P- ISSN: 2355-6684, E-ISSN: 2776-6640
https://ojs.unpkediri.ac.id/index.php/noe

Sistem Otomatisasi Ringkasan Literatur Berbahasa Indonesia Menggunakan Metode *Retrieval-Augmented Generation* (RAG) Dan Model IndoT5

Aisyah Mufidah[1*], Badie'ah[2]
Universitas Islam Sultan Agung Semarang[1,2]
aisyahmufidah@std.unissula.ac.id[1], badieah.assegaf@unissula.ac.id[2]
Corresponding author: Aisyah Mufidah

Abstrak

Peningkatan jumlah publikasi ilmiah di Indonesia menimbulkan tantangan dalam menyaring dan memahami literatur secara efisien. Proses kajian literatur manual memerlukan waktu yang panjang, rentan terhadap bias kognitif, dan sulit mengikuti perkembangan riset terkini. Untuk mengatasi permasalahan ini, penelitian ini mengembangkan sistem otomatisasi ringkasan literatur yang mengintegrasikan Retrieval-Augmented Generation (RAG) dengan model IndoT5 dan pendekatan struktur IMRAD (Introduction, Methods, Results, and Discussion). Sistem menggabungkan proses peringkasan menggunakan IndoT5, indexing berbasis FAISS, serta embedding IndoBERT untuk pencarian dokumen yang relevan secara semantik. Evaluasi sistem menggunakan metrik BERTScore menunjukkan kualitas ringkasan dengan skor precision 0.828, recall 0.881, dan F1-score 0.854. Penilaian menggunakan LLM-as-a-Judge dengan model LLaMA-3-70B menghasilkan skor rata-rata 4.78 dari skala 5 untuk aspek relevansi, kebenaran, dan kelengkapan respons. Hasil penelitian membuktikan bahwa sistem mampu menghasilkan ringkasan yang informatif dan kontekstual, serta mempercepat proses kajian literatur berbahasa Indonesia secara signifikan.

Kata Kunci : *IndoT5, Literatur Akademik, Otomatisasi Ringkasan, Retrieval-Augmented Generation, Text Summarization*

A. PENDAHULUAN

Peningkatan pesat publikasi ilmiah dalam beberapa tahun terakhir telah membawa dampak signifikan bagi dunia riset. Di Indonesia, integrasi lembaga riset ke dalam BRIN mendorong percepatan output penelitian. Handoyo dkk., 2024 menunjukkan bahwa dalam periode 2015–2021, terdapat 12.209 dokumen dari institusi Indonesia di Scopus dengan pertumbuhan rata-rata 30 % per tahun. Bahkan, dalam dua tahun berikutnya (2022–2023), jumlahnya melonjak lagi sebesar 8.081 dokumen dengan laju 36 %. Namun, peningkatan ini menimbulkan tantangan baru: bagaimana menyaring dan memahami literatur dalam jumlah besar secara efisien.

Kajian literatur adalah fondasi krusial dalam proses ilmiah, berperan dalam merumuskan tujuan penelitian, menentukan metodologi, hingga menyusun kerangka teoritis. Sayangnya, proses ini kerap menyita waktu, memerlukan pembacaan intensif, dan rentan terhadap bias kognitif (Ridwan dkk., 2021). Dengan jumlah artikel yang terus bertambah, kemampuan peneliti untuk menganalisis literatur secara manual menjadi terbatas. Tantangan ini makin kompleks ketika literatur ditulis dalam bahasa lokal, seperti Bahasa Indonesia, yang belum sepenuhnya terakomodasi oleh sistem pemrosesan bahasa alami (Cahyawijaya dkk., 2021; Wilie dkk., 2020).

Kemajuan *Natural Language Proccessing* (NLP) telah menghasilkan berbagai model berbasis *Transformer* seperti *BERT* dan *GPT*. Salah satu pendekatan inovatif yang muncul adalah *Retrieval-Augmented Generation* (RAG), yang menggabungkan mekanisme pengambilan informasi (*retrieval*) dan pembangkitan teks (*generation*). Pendekatan ini terbukti mampu mengurangi fenomena *hallucination* dan menghasilkan ringkasan yang lebih relevan (Cheng dkk., 2025; Gupta & Ranjan, 2024). Namun, pengembangan RAG masih berfokus pada bahasa Inggris, dengan sumber data dan model yang tidak selalu sesuai dengan konteks lokal (Hahsler, 2023; Jaber & Gérard, 2025).

Bahasa Indonesia, dengan struktur morfologis kompleks dan variasi dialek yang beragam, sementara ketersediaan korpus berskala besar masih terbatas (Wilie dkk., 2020). Kondisi ini berdampak pada penurunan performa RAG ketika diterapkan pada dokumen ilmiah berbahasa Indonesia (Muhammad dkk., 2025). Selain itu, literatur Indonesia belum sepenuhnya mudah diakses, sementara mayoritas alat bantu literatur digital lebih mendukung konten berbahasa Inggris.

Di sisi lain, struktur artikel ilmiah yang umumnya mengikuti pola IMRAD (*Introduction, Methods, Results, and Discussion*) membuka peluang untuk menyusun sistem ringkasan otomatis yang lebih terarah. Sakti Wiradinata dkk., 2024 menyatakan bahwa *summarization* berbasis IMRAD mampu

mengekstraksi elemen penting seperti variabel, metodologi, dan temuan utama. Dengan memanfaatkan segmentasi ini, sistem peringkasan dapat dibuat lebih sistematis dan informatif.

