



Airtel RAG Customer Support Chatbot

Retrieval-Augmented Generation for
Telecom Customer Service



LLM + SLM Hybrid



Groq LPU



Brand Voice

Project Overview



Use Case

Customer Support Assistant for Airtel



Company

Bharti Airtel Limited

India's Leading Telecom Company



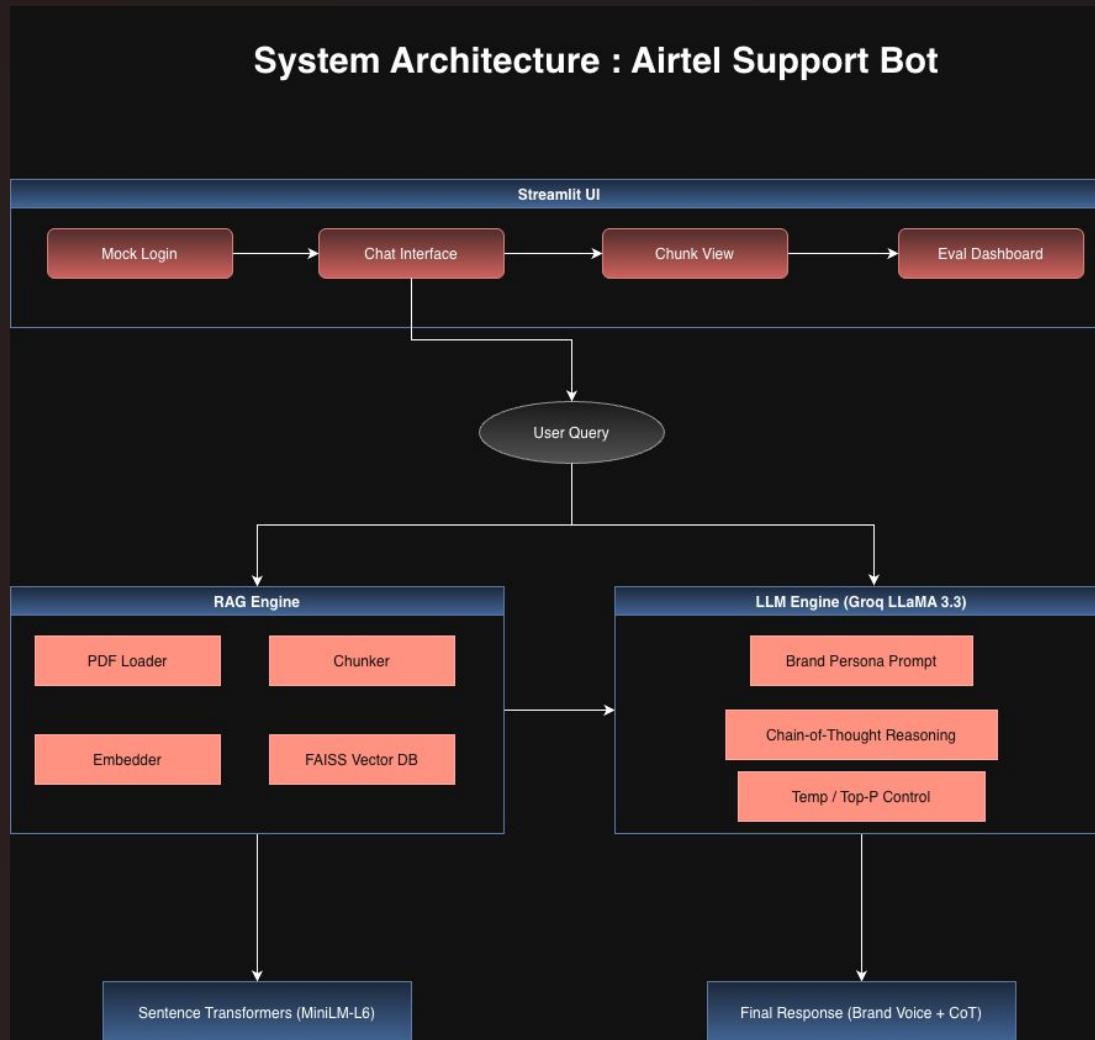
Goal

Build a RAG-based chatbot that answers customer queries about Airtel's **plans, policies, billing, and services** using only verified company documentation

The screenshot shows a Streamlit-based AI assistant interface. At the top left is a sidebar for an 'Admin User' with a 'Logout' button. The main area has a red header bar with the 'Airtel Customer Support' logo and the text 'RAG-Powered AI Assistant — Ask about plans, policies, billing & more'. Below the header are tabs for 'Chat', 'Retrieved Chunks', 'Evaluation', and 'Temperature Comparison', with 'Chat' being the active tab. A text input field says 'Ask about Airtel plans, billing, policies...'. To the right of the input field is a 'Deploy' button. On the left side of the main area, there are sections for 'Generation Parameters' (with sliders for Temperature, Top-P (Nucleus Sampling), Chain-of-Thought, and Retrieved Chunks (top-k)) and 'RAG Statistics' (showing 122 Chunks, 122 Vectors, 500 Chunk Size, and 100 Overlap).

