Suspicious Activity Report (SAR) Generator using Gen-AI

Authors: Jeff Mathew Sam, Mohanad Alkhalaf, Nour Alzaid, Chao Hu

December 5, 2024

Table of Contents

1. Executive Summary

Our Client : Agricultural Bank of China (ABC), New York Branch.

The Agricultural Bank of China (ABC) New York branch is a prominent financial institution that caters to a wide array of corporate and individual clients, offering essential banking services on a large scale.

Challenge:

Three months ago, the Agricultural Bank of China reached out to us with a pressing issue. The manual process of creating Suspicious Activity Reports (SARs) was a major challenge, requiring analysts to dedicate anywhere from a minimum of 2 hours to as much as 2 days for each report. This labor-intensive process led to inefficiencies and increased operational costs. Furthermore, the reliance on cloud-based AI systems raised concerns regarding data privacy, as sensitive customer information could be vulnerable to breaches.

Solution:

To tackle these challenges, we suggested implementing a secure, locally-hosted Generative AI (GenAI) system. This cutting-edge solution automates the generation of SARs, drastically reducing the time needed to file a report from hours or days to just 30 minutes, and lowering the cost per report from $75 to $20. By keeping the system local, we ensured adherence to stringent data privacy standards while achieving high accuracy and efficiency in SAR generation.

Impact:

The proposed solution provides the following advantages:

- Time Savings: Cuts the time required for SAR filing from 2 hours to 2 days down to just 30 minutes, allowing analysts to concentrate on more valuable tasks.
- Cost Efficiency: Reduces the cost per report by over 70%.
- Enhanced Compliance: Lowers data privacy risks by removing reliance on cloud-based systems.

This partnership with the Agricultural Bank of China marks a significant advancement in enhancing compliance efficiency while addressing critical operational challenges.

2. Background and Context

2.1 Understanding Money-Laundering & Anti Money-Laundering (AML)

2.1.1 What is Money Laundering?

A deeper understanding of SARs, why banks generate them, and how they generate them requires a broader contextual understanding of their relation to specific financial crimes. One such financial crime is Money Laundering. Money laundering is the process of concealing the origins of financial assets, enabling their use without revealing the illegal activities that generated them (FinCEN, 2019). It is the process of lying about where the money is from or setting up intricate systems to disguise the source of illegally obtained money, typically through a complex series of financial transactions, to make it appear legitimate. It aims to move illicit funds into the legitimate financial system without attracting attention from authorities.

Three Stages of Money Laundering:

- **Placement**: Introducing illicit funds into the financial system.
- **Layering**: Concealing the source of the funds through complex transactions.
- **Integration**: Merging laundered money into the legitimate economy.

2.1.2 Overview of Anti-Money Laundering (AML)

Anti-money laundering (AML) includes the regulations, policies, and procedures adopted by banks to prevent criminals from transforming illegally obtained funds into legitimate income. The primary aim is to detect and stop money laundering, which fuels terrorism, organized crime, and other unlawful activities. To achieve this, banks continuously scrutinize customer transactions for any suspicious behaviour (ABC Client, personal communication, September 9, 2024).

2.1.3 Key AML Requirements for Banks

To aid the AML efforts, banks leverage various data sources in their AML detection systems. For our client, this includes ‘Know Your Customer’ (KYC) details, transaction monitoring data, and information from alert investigations. KYC data encompasses the customer’s name, address, socio-economic background, occupation, and financial activities. Transaction monitoring focuses on spotting unusual patterns in financial transactions, which are then manually reviewed to assess their suspicious nature based on the customer's KYC information. Most large banks use sophisticated software to monitor and detect unusual patterns in large volumes of financial transactions. The system scans transactions for suspicious activity using pre-set rules and algorithms.

The primary purposes of AML is to fight against money laundering by:

- Manipulation control.
- Protection of financial institutions from being used in channeling proceeds of crime.
- Ensuring compliance with legal and regulatory requirements to uphold the integrity of the financial system.

2.2 Problem Understanding - Current Suspicious Activity Report (SAR) Process

2.2.1 SAR, SAR narratives, and current process for generating SAR narratives

The Suspicious Activity Report (SAR) is a critical document that financial institutions must submit to the Financial Crimes Enforcement Network (FinCEN) when they suspect money laundering or fraud. The process begins with automated transaction monitoring systems that flag suspicious activities based on various triggers, such as structuring, large transactions, rapid movement of funds, unusual transactions by new customers, round-dollar amounts, or transfers to high-risk countries. While these systems can generate false positives and negatives, banks typically adopt a risk-based approach to minimize missing potentially suspicious activities.

When an alert is triggered, an analyst thoroughly reviews the flagged activity by examining the customer's KYC details and conducting both internal and external research. They then create an "Alert narrative" that documents all relevant transaction details, including customer information and findings from their investigations. This document serves as the foundation for determining whether a SAR needs to be filed.

If suspicious activity is confirmed, the bank must file a SAR with FinCEN through the BSA E-Filing System within 30 calendar days of identifying the suspicious transactions. The filing process consists of five key steps: filing institutional contact information, documenting the financial institutions where activity occurred, providing subject information, detailing suspicious activity information, and writing a narrative.

Importantly, the SAR narrative differs from the alert narrative in its structure and content. While the alert narrative contains comprehensive information about the customer and due-diligence findings, the SAR narrative is specifically formatted for law enforcement readability. It excludes sensitive or irrelevant customer information and focuses solely on the pertinent details of the suspicious activity. This careful documentation and reporting process helps banks maintain compliance with the Bank Secrecy Act (BSA) while assisting law enforcement in identifying potential financial crimes.

3. Project Objectives & Project Overview

3.1 Project Objectives

Based on the problem statement and the context for our client, we will now outline the project objectives

Over the first few weeks, we gained a clearer understanding of the client's requirements for developing an automated system in generating SAR narratives using a LLM. The client's focus in this project was not on filling out the SAR form but on creating the SAR narrative that goes into it, as the SAR narrative was a time-consuming task for them.

The client outlined the project objectives as follows:

- **Objective 1** - Identify and choose the optimal LLM model for local deployment.

  The *first objective* was to identify the most suitable LLM models for local operation. Given that the bank manages sensitive customer data and transactions, they aimed to avoid potential data leaks that could arise from utilizing cloud-based LLM solutions from providers like Amazon and Azure. Instead, they sought to develop an in-house solution that operates on their local machines and interfaces with their internal database. Additionally, the client preferred not to implement a system that would involve transmitting their data through vendor APIs or sharing it with any LLM or embedding model vendor. They clarified that their approach would not include training an LLM but rather using a pre-trained LLM to generate the SAR narrative. Our task was to suggest the LLMs that best fit their requirements and could be deployed locally. The client prioritized accuracy as the key factor in assessing the model's performance.

- **Objective 2:** Improve the model's efficiency through the integration of a GPT index (such as LlamaIndex) and using Retrieval-Augmented Generation (RAG) to search relevant documents and enhance the generated narratives.
  The *second objective* focused on examining the benefits of RAG, coupled with LlamaIndex, for sifting through Alert narratives and other documents, including regulatory information, to produce the SAR narrative. The researchers aimed to understand how to create SAR narratives with LLMs employing RAG that align closely with human-level writing.
  The client's purpose was to craft a narrative explaining why a human evaluator deems a

transaction suspicious or not. The goal was not to detect whether a transaction was suspicious or not, as this detail is already captured in the alert narrative.

- **Objective 3:** Establish a local MySQL server database linked to a Jupyter Notebook or Google Colab environment. This database will store customer, transaction, and alert details, including alert narratives.

  In this *third objective*, we aimed to create a local server database featuring tables on KYC, transaction detection, and alert information. Although the client initially requested a local MySQL server setup, we suggested using a PostgreSQL server instead due to Llama Index's enhanced functionality for PostgreSQL. We developed our solution with a PostgreSQL server in a Visual Studio virtual environment, which we then replicated to operate on Google Colab. The client desired a straightforward deployment solution to facilitate database connectivity between the Jupyter Notebook or Google Colab Notebook and an LLM.

  Initially, the client proposed using synthetic data generated by ChatGPT for the project's database tables. However, due to the subpar quality of the data generated by ChatGTP, they opted to provide us with synthetic data based on hypothetical scenarios. The client supplied table column names and data reflecting some attributes from their database, which are utilized for composing alert and SAR narratives. We then populated our tables using the data points given. Considering the project's scope and time constraints, the client created a CSV file containing five typologies (rules) for four customers, detailing KYC, transactions, detection, alerts, and other information. The rules included are 'Cash Structuring,' 'Rapid Fund Transfer,' 'Concentration Account,' 'Large Wire from High Risk Jurisdiction,' and 'New Account Rule.' (see Figure 3) *(Please see file 'GWU Tables and Rules_20240930_v2.csv' attached with this project)*. Further details on the dataset are included in the Database section of the report.

- **Objective 4:** The LLM will extract data from this database to create SAR narratives that include comprehensive details.
  The *fourth objective* aimed to link the database with the RAG and LLM during the generation of the SAR narrative, as it depended on customer, alert, and transaction information stored in the database.

- **Objective 5:** Create a format in Jupyter Notebook or Google Colab that effectively presents SAR narratives in a clear and accessible way.
  The *fifth objective* involved displaying the code for generating the SAR narrative in a comprehensible format, complete with explanations to assist users in understanding its functionality. The client imagined a system where user inputs, such as Alert-ID or

customer-ID, would produce a SAR narrative. Their core focus was on the quality of the output rather than the integration of the proposed solution with their various internal systems.

- **Objective 6:** Analyze the occurances of hallucinations in creating accurate narratives. The *sixth objective* focused on understanding the level of hallucinations present in the output and exploring methods to mitigate them.

- **Objective 7:** Fine-tuning the LLM model for optimal outcomes. The seventh objective focused on refining the LLM to achieve the best possible results, specifically for optimal narrative generation.

3.2 Project Overview

After understanding the client's objectives and the problem statement, we put together a plan of action. At the client's request, we employed a Gantt Chart to outline the project plan based on meeting the project objectives.

Our initial focus was on creating a secure, locally deployed solution for generating SAR narratives. First, we identified locally installable LLMs suitable for this project. We needed a model with a smaller size, ideally between 3 to 15 billion parameters, as anything larger would demand significantly more RAM and GPU resources than standard laptops can support. We were constrained by our available computing resources, as our team members had laptops with either 40GB or 16GB of RAM. However, even these computing resources were limited in their ability to efficiently run these LLMs of these sizes. Consequently, we opted for a 3 billion parameter model for local installation and a 12 billion parameter model that could be run on Google Colab Pro using T4 GPU. We narrowed down our model choices by evaluating their reasoning capability, popularity, performance, and available online evaluations (Model & API Providers Analysis Artificial Analysis, n.d.)(Hugging Face, n.d.) (https://huggingface.co/spaces/mteb/leaderboard). We assessed and chose models based on their size, performance, and accuracy metrics, as these were the client's requirements. Based on our preliminary research, we identified a range of LLMs that could be used - Llama 3.2 3b, Llama 3.1 8b, Llama 2 7b, Mistral Nemo 12B, Mistral 7B, Danube 3 4b, Phi 3.5 mini and MPT 7B(Databricks).

