LEGAL ASSISTANCE SYSTEM FOR CRIMINAL VIOLENCE VICTIMS USING NATURAL LANGUAGE PROCESSING

Adarsh G	(20Z204)
Elanthamil R	(20Z215)
Jeevan Krishna K V	(20Z220)
Nirmal M	(20Z267)
Ajay Deepak P M	(21Z431)

Dissertation submitted in partial fulfilment of the requirements for the degree of

BACHELOR OF ENGINEERING

Branch: COMPUTER SCIENCE & ENGINEERING

of Anna University

APRIL 2024

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

PSG COLLEGE OF TECHNOLOGY

(Autonomous Institution)

COIMBATORE – 641 004

PSG COLLEGE OF TECHNOLOGY

(Autonomous Institution)

COIMBATORE – 641 004

LLM BASED CHATBOT FOR COURSE BASED QUESTION ANSWERING

Bona fide record of work done by

Adarsh G	(20Z204)
Elanthamil R	(20Z215)
Jeevan Krishna K V	(20Z220)
Nirmal M	(20Z267)
Ajay Deepak P M	(21Z431)

Dissertation submitted in partial fulfilment of the requirements for the degree of

BACHELOR OF ENGINEERING

Branch: COMPUTER SCIENCE & ENGINEERING

of Anna University

April 2024

..……	……..……
Dr. N. Gopika Rani	**Dr. G. Sudha Sadasivam**
Faculty guide	Head of the Department

Certified that the candidate was examined in the viva-voce examination held on ……

……	……...
(Internal Examiner)	(External Examiner)

CERTIFICATE

Certified that this report titled "**Legal Assistance System for Criminal Violence Victims using Natural Language Processing**", for the Project Work II (19Z820) is a bonafide work of **Adarsh G (20Z204), Elanthamil R (20Z215), Jeevan Krishna K V (20Z220), Nirmal M (20Z267), Ajay Deepak P M (21Z431)** who have carried out the work under my supervision for the partial fulfillment of the requirements for the award of the degree of Bachelor of Engineering in Computer Science and Engineering. Certified further that to the best of my knowledge and belief, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion.

Adarsh G

Elanthamil R

Jeevan Krishna K V

Nirmal M

Ajay Deepak P M

Place: Coimbatore
Date:

Dr. Gopika Rani N
Designation: Assistant Professor(SG)
Department of Computer Science and Engineering
PSG College of Technology
Coimbatore - 641004

COUNTERSIGNED
HEAD
Department of Computer Science and Engineering
PSG College of Technology
Coimbatore - 641004

TABLE OF CONTENTS

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to **Dr. K. Prakasan**, Principal, PSG College of Technology, for giving us the opportunity to do a project with various facilities and infrastructure, without which the success of the project would not have been possible.

We extend our heartfelt thanks to **Dr. G. Sudha Sadasivam**, Professor and Head, Department of Computer Science and Engineering, PSG College of Technology, for her unfailing support throughout this project.

We express our sincere thanks to a Program Coordinator **Dr. N. Arul Anand**, Professor, Department of Computer Science and Engineering, PSG College of Technology, whose constant support and everlasting enthusiasm made it possible to be completed within the time.

We heartily thank our guide **Dr. N. Gopika Rani**, Assistant Professor(SG), Department of Computer Science and Engineering who was always there to help us and played a major role in the completion of the project and we wish to thank her for her enduring guidance.

We heartily thank our reviewer **Dr. V. Santhi**, Professor, Department of Computer Science and Engineering for her guidance and feedback that played an important role in completion of the project.

We heartily thank our reviewer **Dr. S. Arul Jothi**, Assistant Professor(SG), Department of Computer Science and Engineering for her guidance and feedback that played an important role in completion of the project.

We also wish to express our sincere thanks to our tutor **Mrs. T. Anusha**, Assistant Professor(SG), Department of Computer Science and Engineering, for her guidance that played a vital role in completing my project on time.

Finally we would like to thank the faculty members, staff of CSE Department and friends without whom this project work would not have been completed successfully.

ABSTRACT

Traditional legal processes often present challenges for individuals seeking assistance on appropriate legal actions, leading to confusion and inefficiencies. Therefore, a system for providing legal assistance for criminal violence is introduced. The system is developed using Natural Language Processing (NLP) to address these issues.

The legal assistance system for criminal violence aims to revolutionize the way individuals access and understand legal information. One key feature of the system is its ability to analyze and interpret case details with a high degree of accuracy, allowing users to receive tailored recommendations specific to their situations.

The system incorporates a vast database of legal precedents, statutes, and case law, continuously updated to ensure its recommendations align with the latest legal standards. This dynamic approach not only enhances the reliability of the system but also ensures users are provided with up-to-date information, crucial for navigating the ever-evolving legal landscape.

The system goes beyond mere suggestion by offering comprehensive insights into the legal reasoning behind its recommendations. Users can delve into the specifics of applicable laws, understand the implications of different legal actions, and gain valuable knowledge to make informed decisions. This educational aspect distinguishes the system from traditional legal resources, empowering users with a deeper understanding of the legal nuances related to criminal violence cases.

In addition, the criminal violence legal aid system has an intuitive user interface that breaks down complicated legalese and procedures so that people with different degrees of legal knowledge can utilize it. The system allows users to communicate with it in a conversational way by using Natural Language Processing (NLP). Users can ask questions and get succinct, understandable answers. Through fostering a more inclusive and equitable approach to accessing legal information, this user-centric design seeks to close the gap between the legal system and those seeking assistance. By leveraging cutting-edge technology, the system not only addresses the challenges of traditional legal processes but also strives to democratize legal knowledge, ensuring that everyone, regardless of their background, can navigate the intricacies of criminal violence cases with confidence and understanding.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

NLP	Natural Language Processing
AI	Artificial Intelligence
SIM	Similarity Score Model
LDA	Latent Dirichlet Allocation
IDF	Inverse Document Frequency
TF-IDF	Term Frequency - Inverse Document Frequency
ANLP	Arabic Natural Language Processing
SVM	Support Vector Machine
NER	Named Entity Recognition
GCN	Graph Convolutional Network
GAT	Graph Attention Network
CSV	Comma Separated Values
POS	Parts Of Speech
MLM	Masked Language Model
NSP	Next Sentence Prediction
IPC	Indian Penal Code
ML	Machine Learning
BERT	Bidirectional Encoder Representations from Transformers

CHAPTER 1

INTRODUCTION

In contemporary society, navigating the legal landscape can often be a daunting and perplexing task, particularly for individuals seeking assistance in matters of criminal violence. Traditional legal processes, with their complex terminology, labyrinthine procedures, and ever-changing statutes, frequently present challenges that lead to confusion and inefficiencies. Recognizing these obstacles, a revolutionary system for providing legal assistance for criminal violence has been developed. Leveraging the power of Natural Language Processing (NLP), this system aims to transform the way individuals access and comprehend legal information, offering tailored recommendations and comprehensive insights to empower users in their legal journey.

I) Growing imperative for a New Approach

Traditional legal processes often present significant hurdles for individuals seeking guidance on appropriate legal actions. The complexity of legal language and procedures can lead to confusion, causing delays and inefficiencies in resolving legal issues, particularly in cases involving criminal violence. Accessing timely and accurate legal information can be challenging, for individuals who cannot afford legal representation or lack access to legal resources. Traditional processes often result in delays and inefficiencies due to the time-consuming nature of manual research and analysis. This can prolong legal proceedings and exacerbate the challenges faced by individuals seeking assistance. Legal resources are often inaccessible to marginalized communities or individuals with limited financial means, leading to disparities in access to justice.

II) Natural Language Processing as a transformative technology

Natural Language Processing (NLP) plays a transformative role in the legal assistance system for criminal violence, enabling a paradigm shift in how individuals access and understand legal information. NLP enables the system to interact with users in natural language, mimicking human conversation. This facilitates a more intuitive and user-friendly experience, allowing individuals to pose questions, seek advice, or request information without the need for specialized legal knowledge. The ability to communicate naturally with the system transforms the way users interact with legal resources, making it accessible to individuals with varying levels of expertise. NLP algorithms are used to accurately interpret and process user queries related to criminal violence

cases. These algorithms enable the system to understand the nuances of legal language, identify key concepts, and extract relevant information from user inputs. By effectively understanding user queries, the system can provide tailored recommendations and guidance specific to each user's situation, enhancing the accuracy and relevance of the assistance provided. NLP facilitates continuous learning and improvement of the system over time. Through ongoing updates and feedback mechanisms, the system can adapt to changes in legal standards, new case law, and evolving language usage. This ensures that the system remains up-to-date and relevant, providing users with the most current and accurate information available. The ability to learn and evolve ensures the longevity and effectiveness of the system, making it a valuable resource for individuals navigating the legal landscape.

