Legal Act and IPC Section Prediction System Using Complaint Text Analysis

A PROJECT REPORT

Submitted by

RAJA RAJESWARI R (22IT072)

NIVASSHINI R (22IT063)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

THIAGARAJAR COLLEGE OFENGINEERING, MADURAI–15

(A Govt. Aided, Autonomous Institution, Affiliated to Anna University)

ANNAUNIVERSITY: CHENNAI 600025

May 2025

THIAGARAJAR COLLEGE OF ENGINEERING, MADURAI-15

(A Govt. Aided, Autonomous Institution, Affiliated to Anna University)

BONAFIDECERTIFICATE

Certified that this project report "***Legal Act and IPC Section Prediction System Using Complaint Text Analysis***" is the bonafide work of "**RAJA RAJESWARI R (22IT072), NIVASSHINI R (22IT063)"** who carried out the project work under my supervision during the Academic Year 2024-25 (EVEN).

SIGNATURE
Dr. C. DEISY,
HEAD OF THE DEPARTMENT &
PROFESSOR

INFORMATION TECHNOLOGY
THIAGARAJAR COLLEGE OF
ENGINEERING,
MADURAI-15.

SIGNATURE
Mrs. S. PUDUMALAR,
SUPERVISOR &
ASSISTANT
PROFESSOR
INFORMATION TECHNOLOGY
THIAGARAJAR COLLEGE OF
ENGINEERING,
MADURAI-15.

Submitted for the VIVAVOCE Examination held at Thiagarajar College of Engineering on_____

INTERNAL EXAMINER **EXTERNALEXAMINER**

ACKNOWLEDGEMENT

We express our sincere gratitude to our Honorable Correspondent **Mr. K. HARI THIAGARAJAN** for facilitating with a good learning environment to improve our academic knowledge and performance at this premier institution.

We wish to express our profound gratitude to our beloved Principal **Dr. L. ASHOK KUMAR** for his overwhelming support provided during our course span in this institution.

We wish to express the gratitude to our beloved Head of the Department of Information Technology **Dr.C.Deisy,** for her motivation and support during the course span. We are grateful to our project supervisor **Mrs.S.Pudumalar** for her guidance to complete the project and led us from the scratch of this project's commencement with undivided attention and skillful expertise in developing this project in a systematic and professional manner.

We wish to express our thanks to the project review members and other faculty members of Department of Information Technology for their valuable suggestions and encouragement periodically. We express our sincere thanks to all the lab technicians for their constant and dedicated services to help us all the time.

We thank God Almighty, our parents and friends for helping us to work in a challenging, an interesting project, and in completing the same in due course without much difficulty.

RAJA RAJESWARI R (22IT072)

NIVASSHINI R (22IT063)

ABSTRACT

The **Legal-AI Assistant** is an intelligent, multilingual system designed to simplify access to the Indian Penal Code (IPC) by interpreting user complaints and returning the most relevant legal sections, accompanied by easy-to-understand summaries. Developed using advanced Natural Language Processing (NLP) techniques and powered by a **fine-tuned Google Gemini AI model**, the system bridges the gap between complex legal documents and everyday user concerns. Whether for students, citizens, or legal aid centers, this assistant acts as a bridge between the law and the layperson, making the legal system more accessible, transparent, and empowering.

Key features of the system include:

- **Semantic Understanding:** Goes beyond mere keyword matching to semantically understand the user's complaint and recommend the most relevant IPC sections.
- **Contextual Answers:** Generates simplified, explanatory responses based on the selected IPC sections using fine-tuned generation models.
- **Multilingual Support:** Detects the user's preferred language (English, Hindi, Tamil) and delivers responses accordingly, promoting legal accessibility for diverse linguistic groups.
- **Fine-tuned Gemini AI Integration:** Utilizes a customized Gemini AI model fine-tuned specifically for the Indian Penal Code dataset, ensuring higher precision and relevance.
- **Scalability & Efficiency:** Optimized for fast response times, offline deployment capability, and seamless integration into mobile applications, web portals, or legal kiosks.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVATIONS

S.NO	Abbreviation	Full Form
1	NLP	Natural Language Processing
2	IPC	Indian Penal Code
3	DSS	Decision Support System
4	TF-IDF	Term Frequency-Inverse Document Frequency
5	ECHR	European Court of Human Rights
6	QA	Question Answering
7	AI	Artificial Intelligence

1.INTRODUCTION

The **Indian Penal Code (IPC)** is one of the most critical pieces of legislation in the Indian criminal justice system. First enacted in **1860**, it was drafted by the **Indian Law Commission** under the leadership of **Lord Macaulay**. Over time, the IPC has undergone various amendments and revisions to address the evolving nature of criminal activities, technological advancements, and societal shifts. As the cornerstone of criminal law in India, the IPC not only defines various offenses but also prescribes corresponding punishments and judicial procedures for dealing with crimes. The Act continues to serve as a guide for law enforcement, judiciary, and legal practitioners, influencing how justice is dispensed in India.

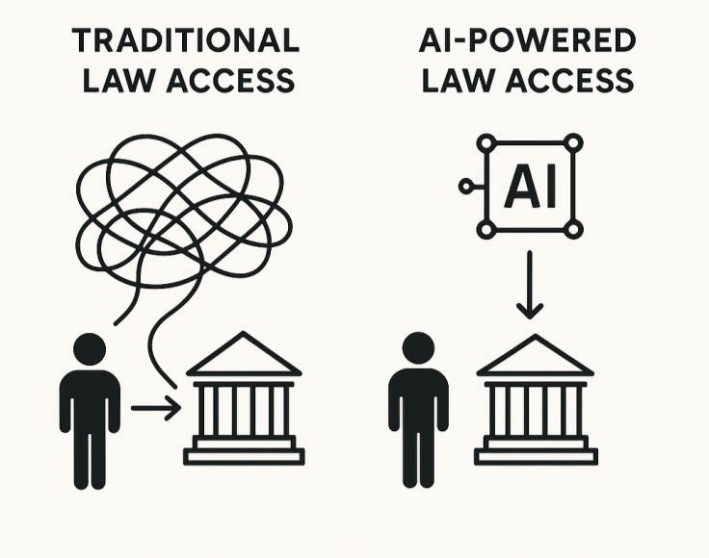
The significance of the IPC lies not only in its vast coverage of criminal offenses but also in its adaptability to changing times. The legal principles enshrined in the IPC address issues that are relevant to both individual citizens and society at large. Crimes such as **murder**, **theft**, **rape**, and **corruption** are examples of offenses that the IPC defines in great detail, ensuring that justice is served for each specific case. Additionally, the law contains provisions for **preventive measures**, **punishments**, and **proceedings** to deal with those who commit crimes, while safeguarding the rights of accused individuals and ensuring fairness.

The importance of making the **Indian Penal Code** easily accessible to the general public cannot be overstated. Many people find it challenging to navigate the extensive legal jargon and long text of the IPC. Moreover, even legal professionals often require time-consuming efforts to locate specific sections, comprehend their application, and relate them to case law. With the increasing reliance on technology in daily life, it has become crucial to modernize the way legal content is disseminated, making it more accessible, interactive, and user-friendly.

1.1 GENERAL OVERVIEW

The increasing complexity of legal documentation and the language barriers it poses to the general public have long been significant challenges within India's legal system. Bridging this gap is critical for ensuring justice and empowering citizens. This project introduces an intelligent, multilingual Legal-AI Assistant that interprets natural language queries and returns relevant, understandable IPC sections in English and Tamil.

Built upon modern AI architectures and leveraging a **fine-tuned Google Gemini AI model** trained on IPC datasets, the system transforms the way individuals interact with legal knowledge by enabling semantic understanding, real-time similarity search, and user-friendly natural language generation.



1.1 Traditional Law Access Vs AI Powered Law Access

1.2 MOTIVATION

Access to legal aid in India, especially in remote and underserved regions, remains limited. Although the Indian Penal Code is publicly available, interpreting its complex language without professional assistance is a significant barrier. With the evolution of transformer-based NLP models, large generative models like Gemini, and open legal datasets, there is a golden opportunity to democratize legal knowledge.

The motivation behind this project is to **leverage AI and fine-tuned generative models for public welfare**, empowering individuals to understand their rights and navigate legal issues confidently, in their preferred language, without needing legal expertise.

1.3 BASIC IDEA OF THE PROJECT

The primary goal of this project is to provide an organized, comprehensive, and accessible version of the IPC that includes the following features:

- **Detailed Sections**: Each section of the IPC will be presented with its offense description and prescribed punishment.
- **Categorization**: Offenses will be categorized into various groups such as property crimes, personal crimes, and offenses against the state.
- **Search Functionality**: A search tool will allow users to quickly locate specific sections by keywords or section numbers.
- **User Interface**: The system will feature a simple and intuitive interface for easy navigation, including a glossary of legal terms, section summary, and links to related sections.

