ISLAMIC UNIVERSITY OF TECHNOLOGY

Development of an Investment Proposal Processing System in Banks using LLM.

By
Md. Nazmus Sadat Nehan (191041009)

A project submitted in partial fulfilment of the requirements for the degree of M.Eng. in Computer Science and Engineering

Academic Year: 2019-2020

Department of Computer Science and Engineering Islamic University of Technology (IUT)
A Subsidiary Organ of the Organization of Islamic Cooperation.
Dhaka, Bangladesh.

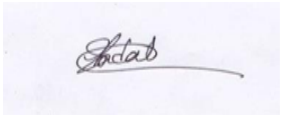
May 2024

Declaration of Authorship

I, Md. Nazmus Sadat Nehan, declare that this project titled, 'Investment Proposal Processing System in Banks' and the work presented in it is my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Any part of this thesis has not been submitted for any other degree or qualification at this University or any other institution.
- Where I have consulted the published work of others, this is always clearly attributed.

Submitted By:



(Signature of the Candidate)

Md. Nazmus Sadat Nehan -
191041009
May-2024

Development of an Investment Proposal Processing System in Banks using LLM.

Approved By:

Dr. Md. Kamrul Hasan
Project Supervisor,
Professor,
Department of Computer Science and Engineering,
Islamic University of Technology.

Dr. Md. Hasanul Kabir
Professor & Head,
Department of Computer Science and Engineering,
Islamic University of Technology.

Dr. Hasan Mahmud
Associate Professor,
Department of Computer Science and Engineering,
Islamic University of Technology.

Lt.Col.Muhammad Nazrul Islam,PhD
Associate Professor
Department of Computer Science and Engineering
Military Institue of Science and Technology.

Abstract

In the realm of modern banking, the efficient processing of investment proposals stands as a pivotal aspect for both financial institutions and their clientele. This project introduces a new approach leveraging Language Model (LLM) technology to streamline and enhance the investment proposal processing system within banks. This project introduces a new way of Investment Proposal Processing System for banks, empowered by Language Model (LLM) technology. Leveraging natural language processing capabilities, the system automates the evaluation of investment proposals, enhancing efficiency and decision-making accuracy. By analyzing textual data, including financial documents and market trends, the system provides comprehensive risk assessments and facilitates transparent decision outcomes. Through its implementation, banks stand to benefit from improved processing times, enhanced risk management, and greater client satisfaction, heralding a new era of data-driven investment operations.

Keyword – Large Language Model; Machine Learning; LangChain; RAG; FLAN-T5

Acknowledgements

All Praise and thanks goes to the most High Allah Subhanu Wata'ala for giving me strength and capabilities to complete this study. I am grateful and highly appreciative to Dr. Kamrul Hasan for the tremendous support he has given me not just as a supervisor but also as mentor whom I look up to, appreciation to Dr. Hasan Mahmudand for his constant motivation and support throughout this study.

Contents

List of Figures

List of Tables

Dedicated to my parents, spouse, and son for their continuous encouragement of my academic endeavors and research . . .

Chapter 1

Introduction

In any country there are many financial institutions, but banks are the most important one. Banks play a pivotal role in the economic infrastructure of any country, serving as the cornerstone of financial intermediation, resource allocation, and economic development. In every nation, banks act as custodians of capital, facilitating the flow of funds between savers and borrowers, enabling individuals, businesses, and governments to access the financial resources necessary for growth, investment, and prosperity.

In addition to their primary function of accepting deposits and extending credit, banks offer a wide array of financial services tailored to the diverse needs of their customers. These services encompass savings and checking accounts, loans and mortgages, investment products, payment processing, wealth management, and risk mitigation through insurance and hedging mechanisms.

In the digital age, banks are undergoing a profound transformation, driven by technological innovation, regulatory reforms, and changing consumer preferences. The advent of digital banking platforms, mobile payment solutions, and block chain technology is reshaping the traditional banking landscape, empowering customers with greater convenience, accessibility, and customization in financial services delivery.

Investment is a critical function for banks, crucial not only for their profitability but also for the growth and stability of the economy. When a bank invests, it allocates funds into various financial instruments and assets with the aim of generating returns while managing risks.

So, here is an overview about how banks typically approach investment and what the processing involves:

- **Investment Strategy Development.**
- **Risk Management.**
- **Due Diligence and Research.**
- **Portfolio Creation.**
- **Execution and Monitoring.**
- **Rebalancing and Adjustments.**
- **Compliance and Reporting.**

An investment proposal document serves as a comprehensive overview of an investment opportunity, providing potential investors with essential information to evaluate the feasibility, risks, and potential returns of the investment. This document is critical for securing investment capital and building trust and confidence among investors. Here's an outline of the key components of an investment proposal document and the risks associated with not having proper documentation:

- **Executive Summary.**
- **Business Plan.**
- **Financial Projections.**
- **Risk Analysis.**
- **Management Team.**
- **Legal and Compliance Considerations.**
- **Investment Terms and Structure.**
- **Due Diligence Materials.**

The risks of not having proper documentation for giving investment include:

- **Legal and Regulatory Risks.**
- **Uncertainty and Lack of Transparency.**
- **Misalignment of Expectations.**
- **Limited Access to Capital.**

The financial sector is witnessing a transformative shift towards automation and advanced technologies. In the investment sector there are huge tasks. For a Bank there will be a formal proposal of investment. Clients need to submit many documents along with the proposal. Our project aims to check all the procedures and give a result that will help the bankers to process the proposal faster and more efficient.

This project aims to introduce a new way of Investment Proposal Processing System for a bank, leveraging the capabilities of Large Language Models (LLM) [1]. This project outlines the development of an Investment proposal processing system leveraging state-of-the-art language models (LLM) from Hugging Face, with a focus on integrating the Flan-T5 model [2]. It will also work with the text to text generation model [3][4] using Artificial Intelligence with the help of LangChain [5]. This system aims to revolutionize the way investment proposals are processed by providing insightful answers to specific questions and offering informed opinions, enhancing decision-making for investment professionals.

1.1 Problem Statement

In today's digital banking landscape, financial institutions grapple with the challenge of efficiently processing vast volumes of investment-related documents. Despite advancements in technology, the traditional methods of document processing remain labor-intensive, error-prone, and time-consuming. Moreover, the complexity and variability of financial documents further exacerbate these challenges, leading to in-efficiencies, delays, and potential compliance risks.

To address these issues, there is a critical need for automated solutions that leverage state-of-the-art Natural Language Processing (NLP) techniques, such as Large Language Models (LLMs) [1], to streamline the processing of bank investment documents. These documents encompass a wide array of financial instruments, including but not limited to prospectuses, fund fact sheets, investment agreements, and regulatory filings. The key challenge lies in developing an LLM-powered system capable of accurately extracting and interpreting relevant information from diverse investment documents while ensuring data integrity, security, and regulatory compliance.

The proposed project aims to develop and deploy a robust bank investment document processing system utilizing LLM technology [1] called Retrieval Augmented Generation (RAG) [6]. It also uses transfer learning [7] and FLAN-T5 [2] models to generate output. The system will provide some question and then with the help of these technologies and methods it will provide the output. We will also use LangChain [5] and Text to Text generation methods [3][4] to simplify the question answering procedures.

So we can say this project aims to develop a robust bank investment document processing system by harnessing the power of Large Language Models (LLMs) in conjunction with Hugging Face and FLAN-T5 frameworks. This system will confront the following key challenges:

- **Document Understanding and Extraction.**
- **Semantic Interpretation and Contextual Analysis.**
- **Scalability and Performance Optimization.**
- **Integration and Deployment.**

By addressing these challenges, the proposed project endeavors to revolutionize the processing of bank investment documents, empowering financial institutions with advanced NLP capabilities to enhance operational efficiency, mitigate compliance risks, and drive informed decision-making in investment management. Through the convergence of Hugging Face, FLAN-T5, and LLM technologies, the project seeks to redefine the frontier of document processing in the financial sector, paving the way for transformative innovation and competitiveness in the digital age of banking and finance.

1.2 Motivation

In an era defined by rapid technological advancements and evolving customer expectations, it is imperative for financial institutions to embrace cutting-edge solutions that streamline operations while enhancing service delivery. But, the traditional investment proposal processing systems often suffer from inefficiencies such as prolonged processing times, manual data entry errors, and limited scalability. These inefficiencies not only hamper operational productivity but also undermine the overall customer experience. To overcome these problems we built a system that will do the entire task automatically which helps to improve time efficiency and customer satisfaction by harnessing the power of LLM[1], FLAN-T5[2], and LangChain[5].

