



Ringkasan Otomatis Dokumen Berita Menggunakan Metode Automatic Text Summarization Berbasis Transformer

Novia Ramadani^{1*}, Mahardika Abdi Prawira Tanjung²

^{1*,2} Program Studi Sistem Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara, Medan, Indonesia

Email: ^{1*}noviaramadhani326@gmail.com, ²mahardikaabdiprawira@umsu.ac.id

Abstrak

Perkembangan teknologi informasi menyebabkan jumlah dokumen berita yang tersedia di media digital semakin meningkat. Kondisi tersebut membuat pembaca membutuhkan waktu yang cukup lama untuk memahami keseluruhan isi berita, terutama karena dokumen berita umumnya memiliki teks yang panjang dan terdiri dari banyak paragraf. Proses peringkasan secara manual juga kurang efektif apabila jumlah berita yang harus dibaca cukup banyak. Selain itu, metode peringkasan yang hanya mempertimbangkan kata atau kalimat tertentu dapat menyebabkan informasi penting terlewat serta menghasilkan ringkasan yang kurang sesuai dengan konteks berita. Penelitian ini bertujuan untuk membangun sistem *Automatic Text Summarization* yang dapat menghasilkan ringkasan dokumen berita secara otomatis dengan mempertahankan informasi utama dari teks asli. Metode yang digunakan adalah pendekatan berbasis *Transformer* yang memiliki kemampuan dalam memahami hubungan antar kata, kalimat, dan konteks dalam suatu dokumen. Tahapan penelitian meliputi pengumpulan data berita, *preprocessing* teks, penerapan model *Transformer*, proses peringkasan, serta evaluasi terhadap hasil ringkasan menggunakan metrik ROUGE dan BLEU. Hasil pengujian menunjukkan nilai ROUGE-1 F1 sebesar 43,48%, ROUGE-2 F1 sebesar 41,79%, ROUGE-L F1 sebesar 43,48%, serta BLEU-4 sebesar 7,43%. Sistem mampu mereduksi dokumen berita sebesar 71,38% dari panjang teks asli. Dengan demikian, penerapan *Automatic Text Summarization* berbasis *Transformer* dapat membantu pembaca memperoleh inti informasi berita secara lebih cepat dan efisien tanpa harus membaca keseluruhan dokumen.

Kata Kunci: *Automatic Text Summarization*, *Transformer*, Ringkasan Otomatis, Dokumen Berita, Pemrosesan Bahasa Alami.

PENDAHULUAN

Informasi dalam bentuk teks merupakan salah satu sumber utama dalam kehidupan sehari-hari, terutama melalui media digital seperti berita *online*. Saat ini, jumlah berita yang dipublikasikan setiap hari semakin banyak sehingga pembaca sering kesulitan membaca semua informasi yang tersedia, terutama jika waktu yang dimiliki terbatas (Maylawati et al., 2023). Dokumen berita biasanya memiliki isi yang panjang dan terdiri dari banyak paragraf; walaupun informasi yang disampaikan lengkap, tidak semua bagian dari berita tersebut penting untuk dibaca secara keseluruhan, sehingga pembaca membutuhkan waktu lebih lama untuk menemukan inti dari berita tersebut.

Tantangan utama dalam memahami isi berita meliputi: (1) banyaknya jumlah berita yang terus bertambah setiap hari; (2) panjangnya teks berita yang membutuhkan waktu lama untuk dibaca; (3) ketidakmampuan metode konvensional dalam memahami isi teks secara mendalam, sehingga ringkasan yang dihasilkan belum optimal; dan (4) potensi hilangnya informasi penting apabila peringkasan hanya bertumpu pada kata atau frasa tertentu tanpa mempertimbangkan makna dan konteks (Itsnaini et al., 2023).

Automatic Text Summarization (ATS) merupakan teknik dalam pengolahan bahasa alami (*Natural Language Processing*/NLP) yang digunakan untuk menghasilkan ringkasan dari suatu teks secara otomatis dengan tetap mempertahankan informasi penting (Dewi & Widiastuti, 2022). Sistem ATS dapat mengalami keterbatasan dalam memahami konteks dan hubungan antar kalimat secara menyeluruh apabila hanya berbasis fitur statistik. Hal ini disebabkan setiap kalimat dalam suatu dokumen saling berkaitan dan membentuk alur informasi; jika hubungan tersebut tidak diperhatikan, maka ringkasan yang dihasilkan berpotensi tidak runtut, kurang jelas, atau kehilangan informasi penting.

