**A logo of a university

Description automatically generated**

Final Capstone Report

Ibukun-Oluwa Emmanuella Addy (61162024)

CS492: Undergraduate Thesis II

2023/2024 Semester III

5th July 2024

**A Comparative Study of RAG-LLM’s and a Traditional Expert System in the Nutrition Therapy of Ghanaians with Sickle Cell Disease**

**ABSTRACT**

Modern Digital Diet Managers offer various artificial intelligence-driven services to help people with special dietary needs or diseases make smarter decisions regarding their food intake and overall nutrition. Computer Vision and Recommendation systems are among the popular artificial intelligence methodologies used in managing the diet of Diabetics, Hypertensives, and Cancer patients; however, this technology is yet to be adopted in the nutrition therapy of Sickle Cell Disease(SCD), especially in Africa, where 80% of SCD cases are located, and 1 in 4 people suffer from the disease. [1]

This study conducted an experiment aimed at comparing and contrasting sickle cell dietary advice reports generated by three distinct systems: an assistant constructed using JSON databases with rule-based conditional programming, a large language model enhanced with a knowledge graph, and a large language model augmented with a vector database.

Python's spaCy library was utilized to determine the similarity between the reports, the system speed was measured using the time difference between sending a request and receiving a response, and the accuracy of the report was validated using the results of the rule-based conditional programming system as a benchmark.

**INTRODUCTION**

Sickle Cell Disease is a genetic red-blood cell abnormality that affects 20-30% of the African population. 80% of all sickle cell disease cases are found in Africa, and since the late 1980s, poor nutrition has been recognized as a challenge for people with sickle cell disease. However, this issue has yet to receive sufficient empirical attention. [2] In 2019, Hibbert and Umeakunne's paper discussed the significance of various other nutrients in managing sickle cell anemia. The researchers delved into the prevalence of herbal medicine for managing symptoms of SCD, placing emphasis on the proven benefits associated with certain leaves, such as Cajanus cajan and Carica papaya, in resisting haemolysis and diminishing sickled red blood cells. Furthermore, they presented an augmented nutritional requirements table for individuals with sickle cell anaemia, encompassing protein, carbohydrates, omega-3 fatty acids, vitamins B6, vitamin B12, vitamin A, vitamin C, vitamin D, vitamin E, folate, magnesium, zinc, and selenium. [3]

A similar point was made by Hyacinth et al, as their research underscored the imperative for heightened intake of vitamin D and zinc among those with sickle cell anaemia, in conjunction with a unique equilibrium of zinc and copper. [3] Furthermore, the inclusion of magnesium was proposed for alleviating pain during sickle cell anaemia episodes, drawing upon its application in paediatric cases.[4]

Researchers affiliated with the Philadelphia Biomedical Research Institute and Ibadan University conducted an experiment involving the collection of blood samples from sickle cell anaemia patients in the United States and Nigeria.

The study identified a subset of red blood cells termed "dense cells" with abnormal characteristics contributing to painful vasoocclusion. Laboratory experiments revealed that specific nutritional antioxidant supplements, hydroxyl radical scavengers, and iron-binding agents could inhibit the formation of dense cells. The recommended daily nutritional intake encompassed 6 grams of aged garlic extract, 4–6 grams of vitamin C, and 800 to 1200 IU of vitamin E. [4]

Nutrition therapy, also known as medical nutrition therapy (MNT), is an approach to treating medical conditions and their associated symptoms through a tailored diet. It involves the assessment of an individual's nutritional status, the development of a personalized nutrition plan, and the ongoing monitoring and adjustment of the plan as needed [5].

Thanks to rapid progress in Artificial Intelligence, computer software has reached an unprecedented level of capability in assisting individuals with their health and fitness decisions, often surpassing humans in decision-making.

According to a study published in the Journal of Intelligent Systems with Applications, personalized nutrition advice generated by A.I can lead to a more significant improvement in diet quality compared to generic recommendations. [6]

In 2019, Hibbert and Umeakunne published a recommended daily allowance table of nutrients for people with SCD. [3] However, there is still a gap, as there is no A.I system to help track those specific nutrient intakes or to examine meals for possible interaction with sickle cell disease medication.