1.3 Background

This section discusses on relational database, Machine Learning, Large Language Model, Some types of Language Models. It will also discuss about Google Colab, Why and how we use Colab, How we integrate our project into Colab. In this section we will also discuss about why we choose Machine Learning over Rule based Models. Let us start the session and discuss about all these:

1.3.1 Relational Databases

A relational database stores information in tables. These tables commonly share information, resulting in the construction of a relationship between them. This is where the word "relational database" originates from.
Columns specify the information contained in a table while rows store the data. Each table contains a column that must contain unique values. This will be the main key. This column can then be used to connect tables in other tables.
Relational database systems often used query language is SQL. It allows you to write custom queries to aid in the construction, search, and filtering of data across one or more tables.

1.3.2 The Relational Database Structure

In the relational database, data tables, views, and indexes are separated from physical storage structures. As a result of this division, database administrators can change storage of physical data without sacrificing access to logical data.

A relational database structure is a foundational framework for organizing and managing data in a systematic manner. At its core are tables, which represent entities or concepts, with rows defining individual records and columns specifying attributes. Key components include primary keys, ensuring unique identification

of records, and foreign keys, establishing relationships between tables to maintain data integrity. Relationships, such as one-to-one or one-to-many, define connections between tables, enabling efficient data retrieval and manipulation. Normalization techniques minimize redundancy and dependency, enhancing data consistency and efficiency. Adhering to ACID properties ensures transactional reliability, guaranteeing data integrity and durability. SQL serves as the standard language for interacting with relational databases, facilitating operations like data querying, updating, and manipulation. Overall, the relational database structure provides a flexible and scalable approach to data management, making it indispensable in various domains, from small-scale applications to enterprise-level systems.

1.3.3 SQLite

SQLite stands for Structured Query Language Lite. It's a relational database management system (RDBMS) that uses SQL as its query language. The "Lite" in SQLite emphasizes its lightweight nature and its minimalistic design, which makes it easy to embed into applications and use in various contexts. It is a lightweight, embedded relational database management system renowned for its simplicity and versatility. SQLite offers a self-contained, server less architecture, meaning it operates within the context of the application that utilizes it without requiring a separate server process. Its zero-configuration setup makes it easy to integrate into applications. SQLite databases are stored as single disk files, making them highly portable across different platforms and eliminating the need for complex setup procedures. Due to its reliability, cross-platform compatibility, and open-source nature, SQLite has gained widespread adoption in various domains, including web browsers, mobile apps, desktop applications, and embedded systems. Overall, SQLite is a robust and efficient solution for developers seeking a lightweight yet powerful relational database engine.

1.3.4 Machine Learning

Machine learning, a branch of artificial intelligence (AI), involves creating algorithms and statistical models that empower computers to learn and improve through experience, rather than through explicit programming [11]. It revolves around the idea of creating systems that can automatically learn patterns and make decisions based on data rather than relying on explicit instructions. In machine learning, algorithms are trained using large datasets, where they analyze and identify patterns, trends, and relationships within the data. This process involves techniques such as supervised learning, unsupervised learning, and reinforcement learning. Supervised learning involves training algorithms on labeled data, where they learn to predict outcomes or classify inputs. Unsupervised learning, on the other hand, deals with unlabeled data and focuses on finding patterns or structures within the data. Reinforcement learning involves training agents to make sequential decisions through trial and error, where they learn to maximize rewards over time. Machine learning has numerous applications across various domains, including image and speech recognition, natural language processing, recommendation systems, autonomous vehicles, healthcare, finance, and more.

1.3.5 Large Language Model

Large Language Model (LLM) is a type of artificial intelligence system designed to understand and generate human-like text at a massive scale [1]. These models are trained on vast amounts of text data sourced from the internet, books, articles, and other textual sources. They utilize deep learning techniques, particularly transformer architectures, to process and generate coherent and contextually relevant text across various tasks, including language translation, text summarization, question answering, and more. LLMs excel in capturing the nuances of human language, demonstrating the ability to understand and produce text that is grammatically correct, semantically meaningful, and contextually appropriate [6]. Their versatility and proficiency in natural language processing tasks have made them valuable tools in various domains, including customer service, content generation, language translation, and academic research.

1.3.6 Lang Chain

In the realm of natural language processing, a "Lang Chain" refers to a sequential process or pipeline involving the analysis, manipulation, and generation of text. It encompasses various linguistic tasks, such as text preprocessing, tokenization, syntactic parsing, semantic analysis, and text generation [5]. The Lang Chain serves as the backbone of many language-related applications, including Chabot, machine translation systems, and sentiment analysis tools. Within this chain, each component plays a crucial role in understanding and processing textual data, ultimately leading to the desired output. For instance, in a Chabot application, the Lang Chain might involve preprocessing user inputs, parsing them for intent and entities, generating appropriate responses, and then post-processing to ensure coherence and relevance.

1.3.7 Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG) is a natural language processing (NLP) framework that combines elements of retrieval-based and generation-based approaches to improve text generation tasks. In RAG, a retrieval model is used alongside a generation model to enhance the generation process. This framework combines the strengths of both retrieval and generation models, aiming to produce more coherent, contextually relevant, and informative text outputs [10].

At its core, RAG consists of two main components: a retrieval model and a generation model. The retrieval model is responsible for searching through a vast repository of pre-existing text or knowledge, often referred to as the retrieval corpus. This corpus could include sources such as web pages, articles, books, or any other textual data relevant to the task at hand. Using advanced retrieval techniques, the model retrieves passages or snippets of text from the corpus that are deemed relevant to the input or the context provided [9].

Once the relevant information is retrieved, it is passed on to the generation model. The generation model, typically based on state-of-the-art deep learning architectures such as transformers, is responsible for producing the final output text. However, unlike traditional generation models that rely solely on the input provided, the generation model in RAG can leverage the retrieved information to enrich and enhance the generation process.

The integration of retrieval-based techniques into the generation process allows RAG to address some of the limitations of pure generation-based approaches. By incorporating external knowledge and context from the retrieval corpus, RAG can produce text that is not only fluent and coherent but also more accurate, informative, and contextually appropriate. This makes RAG particularly well-suited for tasks such as question answering, dialogue generation, text summarization, and content generation, where access to relevant external knowledge can significantly improve the quality of the output.
Overall, Retrieval-Augmented Generation represents a promising direction in NLP research, offering a powerful framework for addressing complex language understanding and generation tasks by effectively leveraging both existing knowledge and advanced generation techniques. As research in this area continues to advance, RAG is expected to play an increasingly important role in developing more sophisticated and capable AI systems for understanding and generating natural language text.

In our project we also focus this method to generate our desired output from our given input.

1.3.8 GPT-3

GPT-3, short for Generative Pre-trained Transformer 3, is one of the most advanced artificial intelligence language models developed by OpenAI. It belongs to a family of models known as Transformers, which are adept at processing and generating natural language. It's the third iteration in the GPT series, known for its exceptional ability to generate human-like text based on input prompts. GPT-3 stands out due to its massive scale, boasting a staggering 175 billion parameters, which are the elements the model learns from data. This vast size allows GPT-3 to comprehend and generate text with impressive coherence, contextually, and grammatical accuracy across a wide range of topics and languages [12] [13]. Its capabilities span from aiding in writing and content creation to powering chatbots, virtual assistants, and various other natural language processing applications. GPT-3 has garnered significant attention for its potential to revolutionize how we interact with and harness the power of artificial intelligence in numerous domains.

1.3.9 Llama-2

LlAMA-2, or Language Model Analysis through Matrix Estimations, is a sophisticated methodology developed to dissect and comprehend the inner workings of large-scale language models, such as GPT-3.LlAMA-2 focuses on estimating the behavior of language models through matrix manipulations, allowing for deeper insights into how these models generate text and make predictions [14]. By probing language models with carefully designed linguistic tests and analyzing the resulting matrices, LlAMA-2 aims to shed light on their capabilities, biases, and limitations. This technique provides valuable insights into the inner workings of language models, enabling researchers to better understand and potentially mitigate issues such as bias and misinformation in AI-generated content. LlAMA-2 represents a significant step forward in the field of natural language processing, offering new avenues for studying and improving the performance of language models.

1.3.10 Flan-T5

Flan-T5 is an open-source LLM that's available for commercial usage. Published by Google researchers, Flan-T5 is an encoder-decoder model pre-trained on a variety of language tasks. The model has been trained on supervised and unsupervised datasets with the goal of learning mappings between sequences of text, i.e., text-to-text. FLAN-T5 is a combination of two: a network and a model. Here, FLAN is Fine-tuned Language Net and T5 is a language model developed and published by Google in 2020 [2]. This model provides an improvement on the T5 model by improving the effectiveness of the zero-shot learning. It is an enhanced version of T5 that has been fine-tuned in a mixture of tasks. During the training phase, FLAN-T5 was fed a large corpus of text data and was trained to predict missing words in an input text via a fill in the blank style objective. This process is repeated multiple times until the model has learned to generate text that is similar to the input data. Once trained, FLAN-T5 can be used to perform a variety of NLP tasks, such as text generation, language translation, sentiment analysis, and text classification. FLAN-T5 model comes with many variants based on the numbers of parameters.