Penelitian-penelitian terdahulu menunjukkan adanya perkembangan metode ATS yang signifikan. Yanuarti dan Azizah (2022) mengimplementasikan *extractive summarization* menggunakan Gensim dan

NLTK dengan akurasi 72,6%, namun hasil ringkasan dinilai kurang natural karena tidak menghasilkan kalimat baru. Itsnaini et al. (2023) menerapkan model T5 untuk ATS berbahasa Indonesia dan memperoleh nilai ROUGE-1 = 0,68, ROUGE-2 = 0,61, dan ROUGE-L = 0,65 yang cukup tinggi, meskipun abstraksi hasil ringkasan belum sepenuhnya optimal. Bahari dan Dewi (2024) menguji performa T5 berbasis *Transformer* pada dataset IndoSum dan menunjukkan bahwa metode ini mampu mengatasi kelemahan RNN dengan nilai ROUGE-1 = 0,61 dan ROUGE-2 = 0,51. Winarko et al. (2025) mengembangkan model abstraktif dengan kombinasi *stacked embeddings* (BERT, BPE, FastText) dan *Transformer decoder* untuk bahasa Indonesia dengan performa tinggi, meskipun membutuhkan sumber daya komputasi yang besar. Agung et al. (2025) membandingkan IndoT5 dan mBART dalam menghasilkan *highlight* berita Indonesia dan menemukan bahwa IndoT5 mencapai performa terbaik berdasarkan ROUGE dan BERTScore.

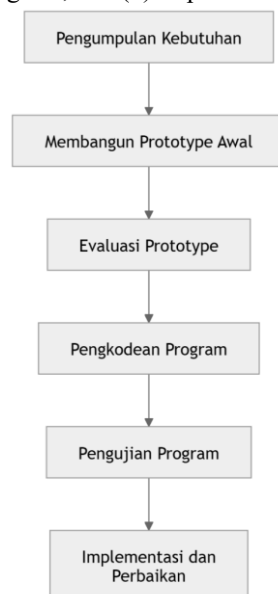
Berdasarkan gap penelitian tersebut, diketahui bahwa pendekatan *Transformer* untuk bahasa Indonesia masih terus berkembang dan berpotensi memberikan hasil ringkasan yang lebih baik dalam hal relevansi kontekstual dibandingkan metode berbasis statistik konvensional. Oleh karena itu, penelitian ini memilih metode berbasis *Transformer* untuk membangun sistem ATS pada dokumen berita berbahasa Indonesia, dengan tujuan menghasilkan ringkasan yang lebih singkat, relevan, runtut, dan tetap mewakili informasi penting dari dokumen asli.

METODE

Metode dan Desain Penelitian

Penelitian ini menggunakan metode penelitian eksperimen dengan pendekatan kuantitatif. Metode eksperimen digunakan untuk menguji kemampuan model *Transformer* dalam menghasilkan ringkasan otomatis pada dokumen berita; hasilnya diukur berdasarkan nilai evaluasi ROUGE dan BLEU. Penelitian ini juga menggunakan pendekatan rekayasa perangkat lunak untuk membangun aplikasi ringkasan otomatis berbasis *web*.

Metode pengembangan sistem yang digunakan adalah model *Prototype*. Model ini dipilih karena memungkinkan peneliti membuat rancangan awal sistem terlebih dahulu, kemudian melakukan evaluasi dan perbaikan secara bertahap hingga sistem sesuai dengan kebutuhan pengguna. Tahapan model *Prototype* meliputi: (1) pengumpulan kebutuhan; (2) membangun prototipe awal; (3) evaluasi prototipe; (4) pengkodean program; (5) pengujian program; dan (6) implementasi dan perbaikan.



Gambar 1. Model Prototype Pengembangan Sistem

Objek dan Subjek Penelitian

Objek penelitian adalah proses peringkasan otomatis dokumen berita menggunakan metode ATS berbasis *Transformer*. Subjek penelitian berupa kumpulan dokumen berita yang diperoleh dari portal berita *online* berbahasa Indonesia, digunakan sebagai data masukan untuk pengujian sistem.

Teknik Pengumpulan Data

Data primer diperoleh langsung dari portal berita *online* sebagai bahan utama penelitian. Data sekunder berupa jurnal, buku, dan artikel ilmiah yang berkaitan dengan NLP, ATS, dan metode *Transformer*. Teknik pengumpulan data meliputi studi literatur, dokumentasi dataset berita, dan observasi terhadap proses kerja sistem.

Kebutuhan Sistem

1. Kebutuhan Fungsional
Aplikasi harus mampu: menerima masukan dokumen berita dalam bentuk teks maupun URL; melakukan *preprocessing* teks; memproses dokumen menggunakan model ATS berbasis *Transformer*; menghasilkan ringkasan otomatis; menampilkan dokumen asli dan hasil ringkasan secara bersamaan; serta menyimpan hasil ringkasan jika diperlukan.
2. Kebutuhan Non-Fungsional
Aplikasi harus memenuhi aspek kinerja (*performance*), keandalan (*reliability*), kemudahan penggunaan (*usability*), kompatibilitas (*compatibility*), dan keamanan (*security*). Aplikasi juga harus bersifat *maintainable*, sehingga model *Transformer* dapat diperbarui apabila tersedia model yang lebih baik.

