Data Science and Engineering (DSE) Record 2025 Vot. 6 No. 1

# THE USE OF LARGE LANGUAGE MODELS IN GROUP CHAT PROGRAM FOR COUNSELING BETWEEN DOCTORS AND HEART DISEASE PATIENTS

Noratap Muangudom [1] and Karn Patanukhom [2]

[1] Master's Degree Program in Data Science, Chiang Mai University, Chiang Mai, Thailand
[2] Department of Computer Engineering, Faculty of Engineering, Chiang Mai University, Chiang Mai, Thailand
noratap_mu@cmu.ac.th

**Abstract.** In recent years, Large Language Models (LLMs) have demonstrated significant potential in various applications, including healthcare, education, and customer support. This study investigates the integration of LLMs into group chat environments to facilitate medical counseling between doctors and heart disease patients. Traditional chatbot systems primarily operate in one-on-one interactions, which can lead to redundant queries and inefficiencies in medical consultations. This research introduces a novel chatbot system designed for group chat settings, allowing multiple users and medical professionals to interact seamlessly within the same conversation.The chatbot system retrieves medical knowledge from a predefined document database using an information retrieval model to ensure responses are relevant and accurate. A verification mechanism is integrated, enabling doctors to review and validate chatbot-generated responses before they are presented to patients. The study employs hypothesis testing and real-world evaluations to measure chatbot performance across three key dimensions: response accuracy, response speed, and user satisfaction. Experimental results indicate that group chat environments improve communication efficiency, reduce repetitive queries, and enhance patient engagement compared to traditional one-on-one chatbot interactions.Furthermore, user feedback highlights the strengths and limitations of the proposed system. While the chatbot successfully provides relevant medical information, challenges remain in ensuring response accuracy, reducing response time, and improving contextual understanding in group conversations. Future work will focus on refining chatbot algorithms, enhancing natural language processing capabilities, and expanding the medical knowledge base to support a wider range of healthcare scenarios. This research underscores the potential of LLMs in transforming digital healthcare support, making medical consultations more efficient, accessible, and collaborative.

**Keywords:** Large Language Model, Natural Language Processing, Information Retrieval, Question Answer System, Chatbot, Generative AI

## 1 Introduction

In today's world, Large Language Models (LLMs) in generative AI have become popular across various fields such as business, education, and healthcare. LLMs are trained on vast amounts of data to achieve optimal performance in understanding natural language, enabling applications like question-answering systems, text content generation, and recommendation systems.

However, while LLMs represent state-of-the-art technology, certain tasks necessitate domain expertise, particularly in fields like medicine where issues of legality and user safety are paramount. For instance, recommending drugs in medical contexts requires specific knowledge and adherence to regulations.

To address this challenge, one solution is to constrain chatbots to provide responses based on predefined documents and then employ domain experts to verify these responses. This approach ensures that the chatbot retrieves information using an information retrieval system to locate relevant documents based on user queries.

Most LLM chatbots are designed for one-on-one interactions, where each chatroom hosts only a single user and bot, as seen with platforms like ChatGPT and Gemini. This setup necessitates creating separate chatrooms for each user, leading to time wastage in explaining problems repeatedly between different users and the chatbot.

To address this issue, we propose integrating chatbots into chat groups, allowing multiple users and chatbots to interact within a single chatroom. This setup would enable users to seamlessly switch between chatting with other users and engaging with different chatbots. This experimental approach aims to demonstrate that chat groups facilitate smoother dialogues compared to single-user chatrooms.

However, LLM chatbots like ChatGPT are specifically designed for one-on-one chatrooms. To use ChatGPT effectively, users must specify their roles as either User, Assistant, or System. Therefore, enabling the chatbot to interact with multiple users necessitates specifying each user's details in the prompt.

Moreover, in terms of information retrieval systems, it's crucial to consider the type of data being queried, especially in group chats where references to others are common. For instance:

User A asks User B: "I feel sick, I have a headache." (User's chat)
User B asks Chatbot: "What's wrong with User A?" (User asks Chatbot)
Chatbot responds to User B: "A has a fever." (Chatbot answers user)

In this scenario, the chatbot may misunderstand the query from User B and retrieve incorrect information because "What's wrong with User A?" is a vague query. The correct query should focus on User A's symptoms, such as "I feel sick, I have a headache." This distinction highlights the importance of precise queries in ensuring accurate responses from the chatbot.

## 2 Literature Review

### 2.1 Basic Data Science Literature

#### 2.1.1 Statistics for Data Science

Statistics for Data Science involves techniques for analyzing data descriptively, such as mean, standard deviation, minimum, maximum, and exploring correlations between variables. It also encompasses statistical tests like t-tests, z-tests, etc., which are crucial for comparing averages or proportions and determining the statistical significance of observed differences

To deepen the analysis, it is important to consider the underlying statistical theories and formulas that support accurate result interpretation. The following methods are commonly used

**1) Normality-Test**

Normality-Test: Assessing whether a dataset follows a normal distribution is a critical step before performing many parametric tests. The Kolmogorov-Smirnov test [1] is a popular method for testing normality. The hypotheses for this test are:

- Null hypothesis (H0): The data is normally distributed.
- Alternative hypothesis (H1): The data is not normally distributed.

$$D = sup_x|F_n(x) - F(x)| \tag{2.1}$$

$D$ is the test statistic that measures the largest absolute difference between $F_n(x)$ and $F(x)$.

$F_n(x)$ is the empirical distribution function of the sample.

