BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1264-1272
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1088

Implementasi Local-First RAG dengan Hybrid Retrieval IndoBERT dan BM25 untuk Pendukung Keputusan Akademik

Romi Wahyudi Hasibuan*, Ahmad Rio Adriansyah, Henry Saptono

Program Studi Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri, Jakarta Selatan, Indonesia
Email: 1,*romi22021ti@student.nurulfikri.ac.id, 2arasy@nurulfikri.ac.id, 3henry@nurulfikri.ac.id
Email Penulis Korespondensi: romi22021ti@student.nurulfikri.ac.id

Abstrak−Kebutuhan akses informasi akademik yang cepat dan akurat bagi pemangku kepentingan institusi pendidikan, seperti manajemen kampus dan *academic advisor*, masih mengandalkan staf administrasi secara manual sehingga menimbulkan inefisiensi operasional. Penelitian ini membangun sistem *Retrieval-Augmented Generation* (RAG) berbasis *open-source*yang dijalankan secara lokal (*local-hosted*) sebagai solusi otomatisasi layanan informasi akademik sekaligus meringankan beban administratif. Sistem mengintegrasikan LLM Mistral-7B-Instruct dengan pendekatan pencarian hibrida pada ekosistem Elasticsearch, memadukan *dense retrieval* berbasis IndoBERT untuk dokumen naratif pedoman akademik dan *sparse retrieval* berbasis BM25 untuk data terstruktur mahasiswa. Evaluasi dilakukan menggunakan metrik ROUGE-1, ROUGE-2, dan ROUGE-L terhadap 60 data uji yang dihasilkan oleh Claude Sonnet 4.6. Sistem berhasil menjawab 58 dari 60 query, dengan nilai ROUGE-L *f1-score* sebesar 0,29. Pola asimetris ditemukan di mana nilai *recall* secara konsisten lebih tinggi dari *precision* pada seluruh metrik, yang diindikasikan sebagai dampak dari kesenjangan kapasitas generasi bahasa antara Mistral 7B dan model referensi. Rata-rata panjang *prompt input* berkisar 1.270—1.380 token berkontribusi pada latensi rata-rata 30 detik per query, yang menjadi tantangan utama sistem saat ini. Penelitian ini memberikan kontribusi dan dapat menjadi *baseline* pengembangan sistem RAG berbasis *open-source* untuk domain berbahasa Indonesia, khususnya dalam lingkup administrasi akademik perguruan tinggi.

Kata Kunci: LLM Lokal; Open-Source; Otomatisasi; Pencarian Hibrida; Retrieval-Augmented Generation

Abstract−The need for rapid and accurate access to academic information for educational institution stakeholders, such as campus management and academic advisors, still relies on manual administrative processes, thereby leading to operational inefficiencies. This study develops a locally hosted, open-source Retrieval-Augmented Generation (RAG) system as an automated solution for academic information services while alleviating administrative burdens. The system integrates the Mistral-7B-Instruct LLM with a hybrid search approach within the Elasticsearch ecosystem, combining IndoBERT-based dense retrieval for narrative academic guideline documents and BM25-based sparse retrieval for structured student data. Evaluation was conducted using ROUGE-1, ROUGE-2, and ROUGE-L metrics against 60 test data points generated by Claude Sonnet 4.6. The system successfully answered 58 out of 60 queries, achieving a ROUGE-L f1-score of 0.29. An asymmetrical pattern was observed, where recall values were consistently higher than precision across all metrics, which indicates the impact of the language generation capacity gap between Mistral 7B and the reference model. An average input prompt length ranging from 1,270 to 1,380 tokens contributed to an average latency of 30 seconds per query, representing the primary contemporary challenge of the system. This research is expected to serve as a baseline for developing open-source RAG systems within Indonesian language domains, specifically in the context of higher education academic administration.

Keywords: Local LLM; Open-Source; Automation; Hybrid Search; Retrieval-Augmented Generation

1. PENDAHULUAN

Di bidang akademik, pelayanan informasi merupakan aspek esensial bagi pemangku kepentingan atau *stakeholder* (manajemen kampus dan *academic advisor*) untuk menunjang operasional institusi[1]. Dalam praktiknya, *stakeholder* menginginkan layanan informasi yang dinamis dan dapat disesuaikan dengan kebutuhan (*customised*) [2]. Namun, layanan konvensional yang masih mengandalkan tenaga manusia (staf) terbukti memiliki berbagai kekurangan, terutama dalam aspek kecepatan, responsivitas, dan prosedur birokrasi dalam pendistribusian informasi [1].

Di sisi lain, pendekatan teknologi seperti *learning analytics dashboard* (LAD) dinilai belum memadai karena informasi yang disajikan cenderung statis dan terlalu umum [2]. Walaupun intervensi manusia telah dikurangi, pendekatan ini masih menyulitkan *stakeholder* untuk memetakan data tersebut menjadi wawasan yang dapat ditindaklanjuti (*actionable*) pada konteks spesifik mereka [2]. Keterbatasan ini sejalan dengan temuan Yan et al. [3] yang menyoroti bahwa visualisasi data pada LAD belum cukup memberikan pemahaman yang utuh tanpa adanya narasi kontekstual. Ketiadaan penjelasan naratif ini sering kali memunculkan kesenjangan interpretasi bagi *stakeholder* yang memiliki keterbatasan dalam literasi visual [3]. Akibatnya, tugas analisis data spesifik tetap harus dibebankan kembali kepada staf. Namun, penelitian Grey et al. [2] menyebutkan secara eksplisit bahwa staf sering kali kewalahan oleh tumpukan data (*overwhelmed by data*) dan merasa bahwa analisis data mendalam bukanlah tugas utama mereka. Lebih lanjut, karakteristik luaran informasi pada berbagai sistem pendukung keputusan saat ini juga masih menjadi kendala, dimana informasi masih sering kali bergantung pada format dokumen statis dan melibatkan proses administrasi yang dilakukan secara luring (*offline*) [4]. Hal ini menyoroti urgensi akan sebuah sistem otomatisasi *on-demand* yang dapat memberdayakan *stakeholder* untuk mengakses informasi secara mandiri sekaligus mereduksi beban kerja repetitif staf administratif.

