MINI PROJECT REPORT

ON

LAW BASED CHAT-BOT

Submitted By

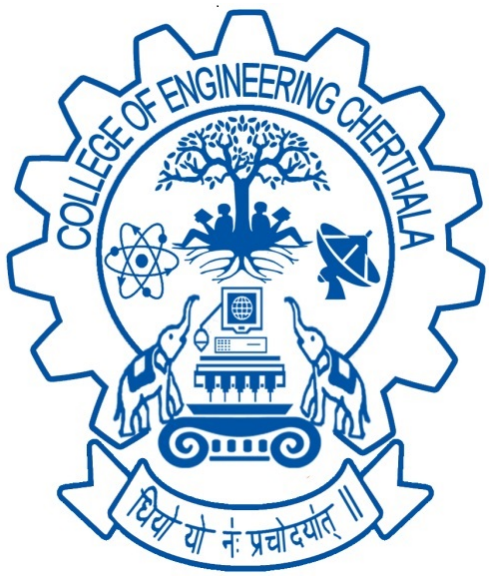
SAM SIMON (CEC23MCA2035)

Guided by

Ms.VISHNUPRIYA G S

in partial fulfillment for the award of the degree of

Master of Computer Application



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COLLEGE OF ENGINEERING, CHERTHALA

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

NOVEMBER 2024

MINI PROJECT REPORT ON

LAW BASED CHAT-BOT

Submitted By

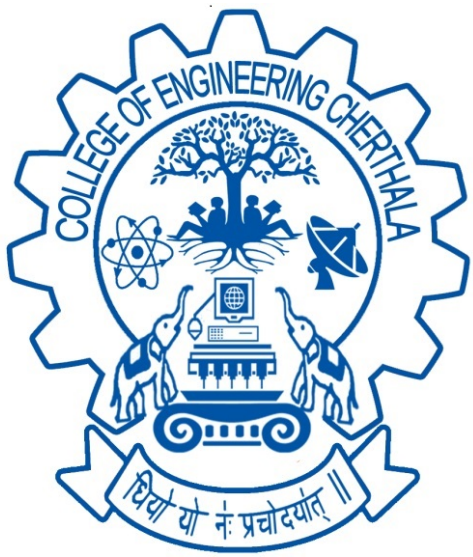
SAM SIMON (CEC23MCA2035)

Guided by

Ms.VISHNUPRIYA G S

in partial fulfillment of the requirements for the award of the degree of

Master of Computer Application
of
APJ Abdul Kalam Technological University



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COLLEGE OF ENGINEERING, CHERTHALA

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

NOVEMBER 2024

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COLLEGE OF ENGINEERING, CHERTHALA

ALAPPUZHA-688541

C E R T I F I C A T E

This is to certify that, the project report titled **"LAW BASED CHAT-BOT"** is a bonafide record of the **20MCA245 Mini Project** presented by **Sam Simon (CEC23MCA2035)**, Third semester Master of Computer Application student, under our guidance and supervision, in partial fulfillment of the requirements for the award of the degree, **Master of Computer Application** of **APJ Abdul Kalam Technological University during the academic year 2024-2025**.

Guide	**Co-ordinator**	**HOD**
Ms.Vishnupriya G S	**Ms.Renjusha Aravind**	**Dr. Preetha Theresa Joy**
Assistant Professor	Assistant Professor	Professor
Dept.of Computer Engg.	Dept.of Computer Engg.	Dept.of Computer Engg.

ACKNOWLEDGEMENT

This work would not have been possible without the support of many people. First and the foremost, I give thanks to Almighty God who gave us the inner strength, resource and ability to complete my project successfully.

I,(Sam Simon) would like to thank **Dr. Jaya V.L**, my Principal, who has provided with the best facilities and atmosphere for the project completion and presentation.I would also like to thank my HoD **Dr. Preetha Theresa Joy** (Professor, Department of Computer Science And Engineering), my project coordinator **Ms.Renjusha Aravind** (Assistant Professor, Department of Computer Science And Engineering), and my guide **Ms.Vishnupriya G S** (Assistant Professor, Department of Computer Science And Engineering) for the help extended and also for the encouragement and support given to us while doing the project.

I would like to thank my dear friends for extending their cooperation and encouragement throughout the project work, without which i could have never completed the project this well.Thank you all for your love and also for being very understanding.

DECLARATION

I hereby declare that the project “Law Based Chat-bot” is a bonafide work done by me during the academic year 2024-2025 under the guidance of (Vishnupriya G S), Assistant Professor at College of Engineering, Cherthala and this report has not been previously formed the basis for the award of any degree, diploma, fellowship or any other similar title or recognition in any other university.

SAM SIMON

CEC23MCA2035

06/11/2024

ABSTRACT

The project aims to create an intelligent chat-bot that helps people find relevant laws in India based on their situations. It uses natural language processing to understand user questions and provide quick legal guidance. The goal is to make legal information more accessible and reduce barriers to seeking legal help. Future improvements could include advanced technology, multilingual support and features like emergency service buttons and list of lawyers.