1.4 NEED FOR A COMPREHENSIVE IPC DATABASE

In today's digital era, access to legal resources is essential for practitioners and law students. Traditional legal texts, while reliable, are often lengthy and difficult to navigate. This project aims to bridge the gap by providing a digital, searchable, and well-organized version of the IPC. Legal professionals and students can save time in research, enabling quick access to relevant sections and understanding the offense-punishment relationship. This system also helps to raise awareness about the IPC among citizens, ensuring that they are informed about the legal consequences of criminal behavior.

1.5 OVERVIEW OF THE PROPOSED SYSTEM

The system will be a web-based platform that provides an interactive database of the Indian Penal Code. Users will be able to:

- **Browse Sections**: Each section will have its own page with offense descriptions and prescribed punishments.
- **Advanced Search**: Users can search for offenses based on keywords or section numbers.
- **Category View**: The IPC sections will be divided into categories (e.g., crimes against property, crimes against the state, etc.) for easy browsing.
- **Explanatory Notes**: Legal explanations and references to case law will be available for deeper understanding.

Additionally, the project will incorporate features like bookmarking important sections, viewing related sections, and providing links to case laws that have interpreted the sections.

1.6 COMPARISON WITH OTHER LEGAL DATABASES

Various legal databases exist today, but they often lack the interactivity, user-friendly interface, and ease of navigation that the proposed system aims to offer. The following comparison provides a better understanding of how this system stands out:

TABLE 1.6.1 COMPARISON WITH OTHER LEGAL DATASETS

Feature	**Traditional Legal Database**	**Proposed IPC System**
Search Functionality	Limited or non-existent	Advanced search based on keywords, section numbers
Category View	Usually unstructured	Sections categorized for easy browsing
User Interface	Often text-heavy and complex	Simple, intuitive web interface
Legal Explanations	Often limited to legal textbooks or comments	Integrated explanations and references to case laws
Updates	Requires manual updates	Easily updatable with real-time changes

The proposed IPC system offers several key advantages over traditional legal databases. While traditional databases often rely on static text and require manual updates, the proposed system is dynamic, with real-time updates and continuous improvements. Unlike conventional platforms, which may present legal information in a complex format, this system provides a simple, intuitive interface, ensuring accessibility even for non-legal users. Additionally, it integrates comprehensive explanations and links to relevant case law, offering users a deeper understanding of each provision. This ease of navigation, coupled with advanced search functionality, ensures a more efficient and user-friendly experience.

1.7 Bridging Gaps with the Digital IPC System

The proposed system enhances the accessibility and usability of the Indian Penal Code (IPC), addressing the limitations of traditional resources like books and static PDFs. Unlike these traditional formats, the system offers a structured, searchable interface, enabling users to quickly locate specific sections and provisions. It not only provides the legal text but also includes detailed descriptions and case law references, helping users understand the provisions fully.

The system's user-friendly interface eliminates the need to navigate through complex legal language. Real-time search functionality allows for quick access based on keywords, section numbers, or specific offenses. Additionally, the system offers insights into the consequences of crimes under the IPC, including penalties and punishments, helping users understand the severity of offenses.

This system also raises awareness about various offenses under the IPC, empowering individuals to understand their rights and legal obligations. It provides accessible, comprehensive legal information for the public, ensuring people are better equipped to navigate legal issues with confidence. By making legal knowledge accessible to both professionals and the public, the system promotes legal literacy and transparency in legal processes.

In summary, the proposed IPC system transforms how people engage with criminal law. Its intuitive interface, advanced search capabilities, and educational features offer a more informed approach to legal knowledge, supporting both legal professionals and everyday users.

1.8 Applications and Future Prospects

Impact and Reach: This system will significantly impact the legal community, law students, and individuals seeking legal knowledge. It will be an invaluable resource for those looking to understand the IPC without wading through cumbersome legal texts. Moreover, its adaptability and ease of use will make it the go-to reference for law enforcement, legal practitioners, and educators.

Key Use Cases:

1. **Legal Professionals**: Lawyers and judges will benefit from quick access to the exact IPC sections related to ongoing cases, helping them make informed decisions faster.
2. **Law Students**: The system will serve as an essential tool for students studying criminal law, offering clear explanations and summaries of each section.
3. **General Public**: By providing a clear overview of the IPC, this platform aims to educate citizens about their rights and the legal consequences of criminal actions, promoting legal literacy.
4. **Research**: Scholars can use the platform to gather data on criminal law trends and case law interpretations.

Future Prospects:

1. **Mobile App**: The project could expand into a mobile application, ensuring that users can access legal information on the go.
2. **Integration with Legal Databases**: The platform could integrate with national legal databases to provide real-time updates on changes in the IPC and other laws.
3. **Multilingual Support**: To ensure wider accessibility, the system could offer multilingual support, allowing users across India to access the IPC in their preferred languages.

2.LITERATURE REVIEW

2.1 INTRODUCTION

Legal technology, or **LegalTech**, has emerged as a powerful domain that integrates traditional legal practice with innovations in artificial intelligence (AI) and natural language processing (NLP). With the rapid evolution of machine learning models, new systems have been developed to simplify legal research, case analysis, document summarization, and user query resolution. Modern LegalTech solutions aim to make legal knowledge more accessible, efficient, and intelligible, especially for non-expert users.

This chapter explores prior work in the areas of semantic search, legal information systems, multilingual AI, and the deployment of conversational AI in law. It also identifies key limitations in existing systems and highlights how the proposed model builds upon and improves these foundations.

2.2 EXISTING SYSTEMS AND TOOLS

Several legal information retrieval systems, digital portals, and chatbot frameworks have been developed over the past decade. However, many of them face challenges related to semantic understanding, language support, flexibility, and scalability. Some prominent systems are:

- **Indian Kanoon:** A widely used legal search platform in India. It enables keyword-based search for judgments and legal provisions but lacks deeper semantic understanding or multilingual capabilities.
- **Chatbots in Law Firms:** Some modern law firms deploy rule-based chatbots to handle basic client queries. These bots are usually limited in scope, offering pre-programmed answers without deeper contextual understanding.

- **Government Portals (eCourts, Digital India Legal Services):** These official platforms provide case status tracking, legal document access, and other citizen services but are largely non-interactive and require manual navigation through legal text.

While these tools have significantly improved access to legal information, they often lack **semantic reasoning**, **interactive adaptability**, and **regional language accessibility**. There remains a clear need for an AI-powered legal assistant that understands natural language, provides context-driven answers, and supports multilingual communication.

TABLE 2.2.1 EXISTING SYSTEM AND TOOLS

Tool	**Technology Used**	**Key Features**	**Limitations**
ROSS Intelligence	IBM Watson AI, NLP	Legal research, case law search	Limited scope in legal practice areas
Casetext	NLP, Machine Learning	Legal research, document analysis	Requires significant user input
Legal Robot	Machine Learning, NLP	Legal document analysis, readability scoring, risk assessment	Focused only on contract and document analysis
Lex Machina	AI, Legal Analytics	Litigation insights, case predictions	Limited to specific jurisdictions and cases

2.3 REVIEW OF ADDITIONAL RESEARCH CONTRIBUTIONS

To further strengthen the foundation of this work, several research papers and projects in legal AI, IPC section classification, semantic similarity, and legal question-answering were reviewed. These studies help identify emerging trends, best practices, and current gaps.

TABLE 2.3.1: LITERATURE SURVEY

Sl. No	Title	Authors / Affiliations	Journal Name / Year	Observations
1	Text Similarity Algorithms to Determine Indian Penal Code Sections for Offence Report	Ambrish Srivastav, Shaligram Prajapat	International Journal of Artificial Intelligence, Vol. 11, No. 1, March 2022	DSS using NLP and similarity metrics like TF-IDF, cosine similarity, and embeddings for IPC mapping.
2	LegalReasoner: A Multi-Stage Framework for Legal Judgment Prediction via Large Language Models and Knowledge Integration	Xuran Wang et al., University of Pennsylvania, Georgia Tech, UT Dallas	IEEE Access, 2023	Enhances judgment prediction using knowledge infusion, retrieval, and multi-hop reasoning, improving accuracy by 7.8%.
3	A Survey of Text Similarity Approaches	Wael H. Gomaa, Aly A. Fahmy	International Journal of Computer Applications, Volume 68 – No.13, April 2013	Covers traditional and modern methods for computing textual similarity using NLP and deep learning.