There are two reasons why existing A.I diet managers do not specifically cater to individuals with sickle cell disease. The first reason is that most A.I-driven technologies are trained on data from first-world countries; hence, they fail to recognize and assess most African meals. The second reason is that sickle cell disease is endemic to Africa where the basic technological space is still growing. [7]

In a pedagogical paper on computer science for food and nutrition, Robbins and Saxton experimented with pigs and fish to determine their nutritional needs. They used a method called broken-line regression to estimate the nutritional needs of the animals. The researchers compared different broken-line regression models and found that a quadratic broken-line model worked best for their data because it considers the non-linear way animals respond to nutrients.

The authors also used a statistical analysis procedure called Non-Linear Mixed models, which helps include additional factors like gender and initial weight. The study highlighted the importance of using sophisticated models to understand how animals respond to nutrients and paved the way for further involvement of artificial intelligence in human nutrition. [8]

**Theoretical Frameworks**

* **Chronic Care Model**

The Chronic Care Model (CCM), developed by Dr. Edward Wagner, is a comprehensive framework to improve the care of patients with chronic diseases. Diverging from traditional acute care models, the CCM advocates for a proactive, patient-centered approach. Its fundamental components include the restructuring of health systems, implementation of clinical information systems and facilitation of patient self-management. Widely employed in managing conditions such as diabetes and hypertension, the CCM emphasizes continuity of care and proactive management to enhance health outcomes. [17]

* **The Technology Task Fit Model**

The Task-Technology Fit (TTF) model is a conceptual framework designed to evaluate how well a particular technology aligns with the requirements and characteristics of a given task or set of tasks within an organization. In this research, the artificial intelligence domain of focus is Natural Language Processing, specifically Large Language Models. Hence, the research seeks to determine what technological system is best fit for the chronic care model task of nutrition therapy of Ghanaians with sickle cell disease, by conducting an experiment to answer the following questions:

**Research Questions**

* What is the speed of Knowledge Graph Augmented Large Language models, Large Language Models with vector embeddings, and traditional rule-based expert systems in the Nutrition Therapy of Ghanaians with Sickle Cell Disease?
* What is the accuracy of Knowledge Graph Augmented Large Language models, Large Language Models with vector embeddings, and traditional rule-based expert systems in the Nutrition Therapy of Ghanaians with Sickle Cell Disease?
* What is the response lucidity of Knowledge Graph Augmented Large Language models, Large Language Models with vector embeddings, and traditional rule-based expert systems in the Nutrition Therapy of Ghanaians with Sickle Cell Disease?
* What is the response robustness of Knowledge Graph Augmented Large Language models, Large Language Models with vector embeddings, and traditional rule-based expert systems in the Nutrition Therapy of Ghanaians with Sickle Cell Disease?
* Can a Rule-Based Expert system generate similar feedback reports to a Knowledge Graph Augmented Large Language Model or, Large Language Models with vector embeddings in the Nutrition Therapy of Ghanaians with sickle cell disease?

**BACKGROUND**

Natural language processing is the application of computational techniques to the analysis and synthesis of natural language and speech.  It uses part-of-speech tagging, named entity recognition, and sentiment analysis methods. NLP is like a translator, analyzing and manipulating human language based on defined rules and structures. []

LLMs are a completely different technology. Instead of interpreting what's being asked, LLMs leverage deep learning to train on extensive text sets and build their own internal understanding of the language itself, and ultimately perform a variety of tasks. Large language models use transformer models and are trained using massive datasets — this enables them to recognize, translate, predict, or generate text or other content. Transformers are a neural network architecture that transforms or changes an input sequence into an output sequence.

**What is Retrieval Augmented Generation?**

Large Language Models (LLMs) showcase impressive capabilities but encounter challenges like hallucination, outdated knowledge, and non-transparent, untraceable reasoning processes. Retrieval-Augmented Generation (RAG) has emerged as a promising solution by incorporating knowledge from external databases. This enhances the accuracy and credibility of the generation, particularly for knowledge-intensive tasks, and allows for continuous knowledge updates and integration of domain-specific information. RAG synergistically merges LLMs’ intrinsic knowledge with the vast, dynamic repositories of external databases. [Xi Jin Ling]

