SATIM challenge Documentation

Link to the project demonstration video:

check out this link for a quick project demonstration : https://drive.google.com/file/d/1a1ovWjJq3R4PfgPZvaQUC96aH-INcw9V/view?usp=sharing

Project Overview

The Compliance Analysis System is a web-based application designed to help organizations assess and maintain compliance with their internal policies and external standards. The system provides intelligent analysis of business use cases and policy documents through natural language processing and machine learning techniques.

The application offers two core functionalities:

1. **Compliance Analyzer** - Evaluates business use cases against organizational policies
2. **Compliance Checker** - Compares policy documents against standard templates or regulations and identifies mismatches

System Architecture

The system is built as a Flask-based REST API backend that processes policy documents and provides compliance analysis through AI-powered semantic matching and analysis.

Core Components

- **Flask Web Framework** - Handles HTTP requests and API endpoints

- **Sentence Transformers** - Provides semantic text embeddings for document similarity
- **FAISS Vector Database** - Enables fast similarity search across policy documents
- **Google Gemini AI** - Performs intelligent compliance analysis and generates recommendations
- **Document Processing** - Supports DOCX and PDF file formats

Functionality 1: Compliance Analyzer

Purpose

The Compliance Analyzer evaluates specific business use cases against an organization's policy framework to determine compliance levels and provide actionable recommendations.

Implementation Flow

1. **Policy Initialization**
    - Organizations upload their policy documents (DOCX format)
    - Documents are processed and split into manageable chunks (1000 characters with 200-character overlap)
    - Each chunk is converted to vector embeddings using the MiniLM sentence transformer model
    - Embeddings are stored in a FAISS vector database for efficient retrieval
2. **Use Case Analysis**
    - Users submit a business use case description through the web interface
    - The system performs semantic search to find the most relevant policy chunks
    - Top 5 most similar policy sections are retrieved based on vector similarity
3. **AI-Powered Compliance Assessment**
    - Google Gemini AI analyzes the use case against relevant policy excerpts

- Generates individual KPI scores (0-1) for each applicable policy
- Identifies specific areas where the use case may be non-compliant
- Provides actionable recommendations for achieving compliance

4. **Results Presentation**
    - Compliance scores are categorized as: Compliant (≥0.85), Partially Compliant (0.5-0.84), or Not Compliant (<0.5)
    - Detailed breakdown shows which policies are affected and how
    - Specific recommendations are provided to address compliance gaps

Technical Implementation

- **Endpoint**: `POST /analyze`
- **Vector Search**: Uses cosine similarity to find relevant policy sections
- **Chunking Strategy**: Overlapping chunks ensure context preservation across document boundaries
- **AI Integration**: Structured prompts guide Gemini to provide consistent, actionable analysis

Functionality 2: Compliance Checker

Purpose

The Compliance Checker compares an organization's policy document against one or more standard templates or regulatory requirements to identify gaps and ensure comprehensive coverage.

Implementation Flow

1. Document Upload

- Users upload their organization's policy document (PDF or DOCX)
- Multiple standard/template documents can be uploaded for comparison
- Documents are processed to extract meaningful clauses and sections

2. Text Processing Pipeline:

- Extracts text from uploaded policy files (PDF/DOCX)
- Text is broken into meaningful "clauses" (sentences/paragraphs over 50 characters)
- Irrelevant content is filtered out using predefined anchor topics like "data privacy", "safety procedures", etc.

3. Similarity Matching Using Transformers:

- We have used SentenceTransformer to convert text clauses into numerical vectors (embeddings)
- We compare policy clauses against standard/reference clauses using cosine similarity
- Setting a threshold (default 0.75) , if similarity score is above this, clauses are considered "matching"

4. Compliance Analysis:

- **Forward matching**: Finds which user policy clauses match standard requirements
- **Coverage checking**: Identifies which standard requirements are missing from the user's policy
- **Compliance score**: Percentage of standard requirements that are covered

Technical Implementation

- **Endpoint**: `POST /check_compliance`
- **Model**: Uses paraphrase-multilingual-MiniLM-L12-v2 for cross-language compatibility
- **Filtering**: Semantic filtering reduces noise and focuses on compliance-relevant content
- **Threshold Configuration**: Adjustable similarity threshold allows fine-tuning of matching sensitivity

API Endpoints

`/initialize` (POST)

Uploads and processes policy documents for the Compliance Analyzer functionality.

`/analyze` (POST)

Analyzes a business use case against loaded policies and returns compliance assessment.

`/check_compliance` (POST)

Compares a policy document against standard templates and returns coverage analysis.

`/health` (GET)

Returns system status and initialization state.

Key Features

- **Intelligent Document Processing**: Handles multiple document formats with smart text extraction
- **Semantic Understanding**: Uses advanced NLP models to understand meaning beyond keyword matching
- **Scalable Vector Search**: FAISS integration enables fast similarity search across large policy collections
- **AI-Powered Analysis**: Gemini AI provides human-like compliance reasoning and recommendations
- **Flexible Thresholds**: Configurable similarity thresholds allow customization for different use cases
- **Comprehensive Coverage**: Bidirectional comparison ensures both compliance and completeness