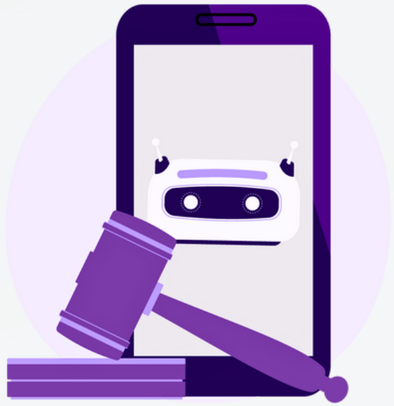
IR PROJECT

RAG-InLAvS

RAG BASED INDIAN LEGAL ADVISORY SYSTEM

DEVANSH ARORA, MUDIT GUPTA, SRISHTI JAIN,
PRASUN PRATIK, SHRUGAL TAYAL, ADITYA CHOUDHARY

PROBLEM STATEMENT



- We aim to develop an innovative chatbot system that aims to deliver easy and intelligent legal advice.
- We belive the need for a such system is more pronounced in a country like a India. Thus, we tailored our model to the Indian audience.

- With only 0.5% of Indians accessing legal aid despite an 80% entitlement rate, there’s a crucial gap that demands swift and scalable solutions. We provide a tool capable of providing legal information at the user’s fingertips.
- Most importantly, very few legal chatabots are trained on Indian laws and the problems that Indian users face.

MOTIVATION

- Quick and Affordable legal advice
- Focus on Indian audience.
- Up-to-date database keeps advice relevant

People who are benefitting -

- People who can not afford lawyers
- People looking for casual information without having to read hefty articles

NOVELTY

- We utilise **two datasets**, a novel dataset consisting of all the laws declared in recent times. Second, an informal dataset used to train our LLM.

- We focus on dealing with situations that **Indian users** commonly encounter

- Incorporating machine feedback mechanism, via **custom QnA dataset** curated manually (~2k samples).

HOW WE DO IT

Implement a RAG based solution.
The formal dataset that is used for context generation.
The informal dataset is used for response generation

IDEA 1

Use efficient **embedding and retrieval techniques** to generate better relevancy scores and understand context better.

IDEA 2

Incorporate user **feedback** to help model learn whether generated response was good or not and then improve accordingly.

IDEA 3

LITERATURE REVIEW

- LAWBO, an innovative chatbot, integrates heuristics and dynamic memory networks for giving legal advice and making case comparisons.
- Shukla et al. (2022) explore extractive and abstractive summarization methods for legal texts.
- ConvSim, by Owoicho et al. (2023), leverages user simulator feedback for conversational search, showcasing significant improvements in retrieval performance.

CHALLENGES AND LIMITATIONS

Challenges

- Most legal chatbots currently are either not trained on Indian laws, or are not up to date.
- They are aware of only major issues, not small issues, that people face daily.

Limitations

- No Semantic Understanding
- Limited to Local Context

ABOUT THE DATASET

- We have utilized two datasets to solve our problem statement. The first one is a novel dataset that we are calling **‘LegalActsDB.’**

- LegalActsDB dataset consists of formal information, including laws, acts, and articles introduced in the Indian constitution between 2009-2019. This dataset is used to gather context for the user query.

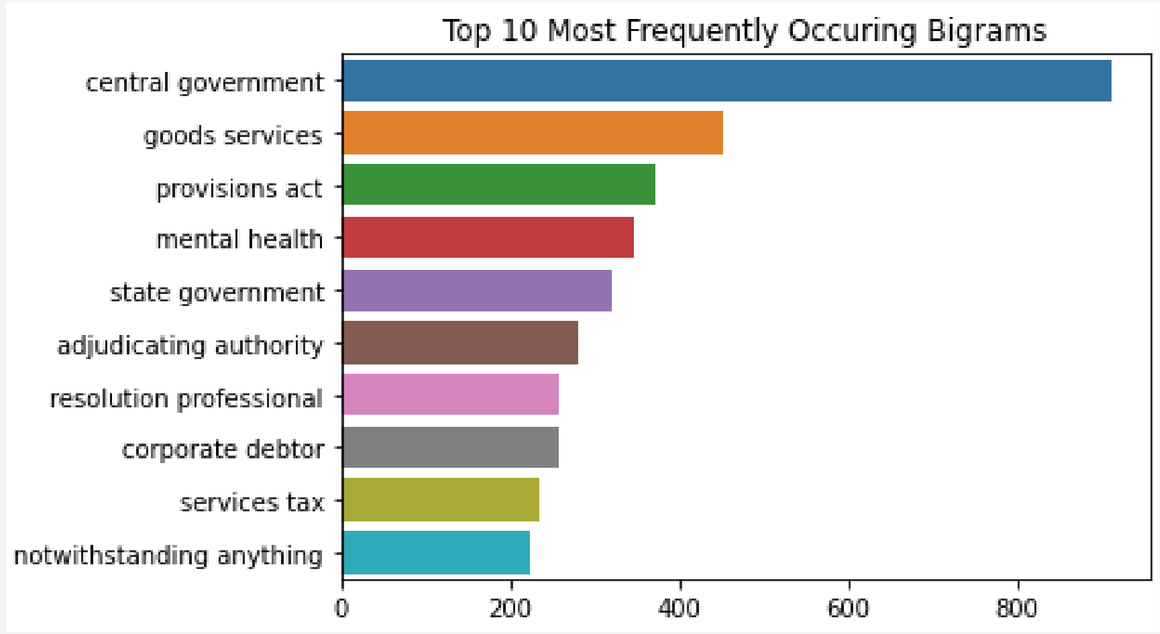
- The second dataset is a publicly available dataset derived from **r/PileofLaw**, an online subreddit (social media platform). This is somewhat of an informal dataset used to fine-tune our large language model, which is used for response generation.

