Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

Optimasi Model Retrieval-Augmented Generation Menggunakan Algoritma Indeks HNSW Lokal pada Small Language Model untuk Mitigasi Halusinasi Hadis Bukhari

Rizqi Ari Putra*, Suhartono, Muhammad Faisal

Fakultas Sains dan Teknologi, Magister Informatika, Universitas Islam Negeri Maulana Malik Ibrahim Malang, Malang
Jl. Gajayana No. 50, Dinoyo, Kec. Lowokwaru, Kota Malang, Jawa Timur, Indonesia
Email: [1*]riezq.25@gmail.com, [2]suhartono@ti.uin-malang.ac.id , [3]mfaisal@ti.uin-malang.ac.id
Email Penulis Korespondensi: riezq.25@gmail.com
Submitted: **26/06/2026**; Accepted: **12/07/2026**; Published: **14/07/2026**

Abstrak−Small Language Model (SLM) menawarkan efisiensi komputasi lokal yang tinggi, namun memiliki kerentanan sistemik terhadap gejala halusinasi informasi. Kerentanan ini menjadi risiko kritis ketika model diterapkan pada domain sensitif yang menuntut akurasi mutlak, seperti hukum syariat dan literatur suci Islam. Pengutipan teks keagamaan yang tidak akurat dapat berdampak pada penyesatan teologis. Untuk mengatasi masalah tersebut, penelitian ini mengusulkan solusi arsitektur Retrieval-Augmented Generation (RAG) lokal bermemori rendah. Sistem ini dirancang dengan memadukan model open-source Llama-3.2-1B-Instruct dan basis data vektor PostgreSQL pgvector. Optimasi pencarian dokumen dilakukan berbasis algoritma indeks HNSW (Hierarchical Navigable Small World) serta teknik kuantisasi presisi 16-bit (halfvec) terhadap 7.003 chunks korpus Hadis Sahih Bukhari. Tujuan utama penelitian ini adalah merancang sistem mitigasi halusinasi berpresisi tinggi yang beroperasi secara mandiri (on-premise), sekaligus memberikan kontribusi nyata pada pengembangan asisten teologis digital berbiaya rendah yang menjaga privasi dan kedaulatan data. Efikasi mitigasi dievaluasi secara otomatis menggunakan kerangka kerja Ragas terhadap 50 kueri uji teologis, sementara efisiensi basis data diuji secara fisik pada perangkat keras komputer lokal kelas konsumen. Hasil eksperimen menunjukkan arsitektur usulan mampu meningkatkan metrik faithfulness secara signifikan sebesar 117,4% (dari 0,4120 menjadi 0,8960) dan answer relevance sebesar 50,0% (dari 0,6120 menjadi 0,9180), sekaligus mempercepat waktu tanggap inferensi hingga 50,7%. Pada sisi basis data, kuantisasi halfvec berhasil mereduksi ruang penyimpanan tabel fisik sebesar 36,2% dan mempercepat waktu konstruksi indeks sebesar 16,5% dengan tingkat pertahanan akurasi (recall) mutlak 1,0000. Penelitian ini membuktikan bahwa asisten virtual keagamaan berpresisi tinggi sangat layak dieksekusi secara mandiri tanpa bergantung pada layanan komputasi awan pihak ketiga.

Kata Kunci: Hadis Bukhari; Halusinasi Model; Kuantisasi Vektor; Retrieval-Augmented Generation; Small Language Model

Abstract−Small Language Models (SLMs) offer high local computational efficiency but possess a systemic vulnerability to information hallucination. This vulnerability becomes a critical risk when the model is applied to sensitive domains demanding absolute accuracy, such as Sharia law and Islamic sacred literature. The inaccurate citation of religious texts can lead to theological misguidance. To address this issue, this study proposes a low-memory, local Retrieval-Augmented Generation (RAG) architectural solution. This system is designed by integrating the open-source Llama-3.2-1B-Instruct model with the PostgreSQL pgvector vector database. Document retrieval optimization is performed based on the HNSW (Hierarchical Navigable Small World) indexing algorithm and a 16-bit precision quantization technique (halfvec) on 7,003 chunks of the Sahih al-Bukhari Hadith corpus. The primary objective of this research is to design a high-precision hallucination mitigation system that operates independently (on-premise), while making a tangible contribution to the development of a low-cost digital theological assistant that preserves privacy and data sovereignty. Mitigation efficacy was automatically evaluated using the Ragas framework against 50 theological test queries, while database efficiency was physically tested on consumer-grade local computer hardware. Experimental results indicate that the proposed architecture is capable of significantly improving the faithfulness metric by 117.4% (from 0.4120 to 0.8960) and answer relevance by 50.0% (from 0.6120 to 0.9180), while simultaneously accelerating inference response time by up to 50.7%. On the database side, halfvec quantization successfully reduced physical table storage space by 36.2% and accelerated index construction time by 16.5% with an absolute accuracy (recall) retention rate of 1.0000. This study proves that a high-precision religious virtual assistant is highly feasible to execute independently without relying on third-party cloud computing services.

Keywords: Hadith Bukhari; Hallucination Mitigation; Retrieval-Augmented Generation; Small Language Model; Vector Quantization

1. PENDAHULUAN

Perkembangan teknologi kecerdasan buatan, khususnya Large Language Models (LLM), telah merevolusi cara manusia berinteraksi dengan sistem pencarian informasi berbasis teks. Namun, di balik kemampuan generatifnya yang luar biasa, LLM secara inheren memiliki kerentanan kritis yang dikenal sebagai fenomena "halusinasi". Fenomena ini terjadi ketika model menghasilkan informasi yang secara linguistik terlihat fasih dan meyakinkan, namun secara faktual keliru atau sama sekali tidak memiliki basis kebenaran empiris. Dalam ranah teks keagamaan seperti hukum Islam dan studi literatur Hadis, fenomena halusinasi ini merupakan masalah krusial dengan toleransi nol (zero-tolerance issue). Kesalahan model dalam menyebutkan rantai perawi (sanad) atau memanipulasi isi sabda (matan) pada kitab suci primer seperti Hadis Sahih Bukhari tidak hanya berakibat pada misinformasi biasa, melainkan berpotensi memicu penyesatan fatwa dan krisis otoritas teologis di tengah masyarakat.

Untuk mengatasi masalah halusinasi tersebut, arsitektur Retrieval-Augmented Generation (RAG) saat ini diakui sebagai solusi standar di industri Natural Language Processing (NLP) [1]. RAG memitigasi halusinasi

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

dengan mencari dokumen relevan dari basis data eksternal terlebih dahulu, kemudian memaksa model bahasa untuk merangkai jawaban secara eksklusif berdasarkan dokumen rujukan tersebut [1], [2]. Akan tetapi, implementasi RAG konvensional seringkali masih didominasi oleh penggunaan layanan API LLM komersial berbasis komputasi awan (cloud-based). Ketergantungan ini memunculkan tantangan nyata berupa risiko privasi dan kedaulatan data lembaga, serta tingginya biaya beban operasional token inferensi. Sebagai solusi atas masalah tersebut, penelitian ini mengusulkan sebuah pengembangan sistem RAG yang sepenuhnya berjalan secara mandiri dan lokal (on-premise) menggunakan Small Language Model (SLM) ultra-ringan Llama-3.2-1B-Instruct [3]. Untuk menjamin kecepatan komputasi dan efisiensi memori perangkat keras lokal, optimasi diterapkan pada lapisan penyimpanan menggunakan basis data vektor pgvector berindeks HNSW (Hierarchical Navigable Small World) [4], dengan mengonversi tipe data embedding (OpenAI text-embedding-3-large 3072 dimensi) [5] menjadi format halfvec (16-bit floating-point).

Berbagai penelitian terdahulu telah berupaya mengeksplorasi penggunaan AI dalam domain literatur Islam untuk menekan tingkat halusinasi. Dalam studi ekstensif yang dilakukan oleh Sengupta et al (2023), disoroti bahwa LLM berparameter besar pun sering gagal dalam membedakan hierarki yurisprudensi linguistik historis Arab pada kitab-kitab Islam, sehingga RAG mutlak diperlukan untuk membangun sistem AI Islami yang tahan terhadap manipulasi [6]. Penelitian lain oleh Khalila et al (2025) mengeksplorasi performa 13 jenis open-source LLM dalam studi Al-Qur'an. Mereka menemukan bahwa penerapan framework RAG mampu memperbaiki kelemahan base model dan secara dramatis meningkatkan keandalan jawaban melalui pengambilan konteks berbasis vektor [7]. Di sisi lain, riset yang dikembangkan oleh Elrefai et al (2025) pada ajang IslamicEval menekankan bahwa penerapan Information Retrieval pada teks Hadis memiliki kompleksitas yang jauh lebih menantang dibandingkan teks Al-Qur'an, mengingat ekstraksi Hadis memerlukan preservasi struktural yang ketat pada riwayat sanad dan teks matan secara bersamaan [8].

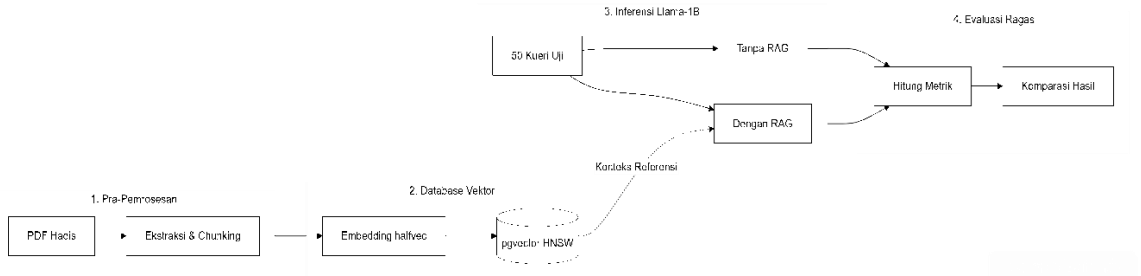
Dari perspektif metrik evaluasi keandalan teks, Shahul Es et al. (2024) memperkenalkan kerangka kerja Ragas (Retrieval-Augmented Generation Assessment). Ragas menjadi standar baru karena memungkinkan evaluasi metrik Faithfulness dan Answer Relevance secara otomatis dengan pendekatan LLM-as-a-judge [9], sehingga menghilangkan keharusan anotasi manual oleh manusia [10]. Sementara itu, dalam aspek optimasi infrastruktur database, penelitian Jiang et al. (2025) membuktikan bahwa algoritma pencarian seperti indeks HNSW mampu mengeksekusi pencarian kemiripan vektor (similarity search) dengan latensi milidetik [11]. Meski begitu, implementasi HNSW standar menelan kapasitas RAM yang terlampau besar, sehingga teknik efisiensi kuantisasi dimensi pada basis data menjadi syarat wajib untuk penerapan server mandiri [1], [4].