Contents

1 INTRODUCTION **1**

2 PROBLEM STATEMENT **2**

2.1 Problem Statement 2

2.2 Objective 2

3 LITERATURE SURVEY **3**

3.1 Case Study 1 3

3.2 Case Study 2 3

3.3 Case Study 3 4

3.4 Case Study 4 4

3.5 Case Study 5 4

4 PROPOSED SYSTEM **5**

4.1 Solution 5

4.2 Feasibility Study 6

4.2.1 Technical Feasibility 6

4.2.2 Social Feasibility 7

4.2.3 Schedule Feasibility 7

4.2.4 Operational Feasibility 7

4.2.5 Economic Feasibility 7

5 SYSTEM ARCHITECTURE **8**

5.1 SYSTEM ARCHITECTURE DIAGRAM 8

5.2 ARCHITECTURE BREAKDOWN 9

6 METHODOLOGY 10

7 SOFTWARE AND HARDWARE REQUIREMENTS 11

7.1 Overall Description 11

7.1.1 Product Perspective 11

7.1.2 Product Function 12

7.2 Software Requirements 12

7.2.1 Python 12

7.2.2 NLTK 12

7.2.3 Flask 13

7.2.4 Pandas 13

7.2.5 Scikit-learn 13

7.2.6 Pickle 14

7.3 Hardware Requirements 14

8 SYSTEM DESIGNS 15

8.1 Use-case Diagram 15

8.2 Sequence Diagram 16

8.3 Data Flow Diagram 17

8.3.1 Level 0 18

8.3.2 Level 1 18

8.3.3 Level 2 19

9 IMPLEMENTATION 20

9.1 Coding Environment Used 20

9.2 Model Training 20

9.3 Storage 20

9.4 Backend 21

9.5 Frontend 21

10 RESULT AND ANALYSIS **22**

10.1 Screenshots 22

10.1.1 User Interface 22

10.1.2 Normal Conversation 23

10.1.3 Provides Legal Suggestions 23

10.1.4 Voice Input 24

11 SAMPLE CODES **25**

11.1 Frontend 25

11.2 Backend 26

11.3 Training Model 26

11.4 Dataset/Training Data 27

12 APPLICATIONS **28**

13 ADVANTAGES AND DISADVANTAGES **29**

13.1 ADVANTAGES 29

13.2 DISADVANTAGES 29

14 FUTURE SCOPE **30**

15 CONCLUSION **31**

REFERENCES **32**

List of Figures

5.1 SYSTEM ARCHITECTURE 8

8.1 Use-case diagram 16
8.2 Chat-bot 17
8.3 Law Based Chat-bot 18
8.4 Legal Chat-bot System 18
8.5 NLP Engine 19

10.1 User Interface 22
10.2 Conversation 23
10.3 Legal suggestions 23
10.4 Voice Integration 24

11.1 Frontend-Design 25
11.2 Backend 26
11.3 Training Model 26
11.4 Data-Collection 27

Chapter 1

INTRODUCTION

Finding legal information in India can be challenging for many people. Legal chatbots can help by making it easy to access legal advice, ask questions, find lawyers, and review documents. These chatbots are simple to use and don't require special training, making them great for reaching a wide audience.

Legal chatbots can change how legal services are provided in India, making them more affordable and accessible. As this technology grows, it's important to use it responsibly and ensure it gives accurate information. Chatbots can also help automate tasks, improve transparency, and encourage people to learn about their rights.

We are developing a chatbot which uses Natural Language Processing (NLP) to help users. LAWYER-GENIE is an AI-powered chatbot that offers legal guidance free of charge. It suggests relevant Indian laws based on user situations and simplifies access to legal information using Natural Language Processing. This tool is designed to make legal knowledge more accessible and understandable for everyone

Chapter 2

PROBLEM STATEMENT

2.1 Problem Statement

The project aims to improve access to legal information for the general public in India by creating a user-friendly chat-bot. Navigating legal documents and understanding complex legal language can be challenging, but this chat-bot will simplify this process. Using machine learning and natural language processing, the chat-bot will help users ask legal questions and access commonly sought legal information, making legal support more accessible and understandable for everyone.

2.2 Objective

To develop "LAWBOT," a smart Indian legal chatbot that utilizes machine learning and natural language processing (NLP) to make legal information more accessible ,user-friendly and cost effective. The primary goals are to:

- **Legal Suggestions:**"Give users specific legal advice based on their case questions, important words.
- **Real-time Processing:** Ensuring the chatbot operates in real-time to offer immediate feedback and assistance to users.

Chapter 3

LITERATURE SURVEY

3.1 Case Study 1

LAWBOT: A Smart User Indian Legal Chatbot using Machine Learning Framework

[1] Legal information in India is often challenging for the average person to comprehend due to complex language and specialized terminology. Lawbot, an intelligent chatbot powered by machine learning, addresses this issue by providing accessible legal assistance. It enables users to ask legal questions, find qualified lawyers, and review legal documents with ease, offering a user-friendly platform that demystifies legal information and enhances access to legal resources across India.

3.2 Case Study 2

LAW-U: Legal Guidance Through Artificial Intelligence

[2]LAW-U is an AI-powered chatbot designed to provide legal guidance to survivors of sexual violence in Thailand by recommending relevant Supreme Court cases. Developed using a dataset of 182 court cases, LAW-U achieves an impressive 88.89 percent accuracy in matching users' inputs to these cases, ensuring that users receive accurate and relevant legal information. The primary aim of LAW-U is to support survivors by providing accessible legal knowledge, raise awareness around legal rights, and empower victims to assert their rights with confidence.

3.3 Case Study 3

A Free Format Legal Question Answering System

[3] This free-format legal question-answering system enables users to ask legal questions in their own words, making it highly accessible and intuitive. Utilizing advanced technology, the system finds and ranks the best answers based on relevance and accuracy. Trained on a diverse dataset of general and legal information, it serves as a valuable tool for businesses, allowing them to obtain precise legal insights and improve decision-making.

3.4 Case Study 4

Generative vs Intent-based Chatbot for Judicial Advice

[4] In providing judicial advice, generative and intent-based chatbots offer distinct approaches. The generative AI chatbot uses deep learning and a dataset of conversations to produce human-like responses and is trained with legal information to enhance contextual understanding. While it delivers rich, nuanced answers, it may lack precision when handling complex legal specifics. In contrast, the intent-based chatbot identifies user intents and offers predefined responses, ensuring high accuracy but limited flexibility as it is tailored to specific scenarios.

3.5 Case Study 5

Legal Solutions - Intelligent Chatbot using Machine Learning

[5] This intelligent AI chatbot is designed to help individuals navigate legal issues by offering basic legal information and personalized guidance. Users can chat in real-time with qualified lawyers and customize their search options based on factors like financial situation and location, making it easy to find relevant, affordable legal support. The chatbot's primary goal is to make legal assistance accessible to everyone, enabling individuals to manage their legal needs more effectively and independently.

Chapter 4

PROPOSED SYSTEM

4.1 Solution

The proposed system aims to develop an AI-powered chatbot, Lawyer-Genie, that offers comprehensive legal guidance while simplifying access to legal information. This system prioritizes user experience through a friendly interface and free access to legal resources. The system will consist of the following components:

1. **User-Friendly Interface:** Develop a user-friendly interface that allows users to easily navigate the chat-bot and access legal information without confusion.

2. **Legal Guidance Provision:** Implement natural language processing (NLP) algorithms to understand user queries and provide accurate legal guidance tailored to individual situations.

3. **Voice Input Processing:** Speech recognition technology will convert user input into text data, followed by prepossessing techniques to clean and normalize the input.

4. **Law Suggestion Engine:** Create a system that suggests relevant laws and legal provisions based on user inquiries and situations, making the information applicable and practical.
   Maintain a continuously updated database of laws, regulations, and legal precedents to ensure users receive current and accurate information.

