Table of Contents

1. Introduction

Introduction to Background

Natural Language Processing (NLP) is a branch of artificial intelligence (AI) that allows machines to comprehend and interact with human language. It encompasses two main components: Natural Language Understanding (NLU), which focuses on interpreting text. It reads human language and converts its unstructured data into structured form to make it understandable for computers, and Natural Language Generation (NLG), which deals with producing and automatically generating human-like language that describe, summarize and explain structured input data - processing thousands of pages per second, to produce meaningful phrases and sentences in the form of natural language from vector space (Figure 1.1). NLP relies on a combination of machine learning, deep learning, and computational linguistics to effectively process language data (Figure 1.2). Machine Learning algorithms and artificial intelligence play a crucial role in natural language processing (NLP) and text analytics by enabling systems to interpret and extract meaning from textual data. They significantly enhance, streamline, and automate core text analysis processes, contributing to more efficient and accurate understanding of text documents (B et al., 2021).

Prior to analysis, textual data must undergo preprocessing and cleaning to remove irrelevant elements and minimize noise, which is crucial for accurate language interpretation. Text-processing techniques such as lemmatization, stop-word removal, noise removal, normalization with stemming and tokenization are well-known. Today, NLP is widely used in numerous real-world applications, such as **sentiment analysis** which refers to analyzing sentiment in written contents or posts to identify emotion which always can't be expressed directly. Many companies also leverage sentiment analysis to gauge consumer opinions to help them improve their products. Another example includes **automatic summarization**. Nowadays, users encounter issues when trying to extract relevant details from vast knowledge bases. Automatic summarization helps by condensing the content of documents and records, while also capturing the emotional context within the data (B et al., 2021).

ChatGPT, a notable application of NLP, goes beyond educational support - it also serves purposes in **research, entertainment, software development, code documentation, debugging, test case generation, regular expression creation, provide translation and integration into immersive learning platforms like the metaverse** - a digital realm where individuals engage with a computer-crafted environment and one another in real time, integrating virtual reality, augmented reality, and mixed reality to deliver immersive, interconnected experiences (Zhang et al., 2022). Large language models (LLMs) have significantly advanced, transforming the landscape of natural language processing (NLP) and information retrieval. It has significantly advanced these capabilities. In this project, an OpenAI-powered LLMs is employed to build an interactive chat system that facilitates document analysis and user interaction
(Tin Tin et al., 2024).

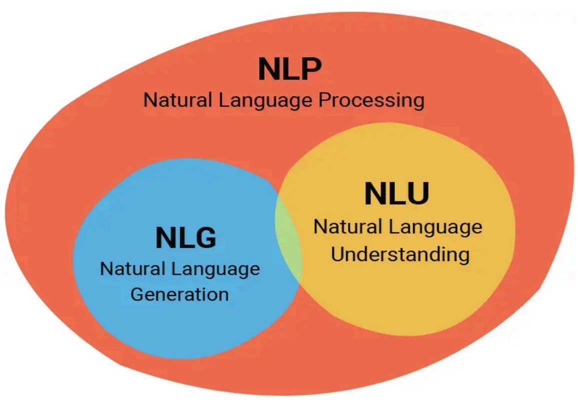


Figure 1.1

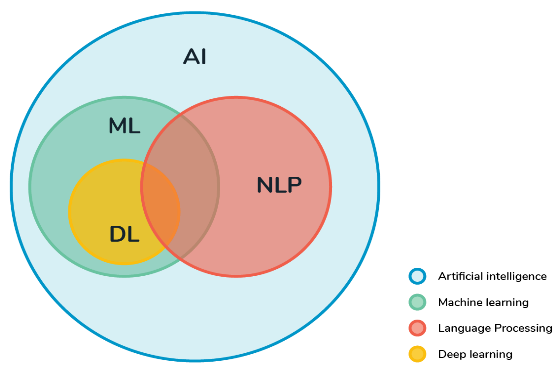


Figure 1.2

Scope

The project aims to build an interactive PDF-based chat application equipped with features such as text extraction, user-defined question answering, and document previewing. Users will have the ability to upload documents, engage in discussions, and receive immediate responses to their queries. The system functions as a real-time assistant, providing on-the-spot explanations and clarifications. A key feature that sets this application apart is its support for multilingual responses. By delivering a dynamic, conversational user interface for navigating and understanding documents, the application enhances user interaction and overall experience. The project will be developed using **Python**, **TypeScript**, **JavaScript** and **CSS**.

Motivation

With the rapid growth of digital information, especially PDFs, users frequently encounter challenges to extract meaningful information when navigating and understanding lengthy or dense textual contents. Traditional methods of document analysis can be time-consuming and inefficient, especially for individuals unfamiliar with technical content. The integration of large language models, such as those developed by OpenAI like GPT-4 which is trendy recently, offers a powerful solution through natural language interaction.

This project is motivated by the need to create an accessible, real-time support tool that allows users to upload PDFs, ask questions directly about the content, and receive instant, context-aware responses. Users can ask questions as they read, receiving instant, tailored responses that help them understand the content more efficiently. This feature enhances the learning process and ensures users remain engaged with and fully grasp the material in the uploaded document. By incorporating intuitive chat-based interaction, the system aims to improve accessibility, enhance comprehension, and streamline the document analysis process for a broad range of users (Taneja & K. Goel, 2025).

Problems

1. **Certain Chatbots Perform Poorly and Cannot Access Past Documents**

   Some chatbots struggle with memory limitations and are unable to retain previous documents in the conversations. As a result, users have to repeat uploading documents and prompt the same questions multiple times, which hampers the overall experience. This highlights the significance of designing chatbots to provide instant context-aware responses to users. It is important to avoid redundancy by preventing repeated actions or similar questions by maintaining a record of previously asked queries in a document database. Since long-term memory plays a crucial role in the chatbot conversations, it becomes evident that **incorporating memory** is a vital step in advancing conversational bots. Therefore, the ultimate goal is to develop a chatbot system equipped with long-term memory capabilities. (Sharma & Sharma, 2023)

2. **Hallucinations and Inaccuracies**

   Hallucinations in large language models (LLMs), defined as factually incorrect, ungrounded, or inconsistent outputs, present a major challenge for an interactive chat application processing PDF or document content. These inaccuracies undermine user trust, as LLMs often produce confident but erroneous answers when handling specialized documents, such as medical reports, legal contracts, or academic papers, where errors can lead to serious consequences like misinformed decisions.

   Hallucinations fall into two categories:

   Ignorance (Case 1):

   Happens when the model lacks the correct information in its parameters, meaning it doesn't have the knowledge needed to answer accurately.

   Error (Case 2):

   Occurs when the model contains the necessary knowledge but still gives an incorrect response, indicating a flaw in its processing or reasoning.

As LLMs are deployed in document-based applications, the risk of hallucinations increases due to the diverse and specialized content in documents like industry reports or research papers. **Ignorance-based hallucinations** are particularly prevalent with niche or emerging topics, while **error-based hallucinations** produce unreliable outputs even for familiar content. The overconfidence of LLMs exacerbates the issue, as users may trust inaccurate responses in a conversational interface designed for quick insights, often requiring manual verification. Current mitigation strategies are inadequate for real-time document interactions, highlighting the need for innovative solutions to ensure reliable performance (Simhi, 2025).

3. **Outdated Knowledge**

The problem of outdated knowledge significantly undermines the effectiveness of interactive chat applications using large language models (LLMs) for conversations with PDF or document content. LLMs rely on static, pre-trained knowledge that quickly becomes obsolete, especially in rapidly evolving fields, leading to inaccurate or incomplete responses when processing recent documents. **The lack of mechanisms to dynamically update information or integrate real-time external knowledge sources**, such as web searches or databases, limits the application's ability to contextualize or verify document content against current data. This results in misaligned responses that fail to reflect the latest terminology, methodologies, or findings, causing a mismatch with user expectations for timely and relevant answers. Consequently, the application's reliability and user trust are compromised, particularly in professional or academic settings where up-to-date insights are critical (Mousavi, 2024).

Objectives

Over time, a range of tools and techniques have emerged to tackle the problem of text-heavy contents from large volumes of digital documents, ranging from simple keyword searches to advanced text mining and natural language processing (NLP) methods. However, many of these approaches still struggle to deliver fast and contextually accurate results. (Roy et al., 2025)

To further enhance current chatbot systems, this project specifically seeks to:

1. Develop a system to extract and process text content from uploaded PDF documents, including handling various PDF structures to ensure accurate and comprehensive extraction.

2. Implement a conversational interface that enables real-time interaction with document content. Users are allowed to instantly ask questions and receive instant as well as relevant clarifications.

3. Enable document preview and highlighting features to visually guide users to the relevant sections of the text in response to their queries.

4. Enable view of similarity between the content of two PDF documents.

5. Enable speech to text and text to speech functionality.

6. Generate factual and highly accurate responses with valid support.

7. Avoid feeding LLMs large sizes of content, feed the LLMs only the relevant information instead to ensure efficiency.

Through these objectives, our project aspires to bridge the gap between static document consumption and dynamic user interaction, ultimately transforming the way users access, understand and engage with complex textual content.

2. Research Background

Literature Review

The rapid evolution of technology, particularly in the field of natural language processing (NLP) and the development of large language models (LLMs), has revolutionized how users interact with textual data. Traditional methods, such as manually downloading and reading PDF documents, often lack interactivity and require users to seek external resources for clarification, which can be inefficient and time-consuming (Tin Tin et al., 2024). This project underscores the need for a more dynamic, user-centric approach by developing an interactive chat-based system that leverages the capabilities of advanced language models, specifically focusing on open-source LLMs. Such systems aim to bridge the gap between static document consumption and real-time, context-aware interaction, enabling users to engage with document content more effectively.

Large Language Models (LLMs) have significantly transformed the landscape of natural language processing (NLP) in several other fields, especially in information retrieval. (MENGHANI, 2021). These models have become foundational in enabling natural, human-like dialogue, driven by the explosion of data and advancements in computational power. The enhanced performance of LLM-based chatbots has led to their widespread adoption across diverse domains, including education, research, and technical support. Their ability to process natural language with high contextual relevance and accuracy, combined with their capacity to handle large-scale information, makes them indispensable for applications requiring precise knowledge extraction and user interaction (Dam, 2024). In this project, we implement three LLMs which are **Gemini-2.0-flash**, **llama 3.3 70B Versatile (**llama3-70b-8192**)**, and **DeepSeek Chat v3 (**deepseek-chat-v3-0324**)** to create an interactive chat system tailored for document analysis, allowing users to upload PDFs, ask questions, and receive contextually relevant responses in real time.

Gemini 2.0 Flash is a multimodal LLM from Google DeepMind, can process text, code, images, and structured data. It is optimized for speed and efficiency, while maintaining strong performance in reasoning, question answering, and cross-document analysis (*Gemini 2.0 Flash | Generative AI on Vertex AI*, 2025). **llama 3.3 70B Versatile (**llama3-70b-8192**)**, developed by Meta, is a large-scale model fine-tuned for a broad range of language tasks, including summarization, instruction-following, and open-domain question answering, with improved accuracy and lower hallucination rates (*Llama-3.3-70B-Versatile*, 2024). **DeepSeek Chat v3 (**deepseek-chat-v3-0324**)**, by DeepSeek-Vision, is an advanced LLM optimized for dialogue, instruction-following, and code generation, offering strong multilingual support and long-context reasoning, ideal for tasks like document analysis and interactive Q&A (*🚀 Introducing DeepSeek-V3*, 2024) (*DeepSeek-V3-0324 Release*, 2025). Together, these models power a robust chat interface that allows users to upload PDF documents, pose natural language queries, and receive contextually accurate, real-time responses grounded in the content of the uploaded materials.

Research has demonstrated the effectiveness of interactive chat interfaces for exploring information within PDFs, emphasizing key elements such as understanding user intent, employing strategic information retrieval methods, and utilizing LLMs to generate contextually appropriate responses (Haristiani, 2019). Building on this foundation, our project aims to provide a chatbot powered by a few large language models (LLM) that facilitate seamless interaction with PDF content, addressing the challenges of navigating and understanding dense textual data.

Building upon the initial contexts introduced earlier, one of the key considerations in the system development is the evaluation and selection of the most appropriate tool or algorithm. To inform this decision, a comparative study was conducted, examining the chosen tool or algorithm alongside other relevant options. Table 2.1 outlines the results of this analysis, providing insights into their respective features, advantages, limitations, and overall suitability for the project's objectives.

On the other hand, based on our recent research, five advanced models have gained significant attention for their capabilities in conversational AI and document interaction: **ChatGPT**, **DeepSeek**, **Claude**, **Gemini**, and **llama**. These models are compared in Table 2.2, Table 2.3,

Table 2.4 which explore their foundational information, key strengths and weaknesses, ideal use cases, and pros and cons in a structured format. This comparison provides a comprehensive understanding of each model's strengths and limitations, enabling an informed decision for our project's requirements.

Portable Document Format (PDF)

Figure 2.1 Adobe PDF Logo

Portable Document Format (PDF) was developed and specified by Adobe Systems (Figure 2.1), incorporated beginning in 1993, and standardized by ISO in 2007 (Adobe Systems Incorporated, 2008). PDF files may be created natively in PDF form, converted from other electronic formats, or digitized from paper, microform, or other hard copy formats.

- PDF allows documents to be created and viewed independently of the environment or software used.
- PDF supports interactive elements like hyperlinks, annotations, and hypertext, enhancing document interactivity.
- PDF is widely used by businesses, governments, libraries, and institutions for important information storage.
- PDF facilitates the electronic exchange of documents and is evolving with applications in electronic workflows and engineering data.

In our project, PDFs are used as the input for the interactive chat application, enabling users to upload their documentation in PDF format for discussion and analysis within the interface.

The impact and opportunity for the application of PDF will continue to grow at a rapid pace.

Analysis of Selected Tool/Algorithm with Any Other Relevant Tools/Algorithms

Tools Comparison	**PyMuPDF (selected)** (McKie, 2019)	**PyPDF2** (Fenniak, 2012)	**pdfplumber** (Vine, 2016)	**pdfminer.six** (Shinyama, 2019)
Type of license and open source license	Open-source AGPL and commercial license agreements from Artifex PyMuPDF License	BSD License PyPDF2 License	MIT License pdfplumber license	MIT License (MIT/X) pdfminer.six license
Year founded	2016	2005	2015	2014
Founder	Jorj X. McKie	Mathieu Fenniak	Jeremy Singer-Vine	Yusuke Shinyama
License Pricing	Free	Free	Free	Free
Supported features	Extracting text, images, and attachments from various document types, OCR, and more PyMuPDF Documentation	Extracting text, images, and attachments from PDF; post-processing of extracted text; encryption and decryption of PDFs, and more PyPDF2	Extracting text, images, and attachments from PDFs; post-processing of extracted text; encryption and decryption of PDFs; rotating, cropping, and	Extracting text, elements, and images from PDF pdfminer.six Documentation

		Documentation	transforming pages; splitting and merging PDF files pdfplumber Documentation	
Common applications	Data extraction, mining, and manipulation for efficient document processing and analysis	Data extraction, mining, and manipulation for document processing and analysis	Data extraction, mining, and manipulation for document processing and analysis, especially tables and text	Document processing and analysis through data extraction and mining
Advantages	Extensive feature set; Python compatibility, and high-performance rendering	Open-source availability; compatibility with Python; and extensive feature set	Strong table extraction; lightweight	Open-source availability; compatibility with Python, and extensive feature set
Limitations	License Restrictions	Limited Support for Interactive Content	Limited support for complex PDFs, no advanced manipulation	Limited Support for Interactive Content

Table 2.1

Justification on Selected Tools/Algorithms

1. Portable Document Format (PDF) Extractor

For extracting content from Portable Document Format (PDF) files, **PyMuPDF** has been selected due to its extensive feature set and high-performance rendering capabilities. It supports extracting text, images, attachments, and even performing OCR, making it a powerful and flexible choice for applications that demand comprehensive document processing. PyMuPDF's compatibility with Python ensures seamless integration and minimal overhead, supporting efficient workflows across a variety of projects. While license restrictions may apply depending on usage, its robust functionality and speed make it a well-suited tool for accurate and efficient PDF content extraction in diverse use cases.