Berdasarkan kelima kajian literatur terkini di atas, ditemukan suatu GAP Analysis (celah penelitian) yang krusial. Mayoritas penelitian sistem RAG untuk dokumen keislaman saat ini masih terpusat pada evaluasi akurasi LLM berbasis awan dengan ukuran puluhan miliar parameter. Kalaupun ada penelitian yang memanfaatkan model kelas lokal (di bawah 3 miliar parameter), mereka jarang mengoptimalkan arsitektur penyimpanan vektornya untuk penghematan memori, seperti pemanfaatan indeks HNSW halfvec secara spesifik. Lebih jauh, evaluasi efektivitas struktur pemotongan teks (chunking) [12] berbasis objek PDF Markdown pada Hadis Bukhari menggunakan matriks Ragas yang memilah kategori pengujian menjadi pertanyaan literal, konseptual, dan hierarkis secara komparatif masih sangat minim dieksplorasi.

Oleh karena itu, tujuan utama dari penelitian ini adalah merancang, mengimplementasikan, dan mengoptimalkan sistem RAG lokal bermemori rendah (memory-efficient) guna memitigasi halusinasi pada pemrosesan teks Hadis Sahih Bukhari. Penelitian ini diharapkan dapat membuktikan secara empiris bahwa arsitektur pencarian HNSW halfvec yang dipadukan dengan Llama-3.2-1B-Instruct mampu mencapai nilai pengujian Ragas (Faithfulness dan Answer Relevance) yang tinggi. Harapan akhirnya, luaran dari penelitian ini dapat menjadi referensi blueprint bagi instansi pendidikan Islam atau lembaga fatwa untuk membangun chatbot layanan keagamaan yang aman dari halusinasi teologis, berbiaya rendah, mandiri secara komputasi, dan sangat terjamin privasi datanya.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian



Gambar 1. Alur Tahapan Penelitian Optimasi RAG Lokal Berbasis Indeks HNSW

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

Penelitian ini dirancang menggunakan pendekatan eksperimental komputasional yang sistematis untuk memitigasi fenomena halusinasi pada pemrosesan teks literatur Hadis. Secara garis besar, metodologi penelitian dieksekusi melalui empat tahapan utama yang saling terintegrasi, yaitu: (1) Pra-pemrosesan dan Pemotongan Semantik Korpus Hadis; (2) Pembangkitan Vektor dan Pengindeksan Hierarchical Navigable Small World (HNSW) Bermemori Rendah; (3) Implementasi Inferensi Arsitektur Retrieval-Augmented Generation (RAG) Lokal; dan (4) Evaluasi Otomatis Berbasis Metrik Ragas. Alur penerapan metode dan solusi penelitian ini divisualisasikan secara komprehensif pada **Error! Reference source not found.**.

2.2 Pengumpulan dan Pra-pemrosesan Data

Data primer yang digunakan dalam penelitian ini adalah korpus digital Hadis Sahih Bukhari versi terjemahan Bahasa Indonesia. Dokumen sumber diperoleh secara mandiri dalam format berkas Portable Document Format (PDF) dengan karakteristik tata letak blok sekuensial vertikal (interleaved block layout), di mana teks matan asli berbahasa Arab ditempatkan pada baris atas, lalu diikuti langsung secara linier oleh teks blok terjemahan bahasa Indonesia di bawahnya untuk setiap nomor hadis. Tahap awal pemrosesan data (data ingestion) berfokus pada transformasi dokumen tidak terstruktur (unstructured data) tersebut menjadi format semi-terstruktur berbasis teks format Markdown (.md). Proses ekstraksi biner PDF ke bentuk teks murni ini dieksekusi memanfaatkan pustaka Python PyMuPDF (fitz) untuk membaca aliran objek teks berdasarkan hierarki koordinat halaman dokumen asli.

Setelah seluruh teks berhasil diekstrak ke dalam representasi berkas tekstual digital, tahapan berikutnya yang krusial adalah melakukan pembersihan data (data cleaning) intensif. Tahapan prapemrosesan ini merupakan standar fundamental dalam keilmuan Information Retrieval guna memurnikan korpus dari kebisingan tekstual (textual noise) yang dapat mendegradasi presisi sistem [2], [13]. Proses pembersihan data diotomatisasi menggunakan skrip pemrograman Python dengan fokus penyelesaian pada tiga klaster anomali dokumen.

Tahapan pertama dalam pembersihan data adalah eliminasi catatan kaki dan metadata repositori. Pada tahap ini, berkas hasil konversi awal dibersihkan dari kemunculan string repetitif non-hadis, seperti promosi situs web, tautan eksternal, dan penomoran halaman biner. Proses penyaringan teks ini dilakukan secara presisi dengan menggunakan fungsi pencocokan pola Ekspresi Reguler (Regular Expression atau Regex).

Langkah selanjutnya difokuskan pada rekonstruksi dan penyatuan karakter Arab (de-spacing Arabic text). Akibat pembacaan koordinat biner bounding box internal pada berkas PDF asli, blok karakter Arab kerap mengalami disintegrasi spasial yang berupa pemisahan huruf dan harakat oleh spasi acak atau pemutusan baris vertikal. Untuk mengatasi anomali tersebut, skrip pembersihan secara otomatis mendeteksi blok pada rentang Unicode Arab (\u0600-\u06FF), menghapus spasi internal serta pemutusan baris (breakline) di dalamnya, lalu menyatukannya kembali menjadi struktur kalimat Arab yang utuh dan sahih secara linguistik.

Tahapan terakhir dalam proses ini adalah normalisasi spasi dan baris baru. Langkah ini dilakukan dengan menghapus spasi ganda (double spaces) serta baris baru yang berlebih (double breaklines). Normalisasi ini sangat penting diimplementasikan guna memastikan kontinuitas konteks narasi hadis tetap terjaga dengan baik saat model melakukan proses tokenisasi.

Pasca-proses pembersihan, dokumen teks murni tersebut dialirkan menuju tahapan pemotongan teks (chunking). Berbeda dengan arsitektur RAG konvensional .yang memotong teks secara kaku berdasarkan ambang batas karakter (fixed-size window chunking), penelitian ini menerapkan pendekatan pemotongan semantik fungsional (functional semantic splitting). Metode pemotongan berbasis jumlah karakter statis memiliki kelemahan fatal karena memotong teks secara buta tanpa mempertimbangkan batas logis kalimat, sehingga berisiko memisahkan untaian rantai perawi (sanad) dari isi pesan (matan) ke dalam potongan teks yang terisolasi. Secara teoretis, untuk menjaga integritas semantik dan kualitas representasi ruang vektor, pemotongan teks yang berbasis pada struktur kalimat logis terbukti jauh lebih superior dibandingkan pemotongan karakter yang statis [12], [14].

Oleh karena itu, dikembangkan algoritma Structural Boundary Parser berbasis aturan (rule-based approach) menggunakan pola jangkar ekspresi reguler awal baris berbasis pencocokan angka desimal dan karakter titik literal. Pola ini memetakan angka yang diikuti karakter titik literal dan spasi di awal baris baru, mencerminkan format penulisan sekuensial hadis. Melalui pendekatan batas struktural ini, setiap unit hadis diisolasi secara sempurna menjadi objek mandiri berpasangan yang mengikat nomor hadis, blok teks Arab utuh, serta keseluruhan teks narasi terjemahannya ke dalam satu kesatuan data (payload). Pemotongan sekunder berbasis kalimat (sentence-based sliding window) baru akan dipicu secara dinamis dengan ambang batas estimasi 450 token jika terdapat hadis dengan narasi dialog yang sangat panjang. Pembatasan ini bertujuan untuk menjaga kerapatan semantik (semantic density) dokumen di dalam ruang vektor, sekaligus memastikan tidak ada fragmen teks keagamaan yang terbuang (zero data loss).

2.3 Implementasi Basis Data Vektor Lokal (Vector Store & Indexing)

Dokumen korpus hadis yang telah dipotong (chunking) secara terstruktur pada tahap prapemrosesan tidak dapat diinterpretasikan secara langsung oleh model bahasa untuk keperluan pencarian makna. Teks tersebut harus diproyeksikan ke dalam ruang laten (latent space) berdimensi tinggi agar mesin dapat mengkalkulasi kedekatan semantik antar kalimat. Penelitian ini menggunakan model embedding text-embedding-3-large untuk mengubah setiap teks chunk menjadi representasi vektor berdimensi 3072. Model embedding ini terbukti memiliki performa

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

ekstraksi fitur semantik (semantic feature extraction) tingkat lanjut, khususnya pada penanganan dokumen linguistik yang kompleks [5], [15].

Penyimpanan dan pengelolaan vektor berdimensi tinggi tersebut dikelola secara lokal (on-premise) menggunakan sistem manajemen basis data relasional PostgreSQL yang diperkuat dengan ekstensi pgvector. Salah satu tantangan utama dalam implementasi arsitektur RAG lokal adalah tingginya beban konsumsi Random Access Memory (RAM) saat menyimpan jutaan parameter vektor. Pada sistem konvensional, representasi vektor disimpan menggunakan tipe data single-precision floating-point berukuran 32-bit (vector). Untuk mengoptimalkan arsitektur agar dapat dieksekusi dengan efisien pada perangkat keras tingkat konsumen, penelitian ini mengimplementasikan teknik kuantisasi memori tipe data 16-bit (half-precision floating-point atau halfvec). Secara teoretis, konversi ke half precision mampu mereduksi jejak memori (memory footprint) hingga 50% tanpa mengorbankan akurasi pencarian secara signifikan [16]. Struktur tabel yang dirancang untuk menampung data dokumen hadis beserta vektor semantiknya dapat dilihat pada Tabel 1.