5. **Chatbot Interaction:** The chatbot will offer a user-friendly interface for mental health support, allowing users to input text or voice commands, with responses tailored to their emotional state.

6. **Free Access:** Ensure that all users can access Lawyer-Genie without any fees, making legal guidance accessible to a wider audience.

4.2 Feasibility Study

The main objective of this study is to determine whether the proposed system is feasible or not. Mainly there are three types of feasibility study to which the proposed system is subjected as described below: Five key considerations are involved in this feasibility

- Technical Feasibility
- Social Feasibility
- Schedule Feasibility
- Operational Feasibility
- Economic Feasibility

The proposed system must be evaluated from a technical viewpoint first, and if technically feasible, their impact on the organization must be assessed. If compatible, the operational system can be devised. Then those must be tested for economic feasibility.

4.2.1 Technical Feasibility

The technologies required for the development is identified. Since, both the hardware and software requirements are satisfied, it is technically feasible.

4.2.2 Social Feasibility

The proposed project will be socially feasible. The social feasibility determines whether the project would be accepted by the people. This assumption would in general examine the probability that the project would have to be accepted by the group of people that are directly affected by the proposed system.

4.2.3 Schedule Feasibility

The primary analysis depicts that the project can be completed by the schedule. Thus the project is feasible.

4.2.4 Operational Feasibility

The proposed project is beneficial because it helps in making faster and reliable decisions. So, users will be encouraged to use it, and it is expected to serve the user's need, which means helping the user in making decisions.

4.2.5 Economic Feasibility

The system will be developed at reasonable cost with the available hardware, software and manpower. So, its benefits overweigh the cost. So, it is economically feasible.

Chapter 5

SYSTEM ARCHITECTURE

5.1 SYSTEM ARCHITECTURE DIAGRAM

architecture refers to the conceptual model that defines the structure, behavior, and more views of a system.

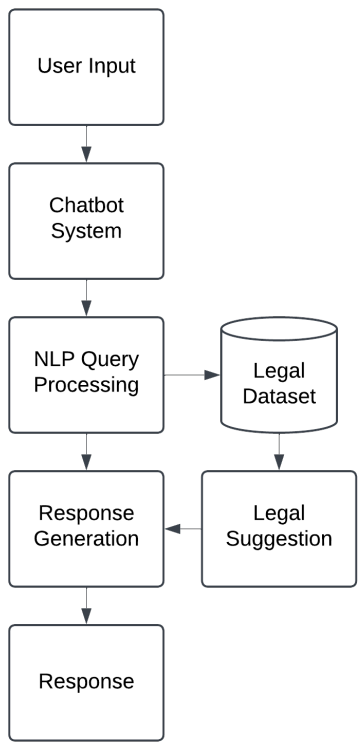


Fig. 5.1: SYSTEM ARCHITECTURE

5.2 ARCHITECTURE BREAKDOWN

1. **User Input:**This is where the user interacts with the chatbot by typing a question or legal query. The input could be about various legal topics or issues they need help with.

2. **Chatbot System:**The chatbot system receives the user input and manages the interaction flow. It passes the input to the natural language processing (NLP) module for further analysis.

3. **NLP Query Processing:**In this part, the NLP module analyzes and interprets the user's input. It identifies key legal terms, entities, and context from the query to understand the user's intent. This analysis enables the chatbot to find the most relevant legal information.

4. **Legal Dataset:**This is the database or repository containing legal information, including laws, regulations, and legal precedents. The NLP module accesses this dataset to retrieve relevant information based on the user's query.

5. **Legal Suggestion:**After retrieving information from the legal dataset, the system generates a legal suggestion that addresses the user's question. This suggestion is the initial response based on the most relevant legal data.

6. **Response Generation:** This module takes the legal suggestion and formats it into a coherent and understandable response. It may also add additional context or explanation to make the response user-friendly.

7. **Response :** The final output from the chatbot system is sent back to the user. This response provides the legal information or advice based on the user's initial input, helping them understand the legal context or next steps.

Chapter 6

METHODOLOGY

1. **Natural Language Processing (NLP):** NLP techniques enable the chatbot to understand and process user queries by identifying keywords, entities, and relevant legal concepts. Tokenization and named entity recognition (NER) allow the chatbot to interpret complex legal language accurately.

2. **Logistic Regression:** A Logistic Regression model, trained on a set of question-answer pairs, classifies user queries to select the most relevant response. This model estimates response likelihoods, ensuring accurate and contextually relevant answers.

3. **TF-IDF Vectorization:** TF-IDF converts text into numerical values, emphasizing terms that are frequent in a query but rare in others. This helps the chatbot prioritize key terms and match user queries to the most relevant answers.

4. **Speech Recognition:** Speech recognition allows users to interact with the chatbot through voice. The system transcribes spoken words into text, providing a hands-free option that enhances accessibility and user experience.

5. **Data Storage and Serialization (Pickle):** The Python Pickle module serializes the trained model, allowing for efficient storage and reloading. This improves response times and ensures the chatbot's legal guidance is readily available without needing to retrain the model.

Chapter 7

SOFTWARE AND HARDWARE REQUIREMENTS

7.1 Overall Description

The purpose of this document is to provide a comprehensive description of the law-based chatbot system, Lawyer-Genie. It will outline the system's objectives and features, the interfaces involved, the functionalities it will offer, the constraints it must operate under, and how the system will respond to external inputs. This system aims to deliver accessible legal guidance, creating a platform for individuals to seek information about their legal rights and responsibilities. Additionally, it serves as an efficient tool for simplifying access to legal resources and information, enhancing the overall user experience, and facilitating timely access to relevant legal support.

7.1.1 Product Perspective

This project aims to develop a web application designed to provide legal support through an interactive chatbot, Lawyer-Genie. Users can engage with the chatbot to receive personalized legal guidance, access a comprehensive range of legal resources, and find relevant laws based on their specific situations. The platform will simplify the process of obtaining legal information and empower users to make informed decisions regarding their legal rights and responsibilities. By offering a user-friendly interface, the chatbot will facilitate meaningful interactions that promote

understanding and access to legal support.

7.1.2 Product Function

This project has many functions. They are listed below:

- 24/7 Availability
- Accurate Responses
- Cost-effective and Reliable

7.2 Software Requirements

7.2.1 Python

Python is a versatile and beginner-friendly programming language known for its simplicity and readability. Created by Guido van Rossum and first released in 1991, Python allows developers to express concepts in fewer lines of code compared to other languages. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. With a vast standard library and a thriving ecosystem of third-party packages, Python is suitable for various applications, from web development and data analysis to artificial intelligence and scientific computing. Its interpreted nature and dynamic typing contribute to its flexibility, making it a popular choice for both beginners and experienced developers.