Streamlit-based Chat Interface

System Architecture



STREAMLIT UI

Mock Login → Chat Interface → Chunk View → Eval Dashboard

RAG ENGINE

PDF Loader → Chunker → Embedder → FAISS DB

LLM ENGINE

Groq LLaMA 3.3 → Brand Persona → CoT Reasoning

EMBEDDINGS

sentence-transformers (MiniLM-L6-v2)

Response Generation: Combines retrieved context with brand voice and CoT reasoning

Technology Stack



LLM

Groq (LLaMA 3.3 70B Versatile)

Ultra-fast inference via Groq's LPU, excellent instruction-following



Embeddings

all-MiniLM-L6-v2

Lightweight (80MB), fast, 384-dim. Ideal for semantic search without GPU



Vector DB

FAISS (Facebook AI)

Fast similarity search, no server needed, persistent storage



Chunking

LangChain RecursiveCharacterTextSplitter

500-char chunks with 100-char overlap for optimal retrieval



UI

Streamlit

Rapid prototyping, built-in chat components



Authentication

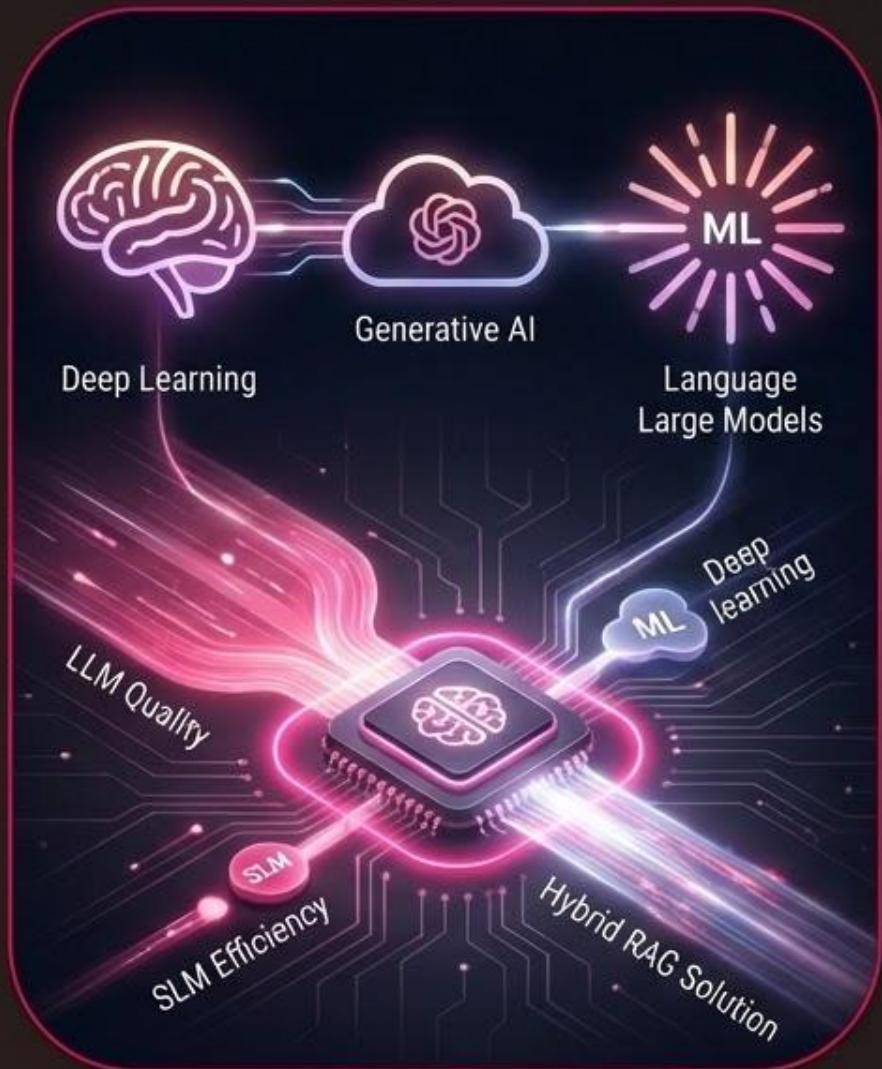
Custom Mock Login (SHA-256)

Simple security gate as required



Hybrid approach: Cloud LLM for quality + Local SLM for efficiency

Hybrid Model Selection



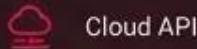
LLaMA 3.3

Used For: Response generation

Why:

Superior brand-voice, CoT reasoning

Size:



Cloud API

Latency: 0.5-2s (LPU)

Cost: Free tier available



MinILM-L6-v2

Used For: Document embeddings



Fast lightweight, no GPU needed



80MB local model

Latency: ~50ms per embedding

Cost: Free (local)



Why Hybrid Approach?

