

DOCUMENT ANALYSIS USING LLM'S

Guided By

Mr Sahibpreet Singh

Mr Puneet Singh

Presented By

Ishwinderpreet Arora

Amay Avasthi

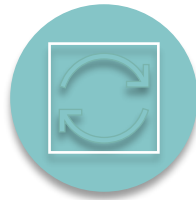
Sai Murali S N

Shania Jairath

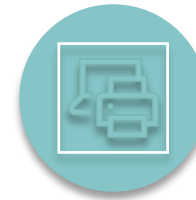
AGENDA



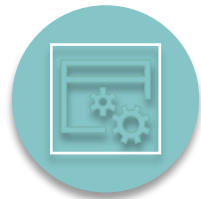
INTRODUCTION



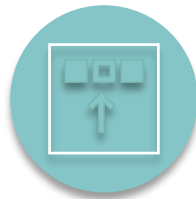
WORKING



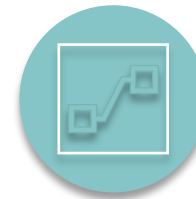
TECHNOLOGIES



TOOLS



USE CASES



CONCLUSION

INTRODUCTION

➤ Document analysis is a process of examining and evaluating documents to extract meaningful information and insights. It involves the systematic review and interpretation of textual, visual, or audio content to understand its context, purpose, and significance.

➤ Document analysis plays a crucial role in various industries, including legal, financial, research, and intelligence, as it provides valuable insights for decision-making, problem-solving, and knowledge discovery.

❖ Importance of Document Analysis

Document analysis plays a crucial role in various industries, including:

➤ Legal: Document analysis is essential in legal proceedings, such as contract review, eDiscovery, and legal research.

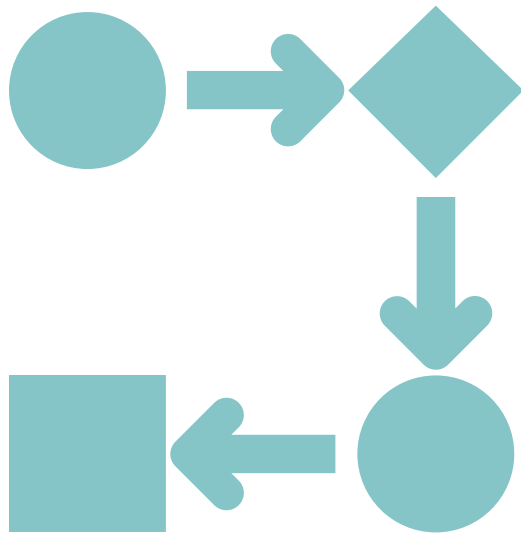
➤ Finance: Document analysis helps in financial statement analysis, fraud detection, and risk assessment.

➤ Healthcare: Document analysis is used for medical record analysis, clinical research, and patient monitoring.

➤ Business: Document analysis enables efficient information extraction, data mining, and competitive intelligence.



Why we choose Document Analysis



Document analysis as a wide range of application. innovation and automation in this field will fasten the process and help achieve maximum productivity.

❖ Here are some reasons why document analysis can be chosen as a method:

1.Information Extraction: Document analysis helps in extracting valuable information from a large volume of unstructured data.

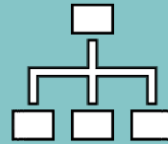
2.Data Organization: Documents often contain unstructured data, and document analysis allows for the organization and structuring of this information.

3.Content Understanding: Analyzing documents helps in understanding the content and context of the information present.

4.Decision Support: Document analysis can provide insights that support decision-making processes.



5. Automated Processing: With advancements in technology, automated document analysis tools and techniques have become more sophisticated. Natural Language Processing (NLP) and machine learning algorithms can be applied to analyze large volumes of text data efficiently.




6. Risk Management and Compliance: Document analysis is crucial in industries where compliance and risk management are paramount. Analyzing documents helps organizations ensure that they adhere to regulatory requirements and identify potential risks.



7. Research and Knowledge Discovery: In academic and scientific research, document analysis is often used to discover new knowledge, trends, or insights within a specific field. It can help researchers stay updated on the latest developments and findings.

