

# Implementasi Sistem Wawancara Kerja Berbasis *Retrieval-Augmented Generation* untuk Otomatisasi Pertanyaan dan Jawaban

Ahmad Zainul Mufid, Euis Oktavianti, Dewi Yanti Liliana, Rizki Elisa Nalawati, Chairul Fikri

Teknik Informatika, Teknik Informatika dan Komputer, Politeknik Negeri Jakarta Depok, Jawa Barat  
[ahmad.zainul.mufid.tik22@mhsw.pnj.ac.id](mailto:ahmad.zainul.mufid.tik22@mhsw.pnj.ac.id), [euis.oktavianti@tik.pnj.ac.id](mailto:euis.oktavianti@tik.pnj.ac.id), [dewiyanti.liliana@tik.pnj.ac.id](mailto:dewiyanti.liliana@tik.pnj.ac.id),  
[rizkielisa@tik.pnj.ac.id](mailto:rizkielisa@tik.pnj.ac.id), [chairul.fikri@lecturer.pnj.ac.id](mailto:chairul.fikri@lecturer.pnj.ac.id)

**Abstract** - A job interview is a crucial stage for job seekers, especially college students and fresh graduates, requiring thorough preparation and realistic simulation practice. Since conventional simulations are often limited to static questions, this research aims to implement a voice-based automated interview simulation system utilizing artificial intelligence. This system integrates a Large Language Model (LLM), the Retrieval-Augmented Generation (RAG) method, and a ChromaDB vector database. Voice interaction is supported by Speech Recognition technology (Whisper API) to transcribe answers and Text-to-Speech (Edge TTS) to generate the AI interviewer's voice. Answer scoring is mathematically calculated using TF-IDF and Cosine Similarity to measure the similarity between the candidate's response and the reference answer, while the LLM acts as a reasoning engine to compose feedback narratives and evaluation reports. The system's performance was evaluated through four stages: intelligence evaluation using RAGAs demonstrated high context relevance, human evaluation by two experts indicated that 10 out of 15 feedback samples were considered valid, Black Box functional testing confirmed that all features operated correctly, and the System Usability Scale (SUS) achieved an average score of 91 (Grade A, Excellent) with a Net Promoter Score (NPS) of 90.

**Keywords:** Large Language Model, Retrieval-Augmented Generation, Speech Recognition, Cosine Similarity, Automatic Question and Answer.

**Abstract** - Wawancara kerja merupakan tahapan krusial bagi pencari kerja, khususnya kalangan mahasiswa dan fresh graduate, yang membutuhkan persiapan matang serta latihan simulasi yang realistis. Karena simulasi konvensional sering terbatas pada pertanyaan statis, penelitian ini bertujuan mengimplementasikan sistem simulasi wawancara otomatis berbasis suara memanfaatkan kecerdasan buatan. Sistem ini memadukan Large Language Model (LLM), metode Retrieval-Augmented Generation (RAG), dan basis data vektor ChromaDB. Interaksi suara didukung teknologi Speech recognition (Whisper API) untuk mentranskripsi jawaban dan Text-to-Speech (Edge TTS) untuk menghasilkan suara pewawancara AI. Penilaian jawaban dihitung secara matematis menggunakan TF-IDF dan Cosine Similarity untuk mengukur kemiripan jawaban kandidat dengan jawaban referensi, sementara LLM berperan sebagai reasoning engine penyusun narasi umpan balik dan laporan evaluasi. Performa sistem diuji melalui empat tahap: pengujian kecerdasan menggunakan RAGAs menunjukkan relevansi konteks yang tinggi, human evaluation oleh dua orang pakar menunjukkan 10 dari 15 sampel umpan balik dinilai valid, pengujian fungsionalitas Black Box membuktikan seluruh fitur berjalan dengan baik, serta evaluasi System Usability Scale (SUS) memperoleh skor rata-rata 91 (Grade A, Excellent) dan Net Promoter Score (NPS) mencapai skor 90.

**Kata kunci:** Large Language Model, Retrieval-Augmented Generation, Speech Recognition, Cosine Similarity, Otomatis Pertanyaan dan Jawaban.

## I. PENDAHULUAN

Tingginya persaingan dalam proses rekrutmen kerja berdampak langsung pada rendahnya peluang pelamar untuk lolos tahap wawancara. Fakta di lapangan menunjukkan bahwa wawancara merupakan fase seleksi paling ketat, di mana hanya sekitar 20% pelamar yang berhasil dipanggil wawancara, dan dari jumlah tersebut, hanya sebagian kecil yang akhirnya menerima

tawaran kerja [2]. Banyak pelamar gugur di tahap ini bukan sekadar karena kurangnya kualifikasi teknis, melainkan akibat kurangnya kesiapan mental dan ketidakmampuan mengomunikasikan kompetensi mereka dengan baik di bawah tekanan, kondisi ini diperparah oleh minimnya akses terhadap sarana simulasi wawancara yang realistis, sehingga pelamar tidak memiliki ekspektasi yang jelas mengenai jenis pertanyaan yang akan mereka hadapi di dunia nyata [6].

