

Implementasi Chatbot Informasi Akademik Berbasis Retrieval-Augmented Generation (RAG) Menggunakan LLaMA 3.1

[Implementation of an Academic Information Chatbot Based on Retrieval-Augmented Generation (RAG) Using LLaMA 3.1]

Muhammad Saddam Heykal Bustomy ¹⁾, Mochamad Alfian Rosid ^{*2)}, Suprianto ³⁾, Hamzah Setiawan ⁴⁾

^{1,2,3,4)}Program Studi Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia

*Email: alfanrosid@umsida.ac.id

Abstract. Providing fast and accurate academic information is a fundamental requirement for universities. However, users often experience difficulties in efficiently searching for specific information on study program websites. This study aims to develop a Retrieval-Augmented Generation (RAG)-based academic information chatbot that is directly integrated with the Informatics Study Program website at Muhammadiyah University of Sidoarjo. The system was built using the LangGraph architecture, ChromaDB vector database, and LLaMA 3.1 model through the Groq API. Testing was conducted using the RAGAS framework to assess the quality of answers and load testing for system performance. The results showed that the system achieved a Context Precision score of 0.86 and a Faithfulness score of 0.77, indicating high relevance and accuracy of answers. In addition, the implementation of the Groq API resulted in an average response latency of 2.11 seconds with a 94% success rate in load testing. These findings indicate that the RAG-based chatbot is effective in overcoming website limitations in delivering academic information at the Informatics Study Program at Muhammadiyah University of Sidoarjo.

Keywords - Academic Chatbot; Retrieval-Augmented Generation (RAG); LLaMA 3.1; Groq API; LangGraph;

Abstrak. Penyediaan informasi akademik yang cepat dan akurat merupakan kebutuhan mendasar bagi perguruan tinggi. Namun, pengguna sering kali mengalami kesulitan dalam mencari informasi spesifik secara efisien pada web program studi. Penelitian ini bertujuan untuk mengembangkan chatbot informasi akademik berbasis Retrieval-Augmented Generation (RAG) yang terintegrasi langsung dengan situs web Program Studi Informatika Universitas Muhammadiyah Sidoarjo. Sistem dibangun menggunakan arsitektur LangGraph, basis data vektor ChromaDB, dan model LLaMA 3.1 melalui Groq API. Pengujian dilakukan menggunakan kerangka kerja RAGAS untuk menilai kualitas jawaban dan pengujian beban untuk kinerja sistem. Hasil penelitian menunjukkan bahwa sistem mencapai skor Context Precision sebesar 0.86 dan Faithfulness sebesar 0.77 yang mengindikasikan tingginya relevansi dan akurasi jawaban. Selain itu, penerapan Groq API menghasilkan rata-rata latensi respon sebesar 2.11 detik dengan tingkat keberhasilan 94% pada pengujian beban. Temuan ini menunjukkan bahwa chatbot berbasis RAG efektif dalam mengatasi keterbatasan situs web dalam penyampaian informasi akademik Program Studi Informatika Universitas Muhammadiyah Sidoarjo.

Kata Kunci - Chatbot Akademik; Retrieval-Augmented Generation; LLaMA 3.1; Groq API; LangGraph;

I. PENDAHULUAN

Penyediaan informasi akademik yang cepat dan akurat merupakan kebutuhan mendasar dalam dunia pendidikan, termasuk di lingkungan Program Studi Informatika Universitas Muhammadiyah Sidoarjo (UMSIDA) [1]. Seiring berkembangnya Artificial Intelligence (AI), pengguna memiliki berbagai kemudahan dalam mencari sumber informasi digital maupun materi pembelajaran dari internet [2]. Program Studi Informatika UMSIDA telah memanfaatkan platform digital seperti situs web resmi, media sosial dan dokumen daring untuk menyampaikan informasi akademik ke mahasiswa [3]. Namun, pada kenyataannya pengguna masih sering mengalami kesulitan dalam mencari informasi yang dibutuhkan secara efisien, khususnya pada informasi mengenai praktikum yang paling sering dicari. Informasi yang sebenarnya sudah tersedia secara lengkap di situs web resmi program studi sering kali ditanyakan kepada pihak administrasi ataupun dosen terkait. Kondisi ini terjadi karena mahasiswa enggan menelusuri banyak halaman web atau tidak mengetahui letak informasi yang relevan.

Permasalahan tersebut menunjukkan bahwa tantangan utama layanan informasi akademik di Program Studi Informatika UMSIDA tidak hanya terletak pada ketersediaan data, tetapi juga mekanisme penyampaian dan kemudahan akses informasi [4]. Website statis memiliki keterbatasan dalam mendukung interaksi dua arah atau pencarian informasi secara kontekstual [5]. Pengguna harus membaca dan menelusuri banyak halaman untuk memperoleh jawaban yang spesifik. Oleh karena itu, diperlukan sebuah sistem yang mampu menyajikan informasi akademik secara interaktif, responsif, dan sesuai dengan kebutuhan pengguna [6]. Salah satu solusi yang relevan dengan permasalahan tersebut adalah pemanfaatan chatbot berbasis *Artificial Intelligence* (AI) [7].

