SHARDA
UNIVERSITY
Beyond Boundaries

Smart AI-Powered Customer Support Chatbot

Presented by:
Basharat Hassan, 2022521926

Submitted to:
Mr. Gyan Chandra Vishwakarma
Asst. Professor

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING,
SHARDA UNIVERISTY, GREATER NOIDA, UTTAR PARDESH

Problem Statement

- ❑ Small businesses struggle to offer 24/7 customer support
- ❑ Hiring human agents is expensive
- ❑ Customers expect instant responses
- ❑ FAQs exist, but not always easy to find or use
- ❑ Problem: No scalable, automated way for small businesses to respond to common customer queries instantly

Project Idea / Solution

- Idea: Create a GenAI-powered chatbot trained on company FAQs/help docs
- Responds instantly to customer queries
- Can be deployed on websites
- Uses either:
  - Retrieval-Augmented Generation (RAG) for dynamic answers or
  - Fine-tuned GPT model or

Why GenAI?

- Understands natural language
- Generates human-like, contextual replies
- Can be trained or enhanced using custom company knowledge
- More scalable and flexible than rule-based bots
- Ideal for unstructured data like help docs or FAQs

Use Case & Users

❑ Use Case: 24/7 automated customer support

❑ Users:

- Small business owners
- Customers visiting websites or messaging on WhatsApp

❑ Industries: E-commerce, SaaS, EdTech, Travel, Local Services

Tech Stack

❑ AI & Language Models

OpenAI GPT-3.5 / GPT-4 – for generating responses / Mistral / LLaMA2 – open-source alternatives
Embedding Model – text-embedding-ada-002 / SentenceTransformers

❑ Retrieval & Vector Store (RAG)

Vector Database – FAISS / Chroma / Pinecone
RAG Framework – LangChain / LlamaIndex

❑ Backend

Language – Python, Frameworks – FastAPI / Flask, Libraries – LangChain, OpenAI API, PyMuPDF (for PDF parsing)

❑ Frontend (Chat Interface)

React.js – for modern web UI or Streamlit – for a quick prototype

❑ Data & Storage

Source Data – FAQs, Help Docs, PDFs, Website text, Company FAQs, Help docs (PDFs, website content)
Optional DB – PostgreSQL / SQLite for logs, user queries

❑ Deployment

Frontend – Vercel / Render
Backend – Railway / Hugging Face Spaces / AWS / GCP

Project Architecture

Goal: Enable the chatbot to automatically answer customer queries using the company's FAQs and help docs, using Retrieval-Augmented Generation (RAG) or Fine-tuned LLM.

Architecture (RAG-based Flow)

[1] User Query
↓
[2] Chat Interface (Web)
↓
[3] Query Preprocessor
↓
[4] Retriever (Search Knowledge Base)
↓
[5] Generator (LLM creates answer)
↓
[6] Response to User

Step-by-Step Components

1. Chat Interface

- Frontend where users interact with the bot.
- It is an a web based chat widget (built with Streamlit, React, Flask).

2. Query Preprocessor

- Prepares the user input (e.g., remove special characters, lowercasing)
- Optional: Add spell correction, language detection

3. Retriever (RAG Part 1)

- Searches a knowledge base of documents
- Documents (FAQs, help docs) are chunked and embedded using models like text-embedding-ada-002
- Stored in a Vector Database (like FAISS, Chroma, Pinecone)
- Retrieves the top-k most relevant chunks based on similarity

4. Generator (RAG Part 2)

- Sends the user query + retrieved context to an LLM (like GPT-4, Mistral, or LLaMA2)
- LLM uses both to generate a human-like, accurate, context-rich response
- This is what makes it “retrieval-augmented” generation

5. Response to User

- The final AI-generated response is sent back to the user in the chat interface
- Can be followed by a feedback button: “Was this helpful?”