Sebagai solusi atas keterbatasan penerapan RAG pada dokumen ilmiah berbahasa Indonesia yang disebabkan oleh minimnya model bahasa lokal dan rendahnya ketersediaan korpus berkualitas serta untuk memanfaatkan potensi segmentasi berbasis IMRAD, integrasi pendekatan RAG dengan model bahasa lokal seperti IndoT5 menjadi pilihan yang menjanjikan. IndoT5 merupakan model transformer adaptif untuk Bahasa Indonesia yang unggul dalam memahami konteks lokal (Yani dkk., 2024). Penelitian ini mengusulkan sistem yang melakukan peringkasan otomatis berdasarkan struktur IMRAD, lalu mengindeks hasil ringkasan tersebut untuk menjawab pertanyaan pengguna secara kontekstual. Pendekatan ini mengombinasikan kekuatan IndoT5 dalam *summarization* dengan efisiensi *indexing* berbasis FAISS dan representasi teks dari *IndoBERT*.

Penelitian ini bertujuan untuk merancang dan mengevaluasi sistem RAG dengan model peringkas IndoT5 yang dapat mengotomatisasi pencarian dan peringkasan literatur ilmiah berbahasa Indonesia. Sistem ini diharapkan dapat mempercepat proses kajian literatur, meningkatkan akurasi pemahaman konten ilmiah, serta memperkuat pemanfaatan teknologi NLP lokal dalam ekosistem riset nasional.

B. LANDASAN TEORI

1. Struktur Artikel Ilmiah berbasis IMRAD

   Artikel ilmiah umumnya menggunakan format IMRAD (*Introduction, Methods, Results, and Discussion*) untuk mengatur alur penyampaian informasi secara sistematis. Bagian *Introduction* menjelaskan konteks, masalah, dan tujuan penelitian; *Methods* mendeskripsikan prosedur dan instrumen yang digunakan; *Results* memaparkan temuan penelitian; sedangkan *Discussion* menafsirkan hasil dan menghubungkannya dengan literatur sebelumnya (Sollaci & Pereira, 2004).

   Dalam konteks *text mining* dan peringkasan otomatis, struktur IMRAD memudahkan segmentasi teks sehingga sistem dapat fokus mengekstraksi informasi spesifik dari setiap bagian. (Sakti Wiradinata dkk., 2024) menunjukkan bahwa pemodelan ringkasan berbasis IMRAD mampu meningkatkan relevansi informasi yang diambil, terutama untuk menemukan variabel, metodologi, dan temuan utama penelitian.

   Struktur IMRAD digunakan sebagai dasar dalam proses peringkasan. Dokumen ilmiah dipecah menjadi bagian-bagian sesuai format ini, lalu diringkas secara terpisah menggunakan IndoT5 untuk mempertahankan esensi dari setiap bagian.

2. Representasi Teks Menggunakan *IndoBERT*

   Representasi teks adalah proses mengubah teks mentah menjadi bentuk vektor yang dapat diproses model pembelajaran mesin. *IndoBERT* adalah *model Bidirectional Encoder Representations from Transformers* yang dilatih khusus untuk Bahasa Indonesia menggunakan korpus besar dari Wikipedia, berita, dan media social (Wilie dkk., 2020).

   Dengan mekanisme *self-attention*, *IndoBERT* menangkap hubungan antar kata dalam konteks kalimat secara dua arah. Model ini menunjukkan kinerja unggul pada berbagai tugas *Natural Language Understanding* (NLU) dalam *benchmark IndoNLU*, termasuk *text classification, named entity recognition*, dan *question answering* (Vaswani dkk., 2017).

   IndoBERT digunakan untuk menghasilkan *embedding* teks yang akan dimanfaatkan dalam proses pencarian dokumen pada RAG. *Embedding* ini memungkinkan sistem untuk menemukan dokumen yang relevan berdasarkan kesamaan semantik.

3. Peringkasan Teks Menggunakan IndoT5

   IndoT5 adalah adaptasi model *Text-to-Text Transfer Transformer* (T5) untuk Bahasa Indonesia, yang bekerja dengan mengubah semua tugas NLP menjadi format *text-to-text* (Raffel dkk., 2020). Model ini dilatih menggunakan korpus Indonesia dalam skala besar sehingga dapat memahami struktur kalimat dan konteks lokal secara lebih baik (Yani dkk., 2024).

   Dalam peringkasan (*summarization*), IndoT5 berfungsi sebagai *encoder–decoder*: *encoder* memproses teks *input*, dan *decoder* menghasilkan ringkasan. Keunggulan IndoT5 dibanding metode ekstraktif murni adalah kemampuannya melakukan *abstractive summarization*, yaitu menyusun kalimat baru yang tetap setia pada makna teks asli. Studi Yani dkk., 2024 menunjukkan bahwa IndoT5 menghasilkan skor *ROUGE* dan *BERTScore* yang lebih tinggi dibanding model generatif non-lokal.

Pada sistem ini, IndoT5 digunakan untuk melakukan peringkasan otomatis pada setiap bagian IMRAD dari dokumen ilmiah. Proses ini bertujuan untuk menyaring informasi penting dan relevan sebelum diintegrasikan ke dalam pipeline RAG.

4. Evaluasi Model

a. Evaluasi Ringkasan Artikel

BERTScore adalah metrik evaluasi yang menghitung kesamaan semantik antara ringkasan sistem dan referensi menggunakan *embedding* dari model seperti *BERT* atau *IndoBERT*. Berbeda dengan *ROUGE* yang berbasis token, *BERTScore* lebih peka terhadap makna dan sinonim (Zhang dkk., 2019). Skor akhir dihitung dari *precision*, *recall*, dan *F1-score*.

BERTScore dihitung sebagai rata-rata dari semua skor kemiripan token prediksi terhadap token referensi, dengan rumus:

$$BERTscore = \frac{1}{n} \sum_{i=1}^{n} \underset{j \in Reference}{max} Cosine\ Similarity(u_i, u_j) \quad \text{.....(1)}$$

dimana:

- n = Jumlah token dalam teks prediksi
- u_i = Token ke-i dari teks prediksi
- v_j = Token ke-j dari teks referensi
- $max\ j \in reference$ = memilih nilai *cosine similarity* maksimum dari token u_i terhadap semua token v_j dalam referensi

b. Evaluasi *Retrieval-Augemented Generation* (RAG)

Evaluasi sistem *Retrieval-Augmented Generation* secara tradisional dilakukan melalui uji manusia (*human evaluation*) atau metrik otomatis berbasis kesamaan teks. Namun, metode ini mahal dan kurang fleksibel. Pendekatan terbaru, *LLM-as-a-Judge*, menggunakan model bahasa besar sebagai evaluator untuk menilai aspek relevansi, akurasi, dan *groundedness* jawaban (Li dkk., 2024).

