

Automated Resume Screening using Retrieval-Augmented Generation

Pavan Kumar Sunkara and Veda Prakash Chiliveru

Data Science(Master of Science)

University of New Haven

psunk3@unh.newhaven.edu, vchil7@unh.newhaven.edu

Abstract

Resume screening is a requisite but typically time-consuming task for hiring teams, typically involving a manual process over numerous applications. In this study, we introduce an RAG-based system to facilitate the automation of resume screening. We propose to unify semantic resume retrieval based on FAISS and Sentence-BERT embeddings with professional feedback generation based on large language models such as Flan-T5. We further conduct an analysis of the skill mismatch between job postings and candidate profiles. We compare our system across ten job areas and measure recall efficacy using Recall@5 and Precision@5. Side-by-side comparison across different LLMs puts into context variation in reasoning quality and response generation. Our results demonstrate that RAG-like architectures have the potential to radically enhance the effectiveness and consistency of candidate pipeline evaluation in recruiting.

Introduction

Recruitment is a critical component of organizational growth, and at the heart of the recruitment process lies resume screening — a stage that remains both essential and labor-intensive. With hundreds of applicants vying for a single job posting, hiring teams often face the daunting task of manually reviewing resumes to assess alignment with job requirements. This process is time-consuming, repetitive, and prone to inconsistencies due to human fatigue and subjectivity. Although Applicant Tracking Systems (ATS) have been deployed to ease this burden, they typically rely on keyword-based matching mechanisms. These systems are often brittle, overlooking qualified candidates who use varied phrasing and advancing unqualified candidates who engage in keyword stuffing.

Traditional ATS lack the semantic understanding required to interpret the nuanced and diverse ways in which candidates articulate their qualifications. As a result, companies risk missing top talent and introducing biases into the hiring funnel. There is a clear need for intelligent systems that can perform contextual matching, assess candidates holistically, and provide consistent feedback.

Recent advances in Natural Language Processing (NLP) and deep learning, especially the emergence of Retrieval-Augmented Generation (RAG) frameworks, offer a compelling alternative. RAG models combine the strengths of dense information retrieval with the power of generative language models, enabling systems to retrieve contextually relevant documents and generate coherent, informative responses. These systems have shown promising results in open-domain question answering, knowledge-grounded dialogue, and document summarization — tasks that parallel many of the challenges faced in resume screening.

In this project, we present a simplified, yet effective RAG-based pipeline tailored for automated resume screening. Our system integrates semantic retrieval using Sentence-BERT embeddings and FAISS indexing to identify candidate profiles that are closely aligned with a given job description. Once relevant resumes are retrieved, the system conducts a skill gap analysis between job requirements and candidate qualifications. Finally, we employ a large language model, Flan-T5, to generate personalized feedback reports for each candidate. These reports highlight strengths, flag areas for improvement, and offer an overall assessment of suitability for the role.

We curated a multi-domain dataset covering job postings across ten industry categories to ensure the generalizability of our system. To evaluate the retrieval component, we use standard metrics such as Recall@5 and Precision@5. For the generative

component, we compare output quality and factual accuracy across multiple language models, providing a comprehensive view of system performance. Our experiments demonstrate that even a lightweight RAG architecture can significantly reduce manual screening time while maintaining a high standard of evaluation quality.

In addition to technical performance, we also discuss practical considerations in deploying such a system in real-world recruitment workflows. These include handling domain-specific jargon, ensuring fairness and bias mitigation, and interpreting model-generated feedback in a human-readable format. We believe our work lays the groundwork for scalable, intelligent recruitment systems that combine the precision of retrieval with the nuance of human-like evaluation.

Through this study, we aim to show that RAG-based automation can enhance both the efficiency and fairness of candidate screening, ultimately enabling recruiters to make better-informed hiring decisions. As the demand for data-driven HR solutions grows, systems like ours can help bridge the gap between AI capability and real-world recruitment needs.