Chatbot merupakan sistem interaktif yang memungkinkan pengguna berkomunikasi dengan sistem melalui pesan teks [8]. Dalam konteks layanan akademik di Program Studi Informatika UMSIDA, chatbot dapat berperan sebagai asisten virtual yang membantu pengguna dalam memperoleh informasi secara efisien [9]. Seiring

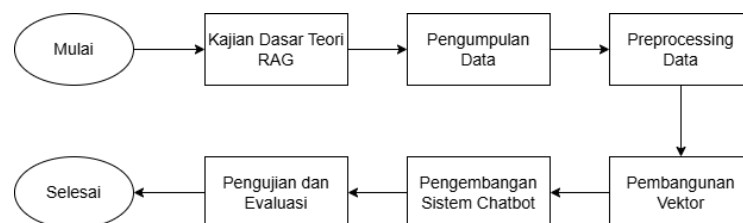
berkembangnya *Large Language Model* (LLM), kemampuan chatbot dalam memahami pertanyaan dan menghasilkan jawaban secara natural semakin meningkat [10]. Namun, LLM cenderung mengalami keterbatasan dalam tugas – tugas yang membutuhkan pengetahuan faktual, karena model hanya mengandalkan parameter internal hasil pelatihan sebelumnya [11].

Beberapa penelitian terdahulu mengusulkan pendekatan Retrieval-Augmented Generation (RAG), yaitu metode yang menggabungkan kemampuan pencarian dokumen (retrieval) dengan kemampuan generatif Large Language Model (LLM) [12]. Penelitian oleh Denina Milasanti [13] mengembangkan chatbot berbasis RAG untuk pencarian artikel ilmiah pada portal GARUDA, sementara penelitian oleh Harifa Nur'aeni [14] menerapkan chatbot RAG untuk layanan informasi pada perpustakaan digital. Kedua penelitian tersebut menggunakan metrik ROUGE sebagai alat evaluasi kinerja chatbot yang dinilai “cukup baik”, meskipun skor numerik ROUGE yang diperoleh relatif rendah. Selain itu, penelitian oleh Hidayat et al. [15] mengembangkan chatbot akademik berbasis RAG yang terintegrasi dengan platform Telegram dan menunjukkan peningkatan akurasi jawaban, namun masih menghadapi kendala pada latensi respon. Secara umum, penelitian-penelitian tersebut menunjukkan potensi chatbot RAG dalam mendukung layanan informasi akademik, namun juga mengindikasikan adanya tantangan pada aspek evaluasi, efisiensi, dan integrasi layanan.

Berdasarkan penelitian-penelitian terdahulu tersebut, dapat diidentifikasi beberapa celah penelitian. Pertama, penggunaan metrik ROUGE menunjukkan adanya potensi ketidaksesuaian antara metrik evaluasi yang digunakan dengan karakteristik tugas tanya-jawab, mengingat ROUGE pada dasarnya dirancang untuk tugas peringkasan teks [16]. Kedua, beberapa penelitian masih mengandalkan platform pihak ketiga atau aplikasi terpisah, sehingga akses pengguna terhadap chatbot belum terintegrasi langsung dengan website resmi Program Studi Informatika UMSIDA. Ketiga, aspek latensi respon dan efisiensi sistem masih menjadi kendala, terutama pada penggunaan LLM proprietary dengan biaya dan waktu respon yang relatif tinggi. Oleh karena itu, penelitian ini berfokus pada pengembangan chatbot informasi akademik berbasis RAG yang terintegrasi langsung dengan website Program Studi Informatika UMSIDA, memanfaatkan sumber data resmi yang terstruktur dalam format JSON, serta menggunakan model LLaMA 3.1 melalui Groq API untuk mencapai waktu respon yang lebih cepat. Pendekatan ini diharapkan mampu menghasilkan sistem chatbot yang lebih responsif, relevan, dan sesuai dengan kebutuhan layanan informasi akademik.

II. METODE PENELITIAN

Metodologi penelitian ini dirancang untuk membangun sistem chatbot informasi akademik program studi Informatika berbasis Retrieval-Augmented Generation (RAG). Tahapan penelitian meliputi studi literatur, pengumpulan data, preprocessing data, pembangunan vektor, perancangan alur kerja chatbot, serta evaluasi kualitas jawaban. Seluruh alur penelitian disusun secara sistematis dan divisualisasikan dalam bentuk diagram alur pada Gambar 2.1.



Gambar 2.1 Tahapan Penelitian

2.1 Kajian Dasar Teori Retrieval-Augmented Generation (RAG)

Sebelum melakukan pengumpulan data, penelitian ini diawali dengan kajian dasar teori *Retrieval-Augmented Generation* (RAG) sebagai landasan konseptual dalam pengembangan chatbot informasi akademik. Kajian ini difokuskan pada pemahaman prinsip kerja RAG yang mengombinasikan proses pencarian dokumen (*retrieval*) dan generasi jawaban (*generation*) menggunakan *Large Language Model*. Pemahaman dasar ini diperlukan untuk menentukan alur kerja sistem, pemilihan komponen utama, serta strategi integrasi RAG dalam chatbot agar mampu menghasilkan jawaban yang relevan, akurat, dan bersumber dari data resmi [12]. Hasil kajian dasar teori ini menjadi acuan dalam pelaksanaan tahap pengumpulan data dan pengembangan sistem selanjutnya.