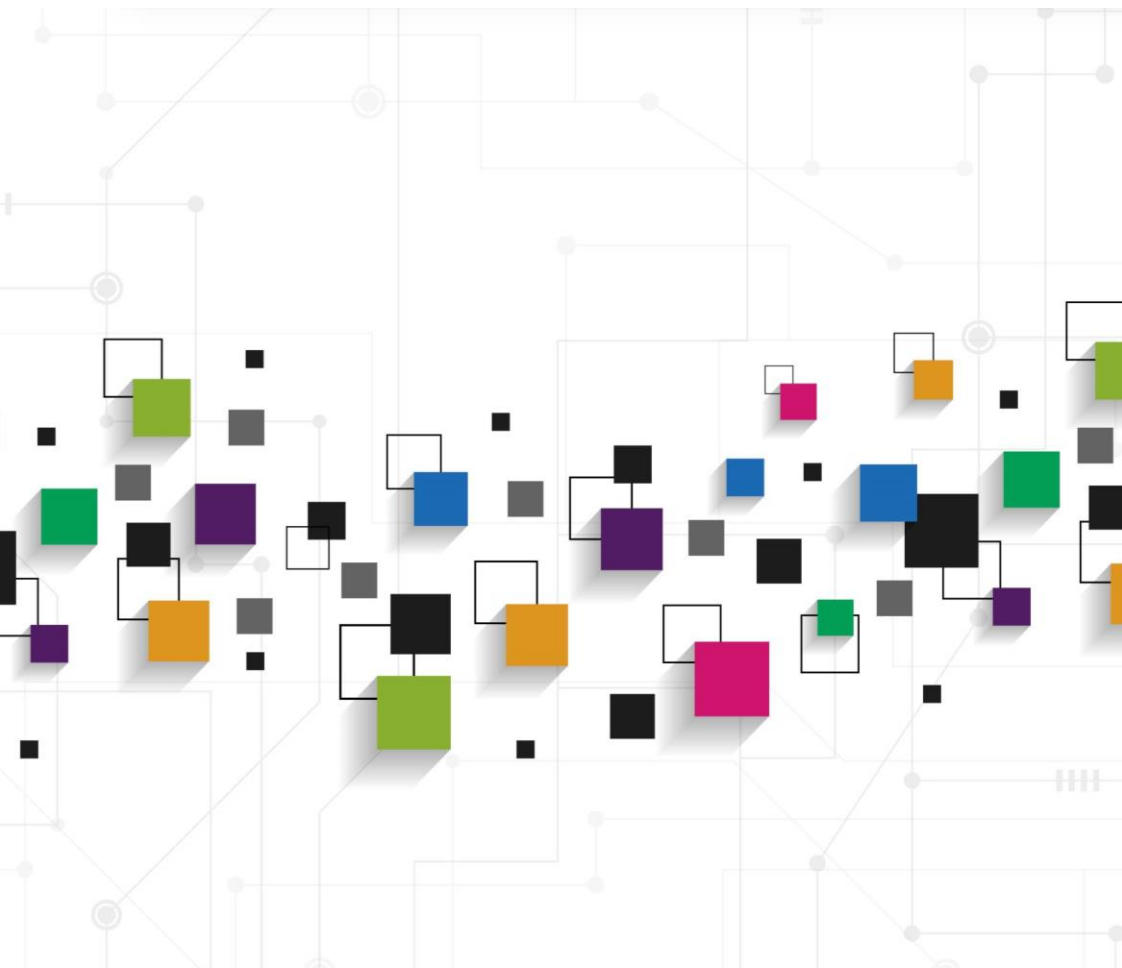
What is language model?

A language model is a statistical method or machine learning model that analyzes and predicts the likelihood of specific sequences of words appearing in a language. In simpler terms, it's like a super-powered dictionary that not only tells you what a word means, but also how likely it is to follow or be followed by other words. This enables them to perform various tasks related to language, such as:

- 
1. Generating text
 2. Translating languages
 3. Answering questions
 4. Summarizing text

5. Writing different kinds of creative content

Data Visualization



What is vector database ?