$F(x)$ is the cumulative distribution function of the normal distribution

**2) T-Test [2]**

The t-test assesses whether the means of two groups are statistically different from each other. There are two main types of t-tests:

- **Paired t-test**: Paired t-test: Used when the observations in the two groups are dependent (e.g., measurements taken before and after a treatment).

$$t = \frac{\overline{d}}{s_d/\sqrt{n}} \tag{2.2}$$

$\overline{d}$ is the mean of the differences between paired observations.
$s_d$ is the standard deviation of the differences.
$n$ is the number of pairs.
$t$ is the t-value.

- **Independent t-test**: Used when the observations in the two groups are independent.

$$t = \frac{\overline{x_1} - \overline{x_2}}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \qquad (2.3)$$

$\overline{x_1}$ is the mean of first group.
$\overline{x_2}$ is the mean of second group.
$n_1$ is the size of first group.
$n_2$ is the size of second group.
$s_1$ is the standard deviation of first group.
$s_2$ is the standard deviation of second group.
$t$ is the t-value.

The p-value then follows from the table with the t-distribution, where the degrees of freedom are obtained via the following equation:

$$df = \frac{(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2})^2}{\frac{1}{n_1 - 1}(\frac{s_1^2}{n_1})^2 + \frac{1}{n_2 - 1}(\frac{s_2^2}{n_2})^2} \qquad (2.4)$$

### 2.1.2 Programming for Data Science

Programming for Data Science is essential for developing and enhancing chatbots. It extends to utilizing programming languages such as JavaScript, C#, and Dart to create applications that operate across various platforms and cater to specific user needs effectively.

## 2.2 Data Science-Topics Specific Literature

### 2.2.1 Cloud Computing

Cloud Computing plays a crucial role in managing and developing backend infrastructure, including virtual machines (VMs) and Docker for system management. It facilitates easy deployment and management of various APIs and services such as Firebase Firestore and OpenAI-text-generation, enhancing scalability and efficiency.

### 2.2.2 Information Retrieval

Information Retrieval techniques are employed in searching for and retrieving relevant information and documents based on user queries. This involves storing documents in various formats to optimize space and ensure rapid access. Evaluating the effectiveness of information retrieval systems helps gauge their performance and relevance.

To understand how to evaluate and optimize information retrieval systems, it is important to delve into some key theoretical models and evaluation metrics.

- **BM25 [3]:** is a ranking function used by search engines to estimate the relevance of documents to a given search query. It builds on the probabilistic information retrieval model and is designed to address some of the shortcomings of earlier models like TF-IDF.

$$BM25(d,Q) = \sum_{q \in Q} IDF(q) \cdot \frac{f(q,d) \cdot (k_1 + 1)}{f(q,d) + k_1 \cdot (1 - b + b \cdot {|d|}/{avg_d})} \tag{2.5}$$

$f(q,d)$ is the term frequency of term q in document d.
$|d|$ is the length of document d.
$avg_d$ is the average document length in the collection.
$IDF(q)$ is the inverse document frequency of term q.
$k_1$ is term frequency saturation parameter (typical values range between 1.2 and 2.0).
$b$ is length normalization parameter (default value is 0.75).

To measure the effectiveness of information retrieval models, several metrics are commonly used. These include Precision@K, Recall@K, Reciprocal Rank, and F1-score@K.

## 2.3 Domain Knowledge Literature

### 2.3.1 Comprehensive Heart Failure Management Program [4]

A Comprehensive Heart Failure Management Program serves as a vital information source for chatbots to answer queries related to caring for heart disease patients. This includes details on medications commonly prescribed for heart disease, recommended exercises, precautions for patients, and other relevant aspects of heart disease management.

### 2.3.2 CataractBot [5]

An AI-powered chatbot developed in collaboration with a tertiary eye hospital in India, represents a significant advancement in healthcare communication. Leveraging large language models (LLMs), CataractBot addresses the challenge of disseminating accurate health information amidst the overwhelming and often inaccurate digital content available to patients. By querying a curated knowledge base, it provides instant, expert-verified responses to cataract surgery-related queries. Its design includes multimodal support and multilingual capabilities, making it accessible to a diverse patient population. In an in-the-wild deployment study involving 49 participants, CataractBot demonstrated its value by offering anytime accessibility, saving time, and accommodating varying literacy levels. The establishment of trust through expert verification was a key factor in its success, suggesting that similar expert-mediated LLM bots could be effectively utilized in other areas of healthcare to enhance patient engagement and information reliability.

### 2.3.3 HUIXIANGDOU-CR [6]

Coreference Resolution in Group Chats delves into the complexities of resolving pronominal references within the context of group chat environments. By preprocessing 58,000 authentic chat lines and manually annotating 2,302 questions, the research emphasizes the challenges posed by the informal and dynamic nature of group conversations. Leveraging the Qwen series models, fine-tuned using Low-Rank Adaptation (LoRA) techniques, the study demonstrates significant improvements in F1 scores, highlighting the efficacy of large language models (LLMs) for this specific NLP task. The integration of scaling law principles further validates the reliability of the manual annotations and the robustness of the model's performance. This research contributes valuable insights into the application of LLMs for coreference resolution in group chats, showcasing the potential of advanced NLP techniques to enhance communication and understanding in digital interactions.

### 2.3.4 CLINFO-AI [7]

The rapid expansion of medical literature challenges healthcare professionals in staying updated. Traditional retrieval systems like PubMed provide access to vast research but lack efficiency in delivering quick, precise answers. Large Language Models (LLMs) have revolutionized natural language processing in healthcare, particularly in summarization and question-answering. However, many remain closed-source and lack systematic evaluations.