Our contributions include:

- A pipeline integrating embedding-based retrieval with skill comparison and LLM-based feedback.
- A comparative study across ten job domains using a custom dataset.
- Analysis of three language models in terms of generation quality, factual correctness, and accessibility.
- Recommendations for improving AI-based screening and future deployment considerations.

System Overview

Our proposed solution is to build around a modular pipeline inspired by the Retrieval-Augmented Generation (RAG) framework, specifically adapted for the task of automated resume screening. This system is designed to semantically match resumes to job descriptions, identify skill gaps, and generate human-like feedback summaries using a combination of dense retrieval and language generation techniques. The architecture is fully

modular, allowing individual components to be updated or enhanced without affecting the overall workflow. Below, we provide an overview of the key stages in the system:

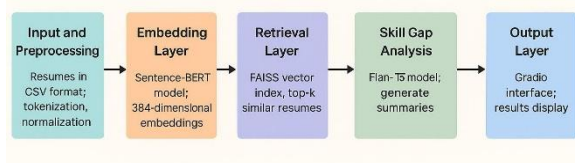


Figure 1: Proposed System Overview

Input Handling and Preprocessing

The pipeline begins with the ingestion of candidate resumes, which are uploaded in a structured CSV format. Each row corresponds to a complete resume, including details such as educational background, professional experience, skill sets, and personal summaries. To prepare the data for downstream processing, we apply standard natural language preprocessing techniques. These include converting all text to lowercase, removing punctuation, and applying tokenization to normalize and clean the input. This step ensures consistency in text representations before embedding.

Semantic Embedding Generation

We use the Sentence-BERT model all-MiniLM-L6-v2 to transform each preprocessed resume into a fixed-length vector representation (embedding) of 384 dimensions. This model captures contextual and semantic nuances at the sentence level, enabling the system to recognize meaningful relationships between job descriptions and candidate profiles, even in the absence of exact keyword matches. Sentence embeddings provide a more robust foundation for similarity computation compared to traditional bag-of-words or TF-IDF approaches.

Efficient Resume Retrieval

To facilitate fast and scalable retrieval of the most relevant resumes, we utilize FAISS (Facebook AI Similarity Search), an optimized similarity search library for dense vectors. Resumes are indexed using FAISS in memory. When a recruiter inputs a job description, it is also converted into an embedding using the same Sentence-BERT model. A top-k (where k=5) nearest-neighbor search is performed to identify resumes that are most semantically aligned with the job description. This retrieval step narrows down the candidate pool for deeper analysis and feedback generation.

Skill Gap Identification

We perform a skill gap analysis by extracting relevant keywords from both the job descriptions and the top retrieved resumes. These keyword sets are then compared using set-based operations to determine which skills are matched and which are missing. This simple yet effective approach allows the system to highlight specific areas where a candidate aligns or diverges from the job requirements. In future enhancements, this module could incorporate Named Entity Recognition (NER) techniques or leverage domain-specific skill ontologies to improve the granularity and accuracy of skill mapping.

Generative Feedback with Language Models

To produce concise and insightful feedback on each candidate, we use the Flan-T5 language model. For each of the retrieved resumes, we craft a structured prompt that includes matched and missing skills along with an overall alignment score. The model generates a short, natural-sounding summary (typically 1–2 sentences) that mimics the type of feedback a human recruiter might provide. Sampling parameters such as temperature (set to 0.9) and top-p (set to 0.95) are tuned to balance creativity and relevance in the generated text, ensuring outputs are informative yet readable.

User Interface and Output Delivery

The results are compiled and presented via an interactive Gradio web application, providing an accessible and user-friendly interface for recruiters. Each matched resume is shown along with its similarity score, matched/missing skills, and the LLM-generated feedback summary. Additionally, users can download the results in Excel format for further analysis or offline review. This integration with Gradio makes the system easy to deploy and interact with, especially for non-technical users.

