

Sistem Ringkasan Dokumen Otomatis Berbasis Word2Vec dan Support Vector Machine

Mochamad Hafid¹, Muhamad Aldi Hawari²

^{1,2}Teknik Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Nusantara PGRI Kediri

E-mail: *¹hafideft1@gmail.com, ²aldihawarii@gmail.com

Abstrak – Perkembangan informasi digital menyebabkan peningkatan jumlah dokumen teks yang harus dipahami secara cepat dan efisien. Hal ini mendorong perlunya sistem ringkasan dokumen otomatis untuk membantu pengguna memperoleh inti informasi tanpa membaca keseluruhan dokumen. Penelitian ini bertujuan untuk membangun sistem ringkasan dokumen otomatis berbasis pembelajaran mesin menggunakan metode Word2Vec dan Support Vector Machine (SVM). Word2Vec digunakan untuk merepresentasikan setiap kalimat ke dalam bentuk vektor numerik yang menangkap hubungan semantik antar kata, sedangkan SVM digunakan untuk mengklasifikasikan kalimat penting dan tidak penting dalam dokumen. Proses peringkasan dilakukan dengan memilih kalimat-kalimat yang diklasifikasikan sebagai kalimat penting untuk membentuk ringkasan akhir. Pengujian sistem dilakukan pada kumpulan dokumen teks dengan membandingkan ringkasan hasil sistem terhadap ringkasan referensi menggunakan metrik evaluasi ROUGE. Hasil pengujian menunjukkan bahwa kombinasi Word2Vec dan SVM mampu menghasilkan ringkasan yang representatif dan relevan terhadap isi dokumen asli.

Kata Kunci — Confusion Matrix, ROUGE, Ringkasan Dokumen, Support Vector Machine, Word2Vec

1. PENDAHULUAN

Pesatnya perkembangan teknologi informasi membuat jumlah dokumen teks digital semakin meningkat, seperti artikel ilmiah, berita, dan berbagai dokumen daring lainnya. Banyaknya informasi tersebut sering kali menyulitkan pembaca untuk menemukan inti pembahasan secara cepat karena keterbatasan waktu dan perhatian. Akibatnya, pembaca harus menghabiskan waktu lebih lama untuk memahami isi dokumen secara keseluruhan. Kondisi ini menunjukkan perlunya sebuah sistem ringkasan dokumen otomatis yang mampu menyajikan informasi penting secara singkat, jelas, dan mudah dipahami tanpa mengurangi makna utama dari dokumen asli.

Namun, membangun sistem ringkasan dokumen otomatis bukanlah hal yang sederhana. Tantangan utama terletak pada bagaimana memahami konteks dan makna setiap kalimat dalam dokumen serta menentukan kalimat mana yang paling penting untuk dimasukkan ke dalam ringkasan. Pendekatan konvensional yang hanya mengandalkan frekuensi kemunculan kata sering kali belum mampu menangkap hubungan makna antar kata secara mendalam[1]. Oleh karena itu, penelitian ini memfokuskan permasalahan pada penerapan metode yang mampu merepresentasikan makna semantik kalimat dengan lebih baik dan mengklasifikasikan tingkat kepentingan kalimat secara akurat menggunakan Word2Vec dan Support Vector Machine (SVM).

Penelitian ini bertujuan untuk merancang dan mengembangkan sistem ringkasan dokumen otomatis berbasis Word2Vec dan SVM yang dapat menghasilkan ringkasan dokumen yang informatif dan relevan. Word2Vec digunakan untuk menangkap hubungan semantik antar kata dalam bentuk vektor, sedangkan SVM berperan dalam menentukan kalimat yang dianggap penting untuk membentuk ringkasan akhir. Kontribusi dari penelitian ini diharapkan dapat memberikan gambaran penerapan kombinasi Word2Vec dan SVM dalam peringkasan dokumen otomatis serta menjadi referensi bagi penelitian selanjutnya di bidang *Natural Language Processing*[2].

2. METODE PENELITIAN

2.1 Studi Literatur dan Analisis

Tahap studi literatur dilakukan untuk memperoleh pemahaman konseptual mengenai *Natural Language Processing* (NLP), khususnya pada teknik peringkasan dokumen otomatis (*automatic document summarization*). Peneliti mengkaji berbagai penelitian terdahulu yang membahas pendekatan peringkasan ekstraktif berbasis pembelajaran mesin, termasuk penggunaan Word2Vec sebagai metode representasi semantik dan Support Vector Machine (SVM) sebagai algoritma klasifikasi. Selain itu, dilakukan analisis terhadap struktur dokumen teks untuk mengidentifikasi karakteristik kalimat penting yang umumnya muncul dalam sebuah ringkasan, seperti posisi kalimat, panjang kalimat, dan kandungan informasi utama. Studi

literatur ini juga mencakup kajian terhadap metrik evaluasi ROUGE yang umum digunakan untuk mengukur kualitas ringkasan otomatis secara kuantitatif.