Retrieval-Augmented LLMs (RetA LLMs) integrate external knowledge sources to enhance accuracy and reduce hallucinations. Clinfo.ai, an open-source RetA LLM system, addresses these issues by dynamically retrieving and synthesizing scientific literature. It employs an LLM chain architecture consisting of query generation, retrieval, relevance classification, summarization, and synthesis.

Clinfo.ai is evaluated using the PubMedRS-200 dataset, containing 200 medical questions with systematic review-based answers. Automated metrics like UniEval and COMET assess coherence, consistency, and relevance, demonstrating that Clinfo.ai outperforms existing tools such as Elicit and Statpearls.

Despite advancements, challenges remain in refining retrieval processes, ensuring factual accuracy, and standardizing evaluations. Future research should enhance query mechanisms, integrate domain-specific knowledge, and incorporate human evaluation to improve AI-generated medical summaries.

### 2.3.5 Transformer [8]

The Transformer is a sequence-to-sequence language model, meaning that both its input and output are structured as sequences, such as text or video. In this context, the primary data type used is text. The architecture of the Transformer model is illustrated in Figure 2.1, consisting of two main components: the Encoder, which learns from the input data, and the Decoder, which is responsible for generating the output. Each of these components comprises several submodules, as detailed below.

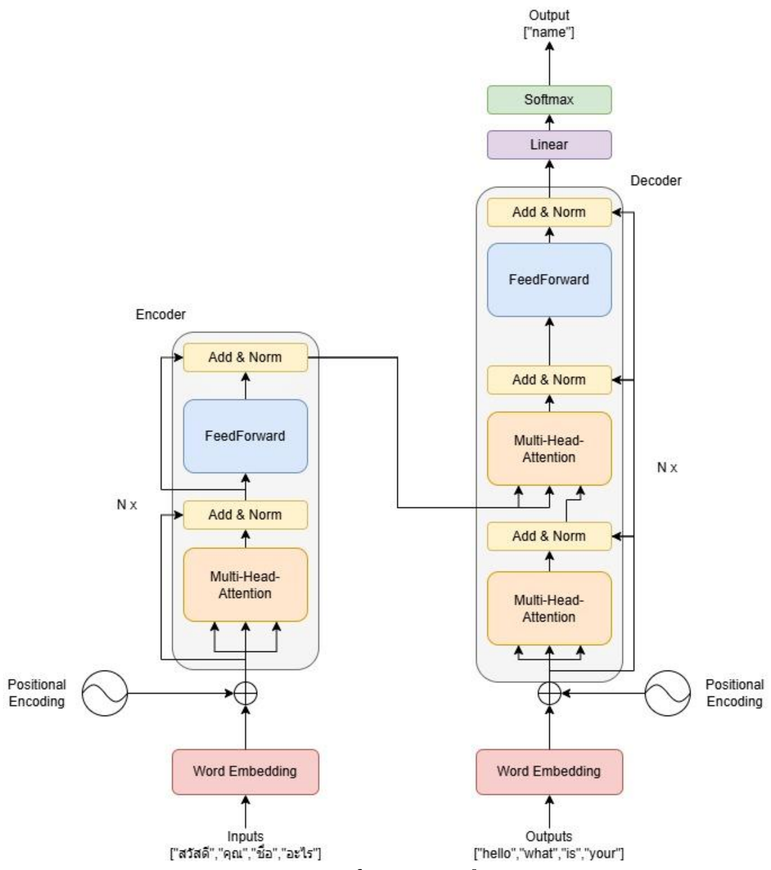


Figure 2.1 Transformer Architecture

- **Word-Embedding and Positional-Encoding:** The Word Embedding layer converts words into meaningful vector representations. These vectors are then combined with values from Positional Encoding, as defined in Equation **2.6.** This step enhances the positional features of words, addressing the fact that the Transformer model is not inherently a sequential model like RNNs or **1**D-CNNs.

$$\begin{aligned} PE_{(pos,2i)} &= \sin\left(pos/10000^{2i/d_{model}}\right) \\ PE_{(pos,2i+1)} &= \cos\left(pos/10000^{2i/d_{model}}\right) \end{aligned} \tag{2.6}$$

$pos$ is position of word in sentence.
$i$ is position of index in word vector.
$d_{model}$ is dimension of word vector.

- **Multi-Head-Attention:** The Multi-Head Attention mechanism consists of multiple Scaled Dot-Product Attention layers, as illustrated in Figure 2.2. Its primary function is to capture relationships between words.

  In Scaled Dot-Product Attention, inputs from Word Embedding and Positional Encoding are multiplied by three weight matrices, $W_q$, $W_k$ and $W_v$, to produce the matrices:
  - Query $(Q)$:Represents the word whose "importance" relative to other words is being determined.
  - Key $(K)$:Represents all words in the sequence, used to match against the Query.
  - Value $(V)$:Represents the final values used as the output of the attention mechanism.

  These $Q$ , $K$ , $V$ matrices are then used to compute attention scores based on Equation 2.7. In the Decoder, certain values may be masked to prevent the model from accessing future information.

  In Multi-Head Attention, attention outputs from multiple layers are concatenated and multiplied by the weight matrix $W_o$ to integrate attention information across all heads, as defined in Equation 2.8.