Kebutuhan Perangkat Keras dan Lunak

Perangkat keras minimum: prosesor Intel Core i5 generasi 8 atau AMD Ryzen 5, RAM 8 GB (disarankan 16 GB), penyimpanan SSD 256 GB, dan NVIDIA GPU dengan CUDA (opsional). Perangkat lunak: Windows 10/11, Python 3.10 atau lebih baru, Visual Studio Code, Flask atau FastAPI (*framework backend*), PyTorch (*framework deep learning*), serta library ROUGE untuk evaluasi.

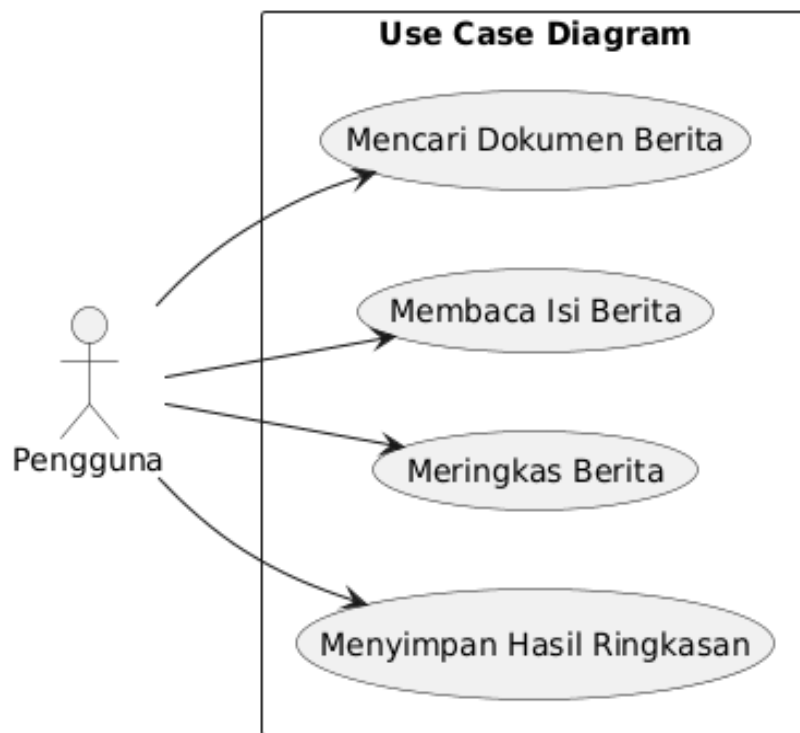
Arsitektur Sistem

Arsitektur sistem terdiri dari enam tahapan utama yang berjalan secara berurutan. Adapun arsitektur sistem yang di maksud adalah sebagai berikut:

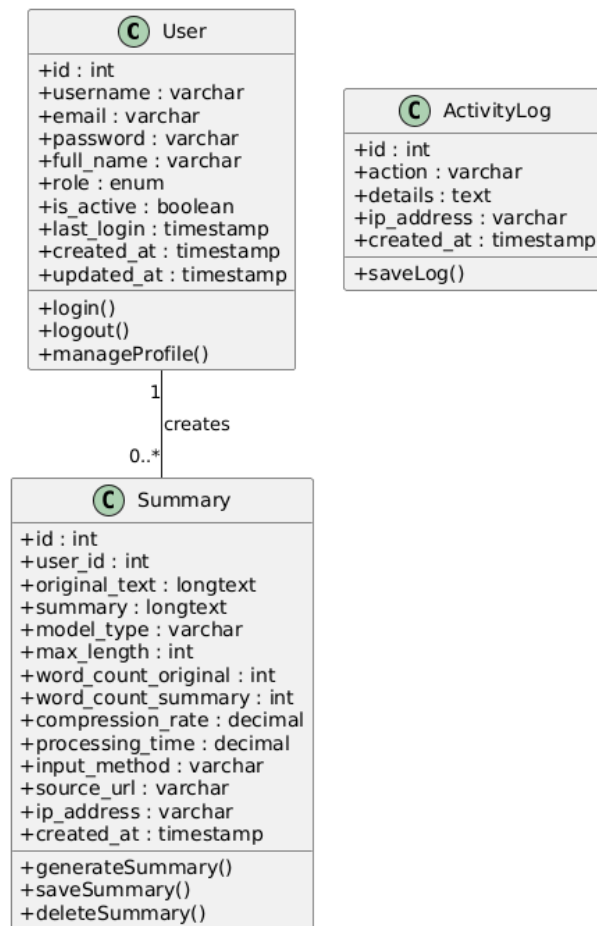
1. Input Data: Pengguna memasukkan dokumen berita melalui antarmuka *web* berupa teks langsung atau URL berita.
2. Preprocessing: Program melakukan pembersihan teks, termasuk menghapus simbol, karakter khusus, dan normalisasi teks.
3. Tokenisasi: Teks diubah menjadi token agar dapat diproses oleh model *Transformer*.
4. Proses *Transformer*: Model *Transformer* melakukan analisis konteks dan menghasilkan representasi teks melalui mekanisme *attention*.
5. *Automatic Text Summarization*: Program menghasilkan ringkasan otomatis berdasarkan hasil pemrosesan model.
6. Output Ringkasan: Hasil ringkasan ditampilkan kepada pengguna beserta statistik kompresi.