Selain menggunakan metrik ROUGE sebagai evaluasi kualitas ringkasan pada level dokumen, penelitian ini juga mempertimbangkan penggunaan *Confusion Matrix* sebagai bentuk evaluasi tambahan pada level kalimat. *Confusion Matrix* digunakan untuk mengukur kinerja model *Support Vector Machine* (SVM) dalam mengklasifikasikan kalimat penting dan tidak penting sebelum proses peringkasan dilakukan. Evaluasi ini memberikan gambaran yang lebih komprehensif mengenai kemampuan model dalam tahap seleksi kalimat, yang menjadi fondasi utama dalam pembentukan ringkasan dokumen ekstraktif.

a. *Natural Language Processing* (NLP)

Natural Language Processing (NLP) adalah cabang dari kecerdasan buatan (*Artificial Intelligence*) yang berfokus pada interaksi antara komputer dan bahasa alami manusia. Tujuan utama NLP adalah memungkinkan mesin untuk membaca, memahami, dan memproses bahasa manusia dengan cara yang bernilai dan bermakna. Secara teknis, NLP mencakup berbagai tugas mulai dari pemrosesan teks dasar seperti tokenization dan stemming, hingga tugas kompleks seperti penerjemahan mesin, analisis sentimen, dan peringkasan teks otomatis.

Dalam perkembangannya, NLP sering dikaitkan dengan pemrosesan pada 10 hingga 20 bahasa utama yang memiliki sumber daya tinggi (*high-resource languages*), seperti Bahasa Inggris atau Mandarin. Munculnya sistem pembelajaran mendalam (*deep learning*) yang membutuhkan data dalam jumlah besar telah secara signifikan meningkatkan kinerja NLP untuk bahasa-bahasa tersebut. Namun, tantangan besar tetap ada pada bahasa dengan sumber daya rendah (*low-resource languages*), di mana ketersediaan dataset berskala besar sangat terbatas. Hal ini menjadikan pengembangan model NLP yang efisien untuk bahasa tersebut sebagai masalah penelitian yang menantang namun krusial untuk menjembatani kesenjangan teknologi bahasa [3].

b. *Support Vector Machine* (SVM)

Support Vector Machine (SVM) merupakan salah satu algoritma supervised learning (pembelajaran terawasi) yang paling populer dan kuat, digunakan secara luas untuk tugas klasifikasi, regresi, hingga deteksi outlier. SVM pertama kali diperkenalkan oleh Cortes dan Vapnik di AT&T Bell Laboratories dan didasarkan pada kerangka kerja teori pembelajaran statistik *Vapnik–Chervonenkis* (VC Theory). Prinsip kerja utama SVM adalah menemukan *hyperplane* (bidang pemisah) terbaik yang memisahkan dua kelas data dengan margin maksimal.

Dalam kasus data yang dapat dipisahkan secara linier, SVM mencari *hyperplane* dengan margin keras atau lunak untuk meminimalkan kesalahan klasifikasi. Namun, kekuatan utama SVM terletak pada kemampuannya menangani data non-linier menggunakan teknik yang disebut *Kernel Trick*. Fungsi kernel memungkinkan algoritma untuk memetakan data dari ruang dimensi rendah ke ruang dimensi yang lebih tinggi di mana data tersebut dapat dipisahkan secara linier. Beberapa fungsi kernel yang umum digunakan meliputi kernel Linier, Polinomial, Radial Basis Function (RBF), dan Sigmoid. Fleksibilitas ini menjadikan SVM sangat efektif dalam berbagai aplikasi, termasuk kategorisasi teks dan klasifikasi citra [4].

c. Word2Vec

Word2Vec adalah teknik pemrosesan bahasa alami yang dikembangkan oleh tim peneliti Google, dipimpin oleh Tomas Mikolov, untuk menghasilkan *word embedding* (representasi vektor kata). Berbeda dengan metode tradisional seperti *Bag of Words* (BoW) atau TF-IDF yang hanya mengandalkan statistik frekuensi kemunculan kata, *Word2Vec* merupakan model prediktif berbasis jaringan saraf (*neural network*). Model ini mampu menangkap makna semantik dan hubungan sintaksis antar kata dengan memetakan kata-kata ke dalam ruang vektor, di mana kata-kata dengan makna serupa akan memiliki jarak vektor yang berdekatan [5].