- A specialized database designed to efficiently store and query high-dimensional vector data, typically embeddings generated from text, images, audio, video, or other unstructured content.
- It excels at similarity search, finding items closest to a given query vector in the vector space.
- It's a key component in modern AI and machine learning applications.

Common examples of vector databases: Pinecone, Faiss, Weaviate, Milvus.

• **Key characteristics:**

- **Vector representation:** Data is stored as vectors, numerical arrays capturing semantic relationships between items.
- **Scalability:** Handles large-scale vector datasets efficiently.
- **Fast retrieval:** Optimized for rapid retrieval of similar items.
- **Integration with ML pipelines:** Works seamlessly with machine learning workflows.

Distinctions from traditional databases:

- Data representation: Vectors (high-dimensional points) vs. structured data (rows and columns).
- Search focus: Similarity search (ANN) vs. exact keyword matching or joins.

Common use cases:

- Recommendation systems: Finding similar products, movies, music, or articles.
- Semantic search: Understanding user intent and context for more relevant results.
- Image and video search: Content-based retrieval based on visual similarity.
- Fraud detection: Identifying anomalous patterns in data.
- Natural language processing (NLP): Tasks like text classification, sentiment analysis, and question answering.

Benefits of using vector databases:

- Fast and accurate similarity search: Optimizes finding similar items in large datasets.
- Scalability for large vector collections: Handles massive vector datasets with ease.
- Support for various vector types and dimensions: Works with diverse embeddings.
- Integration with ML workflows: Seamless incorporation into machine learning pipelines.
- Unlocking new possibilities in AI applications: Powers innovative AI-driven experiences.

What is chroma DB

Chroma DB is an open-source vector database, also known as a vector store. It's designed specifically for storing and retrieving vector embeddings.

But what are vector embeddings? Imagine trying to represent the meaning of a word or a sentence as a point on a map. Each dimension on the map corresponds to some aspect of meaning, and the closer two points are, the more similar the things they represent. Vector embeddings are like those map coordinates - they're numerical representations of data that capture its meaning and relationships.

Chroma DB comes in handy when you're working with large language models (LLMs) or building applications that rely on semantic search. Here's how it works:

Open-source:

- Freedom and flexibility: Users can freely modify, extend, and integrate ChromaDB with other systems without vendor lock-in or licensing costs. And Community-driven development.

Ease of use and developer experience:

- Simple setup and configuration: Can be run in-memory for rapid prototyping or with persistence for production environments, making it accessible for experimentation and development.
- Intuitive API: Python-based API is easy to learn and use, facilitating integration with various machine learning workflows.

Flexible querying capabilities:

- Advanced search operations: Supports range searches, nearest neighbors, similarity searches, and combinations of vector attributes for more complex queries, broadening its applicability.
- Metadata support: Allows filtering and segmentation of results based on additional metadata, enhancing search relevance and context.

Why chroma db over other vector database?

Performance and scalability:

Optimized for vector operations: Built on top of DuckDB and Parquet, which excel in handling vector data, delivering high performance for search and retrieval tasks.

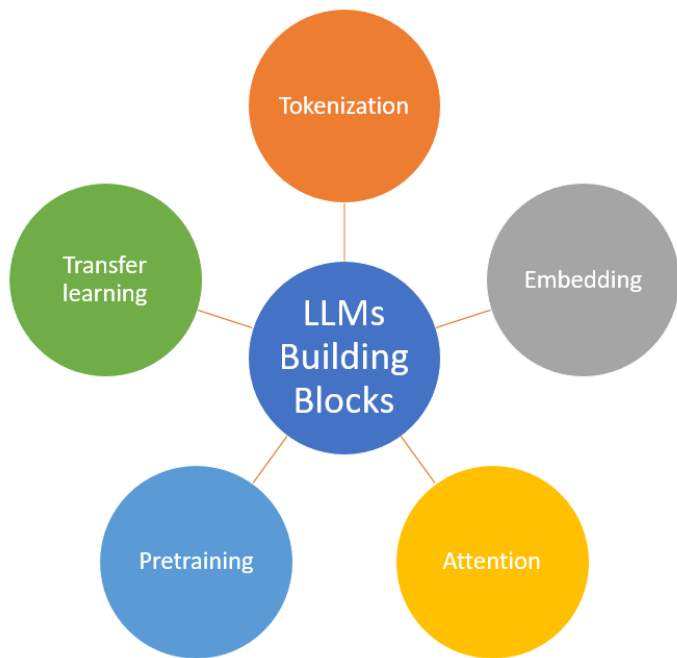
Client-server architecture (in alpha): Enables scaling to larger datasets and distributed deployments, expanding its potential for handling real-world workloads.