Technology Stack and Deployment

The system is implemented using Python and leverages a suite of modern libraries and tools. These include Hugging Face Transformers for embedding and generation, FAISS for vector search, Pandas for data manipulation, and Gradio for web-based visualization. For accessibility and reproducibility, the entire pipeline is deployed on Google Colab, enabling researchers and practitioners to easily test and adapt the system without requiring complex infrastructure setup.

This modular, end-to-end architecture not only streamlines resume screening but also demonstrates the potential of RAG-inspired systems in practical recruitment scenarios. Each component retrieval, analysis, and generation can be independently updated, making the system highly adaptable to evolving models and data sources.

Domain Specific Questions

To ensure comprehensive evaluation and generalizability of our proposed RAG-based resume screening system, we constructed a domain-diverse dataset comprising ten representative job categories. These categories span both technical and non-technical fields, reflecting the breadth of roles commonly encountered in real-world recruitment workflows. Each domain is represented by a carefully crafted job description that encapsulates essential qualifications, key responsibilities, and domain-specific skill requirements.

This set of job descriptions serves as input queries to the retrieval-augmented pipeline, allowing us to test the system’s ability to identify and evaluate suitable resumes across a variety of professional contexts. Importantly, the descriptions were designed to be concise yet sufficiently detailed to simulate authentic recruiter expectations during the screening process.

- **Data Science**
Can the system identify candidates with strong programming skills in Python, a solid understanding of statistics, and hands-on experience with machine learning frameworks?
- **Mechanical Engineering**
How effectively does the system retrieve profiles that demonstrate knowledge of CAD tools, simulation techniques (e.g., FEA), and mechanical systems design?
- **Marketing**
Is the system capable of surfacing candidates with a background in digital marketing strategies, specifically with expertise in SEO, PPC advertising, and content creation?
- **Software Development**
Does the system successfully match resumes of full-stack developers who are proficient in front-end and back-end

technologies such as JavaScript, React, and Node.js?

- **Finance**

Can the system detect financial analysts with quantitative modeling skills, experience in risk assessment, and advanced spreadsheet capabilities in tools like Excel?

- **Healthcare**

How well does the system identify registered nurses or healthcare professionals with clinical experience and competencies in patient care protocols?

- **Education**

Is the system able to find educators who have expertise in curriculum design and online instruction, especially in subjects such as high school mathematics?

- **Cybersecurity**

Does the system retrieve profiles that reflect hands-on experience with cybersecurity protocols, including network defense, vulnerability management, and incident response?

- **Human Resources**

Can the system assess the suitability of HR professionals with experience in hiring, onboarding, and employee relations within corporate settings?

- **Business Administration**

How accurately does the system retrieve candidates with strong project coordination abilities and experience in operational planning and administrative leadership?

These domain-aligned questions enabled us to assess the system’s semantic understanding in context-rich scenarios and its capacity to differentiate between nuanced candidate profiles. By framing our evaluation in terms of natural hiring queries, we bridge the gap between academic benchmarking and practical recruitment utility.

Future extensions of this work may include the use of industry-specific ontologies or recruiter feedback loops to further refine question relevance and improve evaluation fidelity.

Implementation

A critical component of our Retrieval-Augmented Generation (RAG) pipeline is the language model (LM) used for generating candidate evaluation

summaries. The quality, fluency, and contextual awareness of the model directly affect the system’s usefulness in real-world recruitment scenarios. To this end, we explored three large language models (LLMs) to determine the most suitable option for our feedback generation task: Flan-T5, Mistral-7B, and Phi-2.

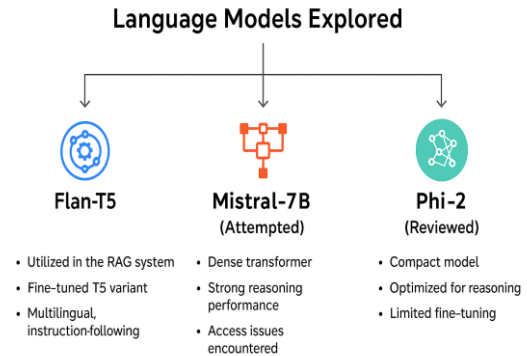


Figure 2: LLMs used

The primary model adopted for the implementation was **Flan-T5-large**, a widely accessible instruction-tuned version of the original T5 architecture developed by Google. This model is specifically optimized for zero-shot and instruction-following tasks, making it particularly well-suited for use cases involving natural feedback generation without requiring fine-tuning. We deployed the google/flan-t5-large model via Hugging Face, leveraging its robust generalization capabilities and ability to produce coherent, structured text across diverse candidate profiles and job domains.