7.2.2 NLTK

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with asuite of text processing libraries for classification, tokenization, stemming, tagging,Natural Language Processing with Python provides a practical introduction to programming for language processing. Written by the creators of NLTK, it guides the reader through the fundamentals of writing Python programs, working with corpora, categorizing text, analyzing linguistic

structure, and more. parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum

7.2.3 Flask

Flask is a lightweight and flexible web framework for Python, designed to simplify the development of web applications. Created by Armin Ronacher, it features a minimalistic approach and easy-to-use syntax, making it suitable for both beginners and experienced developers. Flask includes essential web development features like URL routing, template rendering, and request handling, while allowing developers to choose additional libraries as needed. Following the WSGI (Web Server Gateway Interface) specification, it's compatible with various web servers. With extensive documentation and a vibrant community, Flask supports a wide range of web applications, from simple prototypes to large-scale projects.

7.2.4 Pandas

Pandas is a powerful and flexible data analysis library for Python, designed to make data manipulation and analysis easy and intuitive. Developed by Wes McKinney, it provides data structures like Series and DataFrame, which allow for efficient handling of structured data. Pandas offers a wide range of functionalities, including data cleaning, transformation, and aggregation, making it an essential tool for data scientists and analysts. Its integration with other libraries, such as NumPy and Matplotlib, enhances its capabilities for statistical analysis and data visualization. With its user-friendly syntax and extensive documentation, Pandas is widely used in various fields, from finance to academia, for tasks involving data exploration and analysis.

7.2.5 Scikit-learn

scikit-learn, often referred to as sklearn, is a widely-used machine learning library for Python that provides simple and efficient tools for data mining and data analysis. Built on top of NumPy, SciPy, and Matplotlib, it offers a comprehensive range of algorithms for classification, regression, clustering, and dimensionality reduction. Its user-friendly API allows for easy model training,

evaluation, and tuning, making it accessible for both beginners and experienced practitioners. scikit-learn is ideal for developing machine learning models quickly and effectively, and it includes extensive documentation and tutorials to support users in their data science projects.

7.2.6 Pickle

Pickle is a Python module used for serializing and deserializing Python objects, enabling easy storage and retrieval of complex data types. By converting Python objects into a byte stream, Pickle allows developers to save data structures like lists, dictionaries, and even custom classes to files or transmit them over a network. This functionality is particularly useful for saving machine learning models or data preprocessing pipelines, allowing for efficient model persistence and later reuse without the need for retraining. While Pickle is convenient, it's important to be cautious with untrusted data, as loading pickled objects can pose security risks.

7.3 Hardware Requirements

To run the chatbot on a system, there are several hardware requirements to consider to ensure smooth operation, efficient processing, and reliable performance. The specific requirements may vary based on the complexity of the chatbot, the size of the datasets, and the expected number of concurrent users.

• Processor (CPU): Intel Core i5 or equivalent

• Memory (RAM): 8 GB

• Graphics Processing Unit (GPU): Not essential for basic usage

• Storage: 256 GB SSD

Chapter 8

SYSTEM DESIGNS

8.1 Use-case Diagram

Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform. The below mentioned use case diagram has Two actors:

- Users : The user can access the chatbot whenever required.
- System : The chat-bot generates an appropriate response.

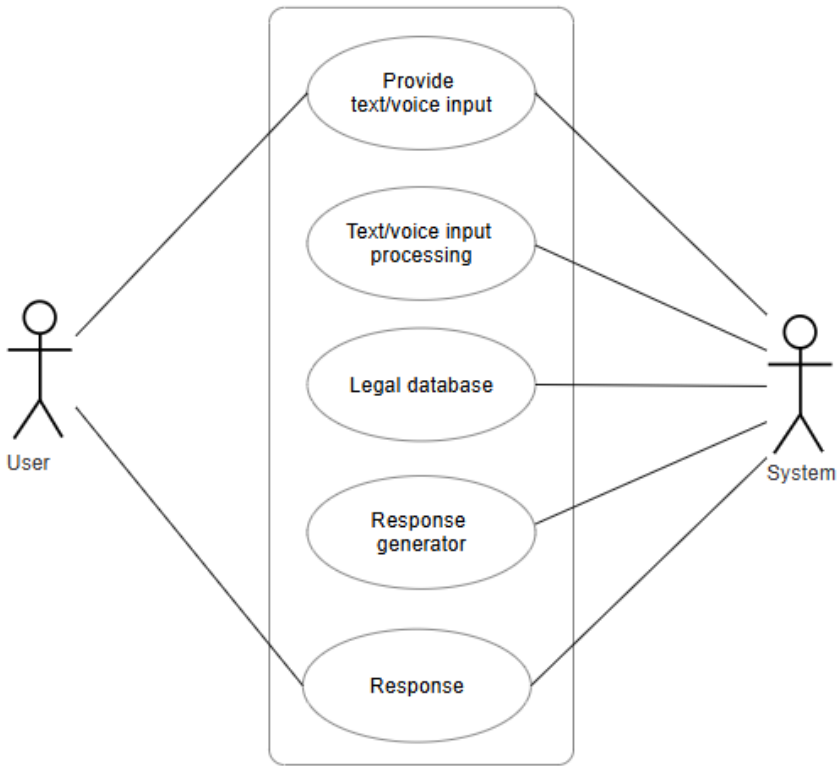


Fig. 8.1: Use-case diagram

8.2 Sequence Diagram

- The sequence diagram show interactions using messages in the Law Based Chat-bot System.
- Users can engage with Lawyer-Genie to seek legal guidance. Users can send messages such as ask Legal Question to inquire about specific legal issues or get Relevant Laws