The RAG research paradigm is continuously evolving, and we categorize it into three stages: Naive RAG, Advanced RAG, and Modular RAG, as showed in Figure 3. Despite RAG method are cost-effective and surpass the performance of the native LLM, they also exhibit several limitations. The development of Advanced RAG and Modular RAG is a response to these specific shortcomings in Naive RAG. A. Naive RAG The Naive RAG research paradigm represents the earliest methodology, which gained prominence shortly after the 3 Fig. 2. A representative instance of the RAG process applied to question answering. It mainly consists of 3 steps. 1) Indexing. Documents are split into chunks, encoded into vectors, and stored in a vector database. 2) Retrieval. Retrieve the Top k chunks most relevant to the question based on semantic similarity. 3) Generation. Input the original question and the retrieved chunks together into LLM to generate the final answer. widespread adoption of ChatGPT. The Naive RAG follows a traditional process that includes indexing, retrieval, and generation, which is also characterized as a “Retrieve-Read” framework [7]. Indexing starts with the cleaning and extraction of raw data in diverse formats like PDF, HTML, Word, and Markdown, which is then converted into a uniform plain text format. To accommodate the context limitations of language models, text is segmented into smaller, digestible chunks. Chunks are then encoded into vector representations using an embedding model and stored in a vector database.

**What is a vector database?**

Vector embeddings are a way to convert words and sentences and other data into numbers that capture their meaning and relationships. They represent different data types as points in a multidimensional space, where similar data points are clustered closer together. A vector database indexes and stores vector embeddings for fast retrieval and similarity search, with capabilities like CRUD operations, metadata filtering, horizontal scaling, and serverless. **It is a store of semi-structured data**

**What is a knowledge graph?**

Structured data, such as knowledge graphs (KGs) [Xi Jin Ling] , which are typically verified and can provide more precise in formation.

KG in constructing the hierarchical structure of documents contributes to maintaining consistency. It delineates the connections between different concepts and entities, markedly reducing the potential for illusions.

A knowledge graph (KG) is a structured representation of knowledge, typically in entities, their attributes, and the relationships between them. It can model real-world information and facilitate understanding and reasoning about data. [18]

LLM-augmented KG Question Answering (KGQA) aims to find answers to natural language questions based on structured facts stored in knowledge graphs. LLMs can serve as:

* Entity/relation Extractors: Identifying entities and relationships mentioned in natural language questions and retrieving related facts in KGs. LLMs can effectively perform this task due to their proficiency in language comprehension.
* Answer Reasoners: Reasoning over retrieved facts to generate answers directly. LLMs can be used for answer reasoning to generate answers based on the retrieved facts.

**What is Prompt-Engineering?**

Prompt engineering is the practice of designing inputs for AI tools that will produce optimal outputs. Even after performing Retrieval Augmented Generation on an LLM, it is still necessary to structure the prompts accordingly.

**What is a Traditional Expert system?**

**Chapter 2**

RELATED WORKS

1. Recommendation Sytems

In 2023, Ahmad and Khan developed a food recommendation system that addressed diverse health conditions concurrently, encompassing iron deficiency, kidney diseases, diabetes, and hypertension. The authors used a nutrition-based food dataset, which contained information about the nutritional content of various foods, such as calories, macronutrients, micronutrients, etc. They also collected input data from the user to generate a health profile containing details about an individual's health, including age, gender, and existing health conditions. [9] A specialized algorithm was then employed to calculate the specific nutritional needs of each individual based on their health profile.

In 2018, Alian and Pandy addressed the rising diabetes epidemic in American Indian communities by proposing a mobile application for proactive diabetes self-care. The application utilizes users' ontological profiles, incorporating socio-economic, cultural, and geographical factors, to deliver personalized eating habit recommendations. The diabetes management system relies on logical programming and a knowledge base built on general diabetes information, food and nutrition facts, and American Indian healthcare guidelines from the American Diabetes Association. This knowledge is translated into rules using a "premise→conclusion" logic form. [10] Expressed in the Semantic Web Rule Language (SWRL), these rules cover diverse aspects of diabetes management and are processed by a reasoning engine using forward chaining. [10]

Efficient nutrition therapy poses a technical challenge, necessitating a deep understanding of machine learning models. Group recommender models, employing Naïve Bayes, SVM, and RM algorithms, face criticism for their narrow focus on single-disease dietary recommendations. Some advocate for collaborative-based algorithms, asserting greater efficacy in providing optimal food suggestions. [11]