Dalam konteks otomatisasi, teknologi *Artificial Intelligence* (AI), khususnya *Large Language Model* (LLM), menawarkan solusi yang menjanjikan [5]. Terlebih lagi, Implementasi LLM yang dikombinasikan dengan teknik *retrieval* terbukti mampu mengoptimalkan proses otomatisasi pengambilan keputusan dari segi validitas maupun efisiensi [6],[7]. Chafiq et al [6] dalam penelitiannya mengubah sistem yang ada menjadi sepenuhnya terotomatisasi. Mulai dari ekstraksi informasi (*retrieval*) menggunakan sistem NLP dan *Transformer*, hingga peringkasan (*summarization*) informasi

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1264-1272
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1088

menggunakan LLM (*generation*). Hal ini memungkinkan evaluator untuk fokus maksimal terhadap penilaian substansi informasi yang disajikan. Pada tingkat yang lebih tinggi, Marques et al [7] menerapkan sistem berbasis LLM pada dua program pendanaan pemerintah Uni Eropa, dimana otomatisasi proses evaluasi dokumen terbukti meningkatkan produktivitas reviewer hingga 20% dan memangkas total waktu evaluasi lebih dari dua bulan.

Dalam ranah pendidikan tinggi, implementasi *chatbot* berbasis LLM telah banyak dieksplorasi, dimana mayoritas pendekatannya masih sangat bergantung pada arsitektur *closed-source* [5], [8], [9]. Penelitian Maryamah et al. [10] mengembangkan *chatbot* dengan menggunakan infrastruktur komersial GPT-3.5 Turbo dan model *embedding* OpenAI ADA. Meskipun terbukti unggul secara luaran, penelitian ini secara eksplisit mengonfirmasi kendala tingginya biaya operasional per token dan kekhawatiran terhadap transmisi data sensitif ke vendor eksternal [10]. Keterbatasan ekonomi dari arsitektur *closed-source* ini dipertegas oleh penelitian pengembangan sistem Unipa-GPT [11]. Tingginya skema tarif API *token-based* dari vendor memaksa peneliti untuk memangkas ukuran dataset secara drastis dan menghambat penerapan optimisasi, sehingga penelitian tersebut sangat merekomendasikan transisi ke LLM *open-source* di masa mendatang. Adopsi model komersial serupa (GPT-3.5 Turbo) juga dilakukan pada pengembangan OwlMentor [12]. Walaupun evaluasi menggunakan *Technology Acceptance Model* (TAM) menunjukkan penerimaan pengguna yang positif, penelitian ini mewarisi ketergantungan mutlak pada ekosistem vendor. Dikotomi arsitektur *close*-source dan *open*-source dikaji secara komprehensif oleh Alier et al. [13], penelitian ini menyimpulkan bahwa infrastruktur *closed-source* memang memberikan performa penalaran yang impresif secara instan, namun terus-menerus memunculkan kekhawatiran terkait pembengkakan anggaran (*cost*). Sebaliknya, infrastruktur sumber terbuka (*open-source*) menawarkan keunggulan finansial jangka panjang dan kebebasan kustomisasi, meskipun adopsinya sering kali terhambat oleh tingginya kompleksitas teknis [13].

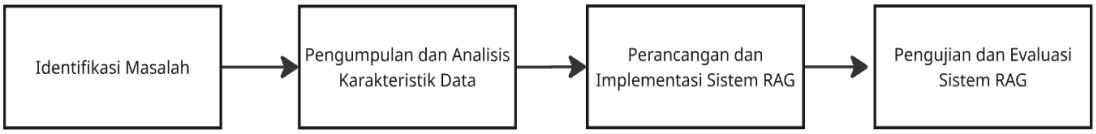
Studi RAG terdahulu [10], [11], [12] yang mayoritas masih mengandalkan perangkat lunak tertutup (*closed-source*) dari vendor pihak ketiga (seperti OpenAI API) memunculkan sejumlah kelemahan kritis. Hal ini menghambat keberlanjutan (*sustainability*) jangka panjang dari segi ekonomi, menimbulkan risiko kebocoran privasi data sensitif, potensi plagiarisme tak terdeteksi, serta isu kepatuhan hukum terkait kedaulatan data yang sulit dikendalikan oleh institusi [14]. Untuk mengatasi celah tersebut, penelitian ini berkontribusi dalam membangun dan menguji sistem RAG berbahasa Indonesia dengan memanfaatkan infrastruktur *local* dan *open-source.* Implementasi mandiri ini menjadi solusi nyata untuk mewujudkan keberlanjutan (*sustainability*) sistem layanan akademik sekaligus menjamin kedaulatan data (*data sovereignty*) institusi secara penuh.

Setelah mempertimbangkan aspek keberlanjutan dan kedaulatan data, pemilihan model LLM menjadi sangat krusial untuk tetap mempertimbangkan kualitas luaran LLM dan juga penyesuaian dengan *resource* pengembangan yang dimiliki. Penelitian Kozhipuram et al [15] memberikan informasi bahwa LLM dengan parameter kecil yang dikombinasikan dengan RAG-augmented mampu bersaing dengan LLM yang mempunyai parameter jauh lebih besar. Lebih lanjut disebutkan bahwa mistral 7B yang menjadi salah satu sampel uji terbukti mengalahkan model model perbandingan lain seperti TinyLlama-1.1B, Llama-1-13B, bahkan Llama-3.1-8B. Mistral 7B konsisten unggul dalam setiap pengujian, baik dari aspek leksikal dan sematik. Lebih lanjut, studi Dayarathne et al. [16] yang mengevaluasi kinerja berbagai model *open-source* pada tugas *Retrieval-Augmented Generation* (RAG) mendemonstrasikan bahwa arsitektur Mistral 7B, secara spesifik Mistral-7B-Instruct, secara konsisten mengungguli kompetitor di kelas parameter yang sama, seperti LLaMA-2-7B dan Falcon-7B, dalam skenario sistem tanya-jawab (*question answering*). Secara arsitektural, keunggalan ini bersumber dari inovasi teknis Mistral 7B itu sendiri, Jiang et al [17] mendemonstrasikan bahwam model ini mampu melampaui Llama 2 13B di seluruh benchmark penalaran, matematika, dan pemahaman bacaan, meskipun hanya menggunakan separuh jumlah parameternya, berkat mekanisme *Grouped Query Attention* (GQA) dan *Sliding Window Attention* (SWA) yang mengoptimalkan efisiensi komputasi.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Penelitian yang dikembangkan dalam artikel ini dilaksanakan melalui tahapan sistematis. Gambar 1 menyajikan diagram alur yang merinci langkah-langkah yang dilakukan.



Gambar 1. Alur Penelitian

Langkah awal yang dilakukan adalah mengidentifikasi permasalahan sebagai dasar pengembangan studi. Kemudian pada tahap kedua analisis karakteristik data dilakukan melalui kajian literatur review untuk melihat pemrosesan yang sesuai. Tahap ketiga merancangan arsitektur sistem yang sesuai. Tahap keempat mengimplementasiakan arsitektur menjadi serangkaian sistem RAG yang utuh. Tahap terakhir adalah melakukan pengujian dan evaluasi sistem yang telah dikembangkan.

