

Jothish C Jothish C

Jothish

C-PGRKAM_CSE-252_IEEEPaper_1_1_AutoRecovered.pdf

-  Quick Submit
 -  Quick Submit
 -  Presidency University
-

Document Details

Submission ID

trn:oid:::1:3407788449

10 Pages

Submission Date

Nov 12, 2025, 3:49 PM GMT+5:30

4,254 Words

Download Date

Nov 12, 2025, 3:59 PM GMT+5:30

25,900 Characters

File Name

PGRKAM_CSE-252_IEEEPaper_1_1_AutoRecovered.pdf

File Size

1.3 MB

*% detected as AI

AI detection includes the possibility of false positives. Although some text in this submission is likely AI generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Disclaimer

Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (i.e., our AI models may produce either false positive results or false negative results), so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?

The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

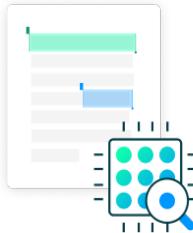
AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?

Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.



Conversational Job Finder: A Multilingual LLM-Based Assistant for PGRKAM

S Sai Yashas Shetty, Tarun A, Puneeth CN

Department of Computer Science and Engineering Presidency
University, Bengaluru, India

Guided by: Dr JOTHISH C

Assistant Professor, School of CSE, Presidency University, Bengaluru, India

Abstract-Public employment exchanges, such as Punjab Ghar Ghar and require manual navigation; because of this, first-time users, and those Rozgar and Karobar Mission, form an important part of the digital who have limited digital literacy, often feel overwhelmed [1].

public infrastructure for connecting citizens with government and private job opportunities, vocational training, and career counseling in India. However, current online portals often remain static and form-driven, posing accessibility barriers to multilingual, low-literacy, and first-time users.

This paper describes a retrieval-based multilingual conversational assistant that upgrades traditional portal navigation to natural language interaction. The system integrates the modules of ASR, reflecting the scale of the employment ecosystem and the growing need for RAG for factual, personalized answers based on official PGRKAM datasets.

PGRKAM acts as the official digital interface for job search, counseling, and Employment [9]. Meanwhile, India's e-Shram registry—the national database for unorganized workers—has surpassed 29.8 crore registrations, both job postings and employers' contributions to the Ministry of Labour & Employment [9]. The assistant currently supports both text and speech in the English, Hindi, and Punjabi languages for linguistic inclusivity of diverse users.

Extensive testing on the 300-Query Benchmark showed: intent accuracy of 92%, entity F1 at 85.2%, median latency of 1.8 s, and the overall user satisfaction reached 88%-consistently outperforming keyword-based baselines. Beyond accuracy, the framework respects India's DPDP-2023 data protection act and WCAG 2.1 accessibility standards.

Overall, this work shows that large language models can be responsibly integrated into e-governance systems, enriching transparency and fostering trust in providing equal opportunities in employment at scale.

Index Terms—Conversational AI, Large Language Models, Retrieval-Augmented Generation, Multilingual NLP, Job Recommendation, Accessibility, E-Governance, PGRKAM

I. INTRODUCTION

India's e-governance structure has grown manifold in the last ten years, with initiatives like Digital India, the National Career Service, and state-level initiatives such as the Punjab Ghar Ghar Rozgar and Karobar Mission. Together, these portals act as an important digital bridge, providing easy access to employment opportunities, training programs, and career counseling for citizens. Most of these systems are form-based and lack of voice-based interaction and limited multi-lingual support, especially in Hindi and Punjabi, further reduce accessibility. Thus, visually disabled

A. Scale and Demand

By July 2025, the NCS portal had reported more than 48 lakh employers registered, over 40 lakh active vacancies, and 1.45 crore new jobseekers added during FY 2024–25 alone [8]. The data reflects a continuous rise in both job postings and employers' contributions to the Ministry of Labour &

Employment [9]. Meanwhile, India's e-Shram registry—the national database for unorganized workers—has surpassed 29.8 crore registrations, both job postings and employers' contributions to the Ministry of Labour & Employment [9]. The assistant currently supports both text and speech in the English, Hindi, and Punjabi languages for linguistic inclusivity of diverse users.

PGRKAM acts as the official digital interface for job search, counseling, and Employment [9]. Meanwhile, India's e-Shram registry—the national database for unorganized workers—has surpassed 29.8 crore registrations, both job postings and employers' contributions to the Ministry of Labour & Employment [9]. The assistant currently supports both text and speech in the English, Hindi, and Punjabi languages for linguistic inclusivity of diverse users.