III) Dynamic Legal Guidance for Enhanced Understanding and Accessibility

One key feature of the legal assistance system is its ability to analyze and interpret case details with a high degree of accuracy. By understanding the nuances of individual cases, the system can provide tailored recommendations specific to the user's situation, thereby addressing the unique needs of each user. The system incorporates a vast database of legal precedents, statutes, and case law, continuously updated to ensure its recommendations align with the latest legal standards. This dynamic approach enhances the reliability of the system and provides users with up-to-date information crucial for navigating the ever-evolving legal landscape. Furthermore, the legal assistance system for criminal violence adopts a user-friendly interface that simplifies complex legal processes, making it accessible to individuals with varying levels of legal expertise. By breaking down barriers to understanding, the system ensures that users can navigate the intricacies of the legal system with confidence. Through the use of Natural Language Processing (NLP), the system enables users to interact with it in a conversational manner. Users can pose queries in natural language and receive clear, concise responses, eliminating the need for specialized legal knowledge and fostering a more inclusive approach to accessing legal information.

Through its user-friendly interface and conversational interaction, the system strives to democratize legal knowledge, ensuring that everyone, regardless of their background, can navigate the intricacies of criminal violence cases with confidence and understanding. By breaking down barriers to access and providing comprehensive guidance, the system empowers individuals to make informed decisions and take appropriate legal actions, ultimately contributing to a

more just and equitable society. In summary, the legal assistance system for criminal violence represents a paradigm shift in the way individuals interact with and understand the legal system. By leveraging the capabilities of NLP, incorporating a vast database of legal resources, and prioritizing user accessibility and education, the system aims to revolutionize legal assistance, making it more accessible, reliable, and empowering for all.

CHAPTER 2

LITERATURE SURVEY

V. Socatiyanurak *et al*.,(2021) proposes the development of a chatbot named LAW-U. LAW-U is designed as an Artificial Intelligence (AI) chatbot to provide legal guidance to survivors of sexual violence. This system is built on Natural Language Processing (NLP) pipelines, using 182 Thai Supreme Court cases related to sexual violence to train LAW-U. It involves stratified 5-fold validation to train and evaluate LAW-U's performance. The system considers different models, including a similarity score model (SIM), a model incorporating common keywords (SIM×KEY), and a model with keywords' synonyms (SIM×KEY×SYN). The experimental results show an accuracy of 88.24% for the testing datasets. [1]

V. Murali et al.,(2018) proposes the development of a chatbot named "ChEMBL Bot" that serves as a conversational agent for the ChEMBL database. ChEMBL is a chemical database containing curated bioactivity data and extensive annotations about compounds, such as their biological relevance, medicinal uses, and pharmacology. The paper outlines the steps taken in developing the ChEMBL Bot, including data collection from ChEMBL, preparation of the bot using Dialogflow, and the creation of a web application. The goal of ChEMBL Bot is to provide an additional means of accessing ChEMBL resources through natural language conversation. [2]

Chenguang Pan and Wenxin Li (2010) propose the development of a research paper recommender system by integrating topic analysis techniques into collaborative filtering methods. The proposed recommender system aims to address this gap by assisting researchers in finding relevant papers in their specific fields of interest. The authors suggest incorporating topic model techniques for analyzing the content of research papers. Topic models are used to extract underlying themes or topics from the textual information in the papers. The proposed system introduces a thematic similarity measurement based on the topic analysis. [3]

Kordabadi et al.,(2022) proposes the development of a movie recommender system that addresses the concern of inappropriate content for children and adolescents. The system employs Latent Dirichlet Allocation (LDA) modeling, a type of topic modeling technique. Machine learning models are employed in the recommendation process. These models likely leverage the

information obtained through LDA and age ratings to suggest movies that align with the user's age group. [4]

U. C. De et al.,(2023) proposes a text-based recommendation system for eCommerce apparel stores, specifically focusing on ladies' apparel. It Utilizes data acquired from the Amazon Product Advertising API in a policy-compliant manner. Focuses on ladies' apparel and aims to find text-based product similarities using three approaches: Bag of Words (BoW), Inverse Document Frequency (IDF), and Term Frequency-Inverse Document Frequency (TF-IDF) applied to product titles. [5]

R. Shrivastava et al.,(2019) outlines a research paper focused on addressing challenges related to news authenticity in the Bengali language. The proposed solution involves leveraging blockchain technology, smart contracts, and incremental machine learning to evaluate news articles in a decentralized manner. The authors aim to combat issues such as biased reporting, misinformation, and the lack of a reliable news evaluation process. [6]

S. Larabi Marie-Sainte et al.,(2019) appears to be an academic paper or survey discussing Arabic Natural Language Processing (ANLP) and machine learning-based systems. The document covers various aspects, including the characteristics and complexity of the Arabic language, the involvement of machine learning algorithms in ANLP, and the challenges encountered in ANLP applications. [7]

Tyagi, Nemika & Bhushan, Bharat (2023) the presented paper offers a comprehensive survey on the application of Natural Language Processing (NLP) in the context of Smart Healthcare. The authors delve into the significant advancements in smart healthcare, facilitated by emerging technologies like artificial intelligence (AI). Specifically, the focus is on the pivotal role played by NLP, a technology fueled by AI, in analyzing and comprehending human language. [8]

H. Liu et al., (2019) the paper emphasizes the increasing importance of leveraging NLP in the context of electronic medical records, particularly for tasks such as information extraction, clinical decision-making, and disease diagnosis. The authors underscore the value of NLP in handling unstructured clinical text, with a particular focus on radiology reports. The study positions itself within the broader landscape of machine learning applications in healthcare, highlighting

the potential of NLP-driven approaches in advancing radiological research and clinical practice. [9]

M. Yang et al.,(2020) this paper explores the challenges and advancements in dynamic graph embedding, with a specific focus on addressing the over-smoothing problem in the context of dynamic graphs. Dynamic graphs, representing evolving structures such as social networks and communication networks, pose unique challenges due to their time-varying nature. The paper analyzes the limitations of existing dynamic graph models, particularly in the context of stacking multiple graph convolutional layers, which can lead to serious feature shrinkage and oversmoothing. The proposed solution, L2 feature normalization, is introduced to counteract these issues by rescaling nodes to a hypersphere of a unit ball. The survey provides a comprehensive review of related works in both topological dependencies and temporal-evolving patterns, emphasizing the need for effective solutions in the dynamic graph embedding domain. [10]

Topic	Algorithm Used	Advantages	Disadvantages
LAW-U: Legal Guidance Through Artificial Intelligence Chatbot for Sexual Violence Victims and Survivors	Natural Language Processing (NLP) pipelines, including similarity score model (SIM), SIM×KEY, SIM×KEY×SYN	● High accuracy (88.24%) on testing datasets. ● Utilizes NLP techniques for legal guidance. ● Stratified 5-fold validation for robust evaluation.	● Limited to Thai Supreme Court cases related to sexual violence. ● Specific to legal guidance for sexual violence survivors.
ChEMBL Bot - A Chat Bot for ChEMBL database	Dialogflow for chatbot development	● Provides a conversational interface for accessing ChEMBL database resources. ● Utilizes natural language conversation for data retrieval.	● Dependency on external tools like Dialogflow. ● Limited to ChEMBL database context.
Research paper recommendation	Integrates topic analysis	● Addresses the gap in finding	● May require significant

with topic analysis	techniques into collaborative filtering methods	relevant research papers. ● Incorporates thematic similarity based on topic analysis.	computational resources for topic modeling. ● Effectiveness may depend on the quality of topic models.
A Movie Recommender System based on Topic Modeling using Machine Learning Methods	Latent Dirichlet Allocation (LDA) modeling for topic modeling	● Addresses the concern of inappropriate content for children and adolescents. ● Leverages LDA and age ratings for movie recommendations.	● Relies on the quality of age ratings and LDA modeling. ● Limited to movie recommendations.
Text-Based Recommendation System for E-Commerce Apparel Stores	Bag of Words (BoW), Inverse Document Frequency (IDF), Term Frequency-Inverse Document Frequency (TF-IDF) applied to product titles	● Utilizes three text-based similarity approaches for eCommerce apparel recommendations. ● Uses data from Amazon Product Advertising API.	● Limited to ladies apparel recommendations. ● May face challenges with varied and evolving product titles.
Product Recommendations Using Textual Similarity Based Learning Models	Bag-of-Words (BoW), TF-IDF(Term Frequency-Inverse Document Frequency)	● Allows for a more nuanced understanding of items, enhancing the recommendation process beyond simple metadata like brand or price. ● By using	● The approach heavily relies on textual data, which might not always be available or may not adequately represent the characteristics of certain products, such as visual

		techniques like Bag of Words (BoW) and TF-IDF, it's possible to customize the similarity metric based on the specific needs of the recommendation system, such as giving more weight to certain words or features.	attributes or user-generated content like reviews. ● Representing products as high-dimensional vectors can lead to increased computational complexity.
Arabic Natural Language Processing and Machine Learning-Based Systems	Support Vector Machines (SVM), Naive Bayes, Decision Trees	● Provides a thorough review of ANLP-based ML systems, offering valuable insights. ● Involvement of ML algorithms in developing ANLP applications	● Scope Limitation: The paper primarily focuses on presenting the characteristics of the Arabic language
Natural Language Processing (NLP) Based Innovations for Smart Healthcare Applications in Healthcare 4.0	Named Entity Recognition (NER), Sentence Classification, Text Summarization,	● It discusses state-of-the-art NLP technologies and methodologies, providing insights into the latest advancements in the field. ● It highlights the integration of ● data analytics, machine learning, and AI in modern healthcare systems.	● Limited scope in terms of in-depth analysis or evaluation of specific NLP algorithms or applications

A Natural Language Processing Pipeline of Chinese Free-Text Radiology Reports for Liver Cancer Diagnosis	Named Entity Recognition (NER), Synonyms Normalization, Feature Selection, Relationship Extraction	• Presents a novel NLP pipeline specifically tailored for Chinese free-text radiology reports. • Achieved a high F1 score of 93.00% in named entity recognition using the BiLSTM-CRF model with lexicon incorporation.	• Limitation of a relatively small corpus for Chinese EMRs, which may affect the generalizability of the findings and the robustness of the NLP pipeline.
FeatureNorm: L2 Feature Normalization for Dynamic Graph Embedding	Graph Convolutional Network (GCN), Graph Attention Network (GAT)	• Introduces a novel method, L2 feature normalization, to address the feature shrinkage and over smoothing problems in dynamic graph embedding. • It provides a thorough analysis of the feature shrinkage and over smoothing issues in dynamic graph embedding.	• The effectiveness of the proposed method may be limited to the specific datasets and scenarios considered in the experiments, and its generalization to other dynamic graph problems needs further validation.