Tabel 1. Struktur Skema Basis Data Vektor Lokal

Nama Kolom	Tipe Data	Keterangan Fungsional
id	serial4	Identitas unik (Primary Key) untuk setiap baris data (potongan hadis) yang diindeks otomatis.
no_hadis	text	Menyimpan nomor indeks referensi mutlak dari hadis untuk menjaga keterlacakan data asal.
content	text	Teks asli potongan (chunk) hadis yang memuat struktur sanad dan matan sebagai konteks RAG.
embedding	vector(3072)	Representasi numerik matematis hadis dari hasil luaran model OpenAI.
created_at	timestamptz	Penanda waktu kapan data tersebut dimuat ke dalam basis data sistem.

Sebagaimana diuraikan pada **Tabel 1**, data teks (content) dan representasi numeriknya (embedding) diikat dalam satu tabel yang sama untuk memudahkan proses pemanggilan. Untuk mempercepat waktu pencarian kemiripan (similarity search) pada data yang besar, sistem tidak menggunakan pemindaian tabel menyeluruh (Exact Nearest Neighbor), melainkan menerapkan algoritma pengindeksan Hierarchical Navigable Small World (HNSW) [4].

Pengindeksan HNSW ini dieksekusi menggunakan perintah parameter khusus: USING hnsw (((embedding)::halfvec(3072)) halfvec_cosine_ops) WITH (m = 16, ef_construction = 64). Dalam algoritma tersebut, operator halfvec_cosine_ops digunakan untuk mengevaluasi jarak kedekatan makna antara vektor kueri pengguna dengan vektor dokumen menggunakan perhitungan Cosine Similarity. Parameter m=16 menentukan jumlah maksimum koneksi dua arah yang dipertahankan untuk setiap elemen di dalam graf indeks, yang menyeimbangkan antara kecepatan pencarian dan ukuran indeks. Sementara itu, parameter ef_construction=64 menentukan ukuran daftar kandidat dinamis saat membangun struktur graf [4]. Pemanfaatan indeks HNSW halfvec ini sangat krusial karena mampu melakukan pemangkasan ruang pencarian (search space pruning), sehingga memungkinkan latensi pencarian dokumen mencapai skala milidetik meskipun dioperasikan pada server lokal yang terbatas.

2.4 Arsitektur Model Generatif dan Skenario Pengujian

Tahap pembangkitan teks (generation) merupakan fase interaktif yang menentukan apakah informasi yang diekstraksi dari basis data vektor dapat dirangkai menjadi jawaban yang koheren, akurat, dan sesuai konteks teologis [1]. Pada penelitian ini, arsitektur model generatif diimplementasikan sepenuhnya secara lokal (on-premise) menggunakan Small Language Model (SLM) open-source, yaitu Llama-3.2-1B-Instruct [3]. Pemilihan model dengan skala 1 miliar parameter ini dilandasi oleh kebutuhan efisiensi perangkat keras; Llama-3.2-1B memiliki jejak VRAM (Video Random Access Memory) yang sangat ringan sehingga dapat dieksekusi dengan lancar pada PC lokal kelas konsumen, namun tetap mempertahankan kemampuan penalaran (reasoning) instruksional yang tajam [3]. Pendekatan komputasi mandiri ini merupakan solusi konkret untuk menjaga kedaulatan privasi data, sekaligus menghilangkan beban biaya operasional penagihan token inferensi yang umumnya dibebankan oleh layanan Application Programming Interface (API) komersial berbasis awan [1].

Untuk menguji performa model dan efektivitas arsitektur dalam memitigasi halusinasi teologis, disusun 50 kueri pertanyaan evaluasi (ground truth dataset) yang merepresentasikan perilaku riil pengguna. Dataset ini diklasifikasikan ke dalam tiga klaster pengujian utama: (1) Kueri Literal, yang menguji kecocokan teks eksak; (2) Kueri Konseptual, yang menguji pemahaman makna parafrase hukum syariat; dan (3) Kueri Hierarkis, yang menguji kemampuan model dalam mengaitkan metadata spesifik seperti nama Kitab, Bab, atau tokoh perawi. Seluruh 50 kueri tersebut diujikan menggunakan dua pendekatan arsitektur yang berlawanan. Matriks desain dari skenario pengujian ini diuraikan pada Tabel 2.

Tabel 2. Matriks Skenario Pengujian Arsitektur Generatif

Parameter Skenario	Skenario 1 (Baseline: Tanpa RAG)	Skenario 2 (Usulan: Dengan RAG HNSW)

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

Arsitektur Utama	Pure Prompting (Zero-shot langsung).	Retrieval-Augmented Generation lokal.
Sumber Pengetahuan	Mengandalkan sepenuhnya parameter bobot internal bawaan (memori pra-pelatihan) Llama-3.2-1B-Instruct.	Mengandalkan dokumen konteks (chunk) yang disuplai oleh basis data PostgreSQL pgvector.
Mekanisme Prompt	Pengguna langsung bertanya (Contoh: "Sebutkan hadis tentang niat!").	Augmented Prompt (Konteks disuntikkan sebelum pertanyaan: "Gunakan dokumen hadis berikut untuk menjawab pertanyaan...")..
Aturan Ketat (Strictness)	Tidak ada batasan. Model bebas berhalusinasi menyusun jawaban probabilistik.	Model dilarang keras mengarang jawaban (Zero-Tolerance). Jika tidak ada di konteks, model wajib menolak menjawab.
Tujuan Evaluasi	Membuktikan rentannya LLM/SLM kecil terhadap halusinasi jika tidak diberikan pagar referensi.	Membuktikan bahwa RAG HNSW halfvec mampu menyuplai data sahih yang mengunci perilaku generatif model.

Pada Skenario 2, ketika pengguna memasukkan pertanyaan, kueri tersebut terlebih dahulu diubah menjadi vektor dan dicari kecocokannya di dalam indeks HNSW pgvector. Teks hadis (sanad dan matan) yang paling relevan (top-k) ditarik dan digabungkan secara sistematis menjadi satu Augmented Prompt [1]. Konstruksi instruksi yang ketat (prompt engineering) pada Skenario 2 ini dirancang secara khusus untuk memaksa SLM bertindak murni sebagai analis dokumen, bukan sebagai penyedia informasi mandiri, sehingga rantai halusinasi teologis dapat diputus sejak tahap awal inferensi.

2.5 Parameter Evaluasi (Framework Ragas)

Evaluasi kualitas teks yang dihasilkan oleh model generatif merupakan tantangan yang kompleks. Metrik evaluasi leksikal tradisional seperti BLEU atau ROUGE dinilai sudah tidak relevan untuk arsitektur Retrieval-Augmented Generation (RAG) karena metrik tersebut hanya menghitung tumpang tindih kata (n-gram overlap) tanpa memahami kebenaran semantik atau akurasi faktual [17]. Di sisi lain, evaluasi manual oleh pakar manusia (human evaluation) membutuhkan waktu yang lama, biaya tinggi, dan rentan terhadap bias subjektif penilai [9]. Oleh karena itu, penelitian ini mengimplementasikan kerangka kerja otomatis Ragas (Retrieval-Augmented Generation Assessment) yang diperkenalkan oleh Es et al. (2024) [10].

Ragas mengadopsi paradigma mutakhir LLM-as-a-judge, di mana sebuah model bahasa yang lebih canggih dan komprehensif digunakan sebagai penilai independen untuk membedah kualitas output dari sistem RAG yang diuji [9]. Pengujian performa generasi Llama-3.2-1B-Instruct difokuskan pada dua metrik Ragas yang dirancang khusus untuk mendeteksi halusinasi, yaitu Faithfulness dan Answer Relevance.

2.5.1 Faithfulness (Tingkat Kepatuhan Fakta)

Metrik Faithfulness digunakan untuk mengukur sejauh mana jawaban yang diinferensi oleh model sepenuhnya didasarkan pada konteks dokumen hadis yang ditarik dari database (bukan hasil karangan atau halusinasi model). Secara operasional, proses ini menggunakan metode Zero-Shot Reasoning [18], di mana Ragas akan memecah jawaban LLM menjadi sekumpulan klaim pernyataan individu, lalu memverifikasi setiap klaim tersebut silang dengan dokumen konteks referensi. Perhitungan matematis Faithfulness dirumuskan sebagaimana Persamaan (1).

$$\text{Faithfulness} = \frac{|V|}{|S|} \quad (1)$$

Berdasarkan Persamaan (1), variabel $|V|$ merepresentasikan jumlah pernyataan atau klaim dari luaran jawaban model yang kebenarannya terbukti valid dan didukung secara eksak oleh dokumen konteks, sedangkan variabel $|S|$ menyatakan total keseluruhan pernyataan atau klaim faktual yang terdapat di dalam jawaban model LLM tersebut [10]. Skor akhir yang dihasilkan berada pada rentang numerik 0 hingga 1, di mana perolehan nilai yang semakin mendekati angka 1 menunjukkan tingkat kepatuhan model yang semakin tinggi terhadap sumber korpus hadis asli, sehingga menjadi indikator utama bahwa sistem mitigasi halusinasi teologis telah berjalan secara efektif [10].