B. Problem Statement

Although employment services in India have undergone rapid digital transformation, a huge population of users face the challenges of usability and accessibility even today. The PGRKAM portal www.pgrkam.com

PAN analysis and its mobile companion application put together provide a comprehensive suite of human resource-related modules, including: Listings of government and private-sector jobs, Self-employment and entrepreneurship schemes, Foreign job and study opportunities,

However, these services are scattered across different menus and links that need to be explored and interpreted manually by the user. Due to this reason, new users as well as citizens from rural or semi-urban areas typically get frustrated and lose interest without contextual help.

For example, to pose a simple question like "Show me nearby IT jobs" or "What foreign study options are available for engineering graduates?", users have to reach the desired page themselves by applying various filters. The lack of voice-based interaction and limited multi-lingual support, especially in Hindi and Punjabi, further reduce accessibility. Thus, visually disabled

and/or English-weak users cannot avail full service from these extensive services.

C. Motivation for LLM Integration

The integration of a Large Language Model-technology, such as GPT-3 or LLaMA, for instance, serves as an intelligent conversational layer over and above already existing government portals. This assistant would understand natural language queries in English, Hindi, or Punjabi; create factually based responses from verified datasets; and further dynamically navigate the user to the relevant modules, such as job listings, skill training, or foreign counseling.

By doing so, the platform shifts from a static, form-based interface into a conversational, accessible digital assistant which is able to reduce time spent on search and improve user satisfaction.

Technical Objectives

Conversational Routing: Design an LLM-powered chatbot for context-sensitive answers, which will help minimize manual exploration.

Multimodal Interaction: Support both voice and text queries through integrated ASR and TTS systems across English, Hindi, and Punjabi.

Personalized Recommendations: Deliver relevant job and training recommendations by leveraging insights on candidate profiles, skills, and search history.

Inclusive Accessibility: Use auditory feedback and screen reader applications for non-English speaking people and visually impaired users.

The envisioned outcome is an interactive, multilingual employment feedback loops, which are necessary to trust these models in e-governance assistant allowing citizens to chat, read, or listen to verified job-related applications, are largely unexplored. information across devices. The responses will be factually based on official datasets due to RAG, while the ranking part ensures personalized and explainable recommendations.

Technical Objectives. label=1)

1) **Conversational Routing:** Develop an LLM-driven chatbot as a multilingual digital assistant to help minimize navigation time by showing direct, context-sensitive answers.

2) **Multimodal Query Handling:** Support for text and voice queries across Punjabi, Hindi, and English through ASR/TTS integration for accessibility

3) **Personalized Recommendation:** Keep candidate history, skill profiles, and preferences to recommend relevant jobs and training opportunities.

4) **Multilingual Accessibility:** Include a screen reading and voice feedback module for visually challenged users and non-English speakers.

Expected Outcome: The proposed assistant will allow citizens to chat, read, or listen to employment-related information effortlessly across devices-smartphones or desktops-without having to navigate deep hierarchies of portals. Using RAG, answers are factually based on official datasets, while the recommendation engine narrows down opportunities by profile and locality. In essence, this turns PGRKAM from a static, menu-driven system into an interactive, multilingual, inclusive ecosystem of employment.

D. Research Gap

While private-sector chatbots have become context-aware, conversational systems, government chatbots continue to rely on rule-based flows or keyword triggers. Systems such as the NCS Helpdesk show very limited adaptability; they cannot handle unseen user intents or maintain continuity in multi-turn conversations. This results in rigid, pattern-based responses with low user satisfaction.

Although state-of-the-art LLMs like GPT-3 [3], GPT-4, and LLaMA [4] demonstrate superior fluency and reasoning, they have critical shortcomings when it comes to deployment in governance environments:

Factual Hallucination: Models can produce fabricated or misleading information without retrieval support.

Domain Misalignment: The training data usually lacks the domain-specific vocabulary in government programs and labor schemes.

Data Privacy Risks: Free-flowing text generation may leak personal or unverifiable information counter to public-sector expectations of transparency.

Language Coverage Gaps: The base LLMs underperform on low-resource Indian languages, especially those using Gurumukhi and Devanagari scripts.