In addition, we needed to verify that the selected LLM was compatible with the libraries and functions of the LlamaIndex framework we intended to use. Subsequently, we planned to establish a PostgreSQL database linked to both the Jupyter Notebook and the Google Colab environment, with the Alert narrative document uploaded separately. The LLM would pull data from this database alongside the alert narrative to craft SAR narratives filled with essential

details. This database has been successfully integrated with both the Jupyter Notebook and Google Colab, housing critical information on customers, transactions, and alerts. With the database operational, we enhanced the LLM's ability to extract and apply this data in generating thorough SAR narratives. Additionally, we sought to gather and utilize regulatory information from online platforms to improve the SAR outputs. To further advance the system's functionalities, our goal was to incorporate the Retrieval-Augmented Generation (RAG) capabilities of the LlamaIndex framework, significantly boosting document processing efficiency and the quality of the generated content narratives.

For the user, we created a clear and accessible code format within Jupyter Notebook and Google Colab, making it straightforward for bank personnel to generate and review SAR narratives. Throughout the development process, we carefully monitored and assessed the prevalence of hallucinations in the generated narratives, implementing measures to ensure accuracy and reliability. We noted down prompts that led to more hallucinations and the revised prompts and LLM parameters that helped reduce these hallucinations.

Subsequently, we planned to fine-tune the parameters of the LLM to determine tuning which parameters aided in optimal narrative generation. We recorded all LLM parameter settings used for each successful and unsuccessful prompt. Throughout the project, we maintained a strong focus on quality control, implementing robust measures to prevent hallucinations in the generated narratives and fine-tuning the model to optimize output accuracy.

This can be viewed in the file (‘*Approach 1 - prompts | Output’* of the file *‘Other Methods 4 (3 LLMs) and Approach -1.xlsx’*) AND approach 2 prompt output and configuration shared with this project.

For the data infrastructure, we made a strategic decision to upgrade from the initially requested MySQL to a PostgreSQL database system, which offered better compatibility with LlamaIndex. We populated this database with synthetic data covering five essential typologies: Cash Structuring, Rapid Fund Transfer, Concentration Account, Large Wire from High Risk Jurisdiction, and New Account Rule. The development environment was established using Jupyter Notebook and Google Colab, where we created a user-friendly interface that allowed bank personnel to generate comprehensive SAR narratives by simply inputting Alert-IDs or Customer-IDs.

Our approaches prioritized the quality of narrative generation over system integration aspects, as per the client's requirements. We ensured that each component of the system – from the database infrastructure to the user interface – was designed with potential future scalability in mind, while never compromising on the core requirements of data security and narrative accuracy. The final deliverable provided the client with a comprehensive, secure, and efficient solution for their SAR narrative generation needs.

To grasp AI technologies, Section 4 below- AI Technologies Overview will summarize essential concepts and terms in this area.

4. AI Technologies Overview

4.1 Large Language Models (LLMs)

4.1.1 What are LLMs?

Large Language Models (LLMs) are advanced AI systems that process and generate human-like text. These models, trained on vast datasets containing billions of words, can understand context, generate coherent narratives, and answer questions that closely resemble human communication. LLMs are critical in applications requiring natural language understanding and generation, such as SAR narrative generation, where clarity and compliance are paramount.

4.1.2 How LLMs Work

LLMs operate on the principle of deep learning, particularly leveraging transformer architectures. These models predict the next word sequentially by analyzing patterns in large text corpora. Key components include:

- **Tokenization:** Text is broken down into smaller units (tokens) for processing.
- **Attention Mechanisms:** The model assigns different levels of importance to words in a context to generate accurate outputs.
- **Pretraining and Fine-tuning:** Models are trained on general datasets and fine-tuned on specific datasets to tailor them to particular tasks, such as SAR narrative generation.

4.1.3 Model Options (Llama, Mistral, Danube):

- **Llama (Meta's Large Language Model):** Known for its efficiency and ability to run locally, it is ideal for data-sensitive environments. (Appendices 9)
- **Mistral (NeMo 12B):** Its high accuracy and strong reasoning capabilities make it suitable for generating detailed SAR narratives with minimal errors. (see Appendices 9)
- **Danube (3-4B):** Lightweight and optimized for personal GPU setups, acting as a fallback model for resource-constrained scenarios. (see Appendices 9)

4.1.4 Limitations and Considerations

- **Resource Intensity:** High memory and computational power requirements.
- **Hallucinations:** Risk of generating incorrect or fabricated information.
- **Regulatory Risks:** Ensuring outputs align with AML and SAR regulatory standards.

- **Data Sensitivity:** Requires careful handling of sensitive financial data to maintain compliance and privacy.

4.2 Retrieval-Augmented Generation (RAG)

4.2.1 What is RAG?

Retrieval-augmented generation (RAG) combines retrieval-based and generative AI techniques to enhance output quality and contextual relevance. In RAG, external structured or unstructured datasets are queried in real time, and the retrieved data is integrated into the model's generative process.

4.2.2 RAG Architecture and Workflow

The RAG process involves:

1. **Data Retrieval:** Querying structured (e.g., databases) and unstructured (e.g., PDFs) data sources.
2. **Embedding Models:** Converting data into vector representations for efficient similarity searches.
3. **Fusion with Generative Models:** Integrating retrieved data into the model's narrative generation process, ensuring outputs are contextually enriched and accurate.

4.2.3 Benefits of RAG for SAR Generation

- **Enhanced Accuracy:** Ensures SAR narratives incorporate relevant details from structured and unstructured data.
- **Compliance Alignment:** Supports the integration of regulatory guidelines into the SAR generation process.
- **Contextual Depth:** Enriches narratives with external data, improving their usefulness for law enforcement and regulatory authorities.
- **Efficiency:** Reduces manual effort by automating data retrieval and narrative synthesis.

4.2.4 Implementation Considerations

- **Data Integration:** Ensuring seamless interaction between structured and unstructured data sources.
- **Model Tuning:** Fine-tuning retrieval and generation components for optimal performance.
- **Resource Management:** Balancing computational demands to maintain efficiency in local setups.

4.3 Embedding Models

4.3.1 Purpose and Functionality

Embedding models convert textual data into numerical vector representations. These vectors capture semantic relationships between words, enabling efficient searches and comparisons within large datasets. In SAR generation, embeddings facilitate retrieving relevant data for RAG workflows.

4.3.2 Vector Representations in AI

Embedding models represent data points in high-dimensional vector spaces. Similar data points are positioned closer together, allowing for rapid similarity-based searches. This ensures that retrieved data aligns closely with the narrative's context for SAR generation.

4.3.3 Integration with RAG

Embedding models are critical in RAG systems for:

- **Query Matching:** Ensuring the model retrieves the most relevant structured and unstructured data.
- **Semantic Searches:** Enhancing the model's ability to understand and incorporate contextually appropriate data.
- **Vector Database Compatibility:** Supporting integrations with tools like pgvector for efficient data retrieval.

4.3.4 Benefits of Embedding in SAR Generation

- **Improved Relevance:** Embeddings ensure data retrieval is contextually accurate.
- **Scalability:** Supports large datasets with efficient similarity searches.
- **Compliance Support:** Embeddings enhance the model's ability to retrieve regulatory and compliance-related data.
- **Enhanced narrative Quality:** This feature enables the generation of detailed and context-rich SAR narratives, reducing manual workload and improving regulatory adherence.

5. Overview of the Solution Process

An overview of the Initial methods we had attempted is described in the Appendices (see Appendices 2). The **overview of the Solution Process has been visualised in** *(Appendices 3)*

5.1 Database Setup and Integration

The database structure is designed to streamline the SAR generation process, ensuring compliance with AML regulations while meeting the client's operational needs. It enables detailed tracking of customer profiles, accounts, and transactions, providing a clear view of financial activities. Suspicious activities are monitored through detections and alerts, allowing for targeted reporting and effective risk management. SAR narratives consolidate insights from multiple data points, ensuring clarity and compliance with regulatory standards. The design supports scalability and flexibility to adapt to evolving business and compliance requirements, as outlined in the client's provided Excel reference (view the file).

5.2 Approach 1

This approach employs Mistral Nemo LLM and an embedding model to generate SAR narratives. The process begins by connecting to a PostgreSQL database to gather relevant information from queries. The system then processes an Alert Narrative PDF, splitting it into logical sections using a parser tool in Llama index. The SAR generation follows a five-paragraph structure: an introduction (static), transaction details (dynamic), customer KYC and research findings (dynamic), SAR recommendations (dynamic), and a conclusion (static), which are generated using Zero-shot learning. To minimize hallucinations and optimize performance, only the dynamic sections utilize Retrieval-Augmented Generation (RAG) through separate vector indices.

5.3 Approach 2

This approach integrates structured PostgreSQL transaction data and unstructured PDF guidelines to generate structured SAR narratives. It uses an embedding model for text representation and a re-rank model to prioritize relevant information. Zero-shot techniques provide exact outputs, while chain-of-thought prompting ensures logical reasoning to guide narrative construction. The output is structured into SAR sections: Introduction, Customer, Patterns, Violations, and Conclusion.

6. Database implementation

6.1 PostgreSQL Implementation

PostgreSQL was selected for its advanced querying capabilities, support for JSON/JSONB, and

ability to integrate vector search (via pgvector) to enhance retrieval processes for SAR generation. The database is hosted locally to ensure data security, adhering to the client's requirement of avoiding cloud-based solutions. PostgreSQL also supports scalability, which is essential for managing large volumes of structured data tied to suspicious activity monitoring and reporting. *(See figure:2.)*

6.2 Schema Design

The schema is structured to efficiently manage and link customer profiles, accounts, transactions, detections, alerts, and SAR narratives. Key highlights of the schema include:

- **Customer and Account Tables**: Track customer details, accounts, and expected activity levels.
- **Transaction Table**: Stores transactional data, including type, direction, and associated accounts or customers.
- **Detection and Alert Tables**: Monitor suspicious activities, rules violated, and analyst feedback.
- **SARNarrative Table**: Consolidates structured data into comprehensive, compliance-ready SAR narratives.
- The schema design aligns with AML requirements, supporting flexible queries and structured reporting. *(See Appendices 4.)*

6.3 Security Configuration

1. **TLS Encryption**
   All database connections are encrypted using TLS to secure data in transit, with regular certificate updates to maintain security.
2. **Access Control**
   Role-based access is enforced with unique usernames and strong passwords for authorized personnel only, with periodic password updates.
3. **Audit Logging**
   Automated logging tracks all database activities, including user access and data changes, with alerts for unusual actions to ensure accountability and security.

7. Methodologies

7.1 Approach 1

7.1.1 Process Overview

Based on insights from the section titled ‘other methods tried’ (Refer to Appendices 5), we opted for a new strategy by directly processing the static components of the SAR text from the database using placeholders within a template. This will be integrated with parts of the Alert narrative document through RAG before submission to the LLM. We set up this method for all five rules requested by the client: ‘Cash Structuring’, ‘Rapid Movements of Funds’, ‘Large Wire to High-Risk Jurisdiction’, ‘Concentration Account’, and ‘New Account Rule’. The steps for implementing this approach are illustrated in the Appendices 2.

The strategy behind this approach was to create a minimum viable solution where the LLM-generated SAR narrative is not hallucinating, not misleading, and providing reliable information, in a natural language flow as close as possible to the structure and meaning of the sample SAR narrative shared by the client. Due to computing resource limitations, including GPU and RAM constraints on our personal computers, we used Google Colab Pro to run the code for these models. We utilized a T4 GPU with a high RAM setting on Google Colab Pro. This approach has been developed based on the scope and bounds of this project and the information shared with regard to this project.