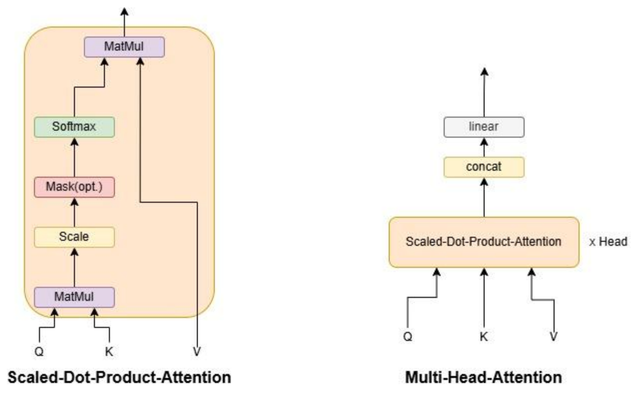


Figure 2.2 (left) Scaled-Dot-Product-Attention (right) Multi-Head-Attention

$$Attention(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2.7}$$

$Q$ is query matrix.
$K$is key matrix.
$V$ is value matrix.
$d_k$ is dimension of key matrix.

$$MultiHead(Q, K, V) = \text{concat}(head_1, \dots, head_2)\, W^o \quad (2.8)$$

$head_i$ is $Attention(Q_i, K_i, V_i)$.
$W^o$is weight output

- **Add And Norm**

  The Add and Norm mechanism consists of two key components:
  - Add (Residual Connection) [9]: A residual connection is applied to increase the model's degree of freedom, allowing it to skip certain layers and facilitate better gradient flow during training.
  - Norm (Normalization Layer) [10]: A normalization layer is used to standardize values, ensuring a mean of 0 and variance of 1.

  To enhance model flexibility, additional learnable parameters, Gamma (γ) and Beta (β), are introduced, enabling the model to adjust the scale and bias dynamically, as formulated in Equation 2.9.

$$y = \frac{x - E[x]}{\sqrt{Var[x] + \epsilon}} \gamma + \beta \quad (2.9)$$

γ is scale parameter (default value is 1).
β is bias parameter (default value is 0).
$\epsilon$ is eps (default value is 1e-5).

- **FeedForward**:The FeedForward network in the Transformer model is a standard neural network that typically consists of two hidden layers, as shown in Figure 2.3. The first layer is followed by a ReLU activation function, after which the output is passed to the second layer, as defined in Equation 2.10.

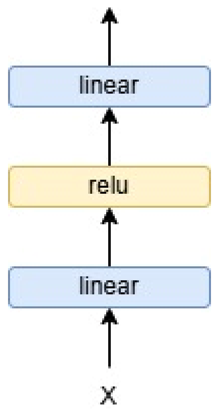


Figure 2.3 FeedForward

$$FNN(x) = \max(0, xW_1 + b_1)\,W_2 + b_2 \tag{2.10}$$

$W_1$is weight parameter of first layer.
$b_1$is bias parameter of first layer.
$W_2$is weight parameter of second layer.
$b_2$is bias parameter of second layer.

- **Softmax [11]:**The Softmax Activation Function transforms input values into a probability distribution, meaning that the sum of all output values equals 1, and each individual value falls within the range [0,1]. The Softmax function is defined by Equation 2.11.

$$Softmax(x_i) = \frac{e^i}{\sum_{j=1}^{n} e^j} \tag{2.11}$$

### 2.3.6 Generative Pre-trained Transformer [12]

The Generative Pre-trained Transformer (GPT) is a large-scale language model designed for various language tasks, such as text summarization, language translation, and question answering. GPT is derived from the Decoder component of the Transformer model, as illustrated in Figure 2.1, and is used to generate sentences.

GPT training is divided into two phases: Pretraining and Fine-tuning.

**Pretraining trains** the model to understand language by learning from a large corpus of text. The model samples sentences and predicts the next word. For example:

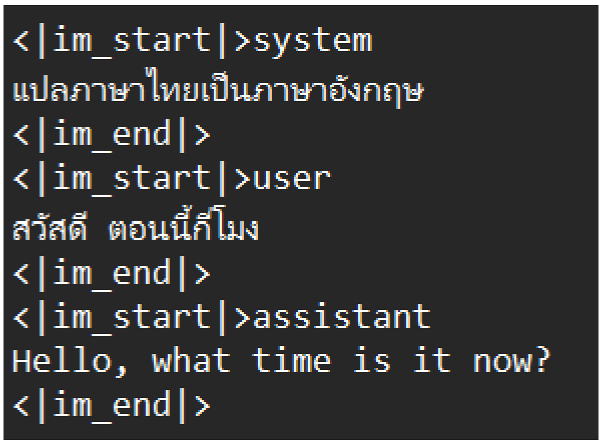
"The quick brown fox jumps over the lazy" → "dog"

**Fine-tuning** applies transfer learning to retrain the model after Pretraining, enabling it to perform specialized tasks such as language translation. For example:

Input
"Translate Thai to English" // System Prompt
"สวัสดี ตอนนี้กี่โมง"
Output
"Hello, what time is it now?"

ChatGPT [13] utilizes the ChatML [14] Format, as shown in Figure 2.4, for chatbot interactions. The ChatGPT training process follows Reinforcement Learning from Human Feedback (RLHF), where human evaluators rate chatbot responses.