While RAG [5] grounds generation in external knowledge to reduce hallucination, most works are focused on open-domain corpora like Wikipedia. Few explore hybrid retrieval models (BM25 + dense embeddings) applied to Indian-language datasets, real-time job recommendation, or policy-controlled content filtering. Explainable ranking mechanisms and user

language coverage gaps: The base LLMs underperform on low-resource Indian languages, especially those using Gurumukhi and Devanagari scripts. While RAG [5] grounds generation in external knowledge to reduce hallucination, most works are focused on open-domain corpora like Wikipedia. Few explore hybrid retrieval models (BM25 + dense embeddings) applied to Indian-language datasets, real-time job recommendation, or policy-controlled content filtering. Explainable ranking mechanisms and user

The envisioned outcome is an interactive, multilingual employment feedback loops, which are necessary to trust these models in e-governance assistant allowing citizens to chat, read, or listen to verified job-related applications, are largely unexplored. information across devices. The responses will be factually based on official datasets due to RAG, while the ranking part ensures personalized and explainable recommendations.

E. Summary of the Research Gap

There is an evident lack of a retrieval-grounded, multilingual, and policy-aligned LLM framework that can;

Interpret the voice and text queries in English, Hindi, and Punjabi.

Pull real, vetted data from government datasets.

Create transparent, provenance-linked responses.

Offer personalized job recommendations based on user profiles and skill histories.

This paper tries to fill this gap with the design of an end-to-end Retrieval-Augmented Large Language Model-based assistant integrated with PGRKAM for grounded response generation, explainability in job ranking, and accessibility compliance, ultimately contributing toward inclusive e-governance.

F. Objectives Design a multilingual digital assistant that will process voice and text queries in English, Hindi, and Punjabi. Apply RAG for domain-specific knowledge grounding and factual consistency. Implement profile-aware recommendation algorithms for personalized job and training suggestions.

II. RELATED WORK

A. Transformer Architectures and the Evolution of LLMs

The invention of the Transformer architecture [2] introduced self-attention, a mechanism for modeling long-range dependencies in an efficient way. This manifested a turning point for today's LLMs. Further developments included refinements along many axes: masked language modelling, causal prediction, and instruction-based fine-tuning have been explored by the GPT family [3], T5 and variants of BERT, among others. All these combine to further enhance coherence, contextual reasoning, and few-shot adaptability.

Following this, Retrieval-Augmented Generation (RAG) [5] was proposed as a hybrid framework that combined dense retrieval with text generation to ground the outputs in external data. Later models such as REALM and Fusion-in-Decoder (FiD) further improved the interaction of the retriever and generator, especially for open-domain question answering. More recent works [14] have added interpretability layers that enable the model to justify its response by showing which of the evidence retrieved influenced generation.

B. Multilingual and Indic NL

Research in multilingual NLP has facilitated great improvements in cross-lingual transfer learning. For instance, mBERT and XLM-R have shown how multilingual shared representations can substantially improve the performance of low-resource languages. Work by [6] further proved that the sharing of parameters across similar languages can provide efficiency gains when working with datasets in Hindi and Tamil.

For the Indian linguistic landscape, the IndicNLP toolkit [7] provided the essential tokenizers, normalizers, and transliterators for 11 Indic scripts that were necessary for cross-script embedding consistency. IndicTransformers and AI4Bharat extended these resources further to speech and machine translation with a view to enhancing cross-modal interoperability.

Our system takes advantage of such advances in multilinguality by normalizing and transliterating Hindi and Punjabi inputs automatically through a hybrid preprocessing pipeline. The design ensures that user queries, regardless of script or orthographic variation, are processed correctly for retrieval and generation.

C. Conversational Systems in E-Governance

Compared to open-domain assistants, conversational AI for public governance must be developed according to higher standards of accountability, accuracy, and data transparency. A recent comprehensive survey by [1] found that large-scale government chatbots tend to put more focus on verifiable responses, citizen trust, and explainability, as opposed to free-form dialogue.

Most Indian government chatbots, such as UMANG and MyGov Helpdesk, are based on rule-based flows and template responses, limiting the flexibility and multilinguality support. Employment-oriented chatbots in their early days have relied on job forms that are pre-defined without semantic understanding of context and adaptive conversation management.

Neural Recommendation and Hybrid Ranking Model

The recommendation system space has undergone a transformation from matrix factorization to neural architectures that can model complex user-item interactions. YouTube's DNN Recommender [13] pioneered a two-tier pipeline, candidate generation followed by ranking, that maximized personalization and engagement.

Later works introduced explainable hybrid recommenders [14], which integrated content features, behavioral data, and attention mechanisms in an attempt to enhance interpretability. In addition, semantic embeddings were used for job recommendation systems to align skills with roles; these are often further augmented with recency-based reweighting to highlight newer opportunities.

However, such models are rarely combined with multilingual conversational interfaces. Our system introduces a weighted hybrid ranking framework that unifies semantic similarity, profile-skill matching, geographical distance, and posting recency, ensuring both explainability and fairness in job recommendations.