In this approach, we utilized the LLM *'Mistral-Nemo-Instruct-2407'* (Mistral Nemo) alongside the embedding model *'sentence-transformers/all-mpnet-base-v2'*. We selected the all-mpnet-base-v2 embedding model from Sentence Transformers due to its superior quality, as evidenced by its top score for encoding sentences, conducting semantic searches, and achieving the highest average performance among all Sentence Transformer models. (https://www.sbert.net/docs/sentence_transformer/pretrained_models.html)
Accompanying the project deliverables is a comprehensive step-by-step guide detailing the methodology and code execution. (*Please refer to the file Methodology_explanation_Mistral_nemo.pdf attached with this project)*

7.1.2 Understanding the structure of the SAR report and Alert narrative

We started by examining the sample SAR report shared by the client for customers ‘C-1’ and ‘C-2’, who had alerts ‘A-1’ and ‘A-2’ respectively. *(Please refer to the attached files LLM-SAR.docx and LLM SAR A-2.docx attached with this project).*

Upon reviewing the sample SAR Report, we discovered that both shared samples (LLM SAR & LLM SAR A-2) adhere to a specific template. We identified five key sections within the SAR. Both Paragraph 1 and Paragraph 5 maintained a consistent template structure, varying only in the customer's name, location, transaction amount, and the specific rules that were violated. Conversely, Paragraphs 2, 3, and 4 were entirely dynamic, containing information specific to each customer and case. According to the current manual process, the SAR report is compiled based on the analyst's remarks within the Alert narrative. Below are the sections details:

- **Introduction** *(Paragraph 1)***:** containing details of the bank filing the report and a summary of alerts for which the bank is filing this SAR. **(Static)**
- **Transaction details** *(Paragraph 2)***:** Describing transaction details and dates on which all the incoming and outgoing transaction alerts have been raised (**Dynamic)**
- **Customer's KYC and Internal/ External research** *(Paragraph 3)* - investigations which contradict validation & reasoning for the above transactions. **(Dynamic)**
- **SAR recommendation** *(Paragraph 4)***:** Why a SAR is being filed based on **(Dynamic)**
- C**onclusion (***Paragraph 5):* Bank's Contact Details for inquiries on the SAR: **(Static)**

Upon reviewing the supplied Alert narrative given to , we identified a template structure that can be distinguished by their headers. Following are the Alert narrative section headers before adding two headers:

- Alert_header - starts with "ALERT narrative"
- Focal_entity - starts with "Focal Entity"
- Determination - starts with "Determination /Rationale:"

The "Determination/Rationale:" also has a sub-section that starts with "Cash Structuring $10k", we will name this "Suspicious_activities:".

We need an effective way to parse the Alert Narrative document. There are different types of node parsers available in the llama index framework. After experimenting with several options, we found that the 'SentenceSplitter' node parser from the text_splitter library in the llama Index framework is the most beneficial for our needs (Node Parser Modules - LlamaIndex, 2021). Sentence Splitter processes the text while preserving context through chunk overlaps and adjustable chunk size parameters. Each section is organized in a mapping structure that includes its complete text and chunks, facilitating easy access to specific parts of the document later. We manually add two section headers into the Alert Narrative document: "Transaction_details:" and "sar_recommendations:". Following are the Alert narrative section headers after adding two additional sections headers (Please see See Appendices 8):

- Alert_header - Starts with "ALERT narrative"
- Focal_entity - starts with "Focal Entity"
- Determination - starts with "Determination /Rationale:"

- Suspicious_activities - starts with “Cash Structuring $10k”
- **Transaction_details** - starts with “Transaction details:”
- **sar_recommendations** - starts with “SAR Recommendation:”

7.1.3 Designing Prompts 1,2, 3, 4 and 5

After preparing the materials we need for the SAR report, we need to decide and design the structure of the prompts. We employ a template format for Paragraph 1 and Paragraph 5, utilizing ‘Prompt_1’ and ‘Prompt_5', respectively.Since these sections are static components of the SAR, automating SAR generation may not necessarily require them to be produced via RAG. Moreover, one could argue that using RAG for these sections could waste computational resources and power, leading to increased costs. As a bank, we recognize the necessity for accuracy in the SAR reports, and processing this section through RAG could result in inaccuracies. Therefore, using placeholders and static text appears to be the most effective method for creating these report sections, as this strategy avoids potential pitfalls and hallucinations.

- ‘Prompt_1’ - placeholder data and aggregated data from table in database are filled into the template.
- ‘Prompt_5’- placeholder data from the tables in the database are filled into the template.

Upon reviewing the Alert narratives, we found that the document adheres to a specific template, allowing it to be divided and segmented by its sections instead.

- ‘Prompt_2’ - Summarizes transaction details chronologically.
- ‘Prompt_3’- Formats and extracts the KYC information and Violations, and adds the internal/external research findings.
- ‘Prompt_4’- Based on the alert narrative, it lists suspicious activities and their implications.

You can view the exact structure and contents of these five prompts in the ipynb file for Approach-1. (*Please see section 5. Generating SAR Report in Approach_1_with_Mistral_Nemo.ipynb attached with this project).* We adjusted the parameters of the LLM to achieve the intended output. You can find a detailed description of the parameters we adjusted in the code explanation document for Approach 1*(Please see file Methodology_explanation_Mistral_nemo.pdf attached with this project)*

In this approach we used zero-shot learning.

7.1.4 Results:

We merged the outputs from the five prompts into one cohesive report. The result is a structured SAR report that integrates database information, document analysis, and formatted responses in a standardized format for regulatory reporting. The outputs for Approach-1 have been included with this project's deliverables. *(Please see tab 'Approach 1 - prompts|Output'* in file *'Other Methods 4 (3 LLMs) and Approach 1 - All prompts and output.xlsx'. The outputs are in the column 'Results SAR-1' and 'Results SAR-2').*

The final output is a structured SAR report organized into the following sections:

- Intro
- Transaction details
- *KYC Violations & Research:*
- SAR Recommendations
- Conclusion

This is a SAR created for Customer_id 'C-2', Alert_id 'A-2' from the database, along with the alert narrative 'A-2 Alert Narrative v2' provided by the client. The client assessed the SAR, giving it a feedback score of 9/10. *(Please see the original SAR shared by the client in file LLM SAR A-2.docx.)*

Intro:
LLM Bank New York("LLM NY") is a wholesale branch of LLM Bank Ltd, a commercial bank located in mainland China. LLM NY is filing this Suspicious Activity Report ("SAR") (Internal SAR Reference Number) A-2, to report 6 transactions totaling $5,628,940.80 between 09/02/2024 and 09/15/2024.

Transaction details:
Between 9/2/2024 and 9/10/2024, RDF Plumbing received three wires totaling $763,228 in account ACC-2, originating from US Processing in the US, Cos Cob Fishery in the US, and HK Industries in Hong Kong. Additionally, RDF Plumbing received an ACH credit of $179,000 from JD Import and Export in the UK.

Subsequently, on 9/14/2024, the customer sent $2,286,712.80 to related account ACC-3. On 9/15/2024, a wire for $2,400,000 was sent from ACC-3 to IRS Legal Services in China.

On 9/23/2024, RDF Plumbing received a wire for $1,598,564 from Palmetto Translation Services, also located in China.

KYC Violations & Research:
LLM NY KYC information identifies RDF Plumbing as a plumbing service. External research was unable to conclusively identify a line of business for Cos Cob Fishery, HK Industries, JD Import and Export, IRS Legal Services, and Palmetto Translation Services.

SAR Recommendations:
These transactions are being reported due to the following:
(1) the apparent use of a concentration account,
(2) a rapid movements of funds, and
(3) transactions with no apparent economic or business purpose

Conclusion:
This SAR pertains to LLM NY Case No A-2. For inquiries, please contact Donald J. Orange Chief Compliance Officer and Chief BSA/AML Officer (646-555-5555) or donaldjorange@llmbank.com or Alyn Mask, General Counsel (646-555-5555) or alynmask@llmbank.com.All supporting documentation is maintained by the Financial Crime Compliance Department at LLM NY.

Please refer to

7.1.5 Observations:

In this approach, we recorded the time for each SAR generated alongside the parameters utilized and the precise structure of the prompt. This allowed us to monitor the time taken per SAR and analyze how the context length of the Alert Narrative influenced the SAR generation time. By methodically tracking both the LLM parameters used for SAR generation and the query engine parameters for each individual prompt, we could clearly identify how significant changes in prompt language impacted the generated output.

In Approach 1, all SARs were executed and tested on Google Colab Pro, using a T4 GPU. The execution of all functions and prompts in one go took between 6 minutes and 30 seconds to 10 minutes, depending on the amount of detail in the 'Transaction_details' and 'sar_recommendation' sections in the provided Alert Narrative PDF. An increase in the number of instructions (or tokens) as input leads to longer output generation times, which doesn't necessarily improve output quality.

Other observations were as follows:

- To minimize hallucinations and ensure that the LLM follows the desired text format, we needed to reiterate specific instructions in the prompts. This helped achieve the formatting we wanted from the LLM. For example:
    - "…Follow the instructions in Requirements"
    - "Do not add your own knowledge"

    - "…write logical sentences with a natural flow"
    - "Use exact formatting:
        - Number each reason with parentheses: "(1)", "(2)", etc.
        - Start each rule violation with "Apparent" "

- We observed that dates were particularly difficult for the LLM to follow. Hence we used:
    - <date> which efficiently translated the date to the format 'mm/dd/yyyy'
- We specifically observed that the mistrals model comprehended some keywords more effectively than others in delivering the intended output. For example:
    - Using "Summarize" gave the output closest to what the client requested, but using "Write a comprehensive summary" and "Generate a comprehensive summary" performed poorly.
- Based on the time-taken to generate outputs we observed that larger text in the prompt took more time to complete. The time taken for SAR generated in Approach-1 can be viewed in the files attached with this project. (Please see '*Approach 1 - prompts | Output'* of the file *'Other Methods 4 (3 LLMs) and Approach -1.xlsx')*.
- Optimizing prompts is key for cost-effective production. This also conserves energy and reduces resources while maintaining the original meaning of the question.
- There are four types of response_modes used to generate responses in query engines. We tested all four methods and found that 'tree_summarize' and 'compact' best suit our use case. We chose 'compact' because 'tree_summarize' takes longer to run the query but generates the same output.

7.1.6 Client-feedback:

This approach was developed based on insights gained from 16 earlier SARs created using its predecessor method. We systematically utilized client feedback on prior versions. All SARs were generated exclusively with the data provided by the client, including Alert Narratives and Sample SAR Reports, without incorporating any scenarios not supplied by the client. The client rated SAR_17 and SAR_18 at 8/10, SAR_19 at 7/10, and SAR_20 at 9/10. *(To view the client feedback for each SAR generated in approach 1, please refer to the attached file 'Generated SAR_17-20 Nov 26 Jeff - ABC ratings - AAR'. Click on Review and Simple Mark-up to access SAR_19 and SAR_20)*

The feedback on SAR_17 and SAR_18 included minor adjustments, such as removing repeated words, changing word capitalizations, adding a '$' sign before amounts, and standardizing the dates in both the 'static' and 'dynamic' sections. For example:

The Following :

213,000.00 between 2024-09-02 and 2024-09-14

should be changed to:

$213,000.00 between 9/2/2024 and 9/14/2024.

And to change the capitalization of words and to remove repetition of the word ‘Apparent’ word in the sar_recommendation section below. For example, From the following:

‘(1) Apparent Cash Structuring $10k, (2) Apprarent Large Wire to High Risk Jurisdiction, (3) Apprarent Rapid Movements of Funds, and (4) transactions with no apparent economic or business purpose.’

to:
‘(1) Apparent Cash Structuring, (2) a large wire to a high-risk jurisdiction, (3) a rapid movement of funds, and (4) transactions with no apparent economic or business purpose.’

We incorporated the feedback into the revised prompts for SAR_19 and SAR_20 and obtained a score of 9/10 for these SAR generated. This is shown in the results section above.