Dalam skema ini, *Large Language Models* (LLM) diberikan prompt evaluasi yang mendeskripsikan kriteria penilaian. Beberapa penelitian (Shuliang Liu dkk., 2025; Zheng dkk., 2023) menunjukkan bahwa skor *dari LLM-as-a-Judge* memiliki korelasi tinggi dengan penilaian manusia, sehingga cocok untuk evaluasi skala besar. Kelebihannya meliputi skalabilitas, fleksibilitas kriteria, dan penghematan waktu, namun kelemahannya adalah potensi bias dari model penilai itu sendiri.

LLM-as-a-Judge adalah metode evaluasi otomatis berbasis LLM yang menilai kualitas keluaran teks berdasarkan dimensi relevansi, *faithfulness*, dan kelengkapan. Penilaian dilakukan dengan skala *Likert* 1–5 menggunakan *prompt* eksplisit dan teknik *chain-of-thought prompting* untuk meningkatkan akurasi penilaian (Gu dkk., 2024; Li dkk., 2024). Model seperti *LLaMA 3* digunakan untuk mengevaluasi keluaran sistem RAG dalam penelitian ini.

Untuk memperoleh nilai akhir dari setiap jawaban, skor dari ketiga aspek tersebut dirata-ratakan dengan rumus sebagai berikut:

$$TotalScore_i = \frac{S_{relevansi} + S_{faithfulness} + S_{kelengkapan}}{3} \quad \text{.....(2)}$$

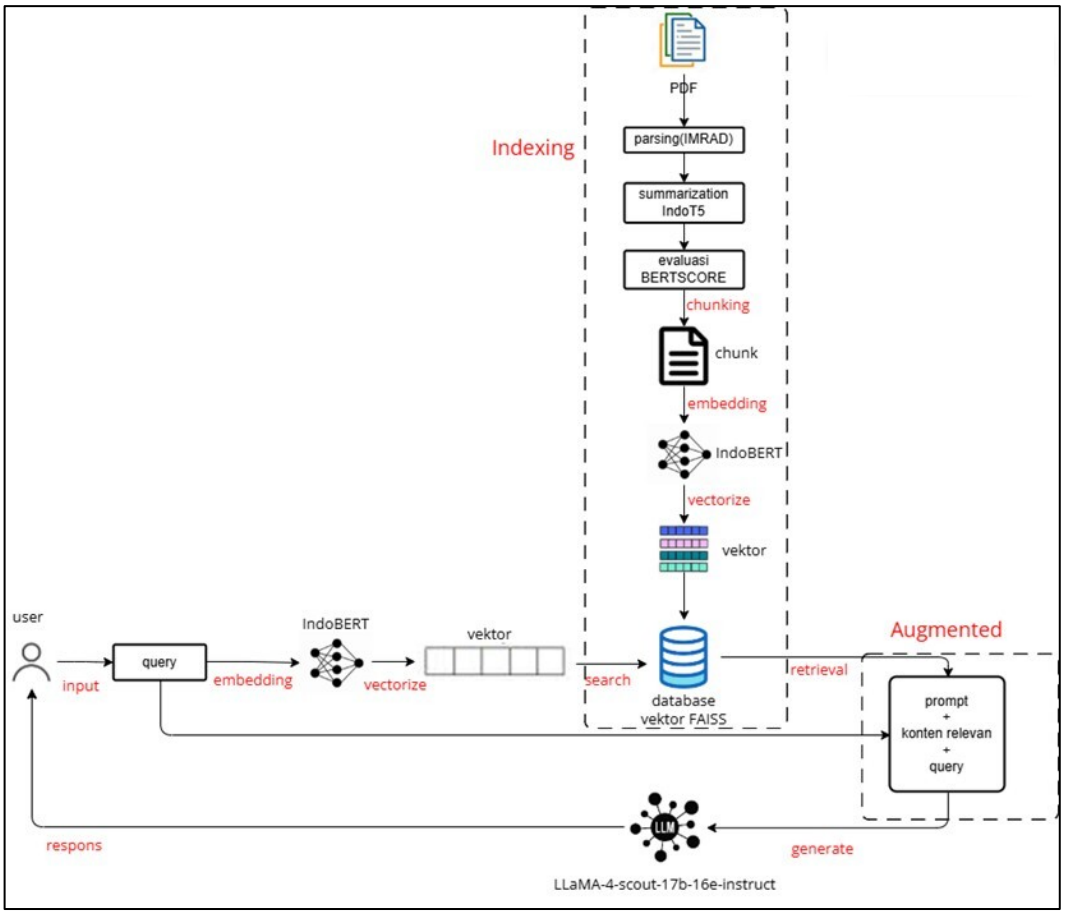
Keterangan:

- TotalScore : Nilai evaluasi akhir untuk jawaban ke-i
- $S_{relevansi}$: Skor relevansi jawaban ke-i
- $S_{faithfulness}$: Skor kesesuaian isi (*faithfulness*) jawaban ke-i
- $S_{kelengkapan}$: Skor kelengkapan jawaban ke-i
- Semua skor berada dalam rentang 1 (sangat buruk) sampai 5 (sangat baik)

C. METODE PENELITIAN

Untuk menggambarkan alur kerja sistem yang dikembangkan, dibuat sebuah *workflow* diagram. Diagram ini menjelaskan tahapan utama mulai dari pengumpulan data hingga proses *retrieval* dan generasi jawaban, sebagaimana ditunjukkan pada Gambar 1.