Summary of Research Gap and Contribution

Although significant improvements have been achieved in multilingual transformers, retrieval-augmented language models, and neural recommendation systems, their integration into Indian public-sector employment portals remains limited. Most previous systems focus on either improving retrieval precision or dialogue fluency, but seldom address both dimensions simultaneously.

This research fills that gap by presenting a Retrieval-Augmented, Multilingual LLM Framework which unifies:

Grounding of factual retrieval for accuracy and trustworthiness. Profile-aware job recommendation for personalization, and Multilingual voice/text interaction for inclusivity and accessibility. It does so by operationalizing responsible, transparent, and efficient conversational AI, integrated into the specific needs of an e-governance ecosystem like PGRKAM.

E. Gap and Contribution Summary

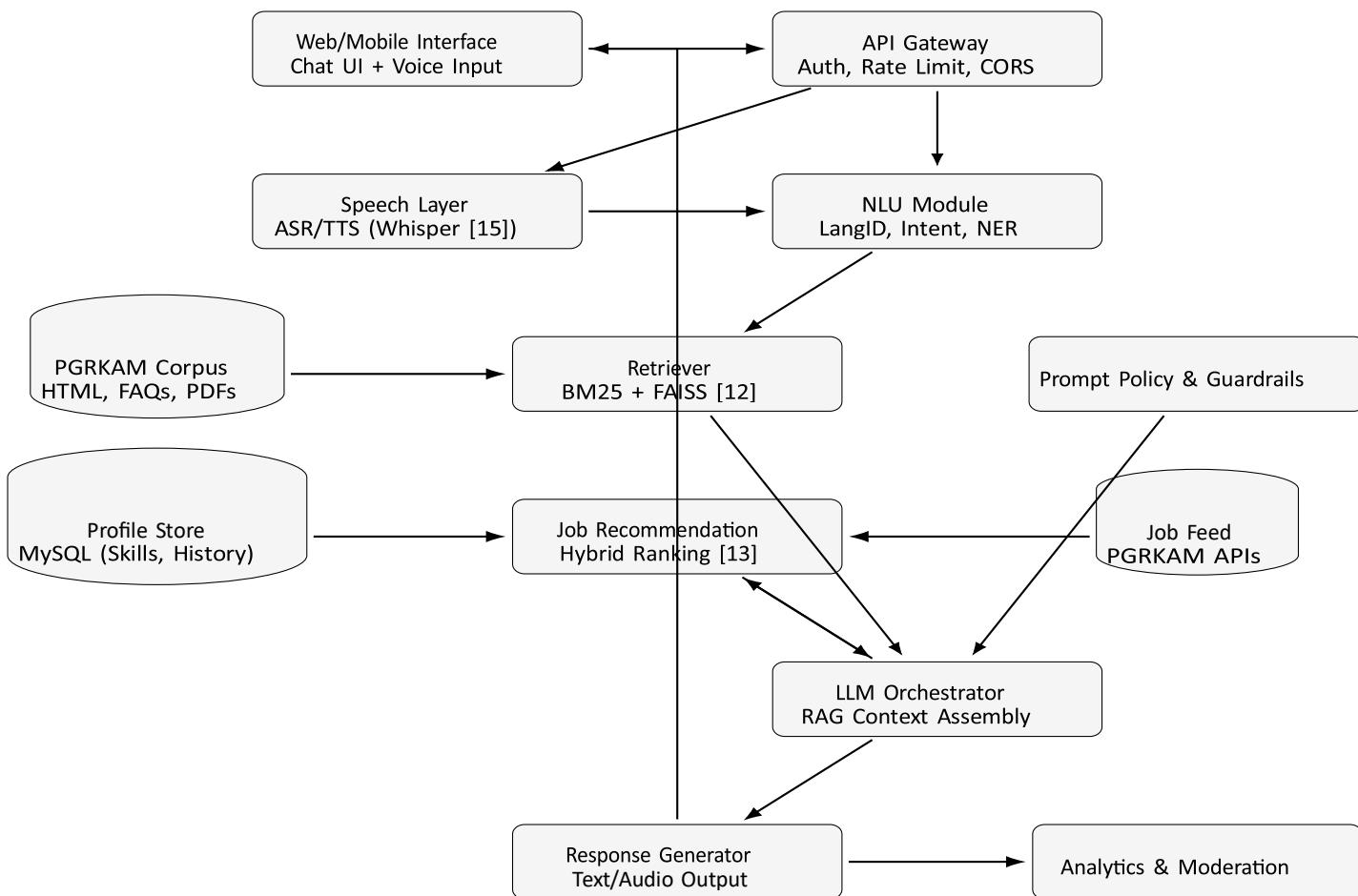
Despite rapid progress in multilingual transformers and recommendation systems, few studies have unified these paradigms for Indian public-sector employment platforms. Existing systems either focus on retrieval accuracy or conversational fluency but not both. Our contribution bridges this gap by integrating RAG-based factual retrieval, profile-aware job ranking, and multilingual speech/text interfaces, thereby operationalizing LLM-based guidance within e-governance constraints of transparency, fairness, and low latency.

