# LegalRAG: an LLM based RAG system to provide legal literacy and support

DISSERTATION

Submitted in partial fulfillment of the requirements of the

Degree : MTech in Artificial Intelligence and Machine Learning.

By

Sachin Shankar Hebbar
2022AC05016

Under the supervision of

Naveen Rathani, Quantitative Analytics Manager

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE
Pilani (Rajasthan) INDIA

December, 2024

**BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI**
**FIRST SEMESTER 2024-25**

DSECLZG628T / **AIMLCZG628T DISSERTATION**

Dissertation Title : LegalRAG: an LLM based RAG system to provide legal literacy and support

Name of Supervisor : Naveen Rathani

Name of Student : Sachin Shankar Hebbar

ID No. of Student : 2022AC05016

Courses Relevant for the Project & Corresponding Semester : 1. Conversational AI
2. Natural Language Processing
3. Information Retrieval
4. Deep Learning

## Abstract

The complex world of India's legal system, with its long history and vast literature, can intimidate many non-professionals. However, everyone needs to have a reasonable level of legal literacy to protect themselves from harm of different kinds. The advent of Large Language Models (LLM), in conjunction with the availability of vast amounts of public data regarding laws, acts, and cases, provides an opportunity to build a system that can act as the bridge between the legal system and the common people.

This work aims to build LegalRAG, a RAG-based system that uses an LLM to provide relevant responses and resources to users' legal queries related to Legal acts of inheritance, divorce, copyright, and consumer protection.

LLMs without any fine-tuning can be used to accomplish the task mentioned above. LLMs might have included legal information in their training datasets; they also carry tremendous knowledge through their parameters. However, this is general knowledge and is not explicitly tuned to contain Indian legal understanding. Other methods, like vanilla fine-tuning and Low-Rank Adapters (LoRA), would fine-tune the LLMs and equip them to answer legal queries, but these require more memory and resources to fine-tune the additional layers.

The proposed method of LegalRAG includes intent recognition followed by Retrieval Augmented Generation (RAG) architecture. User query passes through an intent

recognition model to identify user intent: inheritance, divorce, consumer protection, or copyright-related intent. Depending on the intent, the user query is routed to different RAG systems or knowledge bases.

RAG is a method that combines the LLM's generation capabilities with information retrieval capabilities. Each of the four RAG systems takes in the user query, embeds it, and searches a vector store or knowledge base of Indian Legal data corresponding to one of the four acts. Most relevant context from the search and the user query are sent to the LLM to generate a response.

The hindrances from the previous approaches are resolved through RAG-based architecture as they provide the domain knowledge the LLM would need to generate relevant responses and not hallucinate. This approach requires fine-tuning the intent recognition model but doesn't mandatorily require any fine-tuning of the LLM itself. Therefore, it consumes less time and resources. RAG has the additional benefit of source verification by providing information on the sources, such as the Act name and section number containing the selected context.

The novelty in this idea is with the intent recognition model, as more legal acts get included in the future; only the intent recognition model, which is much smaller in size than the LLM, needs to be re-trained without many updates to the LLM, therefore this approach requires less memory and resources..

LegalRAG would also include a Named Entity Recognition model to identify Personally Identifiable Information of any individuals and mask them before storing the data in vector stores or knowledge bases to protect the individual's data.

LegalRAG will benefit the general public by providing essential legal literacy and offering procedural steps and templates for filing cases under the abovementioned acts.

**Key Words:**

Large Language Model (LLM), Named Entity Recognition (NER), Intent Recognition, fine-tuning, Low-Rank Adapters, Retrieval Augmented Generation (RAG), Embedding model, Natural Language Processing (NLP), Legal Literacy

**BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI**
**I SEMESTER 24-25**
DSECLZG628T / **AIMLCZG628T DISSERTATION**
**Dissertation Outline (Abstract)**

**BITS ID No. 2022AC05016 Name of Student: Sachin Shankar Hebbar**

**Name of Supervisor:** Naveen Rathani

**Designation of Supervisor**: Quantitative Analytics Manager

**Qualification and Experience:** Integrated B.E (Chemical Engg) and M.Sc (Mathematics), 12 years experience in data science and machine learning field

**Official E- mail ID of Supervisor:** Naveen.Rathani@wellsfargo.com

**Topic of Dissertation**: LegalRAG: an LLM based RAG system to provide Legal literacy and support

(Signature of Student) (Signature of Supervisor)

Date: 4/12/2024 Date: 4/12/2024

## 1. Broad Area of Work

The goal of this project is to build a system or a product called LegalRAG. LegalRAG's goal is to utilize the power of LLMs and RAG systems to provide legal literacy and legal support to the common folks of India.