Step-by-Step Components

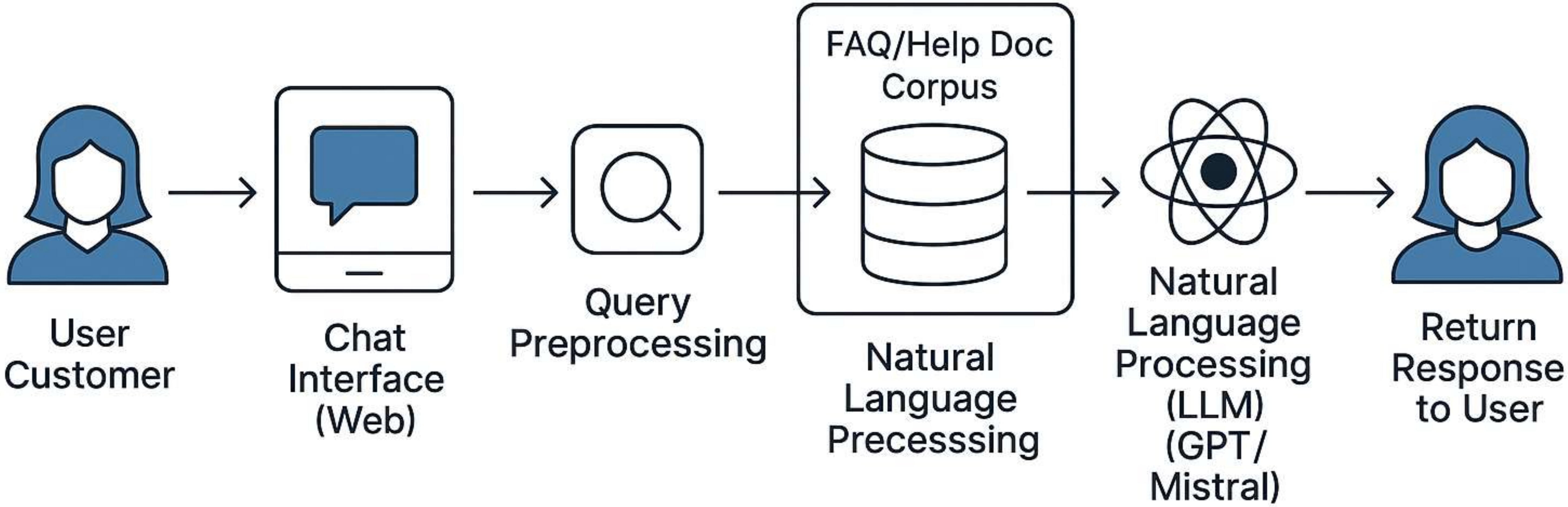
Optional (Advanced)

- Logging system: Logs all user questions and AI answers for review
- Fallback handler: “Sorry, I don’t have info on that. Would you like to talk to a human?”
- Multilingual support: Translate queries and answers

What If You Use Fine-Tuning Instead of RAG?

- You take a base GPT model and fine-tune it on all your FAQ pairs
- Easier for fixed questions, but RAG is better when:
- The content updates often
- You have large documents
- You want transparency (show the source)

Architecture Diagram



Sample Dataset / Demo Flow

- Example FAQ:
  - Q: "How do I return a product?"
  - A: "To return a product, go to your Orders > Return > Select reason..."
- Screenshot (if available) of sample chatbot answering a user query
- Alternatively: Show flow from user question → retrieved doc → generated answer

Benefits & Future Scope

❑Benefits:

- 24/7 automated support
- Scales with business growth
- Reduces customer wait times
- Lower support costs

❑Future Scope:

- Multilingual support
- Voice-enabled bot
- Sentiment detection
- Analytics dashboard (FAQs most asked, bot accuracy)

Conclusion

- Smart, scalable solution to a real-world problem
- Empowers small businesses with AI
- Demonstrates practical use of GenAI + RAG
- Future-proof customer support solution

Thank You