P- ISSN: 2355-6684, E-ISSN: 2776-6640
https://ojs.unpkediri.ac.id/index.php/noe



Gambar 1. *Workflow* Sistem

Gambar 1 menunjukkan alur kerja sistem *Retrieval-Augmented Generation* (RAG) yang terdiri dari dua tahapan utama: *Indexing* dan Pemodelan Sistem.

Tahap pertama adalah *Indexing*, yaitu proses penyimpanan data yang memungkinkan sistem melakukan pencarian informasi secara efisien. Proses ini mencakup:

1. Pengumpulan Data : Artikel dikumpulkan secara manual berdasarkan tiga topik utama, yaitu *Supervised Learning, Unsupervised Learning*, dan *Reinforcement Learning* sebagai sumber kajian literatur.
2. Ekstraksi Teks : File PDF diubah menjadi teks menggunakan pustaka *PyMuPDF* untuk memudahkan pemrosesan lebih lanjut.
3. Pembagian Struktur IMRAD : Teks hasil ekstraksi dibagi menjadi *Introduction, Methods, Results*, dan *Discussion* untuk memisahkan informasi sesuai struktur penulisan ilmiah.
4. Peringkasan Teks : Setiap bagian IMRAD diringkas menggunakan model *IndoT5-base* dengan pendekatan *abstractive summarization*. Contohnya, bagian *Introduction* yang semula 250 kata dapat dipadatkan menjadi ±60 kata tanpa kehilangan makna inti.
5. Evaluasi Ringkasan : Ringkasan setiap bagian dibandingkan dengan abstrak asli jurnal menggunakan *BERTScore*, untuk mengukur kesamaan semantik berdasarkan nilai *Precision, Recall*, dan *F1-score*.
6. *Chunking* dan *Embedding* : Ringkasan digabungkan menjadi satu dokumen, lalu di-*chunk* (dipotong) menjadi potongan teks yang lebih kecil. Setiap chunk dikonversi menjadi vektor *embedding* berukuran 768 dimensi menggunakan *IndoBERT-base*.
7. Penyimpanan ke FAISS : Vektor hasil *embedding* disimpan ke FAISS (*Facebook AI Similarity Search*) tipe *Flat Index* dengan parameter *top-k retrieval* = 5, untuk memfasilitasi pencarian dokumen relevan secara cepat dan akurat.

Tahap kedua adalah Pemodelan Sistem. Proses ini dimulai ketika pengguna menginput *query*. *Query* tersebut di-*embedding* menggunakan *IndoBERT-base* (sama seperti tahap *indexing*), kemudian dilakukan *similarity search* di FAISS untuk menemukan lima dokumen dengan tingkat kesamaan tertinggi. Dokumen relevan ini digabungkan dengan *query* pengguna untuk membentuk

augmented context. Konteks ini kemudian diproses oleh *LLaMA-4-Scout-17B-Instruct* melalui API Groq dengan parameter temperature 0,7 untuk menjaga keseimbangan kreativitas dan akurasi jawaban. *Output* berupa jawaban yang relevan ditampilkan kepada pengguna, disertai ringkasan lima dokumen pendukung sebagai *evidence* untuk meminimalkan potensi *hallucination*.

D. HASIL DAN PEMBAHASAN

1. Hasil Evaluasi Sistem

a. Evaluasi Kualitas Ringkasan menggunakan *BERTScore*

Evaluasi kualitas ringkasan dilakukan dengan membandingkan hasil ringkasan otomatis terhadap abstrak asli dari artikel ilmiah. Pengukuran dilakukan menggunakan metrik *BERTScore*, yang menilai kesamaan semantik berdasarkan tiga indikator utama: *Precision, Recall*, dan *F1-Score*. Tabel 1 menunjukkan contoh hasil evaluasi terhadap lima artikel jurnal yang berbeda.

Tabel 1. Sampel Hasil Evaluasi Ringkasan menggunakan *BERTScore*

No	Judul Jurnal	Precision	Recall	F1-Score
1	Peringkasan Teks Multi Dokumen Berbahasa Indonesia	0.7983	0.8048	0.8015
2	*Text Summarization* pada Artikel Berita	0.8079	0.8492	0.8280
3	Peringkasan Artikel Berita Pendekatan RNN	0.8276	0.8702	0.8484
4	*Clustering* Peringkasaan Dokumen Berita	0.8207	0.8805	0.8496
5	Peringkasan Artikel Berbahasa Indonesia Metode TF-IDF	0.8263	0.8517	0.8388

Dari 100 artikel yang diuji, diperoleh rata-rata skor *BERTScore* sebesar *Precision* 0.828, *Recall* 0.881, dan *F1-Score* 0.854. Hasil ini menunjukkan bahwa model IndoT5 yang digunakan mampu menghasilkan ringkasan yang relevan secara semantik dengan teks referensi, mendekati makna abstrak yang sebenarnya.

b. Hasil Evaluasi Sistem RAG menggunakan metode *LLM-as-a-Judge*

Evaluasi sistem terhadap pertanyaan dilakukan dengan pendekatan *LLM-as-a-Judge*, di mana model *LLaMA-3-70B* digunakan untuk memberikan skor terhadap jawaban sistem berdasarkan tiga aspek: Relevansi, *Faithfulness* (kebenaran), dan Kelengkapan. Tabel 2 menyajikan lima contoh hasil evaluasi.