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1264-1272
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1088

2.2 Pengumpulan dan Analisis Karakteristik Data

Penelitian ini menggunakan basis pengetahuan eksternal yang terdiri dari dua kategori data heterogen. Kategori pertama adalah data pedoman akademik, yakni data yang bersumber dari peraturan universitas, buku panduan akademik, dan kode etik mahasiswa. Data ini berfungsi sebagai kerangka normatif yang menyediakan konteks dan peraturan universitas.

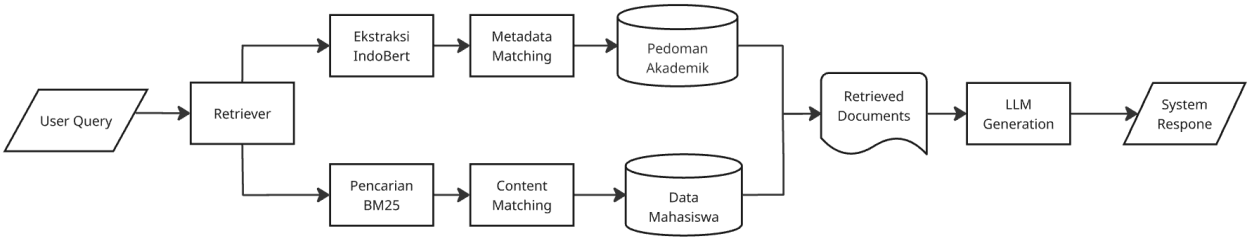
Kategori kedua adalah data akademik mahasiswa, yang berisi rekam jejak dan atribut faktual mahasiswa. Jika data pedoman adalah aturan, maka data akademik mahasiswa berperan sebagai objek evaluasi. Kedua jenis data ini memungkinkan sistem untuk menghasilkan wawasan (*insight*) terkait tingkat kesesuaian (*compliance*) kondisi mahasiswa terhadap standar akademik yang berlaku.

Secara fundamental, kedua jenis data tersebut memiliki karakteristik yang unik sekaligus kontradiktif. Data akademik mahasiswa cenderung bersifat terstruktur dan eksak, seperti Nomor Induk Mahasiswa (NIM), nilai, hingga presensi, yang menuntut presisi leksikal dalam proses pencariannya. Sebaliknya, dokumen pedoman akademik bersifat naratif dan tidak terstruktur, di mana sebuah informasi dapat diekspresikan secara berbeda namun tetap memiliki makna semantik yang sama.

2.3 Perancangan dan Implementasi Sistem RAG

Perancangan arsitektur sistem dikembangkan melalui analisis studi literatur yang dijadikan sebagai landasan empiris dalam menentukan setiap komponen RAG. Diawali dengan tahap penarikan data (*retrieval*), validasi ini merupakan langkah fundamental untuk menjamin akurasi luaran dari LLM [18]. Analisis literatur pada hasil eksperimen Karpukhin et al. [19] memperlihatkan pola yang sangat jelas mengenai kecocokan metode penarikan informasi (*retrieval*) dengan jenis datanya. Temuan tersebut membahas bahwa pada dataset SQuAD, di mana pertanyaan dibuat dengan melihat teks jawaban terlebih dahulu, terdapat banyak kata yang sama persis (*high lexical overlap*) antara pertanyaan dan dokumen. Kondisi ini membuat metode pencarian leksikal (*sparse retrieval*) seperti BM25 bekerja lebih efektif dibandingkan metode padat (*dense retrieval*). Sebaliknya, fenomena berbeda terjadi pada dataset *Natural Questions* (NQ) yang berisi kueri pencarian asli dari pengguna mesin pencari. Berbeda dengan SQuAD, pertanyaan pada NQ murni berasal dari pengguna yang benar-benar mencari informasi tanpa mengetahui jawabannya. Karakteristik ini membuat kueri NQ lebih alami dan variatif secara semantik, sehingga metode *dense retrieval* terbukti jauh lebih unggul.

Temuan empiris tersebut menjadi landasan teoretis utama terhadap perlakuan dataset dalam penelitian ini. Data pedoman akademik diidentifikasi memiliki karakteristik naratif dengan variabilitas semantik yang tinggi, serupa dengan pola dataset NQ, sehingga pendekatan *dense retrieval* dinilai sebagai strategi paling relevan. Di sisi lain, data akademik mahasiswa yang bersifat terstruktur menuntut presisi leksikal mutlak, menjadikan metode *sparse retrieval* sebagai strategi yang paling presisi untuk mengekstraksi data tersebut. Oleh karena itu, arsitektur sistem ini mengadopsi penggabungan kedua metode tersebut menjadi kerangka kerja *hybrid retrieval*.



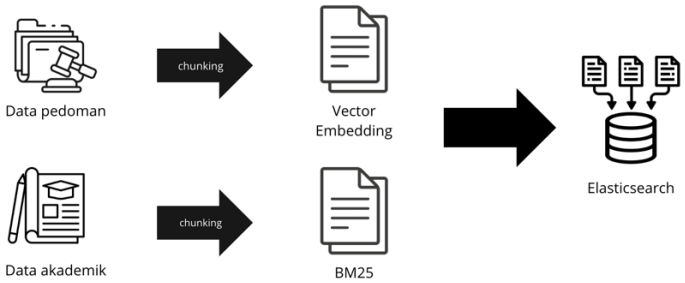
Gambar 2. Arsitektur Sistem RAG

Implementasi arsitektur RAG pada penelitian ini mengintegrasikan model parametrik (LLM) dengan pangkalan data non-parametrik yang dapat diperbarui secara dinamis. Sesuai dengan spesifikasi arsitektur hibrida yang telah dirancang, modul penarikan data (*retrieval*) dibagi menjadi dua jalur pemrosesan. Pada jalur penarikan semantik (*dense retrieval*) untuk data pedoman akademik, IndoBERT bertindak sebagai *encoder* utama. Pemilihan model ini didasarkan pada kompatibilitasnya terhadap prinsip *open-source* dan riwayat prapelatihannya pada korpus berskala besar Indo4B, sehingga sangat relevan untuk merepresentasikan dokumen berbahasa Indonesia. Sementara itu, jalur penarikan leksikal (*sparse retrieval*) untuk mengekstraksi atribut faktual mahasiswa (seperti NIM dan IPK) mengadopsi algoritma BM25. Dengan kerangka probabilistic dan merupakan pengembangan dari pembobotan TF-IDF, metode ini digunakan sebagai standar industri (*de facto method*) dalam sistem temu kembali informasi (*retrieval information system*) [20]

Rancangan susbsistem penarikan informasi (*retrieval*) dikombinasikan dengan model Mistral 7b sesuai hasil pencarian titik temu yang tepat dengan mempertimbangkan kapasitas lingkungan pengembangan (*hardware*). Spesifikasi *hardware* yang dipakai menggunakan Macbook air M1 dengan spesifikasi RAM 16 gb dilengkapi penyimpanan ssd sebesar 1 tb.