1. *Continuous Bag of Words* (CBOW)

Model ini memprediksi kata target berdasarkan konteks kata-kata di sekitarnya. CBOW cenderung lebih cepat dilatih dan memiliki akurasi yang baik untuk kata-kata yang sering muncul.

2. *Skip-Gram*

Kebalikan dari CBOW, model ini menggunakan kata target untuk memprediksi kata-kata konteks di sekitarnya. *Skip-Gram* bekerja lebih baik untuk dataset kecil dan lebih mampu merepresentasikan kata-kata yang jarang muncul. Kedua metode ini menjadikan *Word2Vec* sangat efektif dalam menangkap konteks linguistik yang kompleks.

d. *Confusion Matrix*

Karl Pearson menciptakan *Confusion Matrix* pada tahun 1904, yang awalnya dikenal sebagai tabel kontingensi. Kemudian disebut sebagai matriks klasifikasi, sebelum akhirnya disebut sebagai *Confusion Matrix* dalam ilmu data. Matriks ini terdiri dari empat komponen utama: *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Dengan menganalisis keempat komponen ini, peneliti tidak hanya mengetahui akurasi model, tetapi juga dapat menghitung metrik penting lainnya seperti *Precision*, *Recall*, dan *F1-Score*. Penggunaan istilah *Confusion Matrix* sangat membantu dalam mengurangi kebingungan interpretasi hasil, karena memberikan gambaran rinci tentang di mana model melakukan kesalahan klasifikasi, apakah itu salah memprediksi kelas positif sebagai negatif (Tipe II error) atau sebaliknya (Tipe I error) [6].

e. *Recall-Oriented Understudy for Gisting Evaluation* (ROUGE)

Recall-Oriented Understudy for Gisting Evaluation atau ROUGE adalah sekumpulan metrik yang digunakan untuk mengevaluasi kualitas sistem peringkasan teks otomatis dan penerjemahan mesin. ROUGE bekerja dengan cara menghitung tumpang tindih (*overlap*) unit n-gram (seperti kata atau frasa) antara ringkasan yang dihasilkan oleh sistem (*system summary*) dengan satu atau lebih ringkasan referensi yang dibuat oleh manusia (*gold standard*). Metrik ini sangat populer karena kesederhanaannya dan efektivitasnya dalam mengukur seberapa banyak informasi relevan yang berhasil ditangkap oleh sistem [7].

2.2 Desain Sistem dan *Preprocessing*

Desain sistem peringkasan dokumen dibangun dengan alur kerja yang sistematis, dimulai dari pengolahan teks mentah hingga pembentukan ringkasan akhir. Pada tahap *preprocessing*, teks asli terlebih dahulu diproses melalui tokenisasi, penghapusan *stopwords*, serta normalisasi teks untuk mengurangi noise pada data. Setelah itu, dokumen dipecah menjadi kalimat-kalimat sebagai unit analisis utama. Setiap kalimat kemudian disiapkan untuk tahap ekstraksi fitur dengan memperhatikan karakteristik tambahan, seperti panjang kalimat dan posisi kalimat dalam dokumen. Tahapan *preprocessing* ini bertujuan untuk meningkatkan kualitas representasi data sebelum diproses oleh model Word2Vec dan SVM.

Tahap awal meliputi ekstraksi konten dari dokumen PDF, *text cleaning*, dan identifikasi bagian-bagian jurnal yang paling informatif seperti metode, hasil, pembahasan, dan serta simpulan/kesimpulan. Tahapan dalam pra-pemrosesan data dalam penelitian ini diantaranya :

a. *Tokenization*

Tokenisasi adalah proses membagi kalimat atau teks menjadi bagian-bagian yang lebih kecil, biasa disebut term atau token. Pembagian ini dilakukan berdasarkan spasi kata. Proses tokenisasi dalam penelitian ini dilakukan menggunakan fungsi *word_tokenize()* dari *library Natural Language Toolkit* (NLTK). Fungsi ini akan memecah kalimat menjadi kata berdasarkan pemisah seperti spasi dan tanda baca[8].

b. *Stopwords Removal*

Stopword Removal merupakan kata-kata yang memiliki makna umum, tidak membawa informasi penting, dan tidak mengandung sentimen, namun sering muncul dalam jumlah besar pada teks. Proses *stopword removal* dilakukan untuk mengurangi dimensionalitas data, menurunkan beban komputasi, serta meningkatkan efisiensi dan kinerja analisis[8].