For the purpose of this project, legal acts and data related to inheritance, divorce, consumer protection and copyright laws in India will be used. No other legal acts and areas will be used.

This system will utilize concepts learned in Natural Language Processing, Conversational AI, Information Retrieval, and Deep Learning.

**Purpose:**

The Indian Legal system is very elaborate and can be very difficult for common folks to understand. This project aims at building a system that would make it easier for the general public to access legal literacy and basic legal support using a RAG system.

**Expected outcome of the work:**

A system called LegalRAG that would take in textual user query related to legal systems of India and get a relevant response along with any additional resources and sources cited to ensure the response is trustworthy.

**Literature Review:**

Literature review related to RAG methodologies and other LLM fine-tuning methods like LoRA are shared in the Literature References section.

Based on the review of these papers, it is noted that LLM fine-tuning methods like LoRA require more memory and resources to tune the additional layers. However, RAG doesn't mandatorily require fine-tuning an LLM, therefore it is not as resource intensive as fine-tuning an LLM. The availability of several ways of information retrieval from vector stores in RAG also makes it an optimal choice.

The references and blog articles that provide more information on the legal systems of India are also shared in the same section.

**Existing Processes and its limitations:**

A vanilla LLM or an LLM without fine-tuning can be used to answer Indian legal system related user queries. This is because many LLMs may have Indian legal system data in their training data. LLMs have a huge number of parameters and they contain tremendous knowledge through these parameters.

Some proprietary LLMs may have access to the internet and surf the internet to provide the relevant responses.

LLM fine-tuning methods like LoRA and Q-LoRA exist that can add additional layers to the LLM and fine-tuning these layers on Indian legal system data can also be used to provide relevant response to user queries.

All these methods will provide good results but they have some limitations, such as:

1. A vanilla LLM may hallucinate and provide responses that are not correct.
2. A vanilla LLM's response may not always reflect the latest changes that have been done to the Indian legal system as the training data is older.
3. Proprietary LLMs have a cost and still may hallucinate or provide responses without the links for sources.
4. In some cases, the data may be proprietary and may not be part of the training data and may not be present in the internet (which is not the case for Indian legal data, but it is true for different scenarios)
5. LLM fine-tuning will require additional memory and computational resources for the fine-tuning of these newly added layers to the LLM.

**Justification for the chosen methodology:**

LegalRAG consists of three components:

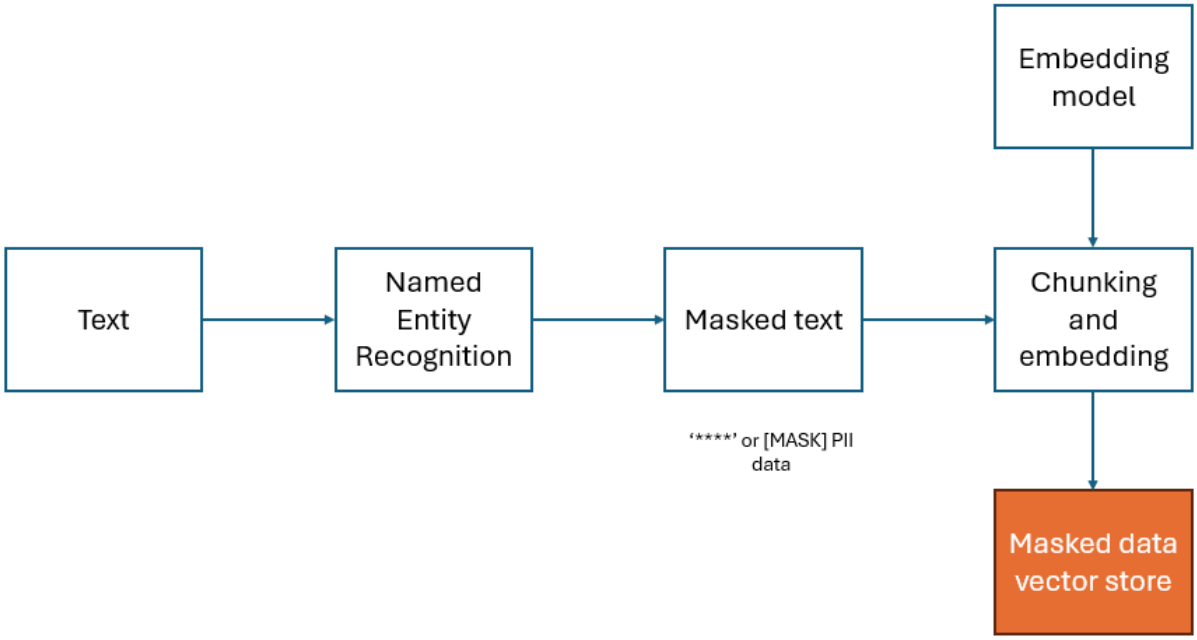
1. Masking Personally Identifiable data using Named Entity Recognition - This step is necessary as the legal acts data and the data related to cases could contain names of the individuals, their age and many other personally identifiable Information (PII) that needs to be masked to protect the individual. A named entity recognition model would identify all the PII data and replace them with '***' or [MASK] before storing the data in the vector store. This is an important step to protect the information of individuals involved in cases.