2.2 Pengumpulan Data

Data diperoleh dari dua sumber utama, yaitu situs web resmi Program Studi Informatika dan dokumen PDF yang berisi informasi umum mengenai program studi. Pengambilan data dari website dilakukan dengan teknik web scrapping menggunakan pustaka bawaan dari LangChain seperti WebBaseLoader untuk mengekstrak teks dari

halaman-halaman yang memuat informasi akademik [17]. Sementara itu, data dari dokumen PDF diperoleh dari file resmi seperti buku panduan akademik.

2.3 Preprocessing Data

Seluruh data yang diperoleh, baik dari website maupun dokumen PDF, dikonversi ke dalam format data terstruktur berbasis JSON. Setiap entri data disusun dalam bentuk `page_content` yang berisi teks utama, dan metadata yang mencakup informasi sumber, kategori, judul, dan url. Proses ini dilakukan untuk menyeragamkan format data dari berbagai sumber dengan memisahkan elemen-elemen penting, seperti jenis sumber, judul dokumen, dan isi konten. Pendekatan ini memudahkan pengolahan data serta meningkatkan konsistensi pada tahap pemrosesan selanjutnya. Contoh hasil preprocessing dan konversi dalam struktur JSON ditunjukkan pada Tabel 2.1. Selanjutnya, data dalam format JSON diproses menjadi potongan-potongan teks (*chunk*) dengan ukuran maksimal 4000 karakter dan overlap sebesar 200 karakter. Pemotongan ini dilakukan untuk menjaga kesinambungan konteks antar potongan teks serta mengurangi risiko hilangnya informasi penting pada saat pembentukan vektor embedding [18].

Tabel 2.1 Contoh Struktur Data JSON

Elemen JSON	Isi
<code>page_content</code>	Layanan Administrasi Surat Online Informatika. Mahasiswa dapat mengajukan Surat Aktif Kuliah, Surat Magang, Penelitian, dan Tugas Akhir melalui formulir ini.
<code>metadata.source</code>	https://informatika.umsida.ac.id/layanan-kemahasiswaan/
<code>metadata.category</code>	Layanan
<code>metadata.title</code>	Formulir Surat Online
<code>metadata.direct_link</code>	https://docs.google.com/forms/d/1ijKTVs1T546WU__zqqEc9JeSfj2HoGVWFqhcGaJr-bs/viewform?edit_requested=true

2.4 Pembangunan Vektor

Setelah tahap preprocessing selesai, potongan teks diubah menjadi representasi numerik (*vector embedding*) yang dapat digunakan untuk pencarian semantik [19]. Proses embedding ini memanfaatkan model *nomic-embed-text* [20] karena kemampuannya menghasilkan vektor yang mampu menangkap makna dan konteks secara presisi dan kompatibel dengan arsitektur sistem yang dikembangkan. Proses pengubahan teks menjadi representasi vektor dapat dilihat pada Tabel 2.2

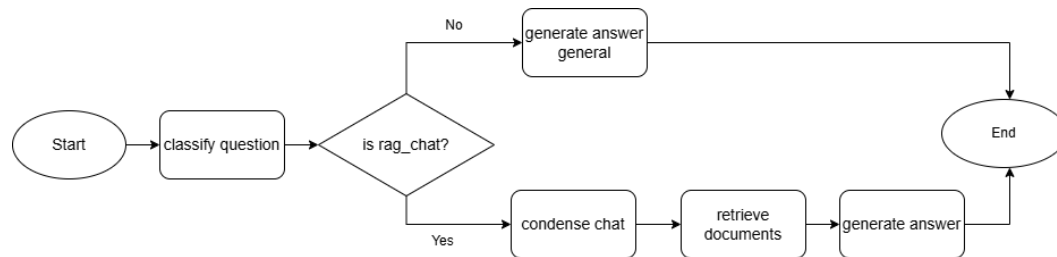
Tabel 2.2 Contoh Konversi Data JSON ke Representasi Vektor

Komponen	Contoh
<code>page_content (JSON)</code>	Layanan Administrasi Surat Online Informatika. Mahasiswa dapat mengajukan Surat Aktif Kuliah, Surat Magang, Penelitian, dan Tugas Akhir melalui formulir ini.
Metadata	Layanan, Formulir Surat Online
Vektor Embedding	[0.021, -0.134, 0.587, ..., 0.042]

Hasil embedding yang diperoleh kemudian disimpan dan dikelola menggunakan ChromaDB sebagai basis data vektor. ChromaDB dipilih karena keunggulannya dalam pencarian semantik dengan kecepatan tinggi, penyimpanan terstruktur, serta integrasi yang baik dengan Retrieval-Augmented Generation (RAG) [21]. Melalui penyimpanan berbasis vektor ini, sistem dapat melakukan pencarian semantik secara efisien untuk menemukan potongan-potongan teks yang paling relevan terhadap pertanyaan pengguna.

2.5 Pengembangan Sistem Chatbot

Proses pengembangan dimulai dengan perancangan alur kerja chatbot menggunakan LangGraph. LangGraph dipilih karena kemampuannya mengatur aliran proses dalam bentuk Directed Acyclic Graph (DAG) [22], sehingga setiap tahap dimulai dari menerima pertanyaan, pencarian data, hingga pembuatan jawaban dapat dikelola secara modular dan terstruktur.