To reduce the number of samples requiring human evaluation, a Reward Model is trained to automate scoring. The assigned scores are then used to refine the model, improving its ability to generate natural responses. The process follows Figure 2.5 and is implemented using Proximal Policy Optimization (PPO) in Reinforcement Learning (RL).



```
<|im_start|>system
แปลภาษาไทยเป็นภาษาอังกฤษ
<|im_end|>
<|im_start|>user
สวัสดี ตอนนี้กี่โมง
<|im_end|>
<|im_start|>assistant
Hello, what time is it now?
<|im_end|>
```

Figure 2.4 Chat ML Format

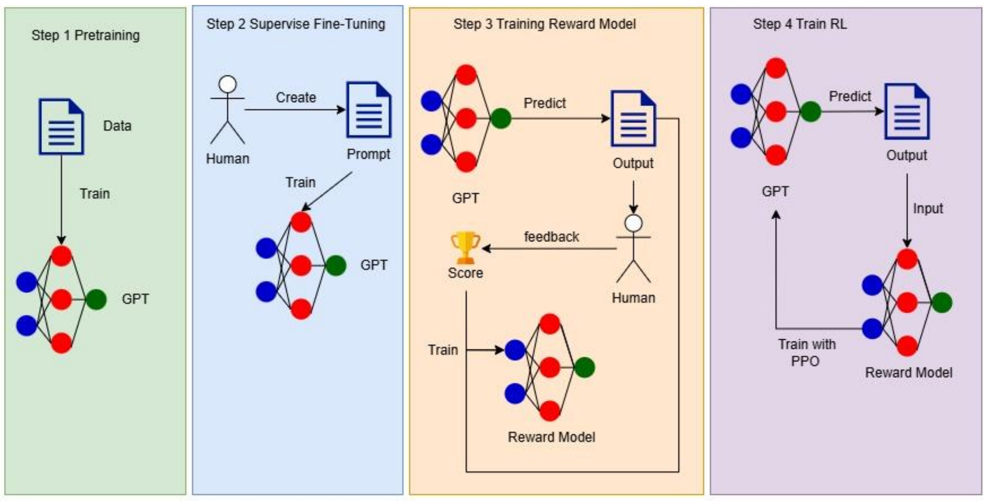


Figure 2.5 RLHF

### 2.3.7 Word Embedding [15]

Word Embedding is a method for converting words into meaningful vectors using a neural network. The words are first transformed into one-hot encoding before being fed into the model.There are two main models:

- Continuous Bag-of-Words (CBOW) Model: Predicts the center word using surrounding words.
- Skip-gram Model: Predicts the surrounding words using the center word instead.

Both models require specifying the number of nodes (vector dimensions) and the window size, which determines the range of words considered for analysis. After training, the output from the hidden layer (h1) is used as the word vector y, as shown in Figure 2.6 and Figure 2.7.

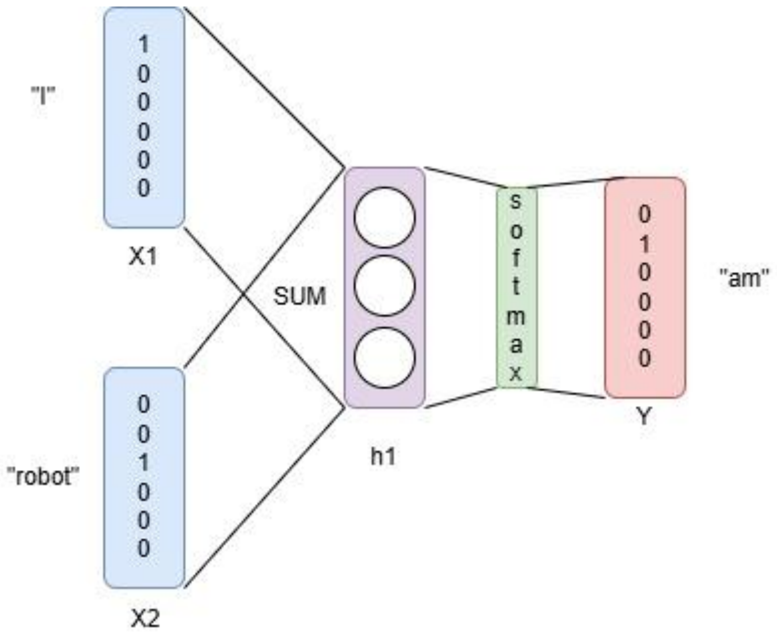


Figure 2.6 CBOW Architecture

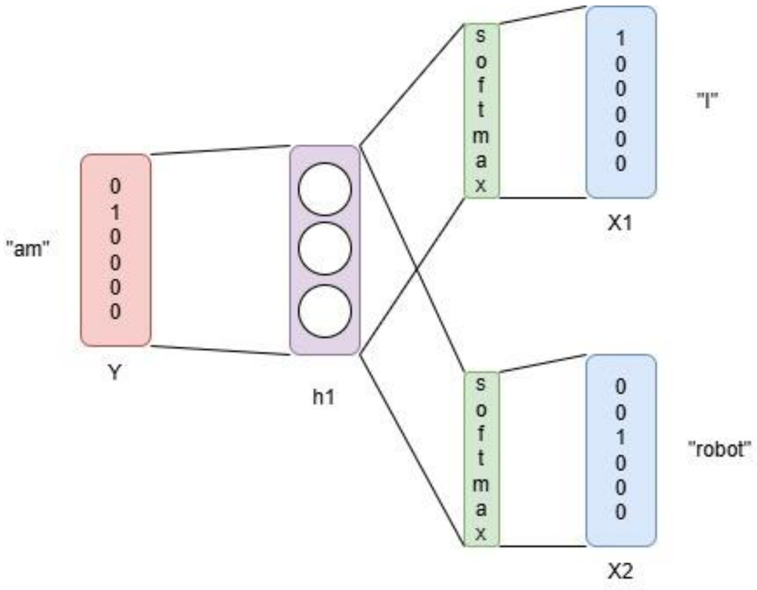


Figure 2.7 Skip-gram-Model

Both models can be used to transform words into meaningful vectors. However, when considering their use cases, CBOW is more suitable for large datasets as it trains faster than the Skip-gram model. Conversely, the Skip-gram model trains more slowly and is better suited for smaller datasets because it learns from the target word to multiple context words, allowing it to capture relationships between less frequent words more effectively.