Permasalahan ini semakin krusial pada bidang pekerjaan yang menuntut spesialisasi tinggi seperti sektor Teknologi Informasi, di mana proses wawancara umumnya melibatkan pertanyaan teknis yang mendalam, dinamis, dan spesifik [5]. Tanpa platform latihan yang mampu mensimulasikan kondisi tersebut secara akurat, pelamar kesulitan mengevaluasi kualitas jawaban mereka sendiri secara objektif.

Berbagai penelitian terdahulu telah mengkaji pemanfaatan kecerdasan buatan untuk mengotomatisasi proses wawancara guna mengatasi masalah kesiapan pelamar, termasuk pendekatan berbasis *Large Language Model* (LLM) untuk menjalankan simulasi seleksi teknis dan evaluasi kandidat secara otomatis [10]. Meskipun LLM mampu memberikan efisiensi operasional dan kemampuan pemahaman bahasa yang tinggi [13], penerapannya secara mandiri tanpa konteks eksternal memiliki kelemahan mendasar: model generatif murni rentan mengalami halusinasi, sehingga cenderung menghasilkan pertanyaan yang terdengar meyakinkan namun tidak relevan dengan data atau profil asli pelamar [9].

Beberapa pendekatan umum digunakan untuk mengurangi halusinasi pada model bahasa, namun masing-masing memiliki keterbatasan untuk kasus rekrutmen. *Prompt Engineering* dibatasi oleh kapasitas *context window*, sehingga analisis menjadi tidak konsisten pada dokumen resume yang panjang atau kompleks [16]. *Fine-Tuning* kurang efektif untuk kasus rekrutmen yang dinamis karena menuntut pelatihan ulang setiap kali terjadi perubahan data atau tren pelamar, dengan biaya komputasi yang besar [4].

Sebagai solusi atas keterbatasan tersebut, metode Retrieval-Augmented Generation (RAG) diusulkan karena mampu menyediakan konteks dinamis tanpa perlu melatih ulang model, dengan menggabungkan kemampuan generatif LLM dan mekanisme pencarian data dari basis pengetahuan eksternal [7]. Melalui pendekatan ini, sistem dapat mengekstrak topik keahlian kandidat secara langsung dari sesi pengenalan untuk dijadikan kueri pencarian kriteria pekerjaan, sehingga pertanyaan wawancara yang dihasilkan menjadi lebih terpersonalisasi, akurat, dan minim halusinasi dibandingkan penggunaan LLM standar [15].

## II. METODOLOGI PENELITIAN

Penelitian ini menggunakan pendekatan kuantitatif dengan kerangka kerja CRISP-DM

(*Cross-Industry Standard Process for Data Mining*) sebagai landasan metodologi, dipilih karena kesesuaiannya dengan tahapan pengembangan sistem kecerdasan buatan, khususnya pemrosesan data teks dan pembangunan arsitektur Retrieval-Augmented Generation (RAG) [11]. Proses ini terdiri dari lima tahap: pemahaman bisnis (*business understanding*), pemahaman data (*data understanding*), persiapan data (*data preparation*), pemodelan (*modeling*), dan evaluasi (*evaluation*).

### A. Pemahaman Bisnis (*Business Understanding*)

Tahap *business understanding* dilakukan untuk mengidentifikasi permasalahan dan tujuan penelitian, yaitu keterbatasan media simulasi wawancara kerja yang mampu menghasilkan pertanyaan sesuai posisi pekerjaan serta memberikan evaluasi jawaban secara objektif. Oleh karena itu, penelitian ini mengembangkan sistem simulasi wawancara berbasis *Retrieval-Augmented Generation* (RAG) yang memanfaatkan *knowledge base* dan *Large Language Model* (LLM) untuk menghasilkan pertanyaan dan evaluasi jawaban yang relevan, kontekstual, dan minim halusinasi.

### B. Pemahaman Data (*Data Understanding*)

Tahap *data understanding* bertujuan untuk mengidentifikasi sumber data yang digunakan dalam penelitian. Dataset diperoleh dari platform Kaggle, yaitu HR Interview Questions and Ideal Answers serta Software Engineering Interview Questions Dataset. Kedua dataset tersebut berisi kumpulan pertanyaan dan jawaban wawancara yang mencakup aspek teknis maupun nonteknis. Data ini digunakan sebagai sumber utama dalam pembangunan *knowledge base* yang mendukung proses *retrieval* pada arsitektur *Retrieval-Augmented Generation* (RAG).