ChromaDB is particularly well-suited for:

Rapid prototyping and experimentation: Its ease of use and local testing capabilities make it ideal for trying out new ideas and approaches.

Applications requiring high query performance: Optimized for fast vector operations, making it suitable for real-time recommendations, semantic search, chatbots, and other such applications.

Projects where customizability and open-source values are important: Its open-source nature grants users control over its development and integration with other systems.



Introduction: Overview of LLM

A Large Language Model (LLM) is a type of natural language processing (NLP) model that is trained on massive amounts of textual data to understand and generate human-like language. These models leverage deep neural networks with a vast number of parameters to capture complex patterns and relationships within language. One prominent example of an LLM is GPT-3, developed by OpenAI. Here's an overview of LLMs:

Key Components of LLMs:

1. Transformer Architecture:

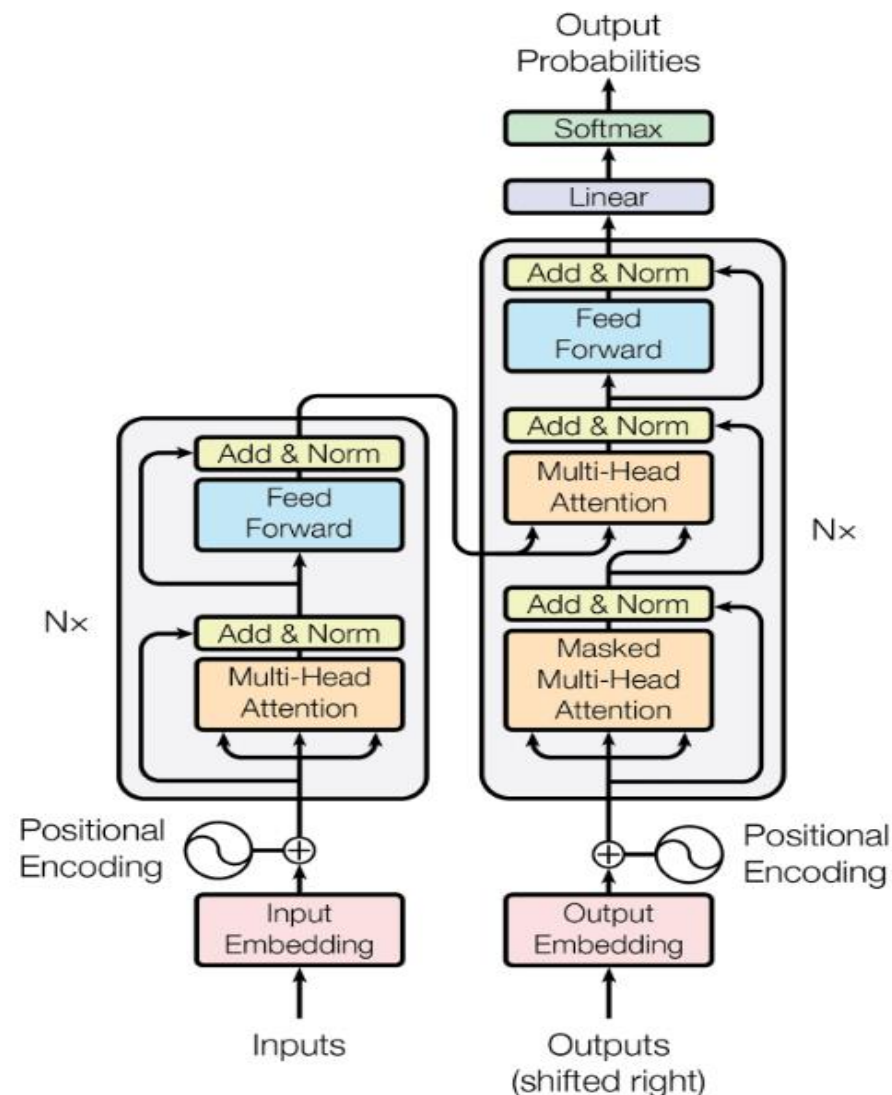
1. LLMs are built on the Transformer architecture, which allows them to capture long-range dependencies and relationships in text efficiently.
2. Transformers enable parallelization of training, making it feasible to handle the large datasets necessary for LLM training.