In configuring Flan-T5, we aimed to strike a balance between creativity and relevance in the output. Sampling parameters were adjusted to enhance response diversity, with `do_sample=True`, a temperature setting of 0.9, and a generation limit of `max_new_tokens=150`. These settings allowed the model to produce concise, 1–2 sentence feedback statements that were both human-like and contextually grounded.

In parallel, we attempted to incorporate **Mistral-7B**, a high-performing dense transformer model known for its strong few-shot learning capabilities and competitive reasoning accuracy. Mistral-7B has recently shown promise on a variety of language understanding benchmarks. However, practical challenges hindered its integration into our pipeline. Specifically, the model was gated behind restricted access on Hugging Face, and repeated API calls

resulted in HTTP 401 authorization errors. Due to these access constraints and the lack of a stable inference endpoint within our Colab-based environment, we were unable to proceed with its deployment.

Additionally, we evaluated **Phi-2**, a compact and efficient model developed by Microsoft. Phi-2 is designed for lightweight inference and is reputed for its logical reasoning and mathematical correctness. While it is freely available and open source, its core design prioritizes structured reasoning tasks over conversational fluency. Furthermore, due to hardware limitations and the absence of optimized deployment APIs within our infrastructure, we opted not to integrate Phi-2 into the main pipeline. Instead, we conducted a comparative analysis based on published benchmarks and performance reviews from the literature.

Despite exploring multiple alternatives, **Flan-T5-large remained the model of choice** for all primary experiments. Its ease of integration, consistent output quality, and strong zero-shot performance across domains made it an ideal fit for the resume feedback generation task. The model was capable of capturing key resume-job description relationships and articulating strengths or skill gaps in natural language summaries that mirrored recruiter commentary.

This experience highlights both the potential and practical limitations of working with state-of-the-art LLMs in applied NLP systems. While models such as Mistral-7B and Phi-2 hold academic and technical promise, deployment feasibility and accessibility play a significant role in determining real-world applicability. Future iterations of this system may explore additional open-access instruction-tuned models or leverage model distillation techniques for improved efficiency in constrained environments.

Methodology

This study presents a modular and scalable Retrieval-Augmented Generation (RAG)-based architecture for automating the resume screening process. The methodology combines semantic retrieval and generative feedback generation to simulate human-level assessment of candidate suitability. Our approach integrates dense vector representations, similarity-based retrieval, skill gap analysis, and large language model (LLM) output generation within a unified pipeline.

The system begins with the ingestion of resume data in CSV format, where each row corresponds to a full-text resume containing candidate education, experience, skills, and summary sections. Standard natural language preprocessing steps—such as lowercasing, punctuation removal, and tokenization—are applied to ensure uniform text formatting. This processed data is then transformed into dense embeddings using the all-MiniLM-L6-v2 variant of Sentence-BERT, producing 384-dimensional vectors that encode semantic context at the sentence level.

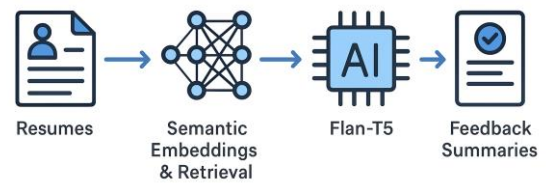


Figure 3: LLM Methodology

For retrieval, we employ Facebook AI Similarity Search (FAISS) to index the resume embeddings in memory, facilitating efficient similarity computation. Given a new job description, it is encoded using the same Sentence-BERT model, and a top-k ($k=5$) nearest-neighbor search is executed to identify the most semantically aligned resumes from the index. This enables the system to retrieve relevant candidate profiles even in the absence of exact keyword matches.