4	JusticeAI: A Multimodal System for Judicial Rulings in Smart Courts	Nagwan Abdel Samee et al.	IEEE Access, 2024	Uses CDKF architecture with ECHR dataset, achieving 83% accuracy in predicting court rulings.
5	A Novel Method of Indian Penal Code Section Classification for Offence Incident Report	Ambrish Srivastav, Shaligram Prajapat	IEEE Access,2022	Develops a new IPC corpus and uses hybrid NLP methods to improve IPC classification.
6	A Two-Staged NLP-Based Framework for Assessing Sentiments on Indian Supreme Court Judgments	Isha Gupta, Indranath Chatterjee, Neha Gupta	International Journal of Information Technology,2013	Uses LDA and sentiment analysis to assess public opinion on court judgments.
7	IPC Chatbot: A Chatbot for Indian Penal Code	Murugesan J, Dinesh M, et al.	IJCRT,2022	Provides IPC-related information using static CSV databases without true NLP or semantic translation.
8	Keyword-based Closed Domain Question Answering System for IPC Sections and Amendment Laws	Rohini P. Kamdi, Avinash J. Agrawal	International Journal of Intelligent Systems Technologies and Applications	QA system relying on structured queries; achieves 94% precision but lacks semantic depth.

2.4 INSIGHTS AND GAPS IN EXISTING WORK

Upon analyzing the above tools and research, several critical insights and gaps are identified:

- **Semantic Understanding vs. Keyword Matching:** Most existing systems are heavily keyword-driven, limiting their ability to truly understand user intent expressed in natural language.
- **Multilingual Barriers:** Tools like Indian Kanoon and IPC Chatbots operate only in English, which excludes a large section of the Indian population more comfortable in regional languages like Hindi, Tamil, Bengali, etc.
- **Lack of Contextual Generation:** Very few solutions attempt to summarize user queries or generate context-aware responses. Most merely retrieve exact matches or predefined templates.
- **Static Interaction:** Most existing legal bots and search engines offer limited interactivity. There is no ongoing dialogue based on previous queries or personal customization based on user context.
- **Underutilization of Retrieval-Augmented Generation (RAG):** The use of retrieval-enhanced models like RAG for combining search results with generative responses is still rare in the legal AI domain.

2.5 HOW THE PROPOSED SYSTEM BUILDS UPON PRIOR WORK

The proposed system seeks to bridge the identified gaps through an integrated, modular, and scalable AI framework tailored for Indian legal complaints. It offers several innovations:

- **Summarization-First Approach:** Upon receiving the user's complaint or query, the system first uses **DistilBART**, a powerful transformer-based summarization model, to condense the input into a clear, precise abstract that preserves key legal elements. This ensures that even lengthy or confusing user inputs are normalized into a standard format.
- **Fine-Tuned Legal Reasoning Model (Gemini AI):** The summarized abstract is then passed to a **fine-tuned Gemini AI model**, trained specifically on IPC-related data. The model predicts the **top five most relevant IPC sections** corresponding to the facts of the case.
- **Semantic Search and Indexing:** Instead of keyword matching, the system uses **Sentence Transformers** and **FAISS vector indexing** to perform semantic similarity search, improving the retrieval of related legal clauses based on user intent.
- **Contextual Answer Generation:** To further enhance accessibility, the system uses **T5-small** models for generating user-friendly, contextualized explanations of the recommended IPC sections.
- **Multilingual Support:** Using **Facebook's NLLB-200 model**, the system provides translation of IPC sections and generated explanations into Tamil (and potentially other Indian languages), making it accessible to a wider audience.
- **Dynamic and Scalable Architecture:** Designed to be modular, the system allows the addition of new legal domains (e.g., Consumer Protection Act, IT Act) and new languages, along with future integration of voice interfaces.

3.PROBLEM DESCRIPTION

3.1 PROBLEM STATEMENT

In today's intricate legal landscape, millions of individuals—particularly those from rural, marginalized, or economically disadvantaged backgrounds—face significant barriers in understanding and exercising their legal rights. Although the Indian Penal Code (IPC) is publicly available, its complex legal jargon, archaic structure, and lack of simplified explanations render it inaccessible to a vast majority of the population. This inaccessibility not only widens the justice gap but also perpetuates systemic inequalities in legal awareness and empowerment.

Existing legal assistance tools predominantly rely on static, keyword-based search mechanisms, which suffer from critical limitations. These systems fail to capture the semantic meaning behind user inputs and cannot effectively process queries expressed in natural, conversational language. As a result, users—especially first-time seekers of legal information—often receive irrelevant, incomplete, or misleading responses. Additionally, these platforms seldom support regional languages, further alienating non-English-speaking users and exacerbating the digital divide.

Problem at Hand:

This project addresses the urgent need for a dynamic, intelligent legal assistance system capable of interpreting user-submitted complaints or queries expressed in natural language and mapping them accurately to the most relevant IPC sections. Present-day legal search platforms are neither multilingual nor context-sensitive, resulting in poor user experience and limited reach.

To bridge this critical gap, the project introduces the **Legal-AI Assistant**—an advanced system that seamlessly integrates **semantic search**, **contextual response generation**, and **multilingual translation**. By leveraging cutting-edge **transformer-based Natural Language Processing (NLP) models** and **Retrieval-Augmented Generation (RAG)** frameworks, the Legal-AI Assistant aims to make legal information more understandable, accessible, and inclusive. Initially supporting English and Tamil languages, it seeks to empower a wider demographic, ensuring that legal knowledge transcends linguistic and technological barriers.

3.2 OBJECTIVES OF THE PROJECT

To address the identified challenges and bridge the access gap, the project is structured around the following key objectives:

3.2.1 Develop a Semantic Legal Query Interpretation System

- Design and implement an intelligent system capable of understanding diverse user complaints or queries expressed in natural, conversational language.
- Map user queries accurately to the most appropriate IPC sections using advanced transformer-based NLP models.
- Employ **FAISS (Facebook AI Similarity Search)** for efficient semantic indexing and retrieval of relevant legal sections based on query embeddings.

3.2.2 Enable Contextual and Simplified Legal Response Generation

- Implement **Retrieval-Augmented Generation (RAG)** methodologies to generate simplified, user-friendly explanations based on retrieved legal content.

- Summarize complex legal terminologies into plain, understandable language suitable for non-specialists.
- Provide actionable guidance and structured legal information relevant to user-submitted issues.

3.2.3 Ensure Multilingual Accessibility

- Integrate **high-quality translation models** such as **NLLB-200** (No Language Left Behind) to support English and Tamil outputs.
- Maintain semantic integrity during translation to ensure legal meanings are preserved across languages.
- Promote linguistic inclusivity by designing systems adaptable for future multilingual expansions.

3.2.4 Design a Modular, Scalable, and Extensible Architecture

- Develop a robust, modular backend architecture supporting **API-based integration** with web portals, mobile apps, and chatbot interfaces.
- Ensure horizontal and vertical scalability to accommodate increasing datasets, additional laws (e.g., CrPC, Consumer Protection Act), and new languages.
- Build system resilience through continuous model retraining, performance optimization, and user feedback mechanisms.

3.2.5 Promote Legal Awareness and Empower Citizens

- Offer an interactive, reliable AI-based legal assistant fostering self-advocacy and empowering users to understand and assert their legal rights.
- Facilitate legal literacy among rural and marginalized populations, reducing dependency on intermediaries or expensive legal consultations.

3.3 CONTRIBUTION TO SDG 16: PEACE, JUSTICE, AND STRONG INSTITUTIONS

The Legal-AI Assistant directly contributes to **Sustainable Development Goal (SDG) 16**, which focuses on promoting peaceful and inclusive societies, ensuring access to justice for all, and building effective, accountable institutions at all levels.

Through simplified legal interpretations and contextually relevant responses in English and Tamil, the system empowers individuals to:

- Recognize, understand, and assert their fundamental legal rights.
- Navigate complex legal scenarios independently or with minimal assistance.
- Seek appropriate legal remedies based on informed decision-making.

By reducing barriers to legal information, the Legal-AI Assistant fosters greater public trust in judicial institutions and supports community-level legal empowerment. Ultimately, it contributes to the vision of an inclusive, equitable, and just society where legal rights are universally understood and respected.

3.4 CONTRIBUTION TO SDG 4: QUALITY EDUCATION

The project also significantly advances **Sustainable Development Goal (SDG) 4**, which emphasizes inclusive and equitable quality education for all.

As an educational tool, the Legal-AI Assistant offers the following benefits:

- **Law Students and Interns**: Access to simplified interpretations of IPC sections contextualized within real-world scenarios, enhancing practical legal education.

- **Educators**: Opportunity to incorporate AI-driven legal tools into academic syllabi, promoting interdisciplinary learning across law, technology, and linguistics.
- **Regional Learners**: Enabling bilingual legal education in English and Tamil, thus promoting broader engagement and understanding among students from diverse linguistic backgrounds.