Table 2.1 : Literature Survey Comparison

CHAPTER 3

DATASET

3.1. Dataset Description

The website “Lawyers Reference” (https://devgan.in/ipc/) contains the collection of IPC sections along with their corresponding descriptions. This website serves as a comprehensive resource where each IPC section is accompanied by a detailed explanation outlining its legal provisions, scope, implications, and illustrations. The website is structured to provide easy access to essential information pertaining to the Indian Penal Code, facilitating in-depth analysis and understanding of the legal framework established by the IPC.

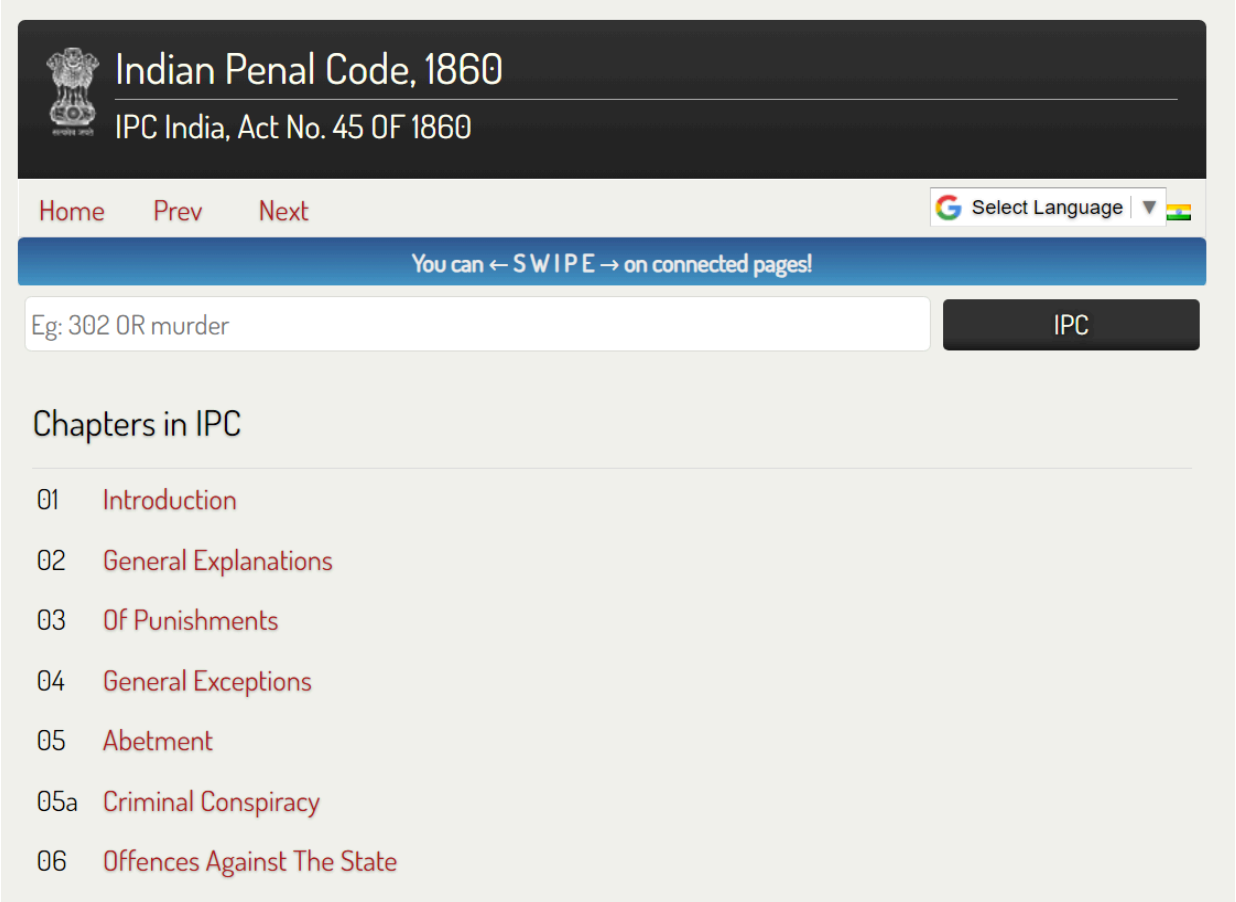

Figure 3.1 - IPC Section Repository Website

The data in the webpage can be scraped to produce structured data in the required format. However, data in the webpages can vary significantly in layout, design, and content presentation, leading to inconsistencies in the scraped data. Data in an Excel spreadsheet offers structured organization with rows and

columns, facilitating easy navigation, filtering, and analysis. This format ensures consistency in formatting and structure, unlike the varied layouts of web pages, which can lead to inconsistencies in scraped data. Data in the CSV format is readily accessible, shareable, and compatible with a wide range of software tools for manipulation and analysis. This level of control allows for custom transformations and analyses to suit specific needs, making CSV a versatile format for data processing.

	A	Section	Description
2	53	Punishments	The punishments to which offenders are liable under the provisions of this Code are:Death;Impris
3	53A	Construction of reference to transportation	Subject to the provisions of sub-section (2) and sub-section (3), any reference to “transportation f
4	54	Commutation of sentence of death	In every case in which sentence of death shall have been passed, the appropriate Government m
5	55	Commutation of sentence of imprisonment for life	In every case in which sentence of imprisonment for life shall have been passed, the appropriate
6	55A	Definition of “appropriate Government	In sections 54 and 55 the expression “appropriate Government” means,—in cases where the sen
7	56	Sentence of Europeans and Americans to penal servitude	Proviso as to sentence for term exceeding ten years but not for life – Rep. by the Criminal Law (F
8	57	Fractions of terms of punishment	In calculating fractions of terms of punishment, imprisonment for life shall be reckoned as equival
9	58	Offenders sentenced to transportation how dealt with until transported	Rep. by the Code of Criminal Procedure (Amendment) Act, 1955 (26 of 1955)
10	59	Transportation instead of imprisonment	Rep. by the Code of Criminal Procedure (Amendment) Act, 1955 (26 of 1955)
11	60	Sentence may be (in certain cases of imprisonment) wholly or partly rigorous or si	In every ease in which an offender is punishable with imprisonment which may be of either descri
12	61	Sentence of forfeiture of property	Rep. by the Indian Penal Code (Amendment) Act. 1921 (16 of 1921)
13	62	Forfeiture of property in respect of offenders punishable with death, transportatior	Rep. by the Indian Penal Code (Amendment) Act. 1921 (16 of 1921) section 4
14	63	Amount of fine	Where no sum is expressed to which a fine may extend, the amount of fine to which the offender
15	64	Sentence of imprisonment for non-payment of fine	In every case, of an offence punishable with imprisonment as well as fine, in which the offender is
16	65	Limit to imprisonment for non-payment of fine, when imprisonment and fine award	The term for which the court directs the offender to be imprisoned in default of payment of a fine s
17	66	Description of imprisonment for non-payment of fine.	The imprisonment which the Court imposes in default of payment of a fine may be of any descript
18	67	Imprisonment for non-payment of fine, when offence punishable with fine only	If the offence be punishable with fine only, the imprisonment which the Court imposes in default o
19	68	Imprisonment to terminate on payment of fine	The imprisonment which is imposed in default of payment of a fine shall terminate whenever that
20	69	Termination of imprisonment on payment of proportional part of fine	If, before the expiration of the term of imprisonment fixed in default of payment, such a proportion
21	70	Fine leviable within six years, or during imprisonment – Death not to discharge prc	The fine, or any part thereof which remains unpaid, may be levied at any time within six years aft
22	71	Limit of punishment of offence made up of several offences	Where anything which is an offence is made up of parts, any of which parts is itself an offence, th

Figure 3.2 - Dataset

The dataset is now presented in the CSV (Comma-Separated Values) file, where each row corresponds to an IPC section, and the key attributes in the dataset are given below.

- Section Number:
    - Each section number in the dataset corresponds to a specific section of the IPC, where every section is uniquely identified by a section number.
- Section Title:
    - Descriptive title or heading associated with each IPC section.
- Description of the Section:

The description of each IPC section, outlines the legal provisions, definitions, stipulations, and illustrations.