1. Expert Systems

The Personalized Diet Recommendation System by Hussain et al, aims to assist cancer patients in planning their daily diets. It employs Case-based Reasoning, Rule-based Reasoning, and genetic algorithms to create customized diet menus based on individual health information. Users input data such as cancer type, treatment stage, activity level, food preferences, allergies, ethnicity, and side effects to build their profiles. [12] The system consists of four modules: user management (for login and authentication), diet planning (utilizing Case-based and Rule-based Reasoning), menu construction (using a food database and Genetic Algorithm), and menu adaptation (suggesting substitutes based on Rule-based Reasoning). The outcome is a system that recommends a personalized daily dietary allowance for energy, carbohydrates, protein, calcium, thiamin, niacin, riboflavin, vitamin A, vitamin C, vitamin D, and vitamin E. Based on these recommendations, the system suggests breakfast, lunch, and dinner menus tailored to the user's specific needs. [12]

1. Computer Vision Systems

In a Swiss study, volunteers were enlisted to record brief videos of their daily food and beverage intake using the goFOODTM Lite application. The application works with single images of real food or barcodes of processed foods. The accuracy of the goFOODTM system's estimation of the participants' calorie and macronutrient consumption was then evaluated. [13].

goFoodsTM utilized a dataset comprising 57,000 images from MyFoodRepo dataset version 2.1. [13] The segmentation task used a Convolutional Neural Network (CNN) based on Mask RCNN pre-trained on the COCO dataset, with ResNet-50 as the backbone.

For the image classification task, approximately 200,000 images were obtained and categorized. Each segmented item was processed by a food recognition network using RegNetY-16GF. Mix-up interpolation was used in the recognition training process. [13]

The food volume estimation module employed depth maps to convert 2-D representations of food items into a 3-D space. Two approaches were used: the Neural-Based Approach and the Geometry-Based Approach. [13] In the Neural-Based Approach, single images captured at a 90° angle were used for depth estimation, leveraging the Zoe model, which incorporates multiple depth modules in an encoder-decoder architecture. [13] The Geometry-Based Approach eliminates the need for a plate in the food image, by detecting key points from reference cards and segmentation masks, rectifying stereo image pairs for depth information, and converting the resulting disparity map into a depth map for volume estimation. [13]

The research found that by using just one image of food taken by the user, there was an average error of 27.41% in estimating calories per person. Additionally, it had errors of 31.27% for carbohydrates (CHO), 39.17% for protein, and 43.24% for fat.

Similar to goFoodsTM, another group of researchers developed an innovative neural network architecture named Delicacy Net, comprising four main modules: an environment feature extraction module, an encoder, a decoder, and a semantic output module. [14]

The system analyzed and identified the food's primary nutrients when provided with images. The process involved extracting environmental features from the images, processing them through the encoder, and presenting the results as a text table using the decoder. Their model exhibited high accuracy in predicting food components, making it applicable in practical scenarios. [14]

In the Food4Me Proof-of-Principle study by Walsh et al., 1607 participants across Europe were offered personalized nutrition advice from human dieticians and the Food4Me automated system. [15] All participant’s nutrient goals were grouped into three broad categories: nutrient goals one to three. In evaluating nutrient-related goals, the results showed generally high agreement between human dieticians and the automated system (92% for goal 1, 87% for goal 2, and 87% for feedback advice). Still, some disagreements were noted, especially in goal three.

Another similar study tested a new U.S.A domiciled computer vision app called SNAQ. SNAQ takes pictures of a user’s food to determine what an adult human should eat and how much. The researchers wanted to see if SNAQ can accurately measure how much energy a woman has consumed, compared to a traditional human method called 24-hour dietary recall (24HR) and a reference method called doubly labeled water (DLW). They found that SNAQ did better than 24HR in estimating how much energy a woman gets from food. SNAQ and 24HR had similar results in assessing energy and the types of nutrients obtained from food. Through statistical comparison, the researchers concluded that SNAQ seems to be better than 24HR at estimating energy intake. Still, more research is needed to see how well it works with a large number of users. [16]

1. Large Language Models

A Copilot framework presents an innovative approach to utilizing LLMs in specialized tasks, eliminating the necessity for model fine-tuning. [19]