Chatbot Developments

A chatbot is a virtual assistant powered by Artificial Intelligence (AI) and Machine Learning (ML), designed to engage with users via text or voice, offering round-the-clock support for task automation and query resolution (Tamrakar, & Wani, 2021). It leverages Natural Language Processing (NLP) to facilitate smooth interactions between humans and machines, enhancing communication in various business domains by understanding and responding to user inputs (Alwahaishi & MEROUANE, 2022).

The concept of chatbots dates back to the 1950s with Alan Turing's experiment to test machine-human differentiation. The earliest chatbot, ELIZA, was created in 1966 by Joseph Weizenbaum at MIT, relying on rule-based keyword matching. This was followed by PARRY, developed by Kenneth Colby, and ALICE by Richard Wallace in 1995, with ALICE earning multiple Loebner Prize awards for its advancements (Gambhir, 2019).

Developing a chatbot involves five main steps: identifying its purpose, deciding between rule-based or Natural Language Processing (NLP) frameworks, defining user intentions, crafting the conversational flow, and choosing a deployment platform such as WhatsApp, Telegram, or Facebook Messenger.

The operational structure of a chatbot starts with user input, which is processed by a Natural Language Understanding (NLU) module to interpret intent, followed by data retrieval from knowledge bases or APIs, and dialogue management to track conversation history (Tamrakar, & Wani, 2021).

Chatbots are applied in sectors like customer support, education, and business, with examples including UNIBOT for university inquiries. Integration with IoT, such as Telegram chatbots for smart workspace management to expand their utility (Tamrakar, & Wani, 2021).

.

Comparison of Chat Models

Model	ChatGPT (Altman, 2022) ChatGPT Official Website	DeepSeek (Wenfeng, 2023) Deepseek Official Website	Claude (Amodei, 2021) Claude Official Website	Gemini (Hassabis, 2023) Gemini Official Website	Llama (Meta Platforms, Inc., 2023) llama Official Wesbite
Logo	ChatGPT	deepseek	Claude	Gemini	
Provider	OpenAI	DeepSeek Labs	Anthropic	Google DeepMind	Meta
Launch Year	2022	2024	2023	2023	2024
Model Type	GPT-based large language model (LLM), e.g GPT-4	Proprietary GPT-style LLM, e.g DeepSeek R1	LLM developed with a safety and alignment focus, e.g Claude v1	Multimodal LLM supporting text, code, and media input, e.g Gemini	Large-scale LLM fine-tuned for broad language tasks, e.g., llama 3.3 70B Versatile
Current Knowledge Access	Good	Moderate	Poor	Excellent	Moderate
Primary Use Case	Exceptional writing skills for content creation	Strong logical reasoning and problem-solving	Strong analytical skills for	Excels in data analysis and processing	Summarization, instruction-following, and

		skills	problem-solving and handling complex queries		open-domain question answering
Multilingual support	Works across many languages for smooth communication	Supports multiple languages, especially for technical fields	Multilingual with a focus on safe usage	Handles multiple languages well, especially in different subject areas	Supports multiple languages with strong task versatility
Data Privacy & Ethics	Strictly follows OpenAI's privacy guidelines	Prioritizes data privacy and security in industrial fields	Strong safety measures to protect user data	Uses Google's security and privacy policies	Follows Meta's privacy and ethical guidelines
Developer/ API Access	Yes	Yes	Yes	Yes	Yes

Table 2.2

Key Strengths and Weak Strengths of Each Model

Model	Key Strengths	Weak Strengths
ChatGPT	Versatile for tasks like web browsing, DALL-E, and voice interactions.	May produce errors; high computational needs; some features paywalled which users need to pay a subscription or a one-time fee to unlock premium features or contents.
DeepSeek	Specializes in industry applications and NLP tools such as NLTK (Natural Language Toolkit) and OpenAI API.	Limited multimodal capabilities; smaller ecosystem, for instance, limited integrations, narrow application scope and smaller developer community
Claude	Prioritizes ethical and safe AI responses.	Inherits training data biases; needs diverse inputs to reduce potential issues.
Gemini	Advanced for processing text, images, audio, and video.	Occasional accuracy issues; some advanced features require subscriptions.
llama	High performance in summarization, instruction-following, and open-domain question answering with lower	Limited multimodal capabilities; requires significant computational

	hallucination rates.	resources for large-scale tasks.

Table 2.3

Pros and Cons of Each Model

Model	**Pros**	**Cons**
ChatGPT	Performs reliably across a wide range of tasks; delivering well-structured, context-aware answers, with easy custom fine-tuning and support for commercial use.	Overly cautious that possibly will exhibit biases in generated outputs; may decline borderline questions, and can be moderately costly to use.
DeepSeek	High performance in problem-solving, excels in logical reasoning for technical tasks like coding and maths queries.	Fact-focused, lacks expressive creativity as struggling with nuanced or open-ended document queries.
Claude	Engaging, adaptable, natural, and flexible responses; enable a broad range of tasks by supporting up to 100k tokens (approximately 75k words) in a single message.	Text-only, no multimodal or web browsing capabilities; may struggle to produce highly human-like responses.

Gemini	Real-time info access via web searches; strong in processing large datasets for document insights.	Weaker in handling complex conversational flows as limited tone and style customization.
llama	High accuracy in summarization and question answering; supports broad task versatility with fine-tuning capabilities.	Less effective for creative tasks; requires significant computational resources.

Table 2.4

Evaluations

Choosing the right AI model depends on one's specific needs, as ChatGPT, DeepSeek, Claude, Gemini, and llama each offer unique strengths. **ChatGPT** is a versatile option for creative and conversational tasks, with strong writing and context retention, ideal for PDF interactions, though it can be cautious. **DeepSeek** fits cost-conscious users seeking technical expertise in logical reasoning, but its lack of creativity limits its use for nuanced queries. **Claude** shines in communication with engaging responses, yet its text-only focus restricts broader tasks. **Gemini**'s multimodal capabilities and Google integration suit productivity work, but its weaker conversational flow may hinder chat tasks. **llama** excels in summarization and question answering with high accuracy, perfect for document queries, despite requiring significant computational resources. For this project's focus on PDF-based chat, ChatGPT and llama are the best fits due to their conversational strengths.

3. Methodology

Application of Algorithms/Libraries

1. **LangChain**

   LangChain serves as an open-source platform aimed at speeding up the construction of tailored AI applications powered by Large Language Models (LLMs). It stands out for its ability to connect with a variety of data repositories and software, earning widespread recognition among AI practitioners. At its core, it enables the development of specialized AI tools beyond simple online interactions, meeting the demand for customized solutions in areas such as education, customer support, healthcare, and media production. It also offers adaptable processes that pair LLMs with prompts to perform a series of tasks. Langchain's critical components include input instructions that act as queries sent to LLMs, crafted dynamically with user requests, sample responses, and guidance. LangChain provides Input Frameworks for consistent prompt generation. In terms of document-based inquiry, it facilitates applications that respond to questions about external files (e.g. PDFs, websites) through steps like importing, dividing, encoding, storing, and searching data. LangChain offers File Importers, Data Splitters, Encoding Models, Data Repositories, and Search Tools to support this (Topsakal & Akinci, 2023).

2. **OpenAI API**

   Application Programming Interface (API) has existed since the era of personal computers and gained prominence with web APIs around 2000. They are now seen as a cornerstone of the digital ecosystem, enabling interconnection between people, applications, and systems. The majority of API research concentrates on technical dimensions, including design, development, usage patterns, security, and management, reflecting their roots in computer science. APIs are increasingly vital for digital transformation, with companies like Netflix and Uber leveraging them for customer experience, ecosystem expansion, and efficiency (Ofoeda et al., 2019). A research named **"Development of OpenAI API-Based Chatbot to Improve User Interaction on the JBMS Website"** incorporated

OpenAI API into a chatbot for the Journal of Business, Management, and Social Studies (JBMS) website to improve user interaction and streamline information retrieval, delivering accurate responses and enhancing the overall user experience. It enables the chatbot to provide precise answers, improving the user experience. The API powers features like ChatGPT-style interactions, allowing the chatbot to explain and summarize details. The OpenAI API was utilized alongside AI, Machine Learning (ML), and Natural Language Processing (NLP) to enhance functionality. (Pires, 2024)

3. **Natural Language Toolkit (NLTK)**

   NLTK is a prominent open-source Python library for Natural Language Processing (NLP), enabling work with human language data. It supports tasks like tokenization, stemming, lemmatization, part-of-speech tagging, parsing, classification, and semantic reasoning. It caters to a diverse audience, including linguists, engineers, students, educators, researchers, and professionals in industry, making it ideal for teaching, research, and building NLP prototypes in areas like computational linguistics, cognitive science, and AI. It offers tools for text processing, including categorization, linguistic analysis, and visualization. Moreover, it integrates with advanced NLP libraries and allows customization for building NLP models, and handles both basic and complex NLP tasks, such as removing stop words, analyzing word frequencies, chunking, and chinking (Bird & Loper, 2024).

4. **BERT (Bidirectional Encoder Representations from Transformers)**

   BERT, introduced by Jacob Devlin in 2018, is a groundbreaking unsupervised, deeply bidirectional language model that processes text by considering both left and right contexts simultaneously, enabling a human-like understanding of language semantics and context. It consists of stacked Transformer encoders, with two primary configurations:

   - **BERT_BASE**: 12 layers, 768 hidden units, 12 attention heads;
   - **BERT_LARGE**: 24 layers, 1024 hidden units, 16 attention heads.

     BERT uses two unsupervised pre-training tasks:
   - **Masked Language Model (MLM)**: Randomly masks 15% of input tokens, predicting them based on context, using a mix of [MASK] tokens (80%), random

tokens (10%), and unchanged tokens (10%) to reduce pre-training/fine-tuning mismatch;

- **Next Sentence Prediction (NSP)**: Trains the model to predict if one sentence follows another, enhancing understanding of sentence relationships for tasks like question answering and natural language inference (Chakkarwar et al., 2023).

BERT can be fine-tuned for various NLP tasks by adding a single output layer, requiring minimal task-specific architecture changes, making it efficient and adaptable. Applications of BERT in NLP tasks include:

- **Text Summarization**: T-BERTSum integrates BERT with neural topic models and LSTM for topic-aware, extractive, and abstractive summarization, reducing information redundancy.
- **Sentiment Classification**: TopicBERT variants (e.g., TopicBERT-ATP, TopicBERT-TA) enhance BERT by incorporating topic-aware sentiment classification, addressing homonymy and polysemy for better corpus-level sentiment analysis.
- **Tweet Analysis**: BERTweet, pre-trained on 850M English tweets, excels in tasks like POS tagging, sentiment analysis, and NER by handling informal language, slang, and emoticons (Chang, 2019) .

5. **Sentence Transformer**

Sentence Transformers, introduced by Reimers and Gurevych in 2019, this approach modifies the BERT framework by incorporating Siamese and triplet network architectures to produce semantically rich sentence embeddings. These embeddings enable efficient comparison of sentences using cosine similarity, making them ideal for tasks like semantic textual similarity. Unlike BERT, which is computationally intensive (e.g. taking 65 hours to compare 10,000 sentence pairs). Sentence Transformers, particularly Sentence-BERT (SBERT), are significantly faster, computing embeddings in approximately 5 seconds and performing comparisons in ~0.01 seconds, enhancing scalability. It generates embeddings that capture sentence-level semantics, outperforming word-level embeddings like GloVe, which struggle with sentence similarity. (Ahmed et al., 2022)

Two models used in this project:

- **all-mpnet-base-v2 (Mpnet)**
- **all-MiniLM-L6-v2 (Minilm-16)**

all-mpnet-base-v2 (Mpnet):

- Use in encoding textual data into high-dimensional vector embeddings for semantic similarity tasks.

all-MiniLM-L6v2 (Minilm-16):

- Minilm-16 is introduced alongside Mpnet and other models as part of the comparative analysis, used to encode titles and abstracts for journal recommendation.

Overall, Sentence Transformers capture nuanced semantic relationships, outperforming them in handling complex scientific texts and interdisciplinary topics (Colangelo, 2025).

6. **MongoDB**

MongoDB is a leading open-source, NoSQL document-oriented database written in C++ (Chauhan, 2019). It organizes data as BSON (binary JSON) documents in collections, similar to tables in relational databases (RDBMS) like MySQL. It is flexible due to the dynamic schema that allows documents in a collection to have different structures, unlike MySQL's rigid schema. The dynamic schema supports diverse data types (e.g. multimedia) and simplifies data updates. It showed its scalability with the auto-sharding splits data across shards, enabling horizontal scaling beyond hardware constraints. It performs well by supporting embedded data models and indexing, reducing I/O operations and enabling faster queries, including keys from nested documents and arrays, as well as offers powerful CRUD operations, data aggregation, and text search. MongoDB excels in big data, real-time web apps, **content management**, social and mobile platforms, user data management, and data hubs, especially for non-critical, small to medium sensor applications with **high write demands** (IJSRD Journal, 2015).

7. **Chroma DB**

   Chroma DB is an open-source, AI-native vector database engineered to streamline the development of applications powered by large language models (LLMs), offering a unified platform for managing embeddings, vector search, document storage, full-text search, metadata filtering, and multi-modal data. Its core features include automatic tokenization, embedding, and indexing for seamless semantic similarity searches, with support for custom or pre-computed embeddings. It efficiently stores documents with metadata, enabling advanced filtering, with each document requiring a unique ID. Chroma supports querying via text or embeddings, delivering the top n most similar results with optional metadata or content filters. It operates as a server with Python and JavaScript/TypeScript SDKs, a lightweight HTTP client (chromadb-client), ensuring flexibility for diverse workflows. (*Chroma*, 2023)

System Design

RAG is an AI framework that enhances large language models (LLMs) by integrating them with external information sources. This enables AI systems to generate more accurate, up-to-date, and contextually relevant responses by retrieving relevant data from the external information sources before generating an answer. (guide & Selvaraj, 2025; Google, n.d.)

RAG combines two components, namely **retrieval** and **generation**. In the proposed system, the **retrieval** component retrieves the relevant information from the selected pdf that is uploaded by the user. In the **generation** component, LLMs, such as **gemini-2.0-flash**, **llama3-70b-8192**, and **deepseek-chat-v3-0324**, are fed with the retrieved information. The LLMs use the information as context to produce coherent and informed responses to the user's query.