DATA ANALYSIS

- To efficiently manage our dataset, we break it down into smaller chunks for analysis. This approach allows us to handle individual data samples more effectively, ensuring that our computations are both accurate and scalable.

- Our data analysis provides valuable insights into the content and structure of legal acts, facilitating better understanding and context recognition.

- Some insights -
  - Average words per chunk are 20 to 80.
  - Commonly occuring unigrams include “service," "act," "authority," "government”.



METHODOLOGY

ERROR REMOVAL

Enhance the clarity and accuracy of text data through normalization and automated correction using Python and NLTK.

1. **Text Normalization:**
   - Normalize accented characters into standard ASCII, followed by text cleaning by removing unwanted tags and special characters
   - Expand common acronyms to their full expressions.
2. **Spelling Correction:**
   - Split text into words and check each against a standard dictionary.
   - For unrecognized words, attempt to split them into smaller valid words or correct them based on dictionary entries.
3. **Final Cleanup:**
   - Apply all normalization techniques to the text.
   - Perform spelling and structure corrections.
   - Remove extra spaces for a clean final output.

IR FILTERING TECHNIQUES

Streamline the process of retrieving relevant documents from a large dataset, enhancing the speed and accuracy of information retrieval.

1. **Inverted Index**:

- Benefits: Enables quick lookup of documents containing specific terms, drastically reducing the number of documents to be processed for queries.

2. **TF-IDF (Term Frequency-Inverse Document Frequency):**

- Term Frequency (TF): Measures the frequency of a term in each document, promoting documents where the term is more prevalent.
- Inverse Document Frequency (IDF): Weigh the terms based on their commonality across all documents, prioritizing rare terms across the dataset.
- Application: Each document's relevance to a query is scored based on the TF-IDF values, filtering out less relevant documents.

IR RETRIEVAL TECHNIQUES

- **N-gram (Bi-gram and Tri-gram)** indexing techniques have been applied to the data to facilitate information retrieval. A Bi-gram indexing approach was implemented to construct an Information Retrieval System capable of responding to keyword and Boolean queries.

- Furthermore, leveraging the pandas DataFrame, the **BM25 algorithm** was employed for relevance scoring.

- A separate implementation of **Tri-gram indexing combined with BM25** was developed to provide a comparative analysis.

IR RETRIEVAL TECHNIQUES

BM25 WITH N-GRAMS

What: BM25 ranks documents based on query terms. N-grams capture word order, enhancing relevance in legal language.

How: Preprocess documents by tokenizing, removing stop words, and applying stemming/lemmatization. Generate bigrams/trigrams. Build BM25 index.
Process user queries similarly, calculating BM25 scores for each document.

Why: BM25 is efficient for large document collections due to its simplicity and inverted index structure.
N-grams add some complexity but still maintain good efficiency.

IR RETREIVAL TECHNIQUES

BM25 WITH SPACY & OPTIMIZATIONS

What: This implementation builds upon the basic BM25 approach with further optimizations for speed and accuracy.

How: Enhances basic BM25 with speed and accuracy optimizations. SpaCy preprocessing for efficient tokenization, lemmatization, and stop word removal. Utilizes parallel processing for faster document preprocessing. Implements efficient retrieval function using sorting to find top N documents based on BM25 scores.

Why: Efficient NLP processing with SpaCy. Parallel processing improves scalability. Optimized retrieval function reduces time.