- FLAN-T5 small (60M)
- FLAN-T5 base (250M)
- FLAN-T5 large (780M)
- FLAN-T5 XL (3B)
- FLAN-T5 XXL (11B)

1.3.11 Google Colab

Google Colab, short for Google Colaboratory, is a free cloud service provided by Google that allows users to write and execute Python code in a browser-based environment, without needing to set up any local development environment. It provides a Jupyter Notebook interface where users can write and run Python code interactively, document their code with Markdown cells, and visualize data with various plotting libraries.

One of the main advantages of Google Colab is that it offers free access to computing resources such as CPU, GPU, and TPU. Users can leverage these resources to train machine learning models, conduct data analysis, or perform other computationally intensive tasks. Additionally, Colab integrates with Google Drive, allowing users to access and communicate with drive and run project from it.

1.3.12 Why we use Google Colab?

Google Colab can be a valuable tool for various reasons, particularly in projects involving data analysis, machine learning, or any task requiring computational resources. Here are some reasons why we use Google Colab in our Project.

We can have free access to computational resources by it like CPU, GPU, and TPU. This can be especially beneficial if we don't have access to powerful hardware locally or if we want to leverage specialized hardware like GPUs or TPUs for training machine learning models. In our project we have done the same because we need to have a very powerful CPU with a handy GPU which is very costly and setting it up is also very time consuming but by Google Colab we can have free access to powerful GPU’s also it is in a virtual state that means we don’t need to set it up. We have an inbuilt Jupyter Notebook interface where we can write and run Python code interactively. Which we use for our project and that’s the reason we use Google Colab for our project.

1.3.13 NGROK

Ngrok is a cross-platform application that creates secure tunnels (paths) to local host machine. It enables developers to expose a local development server to the Internet with minimal effort. The software makes the locally-hosted web server (like computer, laptop, rasbery PI) appear to be hosted on a subdomain of ngrok.com, meaning that no public IP or domain name on the local machine is needed. IT works by establishing a secure tunnel between a public endpoint (provided by Ngrok) and a local server running on the user's machine. This allows external users to access the local server as if it were hosted on a public domain.
So we can say Ngrok is a powerful tool for creating secure tunnels to localhost, enabling developers to easily expose local servers to the internet for testing, development, and remote access purposes. Its simplicity, security features, and flexibility make it a popular choice among developers worldwide.

1.3.14 Why we use NGROK?

In our project we use Google Colab and after completing our project we need to host the app to somewhere and that’s where NGROK comes to rescue. NGROK gives us a funnel over Google Colab and our local host. It gives us a public IP to host the app. After hosting we can run the app by NGROK, Where the backend is in Google Colab and the frontend hosted in NGROK.

1.3.15 Rule-Based Model

A rule-based model refers to the use of conditional statements, such as if-else conditions, switch conditions, and many other to implement logic based on predefined rules or conditions. These conditional statements allow one to define rules or criteria that determine the flow of execution or behavior of a program based on the values of variables or other factors. By satisfying the conditions which defined in the rule based model users can get their output. In rule based model rules are predefined and it can work on simple datasets. From the predefined rules users can get the output of their given input. Sometimes when the input doesn't satisfy the conditions proper output cannot be found.

1.3.16 Why we use Machine learning over Rule-based models?

Machine learning and rule-based models each have their own strengths and weaknesses. In our project we use Machine Learning over Rule-based models. There are several reasons to use Machine Learning over a Rule-based model:

- In our project there are several types of documents. We have to write different rules for different types of document.
- In a bank processing the investment file is a huge task, because a banker needs to check several types of document and their validity. So, we need to write different types of rules for each one of them individually but in Machine learning the system will learn by itself we need not need to write so many rules.
- For a car loan there will be a different document then an agricultural loan there will be different types of return rate and different types of documents to be check. So there will be a variety of date and if we use rule-based model we need to write variety of rules. It will be very time consuming and it will make the project more heavy and difficult to understand. So, for simplify the project and simplify the document processing we use Machine learning in our project.
- As it is an application for banks we need to scale it. Scaling an application in Machine learning is much easier than Rule based model.

1.4 Project Outline

In Chapter 1, we discuss about the introduction and the objective of the study in a concise manner. Chapter 2 deals with the necessary background & literature review for this study. In Chapter 3, we discuss about the proposed methodology, the implementation, NGROK setup and implementation. Here we will see how we host the app in a local-IP. Chapter 4 discusses about the questions that needed to be ask and the output of the proposed system. Chapter 5 draws a conclusion to the proposed system study and discusses future area that can worked on. The final segment of this study contains all the references and credits used.

Chapter 2

Literature Review

Research in Large Language Models, Machine Learning, and Language Processing has been going on for quite some time. Research in banking industry also going on for quite a long time. How to make the banking industry more efficient, how to process all the banking document more easy way, how an Artificial Intelligence can help baking industry and so on. Investment processing of a bank is a very difficult task and research is happening in this sector also. But with the help of Large Language Model and Machine Learning how to improve the investment sector of a bank there is no such data I have found.

Basic Large language Model [1] was discussed in this paper. Large Language Models (LLMs) have surged in prominence due to their exceptional performance across various natural language processing tasks. This has spurred a wave of research spanning architectural enhancements, training strategies, context length extensions, fine-tuning methodologies, multi-modal capabilities, applications in robotics, dataset creation, benchmarking efforts, and efficiency improvements. Literature of the LLM and the advancement in various sector also discussed in this paper. This article aims to provide a concise yet comprehensive overview of recent developments in LLM research.

Architecture of Large Language Model [6] was discussed in this paper. Large Language Models (LLMs) have exhibited remarkable capabilities in various language tasks, such as NLP, translation, and question answering, marking a significant advancement in computerized language processing. This paper provides a concise yet comprehensive overview of LLMs, covering their history, architectures, training methods, resources, applications, impacts, and challenges. It discusses traditional LLM training pipelines, transformer architectures, resources, and training methods, along with datasets used in studies. Applications in diverse domains like healthcare, education, and business are explored, highlighting societal impacts and future potentials. Challenges including ethical concerns, biases, computing resources, privacy, and security are examined, along with strategies to enhance robustness and controllability.

The paper named, Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer [7] discuss about the transfer learning techniques in NLP, proposing a unified framework that transforms text-based tasks into a text-to-text format. It conducts a systematic study comparing pre-training objectives, architectures, datasets, transfer approaches, and other factors across numerous language understanding tasks. Leveraging insights from this exploration, along with scale and a new dataset called "Colossal Clean Crawled Corpus," the paper achieves state-of-the-art results on various benchmarks, including summarization, question answering, and text classification.

The paper Scaling Instruction-Fine-tuned Language Models [2] is where we get our FLAN-T5 model investigates instruction fine-tuning to enhance model performance and generalization on various tasks. It focuses on scaling tasks, model size, and chain-of-thought data. Results show significant performance boosts across different model classes, setups, and evaluation benchmarks. Flan-T5 checkpoints exhibit strong few-shot performance. Instruction fine-tuning emerges as a versatile method for enhancing pretrained language models' performance and usability.

Text to text generation [3] is discussed in this paper. It is a survey paper. This survey addresses the significant advancements in text generation, where state-of-the-art models are revolutionizing various domains by producing human-like text. Despite widespread applications in news, social networks, scriptwriting, and poetry, there's a lack of comprehensive reviews in this area. To fill this gap, the survey presents a systematic mapping study, analyzing 90 primary studies from 2015 to 2021 using the PRISMA framework.

This research paper introduces LangChain [5], a specialized Large Language Model (LLM) designed for automating customer service interactions. It emphasizes the shift from traditional FAQ-based support to context-aware and personalized responses. The framework, named "Sahaay," integrates open-source methodologies, web scraping, fine-tuning, and Google's Flan T5 XXL, Base, and Small models for knowledge retrieval.

This paper discussed about the Investment Mechanism of Islami Bank Bangladesh PLC [8]. Islamic banking operates on principles of Shariah law, which prohibit the payment or acceptance of interest fees for the lending and accepting of money respectively. Therefore, Islamic banks like Islami Bank Bangladesh PLC (IBBL) utilize different investment mechanisms compared to conventional banks. There are some investment modes like Mudaraba, Murabaha, Ijara, Musharaka, Sukuk, Takaful etc. Each of these investment modes has different proceedings and procedures.

This comprehensive review paper delves into the evolution of Retrieval-Augmented Generation (RAG) [9] as a solution to the challenges faced by Large Language Models (LLMs), such as hallucination, outdated knowledge, and non-transparent reasoning processes. RAG integrates external knowledge from databases, enhancing generation accuracy and credibility, particularly for knowledge-intensive tasks, and facilitating continuous knowledge updates and domain-specific information integration. The paper examines the progression of RAG paradigms, including Naive RAG, Advanced RAG, and Modular RAG, while meticulously analyzing the retrieval, generation, and augmentation techniques that form the foundation of RAG frameworks. It discusses state-of-the-art technologies within each component, providing insights into advancements in RAG systems. Additionally, the paper introduces an up-to-date evaluation framework and benchmark. Finally, it outlines current challenges and suggests potential research directions for further development in the field of RAG.