The clients stated stated that SAR_19 and SAR_20 were “really good” and the best amongst the ones shared for approach 1 previously.

7.1.7 Recommendations:

This method illustrated how to implement a minimum viable solution tailored to the client's current needs. As a result, the following next steps emerged from the insights gained through this approach:

- To explore other parameters of the LLM and the query engine, aiming to optimize prompts in prompt engineering. Some additional parameters to leverage include: ‘context_template’, ‘filters’, along with specific ones like ‘temperature’, ‘separator’, ‘truncate’, ‘response_kwargs’,‘template’, and ‘streaming’.

- Investigate and evaluate the performance of the 'accumulate’ and ‘refine’ response modes of the query engine parameters for this use case.

- Evaluate and experiment with different combination of rules in Alert Narratives, along with Customer and Alert numbers, using the same criteria prompts.

- Explore strategies to shorten the context length for RAG prompts while keeping the output unchanged.

7.2 Approach 2

7.2.1 Process Overview

The process for generating structured Suspicious Activity Reports (SARs) is organized into several clear and straightforward steps:

7.2.2 Data Preparation:

The process begins with two primary sources of data:

- **PDF Guidelines:** These are pre-defined documents segmented into specific sections, such as Introduction, Customer Information, Deposit Patterns, and Violations. They provide the structure and content template for the SAR report.
- **PostgreSQL Database:** This stores structured data, such as transaction records, that can be queried to extract relevant information.

7.2.3 Processing Guidelines and Database:

- **PDF Processing:** Text is extracted from the PDF documents and split into smaller, manageable chunks for further use.
- **Database Processing:** SQL queries extract specific data points from the database. This structured data is prepared to complement the unstructured text from the PDF guidelines.

7.2.4 Embedding and Retrieval:

- The extracted text and selected query data are converted into vector embeddings using the all-MiniLM-L6-v2 model.
- These embeddings are passed through a cross-encoder (ms-marco-MiniLM-L6-v2) to perform re-ranking and ensure relevance.
- The refined embeddings are stored in a vector store.
- The stored embeddings are retrieved and prepared for integration into the SAR generation process.

7.2.5 Prompting Strategies:

Retrieved data and context are processed using one of three prompting strategies:

- **Zero-Shot (Exact Format + RAG):** Uses a single prompt to generate each SAR section.
- **Chain of Thought (Exact Format + RAG):** This approach breaks the generation process into multiple steps, ensuring detailed and structured responses.
- **Chain of Thought (Guided Format + RAG):** Guides the model through sequential prompts for comprehensive and formatted responses.

7.2.6 Text Generation:

The refined context and prompts are passed to the LLaMA 3.2-3B model, which generates the SAR content. The output is organized into five key sections: Introduction, Customer Information, Patterns, Violations, and Conclusion.

The diagram below illustrates the process, which provides a detailed visual representation of each step. (see the full process in Appendices 6)

Please refer to the **Jupyter Notebook folder** in the submitted file **Approach-2.zip for more information and details**.

7.2.7 Result:

The SAR output is organized into five main sections:

- Introduction
- Customer Information
- Patterns
- Violations
- Conclusion

Here is a SAR generated for a single rule: cash structuring. It is for a customer with a client feedback score of 8 and was created using LLaMA 3.2-3B:

Introduction

LLM Bank New York Branch (LLM NY), a wholesale branch of LLM Bank Ltd. (China-based), is filing this SAR (Internal SAR Reference Number A-6) to report a series of structured cash transactions (SAR type: Cash Structured Transactions) totaling $80,000 conducted by Mickel Angelo between 2024-11-01 and 2024-11-10

Customer Information

During this period, Mickel Angelo, a resident of the US, made 10 deposits of $8,000 each into account ACC-6 at LLM NY. These deposits were made consecutively over 10 days, suggesting an effort to structure cash deposits below reporting thresholds.

Patterns

According to LLM NY's KYC information, Mickel Angelo's occupation is in Car Sales Commission, a sector not typically associated with cash-intensive transactions. No legitimate source of funds was identified for these deposits, raising concerns regarding the origin of the deposited funds.

Violations
This SAR is being filed solely on the grounds of apparent cash structuring, with indications that these deposits may be intended to evade regulatory reporting requirements.

Conclusion
This SAR pertains to LLM NY Case No. 2024-1234. For inquiries, please contact Donald J. Orange, Chief Compliance Officer and Chief BSA/AML Officer at 646-555-5555 or donaldjorange@llmbank.com, or Alyn Mask, General Counsel, at 646-666-6666 or alynmask@llmbank.com. All supporting documentation is maintained by the Financial Crime Compliance Department at LLM NY.

7.2.8 Client Feedback:

Structure and Format:

The SAR narratives are well-structured and reflect a natural tone, aligning closely with the expectations for professional SAR reports.

Fact Pattern Updates:

The model's difficulty in updating fact patterns, despite generating natural narratives, stems from several challenges. It lacks dynamic context update mechanisms, causing static details that fail to reflect changes, and exhibits weak fact-tracking capabilities, leading to incomplete narratives. Limited integration of structured and unstructured data further hinders its ability to accurately adapt to updates. Insufficient fine-tuning on SAR examples emphasizing dynamic changes and suboptimal performance of the Retrieval-Augmented Generation (RAG) system may also contribute to these issues. Refining prompts, enhancing fact management, optimizing the RAG system, and fine-tuning with dynamic examples can address these challenges, improving the model's ability to generate accurate and adaptable narratives.

Overall Impression:

The generated narratives are considered "very natural," indicating they are close to human-written SARs in quality and readability.

Please refer to the folder **SAR Outputs** in the submitted file **Approach-2.zip for more output and Client Feedback.**

7.2.9 Recommendations:

Human Feedback with Reinforcement Learning:

This process combines AI and human oversight to generate high-quality Suspicious Activity Reports (SARs) for the bank. AI (LLaMA 3.2) drafts SAR sections like "Customer Info" and "Violations," while banking experts review and provide feedback. The AI learns from this feedback, improving its accuracy and compliance over time.

The result is faster, more accurate, and compliant SAR generation, reducing manual effort and enhancing risk management.(see recommendation in Appendices 7)

8. Results and Performance Analysis - Based on client evaluations

Based on the client's insights, the key recommendation is as follows: The preferred model depends on the feasibility of implementing distinct Models/RAGs for each individual rule.

The Mistral model is more effective as a versatile model, capable of generating SAR narratives across various rules. The approach performs flawlessly on the rules:‘Cash Structuring,’ ‘Rapid Fund Transfer,’ ‘Concentration Account,’ ‘Large Wire from High Risk Jurisdiction,’ and ‘New Account Rule.’. The choice of which model to use largely hinges on the practicality of deploying separate models or RAGs for each rule. If that isn't practical, the Mistral model appears to be the better option. For a bank with numerous rules, the Mistral model would likely be the preferred choice.

The LLaMA model is better suited for this specific branch, as it performs exceptionally well on the cash structuring rule and can be independently configured with a Retrieval-Augmented Generation (RAG) system for each rule. Given the branch has a manageable number of rules, this approach is feasible and ensures optimal performance.

9. Risk Assessment

In this project, running the model locally significantly reduces the risk of data leaks compared to cloud-based solutions. However, it does not guarantee complete security, as certain risks remain. To address these concerns, the NIST Artificial Intelligence Risk Management

Framework provides valuable guidance on identifying and mitigating risks for the responsible use of AI.

9.1 Data Privacy Considerations

1. **Training Data Risks:**
    - Risk: Training data may inadvertently include sensitive or personally identifiable information (PII), leading to potential data breaches.
    - Impact: The model could memorize sensitive information, violating data protection regulations.
    - Example: Customer account details, transaction histories, or other sensitive information becoming part of the model's learned patterns**.**
2. **User Interaction Risks:**
    - Risk: Users may input sensitive data during SAR generation, which could be logged or cached unintentionally.
    - Impact: Unauthorized access to logs or cache could lead to data leakage.
    - Mitigation: Use alert IDs instead of sensitive details and disable logging wherever feasible**.**
3. **Temporary Storage Risks:**
    - Risk: Generated SARs or temporary files may be stored on unencrypted disks.
    - Impact: Exposure of confidential activity data to unauthorized access.
    - Mitigation: Encrypt all temporary files and restrict access to authorized personnel.

9.2 Model Limitations and Risks

9.2.1 Hallucination Risks

1. **Fabricated Content:**
    - Risk: The model may generate inaccurate data, such as fake transaction amounts or customer information.
    - Impact: Misleading SARs undermine credibility and violate regulatory standards.
    - Example: A fabricated transaction amount or entity included in a repor**t.**
2. **Template Sensitivity:**
    - Risk: Small changes in prompts can lead to significant variations in output quality.
    - Impact: Inconsistent SARs may challenge compliance and trustworthiness**.**

9.2.2 Information Omission Risks

1. **Incomplete SARs:**

    - Risk: Critical details may be omitted due to limitations in the training data or prompt structure.
    - Impact: Inadequate reports could fail to meet regulatory requirements.
    - Example: Missing transaction details, suspicious behavior patterns, or customer identifiers.
2. **Dependency on Input:**
    - Risk: The model relies solely on provided inputs, lacking contextual awareness or the ability to infer additional relevant information.
    - Impact: Reduced usability for cases requiring integrated external data.

9.3 Technical Challenges and Risks

1. **High Resource Requirements:**
    - Risk: Deploying and fine-tuning large models locally demands significant computational resources.
    - Impact: Increased costs for infrastructure and maintenance.
    - Mitigation: Optimize deployment with model quantization and distillation.
2. **Model Deployment Risks:**
    - Risk: Inadequate encryption or access controls in local deployments may expose sensitive data.
    - Impact: Higher risk of data breaches and operational downtimes.
3. **Cross-Model Incompatibility:**
    - **Risk:** Prompts optimized for one model (e.g., Mistral) may not perform well on others (e.g., OpenAI GPT).
    - **Impact:** Reduced adaptability across different platforms.
4. **Evaluation Challenges:**
    - **Risk:** Human evaluation of SARs may introduce bias or inconsistencies.
    - **Impac**t: Difficulty in establishing objective quality benchmarks.

9.4 Mitigation Strategies

Data Privacy:

- **Anonymization and Desensitization:**
    - Remove PII from training data.
    - Use synthetic or anonymized datasets during model training.
- **Secure Interaction Systems:**
    - Disable interaction logging or implement secure logging with auto-deletion features.
- **Encryption and Access Control:**

    - Encrypt temporary files and enforce strict access controls.

Hallucination Risks:

- **Strict Prompt Engineering:**
    - Use structured prompts to minimize irrelevant outputs.
    - Implement self-validation prompts for quality checks.
- **Validation Pipelines:**
    - Develop tools to flag potential hallucinations for human review.
- **Fine-Tuning:**
    - Train models on SAR-specific datasets to enhance accuracy.

Information Omission Risks:

- **External Knowledge Integration:**
    - Employ Retrieval-Augmented Generation (RAG) to access external data sources.
- **Human-in-the-Loop:**
    - Compliance officers should be involved in reviewing and completing SARs.

Technical Challenges:

- **Efficient Deployment:**
    - Use model optimization techniques, such as distillation, to reduce computational demands.
    - Consider cloud-based deployment for scalability.
- **Cross-Model Compatibility:**
    - Create adaptable workflows for multiple models.
- **Automated Evaluation Metrics:**
    - Use standardized metrics (e.g., BLEU, ROUGE) for consistent SAR evaluations.