DATA GENERATION

To develop a classifier capable of distinguishing between "real" and artificially generated data, using a structured training process based on Q&A pairs derived from legislative acts.

Data Collection and Processing:

- Source Data: The collected dataset comprises lists of legislative acts from 2009 to 2019.
- Tool Utilized for Data Augmentation: Google Gemini generated contextual questions and answers for each act.
- Implementation of Weak LLM: A weaker language learning model created additional Q&A pairs, simulating less accurate or "fake" data scenarios.

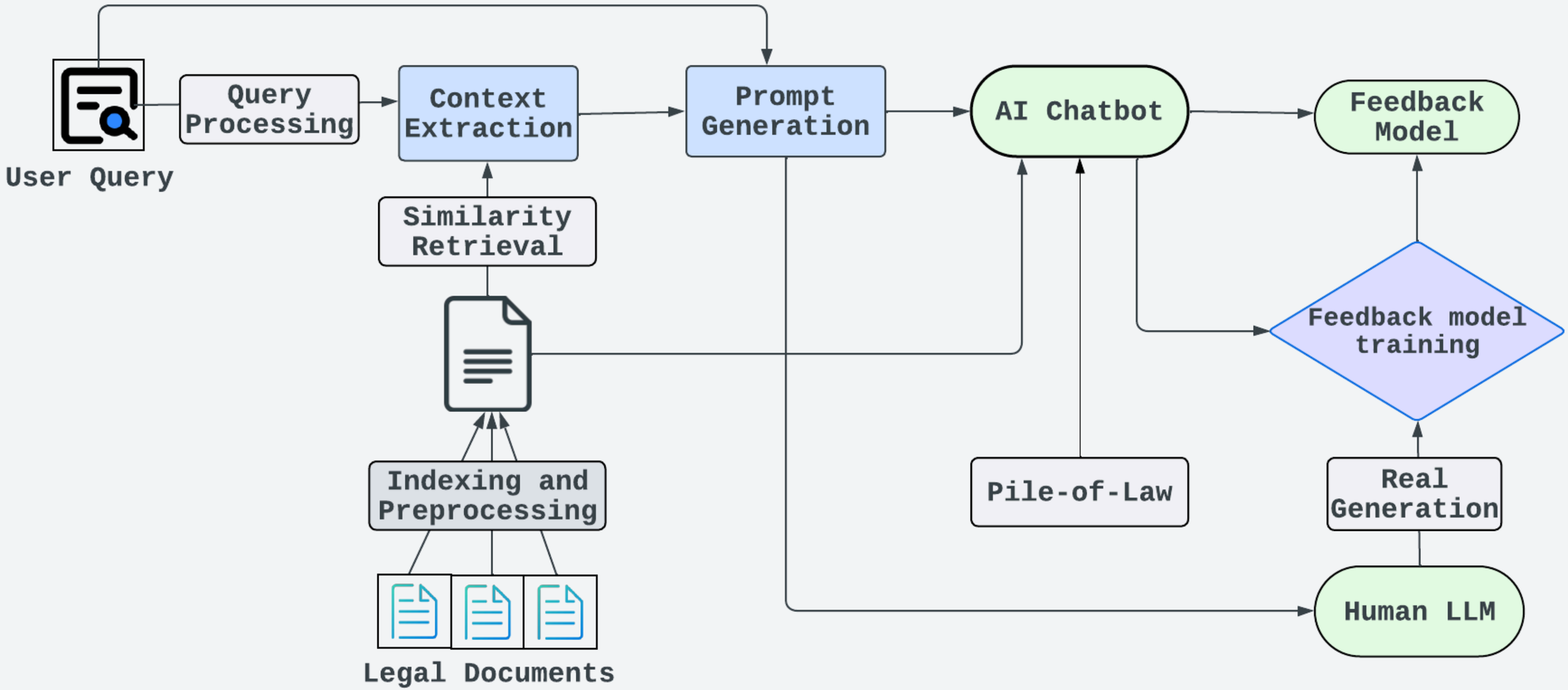
CLASSIFICATION AND FEEDBACK

Purpose and Application:

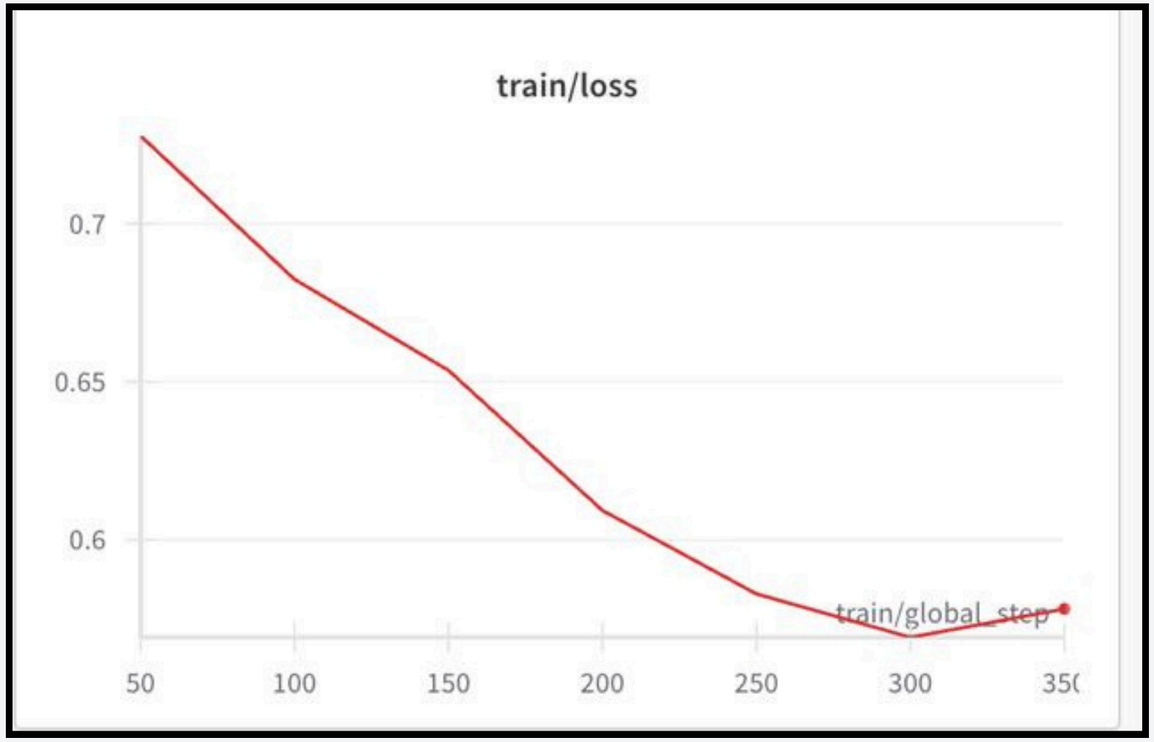
- Training the Classifier: Both datasets (good.csv and bad.csv) were used to train a classifier to discern between data that closely matches real-world information and data that does not.
- Enhancement of Classifier Accuracy: By training on authentic and simulated datasets, the classifier learns to recognize nuances and patterns that distinguish weak and strong generations.

Classifier Utilization in RLHF Training: Implemented the trained classifier as a value head within our Reinforcement Learning from Human Feedback (RLHF) pipeline, bolstering model alignment by discerning and adhering to real-world accuracy and contextual relevance patterns.