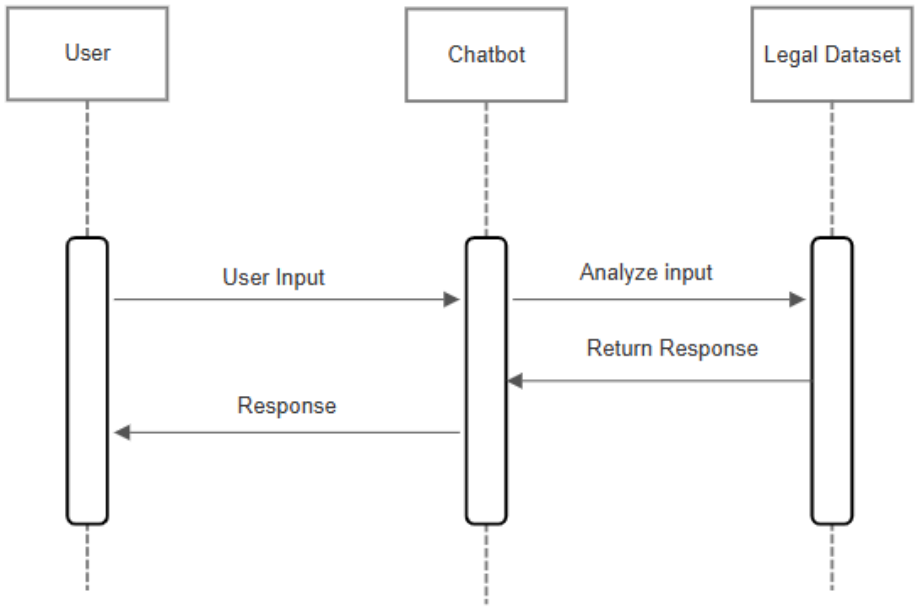


Fig. 8.2: Chat-bot

8.3 Data Flow Diagram

The data flow diagram (DFD) is used for classifying system requirements to major transformation that will become programs in system design. This is starting point of the design phase that functionally decomposes the required specifications down to the lower level of details

Bubbles: Represent the data transformations.

Lines: Represent the logic flow of data.

Data can trigger events and can be processed to useful information. Systems analysis recognizes the central goal of data in organizations.

8.3.1 Level 0

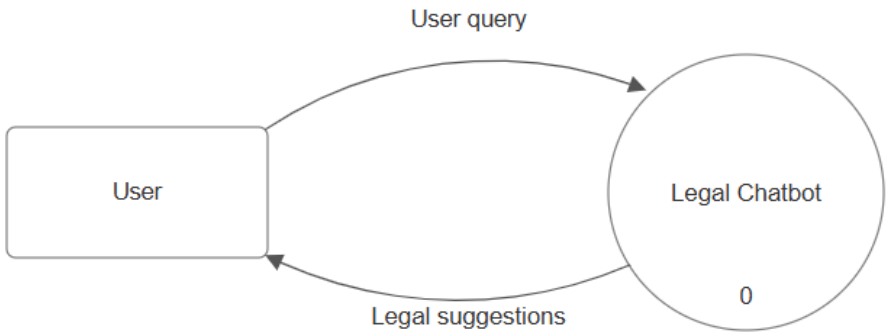


Fig. 8.3: Law Based Chat-bot

8.3.2 Level 1

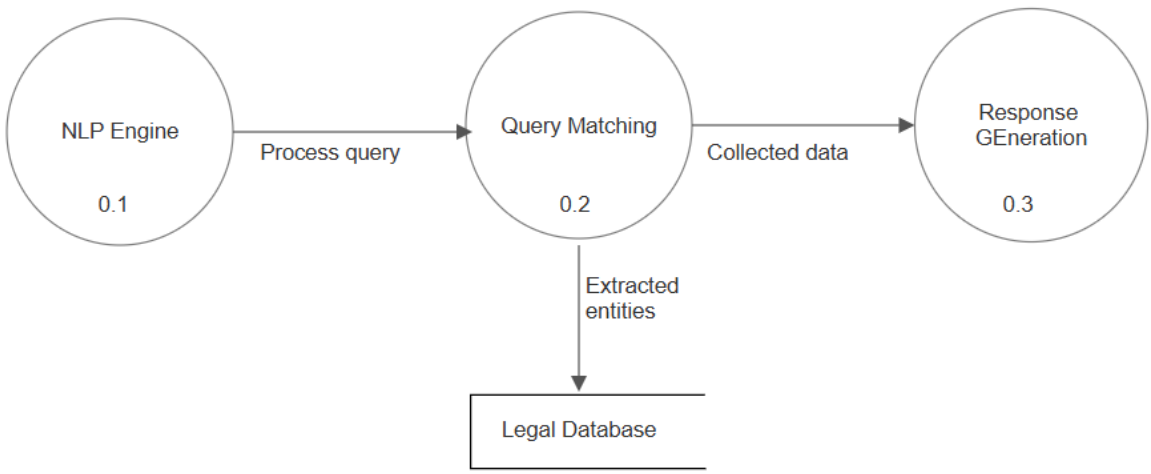


Fig. 8.4: Legal Chat-bot System

8.3.3 Level 2

Fig. 8.5: NLP Engine

Chapter 9

IMPLEMENTATION

The Law Based chat-bot application uses a Flask web framework to connect a trained machine learning model with a responsive front-end, creating an interactive conversational experience

9.1 Coding Environment Used

Visual Studio Code is a source-code editor developed by Microsoft for Windows, Linux and macOS. It includes support for debugging, intelligent code completion etc. . . .

9.2 Model Training

A Logistic Regression model is trained on a dataset of predefined question-answer pairs. The model uses TF-IDF to convert text to numerical features, which helps predict relevant responses.

9.3 Storage

The trained model and vectorizer are serialized (saved) with pickle to a file, making them ready for deployment without retraining.

9.4 Backend

The Flask app loads the saved model and vectorizer upon startup.It accepts user messages via a POST request, preprocesses the text, and uses the model to predict a response.The app includes optional translation support with Googletrans, allowing responses in different languages if configured.

9.5 Frontend

The frontend is designed with HTML, Bootstrap, and jQuery, providing a user-friendly chat interface. jQuery captures user input and displays the chatbot's response in real time, mimicking a conversational experience.

