Code Four

Take-Home Assignment: Wisconsin Law Enforcement Legal Chat RAG System

Overview

Build a proof-of-concept Retrieval-Augmented Generation (RAG) system that enables Wisconsin law enforcement officers to quickly query state statutes, case law, and department policies through a conversational interface.

Time Estimate: 4-6 hours
Submission Deadline: 72 hours from receipt

Background

Wisconsin law enforcement officers need rapid access to accurate legal information during their duties. This includes state statutes, recent case law, department policies, and constitutional guidelines. Your task is to create a RAG system that can provide contextually relevant, accurate legal information through a chat interface.

Core Requirements

Part 1: Document Processing & Indexing (40%)

1. **Data Ingestion Pipeline**
    - Design a system to ingest and process multiple document types:
        - Wisconsin State Statutes (PDF/HTML)
        - Case law summaries (PDF)
        - Department policies (DOCX/PDF)
        - Training materials (various formats)
2. **Document Chunking Strategy**
    - Implement intelligent chunking that preserves legal context

   - Handle hierarchical document structures (chapters, sections, subsections)
   - Maintain metadata (statute numbers, case citations, dates)
3. **Vector Database Setup**

   - Choose and implement a vector database solution (Pinecone, Weaviate, Chroma, etc.)
   - Create embeddings optimized for legal text
   - Design an efficient indexing schema with metadata filtering

Part 2: Retrieval System (30%)

1. **Hybrid Search Implementation**

   - Combine semantic search with keyword matching for statute numbers/case names
   - Implement relevance scoring that prioritizes:
     - Current vs. superseded laws
     - State vs. federal jurisdiction
     - Department-specific vs. general policies
2. **Context Window Management**

   - Design a system to handle related statutes and cross-references
   - Implement citation chain following (e.g., "see also § 940.01")
   - Manage context limits while preserving legal completeness
3. **Query Enhancement**

   - Build query expansion for legal synonyms and related terms
   - Handle common law enforcement abbreviations and terminology
   - Implement spell correction for legal terms

Part 3: Generation & Response (20%)

1. **Response Generation**

   - Integrate with an LLM (OpenAI, Anthropic, or open-source)
   - Design prompts that ensure legally accurate responses
   - Include source citations in all responses
2. **Safety & Accuracy Features**

   - Implement confidence scoring for responses
   - Flag when information may be outdated or jurisdiction-specific

    - Add disclaimers for legal advice vs. information
    - Handle queries about use of force with appropriate care

Part 4: User Interface (10%)

1. **Chat Interface**

    - Build a simple web interface for the chat system
    - Display source documents with highlighting
    - Show confidence levels and metadata
    - Include conversation history
2. **Officer-Specific Features**

    - Quick access buttons for common queries (Miranda rights, traffic stops, etc.)
    - Export functionality for report writing
    - Mobile-responsive design for field use

Technical Specifications

Required Technologies

- **Language:** Python or TypeScript/JavaScript
- **Vector Database:** Your choice (document your selection reasoning)
- **LLM:** OpenAI API, Anthropic API, or open-source model
- **Framework:** FastAPI/Flask (Python) or Express/Next.js (JS/TS)

Sample Data

You should use publicly available Wisconsin legal documents to expand your dataset

UI Style Guide

Your interface should follow our design system for consistency:

Color Palette:

- **Primary Accent:** #005cde (Blue - for CTAs, links, focus states)
- **Secondary Accent:** #2563eb (Lighter blue - for hover states)
- **Success:** #00B96B / #4ade80
- **Warning:** #F4801A / #eab308
- **Error:** #ef4444 / #ea5756

Dark Theme (Default):

- **Background Primary:** #121212
- **Background Secondary:** #0a0a0a
- **Card Background:** #0a0a0a
- **Text Primary:** #ffffff
- **Text Secondary:** #d1d5db
- **Border Light:** #222222
- **Border Medium:** #333333

Light Theme (Optional):