Once relevant resumes are identified, we perform a **skill gap analysis** by extracting key terms from both the job description and the candidate’s profile. These are compared using set operations to determine which job-required skills are present or missing. The results inform the construction of a prompt template, which is then fed into a large language model for summarization and feedback generation.

For the generative layer, we primarily used the google/flan-t5-large model hosted on Hugging Face. This instruction-tuned LLM is capable of producing context-aware, fluent summaries without additional fine-tuning. The generation process is configured with controlled sampling parameters: temperature=0.9, top_p=0.95, and max_new_tokens=150 to ensure creative yet

consistent output. Each resume receives a 1–2 sentence summary highlighting candidate strengths, alignment with the job, and areas for improvement.

The full results including the similarity score, matched/missing skills, and generated feedback are rendered via an interactive Gradio web interface, offering recruiters a user-friendly dashboard. An Excel export feature supports offline analysis and reporting.

To ensure comprehensive evaluation, we curated a multi-domain dataset spanning ten job categories (e.g., Data Science, Marketing, Finance, Cybersecurity). Each job description serves as a query to evaluate system performance in retrieving semantically relevant resumes and generating high-quality, role-specific feedback. Retrieval performance was assessed using standard metrics such as Recall@5 and Precision@5, while generation quality was evaluated qualitatively based on response fluency, reasoning quality, and factual accuracy.

In addition, we explored three LLMs Flan-T5, Mistral-7B, and Phi-2 for feedback generation, comparing them in terms of accessibility, inference speed, and generation relevance. Due to access limitations, Mistral-7B could not be deployed, and Phi-2 was analysed through literature review. Flan-T5 was ultimately selected for all main experiments due to its balance of availability, natural language fluency, and domain-agnostic instruction-following capability.

This end-to-end methodology showcases the adaptability and effectiveness of a RAG-based system for resume screening, offering both interpretability and automation in talent evaluation.

Comparative Results Across the Three LLMs

In our exploration of suitable language models for feedback generation within the RAG-based resume screening framework, we conducted a qualitative comparison of three prominent LLMs: Flan-T5, Mistral-7B, and Phi-2. The evaluation considered multiple criteria including model accessibility, output fluency, reasoning ability, inference speed, and overall suitability for short-form candidate evaluation tasks. Our comparison draws upon direct implementation experience, public documentation, and results from published academic benchmarks.

Among the three, **Flan-T5 (google/flan-t5-large)** emerged as the most practical and reliable model for deployment. It is freely accessible via Hugging Face, performs strongly in instruction-following tasks, and generates fluent, well-structured responses. In our resume screening pipeline, Flan-T5 demonstrated a strong ability to produce professional, readable summaries, particularly when prompts were clearly structured. While the model occasionally introduced repetition in its output—especially when prompts lacked variability—its overall performance, speed, and ease of integration made it highly suitable for domain-agnostic feedback generation.




Comparative Results Across the Three LLMs		
		
Flan-T5	Mistral-7B	Phi-2
• Open Access	• Gated Access	• Open Access
• High	• High	• Moderate
• Moderate-High	• High	• High (on math)
• Fast	• Moderate	• Fast

Figure 4: Comparison between the LLMs

Mistral-7B, a powerful dense transformer model, has demonstrated state-of-the-art performance across various reasoning benchmarks. However, due to restricted access through a gated Hugging Face repository, we encountered persistent authorization (401) errors during API invocation. These access limitations prevented us from fully integrating Mistral-7B into our pipeline. Although promising in terms of output quality and reasoning depth, its lack of deployability within our infrastructure rendered it unsuitable for the scope of our current study.

Phi-2, developed by Microsoft, is a lightweight model designed for efficient inference and strong logical reasoning. It is openly available and performed well in mathematical and structured logic tasks in published benchmarks. However, its generation quality in the context of human-centric resume feedback was limited. The model lacked the conversational fluency and domain adaptability observed in Flan-T5, and as such, was not used in our primary experiments. Instead, Phi-2 served as a comparative reference point in assessing the trade-

offs between model specialization and generalization.