Chapter 10

RESULT AND ANALYSIS

10.1 Screenshots

10.1.1 User Interface

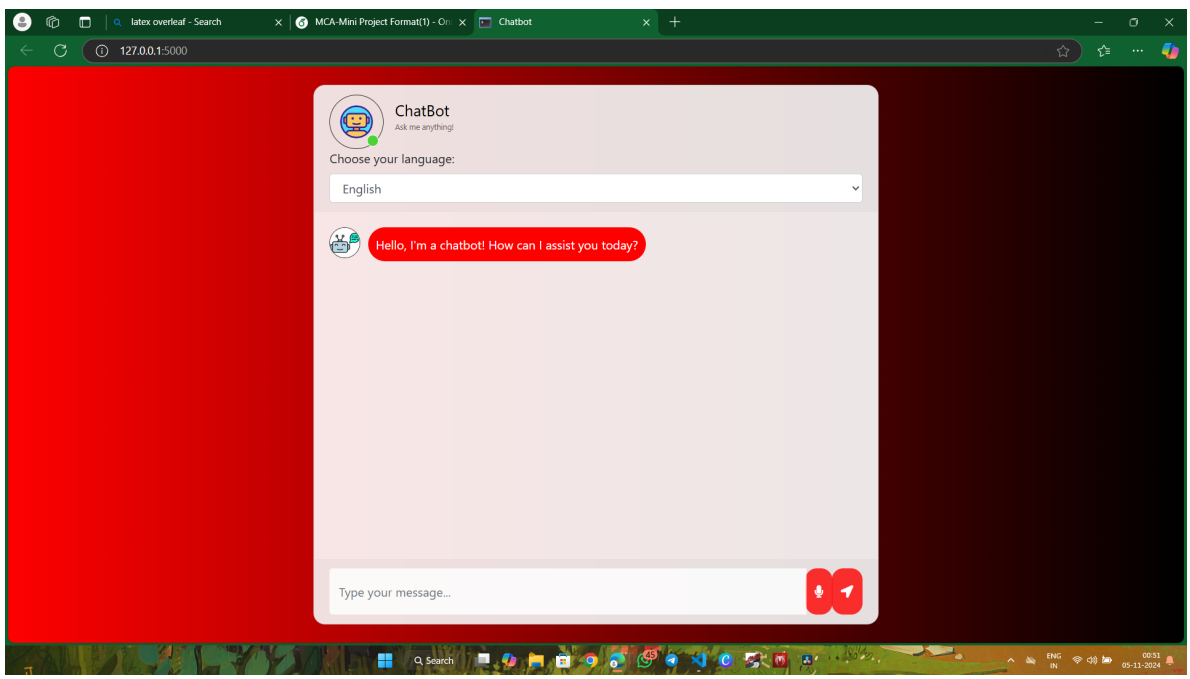


Fig. 10.1: User Interface

10.1.2 Normal Conversation

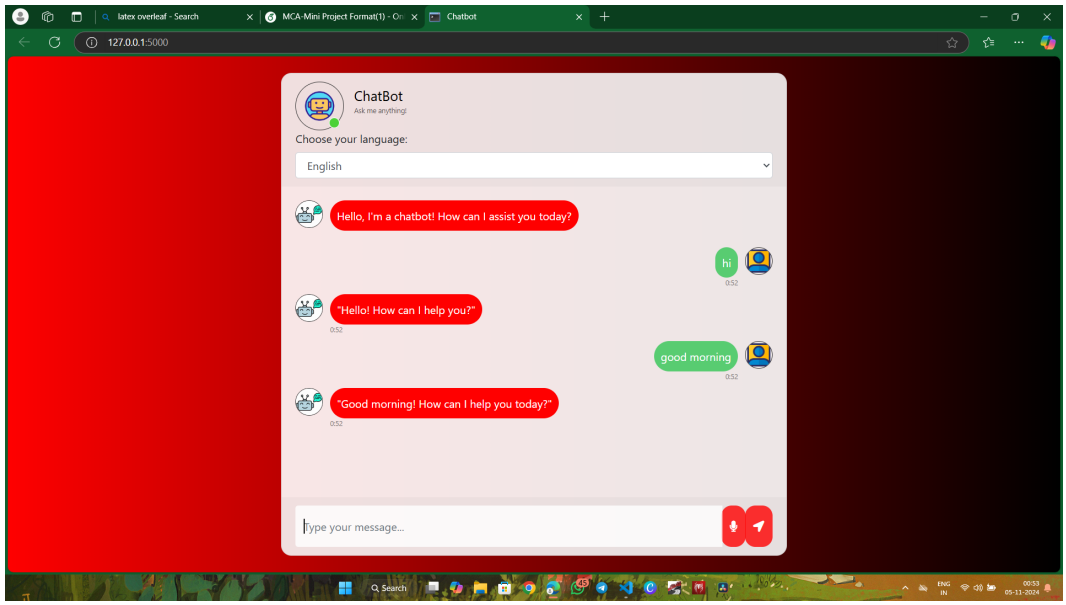


Fig. 10.2: Conversation

10.1.3 Provides Legal Suggestions

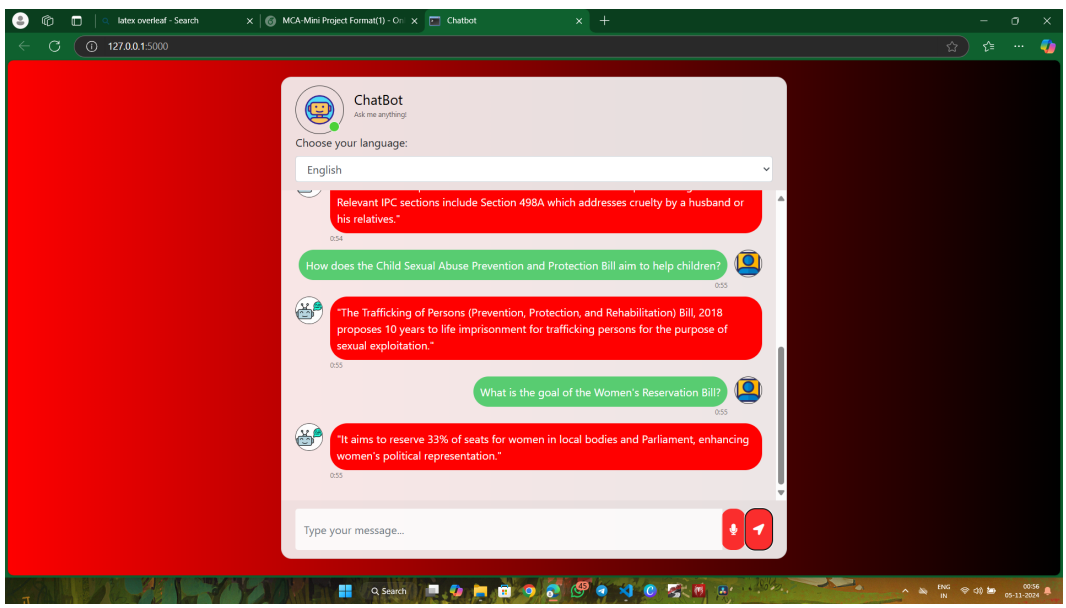


Fig. 10.3: Legal suggestions

10.1.4 Voice Input

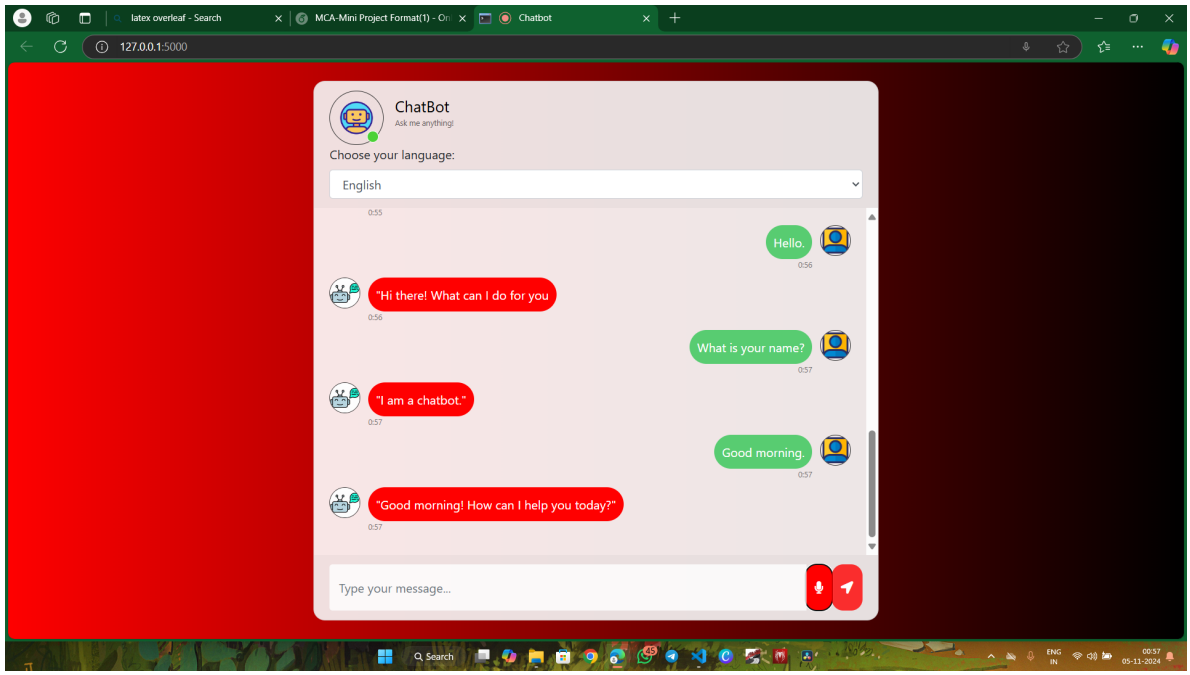


Fig. 10.4: Voice Integration

Chapter 11

SAMPLE CODES

11.1 Frontend

```
<body>
    <div class="container-fluid h-100">
        <div class="row justify-content-center h-100">
            <div class="col-md-8 col-xl-6 chat">
                <div class="card">
                    <div class="card-header msg_head">
                        <div class="d-flex bd-highlight">
                            <div class="img_cont">
                                <img src="https://cdn4.iconfinder.com/data/icons/chatbot-robot-1/100/chatbot_01_01_03_bot_chat_robot_app_artificial_chatbot_dialog-1024.png" c
                                <span class="online_icon"></span>
                            </div>
                            <div class="user_info">
                                <span>ChatBot</span>
                                <p>Ask me anything!</p>
                            </div>
                        </div>
                        <div class="lang_select">
                            <label for="language">Choose your language:</label>
                            <select id="language" class="form-control">
                                <option value="en">English</option>
                                <option value="es">Spanish</option>
                                <option value="fr">French</option>
                                <option value="de">German</option>
                                <option value="zh-cn">Chinese</option>
                                <option value="hi">Hindi</option>
                                <option value="ml">Malayalam</option>
                                <!-- Add more languages as needed -->
                            </select>
                        </div>
                    </div>