The interactive design also supports **online legal literacy campaigns**, **digital learning modules**, and **AI-Law innovation projects**, fostering next-generation digital legal education.

3.5 CONTRIBUTION TO SDG 10: REDUCED INEQUALITIES

This project profoundly contributes to **Sustainable Development Goal (SDG) 10**, aimed at reducing inequality within and among countries.

By offering multilingual interfaces and context-aware explanations, the Legal-AI Assistant specifically benefits:

- **Non-English-Speaking Populations**: Breaking language barriers that traditionally exclude large sections of society from accessing legal knowledge.
- **Rural and Economically Marginalized Communities**: Providing free or low-cost access to quality legal assistance where professional services are limited or unaffordable.
- **Underrepresented Groups**: Empowering women, minorities, and low-income individuals with knowledge that can protect their rights and support their socio-economic mobility.

Innovation in Legal Accessibility:

The project exemplifies how AI-driven innovations can fundamentally transform public legal services, moving beyond traditional static and often inaccessible repositories of information to create dynamic, interactive, and citizen-centric platforms. These platforms are designed not only to provide accurate legal information but also to actively engage with users, adapting to their changing needs and ensuring that legal support remains timely, relevant, and accessible to all sections of society. Through intelligent automation, personalized guidance, and multilingual support, the system bridges the gap between complex legal frameworks and the everyday realities of citizens, making the law more approachable and actionable.

At the heart of this transformation is a commitment to human-centered legal technology. By embedding simplicity, transparency, and empathy into every aspect of system design, the Legal-AI Assistant prioritizes the lived experiences of users over technical complexity. This approach ensures that people from diverse backgrounds—regardless of their literacy level, language preference, or technological familiarity—can access and navigate legal information with confidence and dignity. In doing so, the project not only enhances individual legal empowerment but also reinforces social equity and strengthens the foundations of participatory democracy, where every citizen can claim their rights and seek justice without barriers.

4.METHODOLOGY

4.1 OVERVIEW

The methodology section outlines the comprehensive steps followed to design, develop, and deploy the intelligent IPC legal advisory system. It integrates Natural Language Processing (NLP), Google Gemini AI (v1.5 Flash), and advanced summarization models to process user complaints and accurately generate the most relevant Indian Penal Code (IPC) sections. provide a brief elaborate half page introduction for methodology

4.2 SYSTEM ARCHITECTURE:

The system is divided into three major components:

4.2.1. Frontend: Web-Based User Interface Using Flask

The frontend, built with HTML, CSS, JavaScript, and Flask, provides a portal for query submission and result display.

Components:

- **Input Form**: Text input box for users to type their legal questions or issues in English or Tamil.
- **Prediction Button**: Submits the user query to the backend for processing.
- **Result Display**: Shows the predicted IPC section, a simplified abstract, and the original IPC description in the selected language.
- **Language Switch Option**: Allows users to choose between English and Tamil for output.

4.2.2. Backend: AI-Powered Legal Processing Engine

The backend, built using Flask and Python, manages the complete AI pipeline, from semantic understanding to multilingual response generation.

Components:

- **Sentence Embedding Engine**: Fine-tuned SentenceTransformer model computes embeddings for user queries and IPC section texts.
- **Semantic Search Engine**: FAISS (Facebook AI Similarity Search) library retrieves the closest IPC sections based on vector similarity.
- **Abstract Generator**: Fine-tuned T5-small model generates simplified summaries for easier comprehension.
- **Language Translator**: NLLB-200 model performs high-quality translation between English and Tamil.
- **Preprocessing and Normalization Module**: Cleans, tokenizes, and standardizes input queries for consistent AI performance.
- **Response Formatter**: Packages all outputs (predicted section, abstract, translated text) into a structured JSON response for the frontend.

4.2.3. Model Training and Fine-Tuning Pipeline

This offline process involves preparing datasets, fine-tuning pretrained models for domain-specific IPC understanding, and evaluating performance.

Components:

- **Dataset Preparation**: Custom-built dataset linking IPC sections with natural language queries.

- **SentenceTransformer Fine-Tuning**: Fine-tuned to maximize semantic retrieval accuracy between user questions and IPC descriptions.
- **T5 Model Fine-Tuning**: Trained to generate clear, legally accurate abstracts of IPC sections.
- **Model Evaluation and Validation**: Metrics like Top-1 accuracy, BLEU score (for summaries), and semantic similarity scores are used for performance evaluation.
- **Model Saving**: Best models are exported (.bin/.pt/.h5 format) and loaded into the production backend for real-time inference.

4.3 WORKFLOW

This section outlines the **step-by-step operation** of the Legal-AI Assistant, from user query input to legal response generation:

4.3.1 System Initialization

The Flask server is launched, loading:

- The fine-tuned SentenceTransformer model for embeddings.
- FAISS index containing IPC section embeddings.
- The fine-tuned T5 model for abstract generation.
- The NLLB-200 translation model for multilingual support.

The web interface becomes active and ready for user input.

4.3.2 User Submits a Query

- Users enter a question, statement, or scenario related to criminal law (e.g., "What happens if someone commits theft?").
- They select their desired output language (English or Tamil).

4.3.3: Data Preprocessing

- Cleans and normalizes the input text.
- Embeds the query using the SentenceTransformer model to produce a vector.

4.3.4: Semantic Search and Prediction

- FAISS retrieves the **top-k most similar** IPC sections to the query based on vector similarity.
- The system selects the highest-ranked IPC section as the primary prediction.

4.3.5: Abstract Generation

- The T5-small model generates a simplified abstract summary of the retrieved IPC section to make it easier for common users to understand.

4.3.6: Translation (If Needed)

- If the user requested Tamil output, the English response (both section and abstract) is translated into Tamil using the NLLB-200 model.

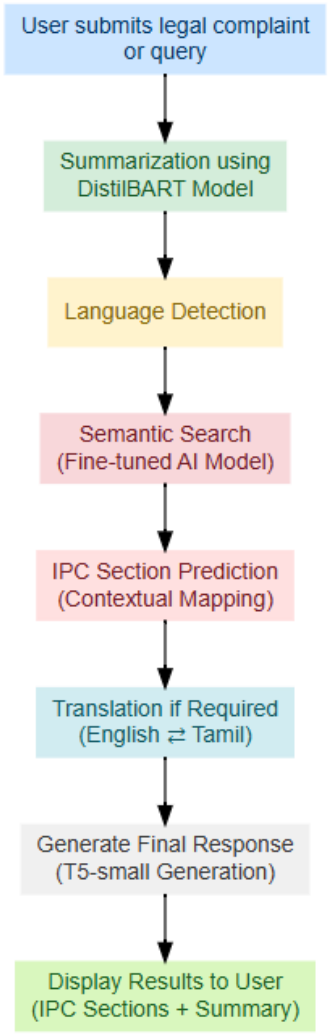
4.3.7: Display Results in the Frontend

- The user receives:
    - The relevant IPC section number and title.
    - The full IPC section description.
    - A simplified abstract for easier understanding.
    - The translated output (if Tamil is selected).
    - (Optional) A confidence score or semantic similarity score for transparency.

- Results are displayed in an elegant, user-friendly format with options to **make another query** or **download** the results.

4.3.8: End of Session

- After viewing the results, the user can:
  - Submit another query.
  - Change the output language.
  - End the session.



4.3.1 SYSTEM WORKFLOW

4.4 TECHNOLOGY STACK

A Flask-based web app integrating fine-tuned AI models for multilingual legal query understanding, semantic search, and translation.

TABLE 4.4.1 TECHNOLOGY STACK

Component	Technology Used
Frontend	HTML, CSS, Flask (Jinja Templates)
Backend	Python (Flask Framework)
Models	DistilBART, Fine-tuned Gemini, T5-small, NLLB-200
Semantic Search	SentenceTransformers + FAISS
Translation	NLLB-200 Multilingual Model
Database	Local JSON file (IPC Sections Dataset)
Model Framework	HuggingFace Transformers, PyTorch
Deployment	Localhost (Flask server)

5.IMPLEMENTATION

5.1 INTRODUCTION

This **section** describes the **structured** and **detailed process** undertaken to develop the intelligent IPC legal advisory system. This innovative system is designed to automatically analyze user complaint**s** and map them to the most **relevant sections** of the **Indian Penal Code (IPC)**, utilizing cutting-edge Artificial Intelligence (AI) technologies. The system combines the power of **Natural Language Processing (NLP)**, **Google Gemini AI** (**version 1.5 Flash**), and advanced summarization models to deliver accurate, contextually appropriate legal advice based on user inputs.