Gambar 2.2 Alur Node LangGraph

Gambar 2.2 memperlihatkan alur kerja chatbot dengan LangGraph. Pertanyaan pengguna diklasifikasikan (classify question) untuk menentukan apakah dapat dijawab langsung oleh model (generate answer general) atau memerlukan pencarian dokumen. Jika memerlukan pencarian, pertanyaan diringkas (condense chat), kemudian mengambil dokumen yang relevan dari vector store (retrieve documents), kemudian pada tahap pembuatan jawaban akhir (generate answer), sistem memanfaatkan model LLaMA 3.1 yang diakses melalui infrastruktur Groq API. Integrasi Groq API dipilih secara spesifik agar sistem mampu menyajikan jawaban yang kontekstual dan akurat dengan latensi yang relatif rendah kepada pengguna.

Seluruh proses pengolahan pertanyaan dan jawaban pada alur tersebut dijalankan melalui API yang berfungsi sebagai penghubung antara modul backend chatbot dan antarmuka web. API ini dirancang untuk menerima input berupa pertanyaan dari pengguna, memprosesnya melalui pipeline RAG, serta mengirimkan respons dalam bentuk teks jawaban. Selanjutnya, API tersebut diintegrasikan dengan frontend website sehingga pengguna dapat berinteraksi langsung dengan chatbot melalui tampilan web yang responsif.

2.6 Evaluasi Sistem

Evaluasi sistem dilakukan untuk menilai kualitas jawaban serta kinerja chatbot. Pada aspek kualitas jawaban, digunakan RAGAS [23], sebuah library open-source yang dirancang untuk mengevaluasi sistem berbasis Retrieval-Augmented Generation (RAG). Penelitian ini memfokuskan evaluasi pada tiga metrik utama, yaitu Faithfulness (ketepatan jawaban), Context Precision (relevansi dokumen), Context Recall (pencarian informasi). Pemilihan ketiga metrik ini didasarkan pada karakteristik layanan informasi akademik yang mengutamakan akurasi fakta. Adapun metrik Answer Relevance tidak disertakan dalam evaluasi ini dikarenakan adanya keterbatasan pada rate limit pada model penilaian (judge model) saat menggunakan Groq API.

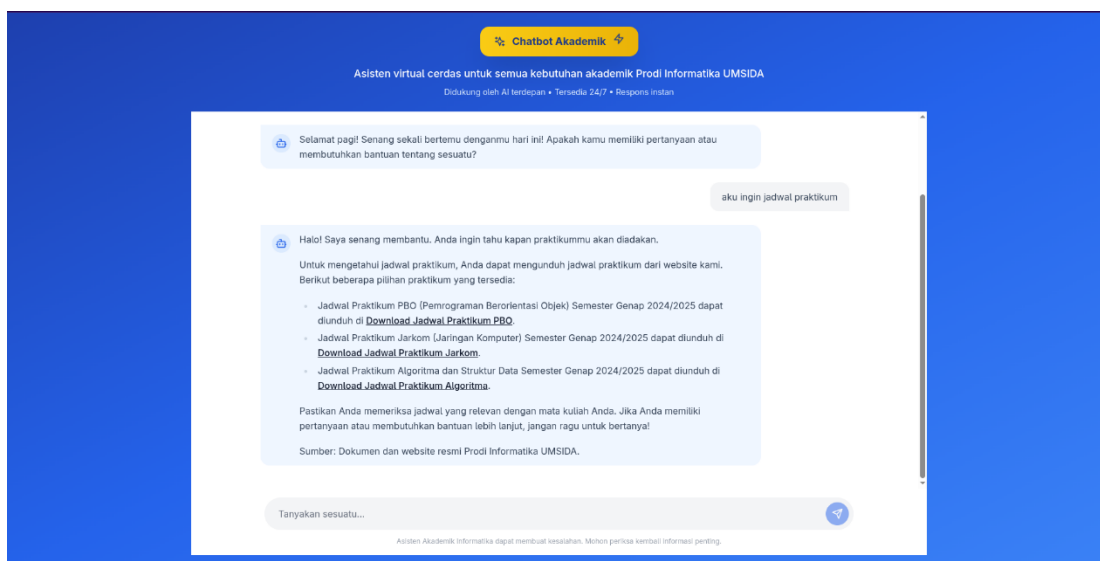
Sementara itu, evaluasi kinerja sistem dilakukan dengan mengukur rata-rata latensi respon dan stabilitas sistem. Latensi dihitung dari waktu yang dibutuhkan sejak pertanyaan dikirim hingga jawaban diterima oleh pengguna. Sedangkan stabilitas diuji dengan menjalankan simulasi beban menggunakan kode sederhana untuk mengetahui apakah chatbot tetap stabil saat digunakan oleh banyak pengguna secara bersamaan.

III. HASIL DAN PEMBAHASAN

Pada bagian ini menjelaskan tentang hasil penelitian dan pembahasan mengenai chatbot informasi akademik berbasis *Retrieval-Augmented Generation* (RAG) yang dikembangkan. Pembahasan difokuskan pada hasil implementasi sistem, pengujian kualitas jawaban chatbot, serta evaluasi kinerja menggunakan metrik yang relevan. Selain itu, bab ini juga membahas hasil interpretasi hasil evaluasi untuk menilai efektivitas sistem dalam menyediakan informasi akademik yang akurat dan kontekstual bagi mahasiswa Program Studi Informatika Universitas Muhammadiyah Sidoarjo.

