**🧠 Contract Sage – Intelligent Legal Document Assistant**

**🗂️ Core Components**

**🔹 LLM for Summarization**

• Model: **Mistral or LLaMA (multilingual finetuned models)**

• Technique: **Prompt Engineering**

• Task: Extract and summarize document purpose, entity relationships, and risks.

**🔹 Named Entity Recognition (NER)**

• Model: **Fine-tuned XLM-R** (multilingual support)

• Task: Extract parties, dates, monetary amounts, obligations, jurisdictions, etc.

**🔹 Anomaly Detection**

• Model: **LLM-based** , prompt engineering to identify language fraud

• Task: Identify unusual or risky legal clauses using both:

• Prompt-based scoring (1–5 risk scale)

• Clause similarity via embedding comparison

**🧠 LLM Prompting Strategy**

Once NER and anomaly detection outputs are ready, they are passed into a summarization prompt like this:

ENTITIES OF INTEREST:

[NER output with entity types and confidence scores]

POTENTIAL ANOMALIES:

[Anomaly detection output with clause references and confidence scores]

SUMMARIZATION INSTRUCTIONS:

- Provide a general overview of document purpose

- Explain the following flagged clauses: [list]

- Clarify relationships between these specific entities: [list]

**🚀 Advanced Enhancements (Optimized for Hackathon)**

**✅ 1. Feedback Loop Integration**

• The LLM dynamically focuses on outputs from other models (e.g., explain flagged clauses or highlight risky relationships).

• Simple routing logic or chained prompts make this work.

• 🔧 No UI required – works well with prompt engineering.

**✅ 2. Explanation Generation**

• For every suspicious clause or anomaly, the LLM generates:

• Plain-English summaries

• Legal implications

• Risk assessments with reasoning

• **Prompt Example**:

*“You’re a legal analyst. Explain this clause in simple terms and rate its risk from 1–5.”*

**✅ 3. Comparative Analysis**

• Use BERT/LLM-based embeddings to compare contract clauses to a vector store of **standard “safe” clauses**.

• Flag low similarity clauses as potentially suspicious.

• 🔹 Fast and scalable using:

• sentence-transformers

• cosine\_similarity

• 🔧 No need for FAISS or full DB – in-memory comparison works for demo.

**⚙️ System Architecture Overview**

1. **Document Input (PDF/Text)**

2. ➤ **NER Model** → Extract key entities

3. ➤ **Anomaly Detection Model** → Flag unusual clauses

4. ➤ **LLM Summarization** → Uses both outputs and the main doc to:

• Summarize document

• Focus on flagged parts

• Provide risk analysis and plain-language output

5. ➤ **Comparative Clause Check** → Embedding similarity vs. standard clause base

**🧪 Evaluation Metrics**

• **Precision/Recall** of NER entities (manually evaluated on legal docs)

• **Clause Risk Detection Accuracy** – Compare flagged vs. ground truth

• **LLM Summarization Quality** – Human-in-the-loop rating

**📦 Future Enhancements (Post-Hackathon)**

• Interactive UI for legal experts to validate output

• Document-type classification for fine-tuned processing

• Layout-aware multimodal parsing for scanned legal contracts