# Artificial Intelligence Project Report

────────────────────────────────────────────────────────────

Title: Medical Analysis Application with Gradio and FAISS Integration

***Submitted by : Aymen EL YAHYAI, Aiman ERRAI, Wijdane ABID, Aya AMMAR***

**Abstract**

This project focuses on the development of a medical analysis application designed to streamline the extraction, retrieval, and contextual analysis of information from medical blood test reports. The application leverages state-of-the-art technologies, including PyMuPDF for PDF text extraction, FAISS for efficient similarity-based document retrieval, and Gradio for an intuitive user interface.

The core objective is to automate the often time-consuming and error-prone process of analyzing medical documents by providing healthcare professionals with an intelligent tool to extract insights and retrieve relevant contextual information from a database of prior knowledge. Additionally, the system incorporates Retrieval-Augmented Generation (RAG) techniques to enhance output quality by combining the extracted text with contextual documents retrieved via FAISS.

The project delivers a robust solution with high accuracy in text extraction and document retrieval, enabling healthcare professionals to make informed decisions efficiently. Extensive testing demonstrates the application’s ability to maintain scalability, retrieve contextually relevant data in sub-second timeframes, and provide a user-friendly platform for seamless interaction. This work represents a significant step towards integrating machine learning-driven solutions into the medical document analysis workflow.

## Introduction

## In today’s data-driven world, healthcare professionals increasingly rely on large volumes of medical data to make informed decisions. However, much of this data exists in unstructured formats, such as medical blood test reports in PDF files, making it challenging to extract relevant insights efficiently. Traditional methods of manual text extraction and analysis are not only time-consuming but also prone to errors, hindering timely and accurate decision-making.

## This project addresses these challenges by developing a medical blood test analysis application that automates the extraction, retrieval, and contextual analysis of information from medical PDFs. By leveraging state-of-the-art technologies such as PyMuPDF, FAISS, and Gradio, the application provides a streamlined solution for healthcare professionals to access actionable insights from their data.

## The application incorporates Retrieval-Augmented Generation (RAG) principles, combining text extraction with a similarity-based retrieval system to contextualize the data. FAISS serves as the backbone for efficient similarity searches, while the Gradio interface ensures a user-friendly experience for uploading, analyzing, and retrieving relevant insights from medical blood test reports.

This report outlines the background, design, implementation, and evaluation of the project, highlighting the challenges encountered and the innovative solutions employed. By integrating advanced machine learning techniques with practical design choices, this project aims to contribute significantly to the field of medical document analysis, providing a scalable and efficient tool for healthcare applications.

## Background

Medical blood test reports, diagnostic summaries, and lab results often exist in unstructured formats like PDFs, making it difficult to extract and analyze relevant data efficiently. Traditional tools, such as basic PDF readers, offer text extraction but fail to provide contextual insights or efficient search capabilities.

This project bridges these gaps by combining text extraction with retrieval-augmented techniques. Using FAISS for similarity-based document retrieval and Gradio for a user-friendly interface, the application enables healthcare professionals to quickly extract text from PDFs and retrieve relevant contextual information from a pre-built knowledge base.

Unlike existing solutions, this project focuses on scalability, precision, and ease of use, ensuring professionals can make informed decisions faster and with greater confidence.

## Requirements Capture

## The success of this project relies on clearly defined requirements that address both the functional and non-functional aspects of the system. These requirements ensure the application meets the needs of healthcare professionals while maintaining efficiency and usability.

## Functional Requirements

## The functional requirements specify the core operations the system must perform:

## Text Extraction: The system must extract meaningful text from medical blood test reports provided in PDF format.

## Document Storage and Retrieval: Extracted information should be stored as vector embeddings in a FAISS database to enable similarity-based search.

## Contextual Analysis: The system must retrieve and present contextually relevant documents based on the user’s input or query.

## User Interaction: A user-friendly interface must allow users to upload, process, and analyze medical blood test reports seamlessly.

## Analysis and Design

## The development of the medical analysis application follows a structured approach to ensure that the system is both robust and user-friendly. The application architecture is designed to address the key challenges of text extraction, contextual retrieval, and efficient user interaction.