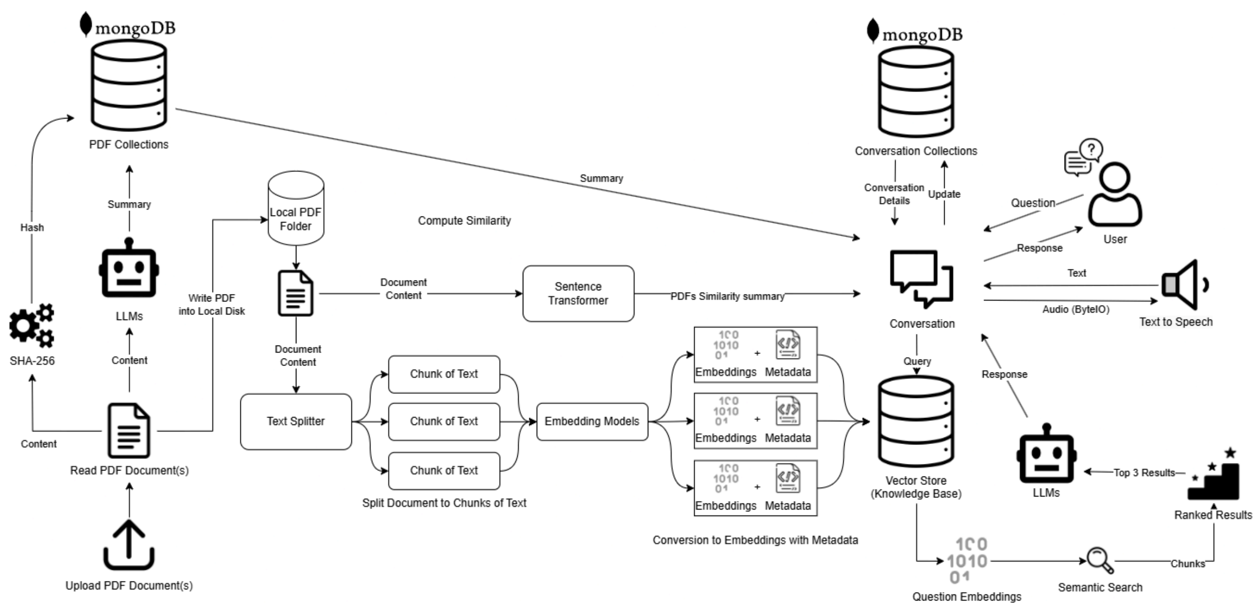


Figure 3.1 (refer to Appendix)

Figure 3.1 illustrates the architecture of the proposed **Retrieval-Augmented Generation (RAG)** system. The process begins when a user creates or selects a conversation. To initiate a chat with the Large Language Models (LLMs), the conversation must be associated with at least one PDF document. Users can either upload a new PDF at any time or select from previously uploaded documents. Upon uploading, the content of the PDF is extracted and cleaned. A hash of the content is generated and checked against existing records in the MongoDB database to avoid duplicate storage. If the PDF is unique, it is saved to a local server folder, and its content is sent to the LLMs for summarization. The summary, along with the hash of the content and the local file path, is stored in MongoDB. When a user creates a new conversation, a corresponding document is added to MongoDB to store the conversation details. The content of the associated PDFs is then split into chunks. These chunks are embedded using an embedding model and stored in a vector store for efficient retrieval. During a conversation, whenever the user asks a question, the vector store retrieves the top three most relevant chunks based on the query. These retrieved chunks serve as context, while the user's question acts as the query input. Both inputs are processed by the LLMs to produce a coherent and contextually relevant response. Additionally, users can view summaries of the selected PDFs and play audio versions of the text. If an existing conversation is selected, its details are loaded from MongoDB, allowing the user to seamlessly continue the discussion with the LLMs.

Strengths

Retrieval-Augmented Generation (RAG) systems offer several key strengths that significantly enhance the capabilities of Large Language Models (LLMs), especially in real-world applications requiring accuracy, contextual relevance, and up-to-date information.

Improved Factual Accuracy and Reduced Hallucinations

One of the primary limitations of standard LLMs is their reliance on static training data, which can lead to outdated or hallucinated (completely false) responses and supported by fake citations. RAG systems address this by retrieving relevant, up-to-date content from external sources at query time. By grounding generation in actual documents, RAG minimizes factual errors and enhances the trustworthiness of outputs with highly accurate and contextually appropriate responses (Xu et al., 2024).

Contextual Relevance

Within the RAG framework, the process begins with the system identifying and retrieving relevant chunks from an extensive document corpus in response to a user's query (Louis et al., 2024). This targeted retrieval ensures that the generative model has access to highly relevant information, enabling it to provide responses that are contextually precise and aligned with the user's intent. As a result, RAG greatly enhances the model's ability to handle complex queries and effectively retrieve precise information from extensive datasets, establishing it as a powerful tool for accurate knowledge extraction.

Efficient Knowledge Integration

In a normal Large Language Model (LLM), all the knowledge is stored inside the model itself. That means if new information becomes available (like a new law or medical discovery), the model needs to be retrained to learn it, which is time-consuming and expensive. RAG systems work differently. They don't store all knowledge inside the model. Instead, they connect the model to an external source of documents, the uploaded PDFs. When the user asks a question, RAG retrieves the most relevant chunks from this external source and gives them to the model to help it answer.

Evaluation Metrics

Accuracy

Based on a simple evaluation method used in classification and information retrieval tasks. It evaluates whether a predicted answer is correct or not, based on a binary decision. Then the accuracy is calculated using the formula: **Accuracy = (Number of Yes (Correct Answers) / Total Number of Predictions) × 100%**. This metric provides a clear and interpretable measure of how often the system provides correct outputs in a binary fashion.

$$Accuracy = \frac{Number\ of\ Correct\ Answers}{Total\ Number\ of\ Predictions} \times 100\%$$

Mean BERTScore

BERTScore is an evaluation metric used in natural language processing (NLP) to assess the quality of candidate text (generated from the model) by measuring its semantic similarity to a reference text (expected). Unlike traditional metrics like BLEU or ROUGE, which rely on surface-level word or n-gram matching, BERTScore leverages contextual embeddings from pre-trained transformer models like BERT, RoBERTa, or DeBERTa to capture deeper semantic meaning. This makes it particularly effective for tasks like machine translation, text summarization, and question answering, where paraphrasing or varied phrasing is common (Mahmood et al., 2024). BERTScore contains three components, namely precision, recall and f1-score.

BERTScore begins with splitting (tokenize) both reference and candidate texts into words (tokens) using a tokenizer compatible with the chosen BERT model. The tokenized texts are passed through a pre-trained BERT model to generate contextual embeddings for each token. Then, BERTScore calculates cosine similarity between the embeddings of tokens in the candidate and reference texts to find the most similar token pairs. Precision matching and recall matching are carried out (Mahmood et al., 2024).

- **Precision Matching**: For each token in the candidate text, identify the token in the reference text that yields the highest cosine similarity score based on their contextual embeddings (Mahmood et al., 2024).

- **Recall Matching**: For each token in the reference text, determine the token in the candidate text that exhibits the highest cosine similarity score based on their contextual embeddings (Mahmood et al., 2024).

Cosine similarity:

$$sim(t_i, t_j) = \frac{t_i \cdot t_j}{||t_i||\ ||t_j||}$$

where t_i and t_j are the embeddings of tokens from the candidate and reference texts, respectively (Mahmood et al., 2024).

Precision:

Precision measures how much of the candidate text's content aligns with the reference text (avoiding irrelevant additions).

$$P_{BERT} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_j \in \hat{x}} max\ sim(\hat{x}_j, x_i)$$

where $\hat{x}$ is the candidate text, x is the reference text, and $|\hat{x}|$ is the number of tokens in the candidate text. This averages the maximum similarity scores for each candidate token (Mahmood et al., 2024).

Recall:

Measures how much of the reference text's content is captured by the candidate text (avoiding omissions).

$$R_{BERT} = \frac{1}{|x|} \sum_{x_i \in x} max\ sim(x_i, \hat{x}_j)$$

This averages the maximum similarity scores for each reference token. The greedy matching ensures each token is matched to its most similar counterpart, maximizing the overall similarity score (Mahmood et al., 2024).

F1-score:

The harmonic mean of precision and recall, providing a balanced measure of similarity.

$$F_{BERT} = 2\frac{P_{BERT} \cdot R_{BERT}}{P_{BERT} + R_{BERT}}$$

This provides a single, balanced measure of similarity.

Jaccard Similarity:

$$Jaccard\ Similarity\ (A, B) = \frac{|A \cap B|}{|A \cup B|}$$

where,

A is the generated response from the LLM.

B is the reference response or another text for comparison.

Jaccard Similarity gauges lexical overlap between two texts by dividing shared tokens ($|A \cap B|$) by total unique tokens ($|A \cup B|$). Ranging from 0 (no overlap) to 1 (identical), it assesses surface-level similarity, complementing BERTScore's focus on semantic meaning. Applied to evaluate stopword effects in LLM outputs, it ensures precise comparison of response content, enhancing reliability in text analysis (Mahmood et al., 2024).

Proposed Test Plan/Hypothesis

Test Plan 1: Evaluating Response Quality of Different Embedding and Large Language Models (LLMs) Combinations

The test covers two embedding models and three LLM models, resulting in six unique model combinations. Each combination will be tested against three predefined questions designed to probe the system's ability to comprehend document content and generate accurate responses.

Objective:

To assess the response quality of the Retrieval-Augmented Generation (RAG) system for question-answering tasks, focusing on the accuracy, relevance, and correctness of responses regarding the uploaded PDF Documents generated by different combinations of embedding models and Large Language Models (LLMs).

Scope:

The test focuses on question-answering tasks using content from uploaded PDF documents. It covers:

- Two embedding models for text representation, namely "**all-MiniLM-L6-v2**" and "**all-mpnet-base-v2**".
- Three LLMs for response generation, namely "**gemini-2.0-flash**", "**llama3-70b-8192**", and "**deepseek-chat-v3-0324**".
- Three PDF documents for information sources.
- Twelve predefined questions to evaluate comprehension and factual accuracy. This includes six close ended questions and six open ended questions.

Test Inputs:

<u>Information Sources:</u>

DOC 1: Application of Machine Learning Models for Early Detection and Accurate Classification of Type 2 Diabetes.pdf (Iparraguirre-Villanueva et al., 2023, #)

This doc is about using machine learning models to detect and classify type 2 diabetes in patients, evaluating five models (K-NN, BNB, DT, LR, SVM) on the Pima Indian dataset, with K-NN and BNB achieving the highest accuracies of 79.6% and 77.2% respectively.

DOC 2: hlb-smart-protect-vantage-insurance-pds-en.pdf (Hong Leong Assurance Berhad, n.d.)

This doc is about the Hong Leong SMART Protect Vantage Insurance, a regular premium investment-linked plan providing death coverage with benefits tied to unit fund performance.

DOC 3: CV_MOHAN (Alam, 2023, #)

This doc is about Aditya Mohan's academic and professional background, detailing his education, work experience in finance, research interests in disclosure and capital markets, and his job market paper on mutual fund manager responses to Morningstar ratings.

<u>Questions and Expected Answers:</u>

A total of twelve questions, including six close ended questions and six open ended questions.

Close Ended Questions

- **Question 1**: Is feature selection an important step in diabetes prediction using machine learning? (DOC 1)

  Expected Answer: Yes
- **Question 2**: Is the Pima Indian Diabetes dataset widely used in diabetes research? (DOC 1)

  Expected Answer: Yes
- **Question 3**: Is the Hong Leong SMART Protect Vantage Insurance an investment-linked plan? (DOC 2)

  Expected Answer: Yes
- **Question 4**: Are surrender and partial withdrawal charges applicable to this policy? (DOC 2)

  Expected Answer: No
- **Question 5**: Does Aditya Mohan have a PhD from Harvard Business School? (DOC 3)

  Expected Answer: No
- **Question 6**: Did Aditya Mohan receive the Eli Ginzberg memorial prize at Columbia Business School? (DOC 3)

  Expected Answer: Yes

Open Ended Questions

- **Question 1**: What is the role of machine learning in diabetes prediction? (DOC 1)

  Expected Answer: Machine learning helps analyze large datasets to identify patterns and build models that can predict the likelihood of diabetes in individuals.

- **Question 2**: What are some machine learning models used for diabetes prediction? (DOC 1)

  Expected Answer: Models like Decision Tree, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naïve Bayes, and Logistic Regression are commonly used.

- **Question 3**: What is the Hong Leong SMART Protect Vantage Insurance about? (DOC 2)

  Expected Answer: The Hong Leong SMART Protect Vantage Insurance is a regular premium investment-linked plan with level cover that provides insurance protection for death only. The policy values vary directly with the performance of the unit funds.

- **Question 4**: What are the key benefits provided by the basic plan of this insurance? (DOC 2)

  Expected Answer: The basic plan offers a Death Benefit, paying the higher of the Basic Sum Assured or the Account Value upon the Life Assured’s death, with reduced percentages for juveniles under age 5. It also provides a Maturity Benefit, paying the higher of the Basic Sum Assured or Account Value at the end of the policy term.

- **Question 5**: What is the focus of Aditya Mohan’s job market paper? (DOC 3)

  Expected Answer: Aditya Mohan’s job market paper, titled "Ratings based incentives and institutional investor response: Evidence from the mutual fund market," examines how Morningstar ratings create strong incentive effects for mutual fund portfolio managers. It shows that managers respond by attempting to improve risk-adjusted returns, driven by both investor flow considerations and career concerns.

- **Question 6**: Did Aditya Mohan receive the Eli Ginzberg memorial prize at Columbia Business School? (DOC 3)

  Expected Answer: Aditya Mohan’s research interests focus on the real effects of disclosure, institutional investors, and capital markets.

Test Criteria:

Trueness:

Accuracy is used to evaluate the trueness. A binary evaluation metric that determines whether the generated response correctly aligns with the expected answer, specifically for close ended questions. A response is marked as “True” if it accurately matches the expected answer and “False” otherwise. Then, the rate of correct response, accuracy, is then computed.

Text Similarity Measurement

Mean BERTScore is used to measure the similarity of the generated response and expected answer, specifically for open ended questions.

- **Precision**: Measures how much of the generated response aligns with the expected answer.
- **Recall**: Measures how much of the generated response is captured by the expected answer (avoiding omissions).
- **F1-Score**: The harmonic mean of precision and recall, providing a balanced measure of similarity.

Both trueness and text similarity measurement ensure the system’s ability to provide factually correct responses, critical for reliable document interaction.

Pass/Fail Criteria

- **Trueness**: At least 80% of responses must be marked as True (correctly matching the expected answer for closed-ended questions)
- **Text similarity measurement**: The average of all components of BERTScore must be at least 0.85 (indicating high semantic similarity between generated text and the expected text) (Amrelle, 2024).

Hypothesis

To formulate a hypothesis regarding the performance of a combined system integrating a sentence transformer and a large language model (LLM), two critical factors must be considered: the performance of the sentence transformer and the performance of the LLM.

The performance of the sentence transformer

The **all-MiniLM-L6-v2** and **all-mpnet-base-v2** are two popular sentence transformer models from the Sentence-BERT family, designed for generating high-quality sentence embeddings for tasks like semantic search, clustering, and text similarity. However, The **all-MiniLM-L6-v2** and **all-mpnet-base-v2** sentence transformers differ in architecture and performance.

all-MiniLM-L6-v2

all-MiniLM-L6-v2, based on the lightweight MiniLM with 6 layers and 384-dimensional embeddings, is fast, compact (~22 MB), and ideal for resource-constrained applications like chatbots or mobile apps, but it has lower accuracy for complex semantic tasks.

all-mpnet-base-v2

Conversely, **all-mpnet-base-v2**, built on the deeper MPNet with 12 layers and 768-dimensional embeddings, is larger (~110 MB) and more accurate, excelling in high-precision tasks like semantic search or document retrieval, though it's slower and resource-intensive. Both are trained on diverse datasets, but **all-mpnet-base-v2** outperforms on benchmarks like STS and MTEB.

Since **all-MiniLM-L6-v2** prioritizes speed and **all-mpnet-base-v2** prioritizes accuracy, it is expected that **all-mpnet-base-v2** has higher accuracy than **all-MiniLM-L6-v2**.

The performance of the LLM

In developing a Retrieval-Augmented Generation (RAG) system, the Instruction-Following Evaluation (IFEval) benchmark is prioritized to assess large language models (LLMs) on their ability to adhere to explicit, verifiable instructions, such as writing over 400 words or including the keyword "AI" at least three times (Zhou et al., 2023). The IFEval results indicate that

llama3-70b-8192 achieves a score of 92.1%, surpassing **deepseek-chat-v3-0324**, which scores 86.1% (docsbot.ai, n.d.). Due to the unavailability of IFEval results for **gemini-2.0-flash**, it is excluded from this analysis. Based on the available data, the ranking of models by IFEval performance is as follows: **llama3-70b-8192** ranks first, followed by **deepseek-chat-v3-0324**.