3.1 Hasil Implementasi Sistem

Hasil implementasi penelitian ini berupa sebuah chatbot informasi akademik berbasis Retrieval-Augmented Generation (RAG) yang telah dikembangkan. Chatbot ini dirancang untuk membantu pengguna dalam memperoleh informasi akademik secara cepat dan kontekstual. Antarmuka chatbot disajikan dalam bentuk komponen interaktif pada halaman website yang responsif, dimana pengguna dapat memasukkan pertanyaan pada kolom chat dan menerima balasan secara real-time. Implementasi antarmuka chatbot dapat dilihat pada Gambar 3.1.



Gambar 3.1 Implementasi Antarmuka Chatbot pada Website

Berdasarkan skenario pengujian, sistem berhasil menjalankan alur LangGraph dengan baik. Ketika pertanyaan diklasifikasikan sebagai rag_chat, sistem melakukan pencarian ke basis data vektor ChromaDB. Sistem tidak hanya mengambil teks, tetapi juga mendukung metadata pendukung seperti tautan (direct_link) yang tersimpan pada format JSON. Hal ini memastikan jawaban yang diberikan tidak hanya deskriptif tetapi juga solutif. Contoh respon chatbot terhadap pertanyaan akademik disajikan pada Tabel 3.1.

Tabel 3.1 Contoh Hasil Respon Chatbot

Pertanyaan	Jawaban
Bagaimana prosedur pengajuan surat keterangan aktif kuliah?	Halo! Untuk prosedur pengajuan surat keterangan aktif kuliah, Anda dapat mengajukan permohonan melalui Layanan Administrasi Surat Online Informatika. Berikut adalah langkah-langkahnya: Buka formulir Formulir Surat Online. Pilih jenis surat yang Anda butuhkan, yaitu Surat Aktif Kuliah. Isi formulir dengan data yang diperlukan, seperti nama, NIM, dan alasan pengajuan. Unggah dokumen-dokumen yang diperlukan, jika ada. Klik tombol "Kirim" untuk mengajukan permohonan. Setelah Anda mengajukan permohonan, tim administrasi akan memproses dan mengirimkan surat keterangan aktif kuliah Anda melalui email atau melalui sistem online. Jika Anda memerlukan bantuan lebih lanjut atau memiliki pertanyaan lain, jangan ragu untuk bertanya!
Kapan jadwal praktikum semester genap untuk mahasiswa Informatika?	Halo! Saya senang membantu menjawab pertanyaan Anda tentang jadwal praktikum semester genap untuk mahasiswa Informatika. Untuk mengetahui jadwal praktikum semester genap, Anda dapat mengunduh jadwal praktikum yang relevan melalui link berikut: Jadwal Praktikum Jarkom: Download Jadwal Praktikum Jaringan Komputer Jadwal Praktikum Algoritma: Download Jadwal Praktikum Algoritma dan Struktur Data Jadwal Praktikum Basis Data: Download Jadwal Praktikum Basis Data

	Silakan mengunduh jadwal yang relevan untuk mengetahui informasi lebih lanjut tentang <u>jadwal praktikum semester genap</u> .
Apa visi dan misi Program Studi Informatika?	Halo! Saya senang membantu menjawab pertanyaan Anda tentang visi dan misi Program Studi Informatika. Visi Program Studi Informatika UMSIDA adalah menghasilkan lulusan yang profesional, unggul, inovatif, dan kompetitif dalam rekayasa perangkat lunak dan sistem cerdas yang adaptif terhadap perkembangan IPTEKS berdasarkan nilai-nilai Islam untuk kesejahteraan masyarakat tingkat ASEAN pada tahun 2038. Misi Program Studi Informatika UMSIDA terdiri dari 5 poin, yaitu: • Pendidikan unggul & inovatif berbasis nilai Islam. • Penelitian IPTEKS untuk kesejahteraan. • Pengabdian masyarakat bereputasi internasional. • Kerjasama berkelanjutan. • Tata kelola profesional & pembinaan kemahasiswaan. Semoga informasi ini membantu Anda memahami visi dan misi Program Studi Informatika UMSIDA!

3.2 Evaluasi Kualitas Jawaban

Evaluasi kualitas jawaban dilakukan menggunakan kerangka kerja RAGAS untuk mengukur seberapa akurat dan relevan jawaban yang dihasilkan oleh model LLaMA 3.1. Pengujian dilakukan terhadap dataset uji yang terdiri dari pertanyaan seputar akademik, administrasi, dan kemahasiswaan. Evaluasi difokuskan pada nilai *Faithfulness*, *Context Precision*, dan *Context Recall*.