Tabel 2. Hasil Evaluasi Jawaban Sistem Menggunakan *LLM-as-a-Judge*

No	Pertanyaan	Relevansi	kebenaran	Kelengkapan	Rata-rata
1	Dalam konteks klasifikasi judul skripsi, algoritma apa yang cocok?	5	5	5	5.00
2	Bagaimana model *supervised learning* digunakan dalam prediksi data medis?	5	5	5	5.00
3	Bagaimana pendekatan *clustering* digunakan untuk menganalisis dokumen?	5	5	5	5.00
4	Apa peran PCA dalam mendukung proses *clustering*?	5	5	5	5.00
5	Apa yang dibahas dalam literatur review tentang *supervised learning*?	5	5	5	5.00
6	Dalam konteks klasifikasi judul skripsi, algoritma mana yang diprioritaskan?	5	5	5	5.00
7	Bagaimana metode DBSCAN digunakan untuk identifikasi pola data?	5	5	5	5.00
8	Bagaimana metode klasifikasi pada "Deteksi *Malware* Berbasis AI"?	5	5	4	4.67

9	Jelaskan bagaimana algoritma *Epsilon-Greedy* digunakan dalam *reinforcement learning*?	5	5	4	4.67
10	Bagaimana *Unity-Gymnasium* digunakan sebagai simulator *reinforcement learning*?	5	5	4	4.67
11	Jelaskan pendekatan simulasi *reinforcement learning* untuk menyelesaikan permasalahan kompleks!	5	5	4	4.67
12	Apa *insight* yang diperoleh dari studi simulasi *reinforcement learning*?	5	5	4	4.67
13	Metode apa yang bagus untuk klasifikasi objek?	5	5	4	4.67
14	Metode apa yang bagus untuk klasifikasi gambar medis?	5	5	4	4.67
15	Metode apa saja yang cocok untuk klasifikasi data besar?	5	5	4	4.67
16	Metode apa saja yang cocok untuk jenis *reinforcement learning* tertentu?	5	5	4	4.67
17	Metode apa saja yang bagus untuk jenis pembelajaran *clustering*?	5	5	4	4.67
18	Bagaimana *supervised learning* diterapkan dalam deteksi emosi?	5	5	4	4.67
19	Bagaimana algoritma *epsilon-greedy* digunakan dalam sistem rekomendasi?	5	5	4	4.67
20	Apa saja pendekatan *supervised learning* yang digunakan dalam prediksi data?	5	5	4	4.67

Dari total 20 pertanyaan yang diuji, sistem memperoleh rata-rata skor keseluruhan 4.78 dari skala 5, dengan skor tertinggi pada aspek relevansi. Hasil ini menunjukkan bahwa sistem mampu menghasilkan jawaban yang umumnya tepat, kontekstual, dan informatif.

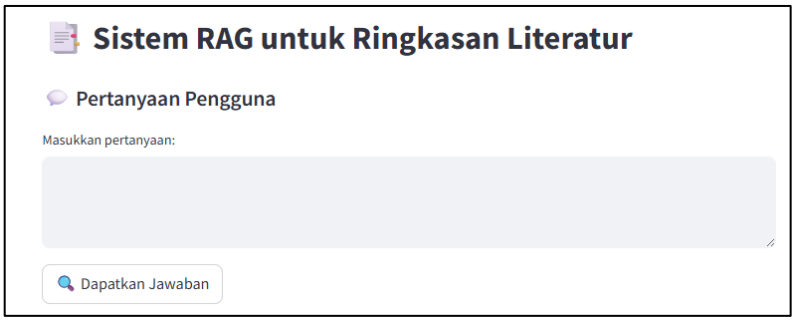
2. Hasil Implementasi Sistem

a. Hasil Perancangan *User Interface*

Antarmuka sistem terdiri dari dua bagian utama, yaitu sisi admin dan sisi *user*. Tampilan halaman sisi admin dapat dilihat pada Gambar 2.

Gambar 2. Tampilan Sistem Sisi Admin

Sementara itu, halaman sisi *user* menampilkan fitur untuk memasukkan pertanyaan dan menghasilkan jawaban. Tampilan sisi *user* ditunjukkan pada Gambar 3.



Gambar 3. Tampilan Sistem Sisi *User*

Gambar 2 adalah *interface* sisi admin yang dirancang untuk membantu admin memasukkan dataset secara otomatis. Halaman ini mengandung beberapa komponen penting yang mendukung proses *input* dokumen sampai masuk ke *database* vektor. Komponen yang ada di halaman sisi admin ini adalah Unggah PDF, admin dapat mengunggah satu file PDF sumber referensi yang terkait dengan topik. Teks dari file PDF ini akan diekstraksi dan diproses hingga dimasukkan ke dalam *database* vektor.

Gambar 3 adalah *interface* sisi *user* yang dirancang untuk membantu *user input* pertanyaan. Halaman ini mengandung beberapa komponen penting yang mendukung proses *input* pertanyaan hingga sistem *generate* jawaban. Komponen-komponen yang ada pada halaman sisi *user* diuraikan di bawah ini:

- *Input* Pertanyaan/*Prompt*

  User diminta untuk memasukkan satu atau lebih *prompt* atau pertanyaan dalam *text area*. Pertanyaan ini akan digunakan untuk membuat jawaban yang relevan. *User* diusahakan membuat pertanyaan/*prompt* yang lebih spesifik mengenai topik yang ingin dihasilkan agar sesuai dengan kebutuhan *user*.
- Tombol *Generate*