Untuk mereduksi kompleksitas infrastruktur, pengintegrasian kedua paradigma pencarian diorkestrasikan dalam satu platform tunggal menggunakan Elasticsearch. Sebagai mesin temu kembali informasi, Elasticsearch tidak hanya mengeksekusi algoritma leksikal BM25 secara efisien, tetapi juga memiliki kapabilitas bawaan untuk penyimpanan dan penarikan vektor (*dense vector search*) [21]. Sentralisasi proses *retrieval* ke dalam satu ekosistem Elasticsearch ini secara signifikan memangkas beban komputasi dan meningkatkan efisiensi operasional system di lingkungan lokal.

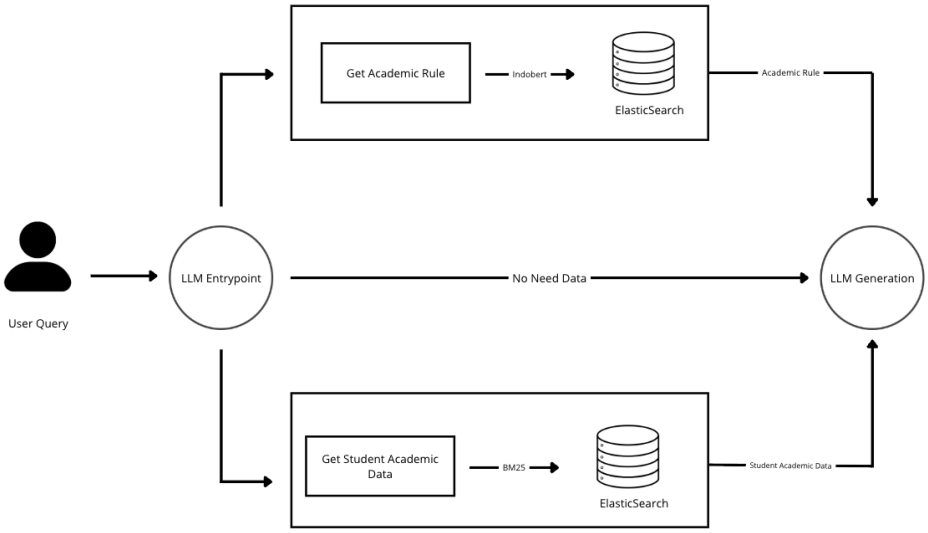
BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1264-1272
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1088



Gambar 3. Pipeline Pencarian Data Dengan Metode Hybrid

Dalam implementasi operasionalnya, fungsi pencarian pada Elasticsearch dienkapsulasi menjadi modul alat (*retrieval tools*) independen. Pada tahap ini, Mistral-7B-Instruct diposisikan sebagai mesin penalaran pusat (*reasoning engine*) yang memiliki otoritas untuk mengevaluasi niat (*intent*) pengguna. Model secara dinamis memformulasikan keputusan, apakah kueri memerlukan rujukan data eksternal, alat pencarian spesifik mana yang harus dipanggil, atau apakah pertanyaan dapat dijawab secara langsung.

Secara arsitektural, orkestrasi alur kerja agen cerdas ini direpresentasikan sebagai graf terarah menggunakan kerangka kerja *LangGraph*. Pendekatan ini memodelkan setiap unit komputasi, seperti fase penalaran (*reasoning*), eksekusi tindakan (*action*), dan sintesis luaran, sebagai simpul (*Node*) [22]. Sementara itu, logika transisi kondisional dihubungkan melalui sisi (*Edge*), misalnya aturan mencari data yang relevan. Konstruksi ini secara fundamental memungkinkan sistem RAG untuk melakukan penalaran bertingkat (*multi-step reasoning*) dan koreksi mandiri (*self-correction*) dalam satu siklus interaksi, sebagaimana divisualisasikan pada Gambar 4.



Gambar 4. Struktrur1 Sistem RAG Dengan LLM Sebagai Entrypoint

Inti dari kapabilitas penalaran otonom agen dalam kerangka kerja *LangGraph* tersebut terletak pada optimasi rekayasa instruksi (*prompt engineering*). Sistem ini mengadopsi struktur *prompt* berlapis yang diawali dengan penetapan persona (*system prompt*) sebagai staf akademik yang beroperasi dengan aturan anti-halusinasi yang ketat. Model diinstruksikan secara absolut untuk membatasi sintesis jawaban hanya berdasarkan konteks dokumen yang ditarik, dan menolak menjawab (*fallback*) apabila informasi faktual tidak ditemukan di dalam pangkalan pengetahuan. Selanjutnya, untuk menutupi keterbatasan bawaan model berparameter 7B dalam mengeksekusi format terstruktur, sistem menginjeksi teknik *Few-Shot Prompting*. Melalui teknik ini, model diberikan beberapa demonstrasi skenario (*input-output*) yang secara paksa mendisiplinkan model untuk selalu menghasilkan luaran pemanggilan alat (*tool calling*) dalam format *JSON routing* yang presisi dan bebas dari galat sintaksis (*syntax error*).

Gambar 5. Sistem Prompt *Chain of Thought* Dalam Sistem RAG

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1264-1272
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1088

Selain standardisasi format luaran, proses kognitif model juga direkayasa menggunakan pendekatan *Chain of Thought* (CoT) yang terintegrasi dengan paradigma *Reasoning and Acting* (ReAct). Gambar 5 memberikan penjelasan bahwa sebelum mengeksekusi alat pencarian atau menyintesis jawaban akhir, model dipaksa untuk mendeklarasikan langkah berpikirnya (*thought process*) beriringan dengan masuknya pertanyaan dari pengguna (*user query*). Model harus menganalisis niat pengguna, menimbang ketersediaan data, serta memutuskan apakah pertanyaan pengguna membutuhkan konteks tambahan. Ketika pertanyaan pengguna membutuhkan data eksternal maka mekanisme *ReAct* akan dijalankan. Agent akan bernalar menentukan data mana yang dibutuhkan *(Reason*) dan memutuskan secara logis (*Action*) alat mana yang harus dipanggil (*tool calling*). Jika pertanyaan pengguna tidak membutuhkan konteks atau data tambahan dari eksternal database, maka sistem akan langsung masuk ke tahap *generation node* untuk menjawab pertanyaan pengguna.

2.4 Pengujian dan Evaluasi

Penilaian sistem RAG menjadi sangat krusial untuk melihat bagaimana performa dari luaran (*output*) yang dihasilkan sistem. Evaluasi luaran sistem RAG dilakukan dengan menggunakan metode *Recall-Oriented Understudy for Gisting Evaluation* (ROUGE). Metode ini membandingkan jawaban yang dihasilkan oleh sistem (*generation*) dengan referensi jawaban yang telah dibuat (*ground truth*)[23]. Pada tahap pengujian, indikator seperti ROUGE-1, ROUGE-2, dan ROUGE-L digunakan untuk melihat diversitas hasil luaran sistem. Diversitas ini ditujukan untuk melihat kemampuan dan kekurangan sistem lebih mendalam.