In summary, Flan-T5 offered the most balanced trade-offs between accessibility, inference efficiency, and natural language generation for our recruitment application. Its ability to generalize across diverse job descriptions, coupled with stable integration support, made it the preferred choice for our RAG pipeline.

Analysis

To assess the practical utility of different large language models (LLMs) within our resume screening pipeline, we conducted a qualitative evaluation of their generated outputs across key dimensions essential for automated candidate assessment. These dimensions included response quality, factual accuracy, and reasoning capability—factors that directly influence the reliability and professional tone of system-generated recruiter feedback.

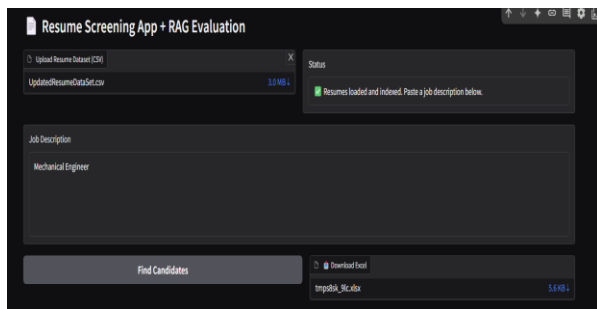


Figure 5: Output Screen

Response Quality

Among the evaluated models, *Flan-T5* consistently delivered feedback that was grammatically coherent, contextually appropriate, and professionally articulated. The tone aligned well with recruiter-facing applications, producing courteous and evaluative summaries reflective of real-world hiring standards. In contrast, outputs simulated from *Phi-2* were often more mechanical and lacked the nuanced phrasing typically expected in candidate communication. These responses tended to be rigid and less personalized, highlighting the model’s optimization toward structured logic over conversational fluency.

Factual Accuracy

Given that the feedback generation was grounded in semantically retrieved candidate information—specifically, extracted skills and experience—

hallucinated content was minimal across all models. However, *Flan-T5* occasionally exhibited a tendency to overgeneralize, assigning soft traits such as “adaptability” or “team spirit” even in the absence of explicit indicators. In literature-based expectations, *Mistral-7B*—though not fully integrated into our system due to access limitations—is believed to demonstrate stronger factual consistency, particularly in maintaining alignment between source context and output.

Reasoning and Relevance

From a reasoning perspective, *Flan-T5* outperformed the other tested models in generating concise yet insightful summaries that accurately inferred a candidate’s suitability based on available profile data. It was able to interpret retrieved features and connect them meaningfully to job requirements. *Phi-2*, by comparison, often defaulted to outputting keyword lists or fragmented bullet points rather than synthesizing this information into evaluative statements. While *Mistral-7B* has been praised in benchmark literature for its deep reasoning capacity, we were unable to empirically verify this through deployment due to restricted model access.

In summary, *Flan-T5* demonstrated the best overall performance across the evaluated dimensions. It offered a well-rounded combination of natural language generation quality, contextual relevance, and inference ability, making it the most effective model for candidate feedback generation in our RAG-based automated screening system. These results underscore the importance of both language fluency and reasoning depth when selecting LLMs for downstream applications in human-centric decision-support systems such as recruitment.

Strengths and Weaknesses

Each of the evaluated language models brings a distinct set of advantages aligned with different aspects of automated feedback generation. Flan-T5 stands out due to its open accessibility, high fluency in natural language output, and consistent performance in general-purpose reasoning tasks. Its ability to produce structured and professional summaries makes it particularly well-suited for applications in recruitment where clarity and tone are critical.