CHAPTER 4

PROPOSED SYSTEM

The system is used to provide legal assistance. The legal assistance system uses the BERT model that is modified to provide improved legal assistance. This system is discussed below in a detailed manner.

4.1. Initial process

The initial process of our legal assistance system involves obtaining user input and performing a series of essential steps to ensure accurate and effective assistance. This input then undergoes a series of crucial processes to ensure the system can understand and respond appropriately. These steps are explained below.

1. Tokenization

   The process of dividing user input into smaller pieces known as tokens is known as tokenization. These symbols could represent words, sentences, or other significant components. Example: For the input "I need legal help", tokenization would produce the tokens ["I", "need", "legal", "help"].

2. Semantic Analysis

   Semantic analysis aims to understand the meaning of the input sentence by analyzing the relationships between words and phrases. Example: In the input "I need legal help", semantic analysis would recognize that the user is requesting assistance related to legal matters.

3. Error Correction

   Error correction involves identifying and rectifying any spelling or grammatical errors in the user input. Example: If the user inputs "I nead legel hep", error correction would correct it to "I need legal help".

4. POS(Part-of-Speech) Tagging

   POS tagging allocates grammatical tags (such as noun, verb, adjective, etc.) to each word in the input. Example: For the input "I need legal help", POS tagging would tag "I" as a pronoun, "need" as a verb, "legal" as an adjective, and "help" as a noun.

5. Vectorization

   The technique of turning words or phrases into numerical vectors for use in mathematical processes is known as vectorization. Example: A

high-dimensional vector is used to represent each word in the input sentence. For instance, "I" might be represented as [0.1, 0.5, 0.3, ...], "need" as [0.2, 0.4, 0.6, ...], and so on.

Afterward, the vectors undergo L2 normalization to standardize their scale for consistent comparison. Finally, the system calculates cosine similarity between the normalized vectors and word embeddings obtained from BERT, a pre-trained language model, to gauge the similarity between the user input and legal concepts.

4.2. BERT Model

BERT, short for Bidirectional Encoder Representations from Transformers, is a groundbreaking natural language processing (NLP) model developed by Google researchers in 2018. It represents a major leap forward in understanding the contextual semantics of language, significantly improving performance across a range of NLP tasks including text classification, named entity recognition, and question answering. Unlike traditional models that process text sequentially, BERT utilizes a bidirectional approach, simultaneously considering both left and right contexts of words. This allows BERT to capture richer contextual relationships within sentences, distinguishing it from earlier methods constrained by their sequential processing approach.

The BERT model follows a two step process:

1. Pre-training on Large amounts of unlabeled text to learn contextual embeddings.
    - BERT is trained on vast amounts of unlabeled text data to acquire contextual embeddings, which represent words considering their context within a sentence. BERT performs several unsupervised pre-training tasks, such as predicting missing words in a sentence (Masked Language Model or MLM task), understanding sentence relationships, or forecasting the next sentence in a pair.
2. Fine-tuning on labeled data for specific NLP tasks.
    - After the pre-training phase, BERT, with its contextual embeddings, undergoes fine-tuning tailored to specific natural language processing (NLP) tasks. This fine-tuning process tailors the model's general language understanding to the nuances of the given task, using labeled data specific to tasks like sentiment analysis, question-answering, or named entity recognition. Parameters of the

model are adjusted to optimize performance for each task's requirements.

4.3. Working of BERT

BERT is designed to use solely the encoder mechanism to generate a language model. Tokens are fed into the Transformer encoder, where they undergo vector conversion before being processed by the neural network. A series of vectors that each represent an input token and provide contextualized representations make up the output. The difficulty lies in selecting a prediction target for language model training. Many models take a directed approach, which may limit context learning, by predicting the word that will appear next in a sequence. BERT uses two cutting-edge training techniques to address this issue:

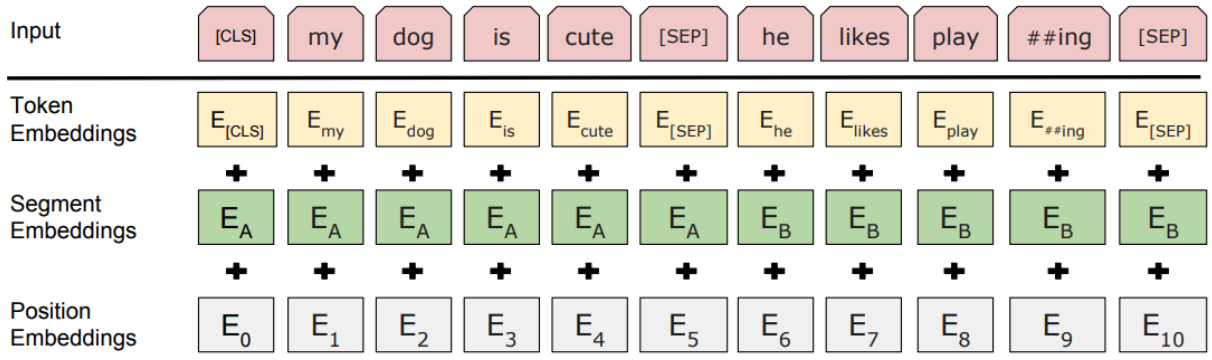

Figure 4.1 - Input Encoding

Prior to addressing the issue statements, which are Masked Language Model (MLM) and Next Sentence Prediction (NSP), as seen in Figure 5.1, all input must be encoded into the aforesaid format. Our objective is to use these problem statements to produce word embeddings as a "side effect."

4.3.1. Masked Language Model (MLM)

During the pre-training phase of BERT, a subset of words in every input sequence are hidden, and the model is trained to forecast the hidden words' initial values by analysing the context given by the surrounding words.

- Making words: Before BERT learns from sentences, it employs a technique called word masking. This involves hiding a portion of the words (around 15%) in the input sentences and replacing them with a special symbol, typically [MASK].
- Guessing Hidden Words: BERT's task is to determine what the hidden words are by analyzing the context of the surrounding words. It's akin to a guessing game where some words are missing, and BERT attempts to fill in the blanks based on the context provided by the adjacent words.

- How BERT learns: BERT learns by incorporating a specialized layer into its learning framework to make predictions about hidden words. It evaluates the accuracy of these predictions by converting them into probabilities, indicating its confidence level in its guesses. For instance, it might assert, "I believe this word is X, and I have high confidence in this assessment."
- Special Attention to Hidden Words: During training, BERT primarily focuses on accurately predicting the hidden words. It places less emphasis on predicting the words that are not hidden. This strategy is because the real challenge lies in figuring out the missing parts, which helps BERT excel in understanding the meaning and context of words.

BERT introduces a classification layer above the encoder output, which plays a crucial role in predicting the masked words. The resulting vectors from this stratum undergo multiplication with the embedding matrix to synchronize them with the vocabulary dimension, thereby facilitating the prediction of the representations of the vocabulary words. Subsequently, the SoftMax activation function is employed to determine the likelihood of every word in the vocabulary, producing a probability distribution across the complete vocabulary for each masked position.

During training, BERT focuses solely on predicting the masked values, using a loss function that penalizes the variance between its predictions and the actual masked words. This approach causes BERT to converge at a slower rate compared to directional models because it disregards the prediction of non-masked words. However, this slower convergence is offset by the enhanced contextual understanding gained through prioritizing masked value prediction.

4.3.2. Next Sentence Prediction (NSP)

BERT uses a classification layer to transform the output of the [CLS] token into a 2×1 shaped vector, which helps it predict whether the second sentence is related to the first. The SoftMax function is then used to determine the likelihood that the second sentence comes after the first.

By determining whether the second sentence in a document follows the first, BERT gains an understanding of the link between sentence pairs during training. In half of the input pairs, the second sentence is what comes after the first sentence in the original document, while in the other half, a sentence is selected at random. Prior to entering the model, the input undergoes preprocessing in order to aid in the differentiation of connected and unconnected sentence pairs.

- A [CLS] token appears at the beginning of the first sentence, and a [SEP] token is added at the end of each sentence.
- A sentence embedding designating whether a given token in the input belongs in Sentence A or Sentence B is added to each token.
- The location of each token in the sequence is indicated by a positional embedding.
- BERT uses a classification layer to transform the output of the [CLS] token into a 2×1 shaped vector, which helps it predict whether the second sentence is related to the first. The SoftMax function is then used to determine the likelihood that the second sentence comes after the first.

The Next Sentence Prediction (NSP) and Masked Language Model (MLM) tasks are trained concurrently with the BERT model. The goal of the model is to minimize the total loss function of the MLM and NSP. This produces a strong language model that is excellent at comprehending the relationships and context of individual sentences.

4.4. Hyper Parameter Tuning of BERT model

Hyperparameter tuning involves selecting the optimal set of hyperparameters for a machine learning model to maximize its performance on a specific dataset. Hyperparameters, which define the model's architecture and learning process, are manually set before training and are distinct from learned parameters. This iterative process aims to improve the model's effectiveness by fine-tuning its configuration.

Several key hyperparameters include the learning rate, batch size, number of epochs, regularization strength, number of hidden layers, and the number of units within each layer. The process of hyperparameter tuning seeks to identify the optimal combination of these parameters to maximize the performance of the model.