Pada ketiga indikator, 3 metrik atau ukuran digunakan untuk melihat kinerja sistem. *Precision* (P), mengukur seberapa banyak kata (n-gram) yang dihasilkan sistem benar-benar ada dalam referensi jawaban (*ground truth*). *Recall (*R) mengukur seberapa banyak kata yang ada pada referensi jawaban (*ground truth*) berhasil dikeluarkan oleh sistem. Terakhir, *F1-Score* (F1) mengukur keseimbangan antara keduanya untuk melihat performa sistem secara keseluruhan.

a. ROUGE-1
Digunakan untuk mengukur luaran sistem dengan membandingkan tepat satu token (unigram) dari hasil luaran sistem (*generation*) dengan referensi jawaban yang dibuat (*ground truth*)

$$P_{\text{ROUGE-1}} = \frac{\text{Count}_{\text{match}}(\text{unigram})}{\text{Count}_{\text{total}}(\text{unigram}_{\text{result}})} \quad (1)$$

$$R_{\text{ROUGE-1}} = \frac{\text{Count}_{\text{match}}(\text{unigram})}{\text{Count}_{\text{total}}(\text{unigram}_{\text{reference}})} \quad (2)$$

$$F1_{\text{ROUGE-1}} = 2 \times \frac{P_{\text{ROUGE-1}} \times R_{\text{ROUGE-1}}}{P_{\text{ROUGE-1}} + R_{\text{ROUGE-1}}} \quad (3)$$

b. ROUGE-2
Digunakan untuk mengukur luaran sistem dengan membandingkan tepat urutan dua kata antara luaran sistem (*generation*) dengan referensi jawaban (*ground truth*).

$$P_{\text{ROUGE-2}} = \frac{\text{Count}_{\text{match}}(\text{bigram})}{\text{Count}_{\text{total}}(\text{bigram}_{\text{hasil}})} \quad (4)$$

$$R_{\text{ROUGE-2}} = \frac{\text{Count}_{\text{match}}(\text{bigram})}{\text{Count}_{\text{total}}(\text{bigram}_{\text{referensi}})} \quad (5)$$

$$F1_{\text{ROUGE-2}} = 2 \times \frac{P_{\text{ROUGE-2}} \times R_{\text{ROUGE-2}}}{P_{\text{ROUGE-2}} + R_{\text{ROUGE-2}}} \quad (6)$$

c. ROUGE-L
Digunakan untuk mengukur sistem dengan melihat urutan kata terpanjang yang muncul pada luaran sistem (*generation*) dan referensi jawaban (*ground truth*).

$$P_{\text{ROUGE-L}} = \frac{\text{LCS}(\text{result,reference})}{\text{Count}_{\text{total}}(\text{unigram}_{\text{result}})} \quad (7)$$

$$R_{\text{ROUGE-L}} = \frac{\text{LCS}(\text{hasil,referensi})}{\text{Count}_{\text{total}}(\text{unigram}_{\text{referensi}})} \quad (8)$$

$$F1_{\text{ROUGE-L}} = 2 \times \frac{P_{\text{ROUGE-L}} \times R_{\text{ROUGE-L}}}{P_{\text{ROUGE-L}} + R_{\text{ROUGE-L}}} \quad (9)$$

3. HASIL DAN PEMBAHASAN

3.1 Hasil Pengujian dan Evaluasi Sistem

Evaluasi kinerja sistem RAG difokuskan untuk mengukur kapabilitas *Large Language Model* (LLM) dalam mensintesis jawaban yang akurat dan relevan berdasarkan pertanyaan *user* dan konteks yang disuplai oleh subsistem *retrieval*. Metode ROUGE dipilih untuk membandingkan secara linguistik antara luaran sistem (*output*) dengan referensi jawaban (*ground truth*). Tipe pertanyaan disusun sedemikian rupa untuk menguji sistem dengan tipe dan karakterisitk data yang berbeda,

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1264-1272
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1088

antara data pedoman akademik dan data akademik mahasiswa. Tabel 1 memberikan gambaran atau sampel pertanyaan yang diujikan kedalam sistem RAG.

Tabel 1. Sampel Data Pengujian Sistem

Pertanyaan	Referensi (*ground Truth*)
Berapakah IPK dari Siti Aminah, dan apakah nilai tersebut sudah melewati batas minimal kelulusan akademik?	IPK Siti Aminah adalah 3.85. Nilai ini sudah melewati batas minimal kelulusan akademik yang mensyaratkan IPK paling rendah 2.00.
Jika Lia Amalia adalah mahasiswa angkatan 2020, berapa sisa masa studi maksimal yang dimilikinya sebelum terancam sanksi Drop Out?	Masa studi maksimal program Sarjana adalah 7 tahun (14 semester). Karena Lia Amalia adalah angkatan 2020, maka batas akhir masa studinya adalah hingga tahun 2027 (semester 14)
Apakah mahasiswa bernama Candra Wijaya dengan total 42 SKS sudah diperbolehkan untuk mengambil mata kuliah Tugas Akhir atau skripsi?	Belum diperbolehkan. Tugas Akhir merupakan syarat akhir kelulusan yang diambil setelah mahasiswa menyelesaikan sebagian besar beban studi. Dengan baru mengumpulkan 42 SKS, Candra Wijaya belum memenuhi kriteria kelayakan akademik untuk menempuh skripsi.
Rani Mulyani memiliki IPK sebesar 3.9. Apakah nilai IPK tersebut sudah memenuhi standar minimum kelulusan yang ditetapkan kampus?	Ya, sangat memenuhi. Batas minimum Indeks Prestasi Kumulatif (IPK) yang disyaratkan untuk bisa menyelesaikan program sarjana adalah paling rendah 2.00.
Siapakah nama mahasiswa yang memiliki NIM 112022010 dan berapakah sisa waktu studi normalnya jika ia berada di angkatan 2022?	Mahasiswa dengan NIM 112022010 adalah Nisa Rahmawati. Sebagai angkatan 2022, masa studi maksimalnya (14 semester/7 tahun) akan berakhir pada tahun 2029.

Dengan mempertimbangkan aspek kedaulatan data, semua data akademik yang dipakai pada tahap ini merupakan data sintesis yang dibuat dengan meniru komponen data asli. Selanjutnya data uji yang digunakan, sebanyak 60 buah disintesis dengan menggunakan bantuan AI (*Claude Sonet 4.6*). Hal ini bertujuan agar data uji yang digunakan bisa semaksimal mungkin menguji sistem, baik dalam aspek leksikal dan semantik.