10. Business Value Proposition

Every organization evaluates the business value before investing in new technologies or methods. At the Agricultural Bank of China New York, we recognize that the business value is equally critical as the implementation of new technology. To grasp the extent of the challenges and the costs associated with writing SARs, we reviewed available factual data. Reports from the Financial Crimes Enforcement Network indicate that banks incur approximately $37 per hour in SAR-related expenses. According to FinCEN's reports for the fiscal year 2023, banks file around 2.5 million SARs annually. Based on insights from our client, completing a SAR can take anywhere from 1 hour to 2 days (considering 16 working hours for 2 days). If the Agricultural

Bank of China New York files 10,000 SARs, the expense ranges from $370,000 to $5,920,000. Filing 20,000 SARs would increase the costs to between $740,000 and $11,840,000. This has been illustrated in Figure 1 in the Appendices.

Therefore, the business advantage for ABC NY lies in lowering the SAR expenses incurred by the bank. In addition to this reduction, banks can utilize LLMs for their ability to generate narratives. This solution also automates a labor-intensive process, allowing analysts who write SARs to prioritize other tasks that require their focus.

11. Conclusion and Recommendations:

To test the scalability of our solutions and universes, we recommend utilizing The George Washington University's High Performance Computing resources to run 70 billion parameter models. This testing will involve use cases with increased complexities, and we have found that optimizing prompts can help reduce the time taken to obtain outputs, making the process more efficient. It is also advisable to further fine-tune the parameters of the query engine used for retrieval-augmented generation (RAG) and examine how to fine-tune embeddings in the model to cater to specific use cases. Additionally, we should conduct other evaluations and benchmarks beyond human evaluation. In addressing Gen-AI Risk Management, we must consider risks associated with model integrity, such as confabulations, data privacy, and information security, and evaluate how to mitigate these risks in a production setting.

References

1. *FFIEC BSA/AML Assessing Compliance with BSA Regulatory Requirements - Suspicious Activity Reporting. (n.d.). Bsaaml.ffiec.gov https://bsaaml.ffiec.gov/manual/AssessingComplianceWithBSARegulatoryRequirements/04*
2. *Financial Crimes Enforcement Network. (n.d.). The SAR narrative guidance.*

https://www.fincen.gov/sites/default/files/shared/sarnarrcompletguidfinal_112003.pdf

3. *FinCEN. (2019). What is money laundering? Financial Crimes Enforcement Network. https://www.fincen.gov/what-money-laundering*
4. *Node Parser Modules - LlamaIndex. (2021). Llamaindex.ai. https://docs.llamaindex.ai/en/stable/module_guides/loading/node_parsers/modules/*

5. *What is a suspicious activity report? (2019). @Westlaw1.*

https://legal.thomsonreuters.com/en/insights/articles/what-is-a-suspicious-activity-report

6. *Financial Crimes Enforcement Network (FinCEN) Year in Review for FY 2023. (n.d.). https://www.fincen.gov/sites/default/files/shared/FinCEN_Infographic_Public_508FINAL_2024_June_7.pdf*
7. *Agency Information Collection Activities; Proposed Renewal; Comment Request; Renewal Without Change of Reports by Financial Institutions of Suspicious Transactions and FinCEN Form 111-Suspicious Activity Report. (2024, February 12). Federal Register. https://www.federalregister.gov/documents/2024/02/12/2024-02747/agency-information-collection-activities-proposed-renewal-comment-request-renewal-without-change-of*
8. *Alammar, J., & Grootendorst, M. (2024). Hands-On Large Language Models: Language Understanding and Generation. O'Reilly Media.*
9. *Bishop, C. M., & Bishop, H. (2024). Deep Learning: Foundations and Concepts (2024th ed.). Cambridge University Press.*
10. *Rothman, D. (2024). RAG-Driven Generative AI: Build custom retrieval-augmented generation pipelines with LlamaIndex, Deep Lake, and Pinecone. Packt Publishing.*
11. *GitHub. (n.d.). Prompt Engineering Guide. Retrieved December 5, 2024, from https://github.com/run-llama/llama_index*
12. *NVIDIA. (n.d.). Building RAG Agents with LLMs. NVIDIA Deep Learning Institute. Retrieved December 5, 2024, from https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-15+V1*

13. *NVIDIA. (n.d.). Introduction to Deploying RAG Pipelines for Production at Scale. NVIDIA Deep Learning Institute. Retrieved December 5, 2024, from https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-19+V1*
14. *Prompting Guide. (n.d.). Prompt Engineering Guide. Retrieved December 5, 2024, from https://www.promptingguide.ai/*
15. *GitHub. (n.d.). Example usage: LlamaIndex. Retrieved December 5, 2024, from https://github.com/run-llama/llama_index#-example-usage*
16. *LlamaIndex. (n.d.). Embeddings - LlamaIndex. Retrieved December 5, 2024, from https://docs.llamaindex.ai/en/stable/module_guides/models/embeddings/*
17. *h2oai/h2o-danube3-4b-base · Hugging Face. (2024, November 30). Huggingface.co. https://huggingface.co/h2oai/h2o-danube3-4b-base*
18. *nvidia/Mistral-NeMo-12B-Instruct · Hugging Face*. (2024). Huggingface.co. https://huggingface.co/nvidia/Mistral-NeMo-12B-Instruct
19. *h2oai/h2o-danube3-4b-base · Hugging Face*. (2024, November 30). Huggingface.co. https://huggingface.co/h2oai/h2o-danube3-4b-base
20. *Model & API Providers Analysis | Artificial Analysis*. (n.d.). Artificialanalysis.ai. https://artificialanalysis.ai/

Appendices

Figure 1.

FinCEN Total number of SARs in FY 2023 - by Banks	Cost – per hour for SAR expenses ($)			
2,500,000	37			
ABC NY- No. of SARs filed per month	Min Time Taken to write SAR - 1 hour	Max Time taken to write SAR - 2 days (16 working hours)	Min Cost ($)	Max Cost ($)
10,000	1	16	370,000	5,920,000
20,000	1	16	740,000	11,840,000
Larger Banks - JP Morgan				
40,000	1	16	1,480,000	23,680,000

Figure 2.

Table Name	Purpose	Key Columns
CustomerLineOfBusiness	Stores business line details	LobID (PK), LineOfBusiness
Customer	Represents customers	CustomerID (PK), CustomerName, CustomerLineOfBusinessID (FK), IncorporationCountryID (FK)
CustomerExpectedProducts	Lists expected customer products	ProductID (PK), ExpectedProduct
CustomerExpectedGeographies	Stores expected customer regions	GeographyID (PK), ExpectedGeography
CustomerProduct	Links customers to products	CustomerProductID (PK), CustomerID (FK), ProductID (FK)

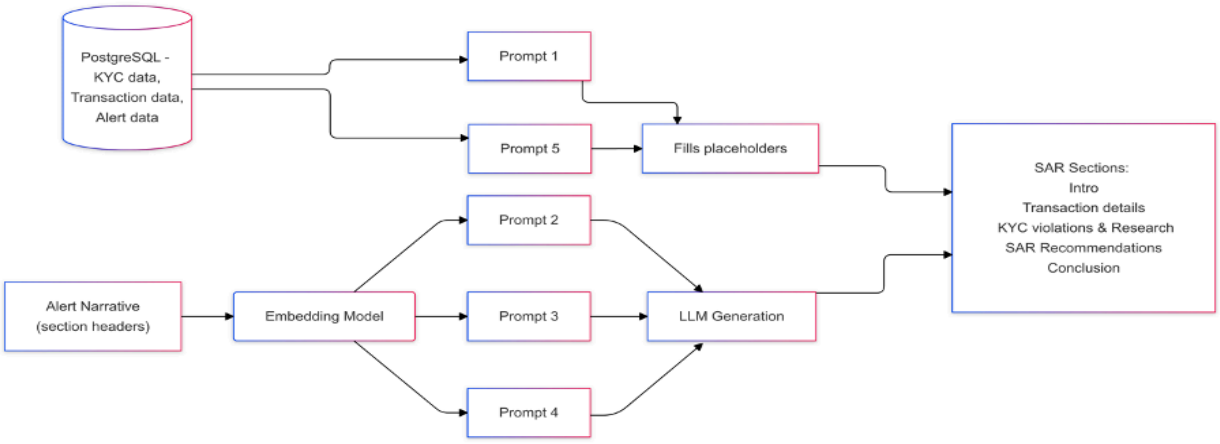
CustomerGeography	Maps customers to geographies	CustomerGeographyID (PK), CustomerID (FK), GeographyID (FK)
Account	Stores customer accounts	AccountID (PK), CustomerID (FK), DateOfOpening, AccountType, ExpectedIncomingActivity, ExpectedOutgoingActivity
Rule	Defines alert rules	RuleID (PK), RuleName, RuleDescription
Transaction	Tracks financial transactions	TransactionID (PK), TransactionDate, AccountID (FK), CustomerID (FK), Amount, Originator, Beneficiary
Detection	Logs suspicious activities	DetectionID (PK), DetectionDate, Resolved, ResolutionDate, InternalInvestigativeReference
Alert	Tracks generated alerts	AlertID (PK), AlertStatus, AnalystComments, AlertDate
DetectionTransaction	Maps detections to transactions	DetectionTransactionID (PK), DetectionID (FK), TransactionID (FK), RuleID (FK), CustomerID (FK), AlertID (FK)
SARnarrative	Stores SAR narratives	narrativeID (PK), CustomerID (FK), TransactionID (FK), DetectionTransactionID (FK), narrativeText, narrativeDate

Figure 3.

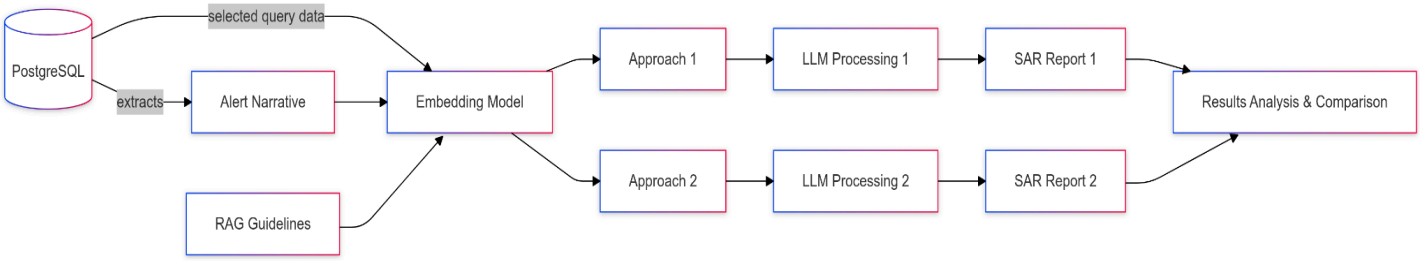
Rule	**Rule Description**
Cash Structuring $10k	Multiple cash deposits each under $10k that aggregate to more than $10k over a week.
Rapid Movements of Funds	Outgoing activity is within 20% of incoming activity above $100,000 within 30 days, excluding internal transfers.
Large Wire to High Risk Jursidiction	Wire over $100K to a high risk jurisdiction
Concentration Account	Customer sees transactions from 5 or more originators during the month. Also during the month, customer sends single transaction to beneficiary worth value within 20% of incoming activity.
New Account Rule	New account sees activity over expected activity during first month after account opening

Appendices 1

Appendices 2.