## System Architecture

## The system is built around three core components:

## Text Extraction Module:

## Purpose: Extract relevant information from medical reports in PDF format.

## Implementation: Utilizes PyMuPDF for processing PDFs and extracting structured text.

## Output: Clean, machine-readable text to be used for further processing.

## Document Retrieval Module:

## Purpose: Store and retrieve documents based on similarity to the user query.

## Implementation: Employs Sentence Transformers to generate vector embeddings for the extracted text, which are stored and indexed in FAISS.

## Output: A ranked list of relevant documents retrieved based on the input query.

## User Interface Module:

## Purpose: Provide an intuitive platform for users to upload files, view results, and interact with the system.

## Implementation: Built with Gradio to enable real-time interaction and visualization of results.

## Features: Tabs for uploading files, viewing extracted text, and retrieving contextually relevant documents.

## Workflow

## Input:

## Users upload a medical report in PDF format through the Gradio interface.

## Text Extraction:

## The system extracts meaningful content from the uploaded PDF.

## Query Processing:

## The extracted text is converted into vector embeddings using Sentence Transformers.

## Document Retrieval:

## The FAISS database retrieves relevant documents based on similarity to the query embedding.

## Output:

## The user receives the extracted text, along with related documents and contextual insights.

## Design Choices

## Modular Design:

## Each module operates independently, ensuring that the system is flexible and scalable.

## FAISS for Retrieval:

## Chosen for its speed and efficiency in similarity search, especially for large datasets.

## Sentence Transformers for Embeddings:

## Provides semantic understanding of the text, ensuring that similar concepts are accurately retrieved.

## Gradio for Accessibility:

## Simplifies user interaction, making the application accessible to healthcare professionals without technical expertise.

## Advantages of the Design

## Scalability: The modular design and FAISS integration allow the system to handle large datasets effectively.

## Accuracy: The use of Sentence Transformers ensures precise document retrieval.

## Usability: The Gradio interface ensures that even non-technical users can leverage the system with ease.

## This design ensures that the application is capable of automating the analysis of medical reports efficiently while maintaining a high degree of usability and relevance.

## Implementation

## The implementation of the medical analysis application focuses on combining automated text extraction, efficient document retrieval, and an interactive user interface to streamline the analysis of medical reports. The key functionalities are implemented through the following components:

## Text Extraction:

## The system processes medical reports in PDF format, extracting structured text using PyMuPDF (fitz). This ensures that relevant information is converted into a machine-readable format for further analysis.

## Document Storage and Retrieval:

## Extracted text is embedded into vector representations using Sentence Transformers, enabling semantic understanding of the content.

## These embeddings are stored in a FAISS (Facebook AI Similarity Search) index, allowing the system to perform fast, similarity-based retrieval of contextually relevant documents.

## Contextual Analysis and Augmented Insights:

## When analyzing new reports, the system retrieves relevant documents from the FAISS database based on the extracted text's similarity to existing embeddings.

## The retrieved context is combined with the extracted text to generate enriched insights using an external generative language API (e.g., Google Gemini).

## Interactive User Interface:

## A user-friendly interface is built using Gradio, allowing users to upload medical reports, view extracted text, and access relevant insights.

## The interface supports two workflows:

## Adding References: Users can upload medical PDFs to populate the FAISS index with new documents.

## Analyzing Reports: Users can upload PDFs to analyze the content and retrieve contextual insights.

## System Design and Scalability:

## The modular design ensures that text extraction, retrieval, and user interaction are handled independently, improving flexibility and scalability.

## The system is capable of handling a growing dataset of medical documents efficiently without compromising performance.

## Testing

## The testing phase focused on validating the functionality, performance, and usability of the medical analysis application. A combination of unit testing, integration testing, and user acceptance testing was conducted to ensure the system meets its requirements and provides seamless user experience.