Tabel 3.2 Hasil Evaluasi RAGAS

Metrik Evaluasi	Nilai
Faithfulness	0.77
Context Precision	0.86
Context Recall	0.75

Berdasarkan Tabel 3.2, sistem mendapat performa tertinggi pada aspek Context Precision dengan skor 0.86. Nilai ini mengindikasikan bahwa retriever sistem sangat efektif dalam memilah dokumen, dimana sebagian besar potongan teks (chunks) yang diambil memiliki relevansi tinggi terhadap pertanyaan pengguna. Sementara itu, skor Faithfulness sebesar 0.77 menunjukkan tingkat konsistensi yang cukup baik antara jawaban model dengan sumber dokumen, meskipun masih terdapat variasi jawaban dari model LLaMA 3.1. Adapun skor Context Recall sebesar 0.75 menandakan bahwa sistem mampu menyaring informasi yang diperlukan, angka ini dinilai memadai untuk tahap implementasi awal sebuah sistem layanan informasi.

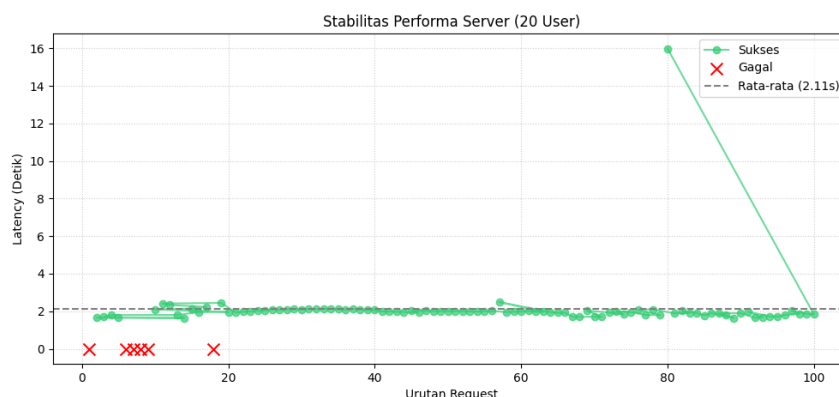
3.3 Evaluasi Kinerja Sistem

Evaluasi kinerja dilakukan untuk mengukur ketahanan dan kecepatan respon dari sistem. Pengujian dilakukan dengan mensimulasikan beban 20 pengguna (concurrent users) secara bersamaan dengan total 100 permintaan (request). Statistik performa sistem dapat dilihat pada Tabel 3.3.

Tabel 3.3 Statistik Performa Sistem

Parameter	Hasil Pengujian
Total Request	100
Sukses Rate	94% (94 Sukses, 6 Gagal)
Throughput	4.14 request/detik
Rata – Rata Latensi	2.11 detik
P95 (95% User)	2.27 detik
Max Latensi	15.99 detik

Hasil pengujian menunjukkan tingkat keberhasilan (success rate) sebesar 94%. Rata-rata latensi respon tercatat sebesar 2.11 detik. Sedangkan pada nilai P95 (Persentil ke-95) berada pada angka 2.27 detik, hal ini menunjukkan bahwa 95% dari permintaan pengguna dilayani dalam waktu kurang dari 2.3 detik. Untuk menganalisis stabilitas latensi, dilakukan pemetaan waktu respon terhadap urutan permintaan yang disajikan pada Gambar 3.2.



Gambar 3.2 Grafik Stabilitas Performa Sistem

Berdasarkan grafik stabilitas pada Gambar 3.2, mayoritas permintaan (titik hijau) berada pada garis datar di kisaran waktu 2 detik. Sedangkan terdapat kegagalan permintaan (tanda silang merah) terjadi pada fase awal pengujian (urutan 1-19), yang mengindikasikan adanya kendala saat sistem membutuhkan waktu inisialisasi untuk memuat model saat pertama kali menerima lonjakan trafik. Selain itu, terdeteksi lonjakan latensi pada permintaan ke-80 dimana latensi melonjak hingga 15.99 detik, kemungkinan hal ini disebabkan oleh mekanisme antrian pada sisi penyedia layanan Groq API. Meskipun terdapat kendala tersebut, sistem mampu memulihkan kinerja dengan cepat kembali ke latensi normal rata-rata 2 detik.

3.4 Pembahasan

Berdasarkan hasil evaluasi yang telah dilakukan, pengembangan chatbot menggunakan metode *Retrieval-Augmented Generation* (RAG) dengan model LLaMA 3.1 melalui Groq API menunjukkan kinerja yang optimal untuk layanan informasi akademik.

1. Implementasi Sistem Chatbot

Penggunaan arsitektur LangGraph memungkinkan alur pemrosesan pertanyaan berjalan secara terstruktur, mulai dari klasifikasi pertanyaan, proses retrieval dokumen, hingga generasi jawaban akhir menggunakan model LLaMA 3.1. Keberhasilan implementasi ini terlihat dari kemampuan sistem dalam mengklasifikasikan pertanyaan yang membutuhkan pencarian dokumen (rag_chat) dan pertanyaan umum yang dapat dijawab langsung oleh model. Selain itu, integrasi basis data vektor ChromaDB memungkinkan sistem mengambil potongan teks yang relevan beserta metadata pendukung, seperti tautan langsung ke sumber informasi. Hal ini menjadikan chatbot tidak hanya bersifat informatif, tetapi juga solutif karena pengguna dapat langsung mengakses sumber resmi yang dirujuk.