Since the output mainly depends on the input, the performance of the sentence transformer is prioritized. The final ranking is as follows: **llama3-70b-8192** with **all-MiniLM-L6-v2**, **llama3-70b-8192** with **all-mpnet-base-v2**, **deepseek-chat-v3-0324** with **all-MiniLM-L6-v2**, and **deepseek-chat-v3-0324** with **all-mpnet-base-v2**.

The above discussion is the first hypothesis which is about the ranking of performance of the combination. The second hypothesis is that all the combinations will pass the test.

Test Plan 2: Evaluate the Impact of Stopword Removal on LLM Response

In our test, we will use the same embedding models **(all-mpnet-base-v2)** across all experiments to ensure that any observed differences are due to the LLM's response behavior, as our evaluation focuses specifically on the impact of stop words on the language model rather than the embedding stage.

Objective

To assess the impact of stop words on the LLM response.

Scope:

The test focuses on question-answering tasks using content from uploaded PDF documents. It covers:

- One embedding models for text representation, namely "**all-MiniLM-L6-v2**"
- Three LLMs for response generation, namely "**gemini-2.0-flash**", "**llama3-70b-8192**", and "**deepseek-chat-v3-0324**".
- Three PDF documents for information sources.
- Six predefined open ended questions to evaluate comprehension and factual accuracy.

Test Inputs:

Information sources:

DOC 1: Application of Machine Learning Models for Early Detection and Accurate Classification of Type 2 Diabetes.pdf (Iparraguirre-Villanueva et al., 2023, #)

This doc is about using machine learning models to detect and classify type 2 diabetes in patients, evaluating five models (K-NN, BNB, DT, LR, SVM) on the Pima Indian dataset, with K-NN and BNB achieving the highest accuracies of 79.6% and 77.2% respectively.

DOC 2: hlb-smart-protect-vantage-insurance-pds-en.pdf (Hong Leong Assurance Berhad, n.d.)

This doc is about the Hong Leong SMART Protect Vantage Insurance, a regular premium investment-linked plan providing death coverage with benefits tied to unit fund performance.

DOC 3: CV_MOHAN (Alam, 2023, #)

This doc is about Aditya Mohan's academic and professional background, detailing his education, work experience in finance, research interests in disclosure and capital markets, and his job market paper on mutual fund manager responses to Morningstar ratings.

<u>Questions:</u>

A total of six open ended questions.

- **Question 1:** What were the top two machine learning models used for diabetes prediction, and what were their accuracies? (DOC 1)
- **Question 2:** Why was the SMOTE technique applied, and how did it affect the dataset balance? (DOC 1)
- **Question 3:** What benefits does the Hong Leong SMART Protect Vantage Insurance provide in case of death? (DOC 2)
- **Question 4:** Under what conditions might this policy lapse, and what should the policyholder do to prevent it?
- **Question 5:** What is Aditya Mohan’s job market paper about, and what key insight does it provide into mutual fund manager behavior? (DOC 3)
- **Question 6:** What were Aditya’s academic distinctions during his MBA at Columbia Business School? (DOC 3)

Test Criteria:

<u>Text similarity measurement</u>

Mean Jaccard Similarity Score is used to measure the similarity of the two generated responses with and without stop words.

Pass/Fail Criteria

Since the test is focused on observing the impact of stop words on the generated responses, there is no pass or fail criteria.

Hypothesis

Since stop words carry very little meaningful information, by removing it, the result will definitely change.

4. Results

The table below (Table 4.1) presents the model versions as indicated by their respective model names.

Name	Model Used
Gemini	gemini-2.0-flash
Llama	llama3-70b-8192
Deepseek	deepseek-chat-v3-0324

Table 4.1

Test 1: Evaluating the Response Quality of Different Embedding and Large Language Models (LLMs) Combinations

Yes/No Task Evaluation (Close Ended):

Sentence Transformer: all-MiniLM-L6-v2				
Question	**LLM**	**Expected Answer**	**Generated Answer**	**Trueness**
Question 1: Is feature selection an important step in diabetes prediction using machine learning? (DOC 1)	Gemini	Yes	Yes	✔
	Llama		Yes	✔
	Deepseek		Yes	✔
Question 2: Is the Pima Indian Diabetes dataset widely used in diabetes research? (DOC 1)	Gemini	Yes	Yes	✔
	Llama		Yes	✔
	Deepseek		Yes	✔
Question 3: Is the Hong Leong SMART Protect Vantage Insurance an investment-linked plan? (DOC 2)	Gemini	Yes	Yes	✔
	Llama		Yes	✔
	Deepseek		Yes	✔

Question 4: Are surrender and partial withdrawal charges applicable to this policy? (DOC 2)	Gemini	No	No	✔
	Llama		No	✔
	Deepseek		No	✔
Question 5: Does Aditya Mohan have a PhD from Harvard Business School? (DOC 3)	Gemini	No	Yes	✘
	Llama		No	✔
	Deepseek		Yes	✘
Question 6: Did Aditya Mohan receive the Eli Ginzberg memorial prize at Columbia Business School? (DOC 3)	Gemini	Yes	Yes	✔
	Llama		Yes	✔
	Deepseek		Yes	✔

Sentence Transformer: all-mpnet-base-v2				
Question	**LLM**	**Expected Answer**	**Generated Answer**	**Trueness**
Question 1: Is feature selection an important step in diabetes prediction using machine learning? (DOC 1)	Gemini	Yes	Yes	✔
	Llama		Yes	✔
	Deepseek		Yes	✔
Question 2: Is the Pima Indian Diabetes dataset widely used in diabetes research? (DOC 1)	Gemini	Yes	Yes	✔
	Llama		Yes	✔
	Deepseek		Yes	✔
Question 3: Is the Hong Leong SMART Protect Vantage Insurance an investment-linked plan? (DOC 2)	Gemini	Yes	Yes	✔
	Llama		Yes	✔
	Deepseek		Yes	✔
Question 4: Are surrender and partial withdrawal charges	Gemini	No	No	✔

applicable to this policy? (DOC 2)	Llama		No	✔
	Deepseek		No	✔
Question 5: Does Aditya Mohan have a PhD from Harvard Business School? (DOC 3)	Gemini	No	Yes	✘
	Llama		No	✔
	Deepseek		Yes	✘
Question 6: Did Aditya Mohan receive the Eli Ginzberg memorial prize at Columbia Business School? (DOC 3)	Gemini	Yes	Yes	✔
	Llama		Yes	✔
	Deepseek		Yes	✔

Summary Task Evaluation (Open Ended)

Sentence Transformer: all-MiniLM-L6-v2						
Question	**LLM**	**Expected Answer**	**Generated Answer**	**BERTScore**		
				Precision	**Recall**	**F1-score**
Question 1: What is the role of machine learning in diabetes prediction? (DOC 1)	Gemini	Machine learning helps analyze large datasets to identify patterns and build models that can predict the likelihood of diabetes in individuals.	Machine learning models are valuable tools that can help identify diabetes and manage this chronic disease. These models are based on information about a person's medical history, risk factors, and genetic makeup and are also used to analyze medical images.	**0.8878**	**0.9169**	**0.9021**
	Llama		Machine learning plays a crucial role in diabetes prediction by analyzing medical history, genetic makeup, and other risk factors to identify potential diabetic patients. Accurate prediction models can aid in early detection and	**0.8843**	**0.9166**	**0.9002**

			diagnosis, enabling timely interventions and improving healthcare outcomes.			
	Deepseek		Machine learning plays a crucial role in diabetes prediction by analyzing medical history, risk factors, and genetic data to identify early signs of diabetes, with models like SVM, random forest, and neural networks achieving high accuracy in classification. These models help improve early detection and decision-making in healthcare, as demonstrated in various studies using different ML algorithms.	**0.8689**	**0.9197**	**0.8936**
Question 2: What are some machine learning models used for diabetes	Gemini	Models like Decision Tree, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naïve Bayes, and	Machine learning models such as K-Nearest Neighbors (KNN), Bernoulli Naive Bayes (BNB), decision trees (DT), logistic regression (LR), support vector	**0.8804**	**0.9288**	**0.9040**

prediction? (DOC 1)		Logistic Regression are commonly used.	machines (SVM), and random forests (RF) can be used for diabetes prediction. Studies show promising results for early detection of diabetes using these models.			
	Llama		Some commonly used machine learning models for diabetes prediction include logistic regression (LR), Naive Bayes (NB), K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), and Multilayer Perceptron (MP).	**0.8800**	**0.9196**	**0.8993**
	Deepseek		Several machine learning models are commonly used for diabetes prediction, including Logistic Regression (LR), Support Vector Machines (SVM), K-Nearest Neighbors (K-NN), Decision Trees (DT), Random Forest (RF), Naive	**0.8745**	**0.9268**	**0.8999**

			Bayes (NB), and Multilayer Perceptron (MLP). Among these, K-NN and SVM have shown strong performance in accuracy for early detection.			
Question 3: What is the Hong Leong SMART Protect Vantage Insurance about? (DOC 2)	Gemini	The Hong Leong SMART Protect Vantage Insurance is a regular premium investment-linked plan with level cover that provides insurance protection for death only. The policy values vary directly with the performance of the unit funds.	Hong Leong SMART Protect Vantage Insurance is a regular premium investment-linked plan with level cover, providing insurance protection for death only, and its policy values vary with the performance of the unit funds. The insurance charges deducted from the account value will increase as you get older.	**0.9513**	**0.9726**	**0.9618**
	Llama		The Hong Leong SMART Protect Vantage Insurance is a regular premium investment-linked plan that provides death-only insurance protection, with policy values that vary directly with the performance of	**0.9803**	**0.9720**	**0.9761**

			the unit funds.			
	Deepseek		Hong Leong SMART Protect Vantage Insurance is a regular premium investment-linked plan offering level death coverage, where policy values fluctuate based on the performance of chosen unit funds. Note that insurance charges increase with age, and premiums are allocated partly to investments while the rest covers commissions and other costs.	**0.9150**	**0.9430**	**0.9288**
Question 4: What are the key benefits provided by the basic plan of this insurance? (DOC 2)	Gemini	The basic plan offers a Death Benefit, paying the higher of the Basic Sum Assured or the Account Value upon the Life Assured's death, with reduced percentages for juveniles under age 5. It also provides a Maturity	The basic plan of this insurance provides death-only protection, and you may be able to enjoy a guaranteed death benefit if the basic plan is in force. Note that insurance charges deducted from the account value will increase as you get older.	**0.8722**	**0.8466**	**0.8592**
	Llama		The basic plan provides a guaranteed	**0.8847**	**0.8441**	**0.8639**

		Benefit, paying the higher of the Basic Sum Assured or Account Value at the end of the policy term.	death benefit, insurance protection, and a cash value component that depends on the performance of the selected investment-linked funds.			
	Deepseek		The basic plan provides death benefit coverage (with a suicide clause for the first 12 months) and includes a 30-day grace period for premium payments. It also offers investment-linked growth potential, though cash values depend on fund performance.	**0.8659**	**0.8386**	**0.8520**
Question 5: What is the focus of Aditya Mohan's job market paper? (DOC 3)	Gemini	Aditya Mohan's job market paper, titled "Ratings based incentives and institutional investor response: Evidence from the mutual fund market," examines how Morningstar ratings create strong incentive effects	Aditya Mohan's job market paper focuses on how Morningstar ratings incentivize mutual fund portfolio managers, leading them to improve risk-adjusted returns, which is influenced by both investor flow considerations and managerial incentives.	**0.9579**	**0.9270**	**0.9422**

	Llama	for mutual fund portfolio managers. It shows that managers respond by attempting to improve risk-adjusted returns, driven by both investor flow considerations and career concerns.	The focus of Aditya Mohan’s job market paper is on ratings-based incentives and institutional investor response, specifically examining how Morningstar ratings affect mutual fund portfolio managers’ behavior.	**0.9409**	**0.9071**	**0.9237**
	Deepseek		Aditya Mohan’s job market paper focuses on how Morningstar ratings create incentives for mutual fund managers to improve risk-adjusted returns, examining both investor flow effects and managerial responses.	**0.9647**	**0.9256**	**0.9447**
Question 6: What are Aditya Mohan’s research interests? (DOC 3)	Gemini	Aditya Mohan’s research interests focus on the real effects of disclosure, institutional investors, and capital markets.	Aditya Mohan’s research interests include the real effects of disclosure, institutional investors, and capital markets. His job market paper focuses on ratings-based incentives and institutional investor response, particularly within the mutual fund market.	**0.9287**	**0.9775**	**0.9525**

	Llama		Aditya Mohan’s research interests are in Real effects of disclosure, Institutional investors, and Capital markets.	**0.9609**	**0.9551**	**0.9580**
	Deepseek		Aditya Mohan's research interests include the real effects of disclosure, institutional investors, and capital markets, with a focus on topics like ratings-based incentives in mutual funds.	**0.9359**	**0.9715**	**0.9534**

Sentence Transformer: all-MiniLM-L6-v2			
LLM	**Mean Precision**	**Mean Recall**	**Mean F1-score**
Gemini	0.9131	0.9282	0.9203
Llama	0.9219	0.9191	0.9202
Deepseek	0.9042	0.9209	0.9121

Sentence Transformer: all-mpnet-base-v2						
Question	**LLM**	**Expected Answer**	**Generated Answer**	**BERTScore**		
				Precision	**Recall**	**F1-score**
Question 1: What is the role of machine learning in diabetes prediction? (DOC 1)	Gemini	Machine learning helps analyze large datasets to identify patterns and build models that can predict the likelihood of diabetes in individuals.	Machine learning algorithms such as LR, Nave Bayes, KNN, SVM, DT, and random forest can be used to predict diabetes and to improve decision-making in healthcare facilities. Some models have reached a high accuracy, precision, recall, and F1-score.	**0.8390**	**0.9039**	**0.8702**
	Llama		Machine learning plays a crucial role in diabetes prediction by enabling the development of accurate models that can predict the likelihood of diabetes or detect the disease based on various factors, ultimately improving	**0.9030**	**0.9223**	**0.9126**

			decision-making in healthcare facilities.			
	Deepseek		Machine learning plays a crucial role in diabetes prediction by analyzing patient data (e.g., biomarkers, demographics) to identify patterns and predict disease risk with high accuracy. Common algorithms include logistic regression, SVM, random forests, and decision trees, achieving accuracies up to 99% in some studies.	**0.8627**	**0.9340**	**0.8969**
Question 2: What are some machine learning models used for diabetes prediction? (DOC 1)	Gemini	Models like Decision Tree, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naïve Bayes, and Logistic Regression are commonly used.	Several machine learning models are used for diabetes prediction, including K-Nearest Neighbors (KNN), Decision Trees (DT), Bayes Net Belief (BNB), Logistic Regression (LR), and Support Vector Machines (SVM). Research has also utilized Random Forest (RF) and Multilayer Perceptron (MP) models.	**0.8793**	**0.9291**	**0.9036**

	Llama		Some machine learning models used for diabetes prediction include K-NN, DT, BNB, LR, SVM, Random Forest (RF), and Neural Networks (NNs). Additionally, ensemble methods like multilayer perceptron (MP) and random tree (RT) have also been employed for this task.	**0.8637**	**0.8694**	**0.8665**
	Deepseek		Common machine learning models for diabetes prediction include logistic regression, decision trees, random forests, support vector machines, and neural networks. Some studies also use K-nearest neighbors, Naive Bayes, and gradient boosting techniques.	**0.8912**	**0.8870**	**0.8891**
Question 3: What is the Hong Leong SMART Protect Vantage	Gemini	The Hong Leong SMART Protect Vantage Insurance is a regular premium investment-linked plan with level cover that	Hong Leong SMART Protect Vantage Insurance is a regular premium investment-linked plan with level cover that provides insurance protection for death only, and its	**0.9851**	**0.9840**	**0.9845**