2.3 Implementasi *Modelling* Word2Vec dan SVM

Pada tahap pemodelan, Word2Vec digunakan untuk membangun representasi vektor kata berdasarkan konteks kemunculannya dalam korpus teks. Model Word2Vec dilatih menghasilkan *embedding* kata yang mampu menangkap hubungan semantik antar kata. Selanjutnya, setiap kalimat direpresentasikan dalam bentuk vektor kalimat dengan cara mengagregasikan vektor kata penyusunnya, misalnya melalui perhitungan rata-rata vektor. Vektor kalimat ini kemudian dikombinasikan dengan fitur tambahan seperti panjang dan posisi kalimat untuk membentuk *feature vector*[9].

$$p(w_j|w_i) = \left[\frac{\exp(v'_{w_j} T_{v_{w_i}})}{\sum_{j'=1}^V \exp(v'_{w_{j'}} T_{v_{w_i}})} \right] \dots \dots \dots (1)$$

Pada rumus diatas, v_w dan v'_w adalah dua representasi dari kata $w \cdot v_w$. Berasal dari baris W , yang merupakan matriks bobot input $\rightarrow$ *hidden weight matrix*, dan v'_w berasal dari kolom W' , yang merupakan *hidden matrix* $\rightarrow$ *Output Matrix*. Dalam analisis selanjutnya, kita menyebut v_w sebagai sebuah vektor input, dan v'_w sebagai sebuah vektor output dari kata w [10]

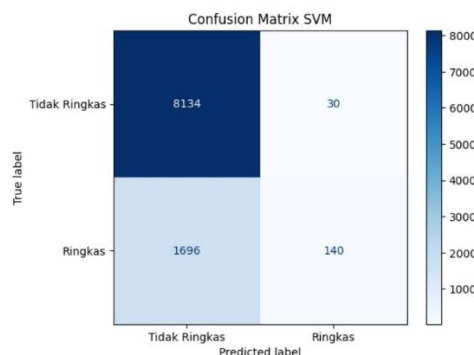
Feature vector tersebut digunakan sebagai masukan bagi algoritma *Support Vector Machine* (SVM) untuk melakukan proses klasifikasi. Model SVM dilatih menggunakan data berlabel untuk membedakan antara kalimat penting dan tidak penting. Setelah proses pelatihan selesai, model SVM digunakan untuk memprediksi tingkat kepentingan kalimat pada dokumen baru. Kalimat dengan skor probabilitas tertinggi kemudian dipilih sebagai ringkasan dokumen otomatis.

$$\min \frac{1}{2} \|\omega\|^2 + \sum_{i=1}^n a_i^* y_i (x_i \cdot x) + b^* \dots (2)$$

a_i *Lagrange multiplier* yang sesuai dengan setiap sampel, dan dalam solusi, $a_i^* \neq 0$ sesuai dengan sampel pelatihan yang merupakan SV. Biasanya, hanya beberapa pengali Lagrange yang bukan nol. b^* adalah ambang batas klasifikasi, yang dapat dihitung dari SV apa pun atau median dari dua SV apa pun dari kedua kelas tersebut[11].

2.4 Evaluasi dan Analisis Hasil

Tahap evaluasi dilakukan dengan membandingkan ringkasan yang dihasilkan oleh sistem terhadap ringkasan acuan menggunakan metrik ROUGE, yang meliputi ROUGE-1, ROUGE-2, dan ROUGE-L. Evaluasi ini bertujuan untuk mengukur tingkat kesesuaian antara ringkasan sistem dan ringkasan referensi berdasarkan kesamaan unigram, bigram, dan urutan kata. Analisis hasil dilakukan untuk menilai efektivitas kombinasi Word2Vec dan SVM dalam mengidentifikasi kalimat penting, serta untuk mengamati pengaruh fitur-fitur tambahan terhadap kualitas ringkasan. Hasil evaluasi ini menjadi dasar dalam menilai performa sistem dan potensi pengembangannya di masa mendatang. Gambar 1 adalah hasil evaluasi dari *Confusion Matrix*.



Gambar 1 Evaluasi *Confusion Matrix*

3. HASIL DAN PEMBAHASAN

Pengujian sistem dilakukan dengan membandingkan ringkasan yang dihasilkan oleh sistem terhadap abstrak asli dari 11 dokumen jurnal ilmiah sebagai ringkasan acuan. Abstrak dipilih karena merepresentasikan ringkasan manual yang disusun oleh penulis dokumen dan dianggap sebagai standar referensi. Proses evaluasi bertujuan untuk mengukur sejauh mana sistem mampu menangkap informasi inti dari dokumen secara otomatis.

