar	1		0				0.061	0.004
7	elektronik	1		0				0.061	0.004
8	ikut	0		2				0.003	0.170
9	islam	1		0				0.061	0.004
10	jabodetabek	1		0				0.061	0.004
11	klaim	1		0				0.061	0.004
12	kumpul	0		1				0.003	0.087
13	organisasi	1		0				0.061	0.004
14	ormas	3		0				0.179	0.004
15	peluang	0		1				0.003	0.087
16	perintah	1		0				0.061	0.004
17	perppu	1		0				0.061	0.004
18	poster	1		0				0.061	0.004
19	sama	0		1				0.003	0.087
20	sampai	0		1				0.003	0.087
21	selebaran	1		0				0.061	0.004
22	terima	0		1				0.003	0.087
23	tolak	0		1				0.003	0.087

Lampiran III Peluang setiap kosakata pada dokumen ke-3 terhadap setiap topik

No.	Vocabulary	Jumlah kosakata ke-v pada topik ke-1 (w_{vk})	Jumlah seluruh kosakata pada topik ke-1	Jumlah kosakata ke-v pada topik ke-2 (w_{vk})	Jumlah seluruh kosakata pada topik ke-2	Jumlah kata dalam Vocabulary	Nilai Eta	$p(w_n \mid z_1, \beta)$	$p(w_n \mid z_2, \beta)$
1	asean	1	12	0	17	26	0.038	0.080	0.002
2	bahas	0		1				0.003	0.058
3	bangsa	1		0				0.080	0.002
4	batas	0		1				0.003	0.058
5	dagang	1		0				0.080	0.002
6	datang	0		1				0.003	0.058
7	dua	0		2				0.003	0.113
8	ekonomi	1		0				0.080	0.002
9	eksklusif	1		0				0.080	0.002
10	harap	0		1				0.003	0.058
11	jumlah	0		1				0.003	0.058
12	lebih	0		1				0.003	0.058
13	legislatif	1		0				0.080	0.002
14	lembaga	1		0				0.080	0.002
15	masa	0		1				0.003	0.058
16	mulai	0		1				0.003	0.058
17	negara	2		0				0.157	0.002
18	phu	0		1				0.003	0.058
19	sahabat	0		1				0.003	0.058
20	singgung	0		1				0.003	0.058

No.	Vocabulary	Jumlah kosakata ke-v pada topik ke-1 (w_{vk})	Jumlah seluruh kosakata pada topik ke-1	Jumlah kosakata ke-v pada topik ke-2 (w_{vk})	Jumlah seluruh kosakata pada topik ke-2	Jumlah kata dalam Vocabulary	Nilai Eta	$p(w_n \mid z_1, \beta)$	$p(w_n \mid z_2, \beta)$
21	temu	0	12	1	17	26	0.038	0.003	0.058
22	tiga	0		1				0.003	0.058
23	tingkat	0		1				0.003	0.058
24	throng	0		1				0.003	0.058
25	zee	2		0				0.157	0.002
26	zona	1		0				0.080	0.002

Lampiran IV perhitungan peluang setiap kata pada dokumen ke-1 dan topik ke-k $p(w, z, \theta)$

No	Kata Pada Dokumen	Peluang Topik ke-1 pada Dokumen ke-1	Peluang Topik ke-2 pada Dokumen ke-1	$p(w_n, z_1, \theta_1)$	$p(w_n, z_2, \theta_1)$	Label Topik
1	jakarta	0.485	0.544	0.00094	0.02959	2
2	direktorat			0.02949	0.00095	1
3	jenderal			0.02949	0.00095	1
4	pajak			0.05803	0.02959	1
5	menteri			0.02949	0.00095	1
6	uang			0.02949	0.00095	1
7	djp			0.02949	0.00095	1
8	kemenkeu			0.02949	0.00095	1
9	minta			0.00094	0.02959	2
10	industri			0.00094	0.02959	2
11	perban			0.02949	0.00095	1
12	siap			0.00094	0.02959	2
13	data			0.00094	0.02959	2
14	transaksi			0.02949	0.00095	1
15	kartu			0.00094	0.02959	2
16	kredit			0.02949	0.00095	1
17	nasabah			0.02949	0.00095	1
18	wacana			0.00094	0.02959	2
19	sempat			0.00094	0.02959	2
20	tenggelam			0.00094	0.02959	2
21	perintah			0.00094	0.02959	2