Insurance about? (DOC 2)		provides insurance protection for death only. The policy values vary directly with the performance of the unit funds.	policy values vary directly with the performance of the unit funds.			
	Llama		The Hong Leong SMART Protect Vantage Insurance is a regular premium investment-linked plan that provides death-only insurance protection with a Level Cover, where the policy values vary directly with the performance of the unit funds.	**0.9739**	**0.9756**	**0.9747**
	Deepseek		Hong Leong SMART Protect Vantage Insurance is a regular premium investment-linked plan offering death-only protection, where policy values fluctuate based on the performance of unit funds. The plan provides a death benefit equal to the higher of the basic sum assured (e.g., RM 500,000) or the account value, with insurance charges increasing as the policyholder ages.	**0.8956**	**0.9417**	**0.9181**

Question 4: What are the key benefits provided by the basic plan of this insurance? (DOC 2)	Gemini	The basic plan offers a Death Benefit, paying the higher of the Basic Sum Assured or the Account Value upon the Life Assured's death, with reduced percentages for juveniles under age 5. It also provides a Maturity Benefit, paying the higher of the Basic Sum Assured or Account Value at the end of the policy term.	The Basic Plan has a sum assured of RM 500,000.00 with coverage for 76 years or until termination, and in the event of death, the amount payable will be the higher of the basic sum assured or the account value at the next valuation date.	**0.8693**	**0.8769**	**0.8731**
	Llama		The key benefits provided by the basic plan of this insurance are a death benefit of RM 500,000.00 or the account value at the next valuation date, whichever is higher, and a coverage duration of 76 years or until policy termination.	**0.8656**	**0.8591**	**0.8624**
	Deepseek		The basic plan provides a sum assured of RM 500,000 for up to 76 years, with a death benefit that pays the higher of the basic sum assured or the account value at the next valuation date. It also includes a 30-day grace	**0.8744**	**0.8915**	**0.8829**

			period for premium payments and allows for nomination of beneficiaries.			
Question 5: What is the focus of Aditya Mohan's job market paper? (DOC 3)	Gemini	Aditya Mohan's job market paper, titled "Ratings based incentives and institutional investor response: Evidence from the mutual fund market," examines how Morningstar ratings create strong incentive effects for mutual fund portfolio managers. It shows that managers respond by attempting to improve risk-adjusted returns, driven by both investor flow considerations and career concerns.	Aditya Mohan's job market paper focuses on the incentive effects of Morningstar ratings on mutual fund portfolio managers and their efforts to improve risk-adjusted returns.	**0.9596**	**0.9098**	**0.9340**
	Llama		The focus of Aditya Mohan's job market paper is on ratings-based incentives and institutional investor responses, specifically how Morningstar ratings influence mutual fund portfolio managers' behavior.	**0.9394**	**0.9052**	**0.9220**
	Deepseek		The focus of Aditya Mohan's job market paper is not explicitly mentioned in the provided content. The documents primarily detail his educational background, work	**0.8621**	**0.8377**	**0.8498**

			experience, and professional qualifications.			
Question 6: What are Aditya Mohan's research interests? (DOC 3)	Gemini	Aditya Mohan's research interests focus on the real effects of disclosure, institutional investors, and capital markets.	Aditya Mohan's research interests include business administration, finance, and economics.	**0.9463**	**0.9186**	**0.9323**
	Llama		Aditya Mohan's research interests are not explicitly mentioned, but based on his education and experience, it can be inferred that his research interests likely relate to finance, economics, or investment strategy.	**0.8873**	**0.9074**	**0.8972**
	Deepseek		Aditya Mohan's research interests include business administration, finance, and economics, with a focus on asset allocation and investment strategies. His background in energy engineering also suggests potential interdisciplinary interests in energy economics or sustainable finance.	**0.8875**	**0.9186**	**0.9028**

Sentence Transformer: all-mpnet-base-v2			
LLM	**Mean Precision**	**Mean Recall**	**Mean F1-score**
Gemini	0.9131	0.9204	0.9163
Llama	0.9055	0.9065	0.9059
Deepseek	0.8789	0.9018	0.8899

Test 2: Evaluating the Impact of Stop Words on Large Language Models (LLMs)

Question	LLM	With Stopwords Output	Stopwords Removed Output	Jaccard Similarities
Question 1: What were the top two machine learning models used for diabetes prediction, and what were their accuracies? (DOC 1)	Gemini	The top two machine learning models used for diabetes prediction were K-NN(SMOTE) and BNB(SMOTE). K-NN(SMOTE) achieved the best performance with an accuracy of 79.6%, while BNB(SMOTE) had an accuracy of 77.2%.	Machine learning models such as K-Nearest Neighbors (KNN), Decision Trees (DT), Naive Bayes (BNB), Logistic Regression (LR), and Support Vector Machines (SVM) have been used for diabetes prediction. The KNN model with SMOTE achieved the best performance with 79.6% accuracy, while the BNB model with SMOTE reached 77.2% accuracy. Other models had varying accuracies, such as DT at 63%, LR with SMOTE at 72.7%, and SVM with SMOTE. In other studies, a DT model achieved a high accuracy rate of 0.9941, and an LR model reached 80% accuracy in predicting blood insulin	**0.29**

			levels.	
	Llama	The top two machine learning models used for diabetes prediction are K-NN with an accuracy of 79.6% and BNB with an accuracy of 77.2%.	Machine learning models have shown promising results in diabetes prediction, achieving accuracies ranging from 61.17% to 99.15%. The best-performing models include K-Nearest Neighbors (K-NN) with an accuracy of 79.6%, Binary Naive Bayes (BNB) with an accuracy of 77.2%, and Decision Trees (DT) with an accuracy of 63%. Other models, such as Support Vector Machines (SVM), have shown lower performances, with an accuracy of 61.17%. Some studies have also shown that machine learning models can achieve high accuracy, precision, recall, and F1-score, such as 0.9941, 0.99, and 99.11%, respectively.	**0.22**
	Deepseek	The top two machine learning models for diabetes prediction are the K-NN	Machine learning models for diabetes prediction show varying accuracies:	**0.19**

		(SMOTE) model with an accuracy of 79.6% and the BNB (SMOTE) model with an accuracy of 77.2%. These models outperformed other techniques in the study.	SVM ranges from 61.17% to 92.64%, DT models achieve up to 99.41%, K-NN reaches 79.6%, BNB 77.2%, LR 72.7%, and random forest (RF) also performs highly. Some studies report neural networks and LR as top performers, while others highlight SVM or DT. Accuracies depend on datasets, features, and preprocessing techniques like SMOTE.	
Question 2: Why was the SMOTE technique applied, and how did it affect the dataset balance? (DOC 1)	Gemini	The SMOTE technique was applied to address the imbalance in the dataset by increasing the number of samples belonging to the underrepresented class, thus increasing the efficiency of ML models. After applying SMOTE, the dataset was balanced to have 50% of people with diabetes and 50% without.	Yes, the SMOTE technique is useful for addressing imbalance problems in a dataset. It is a strategy that increases the number of samples belonging to the underrepresented class, balancing the dataset. For example, applying SMOTE can result in a dataset with 50% of the samples representing one class and 50% representing the other.	**0.33**
	Llama	The SMOTE technique was applied to	The SMOTE technique is indeed useful	**0.40**

		address imbalance problems in the dataset by increasing the number of samples belonging to the underrepresented class. This strategy aimed to increase the efficiency of machine learning models by generating synthetic samples of the minority class. As a result, the dataset balance was significantly improved, as shown in Figure 6, where the application of SMOTE achieved a balanced dataset with 50% of people with diabetes and 50% without diabetes.	for addressing imbalance problems in a dataset. Its strategy is to increase the efficiency of machine learning models by increasing the number of samples belonging to the underrepresented class. This is achieved by creating synthetic samples through a process that calculates the difference between a minority class sample and its K-Nearest Neighbors, multiplies it by a random number between 0 and 1, and adds it to the minority class sample. As a result, the dataset becomes more balanced, which can lead to more accurate predictions and improved performance of machine learning models.	
	Deepseek	SMOTE was applied to address the imbalance problem in the dataset by increasing the number of samples in the underrepresented class, which improved	SMOTE technique helps balance dataset by increasing samples of the minority class.	**0.12**

		the efficiency of machine learning models by ensuring a more balanced distribution of classes. The technique achieved this by generating synthetic samples for the minority class through calculating differences between existing samples and their nearest neighbors, multiplying these differences by random values, and adding them to the original samples, resulting in a dataset where both classes were equally represented, such as achieving a 50-50 split between classes in the diabetes dataset. This balanced dataset allowed machine learning models to perform better, as evidenced by improved accuracy and other performance metrics when trained on the SMOTE-augmented data compared to the original imbalanced dataset.		

Question 3: What benefits does the Hong Leong SMART Protect Vantage Insurance provide in case of death? (DOC 2)	Gemini	Hong Leong SMART Protect Vantage Insurance provides insurance protection that covers death. In the event of death of the Life Assured, the amount payable shall be the HIGHER of the Basic Sum Assured; or Account Value at the Next Valuation Date immediately following the date of notification of death.	Hong Leong SMART Protect Vantage Insurance provides insurance protection in the event of death. In the event of death of the Life Assured, the payable benefit shall be the higher of the Basic Sum Assured or the Account Value on the Valuation Date immediately following the date of notification of death. However, if the Life Assured commits suicide within twelve months of the Issue Date, Alteration Effective Date, or Reinstatement Date, the policy will be void, and the company will refund unallocated premiums and the value of units allocated.	**0.48**
	Llama	In the event of death, the Hong Leong SMART Protect Vantage Insurance provides a Death Benefit, which is the higher of the Basic Sum Assured (RM 500,000) or the Account Value at the	The Hong Leong SMART Protect Vantage Insurance provides a death benefit in the event of the death of the Life Assured. The death benefit payable is the higher of the Basic Sum Assured	**0.60**

		Next Valuation Date immediately following the date of notification of death.	(RM 500,000.00) or the Account Value as of the valuation immediately following the date of notification of death.	
	Deepseek	The Hong Leong SMART Protect Vantage Insurance provides a death benefit, which is paid out to the beneficiaries if the policyholder passes away, excluding cases of suicide within the first 12 months. The benefit amount depends on the policy's unit fund performance.	Hong Leong SMART Protect Vantage Insurance provides a death benefit where the payout is the higher of the basic sum assured or the account value as of the valuation date following the death notification. The policy offers level coverage with insurance protection linked to investment performance, and the death benefit is payable upon the death of the insured, subject to policy terms and exclusions.	**0.24**
Question 4: Under what conditions might this policy lapse, and what should the policyholder	Gemini	The policy will lapse/terminate when the value of investment units is insufficient to pay for the insurance and other charges. To prevent this, ensure timely premium payments, taking advantage of the 30-day grace period if needed, and	To prevent a policy from lapsing, ensure that you satisfy the policy's premium payments and that your investment units are sufficient to cover insurance charges. You are typically given a 30-day grace period for premium payments.	**0.35**

do to prevent it? (DOC 2)		consider topping up your premiums to increase the investment value.		
	Llama	This policy may lapse if the value of the investment units is insufficient to pay for the insurance and other charges. To prevent this, they should ensure that they can afford the premium payable under this plan and that the policy will best serve their needs. They should also pay premiums on time, utilizing the 30-day grace period if necessary.	A policyholder can prevent a policy lapse by paying the premium within the 30-day grace period provided. It is essential to ensure that the premium payable is affordable and aligned with the policyholder's needs to avoid policy lapse or termination.	**0.25**
	Deepseek	The policy may lapse if the investment unit value is insufficient to cover insurance and other charges, or if the life insured commits suicide within 12 months of policy issuance, alteration, or reinstatement. To prevent this, the policyholder should ensure timely premium payments within the 30-day grace period, maintain sufficient	To prevent a policy lapse, the policyholder should pay the premium within the 30-day grace period and ensure sufficient funds are available to cover insurance charges. It is also advisable to review the premium payment plan to align with affordability and maintain the policy's active status.	**0.33**

		investment unit value, and consider additional premium top-ups if needed.		
Question 5: What is Aditya Mohan's job market paper about, and what key insight does it provide into mutual fund manager behavior? (DOC 3)	Gemini	Aditya Mohan's job market paper is titled "Ratings based incentives and institutional investor response: Evidence from the mutual fund market." It finds that Morningstar ratings have strong incentive effects on mutual fund portfolio managers, who respond by trying to improve risk-adjusted returns. The paper indicates that both investor flow considerations and career concerns drive managers' responses to ratings.	Aditya Mohan's job market paper focuses on how Morningstar ratings create incentives for mutual fund portfolio managers. These managers respond to these incentives by trying to improve risk-adjusted returns, driven by investor flow considerations and career concerns.	**0.47**
	Llama	Aditya Mohan's job market paper, titled "Ratings-based incentives and institutional investor response: Evidence from the mutual fund market", explores how Morningstar ratings influence mutual fund managers' behavior. It reveals that managers respond to rating	According to Aditya Mohan's job market paper, a key insight is that they respond to ratings-based incentives by trying to improve their risk-adjusted returns, driven by investor flow considerations and career concerns.	**0.40**

		incentives by attempting to improve risk-adjusted returns, driven by both investor flow considerations and career concerns.		
	Deepseek	Aditya Mohan's job market paper examines how Morningstar ratings influence mutual fund manager behavior, showing that these ratings create strong incentives for managers to improve risk-adjusted returns due to both investor flow considerations and career concerns. The key insight is that ratings-based incentives significantly shape portfolio management decisions.	Aditya Mohan's job market paper reveals that mutual fund managers strongly respond to Morningstar ratings incentives, adjusting their portfolio strategies to improve risk-adjusted returns and attract investor flows.	**0.37**
Question 6: What were Aditya's academic distinctions during his MBA at Columbia	Gemini	During his MBA at Columbia Business School, Aditya Mohan achieved several academic distinctions, including the Eli Ginzberg memorial prize for outstanding work in economics and Dean's Honors with Distinction, awarded to the top 5%	Aditya Mohan has an MBA in Finance and Economics from Columbia Business School, where he was awarded Dean's Honors Distinction and the Eli Ginzberg Memorial Prize. He also has a B.Tech in Energy Engineering from the Indian	**0.28**

Business School? (DOC 3)		of the class. He also received the Michael Corasaniti Fellowship for Security Analysis and maintained a GPA of 4.09/4.0, placing him among the top students in his class.	Institute of Technology. Furthermore, he is expected to complete his Ph.D. in Business Administration from Harvard Business School in 2022.	
	Llama	Aditya's academic distinctions during his MBA at Columbia Business School include the Eli Ginzberg memorial prize for outstanding work in economics, a GPA of 4.09/4.0 (one of seven students in a class of 750 with a GPA greater than 4.0), and Deans Honors with Distinction (awarded to the top 5% of the class).	Aditya's academic distinctions in his MBA from Columbia Business School include the Eli Ginzberg Memorial Prize for outstanding work in economics and Deans Honors Distinction, which is awarded to only 5% of the class. He also had a GPA of 4.09/4.0, one of only seven students in a class of 750 students with a GPA greater than 4.0.	**0.73**
	Deepseek	Aditya Mohan earned his MBA from Columbia Business School with honors, including the Eli Ginzberg Memorial Prize for outstanding work in economics, a GPA of 4.09/4.0 (one of seven students in his class with a GPA above 4.0), and	Aditya earned his MBA from Columbia Business School with honors, including the Eli Ginzberg Memorial Prize for outstanding work in economics, a GPA of 4.09/4.0, Dean's Honors Distinction, and the Michael Corasaniti Fellowship in	**0.61**