Evaluasi dilakukan menggunakan metrik ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*), yang mengukur tingkat kesamaan antara ringkasan otomatis dan ringkasan referensi berdasarkan kesamaan n-gram. Dalam penelitian ini digunakan tiga varian ROUGE, yaitu ROUGE-1 untuk kesamaan unigram, ROUGE-2 untuk kesamaan bigram, dan ROUGE-L untuk mengukur kesamaan urutan kata terpanjang (Longest Common Subsequence). Ketiga metrik tersebut digunakan untuk memberikan gambaran yang komprehensif terhadap kualitas ringkasan yang dihasilkan sistem.

3.1 Hasil Evaluasi Kuantitatif

Pengujian dilakukan terhadap 11 dokumen jurnal berbahasa Indonesia. Ringkasan yang dihasilkan oleh sistem (system summary) dibandingkan dengan abstrak asli dari masing-masing dokumen (reference summary). Evaluasi kuantitatif dilakukan dengan menghitung skor F1 untuk ROUGE-1, ROUGE-2, dan ROUGE-L pada setiap dokumen, kemudian dirata-ratakan untuk memperoleh gambaran performa sistem secara keseluruhan.

Selain menyajikan nilai rata-rata ROUGE, penelitian ini juga menganalisis distribusi skor ROUGE-1 untuk memahami variasi performa sistem pada setiap dokumen uji. Distribusi ini direpresentasikan melalui nilai ROUGE-1 Minimum (R1-Min) dan ROUGE-1 Maksimum (R1-Max). Analisis distribusi ini bertujuan untuk memberikan gambaran yang lebih komprehensif mengenai stabilitas dan konsistensi sistem ringkasan dokumen otomatis berbasis Word2Vec dan *Support Vector Machine*.

Sebaliknya, nilai R1-Min yang mencapai 0.0296 menunjukkan adanya dokumen yang gagal diringkas secara efektif. Hasil analisis menunjukkan bahwa nilai minimum ini bukan disebabkan oleh kelemahan metode Word2Vec atau SVM secara langsung, melainkan oleh kendala pada tahap preprocessing, khususnya kegagalan ekstraksi teks dari dokumen PDF yang memiliki struktur tidak terstandarisasi. Pada kondisi tersebut, sistem tidak memperoleh input kalimat yang memadai untuk diproses lebih lanjut, sehingga tidak mampu menghasilkan ringkasan yang relevan.

Tabel 1. ROUGE Score (Average F1)

ROUGE-1	ROUGE-2	ROUGE-L
0.4070	0.02960	0.3511

Berdasarkan hasil pengujian dalam tabel 1 diatas, diperoleh nilai rata-rata ROUGE-1 sebesar 0.4070, ROUGE-2 sebesar 0.2960, dan ROUGE-L sebesar 0.3511. Skor ROUGE-1 yang relatif lebih tinggi menunjukkan bahwa sistem cukup mampu menangkap informasi inti dokumen secara leksikal pada tingkat unigram. Sementara itu, nilai ROUGE-2 yang lebih rendah mengindikasikan bahwa kesesuaian pasangan kata (bigram) antara ringkasan sistem dan referensi masih menjadi tantangan, yang umum terjadi pada metode peringkasan ekstraktif berbasis klasifikasi kalimat. Nilai ROUGE-L yang cukup stabil menunjukkan bahwa sistem mampu mempertahankan kesamaan urutan kata terpanjang (longest common subsequence), sehingga ringkasan yang dihasilkan tetap memiliki alur informasi yang koheren.

Tabel 2. ROUGE Distribution

Id	R1	R2	RL
Min.	0.0296	0.0000	0.7869
Max.	0.0296	0.7333	0.7869

Selain nilai rata-rata, analisis distribusi skor ROUGE pada tabel 2 dilakukan untuk memahami variasi performa sistem pada setiap dokumen uji. Distribusi skor menunjukkan bahwa ROUGE-1 memiliki nilai minimum sebesar 0.0296 dan maksimum sebesar 0.7869, ROUGE-2 memiliki rentang dari 0.0000 hingga 0.7333, serta ROUGE-L berkisar antara 0.0296 hingga 0.7815. Rentang nilai yang cukup lebar ini menunjukkan adanya perbedaan signifikan dalam kualitas ringkasan antar dokumen.