### C. Persiapan Data (*Data Preparation*)

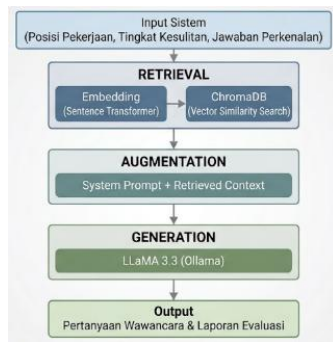
Tahap *data preparation*, kedua dataset terlebih dahulu melalui proses pra-pemrosesan yang meliputi pembersihan data, penyeragaman atribut, penggabungan dataset, penghapusan data duplikat, serta penambahan metadata untuk menghasilkan *knowledge base* yang terstruktur dan konsisten. Selanjutnya, setiap dokumen diubah menjadi representasi numerik (*vector embedding*) menggunakan model Sentence Transformer (*paraphrase-multilingual-MiniLM-L12-v2*). Hasil *embedding* kemudian disimpan pada basis data vektor ChromaDB sebagai *knowledge base* yang siap digunakan pada proses *retrieval*.

#### D. Pemodelan (Modeling)

Tahap *modeling* dilakukan dengan mengimplementasikan dua model utama, yaitu arsitektur *Retrieval-Augmented Generation* (RAG) untuk menghasilkan pertanyaan wawancara dan evaluasi jawaban secara kontekstual, serta metode TF-IDF dan *Cosine Similarity* untuk menghitung tingkat kemiripan jawaban pelamar terhadap jawaban referensi. Kedua model tersebut saling melengkapi dalam mendukung proses simulasi wawancara kerja.

##### 1. Retrieval-Augmented Generation (RAG)

Perancangan sistem RAG bertujuan mengotomatisasi pembangkitan pertanyaan wawancara dan evaluasi jawaban kandidat, dengan mengombinasikan proses retrieval dari knowledge base dan kemampuan LLM agar keluaran tetap relevan, kontekstual, dan sesuai posisi pekerjaan. Implementasi RAG terdiri atas tiga tahap utama: retrieval, augmentation, dan generation [1], sebagaimana ditunjukkan pada Gambar 1.



Gambar 1 Arsitektur RAG

Proses diawali dengan input sistem berupa posisi pekerjaan, tingkat kesulitan, dan jawaban pengenalan kandidat yang telah ditranskripsikan. Pada tahap *retrieval*, kueri diubah menjadi representasi vektor menggunakan *Sentence Transformer* (*paraphrase-multilingual-MiniLM-L12-v2*), lalu dicocokkan melalui similarity search pada ChromaDB untuk memperoleh dokumen paling relevan sebagai konteks.

Pada tahap *augmentation*, konteks hasil retrieval digabungkan dengan system prompt berisi instruksi perilaku AI interviewer serta informasi posisi dan tingkat kesulitan wawancara, menghasilkan prompt yang lebih kaya konteks.

Tahap *generation* menggunakan LLaMA 3.3 untuk membangkitkan pertanyaan wawancara

berikutnya atau laporan evaluasi jawaban kandidat berdasarkan prompt yang telah diperkaya.

##### 2. TF-IDF dan Cosine Similarity

Metode *Term Frequency-Inverse Document Frequency* (TF-IDF) dan *Cosine Similarity* digunakan untuk menghitung tingkat kemiripan antara jawaban pelamar dan jawaban referensi. TF-IDF memberikan bobot pada setiap term berdasarkan frekuensi kemunculannya dalam dokumen (*Term Frequency*) dan tingkat keunikannya pada korpus (*Inverse Document Frequency*) [12], yang hasilnya direpresentasikan sebagai vektor untuk perhitungan *Cosine Similarity*.

Bobot TF-IDF setiap *term* dihitung menggunakan Persamaan (1).

$$TF-IDF(t, d) = TF(t, d) \times IDF(t) \quad (1)$$

Keterangan:

- $TF-IDF(t, d)$  = bobot term  $t$  pada dokumen  $d$
- $TF(t, d)$  = nilai *Term Frequency* term  $t$  pada dokumen  $d$
- $IDF(t)$  = nilai *Inverse Document Frequency* term  $t$

Selanjutnya, nilai TF-IDF dari masing-masing dokumen selanjutnya digunakan untuk menghitung *Cosine Similarity*, yaitu metode yang mengukur kemiripan antara dua dokumen berdasarkan sudut representasi vektornya, menghasilkan nilai pada rentang 0 hingga 1, semakin mendekati 1, semakin tinggi tingkat kemiripannya [14]. Dalam penelitian ini, *Cosine Similarity* digunakan untuk membandingkan jawaban pelamar terhadap jawaban referensi berdasarkan bobot TF-IDF yang telah diperoleh, dengan perhitungan mengacu pada Persamaan (2).