In 2024, Ren et al. compared various Large Language Models against one another to see which was most efficient in serving as a general health consultant. The Healthcare Copilot, designed for medical consultation, comprises Dialogue, Memory, and Processing components. Dialogue facilitates patient interactions, Memory stores conversation data, and Processing generates reports. [19]

The evaluation of the proposed Healthcare Copilot involved the use of four popular Language Model (LLM) backbones: GPT-4 1106-preview, GPT-3.5-turbo, LLaMA2-70b-chat, and ChatGLM3-6B. [19]

The experiment primarily utilizes ChatGPT as virtual patients in simulating medical consultations. Real cases from the MedDialog dataset are used as references to ensure authenticity and relevance in simulated medical scenarios. Four key metrics are used to evaluate the Healthcare Copilot comprehensively:

* Inquiry capability: assesses the ability to ask accurate and relevant questions.
* Conversational fluency: measures the user experience within the dialogue context.
* Response accuracy: showcases the preciseness of model responses.
* Response safety: describes the model's compliance with safety and ethical standards during its responses.

The integration of Healthcare Copilot yielded noticeable enhancements in response accuracy across all models. Additional information gathered by the Inquiry module and rectifying inaccuracies through the Safety module contributed significantly to this improvement. GPT-4, with its advanced capabilities, outperformed other general LLMs, and Healthcare Copilot consistently enhanced its performance, demonstrating remarkable generalization abilities. [19]

**Chapter 3**

**METHODOLOGY**

* **Data Collection**

Nutrient intake allowances for persons with SCD for protein, carbohydrates, omega-3 fatty acids, iron, vitamins B6, vitamin B12, vitamin A, vitamin C, vitamin D, vitamin E, folate, magnesium, zinc, and selenium. (Sub-divided by age and gender) was obtained from the only publicly available recommended daily intake table for sickle cell disease, published by Hibbert and Umeakunne in 2019.

Sickle Cell Disease medication names, compounds, and their possible interactions with certain foods were obtained from the online databases of the American Center for Disease Control, papers from the American National Library of Medicine, and information on approved treatment medications for sickle cell disease from the sickle cell anemia news platform. Chemical compounds present in various foods were obtained from FooDB. FooDB is an openly available internet-based repository of foods and the compounds and nutrients present in them. The Database contains information about approximately 900,000 foods stored in JSON format.

Over sixty scholarly and peer-reviewed publications relating to nutrition for sickle cell disease were downloaded from .......{insert source here}

Nutrients found in Ghanaian foods per serving of 100 grams were obtained from Kaggle, and My Fitness Pal Database, as well as numerous food blog websites, listed in the appendix of this report.

User information such as stomach content, weight, height, age, and sex were provided by user input once the program started.

* **Model Configuration**

1. **Traditional Expert System**

Using a Python programming language, we designed a rule-based system with a set of predefined rules and logical inferences that uses data collected in the data collection stage to advise a user on what can be safely consumed or what needs to be removed from their plate, to adhere to Recommended Dietary Intakes. The traditional expert system used a dictionary as a data structure to represent a hypothetical stomach of a user, which contained foods they had eaten throughout the day.

1. **Large Language Model with Vector Embedding System**

Chromadb software, PyPDF, OpenAI, Chat G.P.T 3.5.

Chunking Strategy: The most common method is to split the document into chunks on a fixed number of tokens (e.g., 100, 256, 512) [88]. L

1. **LLM-augmented KG Question Answering System**

We utilized a pre-trained LLM, specifically Open-AI's Chat-GPT 3.5, and augmented it with a knowledge graph representing domain-specific relationships, such as the relationship between foods, a user's stomach, a user's medication, and chemical compounds found in food.

We created the knowledge graph using Neo4J software. Nodes in the knowledge graph represented entities such as the user, foods, chemical compounds, and medications. The Edges in the graph represent the relationship between the entities, and labels on the edges describe the type of relationship, such as "REACTS\_WITH," "CONTAINS\_COMPOUND,” “CONTAINS,” “NEEDS,” and “HAS\_HEALTH\_EFFECT.”

Although the large language model can respond to questions not provided in neo-4j cipher query format, it best responds to natural language questions with a cypher generation template modeling how the large language model should generate a cypher query and find results in the database, based on this. This template must be provided every time a new language chain is invoked; therefore, for each user request, the large language model must be prompted on how to respond, using the predefined cypher generation template. This prompting via cypher query serves as part of the prompt engineering aspect of the system design.