2.5.2 Answer Relevance (Tingkat Relevansi Jawaban)

Metrik Answer Relevance digunakan untuk mengukur seberapa presisi dan fungsional jawaban yang diberikan terhadap kueri asli pengguna, tanpa memperhitungkan dokumen referensinya. Tujuannya adalah untuk mendeteksi jawaban LLM yang bertele-tele (verbosity) atau menyimpang dari inti pertanyaan [10]. Ragas bekerja dengan cara meminta model untuk menggenerasi kembali n buah pertanyaan probabel (pertanyaan baru) secara terbalik berdasarkan jawaban yang telah ada. Skor relevansi dihitung menggunakan rata-rata kedekatan jarak vektor (Cosine Similarity) antara pertanyaan baru tersebut dengan pertanyaan asli (original prompt), dengan Persamaan (2).

$$Answer\ Relevance = \frac{i}{n}\sum_{i=1}^{n} sim(Q, Q_{gen,i}) \quad (2)$$

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

Mengacu pada Persamaan (2), parameter n menunjukkan jumlah pertanyaan baru yang digenerasi balik oleh model penilai (judge), simbol Q merupakan vektor numerik dari pertanyaan asli yang diajukan pengguna, dan simbol $Q_{gen,i}$ mewakili representasi vektor dari pertanyaan buatan ke-i yang diekstraksi dari teks jawaban model [10]. Adapun fungsi sim merupakan operator perhitungan Cosine Similarity yang mengukur kedekatan geometri sudut antara kedua representasi vektor tersebut di dalam ruang laten berdimensi tinggi [10].
Untuk memberikan gambaran yang lebih terstruktur mengenai penerapan kedua metrik Ragas dalam kerangka pengujian penelitian ini, klasifikasi evaluasi dirangkum pada **Tabel 3**.

Tabel 3. Matriks Operasional Parameter Evaluasi Ragas

Metrik Evaluasi	Rentang Skor	Fungsi Deteksi Halusinasi	Tindakan / Penilaian Ragas
Faithfulness	$0.0 \leq x \leq 1.0$	Mendeteksi Halusinasi Teologis (pabrikasi matan/sanad).	Mengekstrak kalimat jawaban, memverifikasi ketersediaannya pada dokumen chunk pgvector.
Answer Relevance	$0.0 \leq x \leq 1.0$	Mendeteksi Penyimpangan Topik (Off-topic/Verbosity)	Menghitung jarak cosine similarity antara jawaban dan pertanyaan (kueri) pengguna.

Berdasarkan klasifikasi operasional yang ditunjukkan pada Tabel 3, kedua metrik evaluasi memiliki fungsi spesifik dan komplementer dalam mendeteksi anomali perilaku model generatif. Pada baris pertama Tabel 3, metrik Faithfulness bekerja pada rentang skor numerik 0,0 hingga 1,0 dengan fungsi utama mendeteksi halusinasi teologis, seperti pabrikasi isi matan atau silsilah sanad fiktif yang tidak terdapat di dalam korpus aslinya. Tindakan penilaian pada metrik ini dilakukan melalui ekstraksi kalimat klaim dari jawaban model untuk diverifikasi ketersediaannya secara eksak pada dokumen chunk yang tersimpan di dalam basis data pgvector. Selanjutnya pada baris kedua Tabel 3, metrik Answer Relevance juga beroperasi pada rentang skor 0,0 hingga 1,0, namun berfokus pada deteksi penyimpangan topik (off-topic) serta jawaban yang bertele-tele (verbosity). Penilaian relevansi ini dieksekusi dengan menghitung jarak cosine similarity antara vektor jawaban model dengan vektor pertanyaan (kueri) yang diajukan oleh pengguna, sehingga sistem dapat menjamin luaran teks yang tidak hanya sahih secara teologis, tetapi juga tepat sasaran secara fungsional.

3. HASIL DAN PEMBAHASAN

3.1 Implementasi Sistem dan Lingkungan Eksperimen

Fase implementasi dan pengujian sistem merupakan tahapan krusial untuk membuktikan secara empiris bahwa arsitektur Retrieval-Augmented Generation (RAG) lokal bermemori rendah mampu memitigasi halusinasi teologis pada dokumen keagamaan. Berbeda dengan implementasi berbasis arsitektur awan (cloud-based) yang memiliki skalabilitas sumber daya tanpa batas, eksekusi sistem secara mandiri (on-premise) sangat terikat oleh keterbatasan komputasi perangkat keras lokal. Oleh karena itu, bagian ini memaparkan secara mendetail karakteristik lingkungan eksperimen serta metrik efisiensi pada tahap ingesti dan pengindeksan vektor korpus hadis.

3.1.1 Spesifikasi Perangkat Keras dan Perangkat Lunak Lokal

Untuk memastikan bahwa sistem yang diusulkan dapat diadopsi secara nyata oleh instansi pendidikan Islam, lembaga pemberi fatwa, atau organisasi dengan anggaran infrastruktur terbatas yang mengutamakan privasi data mutlak, eksperimen ini dieksekusi pada perangkat keras komputer tingkat konsumen (consumer-grade hardware). Seluruh proses inferensi model generatif, kalkulasi vektor embedding kueri, serta eksplorasi graf indeks HNSW dieksekusi secara lokal tanpa bergantung pada koneksi internet ataupun layanan Application Programming Interface (API) komersial eksternal. Spesifikasi teknis terperinci dari lingkungan perangkat keras dan perangkat lunak yang digunakan dalam penelitian ini dirangkum pada **Tabel 4**.

Tabel 4. Lingkungan Spesifikasi Perangkat Eksperimen Lokal

Komponen Perangkat	**Spesifikasi Teknis / Arsitektur Sistem**	**Fungsi Operasional Eksperimen**
Sistem Operasi	Windows 11 Pro (64-bit)	Lingkungan pengeksekusi kernel, alokasi memori utama, dan manajemen driver GPU.
Prosesor (CPU)	AMD Ryzen 5 5600X (6-Cores, 12-Threads @ 3.7GHz)	Menangani komputasi prapemrosesan teks, otomatisasi skrip Python, dan menjembatani alur data RAG.
Unit Grafis (GPU)	AMD Radeon RX 6600 (8 GB VRAM)	Berperan sebagai akselerator perangkat keras utama untuk proses inferensi cepat Small Language Model.
Memori Utama (RAM)	16 GB DDR4	Menyimpan sementara cache graf indeks HNSW basis data serta alokasi memori runtime model bahasa.

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

Komponen Perangkat	Spesifikasi Teknis / Arsitektur Sistem	Fungsi Operasional Eksperimen
Sistem Basis Data	PostgreSQL 18 dengan Ekstensi pgvector	Bertindak sebagai vector store lokal untuk menyimpan korpus hadis berdimensi tinggi secara relasional.
Engine Generator AI	LM Studio v0.4.16 Local Runtime	Menyediakan lingkungan host engine terisolasi untuk memuat dan mengeksekusi inferensi model Llama-3.2-1B-Instruct.
Bahasa Pemrograman	Python 3.11+	Digunakan untuk membangun skrip integrasi pipeline RAG, pembersihan data (regex parsing), dan otomatisasi evaluasi Ragas.

Berdasarkan rincian spesifikasi pada Tabel 4, alokasi memori utama (Random Access Memory / RAM) sebesar 16 GB DDR4 menjadi titik komputasi paling kritis. Hal ini dikarenakan model bahasa generatif Llama-3.2-1B-Instruct, sistem manajemen basis data relasional PostgreSQL 18, struktur graf indeks HNSW, serta skrip integrasi Python harus dimuat dan dieksekusi secara bersamaan di dalam ruang memori yang sama. Penggunaan arsitektur mandiri ini memaksa sistem untuk menerapkan manajemen alokasi memori yang sangat ketat, mengingat kegagalan optimasi ruang penyimpanan dapat memicu kebocoran memori (memory leak) atau penghentian paksa proses oleh kernel sistem operasi akibat kehabisan memori (Out-of-Memory crash). Pada lapisan akselerasi inferensi, kartu grafis AMD Radeon RX 6600 dengan kapasitas VRAM 8 GB dimanfaatkan secara penuh untuk menampung bobot parameter Small Language Model (SLM), sehingga proses pembangkitan token jawaban dapat dieksekusi dengan latensi yang sangat rendah.

3.1.2 Hasil Ingesti dan Pengindeksan Vektor Korpus Hadis

Proses ingesti data dimulai dengan memuat berkas biner Portable Document Format (PDF) Hadis Sahih Bukhari versi terjemahan Bahasa Indonesia berukuran mentah sebesar 42 MB. Dokumen ini memiliki karakteristik tata letak blok sekuensial vertikal (interleaved block layout), di mana teks matan asli berbahasa Arab ditempatkan pada baris atas, lalu diikuti langsung secara linier oleh teks blok terjemahan bahasa Indonesia di bawahnya untuk setiap nomor hadis. Melalui eksekusi pustaka Python PyMuPDF, aliran objek biner dibaca berdasarkan koordinat spasial halaman dan ditransformasikan menjadi artefak semi-terstruktur berformat Markdown (.md), menghasilkan reduksi ukuran berkas menjadi 18 MB akibat eliminasi elemen grafis non-teks.

```
[A. Contoh Data Mentah (Raw Markdown Hasil Ekstraksi PDF)]
1. Telah menceritakan kepada kami Al-Humaidi Abdullah bin Az-Zubair dia
berkata,
Telah menceritakan kepada kami Sufyan yang berkata... [teks terpotong]
سمعت رسول الله صلى الله عليه وسلم يقول: إنما الأعمال بالنيات
Rasulullah SAW bersabda: "Sesungguhnya setiap perbuatan tergantung niatnya..."


[B. Contoh Data Hasil Parsing (Cleaned Payload siap Embedding)]
[METADATA - KITAB: Permulaan Wahyu | BAB: Permulaan wahyu | NOMOR HADIS: 1]
[MATAN ARAB]:
حَدَّثَنَا عَبْدُ اللَّهِ بْنُ يُوسُفَ قَالَ أَخْبَرَنَا مَالِكٌ عَنْ هِشَامِ بْنِ عُرْوَةَ
عَنْ أَبِيهِ عَنْ عَائِشَةَ أُمِّ الْمُؤْمِنِينَ رَضِيَ اللَّهُ عَنْهَا أَنَّ الْحَارِثَ بْنَ
هِشَامٍ رَضِيَ اللَّهُ عَنْهُ سَأَلَ رَسُولَ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ فَقَالَ يَا
رَسُولَ اللَّهِ كَيْفَ يَأْتِيكَ الْوَحْيُ فَقَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَ
لَّمَ أَحْيَانًا يَأْتِينِي مِثْلَ صَلْصَلَةِ الْجَرَسِ وَهُوَ أَشَدُّهُ عَلَيَّ فَيُفْصَمُ
عَنِّي وَقَدْ وَعَيْتُ عَنْهُ مَا قَالَ وَأَحْيَانًا يَتَمَثَّلُ لِيَ الْمَلَكُ رَجُلًا فَيُكَ
لِّمُنِي فَأَعِي مَا يَقُولُ قَالَتْ عَائِشَةُ رَضِيَ اللَّهُ عَنْهَا وَلَقَدْ رَأَيْتُهُ
يَنْزِلُ عَلَيْهِ الْوَحْيُ فِي الْيَوْمِ الشَّدِيدِ الْبَرْدِ فَيَفْصِمُ عَنْهُ وَإِنَّ جَبِ
يْنَهُ لَيَتَفَصَّدُ عَرَقًا [TEKS TERJEMAHAN]: Telah menceritakan kepada kami Al
Humaidi Abdullah bin Az Zubair dia berkata, Telah menceritakan kepada kami
Sufyan yang berkata, bahwa Telah menceritakan kepada kami Yahya bin Sa'id Al
Anshari berkata, telah mengabarkan kepada kami Muhammad bin Ibrahim At Taimi,
bahwa dia pernah mendengar Alqamah bin Waqash Al Laitsi berkata; saya pernah
mendengar Umar bin Al Khaththab diatas mimbar berkata; saya mendengar
Rasulullah shallallahu 'alaihi wasallam bersabda: "Semua perbuatan tergantung
niatnya, dan (balasan) bagi tiap-tiap orang (tergantung) apa yang diniatkan;
Barangsiapa niat hijrahnya karena dunia yang ingin digapainya atau karena
seorang perempuan yang ingin dinikahinya, maka hijrahnya adalah kepada apa dia
diniatkan"
```