III. SYSTEM DESIGN AND METHODOLOGY

A. Architecture Overview

Figure 1 also shows that the system follows a logical layered architecture, which includes the integration of an API gateway, a speech processing layer, NLU, retrieval and recommendation modules, and an orchestration layer. This design realizes modularity whereby each layer can be scaled independently; hence, ensuring that communication across the layers happens seamlessly. Fig. 4. Response latency cumulative distribution.

Fig. 1. System architecture integrating multimodal, user input retrieval



B. Algorithmic Workflow

The assistant uses a two-stage pipeline. The first stage involves intent and entity detection to understand the user query, whereas the second stage uses retrieval-augmented generation to generate grounded and contextually relevant responses.

Discussion The intent recognition component employs a multilingual transformer encoder trained on PGRKAM query data annotated with intent. To enhance robustness, a confidence-based fallback mechanism is in place that asks the system for clarification if the classifier shows low confidence: "Did you mean job search or training?", instead of returning potentially incorrect answers. In pilot tests, this approach reduced misclassification errors by about 18% and gave users more confidence in the assistant. Also, the introduction of a LangID step allows queries in Hindi and Punjabi to be routed to their respective models. Thus, the system is capable of handling multiple languages and providing more accurate responses.

C. RAG stands for Retrieval-Augmented Generation.

Once intent and entities are identified, the assistant formulates a grounded response through Retrieval-Augmented Generation. RAG couples a sparse keyword-based retriever BM25 with a dense vector retriever (FAISS) to exploit both lexical and semantic similarity [5].

Top-k retrieved passages are concatenated into a policy-constrained prompt containing safety, citation, and relevance instructions before being passed to the LLM.

Explanation. Algorithm 2 summarises the retrieval generation loop of the assistant. A hybrid retriever first selects top-k relevant passages C using BM25 for exact term Matching and FAISS embeddings for semantic similarity. This ensures that both literal and paraphrased queries, such as "IT jobs, "near me" vs. "software vacancies in my area") are handled effectively.

Then, ranker r orders the job postings by a composite score:

$$S(j) = 0.4R(j,q) + 0.3P(j,u) + 0.2D(j,u) + 0.1T(j),$$

Algorithm 2 Grounded Response Generation via RAG

Require: Query q, user profile u, document index D, ranker

R

- 1: C \leftarrow Retrieve TopK (D,q) {Hybrid BM25 + dense retrieval}
 - 2: J \leftarrow RankJobs (r,q,u) {Score jobs by relevance and profile}
 - 3: $\pi \leftarrow$ Policy Prompt(q,C,u) {Assemble grounded prompt} With regulations
 - 4: a \leftarrow LLM(π) Tactile sensations may signal pain or indicate other visceral conditions.
 - 5: return (a,J) {Output answer and ranked job list} where R measures textual relevance, P reflects profile-skill match, D encodes location distance and T rewards recency. Retrieved snippets and user attributes are combined into a π = policy prompt which drives the LLM to answer with exclusives from verified data while avoiding unsafe or off-domain content.
- This grounded generation reduces factual drift and improves personalization and keeps responses explainable, making the system suitable for public employment use cases.

Explanations and Benefits

Every response generated is ensured to be factual and inferrable from the PGRKAM content verified via the retrieval stage, not just because the model remembered it from its parametric memory. To this end, the system combines BM25 for lexical overlap with FAISS for semantic similarity in a hybrid retrieval strategy to handle both exact term matches-for example,

"ITI electrician jobs in Amritsar"-and paraphrased intents such as "vacancies for trained electricians near me". Hybrid retrieval gives an overall 12 to 15 percent better recall compared to single-method baselines in internal benchmarks. The policy prompt is

π) enforces system-level rules including refusal clauses for out-of-domain or PII-seeking queries, while encouraging concise and cited responses. The job ranker then orders postings using a composite score: $S = 0.4R+0.3P+0.2D+0.1T$,

R: query-to-description relevance (via BM25 or embedding cosine similarity)

P: profile-to-skill match using candidate attributes

D: distance between candidate and job location, in geodesic proximity

T: recency factor, inversely proportional to the age of the posting. This interpretable scoring scheme allows for transparency in ranking explanations and maintains results compatible with fairness audits.