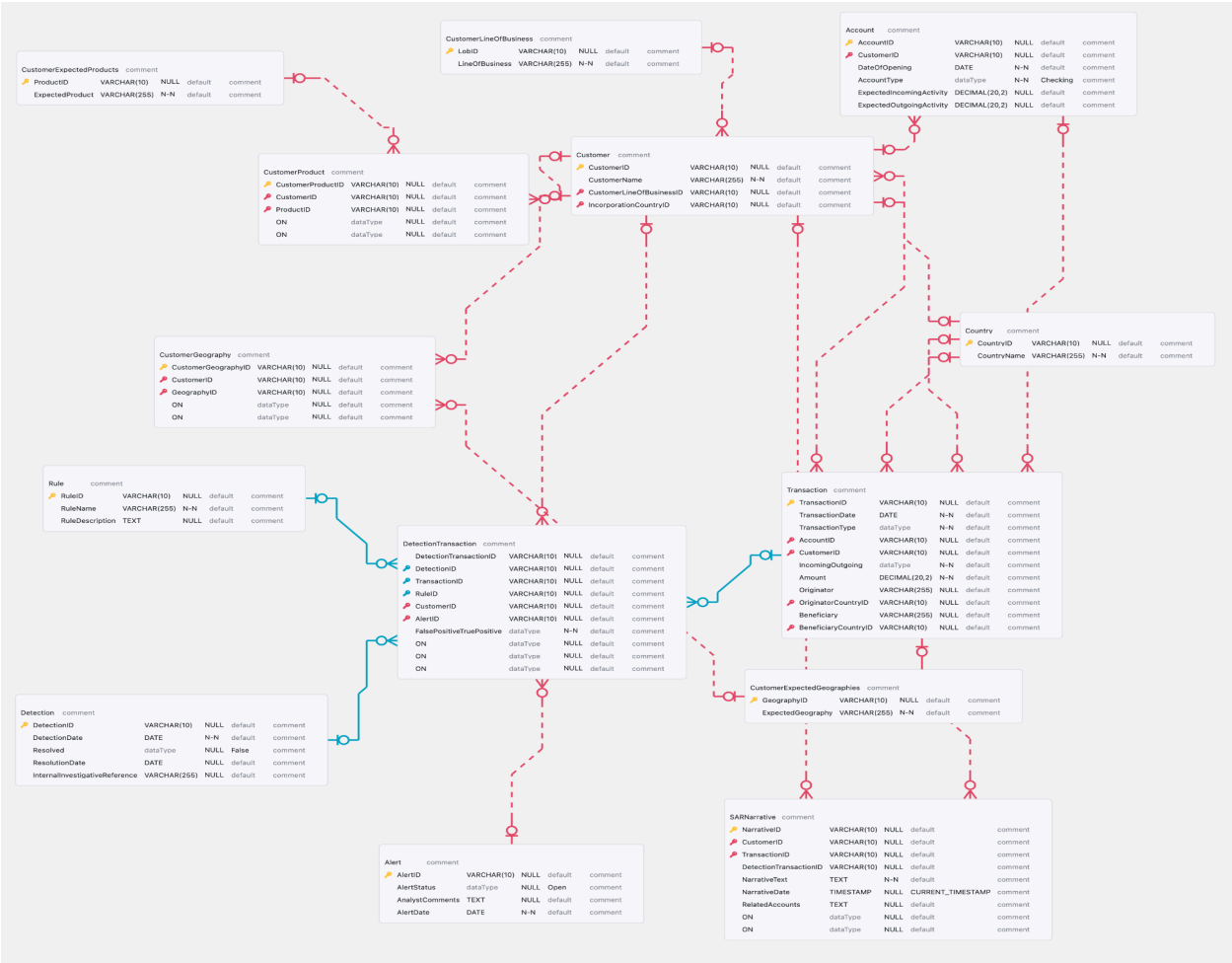


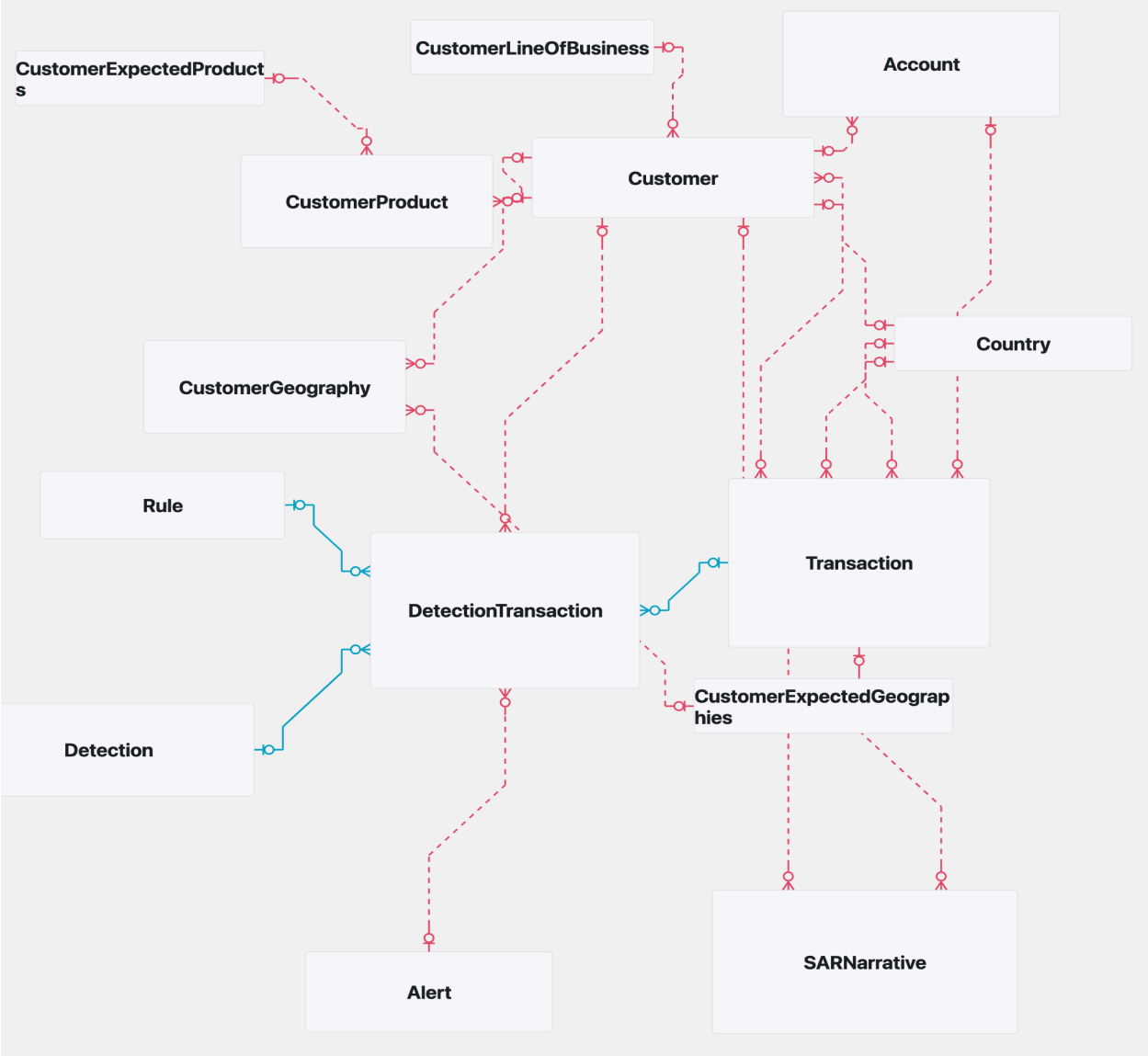
Appendices 3.



This diagram illustrates our solution workflow in a general overview where data is extracted from a PostgreSQL database and processed through an embedding model. Two different approaches are applied to generate SAR reports, which are then compared for results analysis and insights

Appendices 4.


CustomerLineOfBusiness comment
LobID VARCHAR(10) NULL default comment
LineOfBusiness VARCHAR(255) N-N default comment
Account comment
AccountID VARCHAR(10) NULL default comment
CustomerID VARCHAR(10) NULL default comment
DateOfOpening DATE N-N default comment
AccountType dataType N-N Checking comment
ExpectedIncomingActivity DECIMAL(20,2) NULL default comment
ExpectedOutgoingActivity DECIMAL(20,2) NULL default comment
CustomerExpectedProducts comment
ProductID VARCHAR(10) NULL default comment
ExpectedProduct VARCHAR(255) N-N default comment
CustomerProduct comment
CustomerProductID VARCHAR(10) NULL default comment
CustomerID VARCHAR(10) NULL default comment
ProductID VARCHAR(10) NULL default comment
ON dataType NULL default comment
ON dataType NULL default comment
Customer comment
CustomerID VARCHAR(10) NULL default comment
CustomerName VARCHAR(255) N-N default comment
CustomerLineOfBusinessID VARCHAR(10) NULL default comment
IncorporationCountryID VARCHAR(10) NULL default comment
Country comment
CountryID VARCHAR(10) NULL default comment
CountryName VARCHAR(255) N-N default comment
CustomerGeography comment
CustomerGeographyID VARCHAR(10) NULL default comment
CustomerID VARCHAR(10) NULL default comment
GeographyID VARCHAR(10) NULL default comment
ON dataType NULL default comment
ON dataType NULL default comment
Rule comment
RuleID VARCHAR(10) NULL default comment
RuleName VARCHAR(255) N-N default comment
RuleDescription TEXT NULL default comment
DetectionTransaction comment
DetectionTransactionID VARCHAR(10) NULL default comment
DetectionID VARCHAR(10) NULL default comment
TransactionID VARCHAR(10) NULL default comment
RuleID VARCHAR(10) NULL default comment
CustomerID VARCHAR(10) NULL default comment
AlertID VARCHAR(10) NULL default comment
FalsePositiveTruePositive dataType N-N default comment
ON dataType NULL default comment
ON dataType NULL default comment
ON dataType NULL default comment
Transaction comment
TransactionID VARCHAR(10) NULL default comment
TransactionDate DATE N-N default comment
TransactionType dataType N-N default comment
AccountID VARCHAR(10) NULL default comment
CustomerID VARCHAR(10) NULL default comment
IncomingOutgoing dataType N-N default comment
Amount DECIMAL(20,2) N-N default comment
Originator VARCHAR(255) NULL default comment
OriginatorCountryID VARCHAR(10) NULL default comment
Beneficiary VARCHAR(255) NULL default comment
BeneficiaryCountryID VARCHAR(10) NULL default comment
Detection comment
DetectionID VARCHAR(10) NULL default comment
DetectionDate DATE N-N default comment
Resolved dataType NULL False comment
ResolutionDate DATE NULL default comment
InternalInvestigativeReference VARCHAR(255) NULL default comment
CustomerExpectedGeographies comment
GeographyID VARCHAR(10) NULL default comment
ExpectedGeography VARCHAR(255) N-N default comment
SARNarrative comment
NarrativeID VARCHAR(10) NULL default comment
CustomerID VARCHAR(10) NULL default comment
TransactionID VARCHAR(10) NULL default comment
DetectionTransactionID VARCHAR(10) NULL default comment
NarrativeText TEXT N-N default comment
NarrativeDate TIMESTAMP NULL CURRENT_TIMESTAMP comment
RelatedAccounts TEXT NULL default comment
ON dataType NULL default comment
ON dataType NULL default comment
Alert comment
AlertID VARCHAR(10) NULL default comment
AlertStatus dataType NULL Open comment
AnalystComments TEXT NULL default comment
AlertDate DATE N-N default comment



Appendices 5.

———————— other methods tried ----------

After reviewing the sample SAR, We began structuring the solution as three prompts: 'Introduction', 'Body', and 'Conclusion', where multiple questions are asked in each section of the prompts. This can be seen in the file 'MISC methods tried -SAR Template ABC.docx' attached with this project.