2. Pre-training on Diverse Data:

1. LLMs are pre-trained on vast and diverse corpora of text from the internet. This pre-training phase helps the model learn the nuances, context, and semantics of language.

3. Attention Mechanism:

1. LLMs use attention mechanisms to weigh the importance of different words in a sentence or sequence. This mechanism allows the model to focus on relevant information during both training and inference.





Working

1. Data Gathering:

- Collect a diverse set of textual documents relevant to the analysis goals.
- Ensure the dataset is representative of the documents the system will encounter.

2. Preprocessing:

- Clean and preprocess the raw text data to remove noise and irrelevant information.
- Tokenize the text into smaller units, such as words or subwords.
- Convert the text into a format suitable for input into the chosen NLP model.

3. Model Selection:

- Choose a suitable NLP model for document analysis. This can be a pre-trained Large Language Model like GPT-3, BERT, or another model based on the specific requirements of the project.

4. Model Input and Inference:

- Input the preprocessed text data into the selected model.
- Allow the model to process the input and generate relevant representations or predictions.
- In the case of LLMs, leverage their ability to understand context and generate human-like text.



5.Information Extraction:

- Extract valuable information from the model's output. This may include key entities, sentiment analysis, or other relevant insights depending on the project goals.
- For document summarization, generate concise summaries that capture the essence of the document.

6. Post-Processing:

- Refine and post-process the extracted information or summaries as needed.
- Address any noise or inaccuracies introduced during the document analysis process.

7. Evaluation:

- Assess the performance of the document analysis system using predefined metrics.
- Compare the system's results with ground truth data or human evaluations.

8. User Interaction:

- If applicable, design a user interface to allow users to interact with the document analysis system.
- Provide a platform for users to input documents and receive analysis results.



9. Iterative Improvement:

- Analyze feedback from users and any areas of improvement identified during testing.
- Iteratively refine the model, preprocessing steps, or post-processing based on evaluation results.

10. Documentation and Reporting:

- Document the entire document analysis process, including methodology, model details, and performance metrics.
- Create a comprehensive report outlining the outcomes and insights gained from the document analysis.

Technologies

Large Language Models(LLM)

Models like GPT-3 (Generative Pre-trained Transformer 3) or other transformer-based models are the core technology for understanding and generating human-like text.

Several Large Language Models (LLMs) have been developed, each with its own strengths and applications. Here are a few examples:

1.GPT-3 (Generative Pre-trained Transformer 3):