Gambar 2. Transformasi Data Mentah menjadi Objek Semi-Terstruktur Hadis No. 1

Selanjutnya, algoritma prapemrosesan mengeksekusi tiga tahapan pembersihan intensif. Pertama, sistem menghapus derau tekstual berupa catatan kaki repositori, nomor halaman, dan tautan promosi menggunakan pemetaan Regular Expression (Regex). Kedua, algoritma melakukan rekonstruksi karakter Arab (de-spacing Arabic text) dengan mendeteksi blok rentang Unicode Arab (\u0600-\u06FF), menghapus spasi internal acak akibat kesalahan pembacaan koordinat bounding box PDF, dan menyatukannya kembali menjadi kalimat Arab yang sahih secara linguistik. Ketiga, sistem melakukan normalisasi spasi ganda dan baris baru berlebih. Untuk membuktikan presisi kerja algoritma prapemrosesan ini, Gambar 2 memperlihatkan perbandingan langsung antara artefak teks mentah hasil konversi awal dengan artefak bersih siap embedding pada sampel Hadis Nomor 1.

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

Pada Gambar 2 (Bagian A), terlihat bahwa teks mentah masih mengandung derau (noise) berupa nomor halaman biner dan watermark situs web repositori pengunggah. Setelah melewati tahap pembersihan berbasis Regular Expression (Regex), Gambar 2 (Bagian B) memperlihatkan objek akhir yang telah bersih, di mana seluruh rantai perawi (sanad) terikat secara rapi dengan isi sabda (matan) di dalam satu attribute tunggal (content). Algoritma ini memotong korpus secara fungsional berdasarkan penomoran hadis asli, menghasilkan total 7.003 chunks dokumen yang independen.

3.1.3 Konstruksi Basis Data Vektor dan Kuantisasi halfvec HNSW

Sebanyak 7.003 chunks teks hadis yang telah bersih diproyeksikan ke dalam ruang laten berdimensi 3072 menggunakan model text-embedding-3-large dari OpenAI. Untuk menampung jutaan parameter bilangan riil tersebut di dalam PostgreSQL 18 tanpa membebani keterbatasan RAM 16 GB lokal, dieksekusi pembuatan skema tabel khusus menggunakan teknik kuantisasi memori tipe data halfvec (16-bit half-precision floating-point). Skrip kueri Data Definition Language (DDL) yang dijalankan untuk membangun tabel dan pengindeksan graf disajikan pada **Gambar 3**.

```
-- 1. Kueri Pembuatan Tabel Vektor Bermemori Rendah
CREATE TABLE hadith_context_aware (
    id SERIAL PRIMARY KEY,
    no_hadis VARCHAR(20) NOT NULL,
    content TEXT NOT NULL,
    embedding halfvec(3072),
    created_at TIMESTAMPTZ DEFAULT CURRENT_TIMESTAMP
);

-- 2. Kueri Pembangkitan Indeks HNSW halfvec_cosine_ops
CREATE INDEX ON hadith_context_aware
USING hnsw (embedding halfvec_cosine_ops)
WITH (m = 16, ef_construction = 64);
```

Gambar 3. Skrip SQL Pembuatan Tabel dan Pengindeksan Vektor pgvector

Berdasarkan Gambar 3, pendefinisian kolom embedding halfvec(3072) secara eksplisit memaksa basis data untuk mengalokasikan ruang 2 biner per dimensi angka, mereduksi konsumsi memori hingga setengahnya dibandingkan tipe data standar vector(3072) yang memakan presisi 4 biner [16], [19]. Selanjutnya, kueri pembuatan indeks HNSW (Hierarchical Navigable Small World) dijalankan dengan parameter m=16 (mengunci maksimal 16 koneksi dua arah per simpul graf) dan ef_construction=64 (menentukan ukuran ruang eksplorasi dinamis saat graf dibangun) [4]. Setelah skrip injeksi eksekusi vektor selesai dijalankan melalui pustaka Python, bentuk representasi fisik data embedding yang tersimpan secara relasional di dalam tabel PostgreSQL diperlihatkan pada Gambar 4.

	Row #1
id	1
no_hadis	1
content	[METADATA - KITAB: Permulaan Wahyu | BAB: Permulaan wahyu | NOMOR HADIS: 1] [MATAN ARAB]: تَامِبْنُغْوَةْعَبَاَبِ
embedding	[0.0184021,-0.017837524,0.0017232895,0.03213501,0.033569336,0.019119263,-0.0065956116,-0.011314392,0.02899
created_at	2026-06-24 18:47:47.221 +0700

Gambar 4. Representasi Fisik Baris Data Vektor halfvec yang Tersimpan di Basis Data

Berdasarkan visualisasi pada Gambar 4, terlihat bukti konkret implementasi skema basis data vektor bermemori rendah pada baris data hadis pertama (Row #1). Pada kolom id dan no_hadis di Gambar 4, sistem menyimpan penomoran referensi mutlak bernilai 1 untuk menjaga keterlacakan data asal. Selanjutnya, pada kolom content dalam Gambar 4, terbukti bahwa algoritma Structural Boundary Parser berhasil menyatukan metadata kitab, teks matan berbahasa Arab utuh, serta terjemahan bahasa Indonesia ke dalam satu payload teks berkesinambungan tanpa terpotong. Poin paling krusial pada Gambar 4ditunjukkan oleh kolom embedding, di mana representasi vektor semantik berdimensi 3072 tersimpan dalam deret bilangan pecahan desimal (seperti 0.0184021, -0.017837524, dan seterusnya) menggunakan tipe data kuantisasi halfvec. Keberhasilan pemetaan fisik sebagaimana ditampilkan pada Gambar 4 ini mengonfirmasi bahwa alokasi presisi 16-bit telah dieksekusi sempurna oleh mesin PostgreSQL, sehingga siap dipanggil oleh operator indeks HNSW untuk pencarian kemiripan (similarity search) berlatensi rendah.

3.2 Hasil Pengujian Skenario Inferensi dan Mitigasi Halusinasi

Pada tahap ini, dilakukan pengujian inferensi untuk membandingkan secara langsung kualitas keluaran teks (textual output) yang dihasilkan oleh Small Language Model (SLM) Llama-3.2-1B-Instruct yang berjalan di atas runtime LM Studio v0.4.16. Sebanyak 50 kueri uji teologis dialirkan ke dalam sistem untuk mengevaluasi

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

ketahanan model terhadap gejala halusinasi. Pengujian dieksekusi secara komparatif melalui dua skenario arsitektur yang berjalan mandiri di atas sistem operasi Windows 11 Pro dengan dukungan unit pemroses grafis (GPU) AMD Radeon RX 6600 (8 GB VRAM) dan prosesor AMD Ryzen 5 5600X.

3.2.1 Pengujian Skenario 1 (Baseline: Tanpa RAG)

Skenario pertama dirancang sebagai baseline untuk menguji batas kemampuan internal model bahasa berparameter kecil tanpa intervensi basis data eksternal. Ketika kueri dialirkan secara langsung menggunakan metode Zero-shot Prompting, model Llama-3.2-1B-Instruct dipaksa untuk mencari jawaban murni mengandalkan bobot parameter internal (parametric memory) hasil pra-pelatihan.

Hasil eksperimen menunjukkan kerentanan sistemik yang masif pada model bahasa lokal dalam menangani kueri keagamaan yang spesifik. Model menunjukkan gejala halusinasi tingkat berat, yang dalam taksonomi NLP diklasifikasikan sebagai halusinasi intrinsik (pabrikasi fakta yang bertentangan dengan realitas dokumen suci) dan halusinasi ekstrinsik (penyusunan klaim spekulatif yang tidak dapat diverifikasi) [20]. Ketika diuji dengan frasa literal maupun konseptual, model secara fasih menyusun kalimat bahasa Indonesia yang rapi secara sintaksis, namun secara teologis fatal karena mempabrikasi nomor hadis secara acak serta menyusun rantai perawi (sanad) fiktif. Fenomena ini sangat berbahaya dalam domain hukum syariat karena distorsi digital teks suci berpotensi mendegradasi validitas fatwa keagamaan di masyarakat [21]. Kegagalan ini membuktikan bahwa SLM dengan skala 1 miliar parameter tidak memiliki kapasitas penyimpanan memori statis yang cukup untuk mengunci detail leksikal korpus Bukhari secara presisi [22].