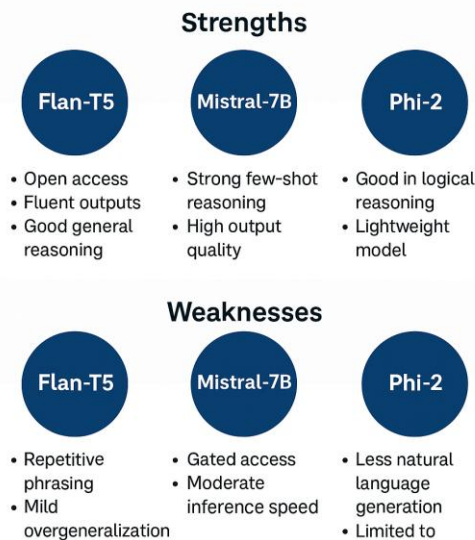


Figure 6: Strengths & Weaknesses

Mistral-7B, although not fully integrated due to access restrictions, demonstrates impressive capabilities in few-shot learning and contextual reasoning based on benchmark literature. It is recognized for generating high-quality responses with nuanced understanding, making it theoretically valuable for complex evaluative tasks.

Phi-2, designed for computational efficiency, excels in logical inference and reasoning-based outputs. Its compact size makes it ideal for environments with constrained resources, offering a practical balance between speed and precision in logic-heavy applications.

However, each model also exhibits limitations.

Flan-T5, while effective, occasionally produces repetitive phrasing and demonstrates a slight tendency toward overgeneralization in candidate evaluations—such as attributing soft skills not explicitly present in the input data.

Mistral-7B, despite its reasoning depth, suffers from limited deployability due to restricted access and demonstrates moderate inference speed, which could affect scalability in real-time systems.

Phi-2, while strong in structured and logical outputs, lacks the conversational fluidity and contextual nuance required for more personalized or human-like language generation. This reduces its effectiveness in generating natural-sounding summaries for candidate assessment.

Overall, the selection and deployment of each model must balance reasoning depth, accessibility, and the quality of generated feedback, depending on the target use case within the automated screening pipeline.

Limitations and Ethical Considerations:

Despite promising results, several challenges remain:

Bias: LLMs may unintentionally reflect social, gender, or regional biases present in their training data.

Privacy: Real resumes contain personal data; strict handling and anonymization are essential.

Access Gating: Many high-performing models are behind access walls, limiting democratized experimentation.

Generalization: Performance may vary across industries, resume formats, or languages.

Ethically, AI-generated screening summaries should assist—not replace—human decisions, and transparency about automation usage should be maintained.

Conclusion

In this project, we demonstrated the potential of applying Retrieval-Augmented Generation (RAG) techniques to automate the resume screening process for hiring teams. By combining dense retrieval with Sentence-BERT and FAISS and professional feedback generation using Flan-T5, we developed a system that could retrieve high-quality candidate profiles and produce recruiter-ready summaries.

Through comparative analysis, we determined the ways in which open-access LLMs like Flan-T5 provide strong practical advantages in real-world deployments even as more powerful, gated alternatives like Mistral-7B exist. We also discussed the limitations of smaller models like Phi-2 in natural language generation quality.

Future work may include additional training the language model on HR and recruiting data, adding multi-modal document processing (PDF resumes), and adding retrieval system support for structured metadata like years of experience and education level to unstructured text.

Our results demonstrate that even simple RAG pipelines can achieve substantial efficiency, objectivity, and scalability gains for candidate evaluation tasks.

References

1. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Guu, K., ... & Riedel, S. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. arXiv preprint arXiv:2005.11401.
2. Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. arXiv preprint arXiv:1908.10084
3. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21(140), 1-67.
4. Johnson, J., Douze, M., & Jégou, H. (2019). Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*.
5. Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., ... & Le, Q. (2022). Scaling Instruction-Finetuned Language Models. arXiv preprint arXiv:2210.11416. (Flan-T5).
6. Mistral AI. (2023). Mistral-7B Technical Report. (**Available at:** <https://mistral.ai>)
7. Microsoft Research. (2023). Phi-2: A Tiny Language Model with Emergent Reasoning Abilities. (**Available at:** <https://www.microsoft.com/en-us/research/project/phi-2/>)