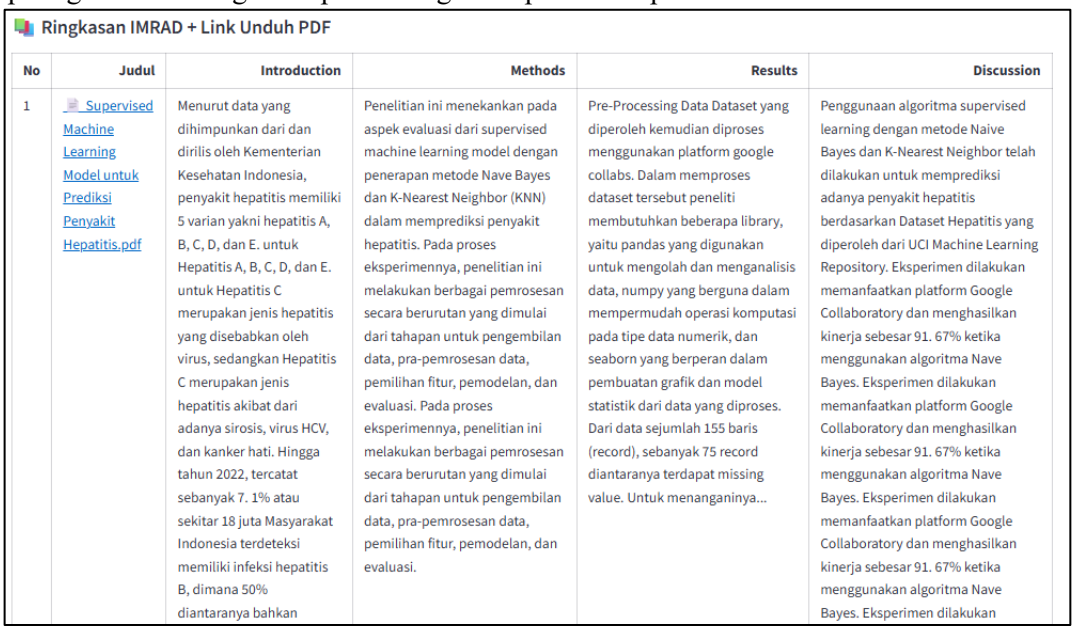
  Setelah semua informasi dimasukkan, *user* dapat menekan tombol "Dapatkan Jawaban" untuk memulai proses pembuatan jawaban. Model RAG akan digunakan oleh sistem untuk mengumpulkan informasi yang relevan dari jurnal-jurnal relevan yang telah di-*embed*. Selanjutnya, menggunakan *llm-4-scout-17b-instruct* untuk membuat jawaban berdasarkan pertanyaan yang diberikan.

b. Hasil *Generate* Jawaban

Hasil pemrosesan sistem dalam menjawab pertanyaan pengguna ditunjukkan pada Gambar 5. Tampilan ini menunjukkan keluaran teks yang dihasilkan oleh model LLM setelah melalui tahap retrieval dan augmentasi konteks.

Gambar 4 Hasil *Generate* LLM

Selain jawaban utama, sistem juga menampilkan ringkasan dari lima dokumen yang paling relevan. Ringkasan pendukung ini dapat dilihat pada Gambar 6.



Ringkasan IMRAD + Link Unduh PDF

No	Judul	Introduction	Methods	Results	Discussion
1	Supervised Machine Learning Model untuk Prediksi Penyakit Hepatitis.pdf	Menurut data yang dihimpunkan dari dan dirilis oleh Kementerian Kesehatan Indonesia, penyakit hepatitis memiliki 5 varian yakni hepatitis A, B, C, D, dan E. untuk Hepatitis A, B, C, D, dan E. untuk Hepatitis C merupakan jenis hepatitis yang disebabkan oleh virus, sedangkan Hepatitis C merupakan jenis hepatitis akibat dari adanya sirosis, virus HCV, dan kanker hati. Hingga tahun 2022, tercatat sebanyak 7. 1% atau sekitar 18 juta Masyarakat Indonesia terdeteksi memiliki infeksi hepatitis B, dimana 50% diantaranya bahkan	Penelitian ini menekankan pada aspek evaluasi dari supervised machine learning model dengan penerapan metode Nave Bayes dan K-Nearest Neighbor (KNN) dalam memprediksi penyakit hepatitis. Pada proses eksperimennya, penelitian ini melakukan berbagai pemrosesan secara berurutan yang dimulai dari tahapan untuk pengembilan data, pra-pemrosesan data, pemilihan fitur, pemodelan, dan evaluasi. Pada proses eksperimennya, penelitian ini melakukan berbagai pemrosesan secara berurutan yang dimulai dari tahapan untuk pengembilan data, pra-pemrosesan data, pemilihan fitur, pemodelan, dan evaluasi.	Pre-Processing Data Dataset yang diperoleh kemudian diproses menggunakan platform google collabs. Dalam memproses dataset tersebut peneliti membutuhkan beberapa library, yaitu pandas yang digunakan untuk mengolah dan menganalisis data, numpy yang berguna dalam mempermudah operasi komputasi pada tipe data numerik, dan seaborn yang berperan dalam pembuatan grafik dan model statistik dari data yang diproses. Dari data sejumlah 155 baris (record), sebanyak 75 record diantaranya terdapat missing value. Untuk menanganinya...	Penggunaan algoritma supervised learning dengan metode Naive Bayes dan K-Nearest Neighbor telah dilakukan untuk memprediksi adanya penyakit hepatitis berdasarkan Dataset Hepatitis yang diperoleh dari UCI Machine Learning Repository. Eksperimen dilakukan memanfaatkan platform Google Collaboratory dan menghasilkan kinerja sebesar 91. 67% ketika menggunakan algoritma Nave Bayes. Eksperimen dilakukan memanfaatkan platform Google Collaboratory dan menghasilkan kinerja sebesar 91. 67% ketika menggunakan algoritma Nave Bayes. Eksperimen dilakukan memanfaatkan platform Google Collaboratory dan menghasilkan kinerja sebesar 91. 67% ketika menggunakan algoritma Nave Bayes. Eksperimen dilakukan

Gambar 5 Menampilkan Salah Satu Ringkasan Dokumen yang Relevan

Hasil *generate* jawaban yang ditampilkan pada Gambar 5 merupakan respons yang dihasilkan sistem setelah menerima pertanyaan pengguna. Proses ini dimulai dengan pencarian dokumen relevan melalui *database* vektor FAISS menggunakan *embedding IndoBERT*. Dokumen yang relevan kemudian dikombinasikan dengan pertanyaan pengguna untuk membentuk konteks *augmented*. Konteks ini selanjutnya diproses oleh model LLM (*LLaMA-4-Scout-17B-Instruct*) untuk menghasilkan jawaban yang sesuai. LLM menghasilkan jawaban berdasarkan konteks ini, sehingga informasi yang disajikan bersumber dari dokumen terverifikasi. Misalnya, untuk pertanyaan "Apa kelebihan penggunaan IndoT5 dalam sistem ini?", sistem menjawab dengan menjelaskan keunggulan IndoT5 dalam memahami struktur Bahasa Indonesia dan kemampuannya mempertahankan esensi teks.