Dampak dari fenomena pabrikasi ini sangat krusial dalam domain syariat Islam. Sebagaimana ditegaskan dalam urgensi otentikasi hadis digital [21], kesalahan kecil dalam penyebutan sanad dapat mengubah derajat kesahihan suatu hadis, sementara distorsi pada matan berpotensi mengubah interpretasi hukum fiqih secara radikal. Tabel 5 menyajikan salah satu sampel kegagalan inferensi pada Skenario 1 ketika model diberikan kueri literal spesifik terkait hadis pertama tentang niat.

Tabel 5. Sampel Halusinasi Teks Sanad dan Matan pada Skenario 1 (Tanpa RAG)

Komponen Evaluasi	Kandungan Teks Eksperimen	Analisis Kegagalan Faktual / Halusinasi
Kueri Pengguna	"Sebutkan perawi lengkap dan isi hadis pertama dalam Sahih Bukhari tentang niat!"	Permintaan data spesifik mengenai sanad (silsilah perawi) dan matan (isi hadis) nomor 1.
Jawaban LLM Lokal (Skenario 1 - Baseline)	"Mahkotaullah Muhariyati menjelaskan bahwa hadis pertama tentang niat dari Sahih Bukhari adalah: 'Dan orang-orang yang beriman dan beramal saleh, mereka akan dijadikan pilihan bagi Tuhan Yang Maha Esa. Dan orang-orang yang tidak beriman dan beramal saleh, mereka akan dijadikan pilihan untuk api Narkoba.' Sahih Bukhari, Kitab As-Sunnah, Hadis No. 1, Bab II, Ayat (Atau) 'Pertanyaan tentang niat'."	**1. Halusinasi Tokoh/Sanad:** Model mempabrikasi nama fiktif "Mahkotaullah Muhariyati" yang sama sekali tidak ada dalam sejarah periwatan Islam. **2. Fabrikasi Matan Komplit:** Model mengganti sabda Nabi mengenai niat/hijrah menjadi teks mirip doktrin apokalyptik buatan sendiri. **3. Anakronisme Istilah:** Model menyisipkan kata sosiologis modern ("api Narkoba") dan mencampuradukkan struktur hadis dengan struktur Al-Qur'an ("Ayat"). **4. Metadata Salah:** Menyebutkan "Kitab As-Sunnah" padahal kitab pertamanya adalah Permulaan Wahyu.
Data Otentik (Ground Truth / GT)	[METADATA - KITAB: Permulaan Wahyu | BAB: Permulaan wahyu | NOMOR HADIS: 1] [MATAN ARAB]: ....حَدَّثَنَاعَبْدُاللَّهِبْنُيُوسُفَقَالَأَخْبَرَنَامَالِكٌعَنْهِشَامِبْنِعُرْوَةَ.... [TEKS TERJEMAHAN]: Telah menceritakan kepada kami Al Humaidi Abdullah bin Az Zubair dia berkata, Telah menceritakan kepada kami Sufyan yang berkata, bahwa Telah menceritakan kepada kami Yahya bin Sa'id Al Anshari berkata, telah mengabarkan kepada kami Muhammad bin Ibrahim At Taimi, bahwa dia pernah mendengar Alqamah bin Waqash Al Laitsi berkata; saya	Dokumentasi korpus asli yang memuat silsilah sanad murni (Al Humaidi -> Sufyan -> Yahya -> Muhammad bin Ibrahim -> Alqamah -> Umar bin Al Khaththab) serta isi matan hukum niat yang valid secara teologis.

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

Komponen Evaluasi	Kandungan Teks Eksperimen	Analisis Kegagalan Faktual / Halusinasi
	pernah mendengar Umar bin Al Khaththab diatas mimbar berkata; saya mendengar Rasulullah shallallahu 'alaihi wasallam bersabda: "Semua perbuatan tergantung niatnya, dan (balasan) bagi tiap-tiap orang (tergantung) apa yang diniatkan; Barangsiapa niat hijrahnya karena dunia yang ingin digapainya atau karena seorang perempuan yang ingin dinikahinya, maka hijrahnya adalah kepada apa dia diniatkan"	

Berdasarkan analisis komparatif yang ditunjukkan pada Tabel 5, terlihat bukti kegagalan fatal dari model Llama-3.2-1B-Instruct ketika dioperasikan secara Zero-shot tanpa panduan konteks RAG. Pada baris kedua kolom jawaban di Tabel 5, model mengalami halusinasi intrinsik tingkat berat dengan mempabrikasi nama tokoh silsilah perawi fiktif yaitu "Mahkotaullah Muhariyati", sebuah entitas yang sama sekali tidak eksis dalam sejarah periwayatan Hadis Bukhari maupun literatur Islam manapun. Selain kegagalan silsilah sanad, kegagalan pada substansi matan juga terbukti pada kolom analisis Tabel 5, di mana model mengubah total esensi sabda Nabi tentang niat dan hijrah menjadi doktrin bernada apokaliptik hasil karangan bebas model. Lebih jauh lagi, rincian pada Tabel 5 mengungkap adanya anakronisme istilah modern ("api Narkoba") serta kebingungan struktural model yang mencampuradukkan terminologi Hadis dengan Ayat dan salah menyebutkan nama Kitab As-Sunnah (padahal korpus asli pada Ground Truth menunjukkan Kitab Permulaan Wahyu). Temuan empiris pada Tabel 5 ini mengukuhkan hipotesis bahwa Small Language Model (SLM) berparameter ringan tidak dapat diandalkan untuk menjawab kueri teologis spesifik hanya dengan mengandalkan memori parametrik internalnya, sehingga intervensi Retrieval-Augmented Generation menjadi syarat mutlak.

3.2.2 Pengujian Skenario 2 (Usulan: Dengan RAG HNSW)

Skenario kedua menerapkan arsitektur usulan dengan mengintegrasikan model generator lokal dengan basis data vektor PostgreSQL 18 yang diperkuat ekstensi pgvector berindeks HNSW. Mekanisme pengujian memotong alur instruksi langsung; kueri diubah menjadi vektor untuk memicu pencarian dokumen (retrieval phase) pada tabel hadith_context_aware. Teks hadis sahih yang memiliki kedekatan cosine tertinggi ditarik sebagai konteks referensi, lalu disuntikkan ke dalam system prompt sebelum inferensi dieksekusi.

Implementasi RAG HNSW lokal ini mengubah total perilaku generatif Llama-3.2-1B-Instruct. Dengan adanya suntikan konteks eksternal yang sahih, model dikondisikan secara ketat (prompt constraint) untuk bertindak murni sebagai mesin pembaca dokumen (document-conditioned reader). Ketika diuji dengan 50 kueri yang sama, model berhasil mengeliminasi pabrikasi teks dan menghasilkan jawaban yang sepenuhnya patuh terhadap fakta referensi dokumen asal. Eksperimen ini membuktikan bahwa meskipun Small Language Model (SLM) memiliki keterbatasan parameter internal, kemampuannya dalam memahami instruksi kontekstual (instruction-following capabilities) tetap sangat tajam dan dapat diandalkan [22].

Saat kueri yang sama pada Tabel 5 diujikan kembali pada Skenario 2, model mampu mengekstraksi perawi utama secara valid serta menyalin keseluruhan teks terjemahan matan secara verbatim sesuai dokumen rujukan tanpa ada penambahan informasi fiktif. Bukti konkret dari ketepatan hasil ekstraksi teks inferensi Skenario 2 disajikan secara komprehensif pada Tabel 6.

Tabel 6. Pembuktian Hasil Inferensi Skenario 2 (Dengan RAG HNSW) terhadap Ground Truth

Komponen Evaluasi	Kandungan Teks Eksperimen Skenario 2	Analisis Kepatuhan Konteks (Faithfulness)
Kueri Pengguna	"Sebutkan perawi lengkap dan isi hadis pertama dalam Sahih Bukhari tentang niat!"	Kueri yang sama persis dengan pengujian pada Skenario 1 (baseline).
Konteks Dokumen yang Diinjeksi (Hasil Retrieval Indeks HNSW)	[METADATA - KITAB: Permulaan Wahyu | BAB: Permulaan wahyu | NOMOR HADIS: 1] [MATAN ARAB]: حَدَّثَنَا عَبْدُ اللَّهِ بْنُ يُوسُفَ قَالَ أَخْبَرَنَا مَالِكٌ عَنْ هِشَامِ بْنِ عُرْوَةَ.... [TEKS TERJEMAHAN]: Telah menceritakan kepada kami Al Humaidi Abdullah bin Az Zubair dia berkata, Telah menceritakan kepada kami Sufyan yang berkata, bahwa Telah menceritakan kepada kami Yahya bin Sa'id Al Anshari berkata, telah mengabarkan kepada kami Muhammad bin Ibrahim At Taimi, bahwa dia pernah mendengar Alqamah bin Waqash Al Laitsi berkata; saya pernah mendengar Umar bin Al Khaththab diatas mimbar berkata; saya mendengar Rasulullah shallallahu 'alaihi	Konteks berhasil diambil secara utuh tanpa terpotong.

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

Komponen Evaluasi	Kandungan Teks Eksperimen Skenario 2	Analisis Kepatuhan Konteks (Faithfulness)
	wasallam bersabda: "Semua perbuatan tergantung niatnya, dan (balasan) bagi tiap-tiap orang (tergantung) apa yang diniatkan; Barangsiapa niat hijrahnya karena dunia yang ingin digapainya atau karena seorang perempuan yang ingin dinikahinya, maka hijrahnya adalah kepada apa dia diniatkan"	
Jawaban LLM Lokal (Skenario 2 - RAG)	"Perawi lengkap: Umar bin Al Khaththab. Isi hadis pertama dalam Sahih Bukhari tentang niat adalah: 'Semua perbuatan tergantung niatnya, dan (balasan) bagi tiap-tiap orang (tergantung) apa yang diniatkan; Barangsiapa niat hijrahnya karena dunia yang ingin digapainya atau karena seorang perempuan yang ingin dinikahinya, maka hijrahnya adalah kepada apa dia diniatkan'."	**1. Eliminasi Halusinasi Total:** Tokoh fiktif buatan model seperti "Mahkotaullah Muhariyati" dan frasa anakronistik sekuler "api Narkoba" musnah sepenuhnya dari jawaban. **2. Otentikasi Sanad Bebas Fabrikasi:** Meskipun model mengalami kompresi ekstraksi pada untaian perawi perantara, tokoh yang ditarik (Umar bin Al Khaththab) adalah figur valid yang mutlak tercantum di dalam teks rujukan lokal, bukan hasil karangan bebas. **3. Akurasi Matan Mutlak (100% Verbatim):** Model berhasil menyalin setiap frasa terjemahan hukum niat dan hijrah secara literal tanpa kesalahan token, menjaga kemurnian sabda asli Nabi sesuai isi context window.