Perancangan Model UML

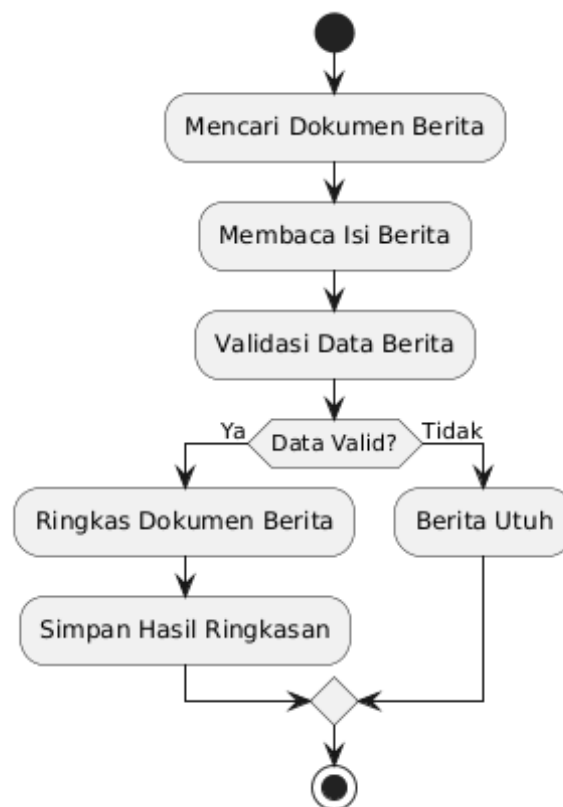
Perancangan model menggunakan *Unified Modeling Language* (UML), meliputi *Use Case Diagram*, *Class Diagram*, *Activity Diagram*, dan *Sequence Diagram*.



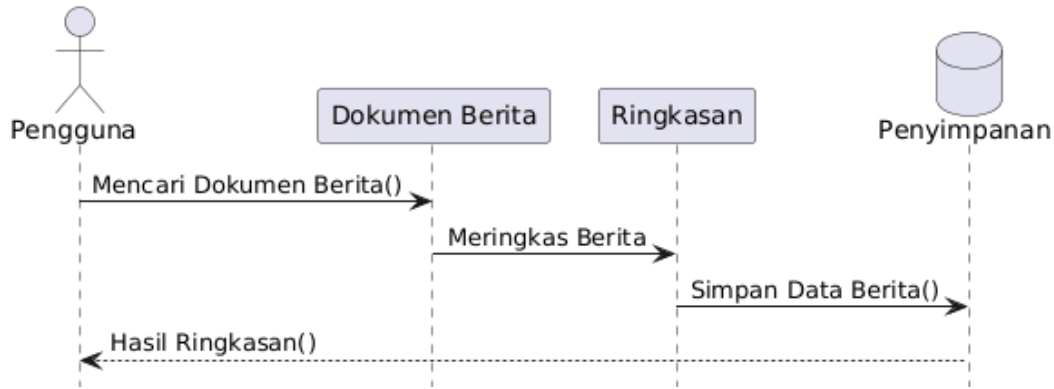
Gambar 2. Use Case Diagram



Gambar 3. Class Diagram



Gambar 4. Activity Diagram



Gambar 5. Sequence Diagram

Perancangan Basis Data

Basis data terdiri dari tiga tabel utama. Tabel Users menyimpan data pengguna dengan field: id, username, email, password, full_name, role, is_active, last_login, created_at, dan updated_at. Tabel Summaries menyimpan hasil ringkasan dengan field: id, user_id, original_text, summary, model_type, max_length, word_count_original, word_count_summary, compression_rate, processing_time, input_method, source_url, ip_address, dan created_at. Tabel Activity Log menyimpan log aktivitas pengguna dengan field: id, action, details, ip_address, dan created_at.

Metode Evaluasi

Evaluasi kualitas ringkasan dilakukan menggunakan dua metrik utama:

1. ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*) mengukur tingkat kesamaan antara ringkasan yang dihasilkan sistem dengan ringkasan referensi melalui perhitungan *n-gram overlap* (Lin, 2004). Tiga varian ROUGE yang digunakan adalah: ROUGE-1 (unigram), ROUGE-2 (bigram), dan ROUGE-L (*Longest Common Subsequence*).
2. BLEU (*Bilingual Evaluation Understudy*) mengukur kesamaan berdasarkan *n-gram* hingga empat kata, dengan penalti (*Brevity Penalty*) apabila panjang *candidate* lebih pendek dibandingkan *reference*.