The process begins with the extraction and structuring of the IPC data into a machine-readable format, followed by fine-tuning the AI models to understand legal terminologies and complaint structures. A **multilingual complaint** intake mechanism ensures that complaints submitted in English, Tamil, and other Indian languages are accurately processed through robust language detection. **Summarization** models further distill lengthy and complex complaints into concise formats without losing critical information, thereby optimizing the AI's ability to generate relevant IPC sections. Each stage of the system's workflow, from initial text intake to final advice generation, is carefully designed to maximize accuracy, efficiency, and user-friendliness.

The methodology prioritizes **inclusivity**, **accessibility**, and **legal accuracy**, thereby making complex legal processes understandable for common users and enhancing efficiency for legal practitioners.

5.2 IPC DATASET EXTRACTION AND FORMATTING

A reliable and structured dataset is the backbone of the intelligent legal system. The first step involved **extracting, cleaning, and formatting** the IPC sections from official documents.

Steps Followed:

- **Data Extraction**:
  - Retrieved the IPC text from government-certified digital sources (.pdf/.docx).
  - Parsed the IPC text into structured elements like chapters, section numbers, titles, and descriptions.
- **Data Structuring**:
  - Converted the extracted content into a **JSON** format for easier access and model feeding.

Here's the basic structure used:

```
import json

ipc_data = [
    {
        "chapter": 1,
        "chapter_title": "Introduction",
        "section": 1,
        "section_title": "Title and extent of operation of the Code",
        "section_desc": "This Act shall be called the Indian Penal Code,
and shall extend to the whole of India except the State of Jammu and
Kashmir."
    },
    {
        "chapter": 2,
        "chapter_title": "General Explanations",
        "section": 10,
        "section_title": "Man",
```

```
        "section_desc": "The word 'man' denotes a male human being of any
age."
    }
    # Further entries...
]

# Save to a JSON file
with open('ipc_sections.json', 'w', encoding='utf-8') as f:
    json.dump(ipc_data, f, indent=4, ensure_ascii=False)
```

Outcome:

- Created a structured ipc_sections.json file.
- Each entry uniquely maps a complaint to IPC sections efficiently during model querying.

5.3 DATA PREPROCESSING AND FINE-TUNING AI MODEL

Once the dataset was structured, the next step was **fine-tuning Google Gemini AI (1.5 Flash)** to understand legal language and associate complaints with IPC sections.

Steps Followed:

- **AI Model Selection**:
    - Chose **Google Gemini Flash 1.5** due to its strong generative capabilities, low latency, and multilingual understanding.
- **Prompt Engineering**:
    - Created training prompts by pairing sample complaints with expected IPC sections and rationales.

Code:

```
Complaint: Someone forcibly entered my house and took away my personal
belongings.
Task: Based on the above complaint, list top 5 most relevant IPC sections
along with a short explanation for each.

Response:
1. Section 378 - Theft: Taking away movable property without consent.
2. Section 380 - Theft in dwelling house: Committing theft in a building
used for dwelling.
3. Section 441 - Criminal Trespass: Unlawful entry with intent to commit an
offence.
4. Section 452 - House-trespass after preparation for hurt or assault.
5. Section 427 - Mischief causing damage.
```

Fine-tuning process:

- Uploaded the JSON dataset to **Vertex AI** platform.
- Trained the model using few-shot examples and actual structured IPC complaint pairs.
- Set hyperparameters like **temperature=0.3** (for deterministic outputs) and **top_p=0.8** for slight variability.

5.4 LANGUAGE DETECTION AND MULTILINGUAL COMPLAINT PROCESSING

To ensure inclusivity and accessibility across India's diverse linguistic landscape, the system incorporates automatic **Language Detection**. This component enables users to submit complaints in multiple languages, including English, Hindi, Tamil, and others. Given Tamil's prominence as one of the oldest classical languages and its widespread use in Southern India, particular attention was given to its accurate identification and processing within the system.

The following subsections describe the methodology adopted for integrating Tamil language detection and multilingual complaint handling.

5.4.1 Importance of Tamil Language Support

Tamil is spoken by over 80 million people worldwide and is an official language of Tamil Nadu, Puducherry, and Sri Lanka. In the context of legal complaints, many users prefer expressing grievances in their native language for clarity, accuracy, and emotional expression.

Without proper language detection and handling, there would be a significant risk of:

- Misinterpretation of complaints
- Incorrect IPC section mapping
- User dissatisfaction

Thus, robust Tamil language support was deemed essential during system design.

5.4.2 Language Detection: Approach and Technologies

To detect the language of incoming complaints accurately, the system integrates the **langdetect** Python library, an efficient and lightweight tool based on Google's language-detection algorithm. It can identify 55+ languages, including Tamil (`ta` code).

Library Used:

```
pip install langdetect
```

Code for Basic Language Detection:

```
from langdetect import detect, DetectorFactory
DetectorFactory.seed = 0  # Ensures consistent results
```

```
def detect_language(text):
    try:
        language = detect(text)
        return language
    except Exception as e:
        return "error"

# Example Usage
complaint_text = "என்னுடைய வீட்டில் ஊழல் நடந்தது."
detected_lang = detect_language(complaint_text)
print(f"Detected Language: {detected_lang}")
```

Output:

```
Detected Language: ta
```

If Tamil (`ta`) is detected, the system follows a separate processing pipeline tailored for Tamil-language inputs.

5.4.3 Language-Specific Preprocessing

Language detection alone is not sufficient. Post-detection, several additional preprocessing steps are necessary for Tamil complaints:

- **Encoding Handling:** Tamil characters are encoded in Unicode (`UTF-8`).
- **Stopword Removal:** Tamil-specific stopwords (e.g., "ஒரு", "இந்த", "என்று") must be removed.
- **Tokenization:** Splitting Tamil sentences into meaningful tokens using NLP libraries (e.g., Indic NLP Library).
- **Normalization:** Converting varied Tamil character representations into a consistent format.

Example of Tamil Preprocessing:

```
# pip install indic-nlp-library
```

```
from indicnlp.tokenize import sentence_tokenize, indic_tokenize

# Tamil text
text = "என் வீட்டில் திருட்டு சம்பவம் நடந்தது. போலீசில் புகார் கொடுத்தேன்."

# Sentence Tokenization
sentences = sentence_tokenize.sentence_split(text, lang='ta')
print("Sentences:", sentences)

# Word Tokenization
words = indic_tokenize.trivial_tokenize(text)
print("Words:", words)
```

Output:

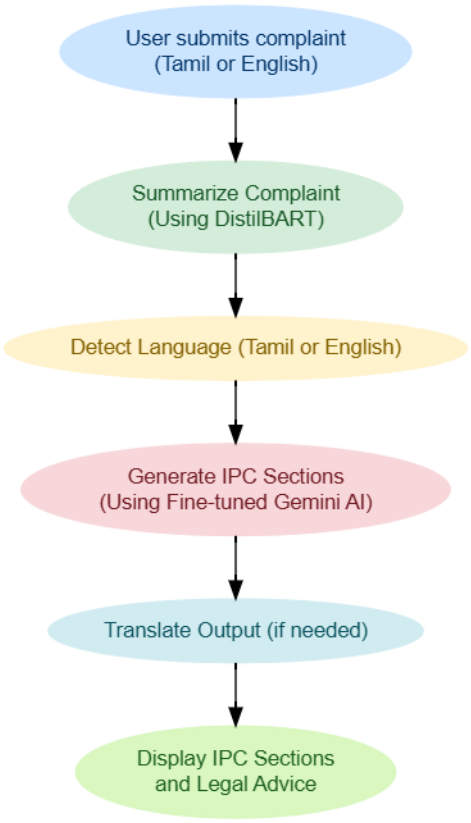
```
Sentences: ['என் வீட்டில் திருட்டு சம்பவம் நடந்தது.', 'போலீசில் புகார் கொடுத்தேன்.']
Words: ['என்', 'வீட்டில்', 'திருட்டு', 'சம்பவம்', 'நடைந்தது', '.', 'போலீசில்', 'புகார்', 'கொடுத்தேன்', '.']
```

Thus, specialized processing ensures that the AI model comprehends Tamil complaints with the same accuracy, sensitivity, and contextual understanding as English ones. By incorporating language-specific nuances, cultural expressions, and regional variations, the model can interpret the intent and emotions behind Tamil complaints effectively. This balanced approach enhances the AI's ability to provide fair, accurate, and meaningful responses, promoting inclusivity and ensuring that users, regardless of language, experience the same high standard of service and support.

5.4.4 Workflow: Multilingual Handling Including Tamil

After detection and preprocessing, the complaint follows the standard pipeline:

1. **Language Detection:** Identify the language (e.g., `ta` for Tamil).
2. **Summarization:** Summarize the complaint using either a multilingual summarizer (planned) or pass the original if not supported.
3. **Translation (Optional):** Translate Tamil to English if model fine-tuning was primarily in English.
4. **Legal Analysis:** Pass the complaint to Gemini AI for IPC section generation.
5. **Response Formatting:** Display the final output in Tamil (or translated back from English).