Selain menampilkan jawaban utama, sistem juga menampilkan ringkasan dari lima dokumen paling relevan (Gambar 6). Fitur ini memberi nilai tambah karena pengguna tidak hanya mendapatkan jawaban singkat, tetapi juga ringkasan pendukung yang membantu memverifikasi kebenaran dan kelengkapan informasi.

Hasil peringkasan yang ditunjukkan pada Gambar 4 merupakan *output* dari model IndoT5 *pretrained* yang telah dilatih khusus untuk Bahasa Indonesia. Proses peringkasan dilakukan setelah dokumen ilmiah dibagi berdasarkan struktur IMRAD, sehingga setiap bagian *Introduction, Methods, Results,* dan *Discussion*, diringkas secara terpisah. Pendekatan ini memastikan bahwa informasi kunci dari tiap bagian tetap terjaga dan tidak bercampur dengan konteks dari bagian lain. Ringkasan yang dihasilkan bersifat abstraktif, artinya model membentuk kalimat baru yang ringkas namun tetap mempertahankan makna inti, bukan sekadar menyalin potongan kalimat dari sumber. Hal ini membuat hasil ringkasan lebih mudah dipahami, relevan, dan fokus pada inti penelitian.

3. Pembahasan

Integrasi metode *Retrieval-Augmented Generation* (RAG) dengan model IndoT5 dan struktur IMRAD pada penelitian ini mampu memberikan solusi efektif untuk otomatisasi ringkasan literatur ilmiah berbahasa Indonesia. Pendekatan ini memanfaatkan segmentasi dokumen menjadi bagian *Introduction, Methods, Results,* dan *Discussion*, sehingga proses peringkasan lebih terarah dan tetap mempertahankan esensi setiap bagian.

Efektivitas sistem terkonfirmasi melalui evaluasi *BERTScore*, yang menghasilkan nilai rata-rata *Precision* 0,828, *Recall* 0,881, dan *F1-score* 0,854—menunjukkan tingkat kesamaan semantik tinggi antara ringkasan sistem dan abstrak asli. Evaluasi jawaban sistem menggunakan *LLM-as-a-Judge* juga memperoleh skor rata-rata 4,78 dari skala 5 pada aspek relevansi, kebenaran (*faithfulness*), dan kelengkapan. Hasil ini membuktikan bahwa sistem tidak hanya mampu

merangkum literatur, tetapi juga memberikan jawaban yang kontekstual dan informatif terhadap pertanyaan pengguna.

Meskipun performa sistem cukup menjanjikan, terdapat beberapa keterbatasan yang teridentifikasi: (1) model IndoT5 belum di-*fine-tune* secara khusus pada domain akademik, sehingga kualitas ringkasan masih dapat ditingkatkan, dan (2) jawaban sistem kadang kurang lengkap atau tidak eksplisit ketika dokumen relevan tidak memuat informasi yang dibutuhkan. Selain itu, ketergantungan pada API eksternal berpotensi menjadi hambatan jika infrastruktur akses terbatas.

Secara keseluruhan, integrasi RAG, IndoT5, dan peringkasan berbasis struktur IMRAD terbukti meningkatkan efisiensi kajian literatur berbahasa Indonesia. Sistem yang dikembangkan tidak hanya membantu peneliti menghemat waktu, tetapi juga menjadi alat pendukung dalam penyusunan kajian literatur yang terstruktur, relevan, dan mudah dipahami.

E. KESIMPULAN DAN SARAN

Penelitian ini menghasilkan sistem otomatisasi kajian literatur berbahasa Indonesia dengan mengintegrasikan *Retrieval-Augmented Generation* (RAG), model IndoT5, dan pendekatan struktur IMRAD. Sistem yang dikembangkan mampu merangkum literatur akademik secara otomatis, menyajikan informasi secara terstruktur, dan memberikan jawaban yang relevan terhadap pertanyaan pengguna.

Penerapan IndoT5 untuk merangkum setiap bagian IMRAD terbukti efektif dalam mempertahankan inti informasi, sekaligus mempermudah proses *indexing* dan pencarian berbasis semantik menggunakan FAISS dan *IndoBERT*. Evaluasi menggunakan *BERTScore* menunjukkan rata-rata *Precision* 0,828, *Recall* 0,881, dan *F1-score* 0,854, yang mencerminkan kesamaan semantik tinggi dengan abstrak asli. Selain itu, penilaian menggunakan *LLM-as-a-Judge* menghasilkan skor rata-rata 4,78 dari skala 5 pada aspek relevansi, kebenaran, dan kelengkapan jawaban.

Hasil ini menunjukkan bahwa integrasi RAG, peringkasan menggunakan model IndoT5 dengan struktur IMRAD dapat mempercepat proses kajian literatur, meningkatkan akurasi ringkasan, serta menyediakan jawaban yang kontekstual dan informatif. Meskipun demikian, sistem masih memiliki keterbatasan seperti panjang *input* yang terbatas, potensi *hallucination*, dan ketergantungan pada API eksternal, yang menjadi peluang perbaikan pada penelitian selanjutnya.