Tingkat Kompresi dihitung menggunakan dua rumus:

$$\text{Compression Ratio} = (\text{Jumlah Kata Ringkasan} / \text{Jumlah Kata Asli}) \times 100\%$$
$$\text{Compression Rate} = (1 - \text{Compression Ratio}) \times 100\%.$$

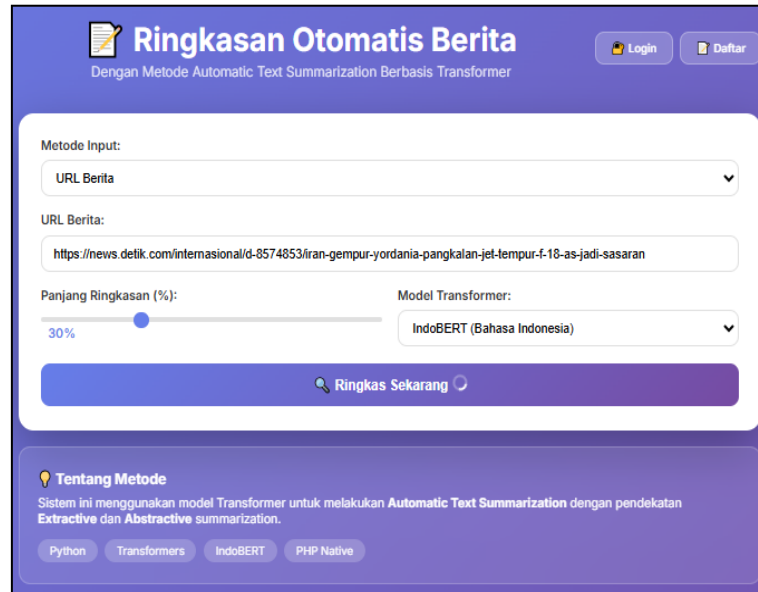
HASIL DAN PEMBAHASAN

Implementasi Antarmuka Sistem

Sistem diimplementasikan sebagai aplikasi berbasis *web* yang terdiri dari antarmuka pengguna (*user interface*) dan antarmuka administrator (*admin interface*).

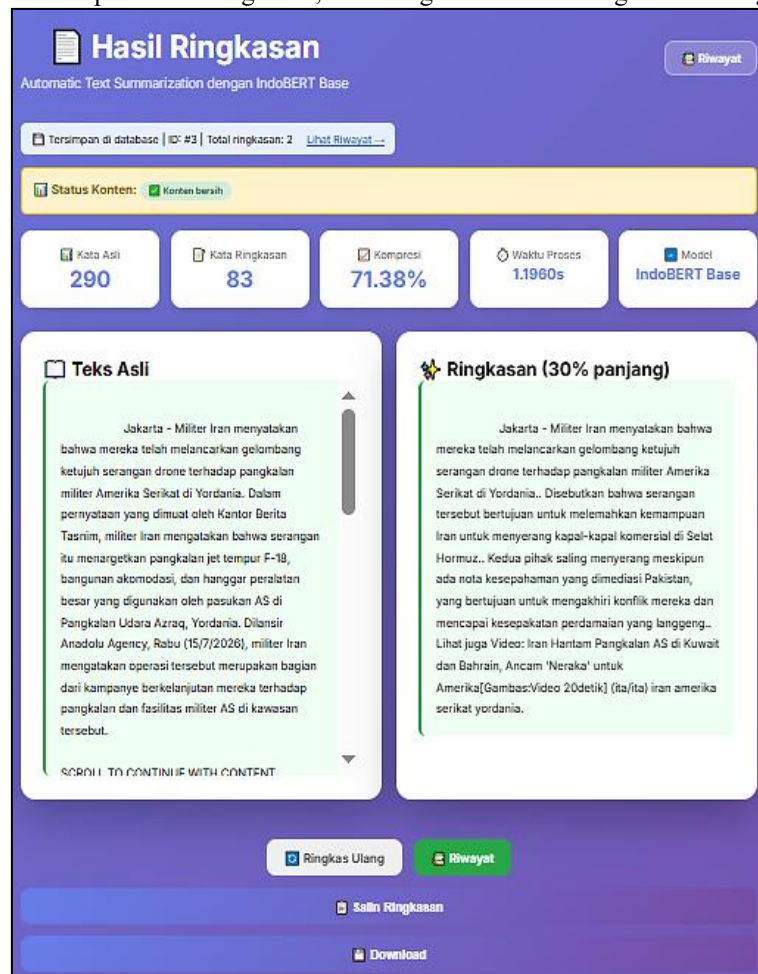
Gambar 6. Halaman Utama Aplikasi

Gambar 6 menampilkan halaman utama sistem yang menyediakan dua metode masukan: teks langsung dan URL berita. Pengguna dapat mengatur persentase panjang ringkasan yang diinginkan.



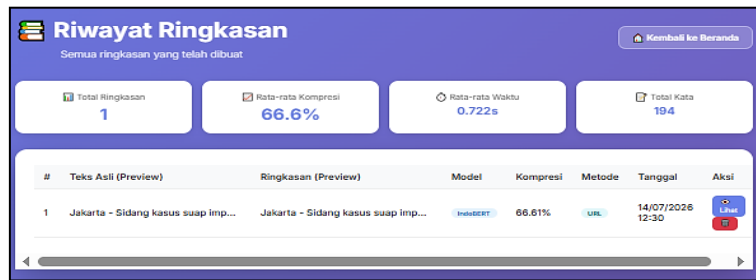
Gambar 7. Antarmuka Pengisian Data/Input Berita

Gambar 7 menampilkan antarmuka pengisian data berita, di mana pengguna memasukkan URL berita atau teks, memilih persentase ringkasan, dan mengklik tombol “Ringkas Sekarang”.