- **Background Primary:** #ffffff
- **Background Secondary:** #f3f4f6
- **Card Background:** #ffffff
- **Text Primary:** #111827
- **Text Secondary:** #374151
- **Border Light:** #e5e7eb

Typography:

- **Font Family:** var(--font-mono) for code/technical content
- **Font Family:** var(--font-sans) for UI text
- **Headers:** Uppercase, 600 weight for section headers
- **Body:** 14-16px, regular weight

Component Patterns:

- **Cards:** 12px border radius, 1px border, subtle shadow on hover
- **Buttons:** 4px border radius, transition all 0.3s ease
- **Inputs:** 4px border radius, focus ring with primary accent
- **Status Badges:** Full rounded (9999px), uppercase text, 0.75rem
- **Loading States:** Skeleton with shimmer animation
- **Modals:** Glass morphism effect with backdrop blur

Animations:

- **Fade In:** 0.5s ease-out for content appearance
- **Hover States:** translateY(-4px) for cards, scale(1.02) for buttons
- **Transitions:** 0.2-0.3s for all interactive elements

Deliverables

1. **Code Repository**
    - Well-documented, production-ready code
    - Clear README with setup instructions
    - Environment configuration files
    - Unit tests for critical components
2. **System Architecture Document**
    - Diagram of the RAG pipeline
    - Explanation of design decisions
    - Scalability considerations
    - Security and privacy measures
3. **Demo Video (5-10 minutes)**
    - Walk through the system architecture
    - Demonstrate 3-5 real queries showing:
        - Statute lookup
        - Case law retrieval
        - Policy clarification
        - Cross-reference handling
4. **Performance Metrics**
    - Retrieval accuracy metrics
    - Response time benchmarks
    - Relevance scoring evaluation
    - Test results on provided query set

Evaluation Criteria

Technical Implementation (40%)

- Code quality and organization
- Appropriate use of RAG architecture
- Efficiency of retrieval system
- Proper handling of legal document structures

Legal Domain Adaptation (30%)

- Accuracy of legal information retrieval
- Appropriate handling of citations and references

- Understanding of law enforcement use cases
- Safety measures for sensitive queries

System Design (20%)

- Scalability considerations
- Security and privacy measures
- Error handling and edge cases
- Documentation quality

Extra Brownie Points (10%)

- Creative solutions to legal-specific challenges
- Additional features that add value
- Performance optimizations
- UI/UX considerations for officers

Bonus Challenges (Optional)

1. **Multi-Jurisdiction Support**
    - Extend system to handle federal law and neighboring state statutes
    - Implement jurisdiction-aware routing
2. **Temporal Awareness**
    - Handle legal changes over time
    - Show when statutes were amended
    - Flag recently changed laws
3. **Advanced Analytics**
    - Track common queries for training needs
    - Identify knowledge gaps in the corpus
    - Generate summaries of new case law
4. **Integration Features**
    - Mock integration with CAD/RMS systems
    - Report generation assistance
    - Citation formatting for court documents

Example Queries to Support

Your system should handle queries like:

- "What are the elements required for OWI 3rd offense in Wisconsin?"
- "Can I search a vehicle during a traffic stop without consent?"
- "What's the statute of limitations for misdemeanor theft?"
- "Show me recent cases about Terry stops in Wisconsin"
- "What's our department policy on pursuit driving?"
- "What Miranda warnings are required for juveniles?"

Submission Instructions

1. Push your code to a GitHub repository (public or provide access)
2. Include all documentation in the repository
3. Share the demo video via YouTube (unlisted) or Loom
4. Email the repository link and video URL to: [founders@codefour.us]

Potentially Helpful (or Unhelpful) Resources

- Wisconsin State Legislature Statutes
- Wisconsin Court System - Recent Opinions
- LangChain Documentation
- OpenAI Embeddings Guide

Good luck! We're excited to see your approach to solving this real-world challenge in law enforcement technology.