Berdasarkan hasil dalam penelitian ini, penulis memberikan beberapa saran yang dapat dijadikan masukan untuk pengembangan lebih lanjut, diantaranya yaitu:

1. Melakukan *fine-tuning* model IndoT5 menggunakan korpus jurnal akademik Indonesia dari berbagai disiplin ilmu untuk meningkatkan akurasi dan cakupan ringkasan.
2. Menambahkan fitur verifikasi otomatis untuk meminimalkan risiko *hallucination*, seperti *reranking* atau perbandingan *output* terhadap isi dokumen.
3. Mengintegrasikan *pipeline* otomatis dari unggah dokumen hingga proses *indexing* untuk efisiensi sistem secara menyeluruh.
4. Menambahkan dukungan input *multi-modal*, seperti grafik dan tabel, melalui integrasi OCR atau ekstraksi tabel otomatis.
5. Menggunakan evaluator lokal berbasis *open-source* agar sistem dapat digunakan secara *offline* tanpa ketergantungan API eksternal.
6. Memperluas jumlah dan ragam dokumen dalam dataset serta menguji sistem terhadap domain non-AI untuk mengukur generalisasi.
7. Melakukan evaluasi lanjutan menggunakan dataset *benchmark* agar performa sistem dapat dibandingkan secara objektif dalam skala akademik.

DAFTAR PUSTAKA

Cahyawijaya, S., Winata, G. I., Wilie, B., Vincentio, K., Li, X., Kuncoro, A., Ruder, S., Lim, Z. Y., Bahar, S., Khodra, M. L., Purwarianti, A., & Fung, P. (2021). *IndoNLG: Benchmark and Resources for Evaluating Indonesian Natural Language Generation*. http://arxiv.org/abs/2104.08200

Cheng, M., Luo, Y., Ouyang, J., Liu, Q., Liu, H., Li, L., Yu, S., Zhang, B., Cao, J., Ma, J., Wang, D., & Chen, E. (2025). *A Survey on Knowledge-Oriented Retrieval-Augmented Generation*. http://arxiv.org/abs/2503.10677

Gu, J., Jiang, X., Shi, Z., Tan, H., Zhai, X., Xu, C., Li, W., Shen, Y., Ma, S., Liu, H., Wang, S., Zhang, K., Wang, Y., Gao, W., Ni, L., & Guo, J. (2024). *A Survey on LLM-as-a-Judge*. http://arxiv.org/abs/2411.15594

Gupta, S., & Ranjan, R. (2024). *A Comprehensive Survey of Retrieval-Augmented Generation (RAG): Evolution, Current Landscape and Future Directions*.

Hahsler, M. (2023). *ARULESPY: Exploring Association Rules and Frequent Itemsets in Python*. http://arxiv.org/abs/2305.15263

Handoyo, S., Prastiti, P. I. D., & Stiaji, I. R. (2024). Bibliometric analysis of publications trends in Indonesian research institutions: A comparison of pre-integration (2015–2021) and post-integration (2022–2023) periods. *European Science Editing*, *50*. https://doi.org/10.3897/ese.2024.e118015

Jaber, E. A., & Gérard, L.-A. (2025). *Signature volatility models: pricing and hedging with Fourier*. https://doi.org/10.1137/24M1636952

Li, D., Jiang, B., Huang, L., Beigi, A., Zhao, C., Tan, Z., Bhattacharjee, A., Jiang, Y., Chen, C., Wu, T., Shu, K., Cheng, L., & Liu, H. (2024). *From Generation to Judgment: Opportunities and Challenges of LLM-as-a-judge*. http://arxiv.org/abs/2411.16594

Muhammad, T., Rahardiansyah, R., Setya Perdana, R., & Fatyanosa, T. N. (2025). *Analisis Teknik Embedding Model NV-Embed pada Large Language Models Berbasis Retrieval Augmented Generation* (Vol. 9, Nomor 2). http://j-ptiik.ub.ac.id

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. Dalam *Journal of Machine Learning Research* (Vol. 21). http://jmlr.org/papers/v21/20-074.html.

Ridwan, M., Ulum, B., Muhammad, F., Indragiri, I., & Sulthan Thaha Saifuddin Jambi, U. (2021). *Pentingnya Penerapan Literature Review pada Penelitian Ilmiah (The Importance Of Application Of Literature Review In Scientific Research)*. http://journal.fdi.or.id/index.php/jmas/article/view/356

Sakti Wiradinata, A., Viny,), & Mawardi, C. (2024). *Jurnal Ilmu Komputer dan Sistem Informasi Abstractive Text Summarization Berita Bahasa Indonesia Menggunakan Retrieval-Augmented Generation*. https://www.cnbcindonesia.com/indeks

Shuliang Liu, Xinze Li, Zhenghao Liu, Yukun Yan, Cheng Yang, Zheni Zeng, Zhiyuan Liu, Maosong Sun, & Ge Yu. (2025). *Judge as A Judge: Improving the Evaluation of Retrieval-Augmented Generation through the Judge-Consistency of Large Language Models*.

Sollaci, L. B., & Pereira, M. G. (2004). The introduction, methods, results, and discussion (IMRAD) structure: a fifty-year survey. Dalam *J Med Libr Assoc* (Vol. 92, Nomor 3).

Vaswani, A., Brain, G., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). *Attention Is All You Need*.

Wilie, B., Vincentio, K., Winata, G. I., Cahyawijaya, S., Li, X., Lim, Z. Y., Soleman, S., Mahendra, R., Fung, P., Bahar, S., & Purwarianti, A. (2020). *IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding*. http://arxiv.org/abs/2009.05387

Yani, M., Siti Khodijah, N., & Mustamiin, M. (2024). *Aplikasi Peringkas Teks Bahasa Indonesia Menggunakan Model Text-to-Text Transfer Transformer (T5)*. https://doi.org/10.37817/ikraith-informatika.v9i2

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2019). *BERTScore: Evaluating Text Generation with BERT*. http://arxiv.org/abs/1904.09675

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., & Stoica, I. (2023). *Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena*. http://arxiv.org/abs/2306.05685