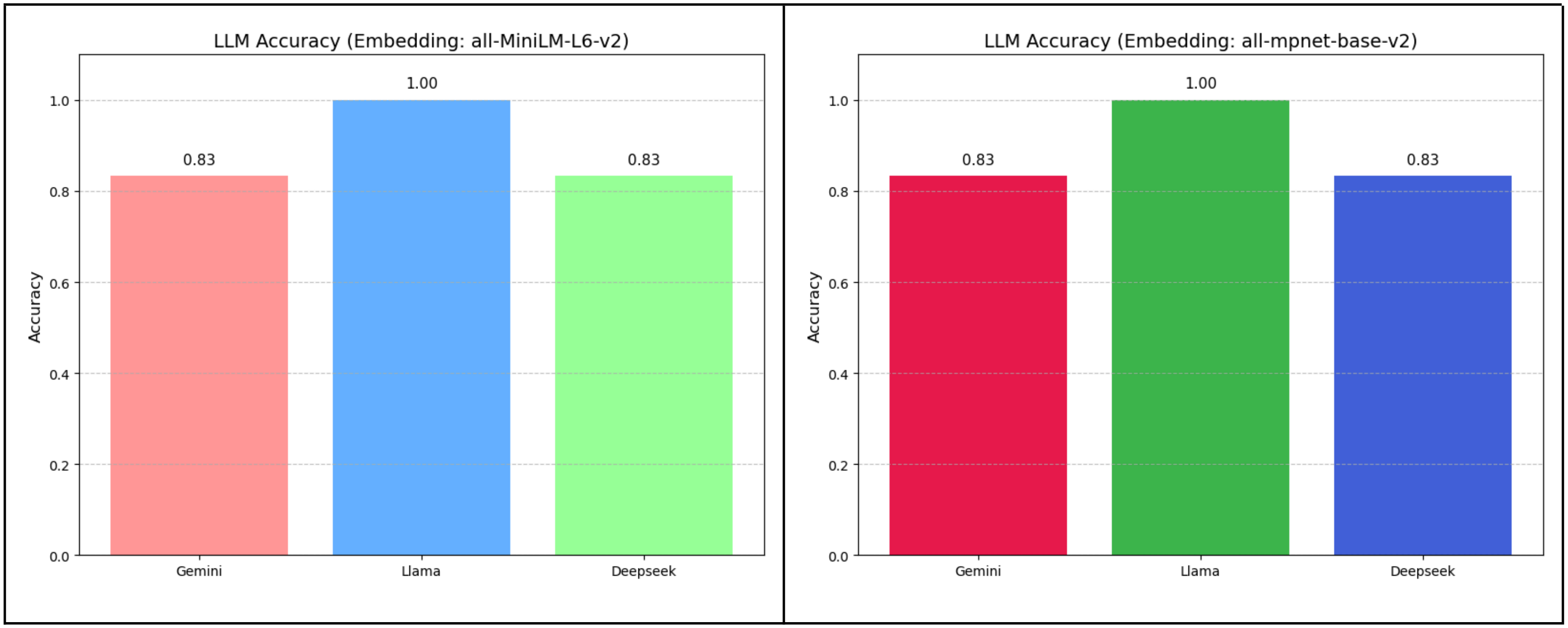
		Dean's Honor with Distinction (awarded to the top 5% of the class).	Security Analysis.	

Sentence Transformer: all-MiniLM-L6-v2	
LLM	**Mean Jaccard Similarity Score**
Gemini	0.3667
Llama	0.4333
Deepseek	0.3100

Results' Discussions

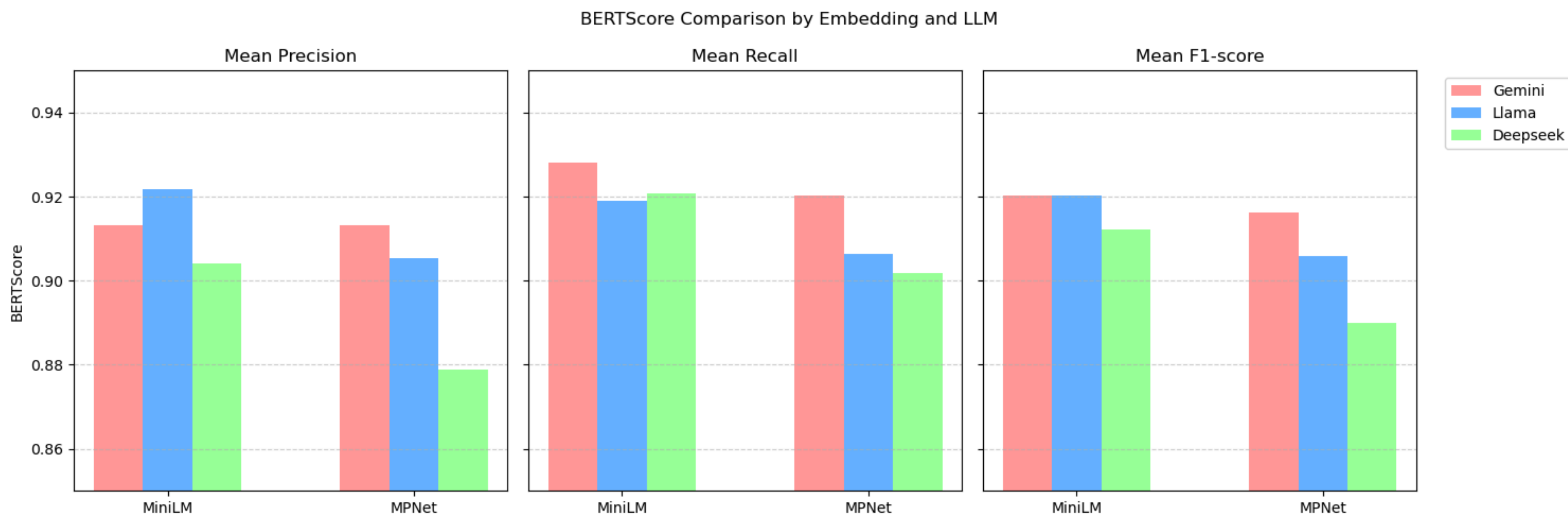
Results of the Response Quality of Different Embedding and Large Language Models (LLMs) Combinations

Yes/No Task Evaluation (Close Ended):



- Llama consistently outperformed the other LLMs in both embeddings, achieving 100% accuracy across all questions.
- Both Gemini and Deepseek scored 83%, each producing one incorrect answer (i.e., 5/6 correct) under both embeddings.
- The embedding model (MiniLM vs MPNet) had minimal impact on the final accuracy, **suggesting tha**t:
    - LLM performance is more influential than embedding choice in close-ended factual tasks.
    - Retrieval quality using either embedding was sufficient for supporting accurate Yes/No generation.

Summary Task Evaluation (Open Ended):

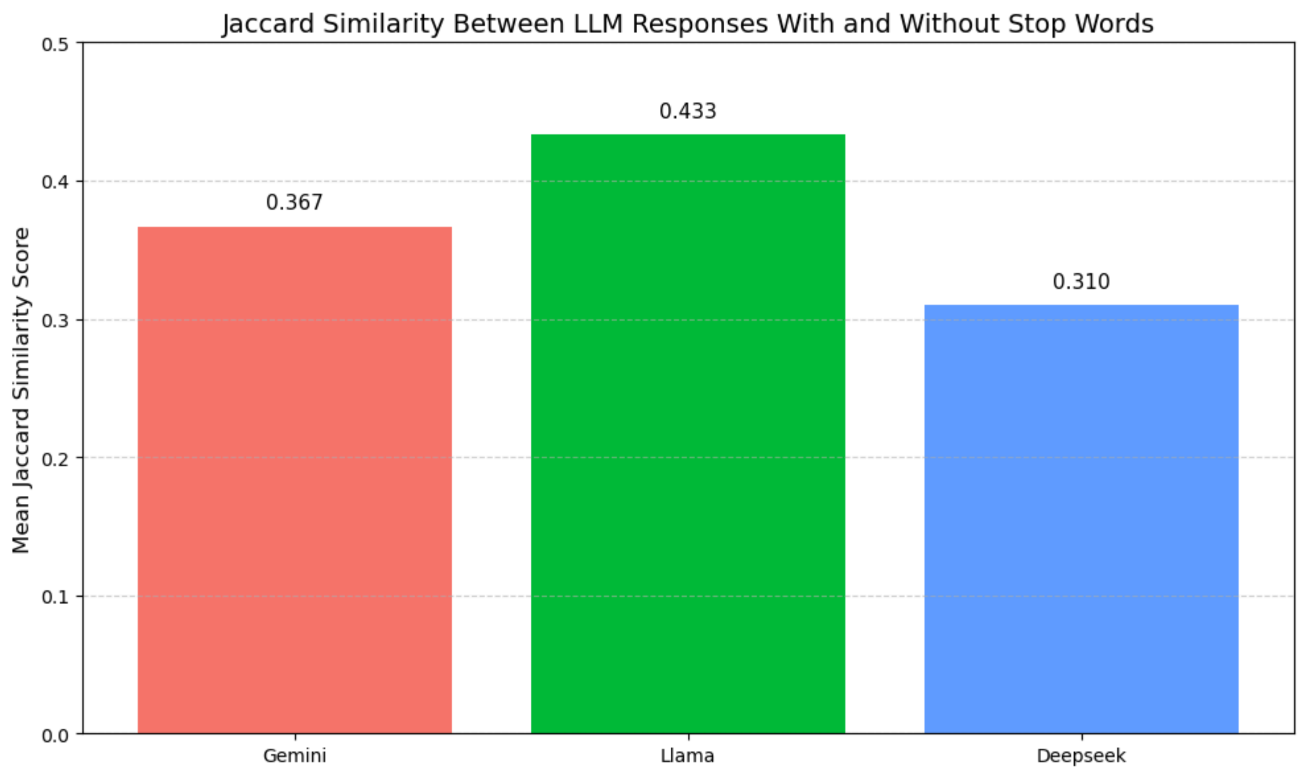


- Among the Embedding Models:
  - **MiniLM outperformed MPNet across all metrics**, suggesting it is more effective at retrieving semantically relevant content that aligns with expected answers.

- Among the LLMs:
  - **Gemini** achieved the **highest recall** in both embeddings and strong F1-scores, indicating it captures more relevant content from the retrieved context.
  - **Llama** had the **highest precision under MiniLM**, showing that its responses were more faithful to the expected wording, though recall slightly declined under MPNet.
  - **Deepseek** consistently scored lower in all metrics, especially under MPNet, highlighting potential weaknesses in semantic alignment or generalization under different retrieval contexts.

These results suggest that both the **retrieval quality (embedding)** and **LLM generation behavior** impact the final semantic quality of open-ended responses. **MiniLM paired with Gemini or Llama offers the most reliable performance** in generating responses that are both precise and complete.

Result of Stop Words on Large Language Models (LLMs)



- **Llama achieved the highest similarity score (0.433)**, suggesting that it produces **stable and consistent outputs** regardless of stop word presence. This indicates that its generation is more semantically grounded and less influenced by superficial lexical elements.
- **Gemini scored moderately (0.367)**, showing some variation in output with stop word removal, but still demonstrating reasonable robustness.
- **Deepseek had the lowest similarity score (0.310)**, implying a higher degree of fluctuation in its generated responses when stop words are removed. This may indicate greater reliance on the exact input structure or weaker generalization.

Models with higher Jaccard similarity (like Llama) are more desirable in applications where **preprocessing (e.g., stop word removal)** is involved, as they exhibit **greater output consistency**. The results reinforce that Llama is more resilient to input perturbations, while Deepseek may require more input stabilization to maintain reliable performance.

Conclusion

This study set out with two main hypotheses:

1. **Performance ranking hypothesis** – Combining a stronger LLM with the more accurate sentence‑embedding model (**all‑mpnet‑base‑v2**) would give the best overall results, producing the following predicted order:
   llama3‑70b‑8192 + mpnet
   - *llama3‑70b‑8192 + MiniLM*
   - *deepseek‑chat‑v3 + mpnet*
   - *deepseek‑chat‑v3 + MiniLM*
2. **Pass‑rate hypothesis** – Every LLM/embedding combination would satisfy the acceptance thresholds:
   - **Trueness**: ≥ 80 % accuracy on Yes/No questions
   - **Text‑similarity**: mean BERTScore ≥ 0.85 on open‑ended answers

Trueness (Accuracy)

Combination	Accuracy	Pass/Fail
llama + MiniLM	**100 %**	**Pass**
llama + mpnet	100 %	**Pass**
gemini + MiniLM	83 %	**Pass**
gemini + mpnet	83 %	**Pass**
deepseek + MiniLM	83 %	**Pass**
deepseek + mpnet	83 %	**Pass**

Hypothesis check:

All six combinations cleared the 80 % bar, so the *pass-rate hypothesis* holds. However, accuracy was driven almost entirely by the **LLM**, not the sentence transformer: mpnet did **not** improve trueness, contradicting the expectation that the deeper embedding would help here.

Open-Ended Similarity (BERTScore)

MiniLM embeddings consistently delivered higher mean BERTScores than mpnet across all three LLMs.

Combination	Precision	Recall	F1	Pass? (BERTScore ≥ 0.85)
Llama + MiniLM	**0.9219**	0.9191	**0.9202**	Pass
Llama + MPNet	0.9055	0.9065	0.9059	Pass
Gemini + MiniLM	0.9131	**0.9282**	**0.9203**	Pass
Gemini + MPNet	0.9131	0.9204	0.9163	Pass
Deepseek + MiniLM	0.9042	0.9209	0.9121	Pass
Deepseek + MPNet	0.8789	0.9018	0.8899	Pass

Hypothesis check:

All six configurations meet both thresholds (lowest F1 = 0.8899), so the pass-rate hypothesis is **confirmed**.

Stop‑Word Impact Test

LLM	Stop‑word Jaccard ↑	Stability Rank
Llama	**0.4333**	1 (Most stable)
Gemini	0.3667	2
Deepseek	0.3100	3 (Most affected)

Hypothesis check:

All three models produced different token sets once stop‑words were removed, confirming the hypothesis. Llama retained the most information (43 % overlap), while Deepseek was most sensitive (31 %). No numeric pass/fail threshold was set for this diagnostic test.

5. Discussion and Conclusion

Achievements

The project has successfully completed several core features for an Interactive Chat Application for Conversations with PDF Document Contents, incorporating a range of Natural Language Processing (NLP) tools and techniques. The following highlights the key achievements and an assessment of how well the project met the stated objectives:

1. **User Interaction**

- Users can upload PDF documents through the user interface. A dropdown menu in the sidebar allows selection of various LLM models. After a PDF is uploaded and a model is chosen, users can initiate question-answering sessions to retrieve information from the document's content. **MongoDB** and **Chroma DB** ensure seamless and dynamic interactions.

2. **Extract and Process PDF Content**

- The system employs **PyMuPDF** to accurately extract text, images, and metadata from diverse PDF structures. Extracted content is stored in **MongoDB** along with generated summaries, ensuring comprehensive processing.

3. **Conversational Interface for Real-Time Interaction**

- An UI facilitates instant question-answering sessions using selected LLMs (**gemini-2.0-flash, llama3-70b-8192, deepseek-chat-v3-0324**). Responses achieve **high trueness** (80%+) and strong **BERTScore** metrics, confirming accuracy and relevance.

4. **Document Preview and Highlighting**

- The application generates PDF summaries and retrieves the top three relevant chunks to guide users to pertinent content. Retrieval via **Chroma DB** provides infrastructure for highlighting relevant sections.