The commentary on GPT-3 by Min Zhang and Juntao Li, published in the MIT Technology Review in 2021, provides an insightful analysis of OpenAI's language model, highlighting its substantial advancements and significant impacts. GPT-3, with its 175 billion parameters, excels in generating human-like text and performing diverse tasks with minimal input, showcasing remarkable zero-shot, one-shot, and few-shot learning abilities [12]. However, the authors also address the challenges and limitations of GPT-3, such as its potential to produce biased content and its high computational demands. They emphasize the ethical and societal implications, including the risks of misuse and the need for responsible usage guidelines. The commentary concludes by calling for future research to enhance model interpretability, reduce biases, and improve efficiency, advocating for collaborative efforts to ensure the responsible development and deployment of powerful AI technologies.

The paper "Llama 2: Open Foundation and Fine-Tuned Chat Models" presents Llama 2, a suite of advanced language models developed by Meta. Llama 2 includes both foundational models, which serve as robust starting points for a variety of language tasks, and fine-tuned chat models specifically optimized for conversational AI applications [14]. The paper details the architecture, training methodologies, and capabilities of these models, emphasizing their improved performance, safety, and scalability compared to previous iterations. Additionally, the authors highlight the importance of open access to these models to foster transparency, collaboration, and innovation in the AI research community. The comprehensive evaluation demonstrates Llama 2's state-of-the-art performance across various benchmarks, reinforcing its potential for diverse real-world applications.

In this project work, we have made a dataset of investment files from banks. We then train the dataset with RAG and then we use FLAN-T5 to get the output in a question answer manner. We also use LangChain and other Machine Learning techniques to make our project more efficient.

Chapter 3

Proposed Approach

With the problem statement identified above the main objective is to come up with an effective system that will optimize the task of an investment banker. The main objective of the system is to build a system that will process the investment proposal files and ask some questions and get the answers by LLM and Machine Learning. When after getting the answers one investment banker can take his decision in a more effective and efficient manner. Below are the key main objectives that is been identified.

- Construct a system that can process data automatically.
- Apply LLM and FLAN-T5 model properly to create a question answering based platform.
- Construct a text to text generation model.

3.1 Requirement Analysis

Requirement analysis of this project is the understanding of the requirements means what the users or investment bankers really want. It involves understanding the needs and objectives of various stakeholders, as well as the current challenges and inefficiencies in the existing investment proposal workflow. So for the requirement analysis of the project we first need to understand the idea of the project. Here the method we follow for gathering and analyzing requirement is question and answering process. We use interviewing techniques for our requirement analysis. So, we need to talk with several investment bankers of different banks to get the knowledge about what they do and how they do. Like for example, what is the type of the investment? How much mortgage is needed?

And so on. Then we need to talk with the investment clients to discuss about what the problem they face and how to solve them. Lastly we need to analyze all our findings and get to a result that will solve all the issues from both perspectives like the bankers and the client's point of view.

For the requirement analysis we need to work with the existing investment proposal processing system. We need to find the difficulties of the task and a way to solve them. Right now there is no such a system everything is done by manually. So, we need to analyze the task and make a system that we do all the processing automatically. After making we also need to verify the system by human testing.

3.2 Data Collection & Preprocessing

In the data collection and preprocessing phase of our project we need to gather relevant data and ensure its quality and consistency for analysis. In our project we work with the original investment file of a bank. We talked with the bank and have our permission to work with some files. As the files are important and confidential so we don't get access to a huge data set. But we also have some sample files for our project too. For data collection we need to follow the steps:

- Talk with the bank and get some investment file. As the files are confidential for a bank because they hold all kinds of information of an investment client personal and professional also business data, it cannot be go to wrong hand. So, we have to go through proper channel with proper guiding to have access to the files.

- Talk with an investment officer and ask what they need to check in the file. One file can hold several types of information & all the information are not needed for giving investment. So, the bankers who are volunteered for the project we need to go and interview them about their need in an investment file.

- Prepare some sample documents. Because the files are confidential we cannot have that much amount of investment file from a bank so, we need to create some sample files for the training and processing of our project.

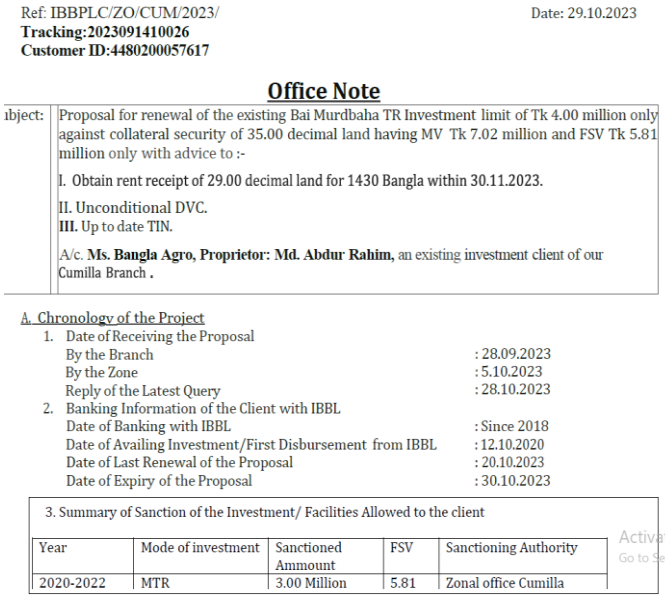
After getting the files as our data we need to preprocess the files also. We need to use preprocessing methods to get the file in our format which we use for our project. Let us discuss some techniques:

- We need to remove the garbage value first. For example, some certification from the government, or some other organization, some information that are not needed for the processing. We need to remove these types of document because the information of those documents is already in the proposal file.

- Convert the file size to our desired size. In our project the desired length of the files are 15 MB.

- Lastly, we need to convert the file to PDF format. As all the documents are mostly scanned document and as we know the information on these documents are very much important for processing an investment. So the data cannot be changed and that's the reason we need PDF format file for our input.

Let us see an example of a sample file for the project:



Ref: IBBPLC/ZO/CUM/2023/ Date: 29.10.2023
Tracking:2023091410026
Customer ID:4480200057617

Office Note

ıbject: Proposal for renewal of the existing Bai Murdbaha TR Investment limit of Tk 4.00 million only against collateral security of 35.00 decimal land having MV Tk 7.02 million and FSV Tk 5.81 million only with advice to :-

I. Obtain rent receipt of 29.00 decimal land for 1430 Bangla within 30.11.2023.

II. Unconditional DVC.
III. Up to date TIN.

A/c. **Ms. Bangla Agro, Proprietor: Md. Abdur Rahim,** an existing investment client of our Cumilla Branch .

A. Chronology of the Project

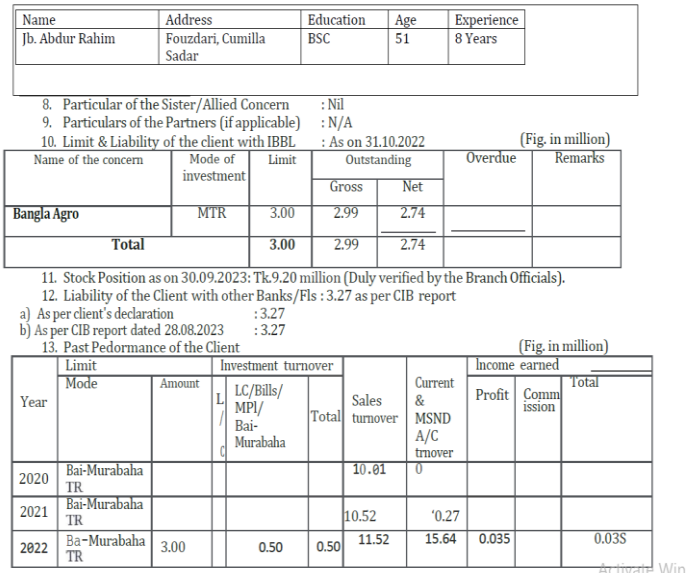
1. Date of Receiving the Proposal
   By the Branch : 28.09.2023
   By the Zone : 5.10.2023
   Reply of the Latest Query : 28.10.2023
2. Banking Information of the Client with IBBL
   Date of Banking with IBBL : Since 2018
   Date of Availing Investment/First Disbursement from IBBL : 12.10.2020
   Date of Last Renewal of the Proposal : 20.10.2023
   Date of Expiry of the Proposal : 30.10.2023

3. Summary of Sanction of the Investment/ Facilities Allowed to the client

Year	Mode of investment	Sanctioned Ammount	FSV	Sanctioning Authority
2020-2022	MTR	3.00 Million	5.81	Zonal office Cumilla