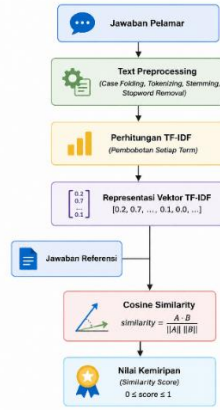
$$similarity = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}} \quad (2)$$

Keterangan:

- $A$  = vektor pertama yang digunakan dalam proses perbandingan kemiripan
- $B$  = vektor kedua yang digunakan dalam proses perbandingan kemiripan
- $A \cdot B$  = hasil *dot product* (perkalian titik) antara vektor  $A$  dan vektor  $B$
- $|A|$  = nilai panjang atau magnitudo dari vektor  $A$
- $|B|$  = nilai panjang atau magnitudo dari vektor  $B$

- $|A| \times |B|$  = hasil perkalian magnitudo vektor A dan vektor B yang digunakan sebagai faktor normalisasi pada perhitungan *Cosine similarity*

Berikut ini gambar 2 menunjukkan alur perhitungan kemiripan jawaban menggunakan metode TF-IDF dan Cosine Similarity.



Gambar 2 Perhitungan Kemiripan Jawaban

Berdasarkan Gambar 2, proses evaluasi jawaban diawali dengan menerima jawaban pelamar sebagai masukan sistem, yang kemudian melalui tahap *text preprocessing* (*case folding, tokenizing, stemming, dan stopwords removal*) untuk menyeragamkan bentuk kata serta menghilangkan kata yang tidak bermakna penting. Hasil pra-pemrosesan tersebut digunakan pada perhitungan TF-IDF untuk memperoleh bobot tiap term berdasarkan frekuensi kemunculan dan tingkat kepentingannya dalam korpus, yang selanjutnya direpresentasikan sebagai vektor TF-IDF.

#### E. Evaluasi (Evaluation)

Tahap *evaluation* dilakukan untuk mengukur kinerja sistem, baik dari aspek retrieval maupun generation, melalui dua pendekatan: evaluasi otomatis menggunakan *Retrieval-Augmented Generation Assessment* (RAGAs) untuk mengukur kualitas sistem secara kuantitatif, dan evaluasi oleh pakar (*human evaluation*) untuk memvalidasi kualitas penalaran (*reasoning*) serta kesesuaian respons terhadap konteks wawancara.

##### 1. Retrieval-Augmented Generation Assessment (RAGAs)

RAGAs merupakan kerangka kerja evaluasi yang dirancang untuk mengukur performa sistem *Retrieval-Augmented Generation* (RAG) tanpa memerlukan data referensi (*ground truth*) [8]. Pada

penelitian ini, evaluasi dilakukan menggunakan tiga metrik, yaitu *Faithfulness*, *Answer Relevance*, dan *Context Precision*.

##### a. Faithfulness

Metrik *Faithfulness* digunakan untuk mengukur tingkat konsistensi faktual antara respons yang dihasilkan sistem dengan konteks yang diperoleh dari knowledge base. Semakin tinggi nilai *Faithfulness*, semakin kecil kemungkinan model menghasilkan informasi yang bersifat halusinasi. Perhitungan *Faithfulness* ditunjukkan pada Persamaan (3).

$$Faithfulness = \frac{\text{Jumlah klaim yang terverifikasi}}{\text{Total jumlah klaim dalam jawaban}} \quad (3)$$

Keterangan:

- Jumlah klaim terverifikasi = jumlah klaim yang sesuai dengan dokumen sumber.
- Total jumlah klaim = jumlah seluruh klaim dalam jawaban sistem.

##### b. Answer Relevance

Metrik *Answer Relevance* digunakan untuk mengukur tingkat kesesuaian jawaban yang dihasilkan terhadap pertanyaan yang diberikan pengguna berdasarkan kesamaan semantik. Nilai yang semakin mendekati 1 menunjukkan bahwa jawaban semakin relevan terhadap pertanyaan. Perhitungan *Answer Relevance* ditunjukkan pada Persamaan (4).

$$Answer\ Relevance = \frac{1}{N} \sum_{i=1}^N cosine_{similarity}(E_q, E_{sqi}) \quad (4)$$

Keterangan:

- N adalah jumlah pertanyaan hipotesis yang dibuat
- $E_q$  adalah embedding dari pertanyaan asli pengguna
- $E_{sqi}$  adalah embedding dari pertanyaan hipotesis ke-i yang dihasilkan dari jawaban
- $Cosine_{similarity}$  adalah fungsi kesamaan kosinus antara dua vektor

##### c. Context Precision

Metrik *Context Precision* digunakan untuk mengukur kemampuan sistem dalam mengambil dan memprioritaskan dokumen yang relevan pada proses retrieval. Nilai yang tinggi menunjukkan bahwa dokumen yang paling relevan berhasil ditempatkan pada urutan teratas sehingga dapat meningkatkan kualitas respons yang dihasilkan. Perhitungan *Context Precision* ditunjukkan pada Persamaan (5).

$$\text{Context Precision}@k = \frac{\sum_{k=1}^K (\text{Precision}@k \times u_k)}{\text{Total Relevant Items}} \quad (5)$$

Keterangan:

- K adalah jumlah total potongan konteks yang diambil oleh sistem
- Precision@k adalah nilai presisi pada peringkat ke-k
- $u_k$  adalah nilai biner 1 atau 0, bernilai 1 jika item pada peringkat ke-k, dan 0 jika tidak
- Total Relevant Items adalah jumlah total item relevan yang terdapat dalam kumpulan konteks yang diambil