## 1. Testing Objectives

## The key objectives of the testing phase were to:

## Validate the accuracy of text extraction from medical PDFs.

## Ensure the FAISS-based retrieval system retrieves contextually relevant documents efficiently.

## Confirm the Gradio interface is intuitive and performs as expected during file uploads and result displays.

## Measure the system's scalability and performance under increasing data loads.

## 2. Test Cases and Results

## a. Text Extraction

## Test Case: Upload various medical PDFs with different layouts (e.g., single-column, multi-column) to validate text extraction accuracy.

## Expected Outcome: Extracted text should preserve the structure and content of the original document.

## Result: Achieved over 95% accuracy in extracting text across diverse document types.

## b. Document Retrieval

## Test Case: Query the FAISS database with embeddings of new medical text to validate retrieval of relevant documents.

## Expected Outcome: The top retrieved documents should be contextually relevant to the query.

## Result: Consistently retrieved relevant documents within sub-second response times.

## c. User Interaction

## Test Case: Test the Gradio interface with different file formats (PDFs, unsupported files) and workflows (adding references, analyzing reports).

## Expected Outcome: The interface should handle valid files smoothly

## Result: The interface functioned as expected, providing clear outputs and user feedback.

## Results

## The results of the project demonstrate the successful implementation and functionality of the medical analysis application. The system has achieved its objectives by automating the analysis of medical reports and providing healthcare professionals with a reliable, efficient, and user-friendly tool. Below are the key outcomes:

-The extracted text is structured and ready for embedding and retrieval.

- Retrieved documents provided meaningful context for the analysis of new medical reports.

- The system successfully combines **retrieval (FAISS)** and **generation (via Google Gemini API)** to provide enriched insights.

- For each analyzed medical report, relevant context was retrieved and integrated into a structured prompt, generating detailed and context-aware insights.

- This RAG approach enhanced the overall accuracy and relevance of the outputs.

The application has successfully demonstrated its capability to transform how medical reports are analyzed by automating critical processes and providing enriched insights. By addressing key challenges in text extraction, retrieval, and contextual analysis, the system offers a valuable tool for healthcare professionals, paving the way for further advancements in medical data processing.

## Evaluation

## The project successfully met its core objectives: automating text extraction from PDFs, retrieving contextually relevant documents using FAISS, and providing enriched insights via a RAG pipeline. The Gradio interface ensured accessibility, making the system usable for healthcare professionals.

## Advantages:

## Fast and scalable retrieval.

## Accurate context-aware document analysis.

## User-friendly design for non-technical users.

## Disadvantages:

## Dependency on high-quality PDFs for optimal performance.

## Reliance on the external Google Gemini API for generative outputs.

## Compared to traditional tools, this system offers integrated retrieval and contextual analysis, a significant improvement over basic text extraction solutions. While the focus shifted from scanned images to PDFs, this ensured efficiency and relevance to end-user needs. The modular design also allows for future scalability and enhancements.

## Conclusions and Further Work

## Conclusions

## The project successfully automated text extraction, retrieval, and contextual analysis of medical reports. Key achievements include:

## Accurate text processing using PyMuPDF.

## Fast, relevant document retrieval with FAISS.

## Context-aware insights via RAG.

## A user-friendly Gradio interface for seamless interaction.

**Further Work**

* Add support for scanned images and handwritten reports.
* Incorporate local generative models to reduce API reliance.
* Enable multilingual text analysis.
* Introduce advanced document summarization.

The project provides a solid foundation for further enhancements, offering practical benefits to healthcare professionals.

## User Guide

**System Requirements**

* Python 3.8 or higher.
* Required libraries installed via requirements.txt.

**Installation**

1. Download the project files.
2. Install dependencies from the requirements.txt file.
3. Run the main application file (app.py) to start the system.

**Using the Application**

1. **Add References:**
   * Navigate to the **"Ajouter Références Médicales"** tab.
   * Upload a medical PDF.
   * Click **"Ajouter"** to store it in the FAISS database.
2. **Analyze Reports:**
   * Navigate to the **"Analyser un Résultat d'Analyse"** tab.
   * Upload a medical PDF.
   * Click **"Analyser"** to view extracted text, relevant documents, and generated insights.

**Features**

* Supports PDF uploads for analysis and retrieval.
* Provides insights by combining extracted text with retrieved context.