5. **View Similarity Between PDFs**
- Sentence Transformers (**all-MiniLM-L6-v2, all-mpnet-base-v2**) generate embeddings to enable content comparison through cosine similarity. **Chroma DB** supports semantic similarity analysis for comparing PDFs.

6. **Speech-to-Text and Text-to-Speech**
- Text-to-speech functionality is implemented to provide audio playback of PDF content.
- The system's design supports potential integration of speech-to-text capabilities.

7. **Factual and Accurate Responses**
- The RAG framework grounds responses in relevant document chunks, minimizing hallucinations.
- **High trueness** and **BERTScore** metrics validate the factual accuracy of responses.

8. **Feed LLMs Relevant Content**
- The system can retrieve relevant chunks from uploaded PDFs and feed them to LLMs instead of feeding the whole document. This enables a more efficient way to allow the LLMs to provide coherence and factual response that is relevant to the context of the user's query.

Evaluation of Objectives

Objective 1:

To assess the response quality of various embedding models (all-MiniLM-L6-v2, all-mpnet-base-v2) and LLMs (Gemini-2.0-flash, LLaMA 3.3 70B Versatile, DeepSeek Chat v3) in generating accurate and contextually relevant responses to questions based on PDF content.

Evaluation:

The project successfully evaluated response quality using BERTScore, Cosine Similarity, Precision, Recall, F1-score, and trueness. Test 1 results showed high trueness (80%+) and strong BERTScore F1-scores (e.g. 0.9618 for insurance questions), confirming accurate responses across all models. The **all-mpnet-base-v2** and **Gemini-2.0-flash** combination excelled in open-ended tasks, achieving superior semantic alignment with PDF content from documents like

Application of Machine Learning Models for Early Detection and Accurate Classification of Type 2 Diabetes. Pass/fail criteria and human evaluation validated contextual relevance, meeting the objective through comprehensive quantitative and qualitative assessments.

Objective 2:

To determine the effect of stopword removal on the quality and similarity of LLM-generated responses compared to responses with stopwords, using Jaccard Similarity.

Evaluation:

The project effectively assessed stopword impact using **Jaccard Similarity** to compare LLM outputs with and without stopwords. Results showed varied Jaccard Similarity scores (e.g. 0.73 for **LLaMA** on academic distinctions; 0.19 for **DeepSeek** on diabetes prediction), indicating stopwords influence lexical overlap, particularly in insurance and academic domains. **LLaMA** achieved the highest mean Jaccard Similarity (0.4333) with **all-MiniLM-L6-v2**, maintaining response quality post-stopword removal. NLTK's stopword processing and pass/fail criteria ensured contextual accuracy, especially for technical content like the SMOTE technique. This objective was achieved, providing clear insights into stopword effects on response quality.

Limitations and Future Works

Limitations

1. **Token Quota Limit**

   Token quotas restrict the amount of text the system can process or generate in a single interaction, limiting its ability to handle large documents or extended conversations. This can result in truncated context, where only part of a PDF or query is analyzed, leading to incomplete or less accurate responses. For users, this means breaking down documents or queries into smaller parts, which disrupts the seamless interaction intended by the system. Technically, token limits can truncate embeddings or retrieved document chunks, reducing response quality.

2. **Lack of Up-to-Date Online Information**

   The system depends on user-uploaded PDFs, limiting access to information beyond those documents. For example, an older machine learning PDF cannot provide recent insights without a newer upload, reducing utility in dynamic fields like technology or medicine. LLM-based chatbots struggle with up-to-date knowledge, as frequent retraining is costly and risks catastrophic forgetting (Dam et al., 2024), where new data causes loss of prior knowledge. Limited access to diverse, high-quality data sources further restricts external information integration. Unlike web-integrated models, the system cannot handle queries needing recent context, like new research, lowering response relevance for research or decision-making.

3. **Short Term Memory**

   The system struggles to retain context across multiple interactions or sessions, forcing users to re-upload documents or repeat questions if earlier context is lost. This creates redundant actions, frustrating users who expect a seamless, human-like conversational experience. While conversation details are stored, the system may not efficiently recall prior queries or document content in later sessions, leading to less coherent responses. This limitation affects the system's ability to maintain continuous context, especially for in-depth document analysis or multi-turn conversations.

4. **Not able to retrieve all relevant information if the content size is too big**

   For large documents, such as academic papers or technical manuals, critical information may be spread across multiple sections. The system splits PDFs into chunks and retrieves only the top few most relevant ones, risking the omission of important details. This can lead to incomplete or inaccurate responses, especially in complex or interdisciplinary documents. The reliance on embedding models and vector storage for retrieval may prioritize highly similar chunks while ignoring less obvious but relevant content. This limitation reduces the system's ability to provide comprehensive answers, particularly for dense or voluminous documents.

Future Works

1. **Dynamic Token Management**

   To overcome token quotas that truncate context and hinder large document processing, dynamic token management prioritizes critical content, such as abstracts, by summarizing or compressing less relevant sections. This ensures key information fits within LLM input limits to maintain accuracy. Leveraging existing chunking and LangChain's summarization tools (e.g. BERT-based) in the Python system, implementation is straightforward. Users enjoy seamless handling of large PDFs or conversations, with improved response quality and uninterrupted interaction, aligning with the goal of a user-friendly, efficient interface.

2. **Real-Time Web Search Integration**

   To address the reliance on static PDFs, which limits access to recent information in fields like medicine, real-time web search via APIs (e.g. PubMed, Google Scholar) fetches current studies or news to complement documents. This avoids costly LLM retraining and data source constraints. By extending the RAG pipeline with APIs, using LangChain for queries and MongoDB for caching, the Python system can easily adopt this. Users receive up-to-date, relevant responses, enhancing research and decision-making while matching web-integrated models for contextual accuracy.

3. **<u>Hierarchical Memory System</u>**

   To prevent short-term memory loss that forces users to repeat actions, a hierarchical memory system organizes MongoDB-stored contexts, prioritizing recent queries for quick retrieval with caching (e.g. Redis). This ensures coherence across multi-turn or cross-session interactions. Feasible with the system's existing MongoDB and LangChain context management, it integrates smoothly into the Python framework. Users experience a human-like conversational flow without re-uploading documents, supporting real-time interaction and improving usability for in-depth document analysis.

4. **<u>Multi-Stage Retrieval Algorithms</u>**

   To tackle incomplete retrieval in large documents, multi-stage retrieval algorithms fetch broad chunks via Chroma DB, then re-rank them based on global context (e.g. document structure). This captures all relevant content, ensuring comprehensive responses for complex PDFs. Feasible with Python libraries like sentence-transformers in the RAG framework, it leverages existing tools. Users benefit from accurate, thorough answers, aligning with the goal of precise extraction and enhancing reliability for analyzing dense or interdisciplinary documents.

Conclusion

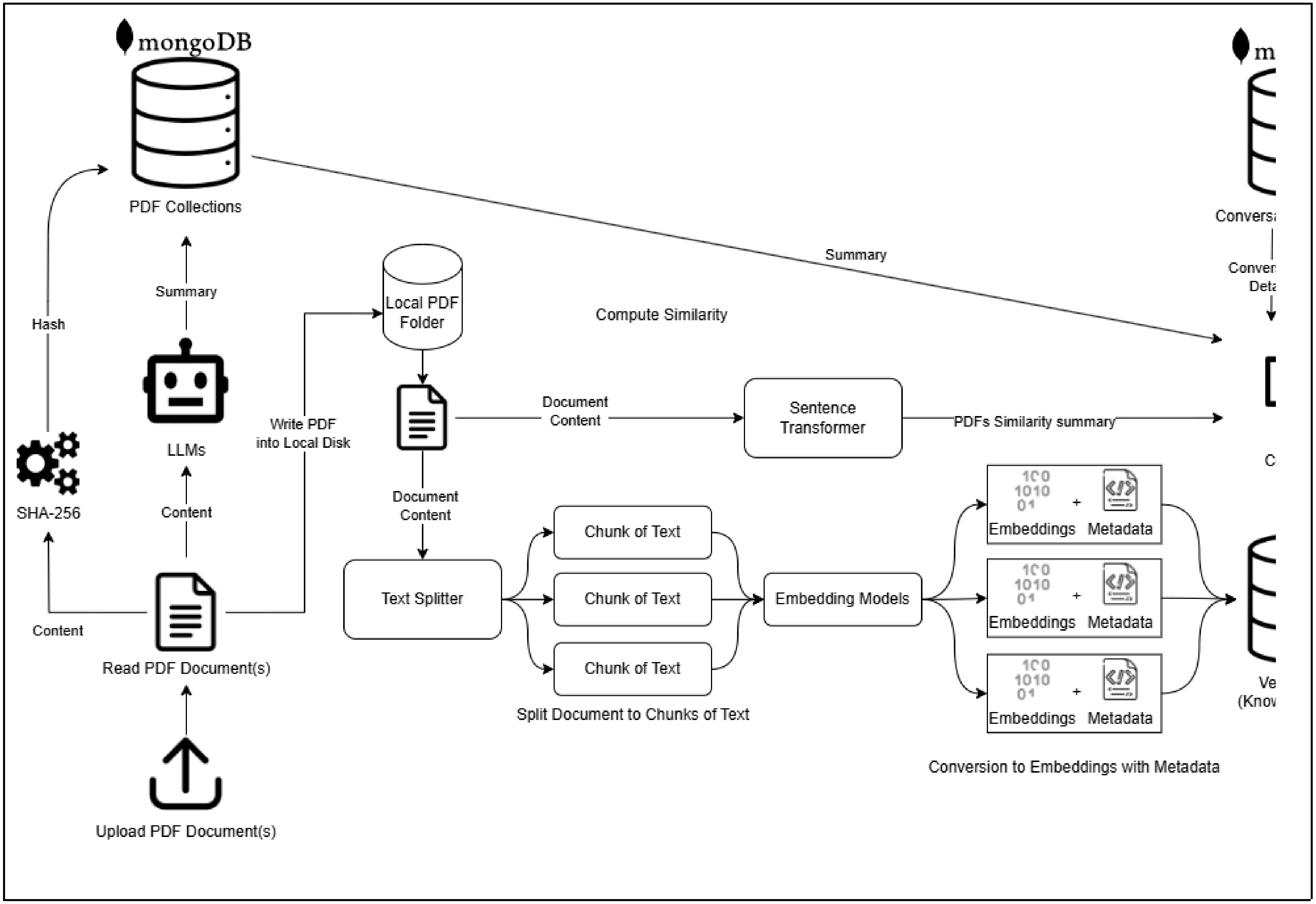
This project successfully developed an Interactive Chat Application for engaging with PDF document content, leveraging advanced Natural Language Processing (NLP) techniques and Large Language Models (LLMs) to transform static document consumption into dynamic, user-centric interaction. By integrating LLMs such as **Gemini-2.0-flash**, **LLaMA 3.3 70B Versatile**, and **DeepSeek Chat v3** with embedding models (**all-MiniLM-L6-v2**, **all-mpnet-base-v2**), the system achieved high response quality, with trueness scores exceeding 80% and BERTScore F1-scores up to 0.9618 for insurance-related queries. The Retrieval-Augmented Generation (RAG) framework, supported by **PyMuPDF** for text extraction and **Chroma DB** for vector storage, ensured accurate content retrieval and minimized hallucinations, addressing key challenges like factual inaccuracies outlined in the project's problem statement.

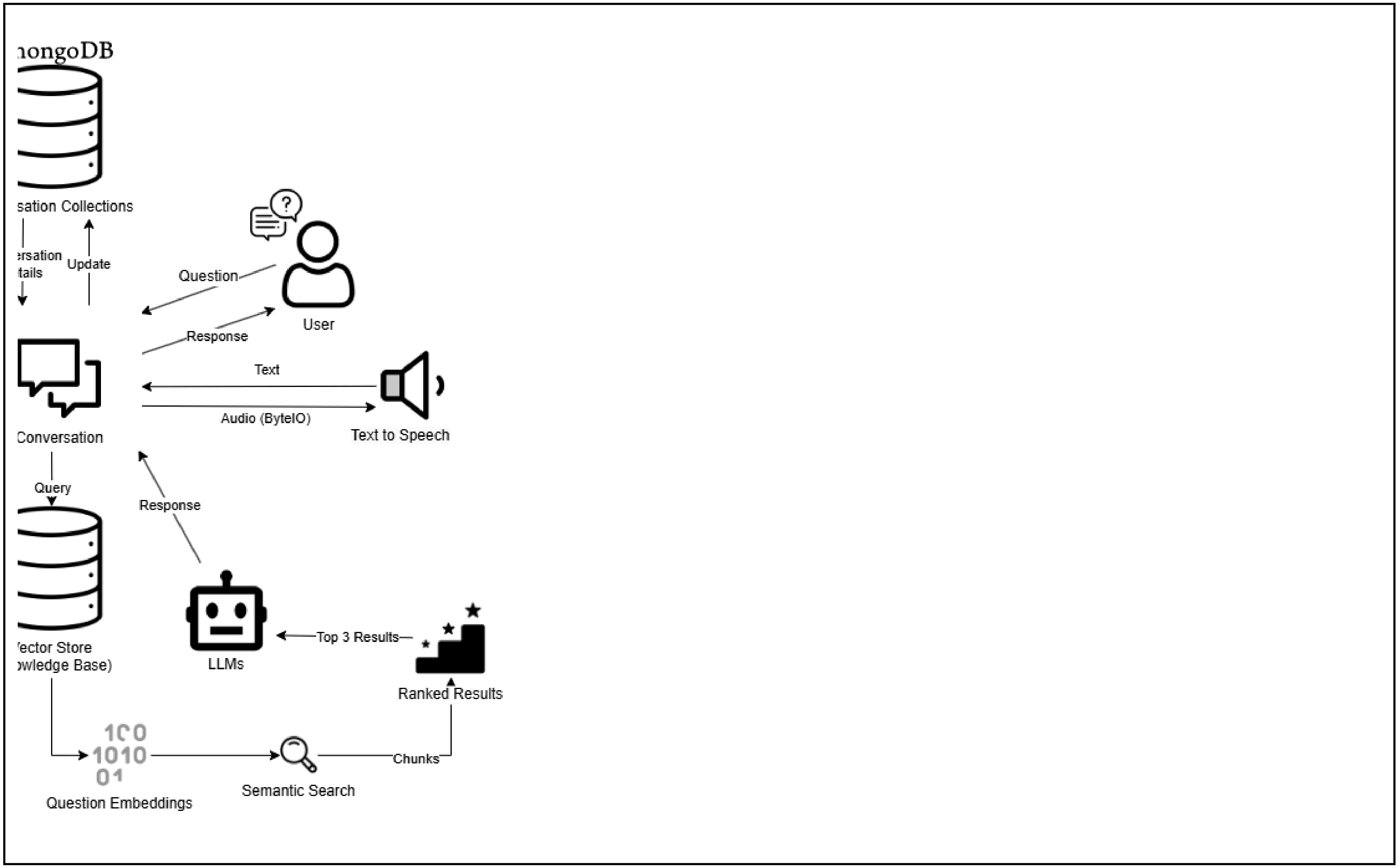
The application met its core objectives, including robust text extraction from diverse PDF structures, real-time question-answering, and document preview capabilities. Test results demonstrated LLaMA's superior performance in close-ended tasks (100% accuracy) and Gemini's strength in open-ended summarization, particularly when paired with **all-mpnet-base-v2**. The evaluation of stopword removal via **Jaccard Similarity** (e.g. LLaMA's 0.4333 score) provided insights into preprocessing impacts, enhancing response consistency across medical, insurance, and academic domains. Features like text-to-speech and similarity comparison between PDFs further enriched user experience, aligning with the goal of accessibility and efficiency.

Despite these achievements, limitations such as token quotas and lack of real-time web integration restricted handling of large documents and up-to-date information. Future work will focus on dynamic token management and web search integration to enhance scalability and relevance. This project underscores the potential of LLM-powered systems to streamline document analysis, offering a scalable solution for education, research, and professional settings, and paving the way for advanced conversational AI applications.