Activate Wi
Go to Settings t

Fig.3.1: Sample Input file part-1



Name	Address	Education	Age	Experience
Jb. Abdur Rahim	Fouzdari, Cumilla Sadar	BSC	51	8 Years

8. Particular of the Sister/Allied Concern : Nil
9. Particulars of the Partners (if applicable) : N/A
10. Limit & Liability of the client with IBBL : As on 31.10.2022

(Fig. in million)

Name of the concern	Mode of investment	Limit	Outstanding		Overdue	Remarks
			Gross	Net		
Bangla Agro	MTR	3.00	2.99	2.74		
Total		**3.00**	2.99	2.74		

11. Stock Position as on 30.09.2023: Tk.9.20 million (Duly verified by the Branch Officials).
12. Liability of the Client with other Banks/Fls : 3.27 as per CIB report
a) As per client's declaration : 3.27
b) As per CIB report dated 28.08.2023 : 3.27
13. Past Pedormance of the Client

(Fig. in million)

Year	Limit		Investment turnover			Income earned			
	Mode	Amount	L/C	LC/Bills/ MPI/ Bai-Murabaha	Total	Sales turnover	Current & MSND A/C trnover	Profit	Commission	Total
2020	Bai-Murabaha TR					10.01	0			
2021	Bai-Murabaha TR					10.52	'0.27			
2022	Ba-Murabaha TR	3.00		**0.50**	**0.50**	**11.52**	**15.64**	**0.035**		0.03S

Fig.3.2: Sample Input file part-2

6. Mode of Disbursement : Deal to deal basis as per usual norms of the Bank
7. Mode of Recovery : Deal to deal basis as per usual norms of the Bank
8. Security

a) Cash/Goods
Bai-Murabaha TR : Without cash security

b) Primary
Bai-Murabaha TR : Lien on goods to be released against TR till disposal and deposit of sale proceeds towards adjustment of related investment A/cs with the Branch on condition that every deal under TR shall have to be adjusted within six months from the date of disbursement and no disbursement be made under TR if any deal under TR remains overdue.

c) Collateral: Registered mortgage/Further charge with Registered Irrevocable Power of Attorney from the mortgagor(s) in Bank's favor to sell out the following mortgaged properties in case of any default in payment of Bank's dues by the client:

(Amount in million)

ST. xo.	Paniculars	As per Branch Valuation		As per Surveyor Valuation	
		MV	FSV	MV	FSV
1	29.00 decimal Land Under Dist. Cumilla, PS&SRO- Nangalkot, Mouza- Horipur, JL.No. Sabek- 498, Hal BS -64, Khatian No. CS — 93, RS-179, BS-751, Plot No. CS -662, RS-662, BS — 1038, owned by Jebunnahar(Mother of the Client)	4.8	4.04	4.8	4.04
2	6.00 decimal Land Under Dist. Narayngonj, PS&SRO- Rupgonj, Mouza- Barpa, JL.No. Sabek CS& SA- 554, Hal RS -201, Khatian No. CS -305,SA-332, RS-375, Mutated - 3224, Plot No. CS &SA - 34 RS-131, Mutated- 131 owned by Md Omar Fanik(Brother of the Client)	2.22	1.77	2.22	1.77
	Total: (19.00+6.00) =25.00 decimal land	7.02	5.81	76.02	5.81

Legal opinion and genuineness certificate are obtained
Name of the enlisted surveyor: MSK Inspection Company Ltd.

D. Whether the following papers/documents/information(s) have been obtain/furnished or not (Yes/No):

1	client's Application	Yes	15	to Date Rent Receipt	Yes
2	Valid Trade License	Yes,	16	Client's Photo	Yes
3	Stock Report	Yes	17	Visit Report	yes
4	RC/ERC	N/A	18	ATCA, KYC	yes
5	Up to Date TIN/VAT	Yes	19	Shariah's Declaration	Yes
6	Legal Opinion	Yes	20	Audit Observation	Yes
7	valuation Certificate	Yes	21	Declaration of the client's NID	Yes
8	DVC-Conditional	Yes			
9	surveyor Valuation	Yes			

10	Location Map /Site land	Yes	22	Training Certificate	No
11	Balance Sheet/Asset* liability position	Yes	23	Credit Rating	Yes
12	ICRRS	N/A	24		
13	Clean CIB Report	Yes			
14	Previous Sanction Advice	Yes			

Fig.3.3: Sample Input file part-3

14. Profit-Loss Account of the client as on 31.12.2022 (Amount in million)

Particulars	For the year ended on 31.12.2022
Sales/Export	11.52
Cost of Goods Sold	8.07
Gross Profit	3.45
Administrative, general & selling Expenses	0.90
EBIT (Operating profit)	2.55
Financial Expenses	0.035
Other Income	0
Net profit before tax and appropriation	2.515
Tax	0.0
Other appropriation	0
Net Profit after tax and appropriation	2.515

15. Net worth Estimation (Branch Assessment) : 30.09.2023 (Fig. in million)

A. Property & Assets	B. Liability & Equity

a. Current Assets	Amount	a. Current Liabilities	Amount
1. Cash & Bank Balance	0.10	1. Investment from Bank/Financial Institutions a) IBBL b) Others	3.27 0.0
2. Stock in trade & inventories/finished goods	9.20	2. Borrowing *from* other sources	0
3. Accounts receivable (Sundry Debtors)	0.92	3. Accounts Payable (Sundry Creditors)	0
4. Advance Deposit & Pre-payment	0.0	4. Others	0.00
5. Other Current Assets			
Sub Total (a)	10.22	b. Long Term Liability	
b. Fixed Asset		c. Other non-current liabilities :	
6. Land, Building & other immovable assets	6.02	B. Total Liabilities (a+b+c)	3.27
7. Plant, Machinery and Furniture & Fixture	0.25	d. Capital/	5.00
8. Other Assets:	8.12	e. Reserves	0.00
		f. Retained Earnings/Net profit for the year transferred to Balance Sheet	0.0
Sub Total (b)	14.39	C. Total Equity (d+e+f)	5.00
A. Grand Total (a+b)	24.61	Grand Total (a+b+c+d+e+f)	8.27
NET WORTH : A-B (total assets — total liabilities)	16.34		

Fig.3.4: Sample Input file part-4

3.3 Model Selection

In our problem statement we identified that the task of making banking related software is a very difficult one. We have seen in our previous discussion that investment is a crucial function of a bank. Our aim is to make software that will use Lang Chain, LLM and Machine Learning to process the investment related documents.

There are several model of LLM. We use Retrieval-Augmented Generation (RAG) [10] as our LLM model. It is an advanced natural language processing (NLP) technique that combines the strengths of retrieval-based and generative models. In RAG, a generative language model is augmented with a retrieval mechanism, allowing it to access and incorporate information from external knowledge sources during the text generation process.

We can say in the ever-evolving landscape of artificial intelligence, Retrieval Augmented Generation (RAG) [9] is making waves. This innovative approach combines the power of large language models with the reliability of factual data retrieval.

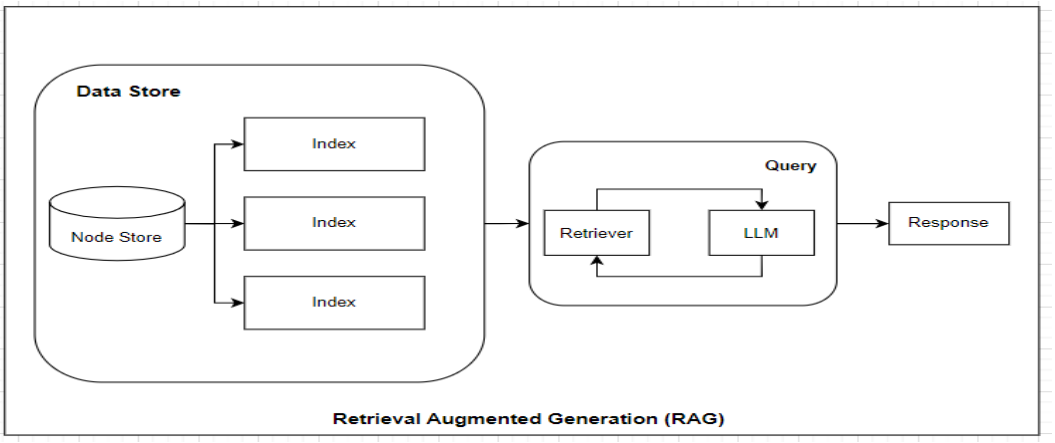


Fig.3.5: Retrieval Augmented Generation (RAG)

Large language models, such as ChatGPT, have revolutionized natural language processing. However, they have a tendency to “hallucinate,” or generate information that sounds plausible but is not grounded in facts. This is where Retrieval Augmented Generation (RAG) comes into play.