The hyperparameters tuned for this BERT model is explained below

1. Hidden Layers

   Hidden layers within the framework of the transformer architecture are designated for pre-training purposes. BERT is composed of a series of uniform transformer encoder layers, with each layer consisting of various sub-layers including multi-head self-attention mechanisms and feedforward neural networks. The quantity of hidden layers in BERT signifies the model's complexity, denoting the quantity of transformer encoder layers arranged consecutively.

The number of hidden layers in the BERT model is increased from 12 to 24. Increasing the number of hidden layers from 12 to 24 in BERT enhances the model's capacity to capture intricate patterns and dependencies in text. This allows for deeper contextual understanding and potentially better performance on complex tasks. However, it also increases computational cost and may require additional data for effective training.

2. Attention Heads

The BERT model is structured such that each transformer encoder layer incorporates numerous attention heads. These attention heads facilitate the model in directing its focus towards dislinct segments of the input sequence concurrently, thereby encompassing various facets of the context. BERT utilizes self-attention to discern dependencies among words in a given sentence, with individual attention heads dedicated to specific segments of the input sequence to acquire unique insights into word relationships. Through the utilization of multiple attention heads, BERT adeptly captures a wide array of features and relationships present within the input text.

The number of attention layers in the BERT model is increased from 12 to 24. Increasing the number of attention heads in the BERT model from 12 to 24 enhances its capacity to capture intricate relationships and nuances within the input text. This expansion allows for more comprehensive contextual understanding and potentially improved performance on complex tasks. However, it also significantly increases computational requirements and may necessitate larger datasets for effective training.

4.5. Generating Embeddings using BERT

Following hyperparameter tuning of the BERT model, word embeddings are acquired. Unlike prior models that processed text in a single direction, BERT adopts a bidirectional approach. Utilizing a transformer architecture, it considers both preceding and following words when encoding a word in a sentence. This bidirectional context understanding enables BERT to grasp a word's meaning based on its complete context within a sentence.

We use this BERT model to generate embeddings of all words in a sentence using the context. This makes BERT tolerant to homonyms(same words and different meanings)

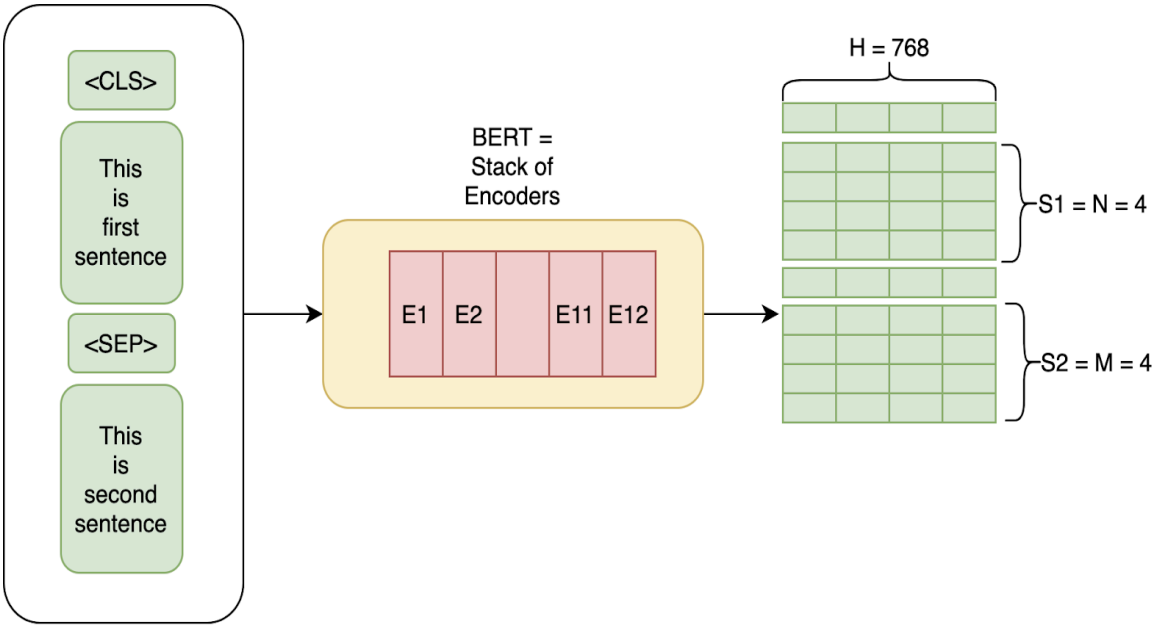

Figure 4.2 - Generating Word Embeddings

The word embeddings are extracted from the last layer of the BERT model.The embeddings have 768 fields which make sure that words have unique embeddings depending on the context of the sentence as shown in Figure 5.2. Now the problem is to model the sections data such that there exists unique vectors for all sections. We use these embeddings to generate sentence embeddings using mean along each 768 fields of all word embeddings in a sentence as shown in Figure 5.3.

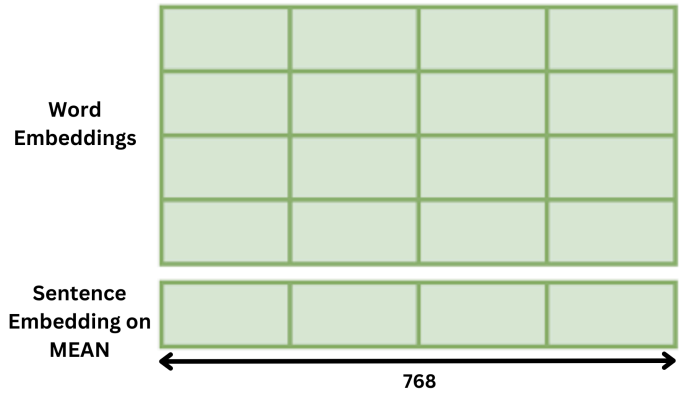

Figure 4.3 - Generation of Sentence Embeddings

4.6. L2 Normalization of the Section Vector

After obtaining word embeddings, preprocessing is crucial for consistent and meaningful representations. L2 normalization is a key step in this process,

adjusting embeddings to have a constant Euclidean norm. This promotes numerical stability and enhances learning effectiveness in downstream tasks.

The magnitude of the sentence embedding will increase depending on the number of words in a sentence or section. In order to capture only the uniqueness and context of a sentence, L2 Normalisation is used to make the magnitude of the vectors equal to 1 as shown in Figure 5.4.

$$x_{norm} = \frac{x}{\|x\|_2}$$

$$\|x\|_2 = \sqrt{x_1^2 + x_2^2 + x_3^2 + \ldots + x_m^2}$$

Figure 4.4 - L2 Normalization of Sentence vectors

4.7. Cosine Similarity of User Input and Section Data

Finally in order to measure the similarity of user input and section data, we employ cosine similarity on the vectors generated as a result of the fore-mentioned processing techniques on the section data and user data. Thus we derive the similarity scores of sections and the user input.

$$similarity(A, B) = cos(\theta) = \frac{A \cdot B}{||A||\,||B||} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sum_{i=1}^{n} B_i^2}.$$

Figure 4.5 - Cosine similarity

After calculating cosine similarity, we identify sections from the Indian Penal Code (IPC) that closely match the user input based on similarity scores. These sections, ranked by their similarity to the input, are displayed in a user interface of the web app created through Streamlit. Users can easily navigate and explore the IPC sections most relevant to their query, facilitating efficient patent search and analysis.

4.8. Streamlit Interface

Streamlit is a widely-used Python library for creating interactive web applications with ease and speed. It simplifies the creation of data-driven web apps, enabling developers to concentrate on writing Python code without needing expertise in web development intricacies. With Streamlit, you can create

interactive user interfaces for data exploration, visualization, machine learning models, and more. The library provides a simple and intuitive API for creating various components such as buttons, sliders, text inputs, and data displays. Developers can write code in a single Python script, defining the layout and behavior of the application.