We would include direct references to the prompt through placeholders referencing fields in the database. To test our hypothesis, we used SQL query engines to run prompts using columns extracted from select statements on database tables. We then used the .query() function of the various SQL query engines to generate the output.

In this approach, we attempted using 'SQLAutoVectorQueryEngine,' 'SQLJoinQueryEngine,' 'RouterQueryEngine,' 'PG Vector QueryEngine,' and 'NL SQL TableQueryEngine.'

1) Other Methods 1:

To deploy this idea, we tried a few methods. The first of this was to use SQL Query engines. Using SQL Query Engines our aim was to build a prompt to directly query the contents of the table, assuming that the model to understood the structure of the table and its contents. However, this turned out to be challenging as we struggled to obtain a single line of text output from the model, in the format we wanted.

Additionally, each of these queries required approximately 10-15 minutes to execute and failed to yield the expected results. You can find the code for this section in the 'Other methods 1' section of the ipynb titled 'Other methods tested'.

Moreover, based on the feedback we received – 'Was the customer using the same product they are expected to in the transactions which alerted the SAR?' is not a question we should ask, if the customer was already using the product they were expected to. It would only appear in the Alert narrative since they are not using the same product.

(https://docs.llamaindex.ai/en/stable/examples/query_engine/pgvector_sql_query_engine/)

(https://docs.llamaindex.ai/en/stable/examples/query_engine/RouterQueryEngine/)

(https://docs.llamaindex.ai/en/stable/examples/query_engine/SQLAutoVectorQueryEngine/)

(https://docs.llamaindex.ai/en/stable/examples/query_engine/SQLJoinQueryEngine/)

(https://docs.llamaindex.ai/en/stable/examples/query_engine/SQLRouterQueryEngine/)

2) Other Methods 2:

Following this we modified our approach. This method can be found in section 'Other methods 2' of the ipynb 'Other Methods Tested'. We use a natural language query engine to translate English questions into SQL queries across multiple related tables containing customer and transaction data. At its core, the code executes an optimized SQL query that joins multiple tables to gather detailed information about customers, accounts, expected products and geographies, lines of business, alerts, rules, and transaction details. After retrieving this data for a specific customer and alert ID, it formats the results into a readable string that can be fed into the language model. The code provides two approaches for prompting the AI: one using a system/user style format and another using a context-based template. Finally, it passes the formatted data to the query engine to generate a narrative about the customer and any rule violations. It creates an automated system to pull relevant data, format it appropriately, and generate natural language descriptions or reports about suspicious activities using AI.

However, this approach also proved ineffective for our purpose as we observed several limitations. Firstly, the response using the SQL query engines was not at all similar to the Sample SAR output shared. Secondly, the results indicated significant Hallucinations. And lastly, these queries took a long run time.

3) Other Methods 3:

Seeing the limitations in 'Other Methods 2', we tried to modify this by creating a template structure, filling in values and text from the database using placeholders and passing this prompt through an SQL Query Engine (NLSQLQueryEngine) as the RAG component. We observed that this method successfully generated a template structure as we wanted. However, the introduction section of the query took 13 to 15 minutes to execute on Google Colab Pro using a T4 GPU. We realized this would be time-consuming if we were to generate a template structure for the rest of the report.

Moreover, we found that the body of the SAR report is dynamic, making it unsuitable for a template structure. We realized we needed a more straightforward approach to generate the parts of the SAR that followed a static template.

Overall, using the various SQL query engines, we understood that some query engines are incompatible with certain LLMs. Lack of structured output and consistency in outputs indicated that these systems may need more time to be ready for production. Thirdly, a long run time of 10-15 mins indicated that this approach would be time-consuming and ineffective. Furthermore, we also observed that specific query engines were incompatible with Mistral Nemo LLM.

4) Other Methods 4—Inspired by the successful template structure, we created a solution combining two methods. For the static sections of the report, we decided to utilize placeholders within the template structure to populate data points from the database. Meanwhile, we decided to directly query the LLM for the dynamic parts of the report. This approach is illustrated in the section 'Other Methods 4' in the ipynb 'Other methods tested'.

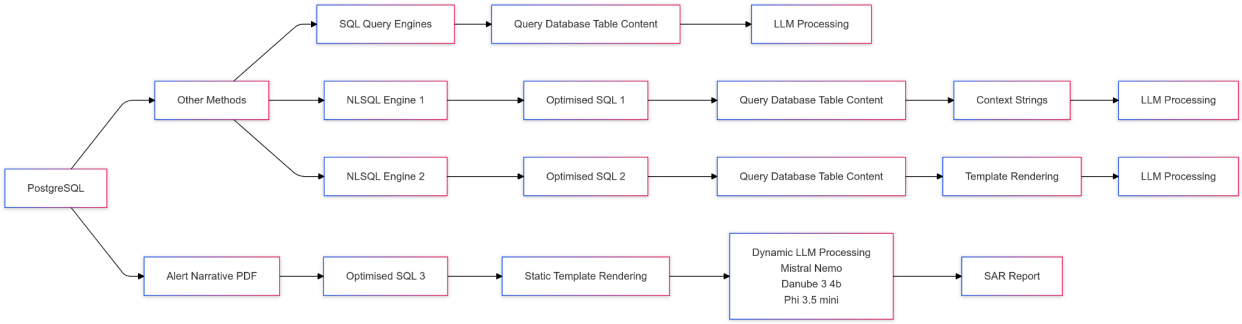
This proved successful, as the resulting text generated was close to the sample SAR shared by the client. We ran this method on Mistral Nemo, Danube 3 4b, and Phi 3.5 mini. We observed interesting differences in the prompt templates that each model understood.

We evaluated the Danube 3 4b and Phi 3.5 mini, as we initially recommended these models to the client. The client was keen to explore the performance of different models. While this wasn't a core requirement of our project, it allowed us to deepen our understanding of how prompting varies in LLMs and share these insights with the client.

After various tests, we arrived at six different prompts on Mistral Nemo, one prompt on Danube, and one prompt on Phi 3.5. We provided the client with the first 16 SARs generated: 12 using Mistral Nemo(6 of SAR-1, 6 of SAR-2), two using Danube (1 of SAR-1, 1 of SAR-2), and two using Phi 3.5 (1 of SAR-1, 1 of SAR-2). The client's feedback for these SARs have been attached with this project. *(Please see file 'Generated Results SAR_1- SAR_16 Jeff 20241117 - with evaluation rating -AAR.docx')*.

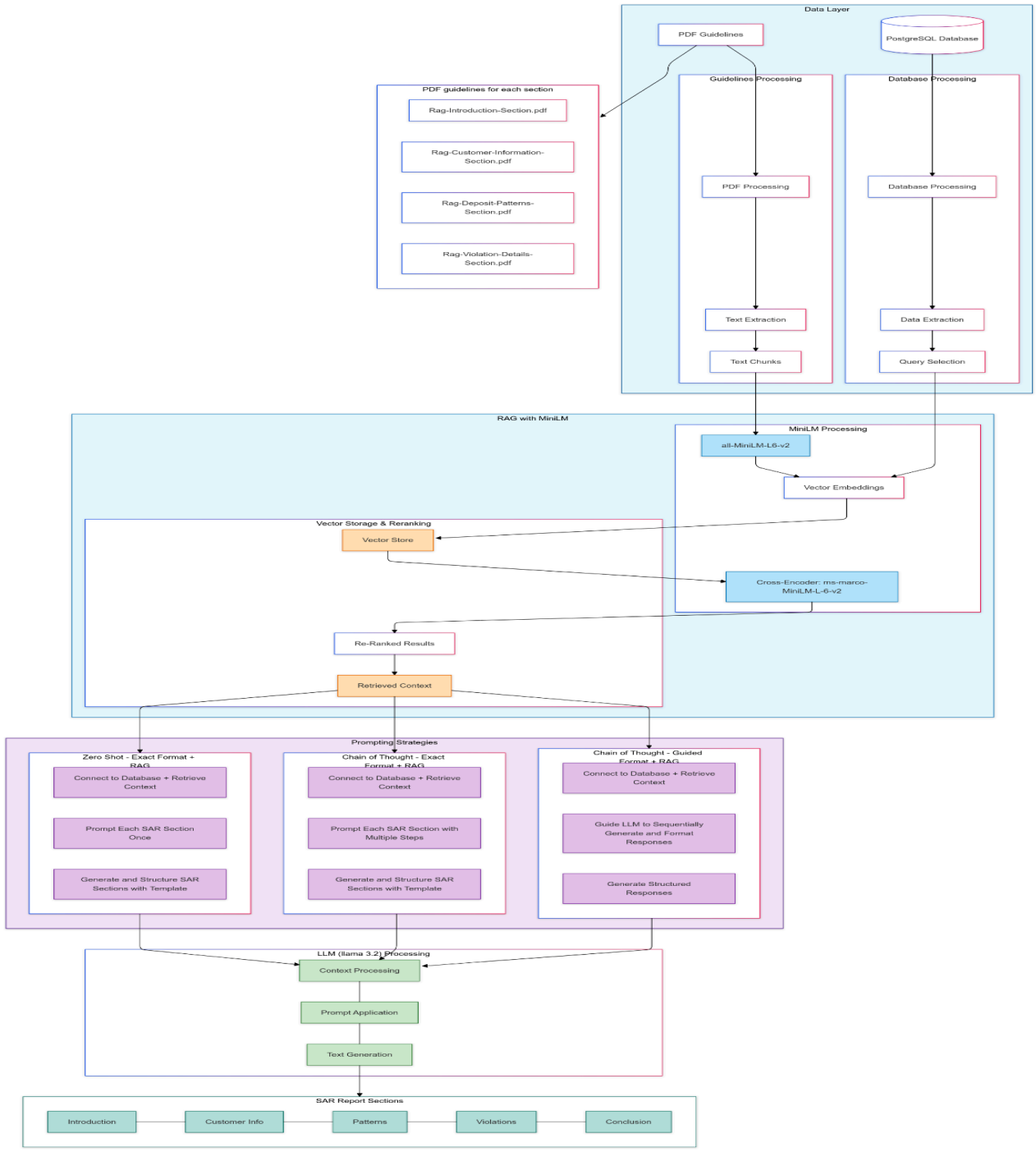
Based on the feedback, we understood that the SAR generated had limitations. For example, the format of the dates in the section generated by the LLM was not consistent with the dates present in the database. In addition, the customer's name did not appear in the output as it should have. Subsequently, some words or structures of the text provided inaccurate descriptions of the context and unintended meaning.

