DAY – 91

19 December 2025

1. System Overview

This project implements an **intelligent bilingual chatbot system** designed to answer user queries related to **Punjab Remote Sensing Centre (PRSC)**. The chatbot supports **English and Punjabi languages** and provides accurate responses using a hybrid approach combining **rule-based logic, machine learning, and Retrieval Augmented Generation (RAG)**.

2. Data Sources Used

The chatbot uses multiple knowledge sources:

- **Text Dataset (dataset.txt)** containing predefined question–answer pairs
- **PDF Document (PRSC.pdf)** from which FAQs are automatically extracted
- **Conversation history** to understand follow-up queries

This ensures better coverage and reliable responses.

3. Text Pre-processing

Before processing user queries, the system performs:

- Text normalization (lowercasing, Unicode cleaning)
- Stopword removal using **NLTK**
- Tokenization and light stemming
- Special handling for **Punjabi Unicode text**

This improves matching accuracy.

4. Question Detection & PDF Processing

The system intelligently detects questions by:

- Identifying question words (what, how, explain, etc.)
- Checking punctuation and numbering patterns
- Removing headers, footers, page numbers, and figure references from PDF

Extracted questions and answers from the PDF are converted into a structured FAQ format.

5. Machine Learning Based Matching

To find the best answer, the chatbot uses:

- **TF-IDF Vectorization**
- **Logistic Regression classifier**
- **Cosine Similarity** for semantic matching

This helps in identifying the most relevant answer even if the question wording is different.

6. Rule-Based & Fallback Matching

If ML confidence is low, the system applies:

- Token overlap–based rule matching
- Keyword similarity scoring

If no suitable answer is found, the chatbot politely asks the user to rephrase the question.

7. Follow-Up Question Handling

The chatbot can understand follow-up queries like:

- "Explain this"
- "Short of above"
- "Briefly tell"

It automatically links them to the **previous meaningful question**, making the conversation more natural.

8. Short Answer Generation

When users request **brief or short answers**, the system:

- Removes long explanations
- Converts responses into concise bullet points
- Extracts only key information

This is useful for quick understanding.

9. RAG (Retrieval Augmented Generation) Integration

For complex queries, the system uses:

- **LangChain**
- **HuggingFace embeddings**
- **Groq LLM**
- **Vector database (Chroma)**

This allows the chatbot to generate context-aware answers directly from the PDF content.

10. Multilingual Support

The chatbot supports **Punjabi and English** using:

- Automatic language detection
- Google Translate API for translation

This makes the system user-friendly for local users.

Intelligent Bilingual Chatbot System

The proposed project focuses on the development of an **intelligent bilingual chatbot system** designed to answer user queries related to the **Punjab Remote Sensing Centre (PRSC)**. The chatbot is capable of interacting with users in both **English and Punjabi languages**, making it suitable for a wider range of users, especially local stakeholders. The main objective of this system is to provide accurate, fast, and automated responses without the need for human intervention.

The chatbot uses a **hybrid question-answering approach**, combining **rule-based techniques, machine learning models, and Retrieval Augmented Generation (RAG)** to improve response accuracy. The system relies on multiple knowledge sources such as a **predefined text dataset (dataset.txt)** and an official **PRSC PDF document**, which is automatically processed to extract useful question–answer information. This ensures that the chatbot remains informative and domain-specific.

Before processing any query, the system performs extensive **text preprocessing**. This includes normalization of text, removal of stopwords using the **NLTK library**, tokenization, and light stemming. Special handling is implemented for Punjabi Unicode text to ensure correct interpretation. These preprocessing steps help reduce noise and improve the efficiency of similarity matching algorithms.

To extract information from the PDF document, the system removes unnecessary content such as headers, footers, page numbers, copyright text, and figure references. A **question detection mechanism** identifies potential questions using keyword patterns, punctuation, and numbering styles. The detected questions are paired with their corresponding answers and added to the chatbot's knowledge base.

For intelligent response generation, the chatbot uses **TF-IDF vectorization and cosine similarity** to measure the similarity between user queries and stored questions. A **Logistic Regression classifier** is trained to identify the most relevant answer. If the machine learning confidence is low, the system applies a **rule-based matching approach** based on keyword overlap and token similarity. This layered strategy increases reliability and reduces incorrect responses.

The chatbot also supports **follow-up query handling**, enabling it to understand references such as "this", "above", or "briefly explain". By analyzing conversation history, the system links follow-up questions to the previous meaningful query, resulting in a more natural and conversational interaction.

For complex or unseen questions, the system uses **Retrieval Augmented Generation (RAG)**. In this approach, relevant sections from the PDF are retrieved using vector embeddings, and a **large language model (LLM)** generates a context-aware response. This allows the chatbot to answer advanced queries beyond simple keyword matching.

Additionally, the chatbot can generate **short and concise answers** when users request brief explanations. Long responses are automatically converted into key points or bullet formats. Language translation is handled using an external translation API, ensuring smooth bilingual communication.