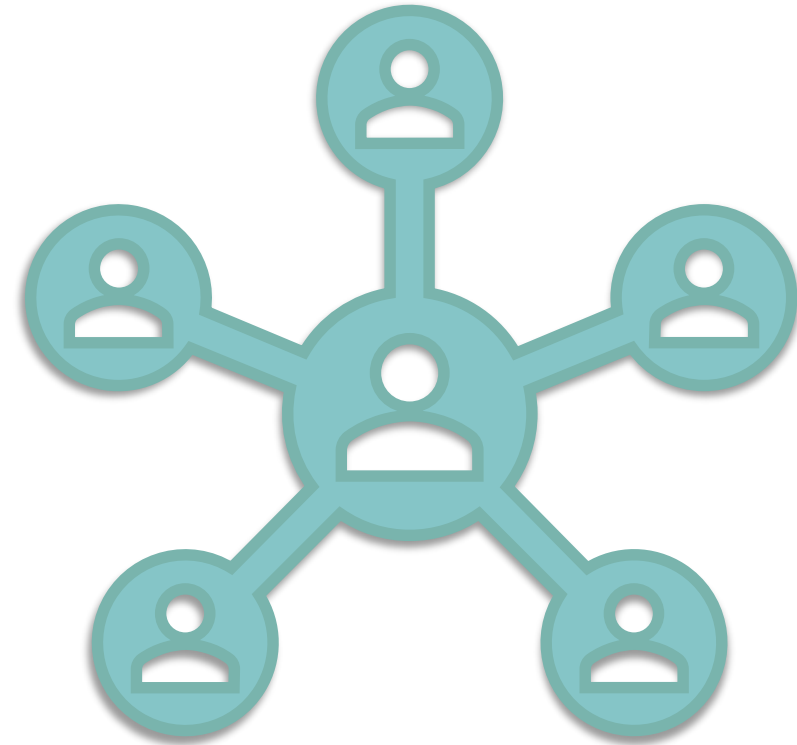
Developed by OpenAI, GPT-3 is one of the largest and most powerful LLMs to date, with 175 billion parameters. It excels in a wide range of natural language processing tasks, including language translation, question answering, and text generation.

2.BERT (Bidirectional Encoder Representations from Transformers):

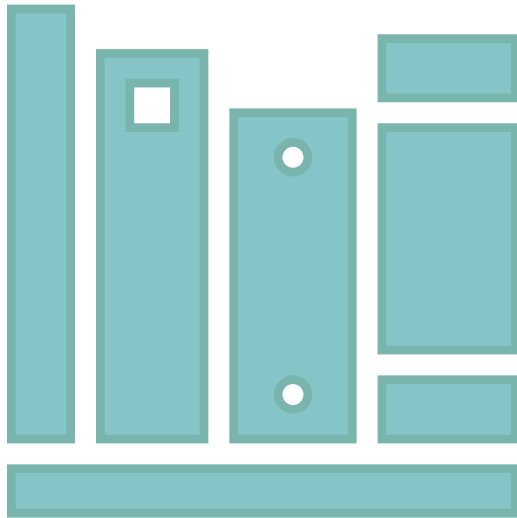
Developed by Google, BERT is known for its bidirectional training and contextual understanding. It has been widely used for various NLP tasks, including sentiment analysis, named entity recognition, and language translation.

3.Gemini:

Gemini is a set of large language models (LLMs) that leverages training techniques from AlphaGo, such as tree search and reinforcement learning. It's intended to become Google's "flagship AI," powering many products and services within the Google portfolio.



Hugging face :



The Hugging Face Transformers library is a popular open-source library for natural language processing (NLP) and machine learning. It provides a collection of pre-trained models and tools that facilitate the development and deployment of state-of-the-art NLP models.

Below are key features and components of the Hugging Face Transformers library.

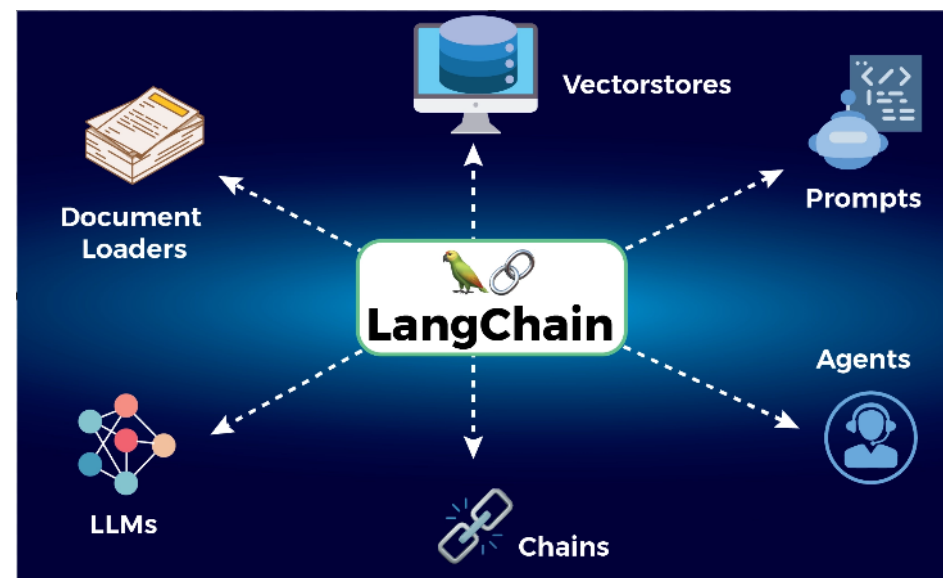
- Pre-trained Models
- Model Hub
- Tokenizers
- Fine-Tuning
- Pipelines
- Integration with Popular Frameworks
- Community and Documentation
- Model Training
- Hugging Face Inference API
- Research Contributions