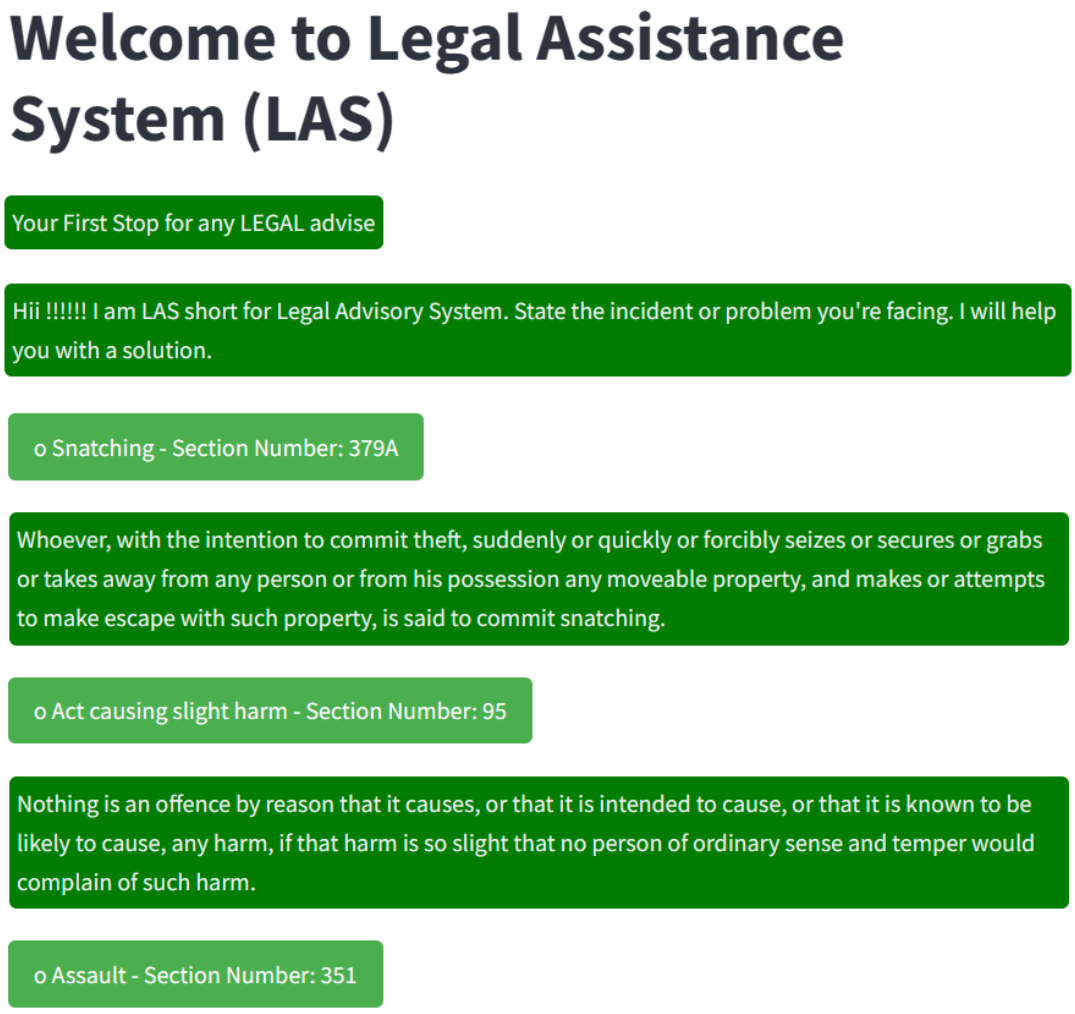

Figure 4.6 - Legal Assistance System UI

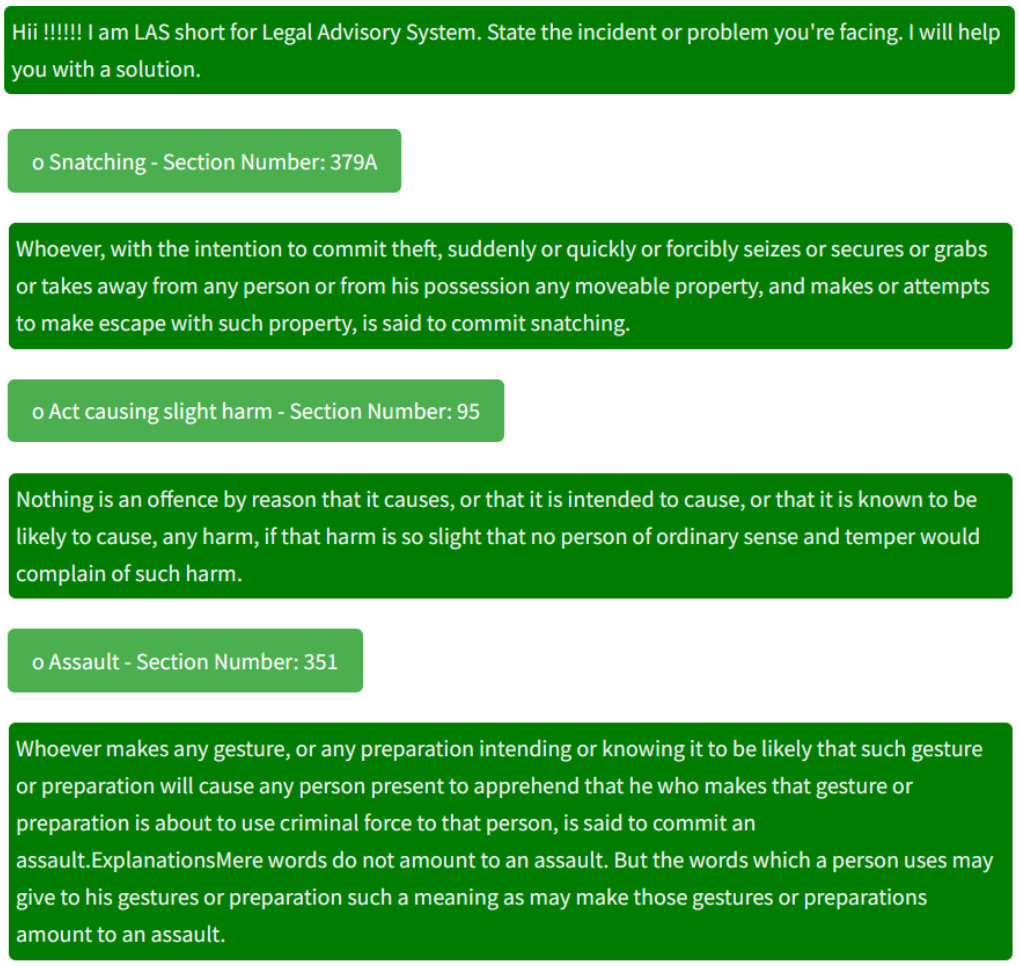

Figure 4.7 - Legal Assistance System UI

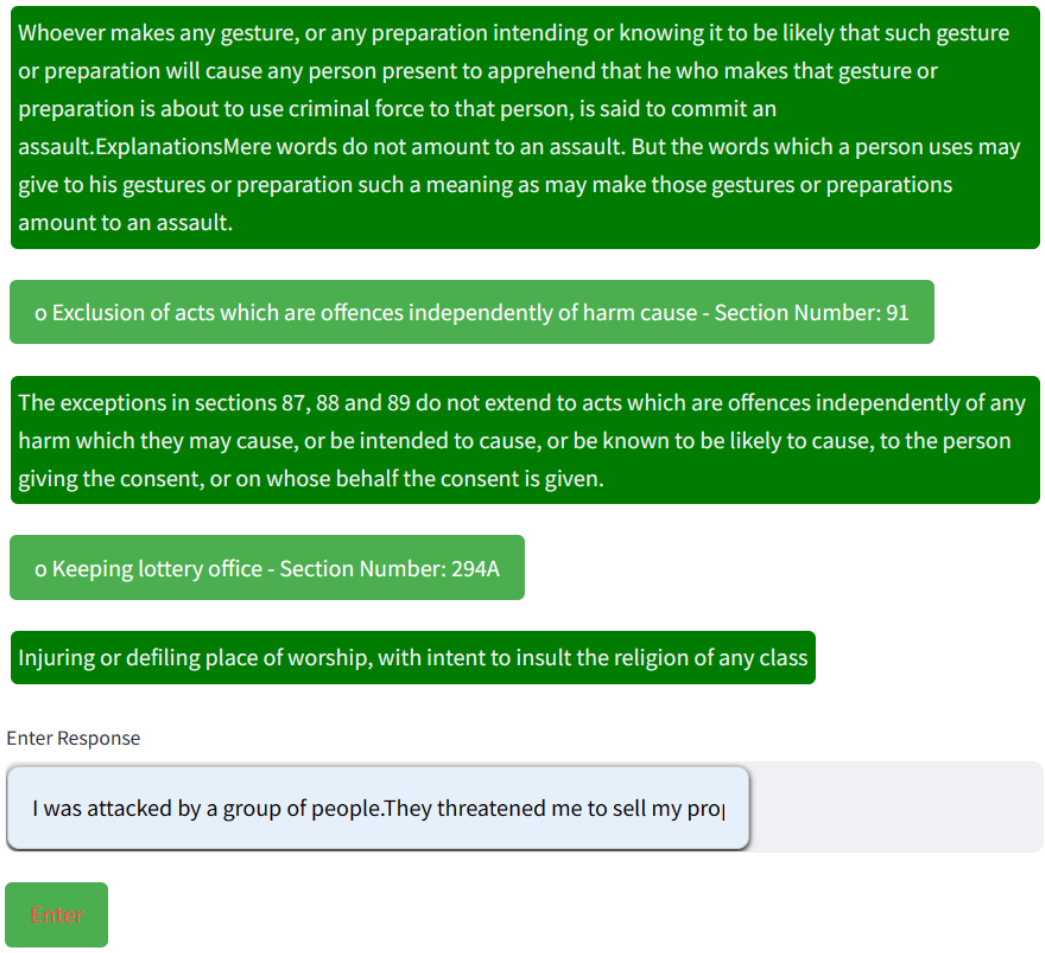

Figure 4.8 - Legal Assistance System UI

CHAPTER 5

RESULTS

5.1 Results of Legal Assistance System

The Legal Assistance System relies on a tuned BERT model with L2 normalization to provide accurate recommendations for user incidents. This model efficiently vectorized the Penal code, which contains various sections, as well as user incident input. To measure the system's efficiency, the TOP-10 method is employed. This method calculates the accuracy of the model by averaging the related recommendations among the top 10 suggestions for 30 user incidents.