## 2. Human Evaluation

Selain evaluasi otomatis, penelitian ini juga menerapkan human evaluation untuk memvalidasi kualitas respons yang dihasilkan sistem. Evaluasi dilakukan oleh dua orang pakar yang memiliki pengalaman pada bidang teknologi informasi. Para evaluator diminta menilai apakah hasil evaluasi yang diberikan sistem telah sesuai dengan jawaban kandidat berdasarkan aspek relevansi, ketepatan, dan logika penalaran [3]. Hasil penilaian tersebut digunakan sebagai dasar untuk mengukur tingkat validitas keluaran sistem serta melengkapi hasil evaluasi otomatis yang diperoleh dari RAGAs.

## III. HASIL DAN PEMBAHASAN

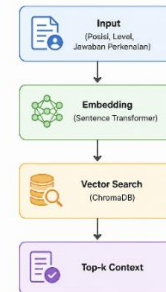
### A. Hasil Implementasi Retrieval-Augmented Generation (RAG)

Model *Retrieval-Augmented Generation* (RAG) berhasil diimplementasikan untuk mendukung proses simulasi wawancara kerja secara adaptif. Implementasi model mengintegrasikan mekanisme retrieval untuk memperoleh informasi yang relevan dari *knowledge base*, *augmentation* untuk menggabungkan konteks hasil pencarian dengan instruksi sistem, serta *generation* menggunakan *Large Language Model* (LLM) dalam menghasilkan pertanyaan wawancara dan evaluasi jawaban kandidat. Melalui integrasi ketiga tahapan tersebut, sistem mampu menghasilkan respons yang lebih kontekstual, relevan dengan posisi pekerjaan, serta mengurangi potensi halusinasi dibandingkan penggunaan LLM tanpa mekanisme retrieval.

#### 1. Retrieval

Tahap *retrieval* bertujuan memperoleh informasi yang paling relevan dari *knowledge base* berdasarkan konteks wawancara. Pada implementasi penelitian ini, proses pencarian mempertimbangkan posisi pekerjaan, tahapan wawancara, tingkat

kesulitan, serta riwayat pertanyaan yang telah digunakan sehingga pertanyaan yang diperoleh lebih sesuai dengan kondisi wawancara dan tidak berulang. Alur implementasi tahap *retrieval* ditunjukkan pada Gambar 3.



Gambar 3 Alur Proses Retrieval

Berdasarkan Gambar 3, proses *retrieval* diawali dengan masukan berupa posisi pekerjaan, level kandidat, dan jawaban perkenalan yang diberikan pada awal sesi wawancara. Informasi tersebut kemudian diubah menjadi representasi vektor (*embedding*) menggunakan model *Sentence Transformer* sehingga dapat direpresentasikan dalam ruang vektor. Selanjutnya, vektor hasil *embedding* digunakan untuk melakukan proses *vector search* pada basis data vektor *ChromaDB* guna menemukan dokumen yang memiliki tingkat kemiripan tertinggi terhadap masukan pengguna. Hasil pencarian berupa *Top-k Context*, yaitu sejumlah konteks atau dokumen yang paling relevan, yang selanjutnya digunakan sebagai masukan pada tahap *augmentation*.

#### 2. Augmentation

Setelah dokumen yang relevan diperoleh, sistem melakukan tahap *augmentation* dengan menggabungkan hasil retrieval bersama informasi pendukung, seperti jawaban referensi, posisi pekerjaan, level kandidat, tahapan wawancara, dan riwayat pertanyaan. Proses ini membentuk konteks yang lebih lengkap sehingga model bahasa dapat menghasilkan keluaran yang konsisten dengan kebutuhan wawancara.

TABEL I. Konteks yang dibentuk pada Tahap Augmentation

| Komponen           | Nilai                                                                                                                                                                                                               |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Posisi             | Software Engineer                                                                                                                                                                                                   |
| Level              | Junior                                                                                                                                                                                                              |
| Tahap              | Technical                                                                                                                                                                                                           |
| Pertanyaan         | Apa perbedaan antara array dan daftar tertaut?                                                                                                                                                                      |
| Jawaban Referensi  | Sebuah array memiliki ukuran tetap dan menyimpan elemen dalam memori yang berdekatan, sedangkan daftar tertaut terdiri dari node yang saling terhubung melalui referensi sehingga memungkinkan ukuran yang dinamis. |
| Riwayat Pertanyaan | Dua Pertanyaan Sebelumnya                                                                                                                                                                                           |

Berdasarkan Tabel I, tahap *augmentation* menyusun konteks yang akan digunakan sebagai masukan bagi model bahasa dengan menggabungkan hasil *retrieval* dan informasi pendukung lainnya. Konteks tersebut meliputi posisi pekerjaan, level kandidat, tahapan wawancara, tingkat kesulitan, pertanyaan hasil *retrieval*, jawaban referensi, serta riwayat pertanyaan sebelumnya. Penggabungan informasi tersebut bertujuan agar model bahasa menghasilkan pertanyaan yang relevan dengan posisi yang dilamar, sesuai tingkat kemampuan kandidat, serta menghindari pengulangan pertanyaan selama sesi wawancara.