### 2.3.8 Retrieval Augmented Generation [16]

Retrieval Augmented Generation (RAG) is a process that enables LLMs to access external information by retrieving data from an Information Retrieval system to use as context. RAG can be categorized into three main types as follows

- **NativeRAG**: This is the basic form of RAG, consisting of four main components: Query, Document, Information Retrieval, and LLM, as shown in Figure 2.8.
- **AdvancedRAG:** This extends NativeRAG by adding two additional components: Pre-Retrieval, which enhances document retrieval efficiency, and Post-Retrieval, which filters retrieved documents to improve answer generation. This prevents the LLM from receiving excessive information, reducing the risk of information overload. The overall process is illustrated in Figure 2.9.
- **ModularRAG**: This concept designs RAG as distinct modules, each with a specific function. These modules work together in various patterns, as shown in Figure 2.10. In real-world applications, the system selects the most suitable pattern based on the given task.

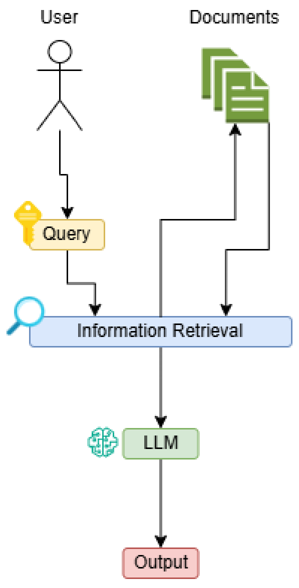


Figure 2.8 NativeRAG

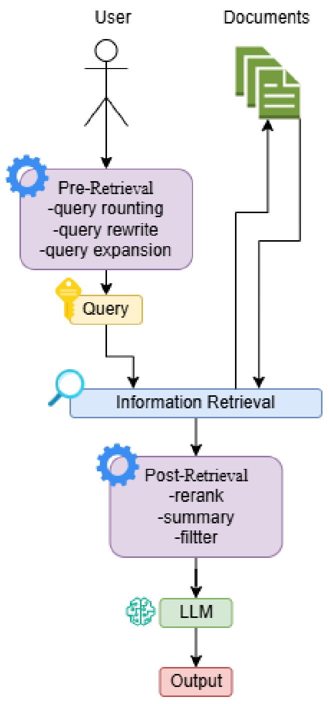


Figure 2.9 AdvancedRAG

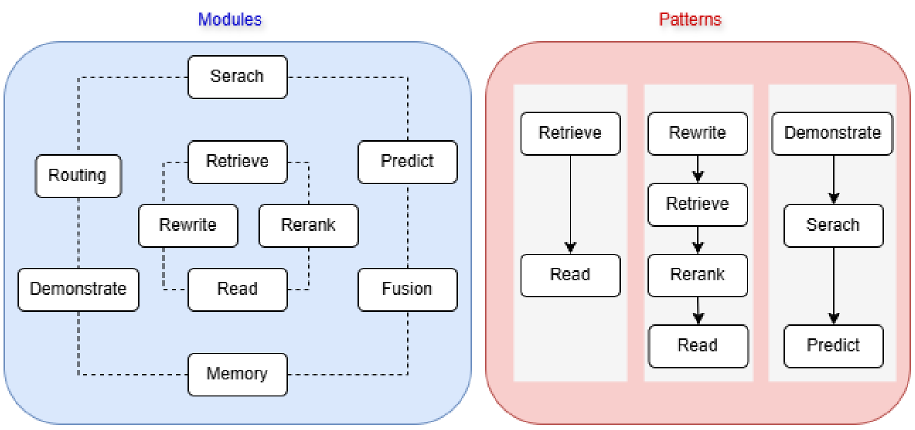


Figure 2.10 ModularRAG

Additionally, when comparing RAG with Fine-tuning, as shown in Figure 2.11, across two domains—External Knowledge and Internal Knowledge—it is evident that RAG excels in External Knowledge, while Fine-tuning is more effective for Internal Knowledge.

RAG offers greater flexibility by leveraging few-shot learning, which does not modify the model's weights, unlike Fine-tuning, which requires adjustments to some of the weights. However, while few-shot learning provides adaptability, it may not always be as effective as Fine-tuning for certain tasks.

