

BANK BOT AI CHATBOT FOR BANKS FAQ'S

MILESTONE 2

Abstract

The Smart Bank Assistant is an AI-powered conversational banking chatbot developed using Python. It automates customer interactions by handling essential banking operations such as balance inquiry, money transfer, loan and card management, and eligibility checks. The system also includes fallback prevention, slot-filling for incomplete inputs, and contextual dialogue flow, ensuring an interactive and intelligent user experience.

Introduction

In the modern banking industry, automation through chatbots is revolutionizing customer service by providing 24/7 assistance. The Smart Bank Assistant leverages Natural Language Understanding (NLU) and rule-based dialogue management to understand user queries and perform operations like checking balances, transferring money, finding ATM locations, and managing cards or loans.

This project is developed as Milestone 2 of the AI Banking Assistant series.

Objectives

- To build a smart conversational assistant that interacts with users using natural language.
- To integrate banking-specific intents like balance check, money transfer, and card/loan management.
- To implement entity extraction and slot filling for context-based interactions.
- To include confirmation prompts and eligibility checks for sensitive operations.
- To handle fallback responses gracefully when the input is unclear.

Problem Statement

Traditional banking systems require manual navigation through apps or visiting branches for small queries. This process is time-consuming and inefficient. The challenge is to create an intelligent banking chatbot capable of understanding natural language, extracting key entities (like account number, amount, city), and executing appropriate actions automatically.

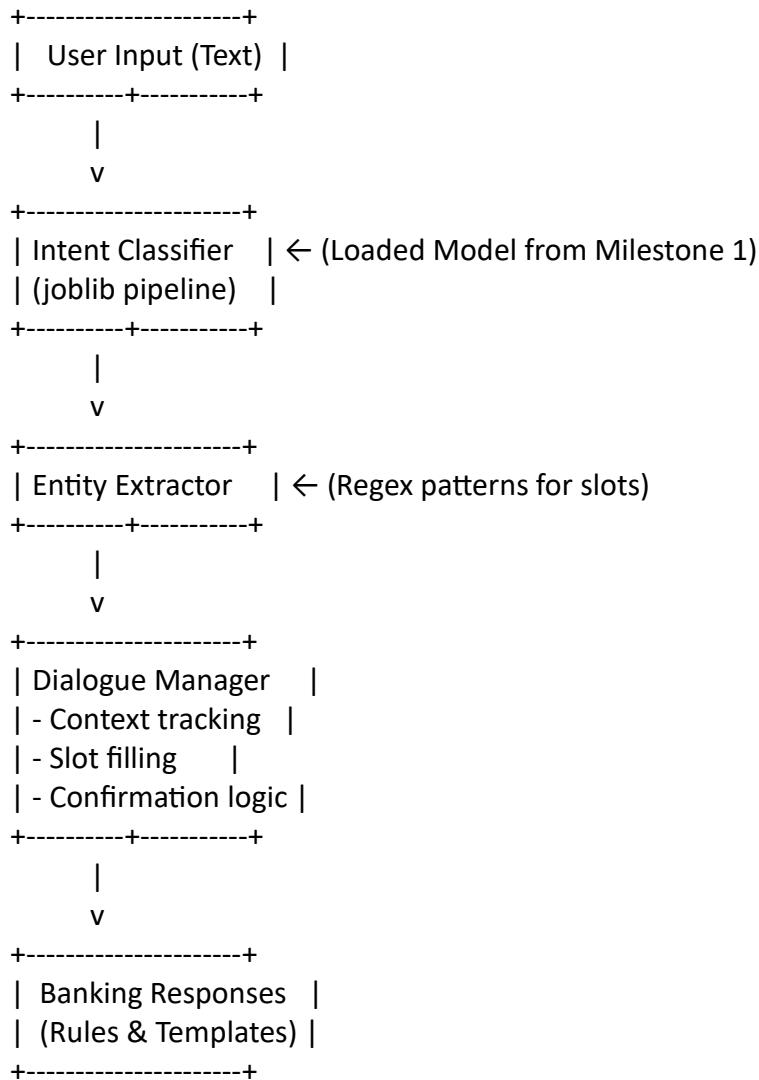
Existing System

Manual or semi-automated chat systems with limited capabilities.
Lack of contextual understanding in static rule-based chatbots.
No dynamic confirmation or eligibility logic for operations.
Users often face difficulty due to unhandled or ambiguous inputs.

Proposed System

The Smart Bank Assistant addresses these issues by:
Using a trained intent model (from Milestone 1) to classify user queries.
Implementing entity extraction with regex for slots like account number, amount, and card type.
Using a Dialogue Manager class to maintain conversation flow and context.
Handling loan and card services through structured menus.
Preventing fallback errors through intelligent prompts and intent normalization.