RAG [10] enhances the reliability of these models by grounding their responses in factual data retrieved from a vector database. This approach not only ensures the accuracy of the generated information but also provides a reference point for users to verify the data. Furthermore, by focusing on factual data relevant to specific domains, RAG allows large language models to concentrate more effectively on domain-specific tasks. This results in more accurate, reliable, and contextually relevant outputs.

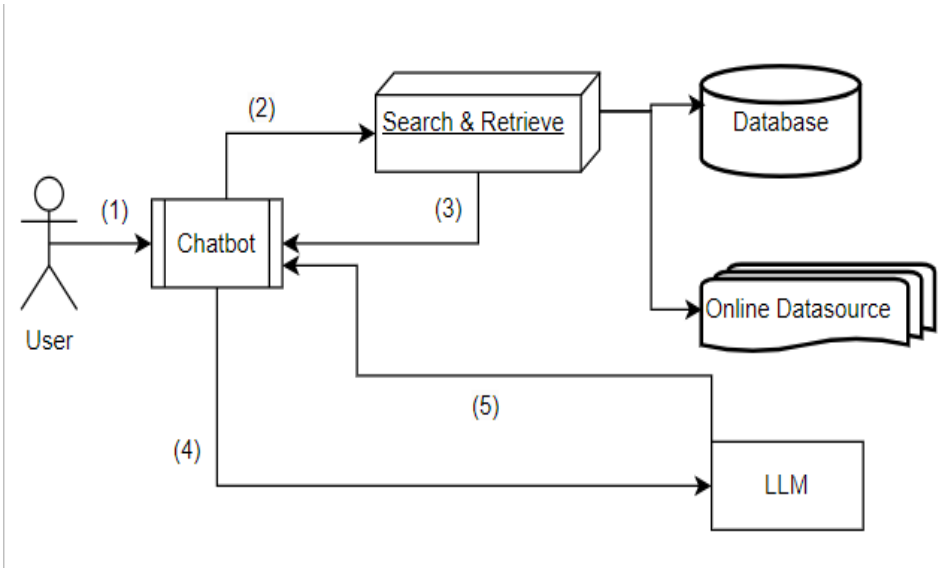


Fig.3.6: LLM using RAG

Now let us discuss how RAG operates:

1. **Document Retrieval**: Initially, the input sequence, usually a question or prompt, is used to perform a semantic search in a vector database. This search retrieves a set of relevant documents. These documents serve as an external knowledge source that the model can refer to.
2. **Combining Input with Retrieved Documents**: The input sequence is combined with the retrieved documents to form an extended context. This extended context contains both the original input and additional information from the retrieved documents.
3. **Passing to Decoder Transformer**: The extended context is then passed to a decoder transformer. It's important to highlight that RAG is compatible with any decoder transformer, not just specific ones. This flexibility allows it to be integrated into various architectures and applications.
4. **Generating Response**: The decoder transformer processes the extended context and generates a response. The response is not just based on the input sequence but is also informed by the information in the retrieved documents. This ensures that the output is grounded in factual data and is relevant to the input.

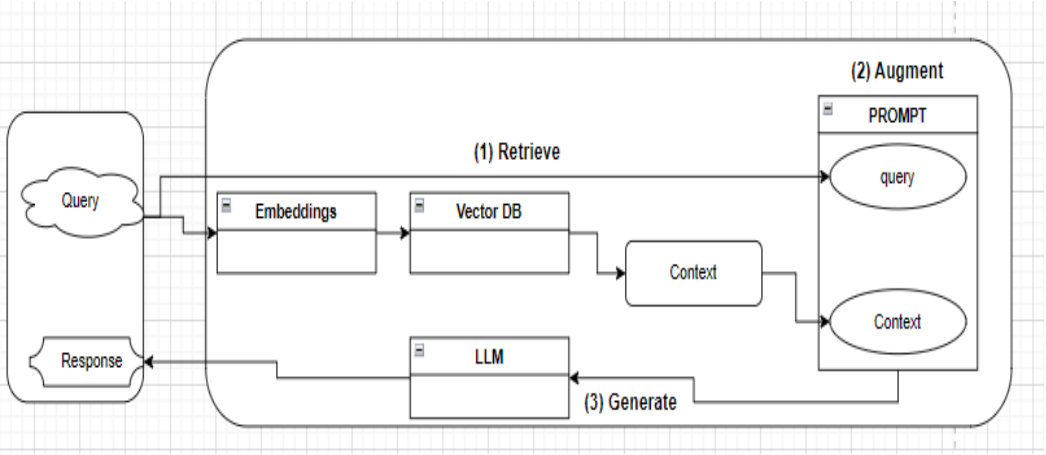


Fig.3.7: Working procedure of RAG

Now let me discuss how I use RAG in our project. As it is a question answer based project. Input files are divided into chunks. Then these data chunks are combined with retrieval documents model and also converted to vector so that LLM and LangChain can retrieve the answers from the document. As we are using RAG our questions will go through the input document and if it doesn't find any appropriate answer it will combine an answer and give the output after analyzing the whole document based on some similar data from the input file.

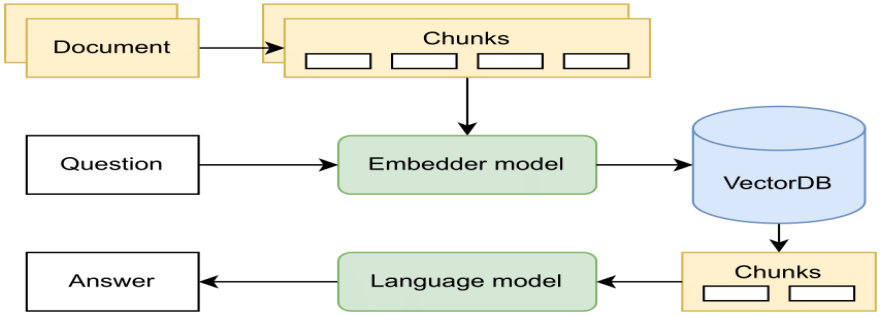


Fig.3.8: RAG in our project

3.4 Question Gathering

One of the major tasks of our project is to question gathering. We have to talk with the investment bankers and ask them what they want to find out from the investment proposal document given by the client. There can be different types of question based on different types of loans also in different types of situations. Here is an overview of how we can gather questions from our bankers who actually do the investment related tasks:

- Talk with the investment bankers.
- Find out what they want to find from an investment processing file.
- Make the questions based on their suggestions.
- Make a priority list of the questions.
- Verify with the investment bankers that our questions are actually viable.

Now let us give an example of some sample questions which we use in our project. In our project clients can also add questions when needed; they can also modify them to.

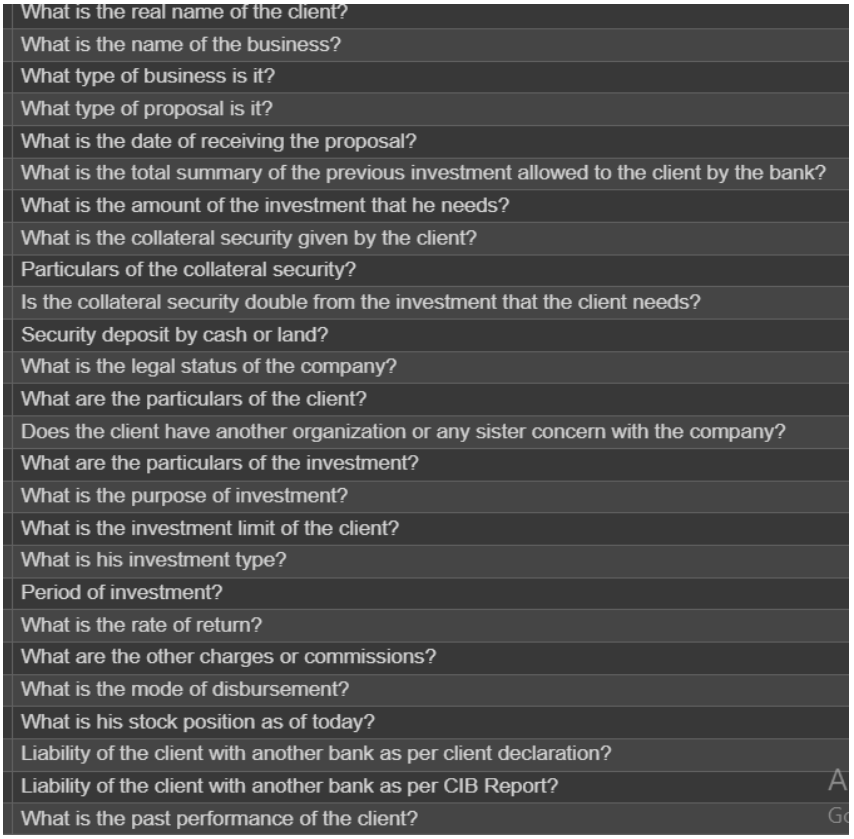


Fig.3.9: Sample Questions

3.5 Evaluation

Evaluating this project called Investment Proposal Processing System in Banks using LLM is a hard task. We have to go for human evaluation as there is no existing system. Human evaluation can vary over banker to banker and bank to bank. Let us discuss how we can evaluate the system:

- We should go for human evaluation.
- Investment banker can analyze the output and find out if it matches their expectations or not.
- We should make the risk assessment with the help of investment bankers.
- We should make the cost estimation for the project.