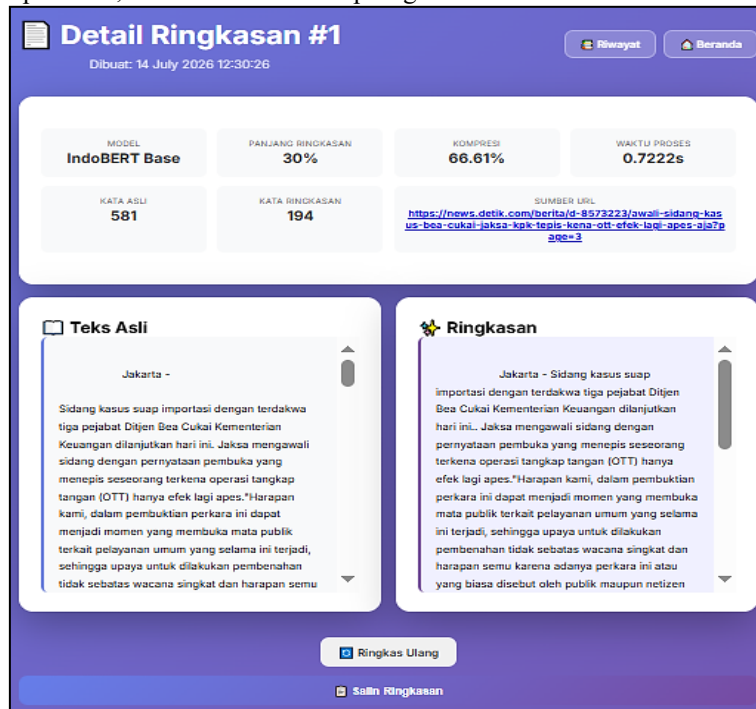
Gambar 8. Antarmuka Hasil Ringkasan

Gambar 8 menampilkan hasil ringkasan secara berdampingan dengan teks asli. Pengguna dapat mengunduh hasil ringkasan melalui tombol *Download* yang tersedia.



Gambar 9. Halaman Admin

Gambar 9 menampilkan halaman administrator yang dapat memantau riwayat seluruh hasil ringkasan, menghapus entri, dan melihat detail tiap ringkasan.



Gambar 10. Halaman Detail Riwayat Ringkasan

Gambar 10 menampilkan detail riwayat ringkasan mencakup model yang digunakan, persentase ringkasan, rasio kompresi, dan waktu proses.

Pengujian Sistem

Pengujian dilakukan terhadap sebuah artikel berita berjudul “Iran Gempur Yordania, Pangkalan Jet Tempur F-18 AS Jadi Sasaran” yang diperoleh dari portal berita Detik.com. Artikel tersebut memiliki 290 kata, 14 kalimat, dan 2.145 karakter.

Tahap Preprocessing

Tahap *preprocessing* terdiri atas tokenisasi dan *stopword removal*. Setelah tokenisasi, jumlah token meningkat menjadi 299 (dari 290 kata), karena tanda baca juga dihitung sebagai token tersendiri. Setelah *stopword removal*, diperoleh 153 kata unik yang digunakan dalam analisis selanjutnya.

Analisis Frekuensi Kata

Frekuensi kata digunakan sebagai dasar penentuan tingkat kepentingan kalimat. Hasil analisis frekuensi kata disajikan pada Tabel 1.

Tabel 1. Analisis Frekuensi Kata

Kata	Frekuensi	Kategori
Iran	12	Entitas Utama
AS/Amerika	9	Entitas Utama
serangan	7	Kata Kunci
Militer	6	Kata Kunci
Yordania	3	Lokasi
Selat Hormuz	3	Lokasi
Drone	3	Kata Kunci
pangkalan	3	Kata Kunci

Pembobotan Kalimat (Sentence Scoring)

Setiap kalimat dinilai berdasarkan enam komponen dengan bobot yang telah ditentukan. Formula scoring yang digunakan adalah:

$$\text{Skor} = \text{Posisi} + \text{Panjang} + \text{Kata Kunci} + \text{Keunikan} + \text{Panjang Lanjutan} + \text{Kata Penting}$$

Detail bobot setiap komponen disajikan pada Tabel 2.