In practice, this means that the system evaluates how well its recommendations align with the user's needs by considering the most relevant sections of the Penal code. A chatbot is integrated into the system to demonstrate its functionality. Users can input their incident queries, and the chatbot retrieves related laws from the dataset.

Through this interface, users can obtain legal advice and information quickly and easily. By leveraging advanced natural language processing techniques, the system ensures that users receive accurate and relevant recommendations. Additionally, the TOP-10 evaluation method provides a quantitative measure of the system's performance, ensuring its effectiveness in assisting users with legal queries.

Some of the User Incident Queries used for the TOP - 10 testing are listed below:

1. A group of individuals entered my house to rob, and during the robbery, they assaulted me.
2. A government official misuses his position and stole public funds
3. An individual forges documents to secure a loan from a bank
4. During a religious procession, some individuals entered a store, assaulted the owner, and stole valuable items
5. A candidate during an election campaign spreads false information about the opponent to sway voters

Model	Avg TOP - 5 Score
BERT	0.16
BERT with L2	0.32

Hyperparameter tuned BERT	0.24
Tuned BERT with L2 (proposed model)	0.56

Table 5.1 - Results of the proposed model using Top 5 score

Model	Avg TOP - 10 Score
BERT	0.12
BERT with L2	0.46
Hyperparameter tuned BERT	0.32
Tuned BERT with L2 (proposed model)	0.68

Table 5.2 - Results of the proposed model using Top 10 score

The table shows the comparison between different BERT models and the proposed model. From the table, it is clear that the proposed model outperformed the other BERT models by a considerable difference. The increase in accuracy of the bert model is primarily due to L2 normalization employed. L2 normalization works as a threshold for the vector created by BERT by providing the same weights to a Penal Code with a small number of words and one with a large number of words. And also it makes the magnitude of the vectors equal to unity.

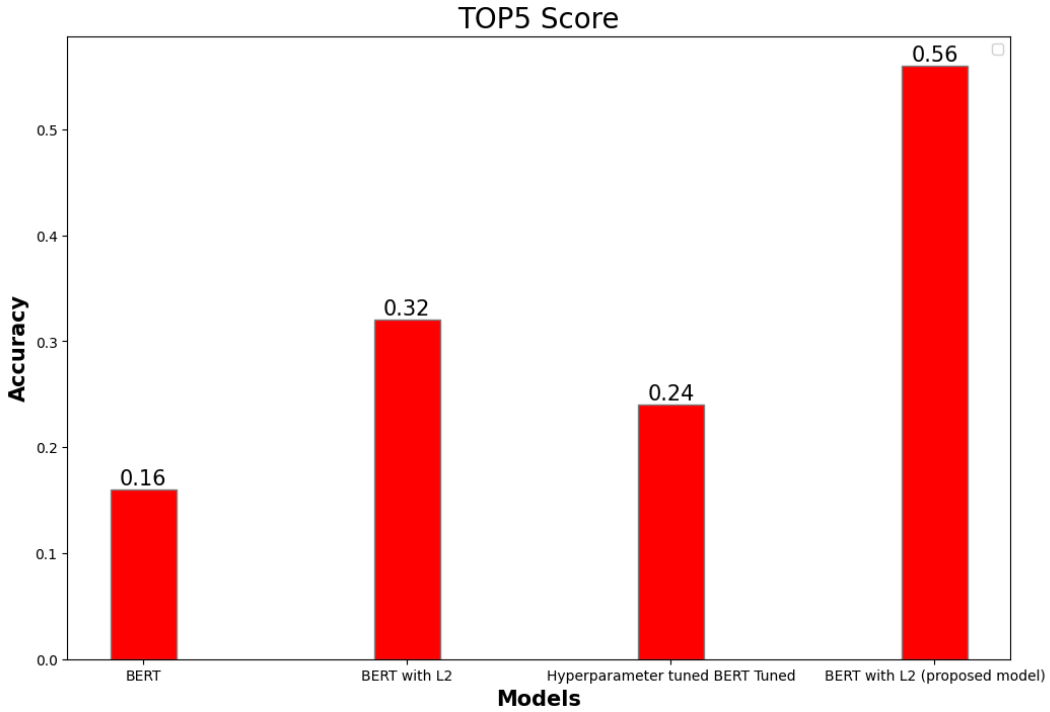

Fig. 5.1 - Top 5 scores

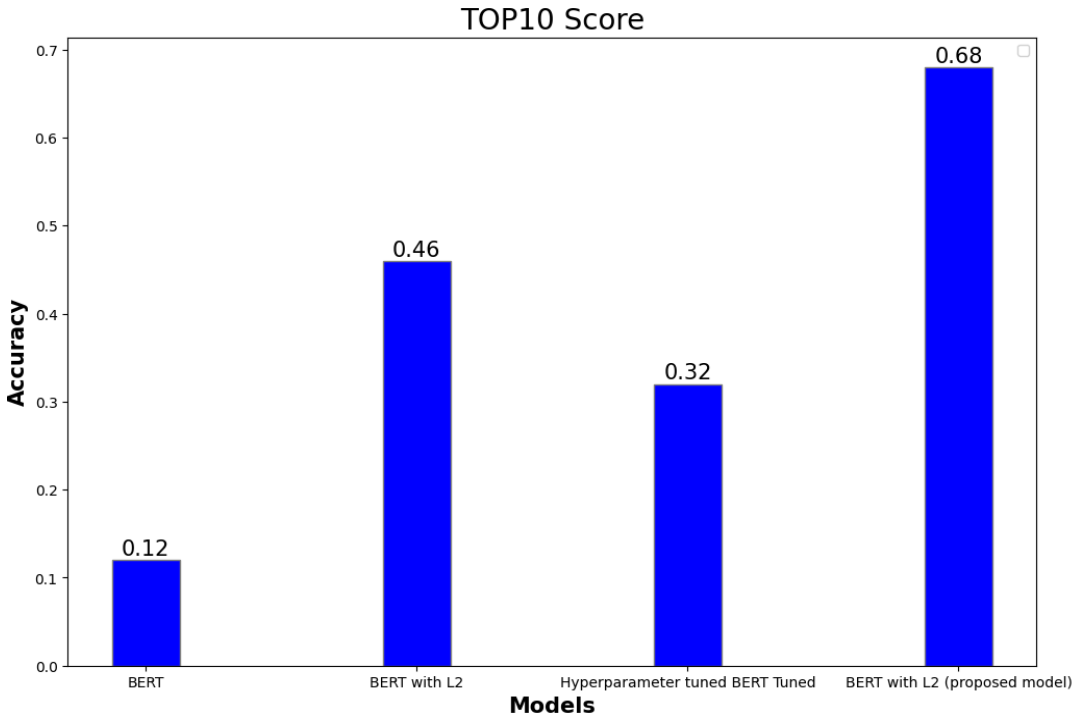

Fig 5.2 - Top 10 scores

CHAPTER 6

SYSTEM ANALYSIS

This chapter focuses on the hardware and software requirements essential to develop, train, test, and implement the system and its modules. It also discusses the datasets used, along with the feasibility of the system.

6.1 Hardware Requirements

The hardware requirements mentioned below are the minimum setting required to deploy the project in a computer system.

- OS: Windows, Mac, or Ubuntu
- CPU: Intel UHD 620 or Radeon Pro 5500 M equipped with powerful GPUs and has dual cores and a clock speed of 2.0 GHz
- Minimum Cores: 6
- Graphical Processor: 16 GB RAM
- Minimum System RAM: 50 GB SSD
- GPU: Nvidia Quadro RTH 4000.

6.2 Software Requirements

The software requirements mentioned are the minimum software settings necessary to run the project in a computer system.

- Python 3.11.4 or above
- Google Colaboratory
- Python Libraries
    1. Tensorflow

        A machine learning framework developed by Google, TensorFlow is open-source and extensively used for constructing and training deep learning models. It provides flexibility and scalability, making it suitable for various applications.

    2. Spacy

        spaCy is an open-source software library for advanced natural language processing (NLP), developed in Python and Cython. It is licensed under the MIT license and primarily maintained by Matthew Honnibal and Ines Montani, founders of Explosion, a software company.

3. NumPy

   A foundational Python library for numerical computing, facilitating efficient operations on large arrays and matrices, crucial for scientific and mathematical applications.

4. Scikit-learn

   Scikit-learn, often referred to as sklearn, is a Python library for machine learning and data modeling, available as open-source. It includes a range of classification, regression, and clustering algorithms like support vector machines, random forests, gradient boosting, k-means, and DBSCAN. The library is designed to seamlessly integrate with other Python libraries such as NumPy and SciPy.

5. PyTorch

   PyTorch, an open-source machine learning (ML) framework, is built on Python and Torch libraries. Torch, also open-source, is a ML library primarily used for creating deep neural networks and is written in the Lua scripting language. PyTorch is a preferred platform for deep learning research, streamlining the transition from research prototyping to deployment.

6. contextualSpellCheck

   contextualSpellCheck offers custom Spacy extensions that your code can utilize, simplifying data retrieval for users. With extensions at the doc, span, and token levels, contextualSpellCheck enhances ease of access to desired information.