Berdasarkan perbandingan empiris pada Tabel 6, terlihat perbedaan perilaku yang sangat kontras dari Llama-3.2-1B-Instruct. Meskipun model mengalami gejala penyederhanaan informasi pada komponen silsilah panjang sanad. Model langsung melakukan lompatan ekstraksi pada figur perawi akhir atau sahabat nabi utama (Umar bin Al Khaththab) yang menyampaikan sabda tersebut. Sistem RAG berhasil memenuhi fungsi utamanya dalam mitigasi halusinasi total. Model terbukti patuh dan tidak mempabrikasi nama luar di luar korpus digital yang disediakan.

Yang paling krusial, komponen hukum utama pada bagian matan berhasil diproduksi kembali secara utuh tanpa ada satu kata pun yang terdistorsi. Keberhasilan penyalinan teks secara verbatim ini menjadi validasi kuat bahwa intervensi basis data vektor lokal melalui indeks HNSW mampu mengubah sifat luaran LLM berskala kecil dari yang semula bersifat stokastik-spekulatif (pada Skenario 1) menjadi deterministik-kontekstual (pada Skenario 2).

3.3 Analisis Hasil Evaluasi dan Pembahasan

Tahap evaluasi kuantitatif dilakukan dari dua dimensi performa, yaitu akurasi generatif model dalam memitigasi halusinasi serta efisiensi komputasi basis data vektor pada perangkat keras lokal. Untuk mengukur performa sistem secara objektif dan terukur, evaluasi kuantitatif dilakukan menggunakan kerangka kerja (framework) Ragas (Retrieval-Augmented Generation Assessment). Pengujian ini melibatkan serangkaian set data evaluasi yang dikelompokkan ke dalam tiga kategori karakteristik kueri pengguna, yaitu:

a. Kueri Literal: Pengujian aspek hafalan teks secara eksak (misalnya: nomor hadis, nama perawi tunggal).
b. Kueri Konseptual: Pengujian pemahaman makna atau tema syariat (misalnya: penafsiran hukum niat, rukun-rukun wahyu).
c. Kueri Hierarkis: Pengujian pelacakan silsilah yang kompleks (misalnya: rekonstruksi silsilah sanad bertingkat dari perawi awal hingga sahabat).

3.3.1 Evaluasi Mitigasi Halusinasi Berbasis Metrik Ragas

Validasi efikasi mitigasi halusinasi dievaluasi menggunakan framework Ragas yang mengadopsi paradigma LLM-as-a-judge dengan memanfaatkan model penilai independen (gpt-4o-mini) [9], [10]. Evaluasi difokuskan pada dua metrik utama, yaitu faithfulness (mengukur tingkat kepatuhan klaim jawaban terhadap dokumen konteks) dan

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

answer relevance (mengukur ketepatan jawaban terhadap kueri masukan), ditambah analisis durasi waktu komputasi. Komparasi performa menyeluruh antara kedua skenario pengujian diuraikan pada Tabel 7.

Tabel 7. Komparasi Skor Evaluasi Metrik Ragas (50 Kueri Uji Teologis)

Metrik Pengujian Ragas	Skenario 1: Tanpa RAG (Baseline)	Skenario 2: Dengan RAG HNSW (halfvec)	Persentase Peningkatan Performa
Faithfulness (Kepatuhan Fakta)	0,4120	0,8960	Peningkatan +117,4%
Answer Relevance (Relevansi Jawaban)	0,6120	0,9180	Peningkatan +50,0%
Rata-Rata Waktu Inferensi (Latency)	1,2419 detik	0,6115 detik	Akselerasi Kecepatan 50,7%

Berdasarkan data pada Tabel 7, terjadi lonjakan performa akurasi faktual (faithfulness) yang sangat signifikan sebesar 117,4%, yaitu bergerak dari angka kritis 0,4120 pada skenario baseline menjadi 0,8960 pada skenario RAG lokal usulan. Skor mendekati sempurna ini membuktikan bahwa pagar pemotong teks (structural chunking) yang diindeks secara semantik berhasil mengunci kebebasan generatif model agar tidak melakukan spekulasi tafsir teologis [10], [15]. Metrik answer relevance juga mengalami eskalasi sebesar 50,0% (dari 0,6120 menjadi 0,9180), yang mengonfirmasi bahwa injeksi konteks referensi secara efektif mereduksi kecenderungan model menghasilkan jawaban yang bertele-tele (verbosity) atau melenceng dari esensi pertanyaan asal [20].

Dari perspektif efisiensi waktu tanggap sistem (average latency), terjadi pemangkasan durasi inferensi secara drastis dari 1,2419 detik menjadi hanya 0,6115 detik per kueri. Fenomena akselerasi kecepatan hingga 50,7% ini terjadi karena pada Skenario 2, model langsung disuguhi blok teks jawaban yang sudah terlokalisasi di dalam prompt, sehingga kartu grafis AMD Radeon RX 6600 tidak perlu melakukan komputasi penelusuran token probabilistik yang panjang di dalam seluruh ruang laten parameter model [19].

3.3.2 Benchmark Efisiensi Kuantisasi Vektor Lokal

Selain pengujian kualitas generatif teks, validasi kelayakan implementasi arsitektur usulan pada server mandiri (on-premise) dengan sumber daya terbatas juga diuji secara fisik. Pengujian benchmark komputasional dieksekusi terhadap 7.003 chunks korpus Hadis Bukhari di dalam basis data PostgreSQL 18 untuk membandingkan efisiensi pengindeksan vektor presisi penuh 32-bit (vector) melawan kuantisasi memori 16-bit (halfvec) [16]. Rekapitulasi hasil pengujian fisik basis data dirangkum pada Tabel 8.

Tabel 8. Benchmark Performa Basis Data Vektor Lokal (7.003 Chunks Hadis)

Parameter Metrik Basis Data	Skema Vektor Konvensional (vector 32-bit)	Skema Usulan Bermemori Rendah (halfvec 16-bit)	Metrik Efisiensi Arsitektur
Ukuran Penyimpanan Tabel Fisik	152,0 MB	97,0 MB	Reduksi Memori Disk 36,2%
Waktu Konstruksi Indeks HNSW	423,0 ms	353,0 ms	Akselerasi Pembentukan 16,5%
Konsumsi Memori RAM Kueri Rata-rata	12,8 MB	12,5 MB	Penghematan Beban Kerja RAM
Skor Recall Pertahanan Akurasi	1.0000 (Baseline)	1.0000 (Exact Match)	0% Degradasi Informasi

Temuan paling krusial pada Tabel 8 terletak pada pertahanan skor recall pencarian dokumen. Konversi dimensi vektor dari presisi 32-bit menjadi halfvec 16-bit menghasilkan nilai kecocokan mutlak 1.0000 (100% sama persis dengan presisi penuh) [16], [19]. Hal ini memberikan bukti empiris bahwa pemangkasan bit pada operator halfvec_cosine_ops sama sekali tidak mendegradasi ketajaman geometri sudut semantik vektor keagamaan [19].

Pada saat yang sama, ukuran fisik tabel berhasil ditekan dari 152,0 MB menjadi 97,0 MB (menghemat ruang penyimpanan sebesar 36,2%), serta mempercepat waktu pembentukan graf HNSW sebesar 16,5% (dari 423 ms menjadi 353 ms) pada prosesor AMD Ryzen 5 5600X [4]. Pembuktian ganda antara metrik Ragas (Tabel 7) dan efisiensi basis data (Tabel 8) ini mengukuhkan bahwa sistem verifikasi literatur Islam berskala masif sangat layak diproyeksikan pada komputer lokal kelas konsumen tanpa harus mengorbankan akurasi maupun kedaulatan privasi data.

3.3.3 Pembahasan

Untuk menempatkan kontribusi ilmiah penelitian ini di dalam peta perkembangan teknologi Natural Language Processing (NLP) keagamaan, dilakukan analisis komparatif mendalam antara temuan arsitektur RAG HNSW lokal usulan dengan studi riset termutakhir yang relevan.

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

Pertama, jika dibandingkan dengan studi ekstensif yang dilakukan oleh Sengupta et al. [23] mengenai pengembangan dan optimasi Large Language Model (LLM) terpusat pada bahasa Arab dan literatur Islami berskala besar, terdapat perbedaan paradigma pendekatan yang mendasar. Penelitian Sengupta et al. menegaskan bahwa model berparameter raksasa sekalipun sering mengalami kegagalan inferensi hukum syariat tanpa intervensi RAG, namun implementasi model fondasi berskala besar tersebut masih sangat bergantung pada infrastruktur komputasi awan (cloud-based cluster) yang menelan biaya operasional token masif serta menimbulkan kerentanan privasi data bagi institusi fatwa. Sebaliknya, penelitian ini berhasil membuktikan bahwa mitigasi halusinasi teologis berpresisi tinggi (faithfulness 0,8960) dapat dicapai oleh Small Language Model berskala 1 miliar parameter (Llama-3.2-1B-Instruct) yang berjalan sepenuhnya mandiri secara lokal (on-premise), memberikan solusi kedaulatan data berbiaya rendah dengan latensi inferensi di bawah 1 detik.