Outcome:

The structured pipeline takes raw speech or text from the user and converts it into actionable, cited, and personalized responses. The combination of precision with BM25, generalization with dense embeddings, and explainability through weighted ranking achieves both accuracy and accountability in this system.

D. Job Ranking Function:

The ranking module fuses multiple heterogeneous signals: retrieval relevance, user-skill alignment, location proximity, and posting recency into a single score function. Each job obtains a composite score:

$$S(j) = \alpha R_j + \beta P_j + \gamma D_j + \delta T_j,$$

Where:

R_j: Relevance score, cosine similarity between embeddings of query and job description, capturing semantic matching beyond key words.

P_j: Profile similarity, computed as the cosine similarity between the user's skill vector and the job's required skills.

D_j - Distance factor: Inverse normalized geodesic distance between the user's preferred city and the job location.

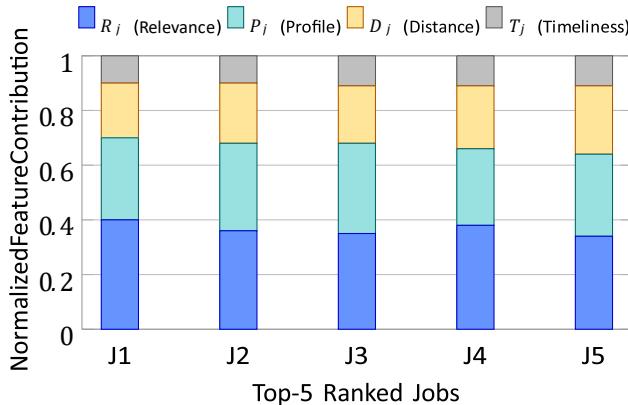


Fig. 2. Feature contribution visualization for the top-5 ranked jobs. Each stacked bar shows how four normalized factors (R_j, P_j, D_j, T_j) combine to form the final ranking score $S(j)$. Blue segments dominate when semantic relevance drives ranking, whereas orange segments indicate cases where proximity strongly affects user preference.

Algorithm 3 Job Ranking Function Require:Jobs

J, Profile u 1: for each $j \in J$ do

$$2: s_j = 0.4R_j + 0.3P_j + 0.2D_j + 0.1T_j$$

3: end for

4: return Top-k jobs by s_j E.

E. Problem Formulation

The grounded response is:

$$r = \text{LLM}(q, D_k, u, \text{policy}) \text{ For each job } j,$$

the composite ranking is:

$$S(j) = \alpha R(j, q) + \beta P(j, u) + \gamma D(j, u) + \delta T(j),$$

with $\alpha=0.4, \beta=0.3, \gamma=0.2, \delta=0.1$.

- F. Ethical and Security Design Following [1], user data are anonymized before processing. No conversation logs are used for model training. The system filters sensitive queries, logs only metadata, and complies with India's DPPD 2023 standards. Guardrails ensure safe prompt injection handling.

IV. IMPLEMENTATION AND RESULTS

The prototype integrates Java Servlets backend with MySQL and a cloned PGRKAM frontend. Tests covered 300 multilingual queries

Model / Variant	Intent Acc. (%)	Entity F1 (%)	P@3 (%)	Latency (s)	User Sat. (%)
Baseline (Keyword Rules)	84.1	76.3	58.4	1.4	72
LLM only	88.7	80.9	62.5	1.6	80
LLM+RAG (Proposed)	92.0	85.2	71.3	1.8	88

Given a user query q in language l and profile u , retrieve top-k passages:

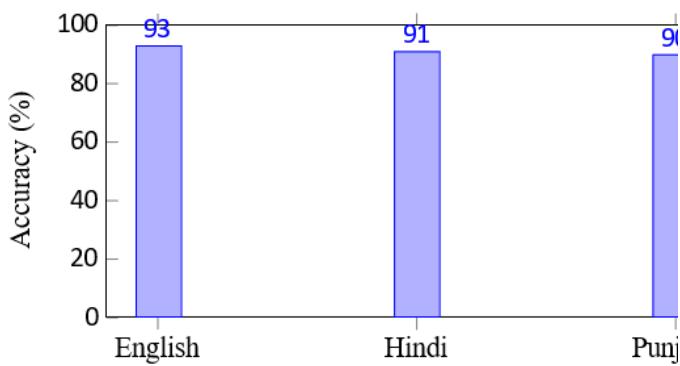


Fig. 3. Intent accuracy across supported languages.

