

Multi-PDF Chatbot

Ramarao Kadiyala - Mentor

Reethu RG Thota

S R Monish Raj

Shreyas Rajiv

Sruthika Sivakumar

PROJECT INTRODUCTION

- This project is a comprehensive Python application that combines text extraction, document analysis, language translation, and email automation.
- The system is designed to facilitate efficient handling of multiple PDF documents, enabling users to upload files, extract text by posing targeted questions, and receive precise language translations along with email capabilities.
- This application is valuable for professionals, researchers, and students seeking to expedite document analysis, perform language translation, and sharing information within their respective domains.

PROJECT FEATURES

- **Efficient Document Understanding**

The application streamlines document comprehension through advanced text extraction, vector representations, and a responsive chatbot, enabling users to interactively explore and gain insights from complex PDF files.

- **Multilingual Access and Collaboration**

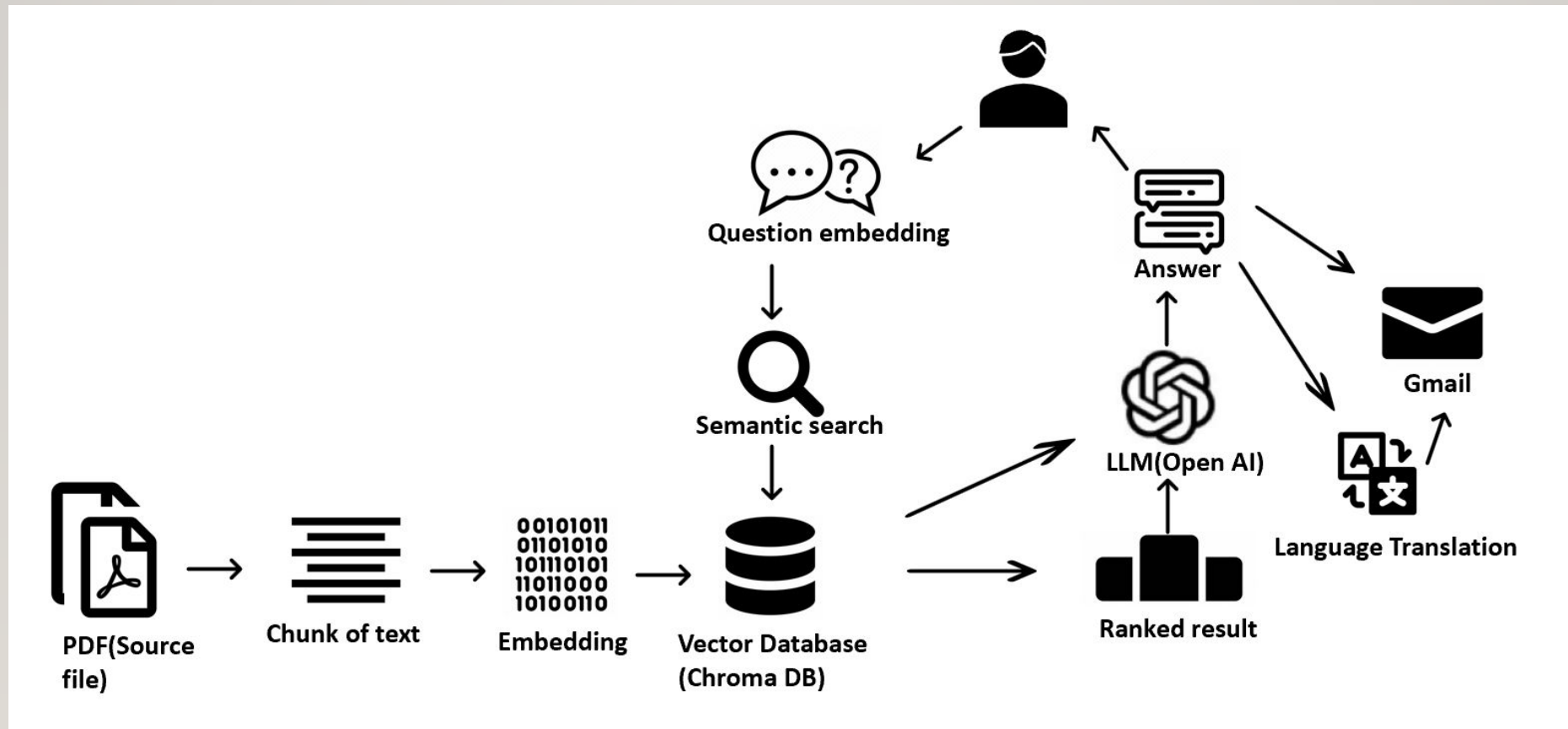
Overcoming language barriers, the tool facilitates precise language translations within the document, promoting collaboration among users with different language preferences and enhancing accessibility to a wider audience.