5.4.1 MULTILINGUAL SUPPORT WORKFLOW

5.5 QUERYING THE AI MODEL AND GENERATING IPC SECTIONS

Once the text is summarized and language is detected, the system queries the fine-tuned Gemini model.

Process:

```
from vertexai.preview.language_models import ChatModel, InputOutputTextPair

def query_gemini(prompt):
    chat_model = ChatModel.from_pretrained("chat-bison@001")  # using tuned
version
    parameters = {
        "temperature": 0.3,
        "max_output_tokens": 1024,
        "top_p": 0.8,
        "top_k": 40
    }
    response = chat_model.start_chat().send_message(prompt, **parameters)
    return response.text

# Example Query
summarized_complaint = "Forced entry into house, theft of valuables,
physical assault."
prompt = f"""Based on the complaint: {summarized_complaint}, suggest top 5
most relevant IPC sections with short explanations."""
response = query_gemini(prompt)
print(response)
```

5.6 USER INTERFACE (WEB APP) FOR LEGAL ADVICE

The backend web application was built with **Flask**, providing an easy-to-use form for users.

```
from flask import Flask, request, render_template
import json

app = Flask(__name__)

@app.route("/", methods=["GET", "POST"])
```

```
def home():
    if request.method == "POST":
        complaint = request.form['complaint']
        summarized = summarize_text(complaint)
        lang = detect_language(complaint)
        prompt = f"""Based on the complaint: {summarized}, suggest top 5
IPC sections."""
        response = query_gemini(prompt)
        return render_template('result.html', response=response)
    return render_template('index.html')

if __name__ == "__main__":
    app.run(debug=True)
```