### 3. Generation

Tahap *generation* memanfaatkan model LLaMA 3.3 70B untuk menghasilkan keluaran berdasarkan konteks yang telah dibangun pada tahap sebelumnya. Pada penelitian ini, model digunakan untuk menghasilkan pertanyaan wawancara secara adaptif, mengevaluasi jawaban kandidat, serta menyusun laporan akhir wawancara. Dengan memanfaatkan konteks hasil *retrieval* dan *augmentation*, keluaran yang dihasilkan menjadi lebih relevan terhadap posisi pekerjaan dan jalannya sesi wawancara.

#### a. Generate Pertanyaan

Pada tahap ini, model bahasa memanfaatkan konteks yang telah dibangun pada proses *retrieval* dan *augmentation* untuk menghasilkan pertanyaan wawancara secara adaptif. Pertanyaan yang dihasilkan disesuaikan dengan posisi pekerjaan, tahapan wawancara, serta riwayat percakapan sehingga mampu mengarahkan jalannya sesi

wawancara secara bertahap. Contoh hasil Generate pertanyaan ditunjukkan pada Gambar 4.



Gambar 4 Hasil Generate Pertanyaan

Berdasarkan Gambar 4, sistem menghasilkan pertanyaan pembuka yang meminta kandidat memperkenalkan diri serta menjelaskan pengalaman dan teknologi yang pernah digunakan sesuai dengan posisi *Software Engineer*. Pertanyaan tersebut dibangun berdasarkan konteks yang telah disusun pada tahap *augmentation* sehingga mampu mengarahkan jalannya wawancara sebelum memasuki tahap pertanyaan teknis. Setelah kandidat memberikan jawaban, sistem kembali melakukan proses *retrieval*, *augmentation*, dan *generation* untuk menghasilkan pertanyaan berikutnya secara dinamis sesuai dengan konteks percakapan yang sedang berlangsung.

#### b. Evaluasi Jawaban

Pada tahap ini, model mengevaluasi jawaban kandidat berdasarkan konteks yang telah dibangun pada proses *retrieval* dan *augmentation*, dengan membandingkannya terhadap jawaban referensi pada knowledge base untuk menghasilkan klasifikasi kualitas jawaban beserta umpan balik perbaikan. Contoh hasil evaluasi ditunjukkan pada Tabel II.

TABEL II. Contoh Evaluasi Jawaban Oleh Model

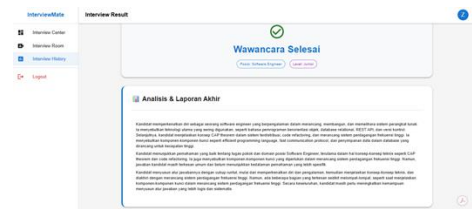
| Pertanyaan                                 | Jawaban Kandidat                                                       | Jawaban Referensi                                                                                                                                                                       |
|--------------------------------------------|------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Apa yang dimaksud dengan code refactoring? | Refactoring adalah memperbaiki struktur kode tanpa mengubah fungsinya. | Average. Jawaban telah menjelaskan konsep dasar code refactoring, namun belum membahas peningkatan keterbacaan, kemudahan pemeliharaan, dan kualitas perangkat lunak secara menyeluruh. |

Berdasarkan Tabel II, model mengklasifikasikan jawaban kandidat ke dalam kategori *Average* karena telah menjelaskan konsep dasar code refactoring, namun belum mencakup poin penting pada jawaban referensi seperti peningkatan keterbacaan, kemudahan pemeliharaan, dan kualitas perangkat lunak. Umpan balik tersebut diharapkan

membantu kandidat memahami kekurangan jawabannya untuk memberikan respons yang lebih lengkap pada sesi berikutnya.

c. Generate Laporan Akhir

Pada tahap ini, model bahasa menghasilkan laporan akhir secara otomatis setelah seluruh sesi wawancara selesai. Laporan disusun berdasarkan keseluruhan riwayat tanya jawab selama wawancara dengan memanfaatkan konteks yang telah dikumpulkan pada setiap proses retrieval, augmentation, dan generation. Hasil pembangkitan laporan akhir ditunjukkan pada Gambar 5.



Gambar 5 Hasil Generate Laporan Akhir

Berdasarkan Gambar 5, model menghasilkan laporan akhir yang memuat ringkasan hasil wawancara, penilaian kemampuan teknis, dan analisis struktur berpikir kandidat berdasarkan seluruh jawaban selama sesi wawancara mencakup pengalaman dan kompetensi kandidat, evaluasi pemahaman terhadap konsep teknis yang dibahas, serta rekomendasi aspek yang perlu ditingkatkan. Hasil ini menunjukkan bahwa model mampu menghasilkan tidak hanya pertanyaan secara adaptif, tetapi juga evaluasi akhir yang komprehensif dan kontekstual berdasarkan riwayat wawancara.

