Assignment No: - 9

Name :- Isha Ghorpade

Enrollnment :- 22420020

Roll :- 381066

Problem Statement

Develop a Chatbot application for a real-world scenario (e.g., customer support, healthcare consultation, or hotel booking) that interacts with users using natural language and provides relevant responses.

Objectives

- To understand natural language processing (NLP) concepts in AI.
- To implement a rule-based or AI-based chatbot.
- To simulate human-like interaction for a specific domain.
- To evaluate the chatbot for accuracy, usability, and response relevance.

Theory

Introduction

A chatbot is a software application designed to simulate conversation with human users using natural language.

- Chatbots can be rule-based (predefined responses) or AI-based (learn from data).
- Widely used in customer support, e-commerce, healthcare, and personal assistants.

Types of Chatbots

1. Rule-Based Chatbots
    - Use if-else rules or pattern matching.
    - Simple to implement but limited flexibility.
2. AI-Based Chatbots
    - Use machine learning and NLP for understanding and generating responses.
    - Can handle complex and dynamic conversations.

Components of a Chatbot

1. Input Processing:
    - Accepts user query.
    - Tokenizes and processes text.
2. Intent Recognition:
    - Understands what the user wants.
    - Uses keyword matching or ML/NLP models.
3. Response Generation:
    - Provides a suitable reply based on rules or trained models.
4. Knowledge Base:
    - Stores domain-specific information for generating responses.
5. Output:
    - Returns response to user.

Methodology

1. Define the real-world scenario (e.g., hotel booking).
2. Identify user intents (greeting, booking room, checking availability, farewell).
3. Build a knowledge base with rules or FAQs.
4. Implement input processing (tokenization, pattern matching).
5. Design response generation logic.
6. Test and refine responses for accuracy and relevance.

Advantages

- Available 24/7 for users.
- Reduces human effort for repetitive queries.
- Provides instant response and improves user experience.

Limitations

- Rule-based chatbots have limited understanding.
- AI chatbots require training data and computational resources.
- May fail to handle ambiguous or complex queries.

Applications

- Customer support for e-commerce websites.
- Healthcare consultation (symptom checking).
- Travel and hotel booking.
- Banking and finance queries.
- Educational tutoring systems.

Python Implementation

```
# Simple Chatbot for Hotel Booking

def chatbot():
    print("Welcome to HotelBot! How can I help you today?")

    while True:
        user_input = input("You: ").lower()

        if "hello" in user_input or "hi" in user_input:
            print("HotelBot: Hello! How can I assist you today?")

        elif "book" in user_input or "room" in user_input:
            print("HotelBot: Sure! Please provide your check-in and check-out dates.")

        elif "availability" in user_input:
            print("HotelBot: We have rooms available for your selected dates.")

        elif "bye" in user_input or "exit" in user_input:
            print("HotelBot: Thank you for visiting! Have a great day!")
            break

        else:
```

```
        print("HotelBot: I'm sorry, I didn't understand that. Can you rephrase?")
```

 # Run Chatbot

chatbot()

How it works:

- Handles basic greetings, booking queries, availability checks, and exit.
- Can be extended to include database integration, AI/NLP models, or GUI interface.

Conclusion

A chatbot application provides an efficient way to interact with users in a specific domain.

- Rule-based chatbots are easy to implement and handle simple queries.
- AI-based chatbots improve user experience by understanding complex queries.
- Chatbots are widely applied in business, healthcare, travel, and education for automation and improved efficiency.