System Architecture



Dataset & Model

Model trained in Milestone 1 using a CSV dataset:

Columns: query, intent, entities

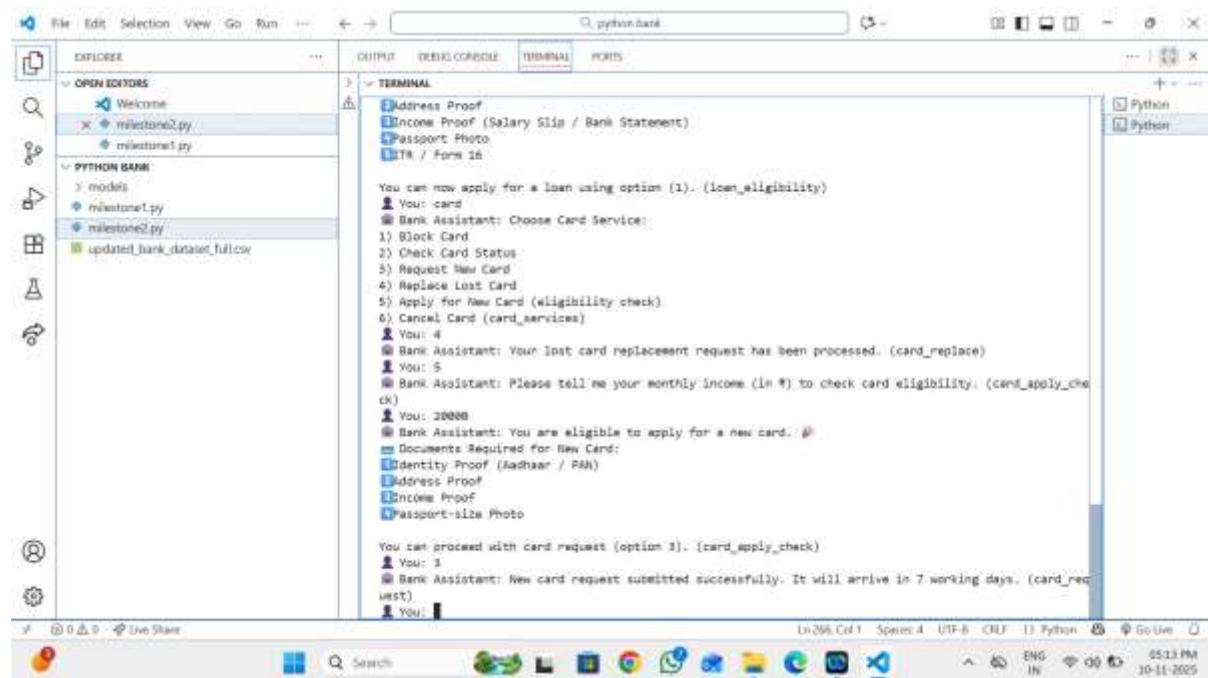
Techniques used:

TF-IDF Vectorizer for text features.

Logistic Regression / SVM classifier for intent detection.

Model saved as: models/intent_pipeline.joblib

Output:



The screenshot shows a Jupyter Notebook interface with the title bar "pythonbank". The left sidebar has sections for "OPEN EDITORS" (Welcome, milestone2.py, milestone1.py) and "PYTHON BANK" (models, milestone1.py, milestone2.py, updated_bank_dataset_full.csv). The main area has tabs for "OUTPUT", "TERMINAL", and "PORTS", with "TERMINAL" selected. The terminal window displays a simulated conversation between a user ("You") and a bank assistant:

```
Address Proof
Income Proof (Salary Slip / Bank Statement)
Passport Photo
ITR / Form 16

You can now apply for a loan using option (1). (loan_eligibility)
>You: card
>Bank Assistant: Choose Card Service:
1) Block Card
2) Check Card Status
3) Request New Card
4) Replace Lost Card
5) Apply for New Card (eligibility_check)
6) Cancel Card (card_services)
>You: 4
>Bank Assistant: Your Iptr card replacement request has been processed. (card_replace)
>You: 5
>Bank Assistant: Please tell me your monthly income (In ₹) to check card eligibility. (card_apply_check)
>You: 20000
>Bank Assistant: You are eligible to apply for a new card. (card_eligible)
>Documents Required for New Card:
Identity Proof (Aadhaar / PAN)
Address Proof
Income Proof
Passport-size Photo

You can proceed with card request (option 3). (card_apply_check)
>You: 3
>Bank Assistant: New card request submitted successfully. It will arrive in 7 working days. (card_request)
>You:
```

Conclusion

The Smart Bank Assistant successfully simulates intelligent banking interactions through Natural Language Processing and rule-based logic. It automates core banking services and demonstrates the potential of conversational AI in improving customer experience within the financial sector.