BackendRAG is an advanced backend server application that leverages the power of AI through various libraries including LangChain, Pytesseract, Sentence Transformers, and Flask to automate document processing and validation tasks. It integrates multiple functionalities for document extraction, text splitting, embedding, and structured data storage using Chroma.

Features

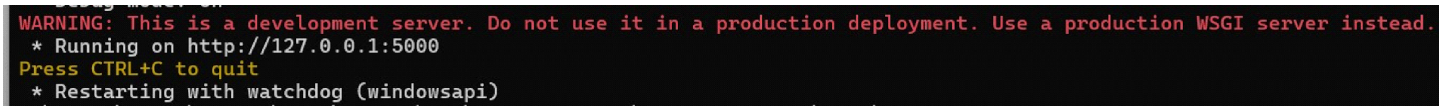
- Document Processing: Supports various formats like PDF, DOCX and txt files
- Text Extraction and Splitting: Utilizes custom text splitters for handling large texts.
- Embeddings and Validation: Employs Sentence Transformers for generating embeddings and cosine similarity for document validation.
- Creation of a vector database using CHROMA DB for storing the structures of documents extracted from reference files submitted by users
- Flask API: A simple API to handle document validation requests dynamically.
- Dynamic Category Structuring: Automatically processes reference documents to create and store structured data
- Session based memory management: Manages document validation results and user interactions in memory on a per-session basis, enabling seamless tracking of session- specific data.

Installation and Test

Steps:

•pip install -r requirements.txt

•python backendrag.py



•streamlit run frontendrag.py

•Ensure the environment variables are set as required:

`GROQ_API_KEY` for LangChain integration.

`HF_API_KEY` for using Hugging Face models.

Example:

Document Validator

Select the category of your file

ASSURANCE de l ecole

Upload the file to validate

Drag and drop file here

Limit 200MB per file • PDF, DOCX, TXT, JPG, JPEG, PNG, BMP, TIFF, TIF

Browse files

3.pdf 341.5KB

Validate Document

Validation Report: **Rapport de validation**

1. Appartient-il à la catégorie 'ASSURANCE de l'école' ?

Non, le document ne appartient pas à la catégorie 'ASSURANCE de l'école'. Le document est un relevé de notes et résultats d'un étudiant, ce qui le classe plutôt dans la catégorie 'ÉTUDIANT' ou 'FORMATION'.

2. Respecte-t-il la structure prédéfinie de cette catégorie ?

Non, le document ne respecte pas la structure prédéfinie de la catégorie 'ASSURANCE de l'école'. La structure attendue pour cette catégorie est généralement la suivante :

- Titre de l'assurance
- Description de l'assurance
- État de l'assurance (par exemple, "en cours", "terminée", "annulée")
- Informations sur les parties prenantes (par exemple, les parties contractantes, les garanties)

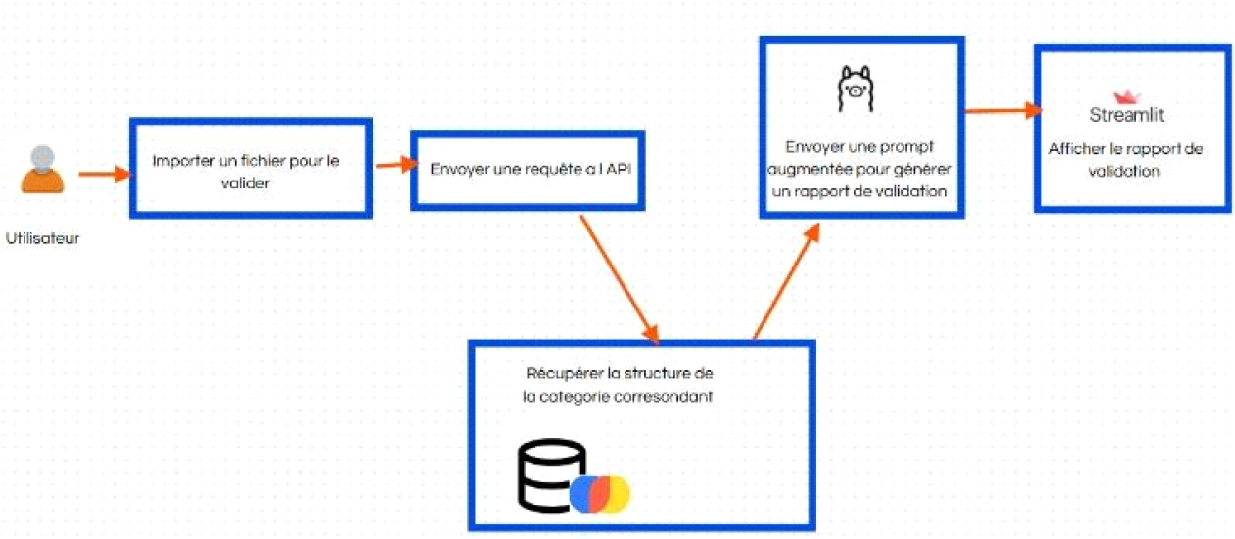
Le document fourni est un relevé de notes et résultats d'un étudiant, qui ne correspond pas à la structure attendue pour une assurance.

En conséquence, le document ne peut pas être considéré comme valide pour la catégorie 'ASSURANCE de l'école'.

Verification: Not Valid

The uploaded document is not valid.

API Architecture



- File Upload: The process begins when the user uploads a file to be validated through the user interface.
- API Request: A request is made to the API. This request sends the file path and the specified category to the API.
- Retrieval of Category Data: The API queries the vector database to extract the appropriate category structure associated with the file's characteristics.
- Validation Report Generation: With the determined category, an enriched request is sent to the Llama model to generate a validation report.
- Presentation and Storage of Results*: The validation report is then presented to the user via a Streamlit interface