Pure SLM struggles with brand voice adherence, CoT reasoning, multi-turn conversations, and detailed structured responses. **Hybrid gives SLM efficiency + LLM quality**



Best of Both Worlds



Efficient Retrieval

Key Features



RAG Pipeline

PDF loading → Chunking →
Embedding → FAISS → Retrieval →
Generation



Brand Voice

Strict "Airtel Assist" persona using
only provided context



Chain-of-Thought

Model explains retrieval logic before
answering



Parameter Controls

Live sliders for Temperature and
Top-P generation parameters



Chunks Display

Dedicated tab showing source chunks
with relevance scores



Mock Login

Username/password authentication
gate with SHA-256 hashing



Evaluation Suite

10-question benchmark with keyword
matching & hallucination detection



Multi-turn Chat

Conversation history maintained for
context across interactions



End-to-end RAG solution with comprehensive evaluation and real-time parameter tuning

Setup & Installation

1

Clone Repository

```
git clone https://github.com/KaustubhNair-bot/AI_Tr_BOT.git  
cd AI_training_BOT/RAG_real_life_use_case_kaustubh
```

2

Virtual Environment

```
python3 -m venv venv  
source venv/bin/activate
```

3

Install Dependencies

```
pip install -r requirements.txt
```

4

Set Up API Key

```
cp .env.example .env  
Add your Groq API key in .env file
```

5

Run the App

```
streamlit run streamlit_app.py
```

6

Login

admin/admin123 agent/agent123 demo/demo123



Get your Groq API key at: <https://console.groq.com/keys>

Evaluation & Performance

Evaluation Categories

- Prepaid Plans
- Postpaid Plans
- Porting / MNP
- Intl. Roaming
- Refund Policy
- Broadband
- Customer Support
- Rewards Program
- Fair Usage Policy
- Cancellation

The screenshot shows the RAG Evaluation Results page. At the top, it displays the model used: "llama-3.3-70b-versatile". Below this, there are four main sections: "RAG Evaluation Results" (Avg Keyword Match: 56%, Avg Hallucination Rate: 7%, Test Cases: 10), "Generation Parameters" (Temperature: 0.30, Top-P: 0.5, Chain-of-Thought: 0.85, Retrieved Chunks (top-k): 5), and "RAG Statistics" (Chunks: 122, Vectors: 122, Chunk Size: 500, Overlap: 100). The "Per-Question Results" section contains a table with 10 rows, each representing a different category with its corresponding keyword score, matched keywords, hallucination rate, and response length.

ID	Category	Keyword Score	Keywords Matched	Hallucination Rate	Response Length
1	Prepaid Plans	50%	2/4	20%	1095
2	Postpaid Plans	40%	2/5	0%	1773
3	Porting / MNP	83%	5/6	0%	1409
4	International Roaming	75%	3/4	0%	1377
5	Refund Policy	100%	4/4	0%	1889
6	Broadband	20%	1/5	50%	1336
7	Customer Support	60%	3/5	0%	2891
8	Rewards	0%	0/6	0%	1458
9	FUP Policy	75%	3/4	0%	1418
10	Cancellation	60%	3/5	0%	1042

Performance Metrics

- Keyword Match Score
- Percentage of expected keywords found in response



Hallucination Rate



Percentage of unsupported numeric claims

Generation Parameters Comparison

Factual

Temp: 0.0
Top-P: 0.5

Lowest Hallucination

Balanced

Temp: 0.3
Top-P: 0.85

Low Risk

Creative

Temp: 0.8
Top-P: 0.95

Higher Risk

Optimal Settings: Temperature 0.0–0.3 with Top-P 0.5–0.85 provides best balance of factual accuracy and brand-voice adherence

Conclusion & Key Takeaways



Security Features

- ✓ SHA-256 hashed passwords
- ✓ Session-based authentication
- ✓ API key stored in .env
- ✓ Context constraints



Optimal Parameters

Temperature	0.0 - 0.3
Top-P	0.5 - 0.85

Best balance of **factual accuracy** and **brand-voice adherence**



Project Success Factors

- ★ Hybrid LLM + SLM approach
- ★ End-to-end RAG pipeline
- ★ Comprehensive evaluation suite
- ★ Brand voice persona enforcement



Future Potential

- Scalable to other telecom companies
- Enhance with more data sources
- Potential for real-time integration

Project Success



The hybrid RAG approach delivers the perfect balance of efficiency, accuracy, and brand consistency for customer support applications

Built by Kaustubh Nair

AI Training Program