**Examples: Question: Which foods are high in calcium and vitamin D?**

**Answer: MATCH (f:Food)-[:CONTAINS]->(n:Nutrient) WHERE toLower(n.name) CONTAINS 'calcium' OR toLower(n.name) CONTAINS 'vitamin d' RETURN f.name AS Food, n.name AS Nutrient**

* **Architectural Design: Layered and Pipe Filter Design.**

The pipe-filter architectural pattern operates by sequentially processing a data stream in a single direction. Data flows through the system as pipes transfer data to filters, with the output of one filter serving as the input for the subsequent one, thus forming a chain of data processing. [19]

Age, sex, meals eaten, and meals desired are collected in the user interface layer, which the user operates. The data from the user interface layer is then sent into a presentation layer that performs data processing, such as string formatting and structuring the data into appropriate input types for large language models or conditional programming systems. Internally, the systems make parallel queries to their databases, i.e., vector database, knowledge graph, and JSON database. After the output, which is the dietary report, is generated from these layers, it is passed to the data analysis layer to compare the outputs.

* **Experiment Setup**

On starting the A.I system, a user provides their age, weight, height, and their sex. This information is used in parallel by all systems to perform any necessary queries or searches. Using it’s dedicated database, each system fetches data about the food nutrient information, chemical composition, and possible drug interactions, the report from each system is intended to critically discuss how eating the meal could impact the user and what steps they could take regarding portioning. , a simple decision tree was used to determine whether or not the user should consume the meal. Each report should also provide possible solutions such as increasing or decreasing the portion of specific components of the meal and waiting for a certain amount of time before consuming the meal in the case of drug interactions. To assess the three systems, the same question consisting of the same meals and age and sex where passed to the systems.

An Example of such a Prompt is:

*“I am a 37 Year Old Female with Sickle Cell Disease. I have eaten 2 Plates of rice and 2 Chicken drumsticks today. I am about to eat 20g of Aprapransa and 10g of Meat for dinner, how might this affect my health?”*

* **Testing Semantic Similarity of Reports**

Python’s spaCy library was utilized to determine the similarity between the reports generated by the systems. SpaCy, with its comprehensive architecture, seamlessly blends linguistic rules, statistical modeling, and neural network techniques. It tokenizes input texts and applies a pipeline that includes part-of-speech tagging and named entity recognition. Additionally, SpaCy utilizes word embeddings to capture semantic information and relationships between words efficiently.