As the aim of the project is to simplify the task of a human being, we should evaluate the project focusing on these criteria.

3.6 User Interface Development

We are building a software that will help the investment bankers to do their task more efficiently. Investment Bankers process the investment proposal files and check if there any problem in the file or the file is ok. They want to find some answers of their questions. Our project aims the same.

As we use machine learning in our project. We have done the project in Google Colab and to show output we use Ngrok which create funneling from the system to a localhost. It will give a public IP where we can host and run the app.

For user interface development for the app we use CSS and HTML to build the web page also for the database we use SqLite. First the user has to log in or register to the system and then there will be a home page after login. In the homepage there is an upload option where we can upload the files. Then we can add or select questions from the sample questions. Then the system will find the answer after going through the entire document. There is a history page where we can find the previously processed files. Previously processed file can also be modified too if needed.

Sign in to your account

Email address

abc@gmail.com

Password Forgot password?

••••••

Sign in

Not a member? Start a 14 day free trial

Fig.3.10: Sample UI-1 (Login)

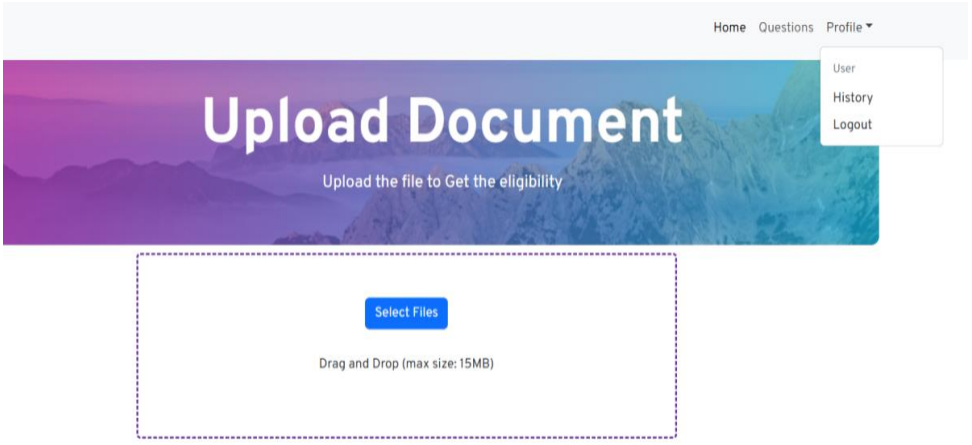


Fig.3.11: Sample UI-2 (Home)

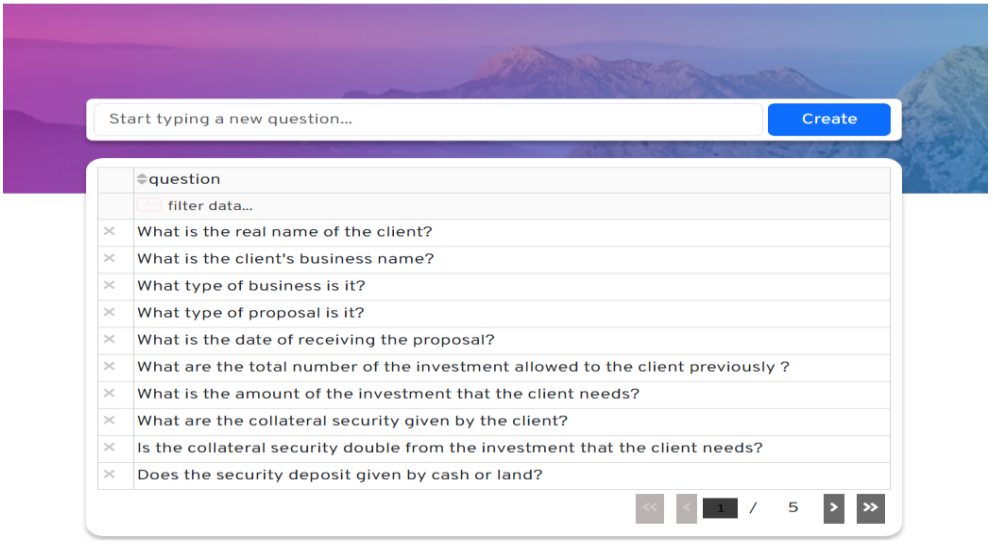


Fig.3.12: Sample UI-3 (Questions)

3.7 Monitoring and Optimization

After successfully completing the project in this section we will discuss about how to monitor and optimize our project. In our project we work with LLM. There are some strategies we can take to monitor and optimize our project. Like:

- We will use human team for monitoring the performance of our project.
- We can track the performance by real time tracking system.
- There can be an automated alert system if the file is done processing.
- We can optimize by different LLM techniques.
- Optimize the decision making of the project by continuous development and use of transfer learning.

3.8 Scalability

Scalability of a project is very much important. Scalability refers to the ability of a system, network, or process to handle an increasing amount of work, growth, or demand without sacrificing performance, stability, or quality. In simpler terms, scalability measures how well a system can adapt and grow to accommodate larger volumes of activity or users without experiencing negative impacts such as slowdowns, failures, or decreased efficiency. We can improve the scalability of our project by various tasks like:

- Giving different types of input file to train more and more.
- Improve the question quality that the system will ask from the input files.
- Run the system in various platforms to see how it performs.
- We use FLAN-T5 large because of low GPU; we can use high GPU to train FLAN-T5 xxl to scale the project and to get accurate output.

Chapter 4

Results and Discussions

In this section, we present the results obtained from implementing the investment proposal processing system utilizing Large Language Models (LLM) within banking institutions. The results encompass various aspects, including efficiency gains, accuracy improvements, and user satisfaction.

Based on our model and architecture the system was tested and the result was generated with the help of some investment bankers who volunteered in the project. As we discuss earlier that the information of the investment files are confidential we cannot have access to a large amount of flies. So, we have gathered 20 files for processing & we interview total 15 investment bankers. So, total 15 investment bankers volunteered for the project. We have shown them the investment proposal processing system & we process the document one by one. We have discussed the result of our project with them and take their opinion about our output. The performance of our project is quite satisfactory among the banks & the bankers. As we have made the software using LLM where we use FLAN-T5 large model the result is satisfactory, but if we use FLAN-T5 xxl model the result will be more humanlike.

Now as we are working with Ngrok , sometimes funnel creation and process huge files may be a problem this can hamper our project performance. If we use a machine with better GPU we can easily solve the issue.

As our system aims to help investment banker by minimizing their task, the performance need to be measured by human. Also we need human testing so see if our result is good enough to solve the issue. After talking with several bankers we found that our system accuracy is nearly 77% in FLAN-T5 large model and FLAN-T5 small model it is nearly 65%, if we can use FLAN-T5 xl and FLAN-T5 xxl it can be much higher like 85% in xl and 98% in xxl approximately. But FLAN-T5 xl and FLAN-T5 xxl needs high GPU based machine which is not common in banking industry.

Table 4.1: Shows performance of the system with different FLAN-T5 model

FLAN-T5 model Performance			
FLAN-T5 small	FLAN-T5 large	FLAN-T5 xl	FLAN-T5 xxl
65%	77%	85%	98%

And for the processing time small dataset can process less data with xxl dataset can process huge amount of data with minimum time. As we use FLAN-T5 large model it can process moderate amount of file at a time, where we have select the number of question to be processed at a time. In our project we the number is 25 questions in a single process with a single file.