The following figure visualises the 4 Other methods we tried:

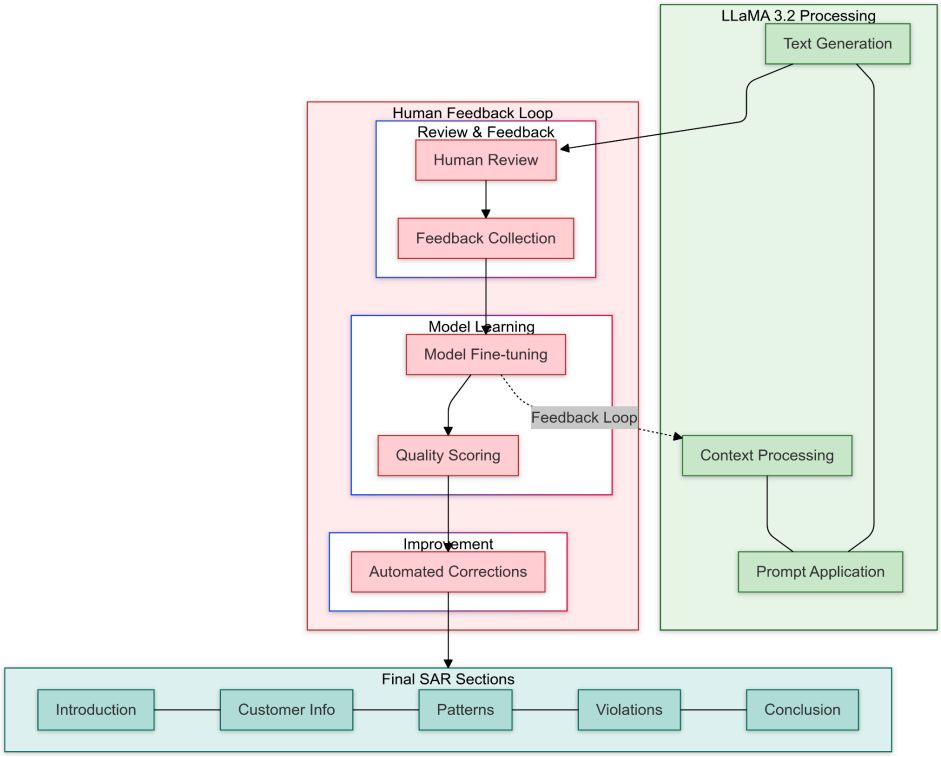


Hence, as the next step, we pivoted on this approach to reach Approach 1, which we present in this project.

Appendices 6



Appendices 7



Appendices 8.

ALERT NARRATIVE

Alert #: A-1 **Create Date:** 9/30/2024

Focal Entity: John Diamond
CIN: C-1
Review Scope: 9/2/2024 – 9/14/2024

Determination / Rationale:
Based on a review of internal and external sources, the reviewed transactions appear to potential suspicious.

Cash Structuring $10k
Rapid Movements of Funds
Large Wire to High Risk Jurisdiction

Transaction details:

The customer made 12 cash deposits for $9,000.00 each, totaling $108,000.00 over the course of 12 consecutive days between 9/2/2024 and 9/13/2024. According to KYC information, the customer is employed in the manufacturing industry, which is not a cash-intensive business and investigation of internal and external sources did not identify a legitimate source of funds for these cash deposits. On 9/14/2024, the customer then sent a wire transfer for $105,000.000 to ACME Investment Management in the Cayman Islands. The customer's KYC information does not indicate any apparent connection between either ACME Investment Management or the Cayman Islands.

SAR Recommendation:

A SAR filing is recommended for the following reasons:

- The customer apparently made 12 structured cash deposits for $9,000 each over 12 consecutive days without a legitimate source of funds.
- Shortly after make the cash deposits, the customer initiated a wire transfer to an unrelated company with which the customer has no apparent connection.
- There is no apparent lawful economic purpose for the customer's activity.
- The involvement of the high risk jurisdiction of the Cayman Islands.

ALERT NARRATIVE
Alert #: A-1 **Create Date:** 9/30/2024

Focal Entity: John Diamond
CIN: C-1
Review Scope: 9/2/2024 – 9/14/2024

Determination / Rationale:
Based on a review of internal and external sources, the reviewed transactions appear to potential suspicious.

Cash Structuring $10k
Rapid Movements of Funds
Large Wire to High Risk Jurisdiction

The customer made 12 cash deposits for $9,000.00 each, totaling $108,000.00 over the course of 12 consecutive days between 9/2/2024 and 9/13/2024. According to KYC information, the customer is employed in the manufacturing industry, which is not a cash-intensive business and investigation of internal and external sources did not identify a legitimate source of funds for these cash deposits. On 9/14/2024, the customer then sent a wire transfer for $105,000.000 to ACME Investment Management in the Cayman Islands. The customer's KYC information does not indicate any apparent connection between either ACME Investment Management or the Cayman Islands.

A SAR filing is recommended for the following reasons:

- The customer apparently made 12 structured cash deposits for $9,000 each over 12 consecutive days without a legitimate source of funds.
- Shortly after make the cash deposits, the customer initiated a wire transfer to an unrelated company with which the customer has no apparent connection.
- There is no apparent lawful economic purpose for the customer's activity.
- The involvement of the high risk jurisdiction of the Cayman Islands.

Based on 3 examples on SAR for depository institutions, from SAR example document.

Intro

"Investigation case number" {CaseNumber}.

1.1. Describe athe customer and athe suspected incident. Take into account the following information:

[PULLED from structured DB]

Summarise the customer demographic information from the customer, country, account, customerexpectedproducts, customerexpectedgeographies, customerlineofbusiness,

- The {customername} suspected of {rolename} where

Describe which rules the customer has violated based based on alert table and role table.

- Include the {rolename} and {roledescription} and {analystcomments} where the rule violations have occurred."

Was the customer using the same products as they are expected to in the transactions which alerted the SAR?

- Is the customer {expectedproduct} violating {Rolename}? I

[if SAR exists for customer ID]

1.2. Was there any previous SAR's filed for the customer? If so, what dates were the previous SAR's filed? What were the reasons for these SAR's?

What 'Red Flags' were noticed that initiated the SAR?

Taking into account the following information:

[Pull analysts comments]

Body

Incident history

"When was the first transaction that took place relevant to a specific alert from the alert table?"

- "Between {Transaction_date} and {Transaction_date} customer

"Who was the beneficiary? And where was the beneficiary located?"

- Summarize {beneficiary},{ beneficiary_location}, where the

"What were the transactions between the sender and beneficiary between those dates?"

- "Summarize the {transaction_date}, {transaction_type} and {transaction_amount} between the sender and the beneficiary between {alert_date} and {alert_date}."

"How does the transaction reflect Cash Structuring?"

1.

Incident details and pattern - Identifying case of Cash Structuring of $10k

"What were the deposit pattern for the customer up to 2 week prior to the alert dates and 2 week after the alert date?
"

Did the customer send between XX Amount and XX Amount before XX date (weeks

- Identifying patterns in these deposits.

What are the details of the violation from the {Analyst Comments}:

“

Internal LLM NY KYC information identified Diamond with the following details: DOB: 4/20/1988; SSN: 123-45-6789; address: 277 Park Ave., New York, NY, 12345; and occupation: manufacturing. There is no apparent connection between Diamond and ACME or the Cayman Islands.

External research was unable to conclusively identify a line of business for ACME.

“

Conclusion

What are the supplementary information to about this customer from external search

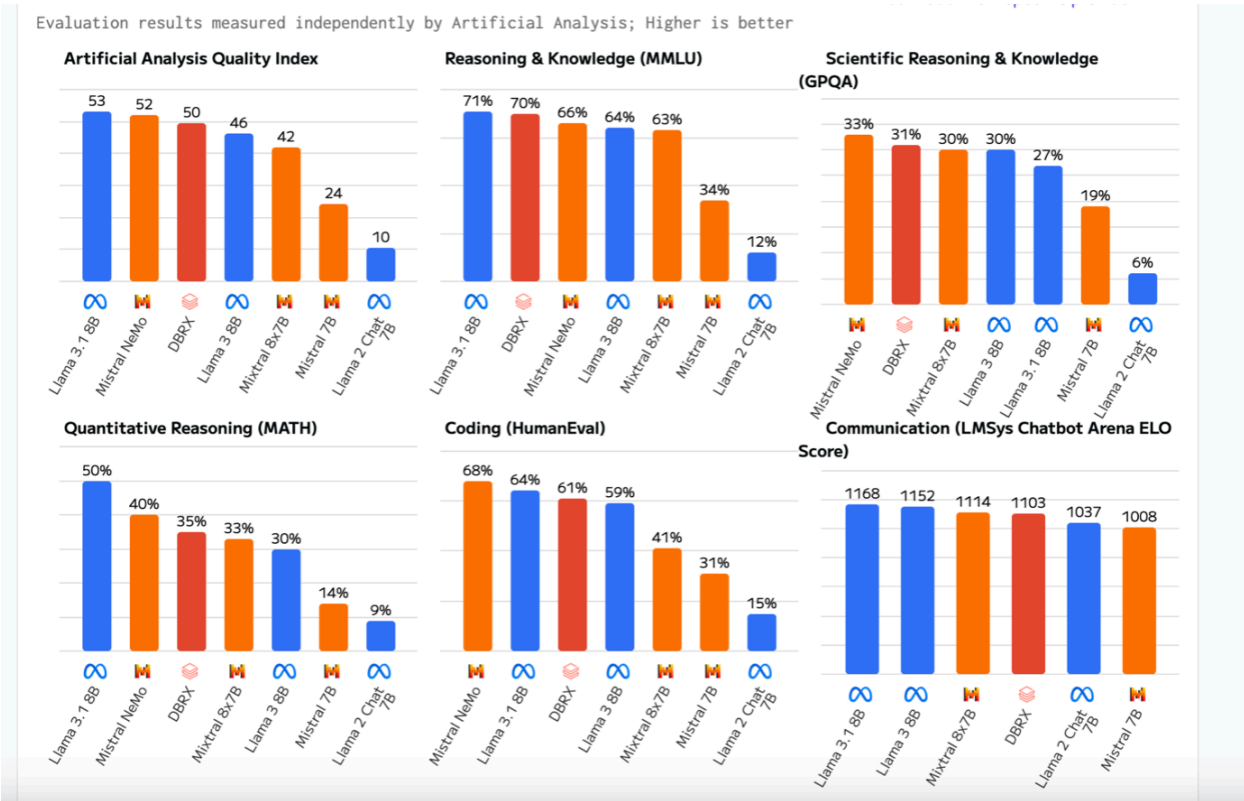
- “Search {Analyst Notes} for external search and supplemental information done about {Customer_name} and {Customer_line_of_business}, {customer_expected_geographies} and {customer_incroporation/residence_country}”

What follow-up actions will the bank take?

- “Search {Analyst Notes} for action bank will take on {Customer_name}”

Appendices 9.

Benchmark	Metric	Danube2 1.8B	Danube3 4B
ARC-c	25-shot	43.69	58.96
Hellaswag	10-shot	73.91	80.36
MMLU	5-shot	37.83	54.74
TruthfulQA	0-shot mc2	40.53	47.79
Winogrande	5-shot	69.30	**76.48**
GSM8K	5-shot	32.30	50.18
ARC-e	25-shot	74.92	83.84
BBH	3-shot CoT	30.39	38.92
CommonsenseQA	3-shot	54.30	**79.52**
CoQA	0-shot F1	68.30	77.23
PIQA	3-shot	78.67	**82.64**
SciQ	3-shot	95.70	97.10
Average		58.32	68.98



	Training tokens	Wino-Grande 5-shot	ARC Challenge 25-shot	MMLU 5-shot	Hella Swag 10-shot	GSM8K 5-shot	TruthfulQA 0-shot	XLSum en (20%) 3-shot	MBPP 0-shot	Human Eval 0-shot
Llama 3.1 8B	15T	77.27	57.94	65.28	81.80	48.60	45.06	30.05	42.27	24.76
Gemma 7B	6T	78	61	64	82	50	45	17	39	32
Mistral-NeMo-Minitron 8B	380B	**80.35**	**64.42**	**69.51**	**83.03**	**58.45**	**47.56**	**31.94**	**43.77**	**36.22**
Mistral NeMo 12B	N/A	82.24	65.10	68.99	85.16	56.41	49.79	33.43	42.63	23.78

Table 1. Accuracy of the Mistral-NeMo-Minitron 8B base model compared to the teacher Mistral-NeMo 12B, Gemma 7B, and Llama-3.1 8B base models. Bold numbers represent the best among the 8B model class