Appendix

Refer to the **next page** for the flowchart/system structure diagram, or access it via this **link**:
https://drive.google.com/file/d/1R8UI2ti4fWJlu7lk_Vyxj2LQMgVAPi5i/view?usp=sharing


mongoDB
PDF Collections
Summary
Hash
Summary
LLMs
SHA-256
Content
Content
Read PDF Document(s)
Upload PDF Document(s)
Write PDF
into Local Disk
Local PDF
Folder
Document
Content
Document
Content
Compute Similarity
Sentence
Transformer
PDFs Similarity summary
Text Splitter
Chunk of Text
Chunk of Text
Chunk of Text
Split Document to Chunks of Text
Embedding Models
Embeddings
Metadata
Embeddings
Metadata
Embeddings
Metadata
Conversion to Embeddings with Metadata



ıongoDB
sation Collections
ersation
tails
Update
Question
User
Response
Text
Audio (ByteIO)
Text to Speech
Conversation
Query
Response
'ector Store
owledge Base)
LLMs
Top 3 Results
Ranked Results
Question Embeddings
Semantic Search
Chunks

References

1. Adobe Systems Incorporated. (2008, July 1). *Document management — Portable document format — Part 1: PDF 1.7*. Adobe Open Source. Retrieved May 3, 2025, from https://opensource.adobe.com/dc-acrobat-sdk-docs/pdfstandards/PDF32000_2008.pdf
2. Ahmed, A., Joorabchi, A., & Hayes, M. J. (2022, June Nil). *On the Application of Sentence Transformers toAutomatic Short Answer Grading in BlendedAssessment*. On the Application of Sentence Transformers toAutomatic Short Answer Grading in BlendedAssessment. Retrieved May 4, 2025, from https://www.researchgate.net/publication/362965124_On_the_Application_of_Sentence_Transformers_to_Automatic_Short_Answer_Grading_in_Blended_Assessment
3. Alam, A. (2023, August). What is Machine Learning? http://dx.doi.org/10.5281/zenodo.8231580
4. Altman, S. (2022, November 30). *Introducing ChatGPT*. OpenAI. Retrieved May 1, 2025, from https://openai.com/index/chatgpt/
5. Alwahaishi, S., & MEROUANE, E. (2022, July Nil). *(PDF) Chatbots Development Using Natural Language Processing: A Review*. ResearchGate. Retrieved May 3, 2025, from https://www.researchgate.net/publication/367369151_Chatbots_Development_Using_Natural_Language_Processing_A_Review#fullTextFileContent
6. Amodei, D. (2021, Nil Nil). *Meet Claude*. Anthropic. Retrieved May 1, 2025, from https://www.anthropic.com/claude
7. Amrelle, K. (2024, May 15). *Guide to Metrics and Thresholds for Evaluating RAG and LLM Models*. LinkedIn. Retrieved May 4, 2025, from https://www.linkedin.com/pulse/guide-metrics-thresholds-evaluating-rag-llm-models-kevin-amrelle-dswje
8. B, P., J.M, N., & T, G. (2021, October Nil). *An Analysis of the Applications of NaturalLanguage Processing in Various Sectors*. An Analysis of the Applications of NaturalLanguage Processing in Various Sectors. Retrieved May 1, 2025, from http://dx.doi.org/10.3233/APC210109

9. Bird, S., & Loper, E. (2024, August 19). *NLTK :: Natural Language Toolkit*. NLTK :: Natural Language Toolkit. Retrieved May 3, 2025, from https://www.nltk.org/
10. Chakkarwar, V., Tamane, S., & Thombre, A. (2023, May 1). *A Review on BERT and Its Implementation in Various NLP Tasks*. A Review on BERT and Its Implementation in Various NLP Tasks. Retrieved May 4, 2025, from https://doi.org/10.2991/978-94-6463-136-4_12
11. Chang, M.-. W. (2019, May 24). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Retrieved May 4, 2025, from https://doi.org/10.48550/arXiv.1810.04805
12. Chauhan, A. (2019, May 5). *A Review on Various Aspects of MongoDb Databases*. A Review on Various Aspects of MongoDb Databases. Retrieved May 4, 2025, from https://www.ijert.org/research/a-review-on-various-aspects-of-mongodb-databases-IJERTV8IS050031.pdf
13. *Chroma DB Introduction*. (2023, April 6). Chroma DB Introduction. Retrieved May 4, 2025, from https://docs.trychroma.com/docs/overview/introduction
14. Colangelo, M. T. (2025, March 13). *A Comparative Analysis of Sentence Transformer Models for Automated Journal Recommendation Using PubMed Metadata*. MDPI. Retrieved May 4, 2025, from https://www.mdpi.com/2504-2289/9/3/67
15. Dam, S. K. (2024, June 18). *A Complete Survey on LLM-based AI Chatbots*. A Complete Survey on LLM-based AI Chatbots. Retrieved May 1, 2025, from https://doi.org/10.48550/arXiv.2406.16937
16. Dam, S. K., Hong, C. S., Qiao, Y., & Zhang, C. (2024, June Nil). *A Complete Survey on LLM-based AI Chatbots*. A Complete Survey on LLM-based AI Chatbots. Retrieved May 4, 2025, from http://dx.doi.org/10.13140/RG.2.2.19901.27368
17. *DeepSeek-V3-0324 Release*. (2025, March 25). DeepSeek API Docs. Retrieved May 4, 2025, from https://api-docs.deepseek.com/news/news250325
18. docsbot.ai. (n.d.). *Gemini 2.0 Flash vs DeepSeek-V3*. docsbot.ai. Retrieved May, 2025, from https://docsbot.ai/models/compare/gemini-2-0-flash/deepseek-v3

19. Fenniak, M. (2012, February 8). *PyPDF2*. PyPDF2. Retrieved May 3, 2025, from https://pypi.org/project/PyPDF2/
20. Fenniak, M. (2012, February 8). *PyPDF2*. Welcome to PyPDF2 — PyPDF2 documentation. Retrieved May 3, 2025, from https://pypdf2.readthedocs.io/en/3.x/
21. Gambhir, P. (2019, December Nil). *REVIEW OF CHATBOT DESIGN ANDTRENDS*. REVIEW OF CHATBOT DESIGN ANDTRENDS. Retrieved May 3, 2019, from https://www.researchgate.net/publication/337927323_REVIEW_OF_CHATBOT_DESIGN_AND_TRENDS
22. *Gemini 2.0 Flash | Generative AI on Vertex AI*. (2025, May 2). Google Cloud. Retrieved May 4, 2025, from https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-0-flash
23. Google. (n.d.). *What is Retrieval-Augmented Generation (RAG)?* Google Cloud. Retrieved May 4, 2025, from https://cloud.google.com/use-cases/retrieval-augmented-generation?hl=en
24. guide, s., & Selvaraj, N. (2025, March 14). *What is Retrieval Augmented Generation (RAG)?* DataCamp. Retrieved May 4, 2025, from https://www.datacamp.com/blog/what-is-retrieval-augmented-generation-rag
25. Haristiani, N. (2019, Nil Nil). *Artificial Intelligence (AI) Chatbot as Language Learning Medium: An inquiry*. Artificial Intelligence (AI) Chatbot as Language Learning Medium: An inquiry. Retrieved May 1, 2025, from https://iopscience.iop.org/article/10.1088/1742-6596/1387/1/012020/pdf
26. Hassabis, D. (2023, May 10). *Gemini*. Google DeepMind. Retrieved May 1, 2025, from https://deepmind.google/technologies/gemini/
27. Hong Leong Assurance Berhad. (n.d.). *This Product Illustration consists of 5 pages and each page forms an integral part of the Product Illustration. A prospective p*. Hong Leong Bank. Retrieved May 5, 2025, from https://www.hlb.com.my/content/dam/hlb/my/docs/pdf/Personal/Insurance/smart-protect-vantage-insurance/hlb-smart-protect-vantage-insurance-pds-en.pdf

28. IJSRD Journal. (2015, October Nil). *A Study on Mongodb Database*. A Study on Mongodb Database. Retrieved May 4, 2025, from https://www.academia.edu/20050112/A_Study_on_Mongodb_Database
29. 🚀 *Introducing DeepSeek-V3*. (2024, December 26). DeepSeek API Docs. Retrieved May 4, 2025, from https://api-docs.deepseek.com/news/news1226
30. Iparraguirre-Villanueva, O., Espinola-Linares, K., Castañeda Rosalynn Ornella Flores Castañeda, R. O. F., & Cabanillas-Carbonell, M. (2023). Application of Machine Learning Models for Early Detection and Accurate Classification of Type 2 Diabetes. *Diagnostics*, *13*(14), 2383. https://doi.org/10.3390/diagnostics13142383
31. *Llama-3.3-70B-Versatile*. (2024, December 6). Llama-3.3-70B-Versatile - GroqDocs. Retrieved May 4, 2025, from https://console.groq.com/docs/model/llama-3.3-70b-versatile
32. Louis, A., Dijck, G. v., & Spanakis, G. (2024, March 24). *Interpretable Long-Form Legal Question Answering with Retrieval-Augmented Large Language Models | Proceedings of the AAAI Conference on Artificial Intelligence*. AAAI Publications. Retrieved May 1, 2025, from https://doi.org/10.1609/aaai.v38i20.30232
33. Mahmood, M., Khan, F., & guide, S. (2024, May 16). *BERT Score Explained | Medium*. Ruman. Retrieved May 4, 2025, from https://rumn.medium.com/bert-score-explained-8f384d37bb06
34. McKie, J. X. (2019, July 28). *Products | Artifex*. Artifex Software Inc. Retrieved May 3, 2025, from https://artifex.com/products#pymupdf
35. McKie, J. X. (2019, July 28). *PyMuPDF 1.25.5 documentation*. PyMuPDF 1.25.5 documentation. Retrieved May 3, 2025, from https://pymupdf.readthedocs.io/en/latest/index.html
36. MENGHANI, G. (2021, June 21). *Efficient Deep Learning: A Survey on Making Deep Learning Models Smaller, Faster, and Better*. Efficient Deep Learning: A Survey on Making Deep Learning Models Smaller, Faster, and Better. Retrieved May 1, 2025, from https://arxiv.org/pdf/2106.08962
37. Mousavi, S. M. (2024, April 10). *Is Your LLM Outdated? Benchmarking LLMs & Alignment Algorithms for Time-Sensitive Knowledge*. Is Your LLM Outdated?

Benchmarking LLMs & Alignment Algorithms for Time-Sensitive Knowledge. https://arxiv.org/html/2404.08700v1

38. Ofoeda, J., Boateng, R., & Effah, o. (2019, July Nil). *Application ProgrammingInterface (API) Research:A Review of the Past to Inform the Future*. Application ProgrammingInterface (API) Research:A Review of the Past to Inform the Future. Retrieved May 3, 2025, from https://www.researchgate.net/publication/334145268_Application_Programming_Interface_API_Research_A_Review_of_the_Past_to_Inform_the_Future
39. *pdfminer.six*. (2014, 09 15). pdfminer.six. Retrieved May 3, 2025, from https://pdfminersix.readthedocs.io/en/latest/faq.html
40. Pires, J. G. (2024, January Nil). *Data Science Using OpenAI: Testing Their New Capabilities Focused on Data Science*. Data Science Using OpenAI: Testing Their New Capabilities Focused on Data Science. Retrieved May 3, 2025, from https://www.researchgate.net/publication/377794084_Data_Science_Using_OpenAI_Testing_Their_New_Capabilities_Focused_on_Data_Science
41. Roy, S., Goswami, M., Nargund, N., Mohanty, S., & Pattnaik, P. K. (2025, January 16). *Conversational Text Extraction with Large Language Models Using Retrieval-Augmented Systems*. Conversational Text Extraction with Large Language Models Using Retrieval-Augmented Systems. Retrieved May 1, 2025, from https://arxiv.org/pdf/2501.09801
42. Sharma, S. K., & Sharma, M. (2023, April Nil). *Chatbots in Educational Institutions: Challenges and Issues*. Chatbots in Educational Institutions: Challenges and Issues. Retrieved May 1, 2025, from https://www.ijcaonline.org/archives/volume185/number3/sharma-2023-ijca-922680.pdf
43. Shinyama, Y. (2019, November 7). *Welcome to pdfminer.six's documentation! — pdfminer.six 20250417.dev3+gb0bc964 documentation*. Welcome to pdfminer.six's documentation! — pdfminer.six 20250417.dev3+gb0bc964 documentation. Retrieved May 3, 2025, from https://pdfminersix.readthedocs.io/en/latest/

44. Simhi, A. (2025, February 18). *Distinguishing Ignorance from Error in LLM Hallucinations*. Distinguishing Ignorance from Error in LLM Hallucinations. Retrieved May 5, 2025, from https://doi.org/10.48550/arXiv.2410.22071
45. Tamrakar,, R., & Wani, N. (2021, April Nil). *Design and Development of CHATBOT: A Review*. Design and Development of CHATBOT: A Review. Retrieved May 3, 2025, from https://www.researchgate.net/publication/351228837_Design_and_Development_of_CHATBOT_A_Review
46. Taneja, K., & K. Goel, A. (2025, February 14). *MuDoC:AnInteractive Multimodal Document-grounded Conversational AI System*. MuDoC:AnInteractive Multimodal Document-grounded Conversational AI System. Retrieved May 1, 2025, from https://arxiv.org/pdf/2502.09843
47. Tin Tin, T., Yu Xuan, S., Man Ee, W., Kuok Tiung, L., & Aitizaz, A. (2024, Nil Nil). *Interactive ChatBot for PDF Content Conversation Using an LLM Language Model*. Interactive ChatBot for PDF Content Conversation Using an LLM Language Model. Retrieved May 1, 2025, from http://dx.doi.org/10.14569/IJACSA.2024.01509105
48. Topsakal, O., & Akinci, T.C. (2023, July 22). *Creating Large Language Model Applications Utilizing LangChain: A Primer on Developing LLM Apps Fast*. Creating Large Language Model Applications Utilizing LangChain: A Primer on Developing LLM Apps Fast. Retrieved May 3, 2023, from https://doi.org/10.59287/icaens.1127
49. Vijayaraghavan, J., & Crabtree, J. S. (n.d.). DIABETES. https://www.medschool.lsuhsc.edu/genetics/docs/DIABETES.pdf
50. Vine, J. S. (2016, February 23). *pdfplumber · PyPI*. PyPI. Retrieved May 3, 2025, from https://pypi.org/project/pdfplumber/#data
51. Wenfeng, L. (2023, July 17). *DeepSeek*. DeepSeek. Retrieved May 1, 2025, from https://www.deepseek.com/en
52. Xu, L., Lu, L., Liu, M., Song, C., & Wu, L. (2024, April 9). *Nanjing Yunjin intelligent question-answering system based on knowledge graphs and retrieval augmented generation technology*. Nanjing Yunjin intelligent question-answering system based on

knowledge graphs and retrieval augmented generation technology. Retrieved May 1, 2025, from https://doi.org/10.1186/s40494-024-01231-3

53. Zhang, X., Chen, Y., Hu, L., & Wang, Y. (2022, October 10). *The metaverse in education: Definition, framework, features, potential applications, challenges, and future research topics*. Frontiers. Retrieved May 1, 2025, from https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2022.1016300/full
54. Zhou, J., Lu, T., Mishra, S., Brahma, S., Basu, S., Luan, Y., Zhou, D., & Hou, L. (2023). Instruction-Following Evaluation for Large Language Models. *Computation and Language*. https://arxiv.org/abs/2311.07911
55. Meta Platforms, Inc., M. (2023, February 24). *Llama*. Llama. Retrieved May 5, 2025, from https://www.llama.com/