                    <div id="messageFormeight" class="card-body msg_card_body"></div>
                    <div class="card-footer">
                        <form id="messageArea" class="input-group">
                            <input type="text" id="text" name="msg" placeholder="Type your message..." autocomplete="off" class="form-control type_msg" required/>
                            <div class="input-group-append">
                                <button type="button" id="voiceInput" class="input-group-text send_btn"><i class="fas fa-microphone"></i></button>
                                <button type="submit" id="send" class="input-group-text send_btn"><i class="fas fa-location-arrow"></i></button>
                            </div>
                        </form>
```

Fig. 11.1: Frontend-Design

11.2 Backend

```python
from flask import Flask, render_template, request, jsonify    "jsonify": Unknown word.
import pickle
import nltk
from nltk.corpus import stopwords    "stopwords": Unknown word.
from nltk.tokenize import word_tokenize
from googletrans import Translator    "googletrans": Unknown word.

# Ensure necessary NLTK data is downloaded
nltk.download('punkt_tab')    "punkt": Unknown word.
nltk.download('stopwords')    "stopwords": Unknown word.

# Load the saved model and vectorizer    "vectorizer": Unknown word.
with open('chatbot_model.pkl', 'rb') as f:
    vectorizer, model = pickle.load(f)    "vectorizer": Unknown word.

# Initialize Flask app
app = Flask(__name__)

# Initialize the Translator
translator = Translator()

@app.route("/")
def index():
    return render_template('chatbot.html')

@app.route('/get_greeting', methods=['GET'])
def get_greeting():
    return "Hello, I'm a chatbot! How can I assist you today?"

@app.route("/get", methods=["POST"])
def chat():
    msg = request.form["msg"]
    #target_language = request.form.get("lang", "en")  # Default to English if no language is specified

    # Get the response from the chatbot model
    response = get_Chat_response(msg)

    # Translate the response to the target language
    #try:
    #translated_response = translator.translate(response, src='en', dest=target_language).text
    #except Exception as e:
    #print("Translation error:", e)
```