3.2 Analisis Efektivitas Metode

Keberhasilan sistem dalam menghasilkan ringkasan dokumen sangat dipengaruhi oleh kombinasi metode Word2Vec dan Support Vector Machine (SVM). Word2Vec digunakan untuk merepresentasikan setiap kalimat ke dalam bentuk vektor numerik yang menangkap hubungan semantik antar kata. Representasi ini memungkinkan sistem untuk memahami konteks kalimat secara lebih baik dibandingkan pendekatan berbasis frekuensi kata semata.

Vektor kalimat yang dihasilkan kemudian digunakan sebagai fitur input bagi model SVM untuk melakukan klasifikasi tingkat kepentingan kalimat. SVM dilatih menggunakan data berlabel (gold labels) untuk membedakan antara kalimat penting dan tidak penting dalam dokumen. Melalui proses ini, sistem mampu memberikan skor probabilitas kepentingan pada setiap kalimat menggunakan fungsi `predict_proba()`, sehingga pemilihan kalimat ringkasan dapat dilakukan secara objektif berdasarkan hasil klasifikasi.

Hasil pengujian menunjukkan bahwa pendekatan klasifikasi kalimat menggunakan SVM cukup efektif dalam memilih kalimat-kalimat yang merepresentasikan inti dokumen. Kalimat dengan skor kepentingan tertinggi umumnya memuat informasi utama seperti tujuan penelitian, hasil eksperimen, dan simpulan, yang selaras dengan isi abstrak referensi.

3.3 Analisis Kualitas Ringkasan yang Dihasilkan

Ringkasan akhir dibentuk dengan memilih tiga kalimat dengan skor kepentingan tertinggi dari hasil prediksi SVM. Pendekatan ini menghasilkan ringkasan yang bersifat ekstraktif dan padat, tanpa mengubah struktur asli kalimat dalam dokumen. Secara umum, ringkasan yang dihasilkan mampu menyampaikan gagasan utama dokumen meskipun jumlah kalimat yang digunakan terbatas.

Capaian skor ROUGE-L yang relatif stabil menunjukkan bahwa sistem tidak hanya mampu memilih kata-kata yang relevan, tetapi juga mempertahankan kesesuaian urutan kata dengan ringkasan referensi. Hal ini mengindikasikan bahwa representasi semantik Word2Vec berperan penting dalam membantu sistem mengenali kesamaan makna antar kalimat, meskipun tidak selalu menggunakan kata yang persis sama.

3.4 Keterbatasan Sistem

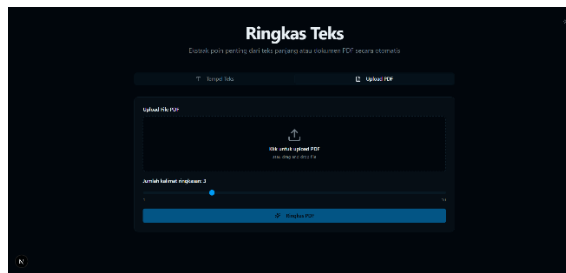
Meskipun sistem menunjukkan performa yang cukup baik, terdapat beberapa keterbatasan yang perlu diperhatikan. Ketergantungan sistem terhadap kualitas teks hasil ekstraksi menjadi faktor utama yang memengaruhi performa. Dokumen PDF dengan format kolom kompleks, tabel, atau encoding karakter yang

tidak standar dapat menyebabkan kegagalan ekstraksi teks, sehingga berdampak langsung pada kualitas ringkasan.

Selain itu, pendekatan ekstraktif yang digunakan masih bergantung pada pemilihan kalimat dari dokumen asli, sehingga sistem belum mampu menghasilkan ringkasan dalam bentuk parafrase atau kalimat baru. Oleh karena itu, pengembangan lanjutan dapat diarahkan pada peningkatan modul ekstraksi teks serta eksplorasi metode peringkasan yang lebih adaptif.