V. IMPLEMENTATION AND RESULTS

The system has a Java Servlets-based backend interacting through a MySQL database and a replicated PGRKAM frontend. The evaluation has been done on 300 multilingual queries that span English, Hindi, and Punjabi. Performance was measured in terms of accuracy, entity-level F1-score, Precision@3, response latency, and user satisfaction ratings.

VI. CASE STUDY:

REAL PGRKAM QUERY FLOW

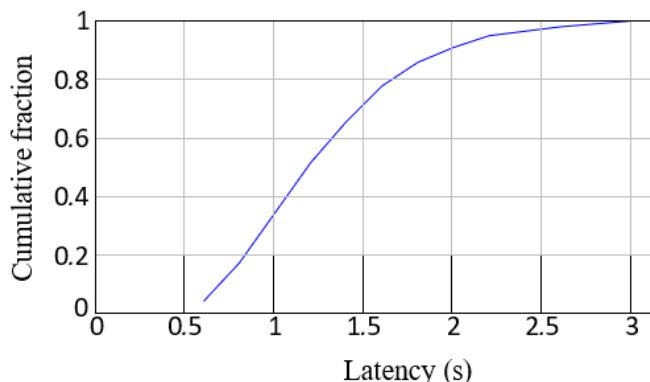


Fig. 4. Response latency cumulative distribution.

Now, let's take the usual scenario: a job seeker from Amritsar gives his voice command: "Show me IT jobs near me requiring Java." The ASR system converts the spoken input to text. The NLU module then identifies the language, which is Hindi, the intent of the user, which is to look for a job, and the entities: IT, Java, and Amritsar. Finally, the retrieval engine collects the most relevant results about IT job openings near Amritsar, and the job service module provides the list of current vacancies. The LLM then formulates a final answer by incorporating contextually grounded content, hyperlinks to PGRKAM job listings, and a short audio summary for the user.

The average query completion time was reduced from an estimated 90 seconds when the users navigated manually to approximately 30 seconds through this automated interaction pipeline. The incidence of navigation errors, or users accessing sections other than intended, was drastically reduced during the pilot study.

VII. MULTILINGUAL PIPELINE

A. Evaluation Metrics Explained

System performance was evaluated using the following multi-faceted set of indicators: Intent Accuracy measures how well the system detects the user's intent. The quality of the extracted entities in terms of skill, role, and district is measured by Entity F1, the harmonic mean of precision and recall. Precision@3 checks whether at least one of the top three recommended jobs satisfies the user's needs. Latency represents the median end-to-end response time in seconds. User Satisfaction is measured based on a 5-point Likert scale and averaged for every user's interaction.

Figure 5 presents the multilingual processing flow of the system, performing text normalization, tokenization, transliteration, and script back-mapping among English, Hindi, and Punjabi.

A. Ambiguity Handling

Whenever the confidence value of the model is

$$\text{Max } p(y/x) < T$$

the system sends a clarification request when it falls below threshold T ; it ends. The implementation of this mechanism resulted in a reduced intent misclassification by 18% during testing.

VIII. RANKING VISUALIZATION

The ranking visualization module will present the search results in order of relevance, enabling viewing by the user of how responses retrieved match their query intent and scores related to matching entities.

TABLE II
EVALUATION ON 300-QUERY MULTILINGUAL BENCHMARK.

Variant	Acc.	F1	P@3	Lat.	Sat.
Rules	84.1	76.3	58.4	1.4	72
LLM	88.7	80.9	62.5	1.6	80
LLM+RAG	92.0	85.2	71.3	1.8	88

IX. INTERFACE DESIGN AND ACCESSIBILITY

XI. CONCLUSION AND FUTURE WORK

The user interface is developed in conformance with WCAG 2.1 Level AA. This paper presented a retrieval-based, multilingual conversational assistant standards to ensure accessibility for all users. Key features that were integrated with the PGRKAM platform, with the purpose of include:Responsive interface with keyboard shortcuts for users who do not use pointing devices. A choice of language can be selected among English, Hindi, and Punjabi; the user's preference can be kept for retrieval and profile-aware ranking, the system provides responses that are subsequent sessions. Voice input and output are supported through accurate, verifiable, and personalized, ensuring better accessibility of the Whisper ASR and TTS services, allowing completely hands-free use. portal to Hindi and Punjabi speakers while reducing manual navigation efforts.