Tabel 2. Komponen dan Bobot Penilaian Kalimat

Komponen	Bobot	Keterangan
Posisi	30%	Kalimat pertama: 0,30; terakhir: 0,20; tengah: 0,10
Panjang	20%	Kalimat dengan 12–28 kata dianggap ideal
Kata Kunci	15%	Berdasarkan jumlah kata kunci dalam kalimat
Keunikan	15%	Berdasarkan rasio kata unik terhadap jumlah kata
Panjang Lanjutan	10%	Berdasarkan normalisasi panjang kalimat
Kata Penting	10%	Berdasarkan jumlah kata penting

Sebagai contoh, kalimat pertama (“Jakarta — Militer Iran menyatakan bahwa mereka telah melancarkan gelombang ketujuh serangan *drone* terhadap pangkalan militer Amerika Serikat di Yordania.”) memiliki 20 kata dengan komponen: posisi = 0,300; panjang = 0,200; kata kunci = 0,120; keunikan = 0,150; panjang lanjutan = 0,100; kata penting = 0,060; sehingga skor mentah (*raw*) = 0,930. Setelah dinormalisasi dengan pembagi 1,315, diperoleh skor akhir = 0,893.

Ranking Kalimat

Hasil *ranking* keempat kalimat terpilih disajikan pada Tabel 3.

Tabel 3. Ranking Kalimat Berdasarkan Skor

Ranking	Nomor Kalimat	Skor
1	1	0,893
2	14	0,656
3	3	0,600
4	8	0,535

Sistem menggunakan parameter ringkasan 30% dari total 14 kalimat, sehingga jumlah kalimat terpilih adalah $[14 \times 0,30] = 4$ kalimat.

Evaluasi ROUGE

Hasil evaluasi ROUGE disajikan pada Tabel 4. Berdasarkan data pengujian, *candidate* ringkasan memiliki 15 kata, sedangkan *reference* memiliki 54 kata, dengan *overlap* sebesar 15 kata (ROUGE-1) dan 14 bigram (ROUGE-2).

Tabel 4. Hasil Evaluasi ROUGE

Metrik	Precision	Recall	F1-Score
ROUGE-1	100%	27,78%	43,48%
ROUGE-2	100%	26,42%	41,79%
ROUGE-L	100%	27,78%	43,48%

Nilai *precision* 100% pada seluruh metrik ROUGE menunjukkan bahwa semua kata dalam *candidate* terdapat pada *reference*. Nilai *recall* yang berkisar 26–28% menunjukkan bahwa ringkasan hanya mencakup sebagian kecil dari *reference*, yang merupakan karakteristik wajar dari metode *extractive* dengan kompresi tinggi. Nilai F1 ROUGE-1 dan ROUGE-L sebesar 43,48% serta ROUGE-2 sebesar 41,79% menunjukkan keseimbangan yang memadai antara *precision* dan *recall*, dan sebanding dengan hasil penelitian terdahulu yang menggunakan metode berbasis fitur pada bahasa Indonesia (Maylawati et al., 2023; Bahari & Dewi, 2024).

Evaluasi BLEU-4

Nilai BLEU-4 yang diperoleh adalah 7,43%. Nilai ini dipengaruhi oleh *Brevity Penalty* (BP = 0,0743) karena panjang *candidate* (15 kata) jauh lebih pendek dibandingkan *reference* (54 kata), dengan rasio $15/54 = 0,278$. Meskipun semua *n-gram* dari *unigram* hingga 4-gram memiliki *precision* 100%, penalti panjang menyebabkan nilai BLEU-4 menjadi rendah. Hal ini merupakan karakteristik umum pada sistem ATS berbasis *extractive* dengan tingkat kompresi tinggi (Lin, 2004).

Analisis Kompresi Teks

Hasil kompresi teks disajikan pada Tabel 5.

Tabel 5. Hasil Kompresi Teks

Parameter	Nilai
Jumlah kata teks asli	290 kata
Jumlah kata ringkasan	83 kata
<i>Compression Ratio</i>	28,62%
<i>Compression Rate</i>	71,38%

Sistem berhasil mereduksi dokumen berita sebesar 71,38%, mempertahankan hanya 28,62% dari panjang teks asli. Hasil ini menunjukkan kemampuan sistem dalam menghasilkan ringkasan yang jauh lebih ringkas sambil tetap mempertahankan informasi utama dari dokumen sumber.

Pembahasan

Secara keseluruhan, sistem ATS berbasis *Transformer* yang dikembangkan berhasil menjalankan proses peringkasan teks melalui tahapan *preprocessing*, analisis frekuensi kata, pembobotan kalimat, *ranking*, dan pembentukan ringkasan. Kalimat dipilih berdasarkan kombinasi karakteristik posisi, panjang, kata kunci, keunikan, panjang lanjutan, dan kata penting, sehingga kalimat dengan karakteristik paling sesuai memperoleh skor tertinggi dan terpilih sebagai bagian dari ringkasan.

Metode *Transformer* memberikan keunggulan dalam pemahaman konteks dan hubungan antar kalimat, menghasilkan ringkasan yang lebih informatif dibandingkan metode berbasis statistik konvensional (Vaswani et al., 2017). Hal ini sejalan dengan temuan Winarko et al. (2025) dan Agung et al. (2025) yang menunjukkan bahwa model berbasis *Transformer* secara konsisten menghasilkan ringkasan dengan relevansi kontekstual yang lebih tinggi untuk bahasa Indonesia.