LangChain

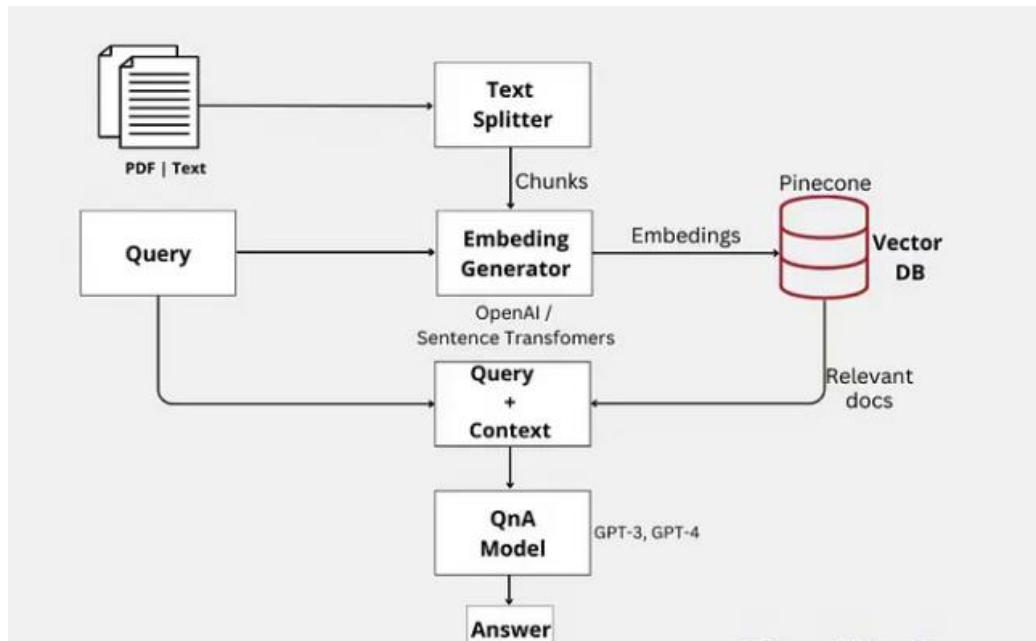
LangChain is a powerful framework aimed at simplifying the development of applications powered by large language models (LLMs). It focuses on building context-aware and reasoning-driven applications by empowering developers with several key components:

Modular Components:

- Chains and Agents:** These allow you to define workflows where the LLM interacts with various input sources (instructions, data) and performs actions like generating text, making decisions, or interacting with external APIs.
- Memory:** Enables persistence of information across interactions, leading to more consistent and context-aware applications.
- Data Augmentation:** Integrate your LLM with external data sources for tasks like question answering, summarization, or code generation.



Development Tools:



- Libraries:** Python and JavaScript libraries offering interfaces and integrations for various components, along with a basic runtime for assembling chains and agents.

- Templates:** A collection of readily deployable reference architectures for diverse tasks, streamlining development for common needs.

- LangServe:** Deploy your LangChain applications as REST APIs for easy integration with other systems.

TOOLS

PyMuPDF (MuPDF):

PyMuPDF, It's a powerful Python library for handling all sorts of tasks related to PDF documents and beyond.