2. Kualitas Jawaban

Skor Context Precision sebesar 0.86 mengindikasikan bahwa sistem retrieval mampu memilih dokumen yang relevan dengan tingkat akurasi yang tinggi. Hal ini menunjukkan bahwa proses preprocessing data ke dalam format JSON terstruktur serta penggunaan model embedding nomic-embed-text berkontribusi signifikan terhadap ketepatan konteks yang diberikan kepada model. Skor Faithfulness sebesar 0.77 menunjukkan bahwa jawaban yang dihasilkan oleh model LLaMA 3.1 sebagian besar konsisten dengan dokumen sumber yang digunakan dalam proses retrieval. Meskipun masih terdapat variasi redaksi akibat sifat generatif LLM, jawaban yang dihasilkan tetap berada dalam batas konteks informasi akademik yang benar. Sementara itu, skor Context Recall sebesar 0.75 menandakan bahwa sistem telah mampu menangkap sebagian besar informasi yang dibutuhkan pengguna, meskipun masih terdapat ruang untuk peningkatan cakupan dokumen yang diambil.

3. Stabilitas dan Kinerja Sistem

Hasil pengujian beban menunjukkan bahwa sistem memiliki tingkat keberhasilan sebesar 94% dengan rata-rata latensi respon 2.11 detik. Nilai P95 sebesar 2.27 detik mengindikasikan bahwa sebagian besar permintaan pengguna dapat dilayani dalam waktu yang relatif singkat dan konsisten. Hasil ini menunjukkan bahwa penggunaan Groq API sebagai infrastruktur inferensi LLM mampu menekan latensi yang sering menjadi kendala pada sistem chatbot berbasis RAG. Lonjakan latensi maksimum hingga 15.99 detik serta kegagalan permintaan yang terjadi pada fase awal pengujian menunjukkan adanya keterbatasan pada proses inisialisasi

sistem dan potensi antrian pada layanan Groq API. Namun demikian, sistem mampu kembali ke kondisi stabil dengan cepat, sehingga secara keseluruhan chatbot dinilai cukup andal untuk digunakan sebagai layanan informasi akademik berbasis web.

Secara keseluruhan, integrasi langsung ke dalam website program studi menggunakan arsitektur *LangGraph* terbukti lebih unggul dalam aspek aksesibilitas dibandingkan solusi berbasis Telegram atau aplikasi pihak ketiga yang dibahas pada penelitian terdahulu.

VII. KESIMPULAN

Penelitian ini menyimpulkan bahwa implementasi chatbot akademik berbasis Retrieval-Augmented Generation (RAG) menggunakan model LLaMA 3.1 melalui Groq API berhasil meningkatkan efisiensi akses informasi mahasiswa dibandingkan situs web statis. Berdasarkan evaluasi, sistem terbukti mampu menyajikan jawaban yang akurat secara kontekstual dengan skor Context Precision mencapai 0.86 dan Faithfulness 0.77. Selain itu, rata-rata latensi respon sebesar 2.11 detik dinilai sangat memadai untuk kebutuhan pencarian informasi umum, memastikan pengguna mendapatkan jawaban tanpa menunggu waktu yang lama. Integrasi arsitektur LangGraph dan basis data vektor ChromaDB pada sistem ini menghasilkan respon yang solutif melalui penyertaan tautan sumber data yang relevan. Berdasarkan implementasi tersebut, penelitian selanjutnya disarankan untuk memperluas cakupan sistem ke tingkat fakultas maupun universitas pada Universitas Muhammadiyah Sidoarjo guna menyediakan layanan informasi yang lebih komprehensif. Pengembangan ini dapat dilakukan dengan mengintegrasikan basis data akademik dari berbagai program studi melalui teknik *automated web scraping* yang lebih luas.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih sebesar-besarnya kepada **Laboratorium Informatika Universitas Muhammadiyah Sidoarjo** yang telah menyediakan seluruh fasilitas yang diperlukan, serta pihak-pihak yang telah membantu dalam penelitian ini.