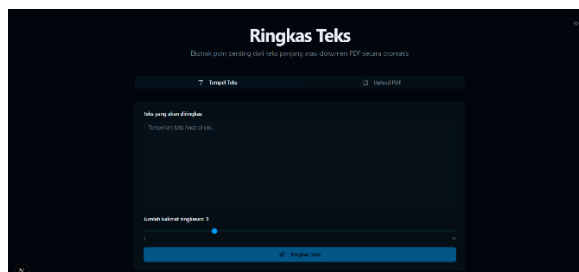
3.5 Desain Menu

Implementasi antarmuka pengguna (*user interface*) pada aplikasi ini dirancang dengan mengedepankan kemudahan akses (*usability*) dan kenyamanan visual. Aplikasi menerapkan tema gelap (*dark mode*) yang modern untuk mengurangi kelelahan mata pengguna saat membaca teks atau hasil ringkasan. Pada halaman utama, pengguna disuguhkan dua opsi metode input data untuk diringkaskan, yaitu melalui unggahan berkas dokumen atau input teks manual. Navigasi antara kedua metode ini dilakukan melalui *tab menu* yang terletak di bagian atas panel utama.



Gambar 2. Halaman peringkasan teks yang menggunakan *Upload File*

Selain fitur unggah berkas, aplikasi juga menyediakan fitur untuk memproses teks secara langsung. Fitur pertama adalah peringkasan berbasis dokumen yang diilustrasikan pada Gambar 2. Pada mode ini, tab '*Upload PDF*' diaktifkan. Pengguna disediakan area khusus untuk mengunggah berkas dengan metode *drag-and-drop* atau dengan mengklik area tersebut untuk membuka *file explorer*. Antarmuka ini dirancang khusus untuk menerima format PDF sebagai masukan. Di bagian bawah area unggah, terdapat komponen *slider* (penggeser) interaktif yang memungkinkan pengguna untuk menentukan parameter jumlah kalimat hasil ringkasan (misalnya 3 kalimat, 5 kalimat, dst.) sesuai kebutuhan. Setelah berkas dipilih dan parameter diatur, proses eksekusi dilakukan dengan menekan tombol biru bertuliskan "Ringkas PDF".



Gambar 3. Halaman peringkasan teks yang menggunakan Teks secara langsung

Alternatif kedua adalah fitur peringkasan teks secara langsung (*direct text input*). Sebagaimana ditunjukkan pada Gambar 3, tampilan ini dapat diakses dengan memilih tab 'Tempel Teks'. Pada halaman ini, elemen *uploader* digantikan oleh kolom teks area (*text area*) yang luas. Pengguna dapat menyalin teks dari sumber lain dan menempelkannya (*paste*) atau mengetik artikel berita secara manual ke dalam kolom tersebut. Konsistensi desain tetap dijaga dengan menyertakan fitur *slider* pengatur jumlah kalimat di bagian bawah, serta tombol eksekusi "Ringkas Teks" untuk memulai proses komputasi algoritma peringkasan.

4. SIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa sistem ringkasan dokumen otomatis berbasis Word2Vec dan *Support Vector Machine* (SVM) mampu menghasilkan ringkasan yang cukup representatif terhadap isi dokumen jurnal berbahasa Indonesia. Penggunaan Word2Vec sebagai metode pembentukan vektor memungkinkan sistem menangkap hubungan semantik antar kata dan kalimat, sementara SVM berperan efektif dalam mengklasifikasikan tingkat kepentingan kalimat. Hasil evaluasi menggunakan metrik ROUGE menunjukkan bahwa sistem mampu menangkap informasi inti dokumen, khususnya pada dokumen dengan struktur teks yang baik dan proses ekstraksi yang optimal.

Selain itu, analisis distribusi skor ROUGE, khususnya ROUGE-1 dan ROUGE *Distribution* (Min dan Max), menunjukkan adanya variasi performa antar dokumen uji. Dokumen dengan format PDF yang terstruktur menghasilkan skor ROUGE yang tinggi, sedangkan dokumen dengan format kompleks mengalami penurunan performa akibat kegagalan preprocessing. Temuan ini menegaskan bahwa kualitas ringkasan tidak hanya ditentukan oleh model Word2Vec dan SVM, tetapi juga sangat dipengaruhi oleh kualitas dan keterbacaan data teks yang diproses oleh sistem.

5. SARAN

Berdasarkan keterbatasan yang ditemukan dalam penelitian ini, penelitian selanjutnya disarankan untuk mengembangkan sistem peringkasan dokumen dengan mengintegrasikan model IndoBERT secara lebih mendalam pada tahap representasi semantik kalimat. Penggunaan IndoBERT tidak hanya sebagai pembanding, tetapi sebagai core embedding model, diharapkan mampu menangkap konteks semantik dan relasi antar kalimat secara lebih akurat dibandingkan Word2Vec yang bersifat statis. Integrasi embedding kontekstual IndoBERT berpotensi meningkatkan kemampuan sistem dalam membedakan kalimat penting dan tidak penting, terutama pada dokumen jurnal yang memiliki struktur bahasa formal dan kompleks.