Fig. 11.2: Backend

11.3 Training Model

```python
import nltk
from nltk.corpus import stopwords    "stopwords": Unknown word.
from nltk.tokenize import word_tokenize
from sklearn.feature_extraction.text import TfidfVectorizer    "Tfidf": Unknown word.
from sklearn.linear_model import LogisticRegression
import numpy as np
import pandas as pd
import pickle

# Download necessary NLTK data
nltk.download('punkt_tab')    "punkt": Unknown word.
nltk.download('stopwords')    "stopwords": Unknown word.

# 1. Data Collection from CSV
# Load the data from a CSV file
data = pd.read_csv('chatbot_data.csv')  # Replace 'chatbot_data.csv' with your actual file name

# Extract the questions and responses
questions = data['Questions'].tolist()
responses = data['Answers'].tolist()

# 2. Preprocessing Function
def preprocess_text(text):
    stop_words = set(stopwords.words('english'))    "stopwords": Unknown word.
    tokens = word_tokenize(text.lower())
    filtered_tokens = [word for word in tokens if word.isalnum() and word not in stop_words]
    return ' '.join(filtered_tokens)

# Preprocess questions
preprocessed_questions = [preprocess_text(q) for q in questions]

# 3. Feature Extraction
vectorizer = TfidfVectorizer()    "vectorizer": Unknown word.
X = vectorizer.fit_transform(preprocessed_questions)    "vectorizer": Unknown word.
y = np.array(responses)

# 4. Model Training
model = LogisticRegression()
model.fit(X, y)

# Save the model and vectorizer    "vectorizer": Unknown word.
with open('chatbot_model.pkl', 'wb') as f:
    pickle.dump((vectorizer, model), f)    "vectorizer": Unknown word.
```

Fig. 11.3: Training Model

11.4 Dataset/Training Data

Fig. 11.4: Data-Collection

Chapter 12

APPLICATIONS

- **Judicial Support :**The chatbot can provide quick access to case summaries, judicial precedents, and essential information about the judicial system, making it useful for legal professionals, researchers, and the general public. By summarizing court rulings and legal judgments, the chatbot offers accessible insights into complex cases. This can aid in preliminary research, case preparation, and understanding case law trends, saving time for legal practitioners and judicial staff.

- **Educational Purpose:** Universities and law schools can utilize the chatbot as a dynamic learning aid for students studying law. "Lawyer-Genie" can simulate legal scenarios, provide definitions of legal terms, and answer queries based on actual statutes and case law, allowing students to apply theoretical knowledge practically. Law professors can also integrate the chatbot into coursework to enhance students' research skills, familiarize them with legal processes, and make learning more interactive and engaging.

- **Government Legal Portals:** Government websites and legal portals can integrate the chatbot to help citizens navigate complex legal documents, understand their rights, and obtain information on government regulations and services. The chatbot can simplify language in legal documents, guide users through government forms, and clarify the steps for submitting complaints or filing requests, reducing dependency on in-person consultations and easing the workload of government employees.

Chapter 13

ADVANTAGES AND DISADVANTAGES

13.1 ADVANTAGES

- **24/7 Availability:** Users can access legal help anytime, making it convenient for those who need assistance outside regular office hours.
- **Accurate Responses:** The AI ensures that users receive reliable and accurate legal information tailored to their needs.
- **Cost-effective and Reliable :** The service is cost-effective and reliable.

13.2 DISADVANTAGES

- **High computational requirements:**Chatbot requires significant processing power and memory due to its advanced machine learning and language processing.
- **Limited Coverage of Legal Topics:**While Chatbot covers many legal topics, it may not answer all legal questions due to dataset gaps or limited access to certain legal documents.
- **False positives:** Due to NLP limitations, Chatbot might misunderstand user queries.
- **Accessibility Barriers:** Users without legal backgrounds may find it challenging to understand chatbot responses if complex terms are used without simplification

Chapter 14

FUTURE SCOPE

The future scope of this project includes several enhancements to improve functionality, user experience, and accessibility. Planned developments include:

- EmergencySupportIntegration:Addingemergencybuttons(Police,Ambulance, Fire Service) and helpline contacts for urgent assistance.
- Advanced NLP and Legal Consultation: Incorporating more sophisticated natural language processing techniques for enhanced accuracy in understanding legal queries and mapping them to appropriate legal advice. This will include deeper integration with legal consultation services.
- Enhanced Voice Interaction: Expanding voice interaction capabilitiesto make the chatbot more accessible, especially for users who may prefer speaking over typing.
- Resource Expansion: Providing a downloadable version of the Indian penal Code (IPC) book for easy offline access.
- Lawyer Contact Integration: Incorporating a directory of lawyer contacts filtered by location to aid users in finding legal professionals nearby for further consultation.

Chapter 15

CONCLUSION

The development of the Law-Based Chatbot, "Lawyer-Genie," demonstrates a significant advancement in providing accessible, user-friendly, and efficient legal support. By using AI-powered interfaces, the chatbot offers a streamlined platform for users to access personalized legal advice, inquire about their rights, and find relevant laws without needing a traditional consultation. This system benefits individuals by simplifying the process of locating legal resources and empowers legal professionals to focus on more complex cases by reducing time spent on repetitive inquiries. Ultimately, this project enhances user interaction and promotes accessibility to legal guidance, thereby contributing to a more efficient and informed user experience.

References

[1] Nikita, E. Srivastav, A. Patel, A. Singh, R. Sharma, D. P. Rana, and R. G. Mehta, "Lawbot: A smart user indian legal chatbot using machine learning framework," in *2024 IEEE 9th International Conference for Convergence in Technology (I2CT)*, 2024, pp. 1–7.

[2] V. Socatiyanurak, N. Klangpornkun, A. Munthuli, P. Phienphanich, L. Kovudhikulrungsri, N. Saksakulkunakorn, P. Chairaungsri, and C. Tantibundhit, "Law-u: Legal guidance through artificial intelligence chatbot for sexual violence victims and survivors," *IEEE Access*, vol. 9, pp. 131 440–131 461, 2021.

[3] S. Khazaeli, J. Punuru, C. Morris, S. Sharma, B. Staub, M. Cole, S. Chiu-Webster, and D. Sakalley, "A free format legal question answering system," in *Proceedings of the Natural Legal Language Processing Workshop 2021*, 2021, pp. 107–113.

[4] M. Wyawahare, S. Roy, and S. Zanwar, "Generative vs intent-based chatbot for judicial advice," in *2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*, vol. 2, 2024, pp. 1–6.

[5] K. R A, K. S, and R. R, "Legal solutions - intelligent chatbot using machine learning," in *2023 Intelligent Computing and Control for Engineering and Business Systems (ICCEBS)*, 2023, pp. 1–6.