Here's a breakdown of its key features:

- Data Extraction:** Extract text, images, tables, and other elements from PDFs, preserving formatting and layout.
- Analysis:** Analyze and understand the structure of PDF documents, including page count, page size, fonts, etc.
- Conversion:** Convert PDFs to various formats like images (PNG, JPEG), text (TXT), HTML, and even other document formats like EPUB.
- It also includes features like Manipulation, Merge and split ,Form Filling and Decryption





➤ **PyPDF2:**

PyPDF2 is a Python library for working with PDF files. It facilitates basic operations like merging, splitting, and extracting text from PDF documents.

➤ **Beautiful Soup:**

Beautiful Soup is a Python library for pulling data out of HTML and XML files. It can be useful for parsing and extracting text and metadata from PDF documents.

➤ **Flask or Django (for Web Applications):**

Flask and Django are popular web frameworks in Python. They can be used to develop web applications that integrate LLMs for PDF analysis, providing a user-friendly interface.

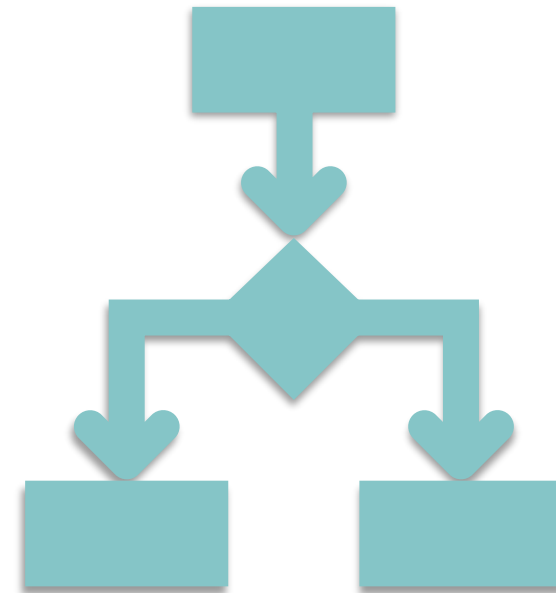
Future Enhancement

Enhanced processing capabilities:

- Multimodal analysis: Beyond text, AI will process various document elements like images, diagrams, tables, and even handwriting, providing a comprehensive understanding of the content.
- Real-time analysis: Documents will be analyzed on the fly, enabling immediate decision-making and action, particularly relevant for live processes like customer service or fraud detection.
- Low-resource settings: AI models will become more efficient, requiring less training data and computational power, making document analysis accessible in resource-constrained environments.

Challenges and ethical considerations:

- Bias: AI models trained on biased data can perpetuate discrimination. Mitigating bias and ensuring fairness will be crucial in responsible document analysis.
- Privacy concerns: Extracting personal information from documents raises privacy concerns. Robust data security measures and ethical frameworks will be needed.
- Explainability and transparency: As AI models become more complex, understanding their decision-making processes will be paramount to building trust and confidence.



USE CASES



Information Extraction: Entity Recognition: Extracting information about entities such as names, dates, locations, and organizations from PDF documents.



Keyphrase Extraction: Identifying and summarizing key phrases or concepts within the PDF content.



Document Summarization: Summarizing Text: Generating concise summaries of lengthy PDF documents, enabling quick understanding of the main points.



Question Answering: FAQ Extraction: Extracting frequently asked questions and generating model-based answers from PDF documents.



Interactive Documents: Creating interactive PDFs where users can ask questions, and the model provides relevant answers.

CONCLUSION

In conclusion, the utilization of Language Models (LLMs) for PDF analysis opens up a plethora of possibilities across diverse domains. The ability of LLMs, such as GPT-3, to understand and generate human-like text allows for sophisticated processing of PDF content. From information extraction to document summarization, question answering, and beyond, the applications are extensive.

The efficiency of LLMs in handling natural language enables streamlined workflows for tasks like legal document analysis, healthcare record summarization, and financial document scrutiny. The technology proves valuable in industries ranging from legal and healthcare to finance and education, providing solutions for tasks that traditionally required extensive manual effort.

THANKYOU!