No	Kata Pada Dokumen	Peluang Topik ke-1 pada Dokumen ke-1	Peluang Topik ke-2 pada Dokumen ke-1	p(w_n, z_1, θ_1)	p(w_n, z_2, θ_1)	Label Topik
22	mulai	0.485	0.544	0.00094	0.02959	2
23	program			0.05803	0.00095	1
24	amnesti			0.05803	0.00095	1
25	pajak			0.05803	0.02959	1
26	juli			0.00094	0.02959	2
27	tunda			0.00094	0.02959	2
28	laku			0.00094	0.02959	2
29	hingga			0.00094	0.02959	2
30	program			0.05803	0.00095	1
31	amnesti			0.05803	0.00095	1
32	pajak			0.05803	0.02959	1
33	akhir			0.00094	0.02959	2
34	maret			0.00094	0.02959	2

Lampiran V perhitungan peluang setiap kata pada dokumen ke-2 dan topik ke-k $p(w, z, \theta)$

No	Kata Pada Dokumen	Peluang Topik ke-1 pada Dokumen ke-2	Peluang Topik ke-2 pada Dokumen ke-2	$p(w_n, z_1, \theta_2)$	$p(w_n, z_2, \theta_2)$	Label Topik
1	dasar	0.611	0.426	0.00155	0.03706	2
2	selebaran			0.03751	0.00153	1
3	poster			0.03751	0.00153	1
4	elektronik			0.03751	0.00153	1
5	edar			0.03751	0.00153	1
6	terima			0.00155	0.03706	2
7	organisasi			0.03751	0.00153	1
8	ikut			0.00155	0.07259	2
9	aksi			0.07348	0.00153	1
10	kumpul			0.00155	0.03706	2
11	sama			0.00155	0.03706	2
12	sampai			0.00155	0.03706	2
13	dapat			0.03751	0.00153	1
14	tolak			0.00155	0.03706	2
15	perppu			0.03751	0.00153	1
16	ormas			0.10944	0.00153	1
17	buka			0.00155	0.03706	2
18	peluang			0.00155	0.03706	2
19	perintah			0.03751	0.00153	1
20	bubar			0.00155	0.03706	2
21	ormas			0.10944	0.00153	1

No	Kata Pada Dokumen	Peluang Topik ke-1 pada Dokumen ke-2	Peluang Topik ke-2 pada Dokumen ke-2	p(w_n, z_1, θ_2)	p(w_n, z_2, θ_2)	Label Topik
22	klaim	0.611	0.426	0.03751	0.00153	1
23	aksi			0.07348	0.00153	1
24	ikut			0.00155	0.07259	2
25	ormas			0.10944	0.00153	1
26	islam			0.03751	0.00153	1
27	jabodetabek			0.03751	0.00153	1

Lampiran VI perhitungan peluang setiap kata pada dokumen ke-3 dan topik ke-k $p(w, z, \theta)$

No	Kata Pada Dokumen	Peluang Topik ke-1 pada Dokumen ke-3	Peluang Topik ke-2 pada Dokumen ke-3	$p(w_n, z_1, \theta_3)$	$p(w_n, z_2, \theta_3)$	Label Topik
1	temu	0.431	0.603	0.00126	0.03480	2
2	tiga			0.00126	0.03480	2
3	lembaga			0.03445	0.00127	1
4	legislatif			0.03445	0.00127	1
5	phu			0.00126	0.03480	2
6	throng			0.00126	0.03480	2
7	bahas			0.00126	0.03480	2
8	jumlah			0.00126	0.03480	2
9	mulai			0.00126	0.03480	2
10	asean			0.03445	0.00127	1
11	zona			0.03445	0.00127	1
12	ekonomi			0.03445	0.00127	1
13	eksklusif			0.03445	0.00127	1
14	zee			0.06763	0.00127	1
15	dagang			0.03445	0.00127	1
16	singgung			0.00126	0.03480	2
17	batas			0.00126	0.03480	2
18	zee			0.06763	0.00127	1
19	dua			0.00126	0.06832	2
20	negara			0.06763	0.00127	1
21	sahabat			0.00126	0.03480	2

No	Kata Pada Dokumen	Peluang Topik ke-1 pada Dokumen ke-3	Peluang Topik ke-2 pada Dokumen ke-3	$p(w_n, z_1, \theta_3)$	$p(w_n, z_2, \theta_3)$	Label Topik
22	dua	0.431	0.603	0.00126	0.06832	2
23	negara			0.06763	0.00127	1
24	bangsa			0.03445	0.00127	1
25	harap			0.00126	0.03480	2
26	lebih			0.00126	0.03480	2
27	tingkat			0.00126	0.03480	2
28	masa			0.00126	0.03480	2
29	datang			0.00126	0.03480	2