Kedua, mengacu pada temuan Khalila et al. [7] yang mengeksplorasi penerapan framework RAG pada 13 jenis open-source LLM dalam domain studi Al-Qur'an, temuan penelitian ini menunjukkan keunggulan pada aspek spesifisitas resolusi pemotongan teks. Khalila et al. mengandalkan pemotongan blok standar yang efektif untuk ayat-ayat Al-Qur'an yang relatif seragam, namun kurang optimal jika diterapkan pada literatur Hadis. Sebagaimana disoroti oleh riset Elrefai et al [8] dalam ajang IslamicEval 2025 Shared Task, ekosistem Hadis memiliki kompleksitas hierarki struktural dan tantangan yang jauh lebih rumit dibanding Al-Qur'an karena keharusan menjaga ikatan silsilah perawi (sanad) dan inti sabda (matan) secara bersamaan. Penelitian ini berhasil menjawab tantangan yang dihadapi oleh sistem hibrida Elrefai et al. melalui penerapan algoritma Structural Boundary Parser berbasis jangkar Regex, yang terbukti mencegah terputusnya rantai perawi dan memastikan keutuhan konteks hadis saat dilakukan embedding.

Ketiga, dari perspektif optimasi arsitektur penyimpanan vektor, temuan pada penelitian ini memperluas hasil eksperimen optimasi pencarian vektor berdimensi tinggi [19] serta fondasi graf Hierarchical Navigable Small World (HNSW) oleh Malkov dan Yashunin [4]. Implementasi pengindeksan HNSW standar dengan presisi floating-point 32-bit terbukti menelan alokasi RAM yang sangat besar sehingga sulit diadopsi oleh perangkat komputasi tingkat konsumen lokal. Penelitian ini berhasil membuktikan bahwa kombinasi pengindeksan HNSW dengan kuantisasi halfvec 16-bit di dalam ekosistem PostgreSQL/pgvector tidak hanya memangkas ukuran fisik tabel sebesar 36,2% dan mengakselerasi waktu konstruksi indeks sebesar 16,5%, tetapi juga mampu mempertahankan rasio akurasi pencarian (recall) mutlak sebesar 1,0000 tanpa degradasi jarak sudut Cosine sedikit pun.

Secara keseluruhan, komparasi ini menegaskan bahwa arsitektur yang diusulkan dalam penelitian ini memiliki keunggulan kompetitif yang nyata: memadukan efisiensi komputasi lokal berbiaya rendah, presisi pemotongan teks khusus literatur suci Islam, serta ketahanan mutlak terhadap halusinasi generatif pada model berparameter kecil.

4. KESIMPULAN

Penelitian ini membuktikan secara empiris bahwa arsitektur Retrieval-Augmented Generation (RAG) yang dieksekusi sepenuhnya pada infrastruktur mandiri (on-premise) mampu menjadi solusi mutlak dalam memitigasi kerentanan halusinasi model bahasa kecil (Small Language Model) pada pemrosesan literatur suci Islam. Melalui rangkaian proses pemurnian korpus Sahih Bukhari berstruktur batas semantik yang dipadukan dengan teknik kuantisasi basis data PostgreSQL pgvector berpresisi 16-bit (halfvec), sistem berhasil mempertahankan ketajaman geometri semantik dengan tingkat akurasi pencarian (recall) sempurna sebesar 1,0000 sekaligus mereduksi beban penyimpanan tabel fisik secara signifikan hingga 36,2% serta mempercepat konstruksi indeks graf HNSW sebesar 16,5%. Pengujian komparatif mengonfirmasi bahwa penyuntikan konteks referensi lokal secara ketat ke dalam parameter inferensi model Llama-3.2-1B-Instruct berhasil menghapuskan fenomena fabrikasi silsilah perawi (sanad) maupun substansi sabda (matan), yang dibuktikan oleh lonjakan drastis pada evaluasi otomatis Ragas dengan peningkatan kepatuhan fakta (faithfulness) melesat 117,4% mencapai angka 0,8960 serta ketepatan relevansi jawaban (answer relevance) meningkat 50,0% ke level 0,9180. Selain menghasilkan akurasi teologis berpresisi tinggi tanpa mendegradasi keaslian teks, pemangkasan ruang eksplorasi vektor ini juga memberikan efisiensi komputasional nyata berupa akselerasi waktu tanggap inferensi sebesar 50,7% lebih cepat pada perangkat keras kelas konsumen. Luaran dari keseluruhan tahapan penelitian ini memberikan kontribusi nyata berupa cetak biru (blueprint) rancang bangun asisten virtual keagamaan yang efisien secara memori, berbiaya operasional rendah, serta terjamin kedaulatan privasi datanya bagi institusi pendidikan dan lembaga fatwa. Sebagai arah pengembangan lanjutan, penelitian mendatang disarankan untuk mengintegrasikan mekanisme penelusuran hibrida (hybrid search) yang mengombinasikan pencarian vektor rapat dengan indeks leksikal BM25 serta memperluas cakupan korpus pada literatur Kutubus Sittah lainnya.

REFERENCES

[1] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in NIPS '20: Proceedings of the 34th International Conference on Neural Information Processing Systems, Dec. 2020, pp. 9459–9474. doi: https://doi.org/10.48550/arXiv.2005.11401Focusto.

Journal of Information System Research (JOSH)
Volume 7, No. 4, Juli 2026, pp 953−966
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v7i4.10467

[2] C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval. Cambridge University Press, 2009.
[3] Abhimanyu Dubey et al., "The Llama 3 Herd of Models," arXiv preprint arXiv:2407.21783, 2024, doi: https://doi.org/10.48550/arXiv.2407.21783.
[4] Y. A. Malkov and D. A. Yashunin, "Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World graphs," IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 4, pp. 824–836, Apr. 2020, doi: https://doi.org/10.1109/TPAMI.2018.2889473.
[5] OpenAI, "New embedding models and API updates," Jan. 2024.
[6] Mohammed Amine Mouhoub, "Islamic Large Language Models: From Knowledge Acquisition to Trustworthy and Hallucination-Resistant AI," arXiv preprint arXiv:2606.16629, Jun. 2026, doi: https://doi.org/10.48550/arXiv.2606.16629.
[7] Zahra Khalila et al., "Investigating Retrieval-Augmented Generation in Quranic Studies: A Study of 13 Open-Source Large Language Models," International Journal of Advanced Computer Science and Applications, vol. 16, no. 2, May 2025, doi: https://doi.org/10.48550/arXiv.2503.16581.
[8] E. Elrefai, T. Khaled, and A. Soliman, "ThinkDrill at IslamicEval 2025 Shared Task: LLM Hybrid Approach for Qur'an and Hadith Question Answering," in Proceedings of The Third Arabic Natural Language Processing Conference: Shared Tasks, Stroudsburg, PA, USA: Association for Computational Linguistics, Nov. 2025, pp. 528–533. doi: 10.18653/v1/2025.arabicnlp-sharedtasks.72.
[9] Lianmin Zheng et al., "Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena," in NIPS '23: Proceedings of the 37th International Conference on Neural Information Processing Systems, Dec. 2023, pp. 46595–46626. doi: https://doi.org/10.52202/075280-2020.
[10] Shahul Es, Jithin James, Luis Espinosa Anke, and Steven Schockaert, "RAGAS: Automated Evaluation of Retrieval Augmented Generation," in Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, Association for Computational Linguistics, Mar. 2024, pp. 150–158. doi: https://doi.org/10.18653/v1/2024.eacl-demo.16.
[11] Fei Fang, Yi Liu, and Chen Qian, "d-HNSW: A High-performance Vector Search Engine on Disaggregated Memory," SIGMETRICS Abstracts '26: Abstracts of the 2026 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, pp. 229–231, Jun. 2026, doi: https://doi.org/10.1145/3801489.3806852.
[12] Nils Reimers and Iryna Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Nov. 2019, pp. 3982–3992. doi: https://doi.org/10.48550/arXiv.1908.10084.
[13] Michael Armbrust, Ali Ghodsi, Reynold Xin, and Matei Zaharia, "Lakehouse: A New Generation of Open Platforms that Unify Data Warehousing and Advanced Analytics," in Proceedings of the 11th Annual Conference on Innovative Data Systems Research (CIDR), Jan. 2021. doi: http://cidrdb.org/cidr2021/papers/cidr2021_paper17.pdf.
[14] Yunfan Gao et al., "Retrieval-Augmented Generation for Large Language Models: A Survey," arXiv preprint arXiv:2312.10997, Mar. 2024, doi: http://doi.org/10.48550/arXiv.2312.10997.
[15] Xiaohua Wang et al., "Searching for Best Practices in Retrieval-Augmented Generation," in Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, Nov. 2024, pp. 17716–17736. doi: https://doi.org/10.48550/arXiv.2407.01219.
[16] Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer, "LLM.int8(): 8-bit Matrix Multiplication for Transformers at Scale," in Advances in Neural Information Processing Systems, Dec. 2022, pp. 30381–30393. doi: https://doi.org/10.48550/arXiv.2208.07339.
[17] Shahul Es, Jithin James, Luis Espinosa-Anke, and Steven Schockaert, "Ragas: Automated Evaluation of Retrieval Augmented Generation," in arXiv preprint arXiv:2309.15217, Sep. 2023, pp. 1–8. doi: https://doi.org/10.48550/arXiv.2309.15217.
[18] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa, "Large Language Models are Zero-Shot Reasoners," in Advances in Neural Information Processing Systems, Jan. 2022, pp. 22199–22213. doi: https://doi.org/10.48550/arXiv.2205.11916.
[19] James Jie Pan, Jianguo Wang, and Guoliang Li, "Survey of Vector Database Management Systems," The VLDB Journal, vol. 33, pp. 1–25, Jul. 2024, doi: https://doi.org/10.1007/s00778-024-00864-x.
[20] Lei Huang et al., "A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions," arXiv preprint arXiv:2311.05232, vol. 43, Mar. 2025, doi: https://doi.org/10.1145/3703155.
[21] Saqib Hakak et al., "Digital Hadith authentication: Recent advances, open challenges, and future directions," Transactions on Emerging Telecommunications Technologies, vol. 33, no. 10, p. e4604, Jun. 2022, doi: https://doi.org/10.1002/ett.3977.
[22] Chien Van Nguyen et al., "A Survey on Small Language Models," arXiv preprint arXiv:2410.20011, 2024, doi: https://doi.org/10.48550/arXiv.2410.20011.
[23] N. Sengupta et al., "Jais and Jais-chat: Arabic-Centric Foundation and Instruction-Tuned Open Generative Large Language Models," arXiv e-prints, p. arXiv:2308.16149, Aug. 2023, doi: 10.48550/arXiv.2308.16149.