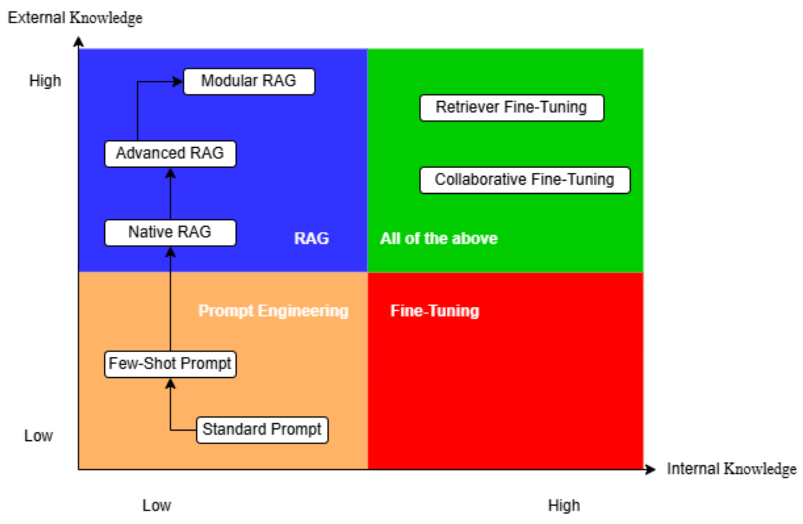


Figure 2.11 RAG vs Fine-Tuning

## 3 Data and Methodology

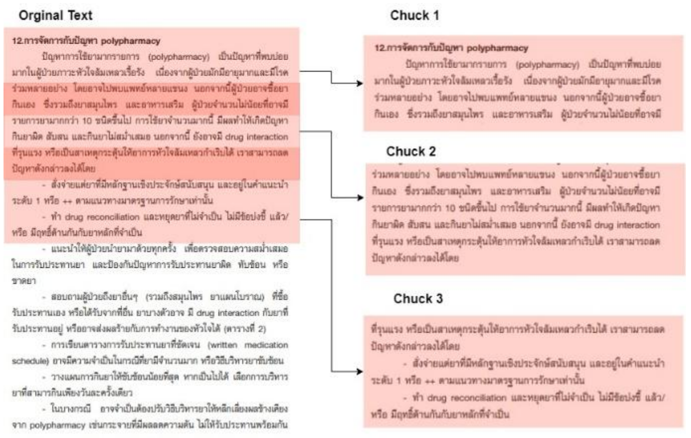
### 3.1 Data

The information used as a document source for the chatbot's responses will be based on the document “Comprehensive Heart Failure Management Program”, which contains a total of 24,174 words and consists of the following five chapters:

- Chapter 1 - Types, causes, pathophysiology Signs and symptoms of heart failure. Laboratory tests for diagnosis and treatment of heart failure.
- Chapter 2 - Drug treatment in heart failure patients with reduced heart muscle function.
- Chapter 3 - Integrated treatment management for heart failure patients.
- Chapter 4 - Common errors and problems in outpatient treatment of patients with chronic heart failure.
- Chapter 5 - The role of the heart failure clinic nurse.

The document will be divided into smaller articles and embedded using OpenAI Embedding and BM25, then processed through Pipecone for related document retrieval. This approach addresses the context window limitation of GPT-4, which supports up to 8,192 tokens per input, while also optimizing costs.

To divide the document effectively, we employ chunking with overlap. Each chunk contains a fixed number of words or sentences, ensuring manageable segments that facilitate accurate topic coverage.


Orginal Text

**12.การจัดการกับปัญหา polypharmacy**

ปัญหาการใช้ยามากรายการ (polypharmacy) เป็นปัญหาที่พบบ่อยมากในผู้ป่วยภาวะหัวใจล้มเหลวเรื้อรัง เนื่องจากผู้ป่วยมักมีอายุมากและมีโรคร่วมหลายอย่าง โดยอาจไปพบแพทย์หลายแขนง นอกจากนี้ผู้ป่วยอาจซื้อยากินเอง ซึ่งรวมถึงยาสมุนไพร และอาหารเสริม ผู้ป่วยจำนวนไม่น้อยที่อาจมีรายการยามากกว่า 10 ชนิดขึ้นไป การใช้ยาจำนวนมากนี้ มีผลทำให้เกิดปัญหากินยาผิด สับสน และกินยาไม่สม่ำเสมอ นอกจากนี้ ยังอาจมี drug interaction ที่รุนแรง หรือเป็นสาเหตุกระตุ้นให้อาการหัวใจล้มเหลวกำเริบได้ เราสามารถลดปัญหาดังกล่าวลงได้โดย

- สั่งจ่ายแต่ยาที่มีหลักฐานเชิงประจักษ์สนับสนุน และอยู่ในคำแนะนำระดับ 1 หรือ ++ ตามแนวทางมาตรฐานการรักษาเท่านั้น
- ทำ drug reconciliation และหยุดยาที่ไม่จำเป็น ไม่มีข้อบ่งชี้ แล้ว/หรือ มีฤทธิ์ต้านกันกับยาหลักที่จำเป็น
- แนะนำให้ผู้ป่วยนำยามาด้วยทุกครั้ง เพื่อตรวจสอบความสม่ำเสมอในการรับประทานยา และป้องกันปัญหาการรับประทานยาผิด ทับซ้อน หรือขาดยา
- สอบถามผู้ป่วยถึงยาอื่นๆ (รวมถึงสมุนไพร ยาแผนโบราณ) ที่ซื้อรับประทานเอง หรือได้รับจากที่อื่น ยาบางตัวอาจ มี drug interaction กับยาที่รับประทานอยู่ หรืออาจส่งผลร้ายกับการทำงานของหัวใจได้ (ตารางที่ 2)
- การเขียนตารางการรับประทานยาที่ชัดเจน (written medication schedule) อาจมีความจำเป็นในกรณีที่ยามีจำนวนมาก หรือวิธีบริหารยาซับซ้อน
- วางแผนการกินยาให้ซับซ้อนน้อยที่สุด หากเป็นไปได้ เลือกการบริหารยาที่สามารถกินเพียงวันละครั้งเดียว
- ในบางกรณี อาจจำเป็นต้องปรับวิธีบริหารยาให้หลีกเลี่ยงผลข้างเคียงจาก polypharmacy เช่นกระจายที่มีผลลดความดัน ไม่ให้รับประทานพร้อมกัน