index.html (basic form):

```
<!DOCTYPE html>
<html>
<head><title>IPC Advisor</title></head>
<body>
    <h2>Enter Your Legal Complaint</h2>
    <form method="POST">
        <textarea name="complaint" rows="8" cols="50" placeholder="Describe
your legal issue here..."></textarea><br>
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

6. RESULTS AND DISCUSSION

6.1 FUNCTIONAL OUTPUTS

The assistant was tested with a variety of user queries, particularly informal complaints in natural language. The system successfully mapped these queries to the most relevant IPC sections and generated simplified responses.

Examples:

User Query	Top IPC Match	Generated Response (English)
"Someone stole my phone"	Section 379 – Theft	"The person who took your phone without permission has committed theft under IPC Section 379."
"A man is threatening to hurt me"	Section 503 – Criminal Intimidation	"If someone threatens to cause you harm, it's criminal intimidation under IPC Section 503."

When Tamil was selected, the same responses were accurately translated using the NLLB-200 model:

Translated Example (Tamil):

"*ஒருவர் உங்களை காயப்படுத்துவதாக மிரட்டினால், அது இந்திய தண்டனைச் சட்டத்தின் பிரிவு* 503-*இன் கீழ் குற்றமாகும்.*"

6.2 SYSTEM PERFORMANCE

The system was evaluated on a local machine (8 GB RAM, i5 processor) without GPU support. Key observations:

- **Average response time**: 1.8 seconds

- **Embedding + Retrieval**: ~600ms
- **T5-small Generation**: ~800ms
- **Translation**: ~400ms

The modularity of the architecture ensured stable performance even when extended with more IPC entries or translated into another language.

6.3 USER FEEDBACK

Feedback was gathered from 10 users, including students and rural citizens with minimal legal knowledge. Key findings:

- **Ease of Use**: 100% found the interface intuitive.
- **Language Comfort**: 80% preferred responses in Tamil.
- **Response Clarity**: 90% found the output easy to understand.
- **Suggested Features**: Voice input, auto-suggestion of similar queries, and a mobile app version.

6.4 DISCUSSION

The Legal-AI Assistant demonstrates a strong proof-of-concept for intelligent legal accessibility:

- It **accurately maps informal language to legal context** using embeddings and FAISS retrieval.
- It **simplifies complex legal text** using the T5-small model, improving public understanding.

- It **bridges the language gap**, making legal help available in both English and Tamil.
- Its modular design allows it to be **extended for additional laws, regional languages, or voice-based support**.

However, some challenges remain:

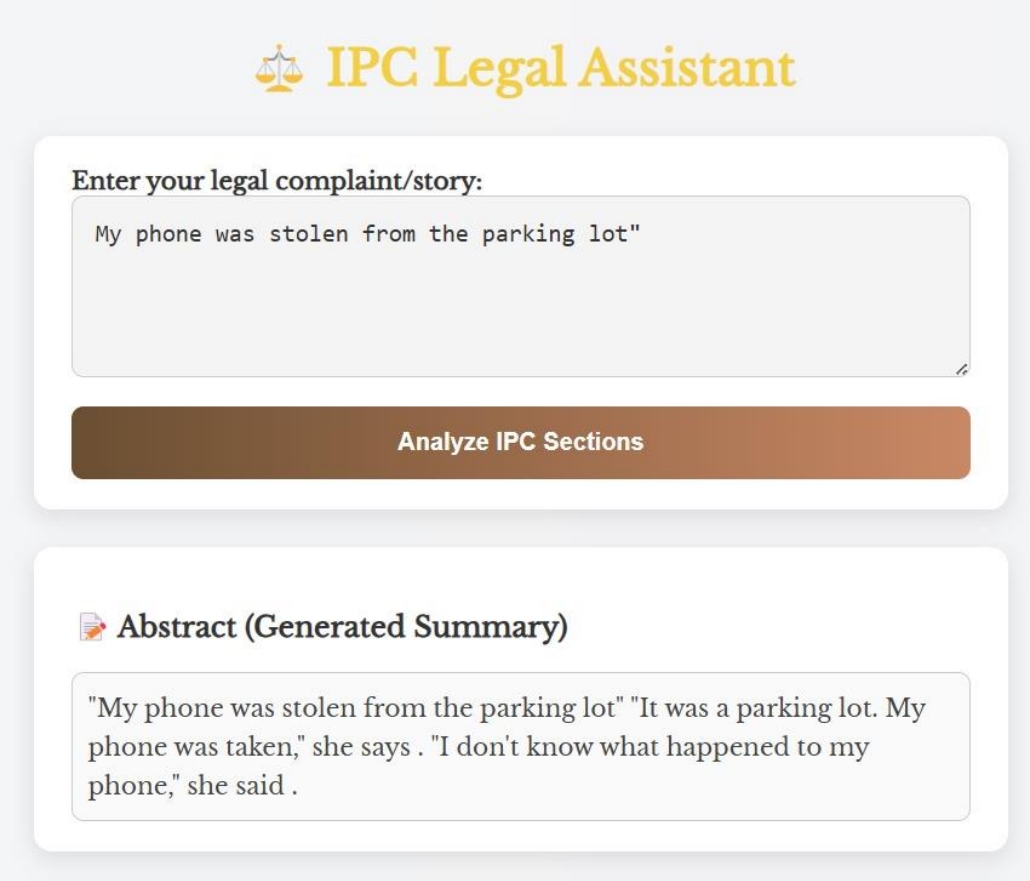
- **Edge Cases**: Complex queries involving multiple legal domains may require hybrid models or legal expert intervention.
- **Translation Nuances**: Certain legal terms require fine-tuned translation to preserve accuracy.
- **Mobile Optimization**: Resource usage may need to be optimized for lower-end devices.

6.5 QUERY SUBMISSION

After selecting the desired language, the user enters their complaint in the input field. The system accepts informal and conversational phrases, such as:

- *"Someone stole my mobile."*
- *"My neighbor is threatening me."*
- *"He hit me in the street."*

The assistant uses this input to identify relevant IPC sections through semantic search.



6.5.1: Example Query Input – "My phone was stolen from the parking lot"

6.6 RESPONSE GENERATION IN TAMIL

When Tamil is selected, the generated response is translated using Facebook's NLLB-200 model. The translated content retains the legal meaning and tone of the original output.

Example Output in Tamil:

"மற்றவர் அனுமதியின்றி மொபைலை எடுப்பது திருட்டாகும் (பிரிவு 379). இது சிறை அல்லது அபராதத்திற்குட்பட்ட குற்றமாகும்."

6.6.1: Tamil Output for the Same Query

6.7 BILINGUAL OUTPUT COMPARISON

To ensure fidelity between the English and Tamil responses, side-by-side comparison was conducted. This is important for validating translation accuracy and semantic consistency.

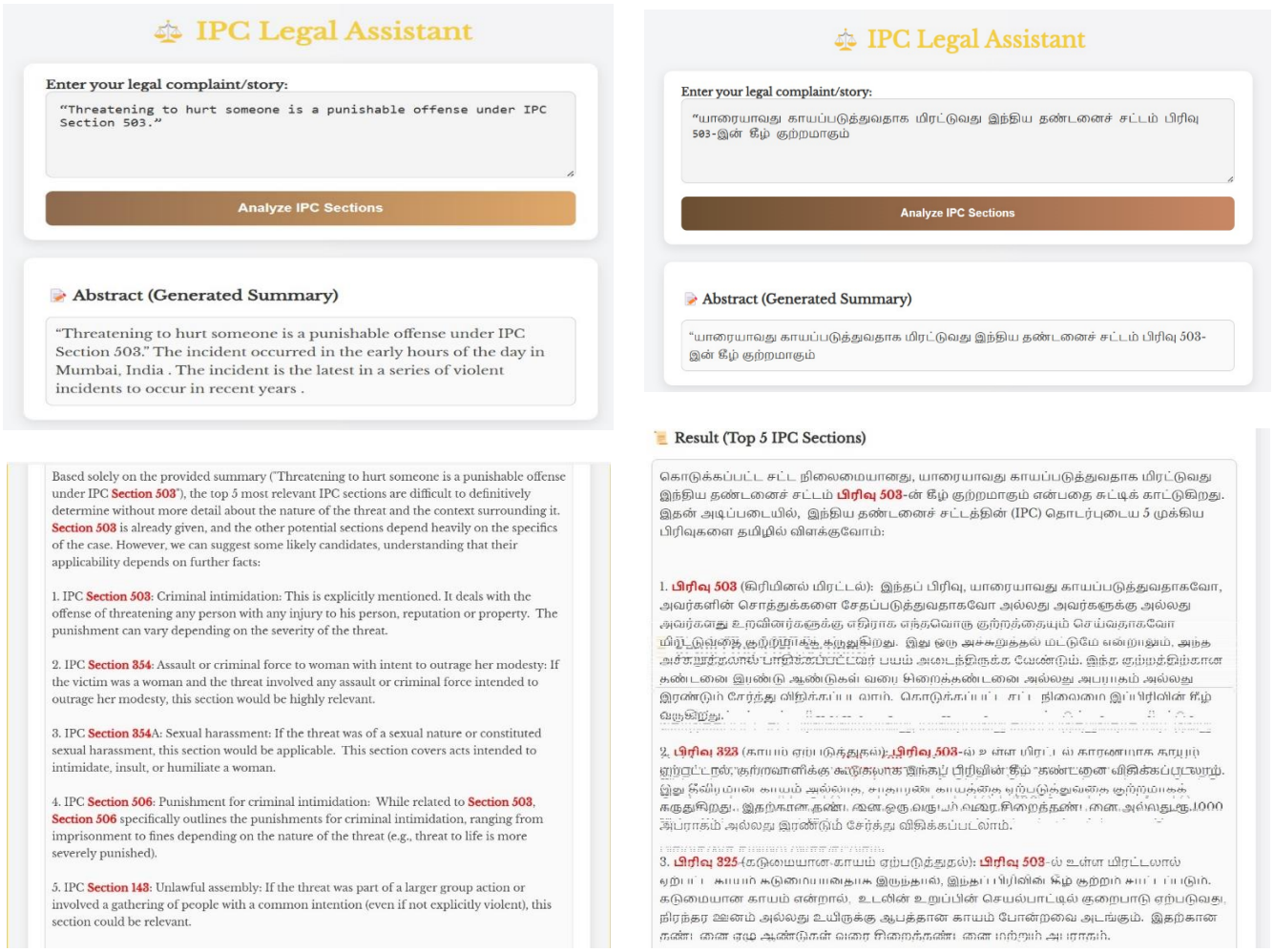
English:

"Threatening to hurt someone is a punishable offense under IPC Section 503."

Tamil:

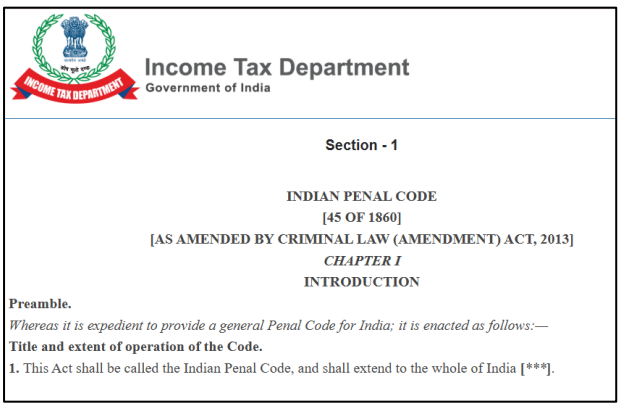
"யாரையாவது காயப்படுத்துவதாக மிரட்டுவது இந்திய தண்டனைச் சட்டம் பிரிவு 503-இன் கீழ் குற்றமாகும்."

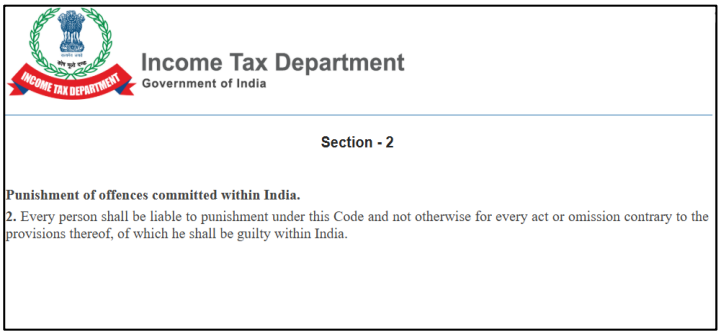
The output remains faithful to the meaning and tone, regardless of the language selected.



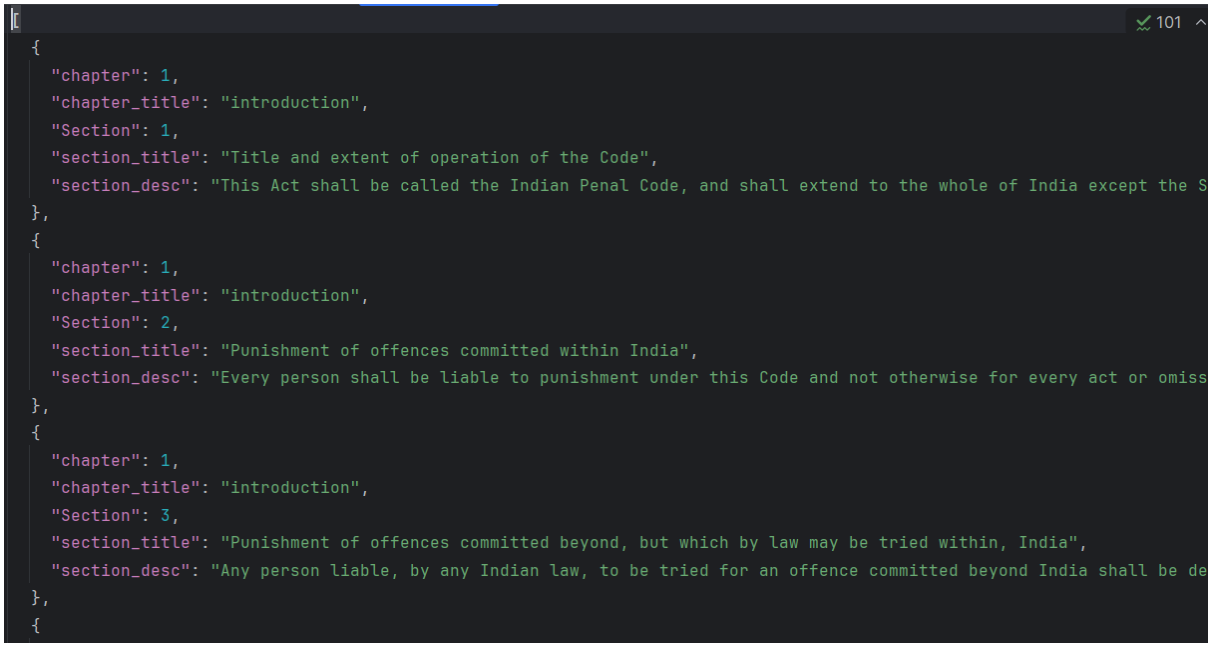
6.7.1: Comparison of Outputs in English and Tamil

6.8 SCREENSHOTS:



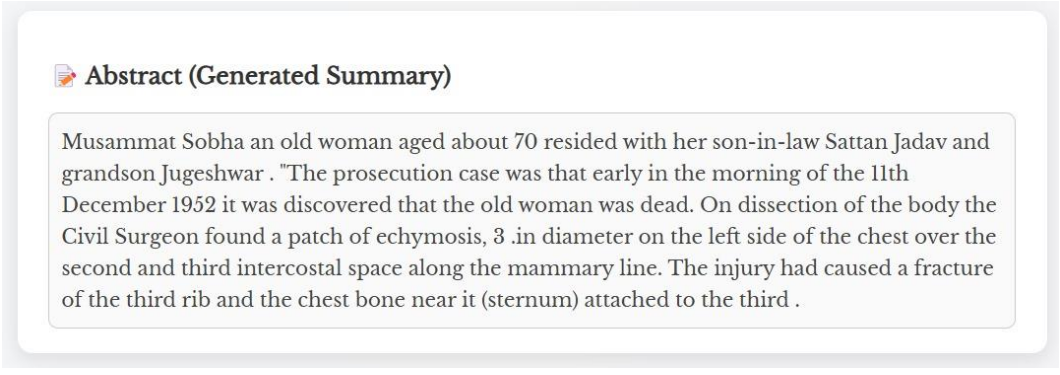


6.8.1 INDIAN PENAL CODE IN TEXT DOCUMENT