Tabel 2. *ROUGE Evaluation Summary*

Metrik	Average Precision	Average Recall	Average F1-Score
ROUGE-1	0.36	0.48	0.38
ROUGE-2	0.16	0.22	0.17
ROUGE-L	0.29	0.39	0.30

Hasil pengujian sistem menggunakan tiga ukuran ROUGE tersaji pada Tabel 2. Pada ROUGE-1 yang mengukur overlap unigram, sistem mencatatkan rata-rata *precision* 0,36, *recall* 0,48, dan *f1-score* 0,38. Nilai ROUGE-2 yang mengukur overlap bigram berurutan turun secara signifikan dengan *f1-score* 0,17, yang mencerminkan keterbatasan sistem dalam mereproduksi frasa atau rangkaian kata spesifik dari jawaban referensi. ROUGE-L yang mengukur kesamaan urutan kata terpanjang (*Longest Common Subsequence*) mencatatkan *f1-score* 0,30, berada di antara kedua metrik sebelumnya. Penurunan bertahap dari ROUGE-1 ke ROUGE-2 ini merupakan pola yang lazim ditemukan pada sistem generatif, mengingat semakin panjang unit pencocokan (n-gram), semakin kecil probabilitas kemunculan urutan kata yang identik antara dua teks yang ditulis secara independen.

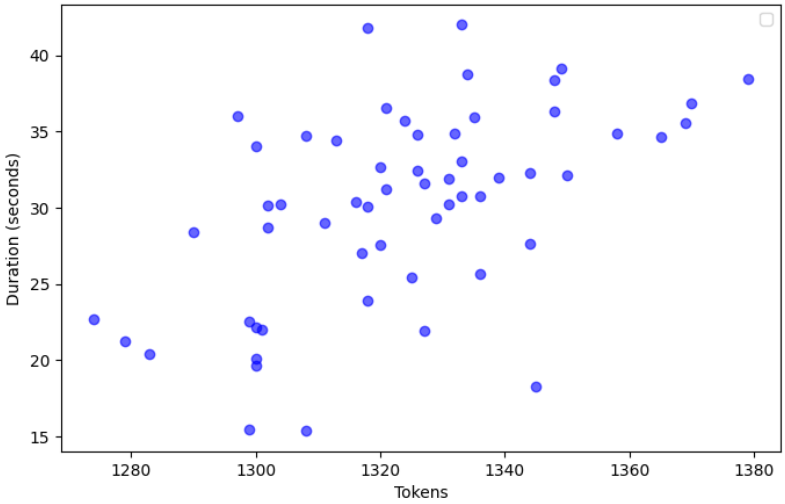
Pola asimetri antara *precision* dan *recall* yang konsisten pada seluruh metrik perlu diinterpretasikan dengan kehati-hatian. Nilai *recall* yang secara konsisten lebih tinggi dari *precision*, dengan selisih rata-rata sekitar 0,10 sampai 0,12 poin pada seluruh metrik, secara teknis hanya menunjukkan bahwa sebagian token dalam jawaban referensi berhasil ditemukan kembali dalam respons sistem. Namun demikian, ROUGE tidak dapat membedakan apakah token tersebut hadir dalam konteks yang benar secara faktual, atau sekadar terjadi *overlap* leksikal secara kebetulan. Nilai *precision* yang lebih rendah mengindikasikan bahwa respons sistem mengandung proporsi token yang cukup besar namun tidak terdapat dalam referensi, yang dapat berupa elaborasi tambahan, sinonim, atau reformulasi kalimat. Oleh karena itu, klaim bahwa sistem berhasil melakukan *retrieval* dan sintesis informasi secara substansial tidak dapat ditopang oleh metrik ROUGE semata.

Yang lebih relevan untuk didiskusikan adalah akar dari asimetri ini, yang bersumber dari kesenjangan kapasitas generasi antara kedua model. Jawaban referensi dihasilkan oleh Claude Sonnet 4.6, sebuah model berskala besar dengan kapasitas generasi bahasa yang jauh melampaui Mistral 7B. Model dengan skala lebih besar cenderung menghasilkan jawaban yang lebih elaboratif, kohesif, dan kaya diksi. Akibatnya, terdapat *generation style gap* yang inheren antara referensi dan *output* sistem: Mistral 7B menghasilkan respons yang lebih ringkas dengan pilihan kata berbeda, sehingga overlap leksikal yang tertangkap ROUGE menjadi terbatas meskipun informasi inti yang disampaikan berpotensi ekuivalen secara semantik. Kondisi ini memperkenalkan bias evaluasi yang perlu diakui secara eksplisit sebagai limitasi metodologis penelitian ini. Secara praktis, skor ROUGE yang dihasilkan merupakan ukuran kemiripan gaya penulisan (stylistic similarity) sekaligus kemiripan konten, sehingga skor yang rendah tidak dapat sepenuhnya dipisahkan dari

BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 6, No 4, June 2026 | Hal 1264-1272
https://hostjournals.com/bulletincsr
DOI: 10.47065/bulletincsr.v6i4.1088

pengaruh perbedaan kapasitas generasi kedua model. Idealnya, penelitian lanjutan perlu melengkapi evaluasi ini dengan jawaban referensi yang disusun oleh domain *expert* secara manual, atau menggunakan model referensi dengan skala yang sebanding dengan Mistral 7B, agar perbandingan yang dihasilkan lebih adil dan representatif.

Dari 60 query pengujian yang dijalankan, sistem mengalami kegagalan parsial pada 2 *query* (3,3%). Kegagalan ini tidak bersifat total, sistem masih mampu menyusun rencana pencarian (*query formulation*) dan merespons secara parsial, namun gagal menyelesaikan siklus *tool-calling* hingga tahap eksekusi *retrieval* yang sesungguhnya. Secara teknis, agen berhasil memasuki fase penalaran (*reasoning*) dan menentukan alat pencarian yang seharusnya dipanggil, tetapi proses pemanggilan alat tidak berhasil dieksekusi sehingga jawaban akhir yang dihasilkan tidak berlandaskan konteks dokumen yang relevan. Fenomena ini kemungkinan besar dipicu oleh dua faktor yang saling berkaitan: pertama, ambiguitas instruksi pada *system prompt* yang tidak cukup tegas mendefinisikan kondisi *fallback* ketika *tool-calling* gagal; dan kedua, beban konteks yang tinggi akibat panjang *prompt* yang mencapai 1.270—1.380 token, yang berpotensi menyebabkan model kehilangan fokus terhadap instruksi *routing* utama pada segmen akhir *context window*-nya. Meskipun proporsinya kecil (3,3%), kasus ini mengindikasikan adanya celah pada robustisitas komponen *agentic* sistem yang perlu ditangani, terutama melalui penambahan mekanisme *retry* otomatis dan penguatan instruksi penanganan kesalahan (*error-handling fallback*) pada *system prompt.*



Gambar 6. Perbandingan jumlah Token dengan latensi waktu *generated system*

Di luar evaluasi kualitas *output*, aspek performa komputasional sistem juga dianalisis melalui hubungan antara panjang *prompt token* dengan latensi respons. Berdasarkan Gambar 6, distribusi *prompt token* pada seluruh *query* pengujian terkonsentrasi pada rentang 1.270 sampai 1.380 token. Rentang ini mencerminkan karakteristik arsitektur RAG yang dibangun, di mana setiap *query* membawa beban konteks yang besar akibat penggabungan dokumen hasil *retrieval* ke dalam prompt.