7. NLTK

   Natural Language Toolkit (NLTK) is a collection of libraries and tools designed for both symbolic and statistical natural language processing tasks in English. Written in Python, it provides support for various functionalities such as classification, tokenization, stemming, tagging, parsing, and semantic reasoning.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1. Conclusion

In conclusion, the legal assistance system for criminal violence represents a groundbreaking solution to the challenges individuals face when seeking legal guidance. By leveraging Natural Language Processing (NLP) technology, the system revolutionizes the way users access and understand legal information. Its key features, including natural language interaction, accurate analysis of case details, comprehensive insights into legal reasoning, and user-friendly interface, empower individuals with varying levels of legal expertise to navigate the complexities of criminal violence cases confidently and effectively.

Additionally, the system's ongoing learning and development guarantee that it stays current with the most recent legal norms and practices, giving consumers dependable assistance throughout time. Through establishing a connection between the legal system and those in need of aid, the project advances accessibility and inclusivity in legal services.

Ultimately, the legal assistance system for criminal violence aims to democratize legal knowledge and foster a more just and equitable society. By offering personalized, accurate, and accessible guidance, the system empowers individuals to make informed decisions and take appropriate legal actions, contributing to enhanced safety, efficiency, and fairness in the legal process.

7.2. Future Work

The Penal Code dataset can be made more diverse and large. This will help to increase the scope of the project further where it could be used to provide solutions for a wide variety of user incidents. BERT is able to provide unique vectors thanks to the large dimension of the embeddings and hence it can be used for deducting intricate details in the user incident.

The efficiency of the model can be increased by using the BERT embeddings to train an ML model which is trained on a dataset that contains user incident embeddings and corresponding Penal Code embeddings.This ML model could be then used to provide recommendations based on the user incidents.The BERT embeddings can be used for a variety of NLP tasks like Question-answering and Context Understanding.

CHAPTER 8

CHALLENGES

The Legal Assistance System for Criminal Violence Victims using Natural Language Processing holds immense potential for revolutionizing the legal assistance system for criminal violence, enabling a paradigm shift in how individuals access and understand legal information. However, as with any innovative technology, it comes with its fair share of challenges and hurdles.

These are the challenges faced:

a) **Data Quantity and Quality:** Acquiring sufficient data covering diverse criminal violence cases and legal nuances proves challenging due to fragmentation. Ensuring data quality involves navigating complex legal language, addressing biases, and filtering noise while safeguarding sensitive information.

b) **Accuracy and False positives:** Ensuring high accuracy while minimizing false positives in a Legal Assistance System for Criminal Violence Victims using Natural Language Processing (NLP) presents significant challenges. Legal texts are often intricate and nuanced, requiring precise interpretation to avoid misclassifications. Balancing the system's sensitivity to detect relevant legal information with the risk of generating false positives demands sophisticated NLP algorithms and careful validation processes.

c) **Scalability:** Achieving scalability encounters notable challenges. As the volume of legal data grows, ensuring efficient processing and analysis becomes increasingly complex. Adapting NLP models to handle large-scale datasets while maintaining performance standards necessitates robust infrastructure and optimization strategies, posing significant hurdles to system scalability.

d) **Environmental Variability:** Legal texts can vary widely in style, terminology, and jurisdictional nuances, complicating model generalization. Adapting NLP algorithms to accommodate this diversity while maintaining accuracy and reliability across different legal contexts requires careful consideration and continuous monitoring, posing significant hurdles to system consistency.

e) **Hardware Limitations:** Processing large volumes of legal text data requires substantial computational resources, which may exceed the

capabilities of available hardware. Addressing these constraints involves optimizing algorithms and infrastructure to balance performance requirements with hardware capabilities, necessitating careful resource management and potentially hindering system scalability and responsiveness.

CHAPTER 9

BIBLIOGRAPHY

1. V. Socatiyanurak et al., "**LAW-U: Legal Guidance Through Artificial Intelligence Chatbot for Sexual Violence Victims and Survivors,**" in IEEE Access, vol. 9, pp. 131440-131461, 2021, doi: 10.1109/ACCESS.2021.3113172.
2. V. Murali, R. J. Sarma, P. A. Sukanya, and P. Athri, **"ChEMBL Bot - A Chat Bot for ChEMBL database,"** 2018 Fourteenth International Conference on Information Processing (ICINPRO), Bangalore, India, 2018, pp. 1-6, doi: 10.1109/ICINPRO43533.2018.9096710.
3. Chenguang Pan and Wenxin Li, **"Research paper recommendation with topic analysis,"** 2010 International Conference On Computer Design and Applications, Qinhuangdao, China, 2010, pp. V4-264-V4-268, doi: 10.1109/ICCDA.2010.5541170.
4. M. Kordabadi, A. Nazari, and M. Mansoorizadeh, **"A Movie Recommender System based on Topic Modeling using Machine Learning Methods,"** International Journal of Web Research, vol.5, no.2, pp.19-28, 2022, doi: 10.22133/ijwr.2022.370251.1139.
5. U. C. De, B. B. Dash, T. M. Behera, T. Samant and S. Banerjee, **"Text-Based Recommendation System for E-Commerce Apparel Stores,"** 2023 International Conference for Advancement in Technology (ICONAT), Goa, India, 2023, pp. 1-4, doi: 10.1109/ICONAT57137.2023.10080436.
6. R. Shrivastava and D. S. Sisodia, **"Product Recommendations Using Textual Similarity Based Learning Models,"** 2019 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2019, pp. 1-7, doi: 10.1109/ICCCI.2019.8821893.
7. S. Larabi Marie-Sainte, N. Alalyani, S. Alotaibi, S. Ghouzali and I. Abunadi, **"Arabic Natural Language Processing and Machine Learning-Based Systems,"** in *IEEE Access*, vol. 7, pp. 7011-7020, 2019, doi: 10.1109/ACCESS.2018.2890076.
8. Tyagi, Nemika & Bhushan, Bharat. (2023). **Natural Language Processing (NLP) Based Innovations for Smart Healthcare** Applications in Healthcare 4.0. 10.1007/978-3-031-22922-0_5.

9. H. Liu et al., **"A Natural Language Processing Pipeline of Chinese Free-Text Radiology Reports for Liver Cancer Diagnosis,"** in IEEE Access, vol. 8, pp. 159110-159119, 2020, doi: 10.1109/ACCESS.2020.3020138.
10. M. Yang, Z. Meng and I. King, "**FeatureNorm: L2 Feature Normalization for Dynamic Graph Embedding**," 2020 IEEE International Conference on Data Mining (ICDM), Sorrento, Italy, 2020, pp. 731-740, doi: 10.1109/ICDM50108.2020.00082.
11. A. Trivedi, A. Trivedi, S. Varshney, V. Joshipura, R. Mehta and J. Dhanani, **"Extracted Summary Based Recommendation System for Indian Legal Documents,"** 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-6, doi: 10.1109/ICCCNT49239.2020.9225290.
12. H. Wan, **"A Novel Deep Information Search Algorithm for Legal Case Text Recommendation System,"** 2023 8th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 2023, pp. 1744-1749, doi: 10.1109/ICCES57224.2023.10192699.
13. D. V. Bagul and S. Barve, **"A novel content-based recommendation approach based on LDA topic modeling for literature recommendation,"** 2021 6th International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 2021, pp. 954-961, doi: 10.1109/ICICT50816.2021.9358561.
14. O. Kovalchuk, S. Banakh, M. Masonkova, K. Berezka, S. Mokhun and O. Fedchyshyn, **"Text Mining for the Analysis of Legal Texts,"** 2022 12th International Conference on Advanced Computer Information Technologies (ACIT), Ruzomberok, Slovakia, 2022, pp. 502-505, doi: 10.1109/ACIT54803.2022.9913169.
15. M. A. I. Mahmud et al., **"Toward News Authenticity: Synthesizing Natural Language Processing and Human Expert Opinion to Evaluate News,"** in IEEE Access, vol. 11, pp. 11405-11421, 2023, doi: 10.1109/ACCESS.2023.3241483.
16. B. Bulut, B. Kaya, R. Alhajj and M. Kaya, **"A Paper Recommendation System Based on User's Research Interests,"** 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), Barcelona, Spain, 2018, pp. 911-915, doi: 10.1109/ASONAM.2018.8508313.

17. P. Sirisha, G. L. Devi and N. Ramesh, **"Plot-Topic based Movie Recommendation System using WordNet,"** 2022 9th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2022, pp. 45-49, doi: 10.23919/INDIACom54597.2022.9763244.
18. T. Shaik *et al*., **"A Review of the Trends and Challenges in Adopting Natural Language Processing Methods for Education Feedback Analysis,"** in *IEEE Access*, vol. 10, pp. 56720-56739, 2022, doi: 10.1109/ACCESS.2022.3177752.
19. D. C. Cavalieri, S. E. Palazuelos-Cagigas, T. F. Bastos-Filho and M. Sarcinelli-Filho, **"Combination of Language Models for Word Prediction: An Exponential Approach,"** in *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 9, pp. 1481-1494, Sept. 2016, doi: 10.1109/TASLP.2016.2547743.
20. M. Omar, S. Choi, D. Nyang and D. Mohaisen, **"Robust Natural Language Processing: Recent Advances, Challenges, and Future Directions,"** in *IEEE Access*, vol. 10, pp. 86038-86056, 2022, doi: 10.1109/ACCESS.2022.3197769.