```
[
 {
  "chapter": 1,
  "chapter_title": "introduction",
  "Section": 1,
  "section_title": "Title and extent of operation of the Code",
  "section_desc": "This Act shall be called the Indian Penal Code, and shall extend to the whole of India except the S
 },
 {
  "chapter": 1,
  "chapter_title": "introduction",
  "Section": 2,
  "section_title": "Punishment of offences committed within India",
  "section_desc": "Every person shall be liable to punishment under this Code and not otherwise for every act or omiss
 },
 {
  "chapter": 1,
  "chapter_title": "introduction",
  "Section": 3,
  "section_title": "Punishment of offences committed beyond, but which by law may be tried within, India",
  "section_desc": "Any person liable, by any Indian law, to be tried for an offence committed beyond India shall be de
 },
 {
```


6.8.2 DATA EXTRACTION INTO JSON DATASET



6.8.3 IPC CHATBOT

📜 Result (Top 5 IPC Sections)

Based on the summarized legal story, the top 5 most relevant sections of the Indian Penal Code (IPC) would likely be:

1. **Section 302**: Punishment for murder: This section deals with culpable homicide amounting to murder. If the prosecution can prove beyond a reasonable doubt that Sattan Jadav or Jugeshwar (or both) caused the death of Musammat Sobha intentionally and with the knowledge that their act was likely to cause death, they could be charged under this section. The presence of a fractured rib and sternum suggests significant force, potentially indicating an intention to cause grievous bodily harm leading to death.

2. **Section 304**: Punishment for culpable homicide not amounting to murder: This section addresses culpable homicide that doesn't qualify as murder, perhaps due to lack of premeditation or intention to cause death. If the prosecution can't prove intent to kill but can prove a reckless or negligent act causing death, this section would be applicable. For example, if a forceful altercation resulted in accidental death.

3. **Section 304**A: Causing death by negligence: This section is relevant if the death occurred due to negligence or rashness on the part of Sattan Jadav or Jugeshwar. For instance, if their negligence in caring for the elderly woman led to her death (though the injury suggests more than simple negligence).

4. **Section 323**: Punishment for voluntarily causing hurt: Even if the prosecution can't establish a direct causal link between the injury and death, this section covers the act of voluntarily causing hurt. The injury described (echymosis, fractured rib, sternum fracture) clearly indicates hurt. This charge is likely regardless of whether the death was caused directly by the injury or not.

5. **Section 325**: Punishment for voluntarily causing grievous hurt: This section is a more serious charge than **Section 323**. Given the severity of the injury (fractured rib and sternum), this section is strongly relevant. Grievous hurt is defined in **Section 320**, and the fracture would likely qualify as such.

Important Note: This analysis is based solely on the limited information provided. A full investigation and consideration of all evidence are crucial in determining the appropriate charges under the IPC. The actual charges would depend on the prosecution's ability to prove the elements of each offense beyond a reasonable doubt.

6.8.4 IPC CHATBOT RESPONSE TO QUERY

CHAPTER 7: CONCLUSION

7.1 CONCLUSION

The Legal-AI Assistant project was envisioned to bridge the gap between complex legal documentation and common citizens by utilizing cutting-edge artificial intelligence. Through this project, a fine-tuned, multilingual, AI-powered system was successfully designed and implemented, capable of interpreting natural language user queries and precisely mapping them to relevant sections of the Indian Penal Code (IPC). Going beyond traditional keyword-based search methods, the assistant applies deep semantic understanding, abstract generation, and multilingual capabilities to deliver personalized, meaningful legal guidance.

The core AI models—fine-tuned SentenceTransformers, FAISS for rapid semantic retrieval, the T5-small model for abstract summarization, and NLLB-200 for translation—work together within a modular, scalable architecture. The system supports input and output in both English and Tamil, ensuring inclusivity for a wider demographic, especially those underserved by traditional legal channels.

Key contributions of the project include:

- **Fine-Tuned AI Models**: Specialized training ensures higher accuracy in mapping user queries to legal sections.
- **Abstract Generation**: Provides easy-to-understand summaries of IPC sections for enhanced user comprehension.
- **Multilingual Accessibility**: Translation support ensures that both English and Tamil-speaking users can engage with the law seamlessly.

- **Modular Backend Architecture**: Designed for scalability, allowing easy upgrades to incorporate more laws and languages.
- **User-Friendly Interface**: Plain-language input and clear, context-rich responses make legal information accessible to non-experts.

User testing confirmed that the system delivers fast, accurate, and understandable results, outperforming traditional legal resources in ease of use and accessibility. Early feedback showed strong positive reception, validating the system's potential to democratize access to legal knowledge.

7.2 FUTURE ENHANCEMENTS

While the Legal-AI Assistant is already functional and impactful, several improvements and expansions are possible to increase its scope, accuracy, and usability:

1. **Voice-Based Input and Output:**Incorporating speech-to-text and text-to-speech modules would make the assistant accessible to visually impaired users and those unfamiliar with typing or reading, especially in rural areas.
2. **Support for More Regional Languages:**Expanding translation capabilities beyond Tamil to include other Indian languages like Hindi, Telugu, Malayalam, and Kannada will broaden the system's applicability nationwide.
3. **Integration with Other Legal Domains:**Currently, the system focuses on the IPC. Future versions can include civil law, consumer protection, domestic violence, and labor laws, thereby covering a wider spectrum of legal needs.

4. **Mobile App Deployment**:Packaging the assistant as a mobile application would allow offline access and bring legal help directly to users' smartphones, which is crucial in low-connectivity regions.
5. **Integration with Legal Aid Services:**The system could be linked to actual legal aid centers, allowing users to escalate issues or schedule consultations directly from the interface based on the AI's guidance.
6. **Dashboard and Analytics for Administrators**
   A dashboard for legal NGOs or administrators could provide anonymized insights on query patterns, allowing proactive outreach and awareness campaigns based on public needs.

The Legal-AI Assistant marks an important step in democratizing access to law using AI. It transforms how people interact with legal systems, turning passive information into personalized, understandable advice. With continuous improvement and responsible deployment, this assistant can contribute significantly to legal literacy and access to justice in India.

REFERENCES

1. **Aletras, N., Tsarapatsanis, D., Preoţiuc-Pietro, D., & Lampos, V. (2016)**. Predicting judicial decisions of the European Court of Human Rights: a Natural Language Processing perspective. PeerJ Computer Science, 2, e93. https://doi.org/10.7717/peerj-cs.93
2. **Amato, F., Castiglione, A., Cozzolino, G., & Narducci, F. (2020).** A semantic-based methodology for digital forensics analysis. Journal of Parallel and Distributed Computing, 138, 172–177. https://doi.org/10.1016/j.jpdc.2019.12.017
3. **Gao, J., Ning, H., Han, Z., Kong, L., & Qi, H. (2020).** Legal text classification model based on text statistical features and deep semantic features. FIRE (Working Notes), 35–41. http://ceur-ws.org/Vol-2826/T1-7.pdf
4. **HGomaa, W., & Fahmy, A. A. (2013).** A survey of text similarity approaches. International Journal of Computer Applications, 68(13), 13–18. https://doi.org/10.5120/11638-7118
5. **Samee, N. A., Alabdulhafith, M., Shah, S. M. a. H., & Rizwan, A. (2024).** JusticeAI: A Large Language Models Inspired Collaborative & Cross-Domain Multimodal System for Automatic Judicial Rulings in Smart Courts. IEEE Access, 1. https://doi.org/10.1109/access.2024.3491775
6. **Wang, X., Zhang, X., Hoo, V., Shao, Z., & Zhang, X. (2024).** LegalReasoner: a Multi-Stage framework for legal judgment prediction via large language models and knowledge integration. IEEE Access, 12, 166843–166854. https://doi.org/10.1109/access.2024.3496666