Namun, hasil pengujian juga mengidentifikasi keterbatasan: terdapat kalimat yang secara statistik memperoleh skor tinggi tetapi kurang relevan sebagai informasi utama, seperti kalimat yang mengandung unsur “Lihat juga Video...” yang ikut terpilih. Kondisi ini menunjukkan bahwa pembobotan berbasis fitur statistik masih memiliki keterbatasan dalam memahami relevansi semantik suatu kalimat (Yadav et al., 2023). Oleh karena itu, pengembangan lanjutan dapat dilakukan dengan menambahkan mekanisme pengenalan konten non-utama seperti tautan, iklan, atau elemen editorial.

KESIMPULAN

Berdasarkan hasil penelitian ini, beberapa kesimpulan dapat diambil. Pertama, penelitian berhasil membangun aplikasi yang mampu menghasilkan ringkasan dokumen berita secara otomatis menggunakan metode ATS berbasis *Transformer*. Kedua, model *Transformer* mampu memahami konteks dan hubungan antar kalimat sehingga ringkasan yang dihasilkan tetap mempertahankan informasi penting dari dokumen asli. Ketiga, sistem berhasil mereduksi dokumen berita sebesar 71,38% dengan nilai ROUGE-1 F1 sebesar 43,48%, ROUGE-2 F1 sebesar 41,79%, ROUGE-L F1 sebesar 43,48%, dan BLEU-4 sebesar 7,43%. Keempat, aplikasi yang dikembangkan dapat menjadi alternatif dalam membantu pengguna memperoleh inti informasi dari dokumen berita yang panjang sehingga proses membaca menjadi lebih praktis dan efisien.

Saran untuk pengembangan selanjutnya: (1) menggunakan model *Transformer* yang lebih mutakhir seperti BART atau PEGASUS untuk meningkatkan kualitas ringkasan; (2) menambahkan kemampuan untuk memproses berbagai format dokumen seperti PDF dan DOCX; (3) mengembangkan kemampuan multi-bahasa, khususnya bahasa Indonesia dan Inggris; dan (4) melakukan *fine-tuning* model menggunakan dataset berita berbahasa Indonesia dalam jumlah lebih besar agar hasil ringkasan lebih akurat dan sesuai karakteristik bahasa Indonesia.

DAFTAR PUSTAKA

- Agung, I. G. I. U. P., Trisna, I. N. P., & Kadek, N. (2025). Abstractive text summarization to generate Indonesian news highlight using Transformers model. *Indonesian Journal of Information Systems*, 7(2), 1248–1263. <https://doi.org/10.51519/journalisi.v7i2.1082>
- Bahari, A., & Dewi, K. E. (2024). Peringkasan teks otomatis abstraktif menggunakan Transformer pada teks bahasa Indonesia. *KOMPUTA: Jurnal Ilmiah Komputer dan Informatika*, 13(1).
- Dewi, K. E., & Widiastuti, N. I. (2022). Automatic summarization of Indonesian texts using a hybrid approach. *Jurnal Nasional Teknologi dan Sistem Informasi*, 15(1). <https://doi.org/10.25077/TEKNOSI.v8i1.2022.33-41>
- Itsaini, Q. A., Sholihati, I., & Putri, F. A. (2023). Abstractive text summarization using pre-trained language model “Text-to-Text Transfer Transformer (T5)”. *Jurnal Ilmiah Ilmu Komputer*, 15(1), 124–131.
- Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out* (pp. 74–81). Association for Computational Linguistics. <https://aclanthology.org/W04-1013/>
- Maylawati, D. S., Kumar, Y. J., & Kasmin, F. (2023). Feature-based approach and sequential pattern mining to enhance quality of Indonesian automatic text summarization. *Indonesian Journal of Electrical Engineering and Computer Science*, 30(3), 1795–1804. <https://doi.org/10.11591/ijeecs.v30.i3.pp1795-1804>
- Monica, S., Girsang, A. S., Aurelia, M., & Lee, S. (2025). Category-aware abstractive text summarization. *arXiv preprint arXiv:2501.XXXXX*, 1–10.

- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 5998–6008. <https://doi.org/10.48550/arXiv.1706.03762>
- Winarko, E., Tanoto, L., & Reza, M. H. (2025). Indonesian abstractive text summarization using stacked embeddings and Transformer decoder. *Jurnal Ilmu Komputer dan Informasi*, 52(4), 1051–1061.
- Yadav, A. K., Ranvijay, Yadav, R. S., & Maurya, A. K. (2023). State-of-the-art approach to extractive text summarization: A comprehensive review. *Multimedia Tools and Applications*, 82(19), 29135–29197. <https://doi.org/10.1007/s11042-023-14613-9>
- Yanuarti, R., & Azizah, H. (2022). Implementasi text summarization pada reading comprehension menggunakan library Python. *Jurnal Informatika dan Sistem Informasi*, 2(1), 43–51.