REFERENSI

- [1] A. A. Zulfa, T. Ibrahim, and O. Arifudin, "Peran Sistem Informasi Akademik Berbasis Web Dalam Upaya Meningkatkan Efektivitas Dan Efisiensi Pengelolaan Akademik Di Perguruan Tinggi," *J. Tahsinia*, vol. 6, no. 1, pp. 115–134, 2025.
- [2] S. Naikar and C. Indraji, "AI-Driven Innovation in Information Retrieval: Transforming Library Operations and User Engagement," *Int. Conf. Glob. Perspect. Open Sci. Bridg. Res. Access Innov.*, no. May, pp. 1–7, 2025, [Online]. Available: http://papers.ssrn.com/abstract_id=5261229
- [3] L. Costaner, P. Studi, T. Informatika, F. Ilmu, K. Universitas, and L. Kuning, "Aplikasi Chatbot untuk Layanan Informasi dan Akademik Kampus Berbasis Artificial Intelligence Markup Language (AIML)," pp. 291–300, 2020.
- [4] F. Purwani *et al.*, "IMPLEMENTASI APLIKASI CHATBOT SEBAGAI MEDIA INFORMASI PADA PENGEMBANGAN SISTEM AKADEMIK UNIVERSITAS ISLAM NEGERI (UIN) RADEN FATAH PALEMBANG MENGGUNAKAN ARTIFICIAL INTELLIGENCE MARKUP LANGUAGE," vol. 1, no. 4, pp. 1–7, 2024.
- [5] M. F. Ajiz, M. Faza, S. Ramadan, H. D. Mutia, P. D. Yanuari, and I. Pendahuluan, "Pengembangan Aplikasi Chatbot Informasi Akademik Berbasis Web Menggunakan Metode Artificial Intelligence Markup Language (AIML)," vol. 15, no. 2, 2023.
- [6] N. Cahyono, "Perancangan Sistem Informasi Penyewaan Alat Outdoor Berbasis Web menggunakan Metode Waterfall pada InOutdoors Rental Sidoarjo," no. 1, pp. 1–23, 2024.
- [7] A. Alvin, R. Robet, and A. Tarigan, "Implementasi chatbot Otomatis Akademik Berbasis Web Menggunakan LLM dan Rule-Based System Studi Kasus : STMIK Time Implementation of Web-Based Academic Automated chatbot Using LLM and Rule-Based System Case Study : STMIK Time," no. 3, pp. 651–665, 2025, doi: 10.26798/jiko.v9i3.2209.
- [8] M. I. T. Khaqiqi and N. H. Harani, "Peningkatan Kinerja Chatbot NLP Asisten : Tinjauan Literatur tentang Metode dan Akurasi dalam Aplikasi Berbasis Percakapan," vol. 05, no. 01, pp. 50–59, 2024.
- [9] D. Apriliani, S. F. Handayani, I. T. Saputra, T. Informatika, and P. H. Bersama, "Implementasi Natural Language Processing (NLP) Dalam Pengembangan Aplikasi Chatbot Pada SMK YPE Nusantara Slawi," vol. 22, no. 4, pp. 1037–1047, 2023.
- [10] S. Abedu and A. Abdellatif, "LLM-Based Chatbots for Mining Software Repositories : Challenges and Opportunities," no. June 2024, 2025, doi: 10.1145/3661167.3661218.
- [11] C. Y. Kim, C. P. Lee, and B. Mutlu, "Understanding Large-Language Model (LLM)-powered Human-Robot Interaction," *ACM/IEEE Int. Conf. Human-Robot Interact.*, no. Llm, pp. 371–380, 2024, doi: 10.1145/3610977.3634966.
- [12] S. Vidivelli, M. Ramachandran, and A. Dharunbalaji, "Efficiency-Driven Custom Chatbot Development: Unleashing LangChain, RAG, and Performance-Optimized LLM Fusion," *Comput. Mater. Contin.*, vol. 80, no. 2, pp. 2423–2442, 2024, doi: 10.32604/cmc.2024.054360.
- [13] Milasanti D, "SISTEM CHATBOT BERBASIS LARGE LANGUAGE MODEL (LLM) DAN RETRIEVAL AUGMENTED GENERATION (RAG) PADA ARTIKEL ILMIAH GARUDA KEMDIKBUD," 2024.
- [14] Nu'aeni H, "PENGEMBANGAN CHATBOT UNTUK SISTEM INFORMASI PERPUSTAKAAN DIGITAL (DIGILIB) MENGGUNAKAN RETRIEVAL-AUGMENTED GENERATION (RAG)," 2025.
- [15] L. R. Hidayat, G. Pasek, S. Wijaya, and R. Dwi Yansaputra, "Optimalisasi Layanan Sistem Informasi Mahasiswa dengan Integrasi Telegram : Chatbot Retrieval-Augmented-Generation berbasis Large Language Model (Optimization of Student Information System Services with Telegram Integration : Chatbot Retrieval-Augmented Generation based on Large Language Model)," 2021. [Online]. Available: <http://jtika.if.unram.ac.id/index.php/JTIKA/>
- [16] M. Grusky, "Rogue Scores," vol. 1, pp. 1914–1934, 2023.
- [17] U. States, "Towards Seamless User Query to REST API Conversion," no. October 2024, 2025, doi: 10.1145/3627673.3680275.
- [18] I. Mohr, D. J. Williams, B. Wang, and H. Xiao, "L ATE C HUNKING : C ONTEXTUAL C HUNK E MBED -," pp. 1–14, 2024.
- [19] M. A. Rosid, D. O. Siahaan, and A. Saikhu, "Sarcasm Detection in Indonesian-English Code-Mixed Text Using Multihead Attention-Based Convolutional and Bi-Directional GRU," no. June, pp. 137063–137079, 2024.
- [20] Z. Nussbaum, J. X. Morris, and B. Duderstadt, "Nomic Embed : Training a Reproducible Long Context Text Embedder," vol. 002, pp. 1–17, 2025.

- [21] T. Chatbot, "Implementation of Retrieval-Augmented Generation (RAG) and Large Language Models (LLM) for a Document and Tabular-Based Chatbot System," pp. 19–23, 2025.
- [22] J. Wang and Z. Duan, "Agent AI with LangGraph: A Modular Framework for Enhancing Machine Translation Using Large Language Models," 2024, [Online]. Available: <http://arxiv.org/abs/2412.03801>
- [23] S. Es, J. James, L. Espinosa-anke, and S. Schockaert, "RAGA S : Automated Evaluation of Retrieval Augmented Generation," pp. 150–158, 2024.