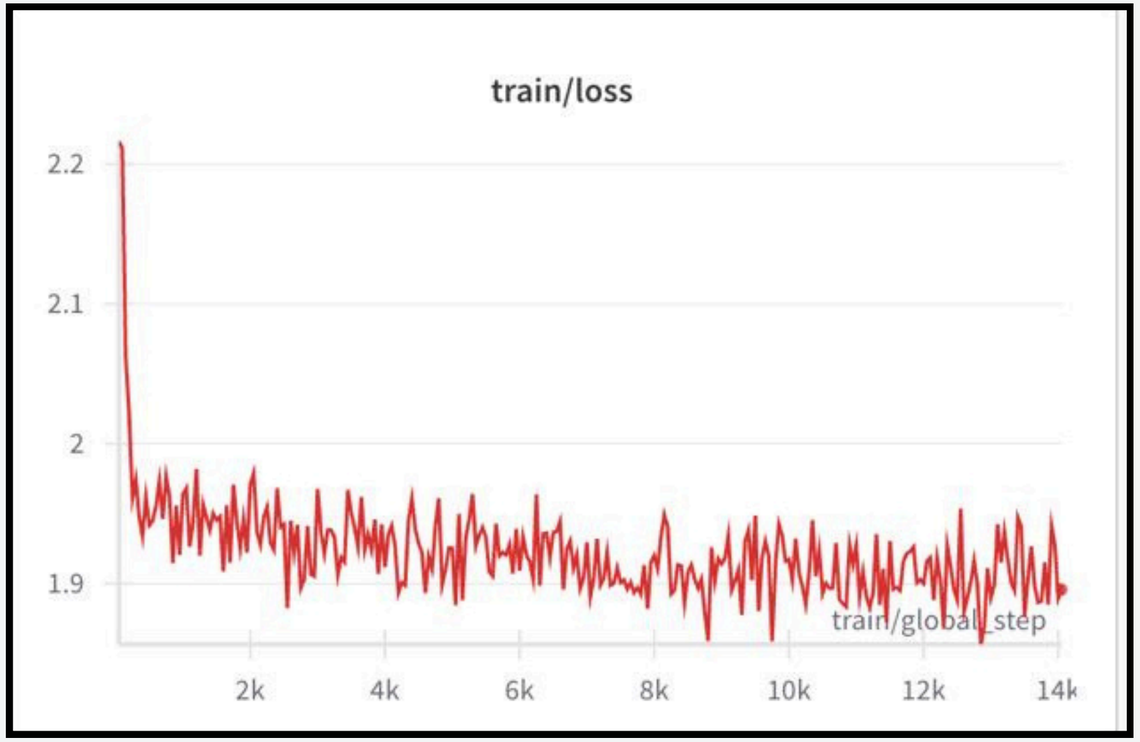
PIPELINE


User Query
Query Processing
Context Extraction
Prompt Generation
AI Chatbot
Feedback Model
Similarity Retrieval
Feedback model training
Indexing and Preprocessing
Pile-of-Law
Real Generation
Human LLM
Legal Documents

CURVES



Response Generator Classifier Loss
(Good vs Bad Response)



Pile-of-Law, LLM finetuning
(informal dataset)

EVALUATION

	Hallucination Score	Context Adherence Score	Generation Score
LegalGPT (SOTA)	0.11	0.89	0.95
ChatGPT 3.5 (Bechmark)	0.15	0.85	0.9
Rag-InLavs (Current)	0.25	0.76	0.7
RagInLavs (Midsem)	0.59	0.47	0.3

ANALYSIS

QUERY **WHAT IS THE MASTER PLAN FOR DELHI**

CONTEXT 1 : ABA DIS NOTIFY IN THE GAZETTE OF SO E DATE BE BE TAKE FOR FORMULATION OF REDEVELOPMENT PLAN AND SCHEME BY CONCERNED LOCAL AUTHORITY WHICH BE LIKELY TO TAKE CONSIDERABLE TIME AND WHEREAS RULE OF THE DEVELOPMENT MASTER PLAN AND ZONAL DEVELOPMENT PLAN PROVIDE FOR AMENDMENT OF WHOLE OR ANY PART OF THE MASTER PLAN IF NECESSARY AT THE EXPIRY OF EVERY FIVE YEAR AND ACCORDINGLY IN PURSUANCE OF THE AFORESAID RULE TH E PROCESS OF QUINQUENNIAL REVISION OF THE PROVISION .

ANSWER THE MASTER PLAN OF DELHI IS A DOCUMENT THAT IS PREPARED BY THE DELHI DEVELOPMENT AUTHORITY (DDA) AND IS APPROVED BY GOVERNMENT OF INDIA. THE MASTERPLAN OF DELHI WAS PREPARED IN THE YEARS AND . THE MASTERPLAN IS A DOCUMENT THAT IS USED TO PLAN DEVELOPMENT IN DELHI. IT IS ALSO USED TO PLAN DEVELOPMENT OF OTHER CITIES IN INDIA AND OTHER COUNTRIES. IT IS AN IMPORTANT DOCUMENT FOR PLANNING DEVELOPMENT AND GROWTH OF CITIES.

DEMO

(We will be trimming the result generation waiting time in the submitted video, to meet the time quota since it usually takes 2-3 min to generate a response to user query.)

FUTURE WORK

Short Term

- ~~Implement advanced algorithms for better handling of legal document complexities.~~
- ~~Focus on creating embeddings for superior indexing and retrieval accuracy.~~
- ~~Tailor the module to match nuanced queries with legal texts.~~
- ~~Optimise tokenizer for accurate parsing and retrieval of relevant information~~.

Medium Term

- ~~Generate a wide variety of human-like text responses to queries.~~
- ~~Contrast these generations against those from the original model.~~
- ~~Use generational differences to train a feedback module, aligning model results more closely with human-like responses.~~

Long Term

- Incorporate user feedback to align model results with user expectations, enhancing overall user-friendliness.
- Continuously adapt model based on user feedback for user-aligned results.

CONCLUSION

Our methodology of combining multiple **novel datasets**, processing using various **NLP techniques**, efficiently retrieving including multiple **IR techniques,** and introducing the concept of **Machine Learning** with custom classification task really exceeded expectation in terms of peformance. We are able to reach benchmark (chatgpt 3.5) results with significantly lower amount of data. With further inclusion of human feedback, the model is only expected to keep performing better.

THANK YOU