Selain itu, penelitian lanjutan disarankan untuk menggabungkan embedding IndoBERT dengan pendekatan klasifikasi yang lebih adaptif, seperti fine-tuning IndoBERT untuk tugas klasifikasi kalimat atau menggabungkannya dengan SVM sebagai hybrid classifier. Pendekatan ini memungkinkan pemanfaatan kekuatan representasi kontekstual IndoBERT sekaligus mempertahankan keunggulan SVM dalam klasifikasi berbasis margin. Evaluasi sistem juga dapat diperluas dengan menggabungkan metrik ROUGE dan confusion matrix dengan human evaluation oleh pakar, sehingga kualitas ringkasan tidak hanya dinilai dari kesamaan leksikal, tetapi juga dari aspek keterbacaan, koherensi, dan relevansi isi secara semantik.

DAFTAR PUSTAKA

- [1] K. Kurniawan and S. Louvan, “IndoSum: A New Benchmark Dataset for Indonesian Text Summarization,” *Proceedings of the 2018 International Conference on Asian Language Processing, IALP 2018*, pp. 215–220, Oct. 2018, doi: 10.1109/IALP.2018.8629109.
- [2] N. J. Perdana, V. C. Marwadi, and N. Andrew, “Implementasi Metode Word2Vec Dan TextRank Dalam Aplikasi Mobile Peringkat Berita Olahraga,” *Jurnal Ilmu Komputer dan Sistem Informasi*, vol. 12, no. 2, Jul. 2024, doi: 10.24912/JIKSI.V12I2.31561.
- [3] M. A. Hedderich, L. Lange, H. Adel, J. Strötgen, and D. Klakow, “A Survey on Recent Approaches for Natural Language Processing in Low-Resource Scenarios,” *NAACL-HLT 2021 - 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, pp. 2545–2568, 2021, doi: 10.18653/V1/2021.NAACL-MAIN.201.
- [4] A. Cemiloglu *et al.*, “Support Vector Machine (SVM) Application for Uniaxial Compression Strength (UCS) Prediction: A Case Study for Maragheh Limestone,” *Applied Sciences 2023, Vol. 13, Page 2217*, vol. 13, no. 4, p. 2217, Feb. 2023, doi: 10.3390/APP13042217.
- [5] M. Mars, “From Word Embeddings to Pre-Trained Language Models: A State-of-the-Art Walkthrough,” *Applied Sciences 2022, Vol. 12, Page 8805*, vol. 12, no. 17, p. 8805, Sep. 2022, doi: 10.3390/APP12178805.
- [6] S. Sathyanarayanan, “Confusion Matrix-Based Performance Evaluation Metrics,” *African Journal of Biomedical Research*, pp. 4023–4031, Nov. 2024, doi: 10.53555/AJBR.V27I4S.4345.
- [7] A. Al Foysal and R. Böck, “Who Needs External References?—Text Summarization Evaluation Using Original Documents,” *AI 2023, Vol. 4, Pages 970-995*, vol. 4, no. 4, pp. 970–995, Nov. 2023, doi: 10.3390/AI4040049.
- [8] P. Ayuningtiyas, K. D. Tania, and W. K. Sari, “Sentiment-Based Knowledge Discovery pada Aplikasi iPusnas Menggunakan Metode Machine Learning dan Deep Learning,” *Journal of Applied Informatics and Computing*, vol. 9, no. 5, pp. 2486–2497, Oct. 2025, doi: 10.30871/JAIC.V9I5.10258.

- [9] R. W. Syaputri, R. P. Sari, and K. W. Mahawuni, “Smart Health : Sistem Rekomendasi Obat Berdasarkan Gejala dengan Pendekatan Word2Vec,” *Seminar Nasional Teknologi & Sains*, vol. 4, no. 1, pp. 136–141, Jan. 2025, doi: 10.29407/7XVMTB60.
- [10] X. Rong, “word2vec Parameter Learning Explained,” Nov. 2014, Accessed: Dec. 19, 2025. [Online]. Available: <https://arxiv.org/pdf/1411.2738>
- [11] J. Li, J. Wang, and E. Xue, “Applying a Support Vector Machine (SVM-RFE) Learning Approach to Investigate Students’ Scientific Literacy Development: Evidence from Asia, Europe, and South America,” *Journal of Intelligence 2024, Vol. 12, Page 111*, vol. 12, no. 11, p. 111, Nov. 2024, doi: 10.3390/JINTELLIGENCE12110111.