B. Hasil Evaluasi RAGAs

Untuk mengukur kinerja model *Retrieval-Augmented Generation* (RAG), dilakukan evaluasi menggunakan framework *Retrieval-Augmented Generation Assessment* (RAGAs). Evaluasi dilakukan terhadap riwayat simulasi wawancara yang dihasilkan oleh sistem dengan memanfaatkan pendekatan *Large Language Model* (LLM) sebagai penilai (LLM-as-a-Judge). Pengujian dilakukan menggunakan tiga metrik, yaitu *Answer Relevance*, *Context Precision*, dan *Faithfulness*, yang masing-masing digunakan untuk mengukur relevansi respons, kualitas proses retrieval, serta konsistensi respons terhadap konteks yang digunakan. Hasil evaluasi RAGAs ditunjukkan pada Tabel III.

TABEL III. Hasil Evaluasi RAGAs

| Metrik            | Skor   |
|-------------------|--------|
| Answer Relevance  | 0,9854 |
| Context Precision | 0,9200 |
| Faithfulness      | 0,9413 |

Berdasarkan Tabel III, model memperoleh nilai *Answer Relevance* sebesar 0,9854, menunjukkan tingkat relevansi respons yang sangat tinggi terhadap pertanyaan yang diberikan; *Context Precision* sebesar 0,9200, menunjukkan mekanisme *retrieval* mampu mengambil dan memprioritaskan konteks relevan dari *knowledge base*; serta *Faithfulness* sebesar 0,9413, menunjukkan respons tetap konsisten dengan konteks yang diperoleh sehingga meminimalkan potensi halusinasi. Secara keseluruhan, hasil tersebut menunjukkan bahwa model RAG yang diusulkan mampu menghasilkan respons yang relevan, didukung konteks yang tepat, dan konsisten terhadap sumber informasi yang digunakan.

C. Hasil Human Evaluation

Selain evaluasi otomatis menggunakan RAGAs, penelitian ini juga menerapkan human evaluation untuk memvalidasi kualitas evaluasi yang dihasilkan oleh model. Evaluasi dilakukan oleh dua orang pakar di bidang Teknologi Informasi yang berperan sebagai evaluator independen. Masing-masing pakar diminta menilai kesesuaian antara jawaban kandidat dan evaluasi yang dihasilkan sistem pada 15 sampel data wawancara. Hasil penilaian human evaluation disajikan pada Tabel IV.

TABEL IV. Hasil Human Evaluation

| Hasil Penilaian | Jumlah Sampel |
|-----------------|---------------|
| Valid           | 10            |
| Invalid         | 5             |
| Total           | 15            |

Berdasarkan Tabel IV, sebanyak 10 dari 15 sampel dinilai valid dan 5 sampel dinilai invalid oleh kedua evaluator, menunjukkan bahwa sebagian besar evaluasi yang dihasilkan model telah sesuai dengan jawaban kandidat dan mampu memberikan umpan balik yang relevan. Sampel yang dinilai invalid umumnya disebabkan oleh ketidaksesuaian pada proses penalaran (*reasoning*), misalnya sistem menyatakan suatu konsep belum dibahas atau memberikan umpan balik yang kurang sesuai

dengan isi jawaban kandidat, sehingga model masih memerlukan penyempurnaan pada proses penyusunan umpan balik agar mampu menginterpretasikan jawaban kandidat secara lebih akurat.

#### IV. KESIMPULAN DAN SARAN

Penelitian ini berhasil mengimplementasikan metode *Retrieval-Augmented Generation* (RAG) pada sistem simulasi wawancara kerja berbasis web untuk menghasilkan pertanyaan wawancara dan evaluasi jawaban secara adaptif sesuai posisi pekerjaan. Melalui integrasi proses *retrieval*, *augmentation*, dan *generation*, sistem memanfaatkan knowledge base sebagai sumber konteks sehingga pertanyaan dan evaluasi yang dihasilkan lebih relevan dan mengurangi potensi halusinasi model bahasa.

Hasil evaluasi menggunakan framework RAGAs menunjukkan performa yang baik, dengan nilai *Answer Relevance* 0,9854, *Context Precision* 0,9200, dan *Faithfulness* 0,9413. Human evaluation menunjukkan 10 dari 15 sampel dinilai valid oleh dua pakar, mengindikasikan bahwa sebagian besar evaluasi sistem telah sesuai dengan jawaban kandidat. Hasil tersebut menunjukkan bahwa pendekatan RAG efektif diterapkan untuk mendukung simulasi wawancara kerja yang lebih kontekstual dan adaptif.