2. Intent Classification - This step is needed because the goal of the project is to provide relevant responses for four different legal acts. It is possible to have all these legal acts stored in the same location, but storing them in different locations and using an intent classification model to route the user query to the relevant legal act storage location will improve the response accuracy of the LLM.

3. RAG system
    a. RAG is an effective method to include non-parametric domain knowledge to an LLM's response without fine-tuning the LLM. Therefore it saves the memory and computational cost that comes from fine-tuning.
    b. RAG system also provides sources from where it got the context for a particular user query, this is a simpler way to ensure the LLM is not hallucinating.
    c. The RAG architecture makes it easy to update the domain knowledge frequently to reflect the updated Indian legal data as it requires a simple vector store update instead of re-fine tuning the LLM.

**Project methodology:**

The below architecture shows a simple methodology that will be followed to store the data. Minor changes could be done to the architecture as the project progresses but overall structure would remain the same.




The above architecture shows that the textual data from the different legal acts will be passed through a Named Entity Recognition (NER) model that would identify all the names, ages, aadhaar number, phone number and other PII data and mask them with '****' or [MASK] so that individual data is protected. The Masked data is chunked, and embedded using the embedding model and indexed as a vector store.

The architecture diagram will repeat four times as there are 4 legal acts considered for this project. However, the same NER and embedding model will be used for all 4 acts. Chunking strategy could differ from one legal act to another depending on the size of the data.

Here are the PII data that will be identified by the NER model and masked for Inheritance act and cases related to inheritance:

- Names of the deceased person and heirs
- Date of birth and age of individuals
- Family relationship and names (example: spouse, children, siblings)
- Aadhaar number, PAN
- contact numbers
- valuation of the inheritance
- address of the inheritance location and will information

PII data related to divorce act and cases that will be identified and masked with the help of NER model:

- Names of both spouses

- date of birth and age of individuals
- date and place of marriage, marriage certificate information
- names and ages of children, dependents, guardians involved
- residential address and phone number
- aadhaar number and PAN

PII data related to consumer protection and cases that will be identified and masked with the help of NER model:

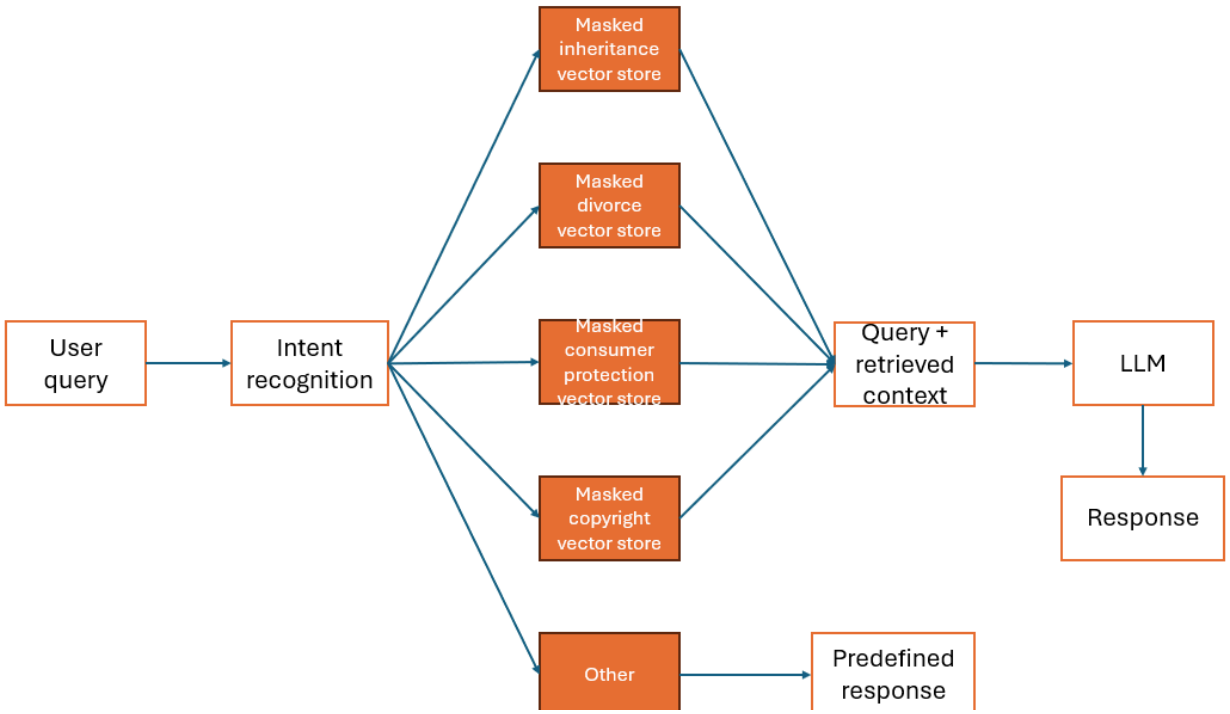
- Name of the complainant (consumer) and the respondent (business)
- Age and date of birth
- address and phone number, email addresses
- credit card or debit card information
- dates of transaction

PII data related to copyright and cases that will be identified and masked with the help of NER model:

- Name of the copyright owner
- date of birth and age
- aadhaar, PAN and phone number
- Copyrighted work case and docket numbers
- Contracts, agreements and license numbers

Note that the Embedding model shown here is part of the RAG architecture. The same embedding model will be used to embed the masked text and index it into the vector store as well as convert the user query into an embedding to search through the vector store and retrieve the context.

Below architecture shows the main project workflow from the user query to the relevant response:




User query is taken into an intent classification or intent recognition model. The intent will be one of the five: inheritance, consumer protection, divorce, copyright or Other. Based on the user intent, the query is routed to the relevant vector store. If the intent is 'Other' then there is no data for the LLM to answer that question, so a predefined response such as 'Please ask questions related to inheritance, divorce, copyright or consumer protection' will be shared with the user.

The data related to the above mentioned four acts will pass through NER to identify PII data and mask them. The masked data would be chunked, embedded and indexed in the relevant vector store.

The user query that is routed to the relevant vector store based on the user intent will be embedded (using the same embedding model as that of the masked text) and searches the masked data vector store to get the relevant context.

The relevant context along with the user query is sent to the LLM to generate the response to the user query along with the source information to ensure trust with the provided response.

**Novelty in the method:**

The use of an intent recognition model before the RAG systems is the novelty in this method. Intent recognition is very helpful because as more legal acts get added, only the intent recognition model will need to be re-trained or fine-tuned to involve more classes or intents without making any updates to the LLM.

This is useful as the intent recognition model is much smaller than LLM and therefore will consume less memory and resources to re-train.

This approach makes it easier to add more legal acts and make the LegalRAG system even more comprehensive as time passes without much updates to the LLM. .

**Benefits:**

LegalRAG will benefit the common folks to better understand the legal acts of India. It will also provide basic legal support such as filing cases, templates to fill and steps to follow for any issues they are facing under any of the four acts mentioned above.

## 2. Objectives

The objectives of my project are as follows:

- To provide legal literacy to the general public of India.
- To provide basic legal support for filing cases to the common folks of India.
- To provide source verification and resources for all the responses the system provides and enhance user trust.
- To ensure consumer data is protected and masked.

## 3. **Scope of Work**

Scope of this dissertation is to design and develop a product called LegalRAG that includes Named Entity Recognition model, intent classification model and a RAG based system. This product/system will identify user intent from the textual query and route the user query to different RAG systems that will include data stored in vector stores related to one of the following: acts of inheritance, divorce, consumer protection, or copyright in India. These data stored in a vector store will have any PII data masked with the help of a Named Entity Recognition model. The retrieved context from the vector stores along with the user query will be sent to an LLM to generate a relevant response to the user's query.