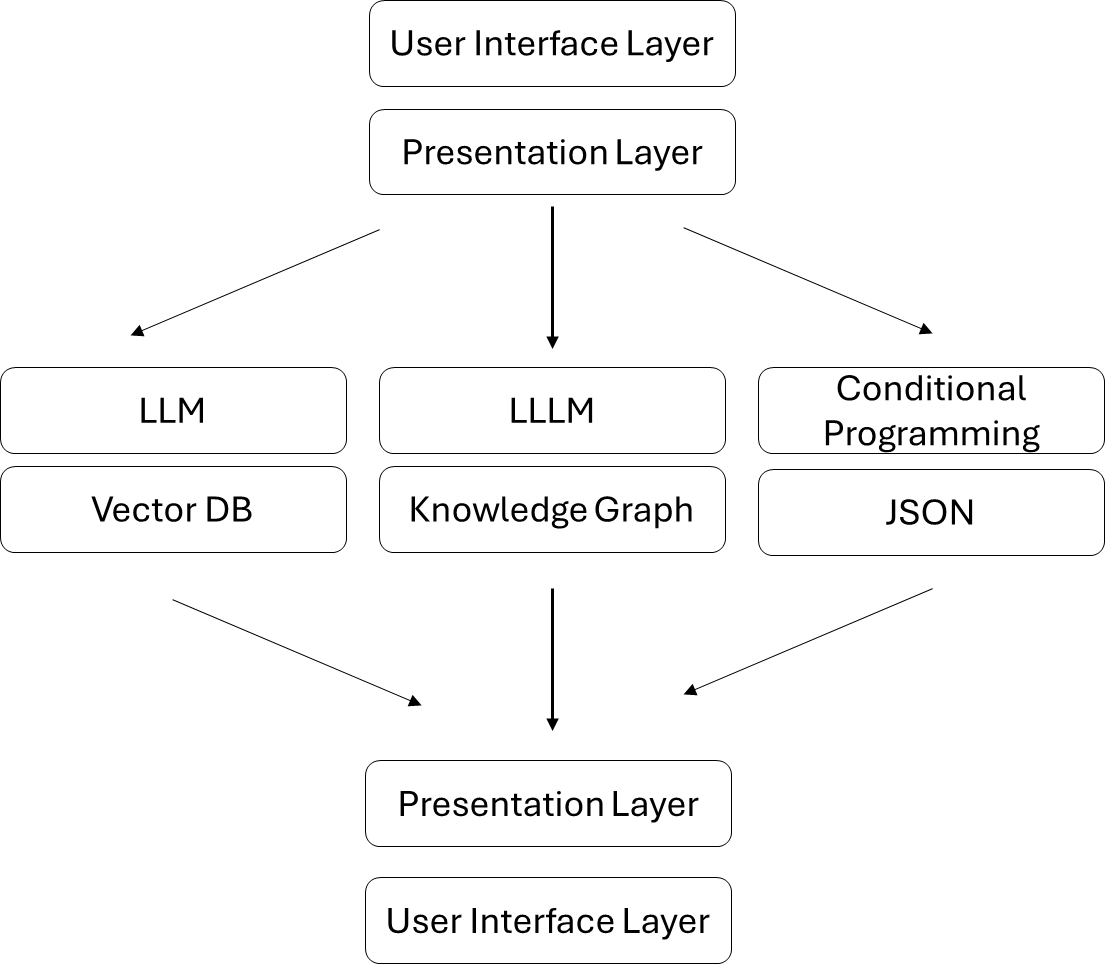
* **Testing Speed of Report Generation**
* **Testing Accuracy of Generated Reports**
* **Testing Lucidity of Generated Reports**
* **Testing Robustness of Reports**

**A group of circles on a black background

Description automatically generated**

**A diagram of several cylindrical objects

Description automatically generated**

****

**Chapter 4**

**Results**

**Semantic Similarity**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Previous Meals** | **Desired Meal** | **Age** | **Sex** | **KG-Augmented LLM** | **Traditional Expert System** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Mathematical Accuracy**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Previous Meals** | **Desired Meal** | **Age** | **Sex** | **KG-Augmented LLM** | **Traditional Expert System** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Lucidity**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Previous Meals** | **Desired Meal** | **Age** | **Sex** | **Semantic Similarity Score Between LLM and Traditional Expert System** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Factual Accuracy**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Previous Meals** | **Desired Meal** | **Age** | **Sex** | **Semantic Similarity Score Between LLM and Traditional Expert System** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Robustness**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Previous Meals** | **Desired Meal** | **Age** | **Sex** | **KG-Augmented LLM** | **Traditional Expert System** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

'I am a 1-18y-Year-Old Male. I have eaten 10g of Hummus. I am about to eat 20g of Firm Tofu, should I eat it?',

'result': "To determine if you should eat the 20g of Firm Tofu, we need to calculate the total nutrient intake from the Hummus and the potential intake from the Firm Tofu and compare it to your Recommended Daily Intake for Carbohydrates.\n\nGiven that you are a 1-18 year old Male, the quantity needed of Carbohydrates is 130 units.\n\nFrom the provided information, we can see that the Hummus contains 4.6 units of Carbohydrates per 100g. You have eaten 10g of Hummus, so you have consumed 0.46 units of Carbohydrates from the Hummus.\n\nNow, let's calculate the potential Carbohydrate intake from the Firm Tofu:\nFirm Tofu contains 0.3 units of Carbohydrates per 100g. If you eat 20g of Firm Tofu, you will consume 0.06 units of Carbohydrates from the Firm Tofu.\n\nTherefore, your total Carbohydrate intake would be 0.46 + 0.06 = 0.52 units.\n\nSince your Recommended Daily Intake for Carbohydrates is 130 units, it is safe for you to eat the 20g of Firm Tofu as it will not exceed your Carbohydrate intake limit for the day."}

**Conclusion**

**Limitations/Future Work**

Accounting for Network Delays or server capacity of chosen LLM

Incorporating users BMI and activity level

**Appendix**