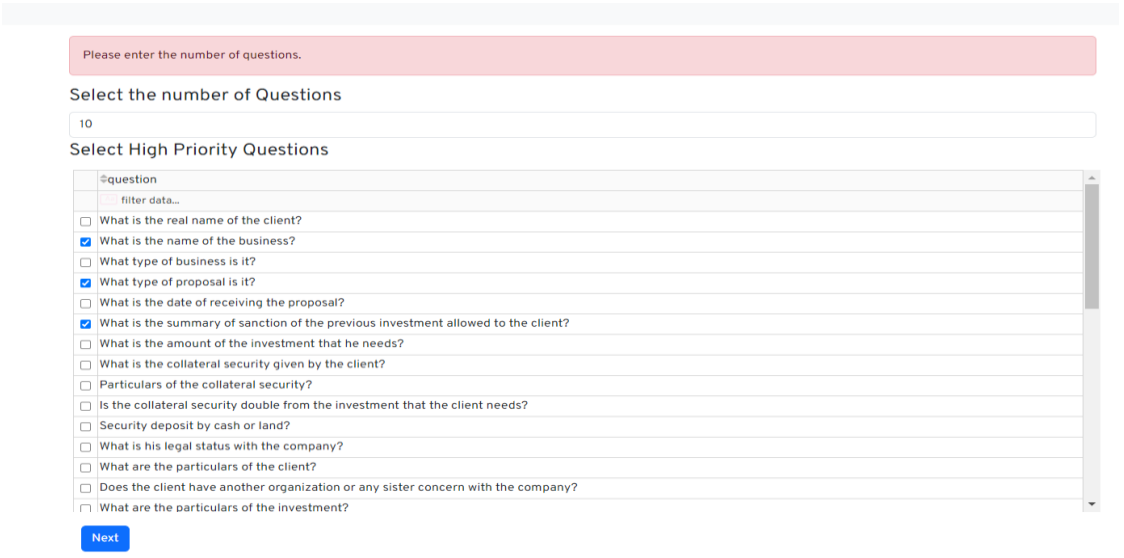


Fig.4.1: Processing part of the output

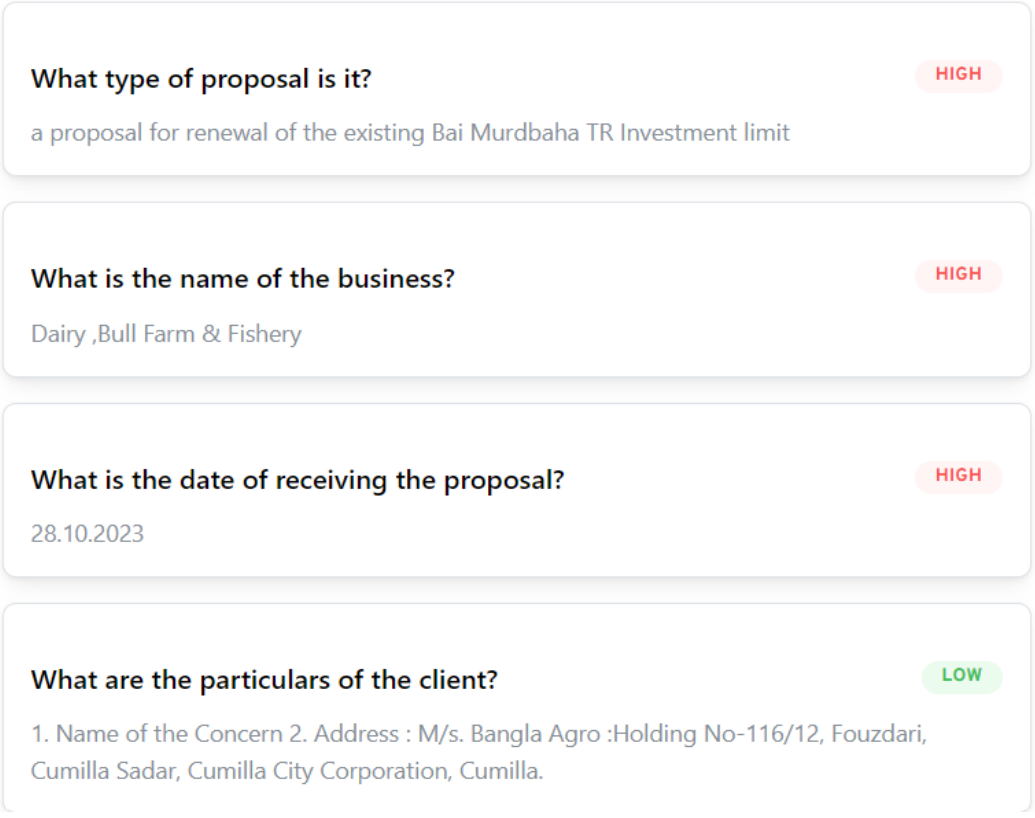


Fig.4.2: Sample Output

Chapter 5

Conclusion and Future Work

In conclusion, the implementation of an investment proposal processing system leveraging Language Model technology presents a significant opportunity for banks to enhance efficiency, accuracy, and customer satisfaction in their investment operations. By harnessing the capabilities of Language Models, such as the Large Language Model (LLM), banks can streamline the entire investment proposal lifecycle, from initial submission to final approval, while also ensuring compliance with regulatory requirements.

Through the automation of routine tasks, intelligent data extraction, and natural language understanding, this system promises to reduce processing times, minimize errors, and empower bank staff to focus on higher-value activities, such as personalized client interactions and strategic decision-making. Moreover, the integration of machine learning algorithms enables continuous improvement and adaptation to evolving market dynamics and customer preferences. Also by embracing the power of Language Models, banks can revolutionize their investment operations, unlock new opportunities for growth, and ultimately, strengthen their position in the marketplace. With the right strategic vision and execution, the future of investment proposal processing in banks is bright, powered by the transformative potential of advanced language technology.

As we have a limited capacity we can suggests some future work that can be done to the project to enhance the processing more and get more useful and better result. Some of our suggestions are:

- Developing specialized modules within FLAN-T5 to identify and evaluate potential risks associated with investment proposals can enhance overall risk management practices.
- We can use better fine tuning to improve the preprocessing of the input files. This involves optimizing model parameters, refining retrieval mechanisms, and adapting the model to evolving market conditions.
- Enhancing the model's capability by integrating additional external data sources can further enrich the context and relevance of investment proposal analysis. Incorporating structured financial data, market trends, and regulatory information can improve decision-making accuracy and robustness.
- Try the system with different model like LLAMA or GPT and see which one performs better.
- Future work should focus on optimizing system architecture, implementing efficient resource management strategies, and leveraging distributed computing technologies to accommodate increasing workloads.
- Improving the user experience and accessibility of the investment proposal processing system is vital for driving user adoption and satisfaction. Future efforts should concentrate on designing intuitive user interfaces, providing comprehensive training materials, and offering multi-platform support to enhance usability.

By addressing these areas of future work, we can further advance investment proposal processing capabilities using LLM, empowering banks to make more informed decisions, streamline operations, and deliver enhanced value to their clients and stakeholders.

Bibliography

[1] H. Naveed, A. U. Khan, S. Qiu, *et al.*, "A comprehensive overview of large language mod- els," *arXiv preprint arXiv:2307.06435*, 2023.

[2] H. W. Chung, L. Hou, S. Longpre, *et al.*, "Scaling instruction-finetuned language models," *arXiv preprint arXiv:2210.11416*, 2022.

[3] N. Fatima, A. S. Imran, Z. Kastrati, S. M. Daudpota, and A. Soomro, "A systematic liter- ature review on text generation using deep neural network models," *IEEE Access*, vol. 10, pp. 53 490–53 503, 2022.

[4] J. Li, T. Tang, W. X. Zhao, J.-Y. Nie, and J.-R. Wen, "Pretrained language models for text generation: A survey," *arXiv preprint arXiv:2201.05273*, 2022.

[5] K. Pandya and M. Holia, "Automating customer service using langchain: Building custom open-source gpt chatbot for organizations," *arXiv preprint arXiv:2310.05421*, 2023.

[6] M. A. K. Raiaan, M. S. H. Mukta, K. Fatema, *et al.*, "A review on large language models: Architectures, applications, taxonomies, open issues and challenges," *IEEE Access*, 2024.

[7] C. Raffel, N. Shazeer, A. Roberts, *et al.*, *Exploring the limits of transfer learning with a unified text-to-text transformer*, 2023. arXiv: 1910.10683[cs.LG].

[8] M. Khorshed, "2022 report on investment mechanism of islami bank bangladesh limited i research report investment mechanism of islami bank bangladesh limited course title: Research matodology submitted to submitted by mohammed khorshed bba in finance report on investment mechanism of islami subject: Submission of research report titled on"report on investment mechanism of islami bank bangladesh limited."," Ph.D. dissertation, Jun. 2022.

[9] S. Liu, Y. Chen, X. Xie, J. Siow, and Y. Liu, *Retrieval-augmented generation for code summarization via hybrid gnn*, 2021. arXiv: 2006.05405[cs.LG].

[10] Y. Gao, Y. Xiong, X. Gao, *et al.*, *Retrieval-augmented generation for large language mod- els: A survey*, 2024. arXiv: 2312.10997[cs.CL].

[11] Z. Sun and Z. Wu, *Handbook of Research on Foundations and Applications of Intelligent Business Analytics*. IGI Global, 2022.

[12] M. Zhang and J. Li, "A commentary of gpt-3 in mit technology review 2021," *Fundamental Research*, vol. 1, no. 6, pp. 831–833, 2021.

[13] L. Floridi and M. Chiriatti, "Gpt-3: Its nature, scope, limits, and consequences," *Minds and Machines*, vol. 30, pp. 681–694, 2020.

[14] H. Touvron, L. Martin, K. Stone, *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.