Dari grafik tersebut terlihat bahwa latensi sistem berada pada rentang 15 sampai 42 detik dengan rata-rata sekitar 30 detik per *query*. Yang menarik, pola *scatter* tidak menunjukkan korelasi linear yang kuat antara jumlah token dengan durasi, titik-titik data tersebar vertikal secara lebar pada rentang token yang sama. Hal ini mengindikasikan bahwa latensi tidak semata-mata ditentukan oleh panjang *prompt*, melainkan juga dipengaruhi oleh faktor lain seperti kompleksitas token yang harus di-generate sebagai output dan kondisi komputasi pada saat inferensi berlangsung.

Latensi rata-rata sekitar 30 detik per *query* pada dasarnya dapat ditekan secara signifikan melalui penerapan mekanisme *keep-alive* pada Ollama. Tanpa konfigurasi ini, model Mistral 7B berpotensi di-*unload* dari memori di antara *request*, sehingga setiap *query* memicu proses *cold start* yang menambah overhead hingga 40% dari total durasi sebagaimana teridentifikasi pada tahap analisis metadata. Dengan mengaktifkan *keep-alive*, model tetap ter-*load* di memori selama sesi pengujian sehingga seluruh durasi dapat dialokasikan murni untuk proses *prompt evaluation* dan *token generation*.

3.2 Pembahasan

Berdasarkan hasil evaluasi yang telah dipaparkan, terdapat tiga temuan utama yang perlu didiskusikan secara mendalam. Pertama, nilai ROUGE yang dihasilkan sistem menunjukkan performa yang terbatas, dengan ROUGE-L *f1-score* sebesar 0,29 sebagai ukuran yang paling representatif. Sebagaimana telah dianalisis, keterbatasan ini tidak dapat dilepaskan dari kesenjangan kapasitas generasi bahasa antara Mistral 7B sebagai *backbone* sistem dengan Claude Sonnet 4.6 yang digunakan sebagai generator jawaban referensi. Perbedaan skala model yang sangat signifikan ini memperkenalkan *generation style gap* yang turut menekan skor ROUGE secara struktural, terlepas dari kualitas aktual jawaban sistem. Kondisi ini menjadi limitasi metodologis yang perlu diakui, dan idealnya evaluasi dilengkapi dengan pengujian menggunakan jawaban referensi yang disusun secara manual oleh domain *expert* untuk memperoleh gambaran yang lebih adil.

Kedua, keterbatasan inheren metrik ROUGE sebagai satu-satunya instrumen evaluasi perlu menjadi perhatian. ROUGE hanya mengukur *overlap* leksikal pada level n-gram dan tidak mampu menangkap kebenaran semantik maupun relevansi faktual dari jawaban yang dihasilkan. Sebuah respons yang secara substansi keliru namun kebetulan berbagi beberapa token dengan referensi akan tetap mendapat skor positif, begitu pula sebaliknya. Oleh karena itu, evaluasi berbasis *LLM-as-a-Judge* diperlukan sebagai pelengkap pada penelitian lanjutan, di mana model dengan kemampuan penalaran tinggi ditugaskan untuk menilai relevansi, akurasi faktual, dan koherensi jawaban secara holistik, sesuatu yang tidak dapat dilakukan oleh metrik berbasis statistik seperti ROUGE.

Ketiga, sistem menunjukkan celah robustitas yang tercermin dari kegagalan pada 2 dari 60 *query* pengujian (3,3%). Kegagalan ini tidak bersifat acak, melainkan berkorelasi dengan karakteristik prompt yang panjang. Dengan rata-rata *prompt token* berada di rentang 1.270-1.380 token per *query*, sistem menanggung beban konteks yang besar pada setiap inferensi. Beban ini tidak hanya berdampak pada latensi yang rata-rata mencapai 30 detik, tetapi juga berpotensi menurunkan kualitas output secara keseluruhan. *Prompt* yang terlalu panjang dapat menyebabkan model kehilangan fokus terhadap instruksi utama, yang pada kasus ekstrem berujung pada kegagalan agen dalam menyelesaikan siklus *tool-calling* hingga tahap generasi jawaban. Temuan ini mengindikasikan bahwa optimasi panjang konteks merupakan variabel kritis yang belum dieksplorasi dalam penelitian ini.

4. KESIMPULAN

Penelitian ini berhasil membangun sistem *Retrieval-Augmented Generation* (RAG) berbasis *open-source* yang dijalankan secara lokal sebagai solusi otomatisasi layanan informasi akademik. Sistem mengintegrasikan LLM Mistral-7B-Instruct dengan pendekatan *hybrid retrieval* dalam ekosistem Elasticsearch, memadukan *dense retrieval* berbasis IndoBERT untuk dokumen pedoman akademik dan *sparse retrieval* berbasis BM25 untuk data terstruktur mahasiswa. Evaluasi terhadap 60 *query* uji menggunakan metrik ROUGE menghasilkan ROUGE-L *f1-score* sebesar 0,29 dengan tingkat keberhasilan sistem menjawab *query* 58 dari 60 *query*. Pola asimetri *recall* lebih tinggi dari *precision* yang konsisten diidentifikasi sebagai dampak dari *generation style gap* antara Mistral 7B dengan model referensi Claude Sonnet 4.6, sekaligus menjadi limitasi metodologis yang perlu diakui. Latensi rata-rata 30 detik per *query* menjadi tantangan performa utama yang dikontribusikan oleh beban konteks *prompt* pada rentang 1.270 sampai 1.380 token. Penelitian ini memberikan kontribusi dan dapat dijadikan sebagai *baseline* awal pengembangan sistem RAG berbahasa Indonesia menggunakan infrastruktur *open-source* sepenuhnya. Beberapa arah pengembangan yang diidentifikasi antara lain: eksplorasi *sweet spot* parameter *retrieval* mencakup nilai *top-k* dan panjang *chunk* yang optimal; investigasi performa Mistral 7B pada tugas berbahasa Indonesia beserta kemungkinan *fine-tuning* dengan korpus lokal; adopsi kerangka evaluasi yang lebih komprehensif seperti RAGAS atau *LLM-as-a-Judge*; serta penyelarasan spesifikasi *hardware* dengan kebutuhan komputasi *tools* yang digunakan.