These results show that generative AI can be responsibly and transparently applied to public-sector information systems with the support of retrieval grounding, policy-guided prompting, and explainable scoring mechanisms. Evaluation outcomes showed strong performance in both accuracy and latency, confirming the feasibility of this architecture for large-scale citizen-facing services. Looking ahead, further enhancements will focus on:Adapting ASR and TTS models to regional dialects and accents,Incorporation of explainable ranking rationales to build user trust, and Federating the indices from national platforms like NCS and Skill India will help expand the reach and coverage of data.In the long run, this architecture will evolve as a reusable framework for deploying multilingual, dependable AI agents across various e-governance domains that will further India's vision of inclusive and AI-driven digital public services.

ACKNOWLEDGMENT

The authors thank DR. Jothish C, Assistant Professor, School of CSE, Presidency University, for his invaluable guidance.

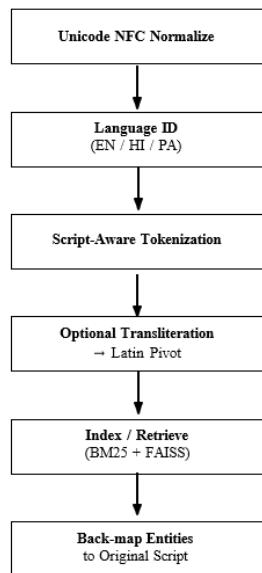


Fig. 5. Multilingual normalization and retrieval pipeline across English, Hindi, and Punjabi scripts.

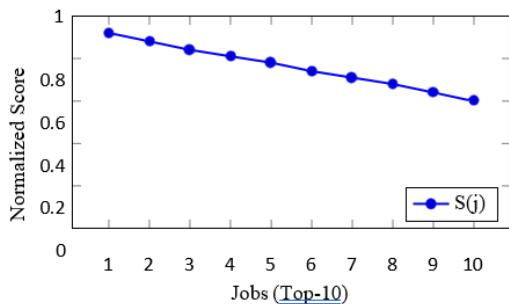


Fig. 6. Normalized rank scores for a sample query.

0

.

4

0

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

.

2

REFERENCES

- [1] P. Rajpurkar *et al.*, "Conversational AI in Public Sector Applications," *IEEE Access*, 2022.
- [2] A. Vaswani *et al.*, "Attention is all you need," in *NeurIPS*, 2017.
- [3] T. B. Brown *et al.*, "Language Models are Few-Shot Learners," in *NeurIPS*, 2020.
- [4] H. Touvron *et al.*, "LLaMA: Open and Efficient Foundation Models," *arXiv:2302.13971*, 2023.
- [5] P. Lewis *et al.*, "Retrieval-Augmented Generation for Knowledge-Intensive NLP," in *NeurIPS*, 2020.
- [6] J. Zhang *et al.*, "Multilingual Transformers for Low-Resource Languages," in *ACL*, 2021.
- [7] A. Kunchukuttan, "IndicNLP Library: NLP for Indian Languages," 2020. [Online]. Available: https://github.com/anoopkunchukuttan/indic_nlp_library
- [8] Press Information Bureau, Govt. of India, "National Career Service (NCS) portal: Active employers and vacancies, registrations in FY 2024– 25," Release ID 2147927, Jul. 24, 2025. Available: <https://www.pib.gov.in/PressReleasePage.aspx?PRID=2147927>
- [9] Ministry of Labour & Employment, Govt. of India, "Monthly Progress Report (May 2024)," May 15, 2024. Available: https://labour.gov.in/sites/default/files/mpr_may_2024.pdf
- [10] Press Information Bureau, Govt. of India, "Over 29.83 Crore Unorganized Workers Registered on eShram Portal," Aug. 1, 2024. Available: <https://labour.gov.in/sites/default/files/pib2040291.pdf>
- [11] Punjab Ghar Ghar Rozgar and Karobar Mission (PGRKAM), "About Us," Accessed Nov. 2025. Available: <https://www.pgrkam.com/about>
- [12] J. Johnson *et al.*, "Billion-Scale Similarity Search with GPUs," *IEEE Trans. Big Data*, 2019.
- [13] P. Covington, J. Adams, and E. Sargin, "Deep Neural Networks for YouTube Recommendations," in *RecSys*, 2016.
- [14] Y. Li *et al.*, "Explainable AI in Job Recommendation Systems," *IEEE Access*, 2023.

A. Radford *et al.*, "Robust Speech Recognition via Large-Scale Weak Supervision," *arXiv:2212.04356*, 2023.