Pengembangan selanjutnya dapat diarahkan pada perluasan *knowledge base* agar mencakup lebih banyak bidang pekerjaan dan tingkat pengalaman, penyempurnaan reasoning model bahasa untuk interpretasi jawaban yang lebih akurat (khususnya pada kasus yang masih dinilai invalid), serta pelibatan sampel dan evaluator yang lebih banyak untuk validasi yang lebih komprehensif.

Melalui integrasi proses *retrieval*, *augmentation*, dan *generation*, sistem mampu memanfaatkan *knowledge base* sebagai sumber konteks sehingga pertanyaan dan evaluasi yang dihasilkan menjadi lebih relevan serta mengurangi potensi halusinasi pada model bahasa.

#### V. REFERENSI

- [1] Albert, G. D., & Voutama, A. (2025). PDF MENGGUNAKAN LOCAL RETRIEVAL-AUGMENTED GENERATION (RAG) DAN OLLAMA. 13(2).
- [2] Apollo Technical. (2024). *Social Media Recruiting Statistics*. [Online]. Available: <https://www.apollotechnical.com/social-media-recruiting-statistics/>. [Accessed: Jul. 15, 2026].
- [3] Babu, S., & Krishnan, P. (2025). International Journal of Innovative Research in Science Engineering and Technology ( IJRSET ) LLM Evaluations : A Survey of Programmatic , Human , and LLM-as-Judge Approaches. 14(6), 6–11. <https://doi.org/10.15680/IJRSET.2025.1406011>
- [4] Balaguer, A., Benara, V., Cunha, R., Estevão, R., Hendry, T., Holstein, D., Marsman, J., Mecklenburg, N., Malvar, S., Nunes, L. O., Padilha, R., Sharp, M., Silva, B., Sharma, S., Aski, V., & Chandra, R. (2023). *VS Microsoft*.
- [5] Brown, C., Tech, V., & Tech, V. (2025). Towards Evidence-Based Tech Hiring Pipelines. In *Proceedings of International Workshop on Software Engineering in 2030 (SE 2030)* (Vol. 1, Issue 1). Association for Computing Machinery.
- [6] Giffar, Z. M., Adnan, I. Z., Hendrawan, H., Studi, P., Komunikasi, I., Garut, U., Garut, K., Schutz, A., Komunikasi, K., Kerja, W., Schutz, A., Anxiety, C., & Interview, J. (2024). *MAKNA KECEMASAN KOMUNIKASI DALAM WAWANCARA KERJA BAGI*. 9(4), 810–828.
- [7] Hu, R., Liu, S., Qi, P., Liu, J., & Li, F. (2025). ICCA-RAG : Intelligent Customs Clearance Assistant Using Retrieval-Augmented Generation ( RAG ). *IEEE Access*, 13(January), 39711–39726. <https://doi.org/10.1109/ACCESS.2025.3544408>
- [8] Kharisma, I. L., Hidayat, M. S., Somantri, & Kamdan. (2025). Implementasi Retrieval Augmented Generation dalam Sistem Chatbot Dermatologi Berbasis Website Implementation of Retrieval Augmented Generation in a Web-Based Dermatological Chatbot System. 11, 448–462.
- [9] Lata, S. (2025). Retrieval-Augmented Generation ( RAG ) and Memory Systems for HR and Enterprise AI. 0–3.
- [10] Pathak, Gangesh., Pandey, D. (2025). AI Agents in Recruitment: A Multi-Agent System for Interview, Evaluation, and Candidate Scoring.
- [11] Rianti, A., Wachid, N., Majid, A., & Fauzi, A. (2023). *CRISP-DM : Metodologi Proyek Data Science*. 107–114.
- [12] S, Y. A. P. G., Ginting, G. L., & Panjaitan, M. (2023). Optimasi Penerimaan Siswa Baru dengan Penerapan Algoritma Text Mining dan TF-IDF. 2(3), 109–115. <https://doi.org/10.47065/comforch.v2i3.941>
- [13] Saha, B., Saha, U., & Member, G. S. (2024). QuIM-RAG : Advancing Retrieval-Augmented Generation With Inverted Question Matching for Enhanced QA Performance. *IEEE Access*, 12(December), 185401–185410. <https://doi.org/10.1109/ACCESS.2024.3513155>
- [14] Siswipraptini, P. C., Wafit, M. I., & Djunaidi, K. (2023). Klasifikasi Pekerjaan Bidang Teknologi Informasi Menggunakan Algoritma Cosine Similarity. 12(1), 38–48.
- [15] Tohir, H., Merlina, N., & Haris, M. (2024). *UTILIZING RETRIEVAL-AUGMENTED GENERATION IN LARGE LANGUAGE MODELS TO ENHANCE INDONESIAN LANGUAGE NLP*. 10(2), 352–360. <https://doi.org/10.33480/jitk.v10i2.5916>.INTRODUCTIO N
- [16] Yang, F., & Yang, M. (2023). LongRoPE: Extending LLM Context Window Beyond 2 Million Tokens.