REFERENCES

[1] C. Sunaengsih, A. Komariah, D. A. Kurniady, M. Thahir, and B. Tamam, "Academic Service Quality Survey in Higher Education," in *Advances in Social Science, Education and Humanities Research*, Atlantis Press, 2021, pp. 193–198. doi: 10.2991/assehr.k.210212.041.

[2] G. Gray, A. E. Schalk, G. Cooke, P. Murnion, P. Rooney, and K. C. O'Rourke, "Stakeholders' Insights on Learning Analytics: Perspectives of Students and Staff," *Comput. Educ.*, vol. 187, p. 104550, Oct. 2022, doi: 10.1016/j.compedu.2022.104550.

[3] L. Yan *et al.*, "VizChat: Enhancing Learning Analytics Dashboards with Contextualised Explanations Using Multimodal Generative AI Chatbots," in *ResearchGate Preprint*, Springer, Cham, 2024, pp. 180–193. doi: 10.1007/978-3-031-64299-9_13.

[4] S. Samaranayake, A. D. A. Gunawardena, and R. R. Meyer, "An Interactive Decision Support System for College Degree Planning," *Athens Journal of Education*, vol. 10, no. 1, pp. 101–116, Jan. 2023, doi: 10.30958/aje.10-1-6.

[5] Prashant, M. Poriye, P. Mittal, and N. Sharma, "Automating University Administration: A Systematic Review of Chatbot Applications in Higher Education," in *Proceedings of the 3rd International Conference on Artificial Intelligence, Machine Learning and Cybersecurity*, Nov. 2025, pp. 314–323. doi: 10.21467/proceedings.7.6.36.

[6] N. Chafiq, M. Ghazouani, and R. El Gounidi, "From Manual Review to AI Automation: An NLP-Powered System for Efficient CV Processing in Academic Admissions," *LatIA*, vol. 3, p. 315, May 2025, doi: 10.62486/latia2025315.

[7] J. D. S. Marques, A. V. Duarte, A. Carvalho, G. Rocha, B. Martins, and A. L. Oliveira, "Leveraging LLMs to Streamline the Review of Public Funding Applications," Oct. 2025, [Online]. Available: http://arxiv.org/abs/2510.09674

[8] A. Vallejo Blanxart and R. Nicolas Sans, "The Role of Generative AI Chatbots in Higher Education: A Student-Centric Conceptual Analysis of Benefits, Ethics, and Privacy Concerns," *J. Technol. Sci. Educ.*, vol. 15, no. 3, p. 810, Dec. 2025, doi: 10.3926/jotse.3643.

[9] J. Dempere, K. Modugu, A. Hesham, and L. K. Ramasamy, "The Impact of ChatGPT on Higher Education," *Front. Educ. (Lausanne).*, vol. 8, p. 1206936, Sep. 2023, doi: 10.3389/feduc.2023.1206936.

[10] M. Maryamah, M. M. Irfani, E. B. Tri Raharjo, N. A. Rahmi, M. Ghani, and I. K. Raharjana, "Chatbots in Academia: A Retrieval-Augmented Generation Approach for Improved Efficient Information Access," in *2024 16th International Conference on Knowledge and Smart Technology (KST)*, IEEE, Feb. 2024, pp. 259–264. doi: 10.1109/KST61284.2024.10499652.

[11] I. Siragusa and R. Pirrone, "Unipa-GPT: Large Language Models for University-Oriented QA in Italian," *Italian Journal of Computational Linguistics*, vol. 10, no. 2, p. 107, Apr. 2025, doi: 10.17454/IJCOL102.06.

[12] D. Thüs, S. Malone, and R. Brünken, "Exploring Generative AI in Higher Education: A RAG System to Enhance Student Engagement with Scientific Literature," *Front. Psychol.*, vol. 15, p. 1474892, Oct. 2024, doi: 10.3389/fpsyg.2024.1474892.

[13] M. Alier, J. Pereira, F. J. García-Peñalvo, M. J. Casañ, and J. Cabré, "LAMB: An Open-Source Software Framework to Create Artificial Intelligence Assistants Deployed and Integrated into Learning Management Systems," *Comput. Stand. Interfaces*, vol. 92, p. 103940, Mar. 2025, doi: 10.1016/j.csi.2024.103940.
[14] J. Swacha and M. Gracel, "Retrieval-Augmented Generation (RAG) Chatbots for Education: A Survey of Applications," *Applied Sciences*, vol. 15, no. 8, p. 4234, Apr. 2025, doi: 10.3390/app15084234.
[15] A. Vinayan Kozhipuram, S. Shailendra, and R. Kadel, "Retrieval-Augmented Generation vs. Baseline LLMs: A Multi-Metric Evaluation for Knowledge-Intensive Content," *Information*, vol. 16, no. 9, p. 766, Sep. 2025, doi: 10.3390/info16090766.
[16] R. Dayarathne, U. Ranaweera, and U. Ganegoda, "Comparing the Performance of LLMs in RAG-Based Question-Answering: A Case Study in Computer Science Literature," in *Technology Integration in Higher Education*, vol. 228, Springer, Singapore, 2025, pp. 387–403. doi: 10.1007/978-981-97-9255-9_26.
[17] A. Q. Jiang *et al.*, "Mistral 7B," Oct. 2023, [Online]. Available: http://arxiv.org/abs/2310.06825
[18] J. Alammar and M. Grootendorst, *Hands-On Large Language Models: Language Understanding and Generation*. O'Reilly Media, Inc., 2024.
[19] V. Karpukhin *et al.*, "Dense Passage Retrieval for Open-Domain Question Answering," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Stroudsburg, PA, USA: Association for Computational Linguistics, Sep. 2020, pp. 6769–6781. doi: 10.18653/v1/2020.emnlp-main.550.
[20] C. Zhai and S. Massung, *Text Data Management and Analysis: A Practical Introduction to Information Retrieval and Text Mining*. New York, NY, USA: Association for Computing Machinery and Morgan & Claypool, 2016. doi: 10.1145/2915031.
[21] "Dense Vector Field Type," Elasticsearch Reference. Accessed: Apr. 22, 2026. [Online]. Available: https://www.elastic.co/docs/reference/elasticsearch/mapping-reference/dense-vector
[22] "Thinking in LangGraph," LangChain Documentation. Accessed: Apr. 19, 2026. [Online]. Available: https://docs.langchain.com/oss/python/langgraph/thinking-in-langgraph
[23] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81. Accessed: Jun. 07, 2026. [Online]. Available: https://aclanthology.org/W04-1013/