## 4. **Detailed Plan of Work (Sample)** (for 16 weeks)

| Serial Number of Task/Phases | **Tasks or subtasks to be done** (be precise and specific) | **Start Date-End Date** | Planned duration in weeks | Specific Deliverable in terms of the project |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| 1 | Initial part:<br>1. Finalizing the dissertation idea<br>2. Preliminary literature review<br>3. Abstract or outline preparation | 25/11/2024 - 08/12/2024 | 2 weeks | Submit the abstract in Viva portal |
| 2 | Data collection, Named Entity Recognition and masking:<br>1. Collecting the data related to four acts and preparing a document for each of them<br>2. Identifying the PII data from acts and cases using a Named Entity Recognition model<br>3. Masking the PII data<br>4. Choice of the right system for coding (LangChain or LlamaIndex or any other suitable one)<br>5. Experiment with different chunking strategies<br>6. Embed and store the masked PII data in vector store | 09/12/2024 - 22/12/2024 | 2 weeks | No deliverables in the Viva portal |
| 3 | Building Intent Recognition model:<br>1. Create a sample dataset with various types of user queries that may come based on the collected data. Keep high diversity in the queries present in the data<br>2. Experimenting with multiple classification models to get the best working intent classification<br>3. Choice of keeping an 'Other' intent in case the user query is unrelated to the four acts mentioned above. | 23/12/2024 - 05/01/2024 | 2 weeks | No deliverables in the Viva portal |
| 4 | Building RAG<br>1. Create initial RAG systems for all four of the acts<br>2. Provide some basic evaluation on the LLM responses for each of the four legal acts<br>3. Create a ppt for the midsem evaluation and demo of intent classification model and the RAG systems (integration of both systems are not yet done) | 06/01/2024 - 19/01/2024 | 2 weeks | Mid semester work submission in the Viva portal |

| | | | | |
|---|---|---|---|---|
| 5 | Integrating the two systems (RAG and intent recognition)<br>1. Combining the two systems and experiment with various parts of it to see it works properly<br>2. Midsem ppt and demo | 20/01/2024 - 02/02/2024 | 2 weeks | Mid semester evaluation results in the Viva portal |
| 6 | Evaluation system<br>1. Creating an evaluation dataset<br>2. Building evaluation system for the intent classification and the RAG system and the LLM response.<br>a. This will include checks for hallucination<br>b. precision and recall of responses on the evaluation dataset<br>c. to check the relevance of the retrieved context<br><br>Make any changes based on the feedback from midsem evaluation | 03/02/2024 - 16/02/2024 | 2 weeks | No deliverables in the Viva portal |
| 7 | Improve the evaluation results<br>1. Try with different LLMs to improve the quality of the results<br>2. Improve the retrieval mechanism by experimenting with different techniques like HyDE and Query fusion, neural re-reranker etc.<br>3. Try various prompts to keep the response concise<br>4. Prepare the ppt and demo<br>5. Submit the end sem evaluation documents | 17/02/2024 - 02/03/2024 | 2 weeks | Submit the end sem evaluation documents in the viva portal |
| 8 | End sem demo and ppt presentation | 03/03/2024 - 19/03/2024 | 2 weeks 3 days | End sem evaluation results in the Viva portal |

## 5. Literature References

The following are referred journals from the preliminary literature review.

*[1] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks",*

*arXiv:2005.11401, URL https://arxiv.org/abs/2005.11401*

*[2] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan et al., "Retrieval-Augmented Generation for Large Language Models: A Survey", arXiv:2312.10997, URL https://arxiv.org/abs/2312.10997*

*[3] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, "LoRA: Low-Rank Adaptation of Large Language Models", arXiv:2106.09685, URL https://arxiv.org/abs/2106.09685*

*[4] Department of Justice, URL https://doj.gov.in/*

*[5] iPLeaders Blog (Powered by LawSikho), URL https://blog.ipleaders.in/*

*[6] Legislative Department of India, URL https://legislative.gov.in/*

**Supervisor's Rating of the Technical Quality of this Dissertation Outline**

EXCELLENT / GOOD / FAIR/ POOR (Please specify): _____

**Supervisor's suggestions and remarks about the outline (if applicable).**

Date 4/12/2024 _____

(Signature of Supervisor)

Name of the supervisor: Naveen Rathani

Email Id of Supervisor: naveen.rathani@gmail.com

Mob # of supervisor: +91 80950 00395