- **Workflow Optimization with Email Automation**

The integration of GMail API automates the email process, allowing users to seamlessly share valuable document excerpts or insights with colleagues. This feature optimizes communication workflows and enhances overall productivity.



PROJECT ARCHITECTURE



TOOLS AND TECHNOLOGIES USED

- Streamlit - Frontend user interface
- Langchain -
 - CharacterTextSplitter
 - OpenAIEmbeddings
 - Load_qa_chain
 - PromptTemplate
- OpenAI LLM - Used for Retrieval-Augmented Generation (RAG)
- Chroma DB - Vector Database
- GMail API - Integration and authentication with Google Cloud Platform (GCP)
- PyPDF2 - Retrieve text and metadata from PDF

DEMO AND CONCLUSION

Ask your PDF

Upload your PDF



Drag and drop files here

Limit 200MB per file • PDF

Browse files



Introduction to the Design and Analysis of Algorithms (3rd ed.) [Levitin 2011-10-0... 9.1MB X



CP.pdf 3.8MB X

Ask a question about your PDF:

what is dfs and bfs algorithm and give C code

DFS (depth-first search) and BFS (breadth-first search) are two common graph traversal algorithms used to search for a specific element or path in a graph data structure.

DFS uses a stack data structure to keep track of visited nodes and explore the graph in a depth-first manner, meaning it will go as deep as possible in a graph before backtracking and exploring other paths.

BFS, on the other hand, uses a queue data structure to keep track of visited nodes and explore the graph in a breadth-first manner, meaning it will explore all nodes at a given depth before moving on to the next depth level.

Here is an example of a DFS and BFS implementation in C:

```
// DFS function void DFS(int start, int visited[], int graph[][MAX], int n) { int i; visited[start] = 1; // mark starting node as visited printf("%d ", start); // print the starting node
```

```
    // recursively visit all adjacent nodes that have not been visited
    for(i = 0; i < n; i++) {
```

```
// Structure pour représenter un nœud dans le graphe struct Node { int value; struct Node* next; };
```

```
// Structure pour représenter le graphe struct Graph { int numVertices; struct Node** adjLists; };
```

```
// Fonction pour créer un nouveau nœud avec une valeur donnée struct Node* createNode(int value) {
struct Node* newNode = malloc(sizeof(struct Node)); newNode->value = value; newNode->next = NULL;
return newNode; }
```

```
// Fonction pour créer un graphe avec un nombre donné de sommets struct Graph* createGraph(int
vertices) { struct Graph* graph = malloc(sizeof(struct Graph)); graph->numVertices = vertices;
```

```
    // Allocation de mémoire pour les listes d'adjacence
    graph->adjLists = malloc(vertices * sizeof(struct Node*));
    for (int i = 0; i < vertices; i++) {
        graph->adjLists[i] = NULL;
    }
```

```
    return graph;
```

```
}
```

// Reste du code...

Do you want a copy of this to be emailed?

Yes

Enter Email Address

shreyasrajiv327@gmail.com

Press Enter to apply