Chuck 1

**12.การจัดการกับปัญหา polypharmacy**

ปัญหาการใช้ยามากรายการ (polypharmacy) เป็นปัญหาที่พบบ่อยมากในผู้ป่วยภาวะหัวใจล้มเหลวเรื้อรัง เนื่องจากผู้ป่วยมักมีอายุมากและมีโรคร่วมหลายอย่าง โดยอาจไปพบแพทย์หลายแขนง นอกจากนี้ผู้ป่วยอาจซื้อยากินเอง ซึ่งรวมถึงยาสมุนไพร และอาหารเสริม ผู้ป่วยจำนวนไม่น้อยที่อาจมี

Chuck 2

ร่วมหลายอย่าง โดยอาจไปพบแพทย์หลายแขนง นอกจากนี้ผู้ป่วยอาจซื้อยากินเอง ซึ่งรวมถึงยาสมุนไพร และอาหารเสริม ผู้ป่วยจำนวนไม่น้อยที่อาจมีรายการยามากกว่า 10 ชนิดขึ้นไป การใช้ยาจำนวนมากนี้ มีผลทำให้เกิดปัญหากินยาผิด สับสน และกินยาไม่สม่ำเสมอ นอกจากนี้ ยังอาจมี drug interaction ที่รุนแรง หรือเป็นสาเหตุกระตุ้นให้อาการหัวใจล้มเหลวกำเริบได้ เราสามารถลดปัญหาดังกล่าวลงได้โดย

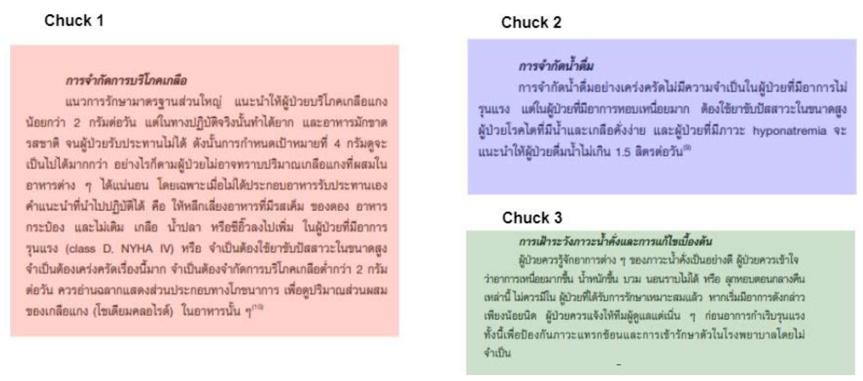
Chuck 3

ที่รุนแรง หรือเป็นสาเหตุกระตุ้นให้อาการหัวใจล้มเหลวกำเริบได้ เราสามารถลดปัญหาดังกล่าวลงได้โดย

- สั่งจ่ายแต่ยาที่มีหลักฐานเชิงประจักษ์สนับสนุน และอยู่ในคำแนะนำระดับ 1 หรือ ++ ตามแนวทางมาตรฐานการรักษาเท่านั้น
- ทำ drug reconciliation และหยุดยาที่ไม่จำเป็น ไม่มีข้อบ่งชี้ แล้ว/หรือ มีฤทธิ์ต้านกันกับยาหลักที่จำเป็น

Figure 3.1 fix size base preprocess

The challenge arises when a question like "how to deal with polypharmacy problems" finds the best match in the first chunk but not in subsequent chunks. To resolve this issue, manual segmentation is applied per topic, ensuring each segment contains less than 1,500 tokens.


Chuck 1

*การจำกัดการบริโภคเกลือ*

แนวการรักษามาตรฐานส่วนใหญ่ แนะนำให้ผู้ป่วยบริโภคเกลือแกงน้อยกว่า 2 กรัมต่อวัน แต่ในทางปฏิบัติจริงนั้นทำได้ยาก และอาหารมักขาดรสชาติ จนผู้ป่วยรับประทานไม่ได้ ดังนั้นการกำหนดเป้าหมายที่ 4 กรัมดูจะเป็นไปได้มากกว่า อย่างไรก็ตามผู้ป่วยไม่อาจทราบปริมาณเกลือแกงที่ผสมในอาหารต่าง ๆ ได้แน่นอน โดยเฉพาะเมื่อไม่ได้ประกอบอาหารรับประทานเอง คำแนะนำที่นำไปปฏิบัติได้ คือ ให้หลีกเลี่ยงอาหารที่มีรสเค็ม ของดอง อาหารกระป๋อง และไม่เติม เกลือ น้ำปลา หรือซีอิ๊วลงไปเพิ่ม ในผู้ป่วยที่มีอาการรุนแรง (class D, NYHA IV) หรือ จำเป็นต้องใช้ยาขับปัสสาวะในขนาดสูง จำเป็นต้องเคร่งครัดเรื่องนี้มาก จำเป็นต้องจำกัดการบริโภคเกลือต่ำกว่า 2 กรัมต่อวัน ควรอ่านฉลากแสดงส่วนประกอบทางโภชนาการ เพื่อดูปริมาณส่วนผสมของเกลือแกง (โซเดียมคลอไรด์) ในอาหารนั้น ๆ[13]

Chuck 2

*การจำกัดน้ำดื่ม*

การจำกัดน้ำดื่มอย่างเคร่งครัดไม่มีความจำเป็นในผู้ป่วยที่มีอาการไม่รุนแรง แต่ในผู้ป่วยที่มีอาการหอบเหนื่อยมาก ต้องใช้ยาขับปัสสาวะในขนาดสูง ผู้ป่วยโรคไตที่มีน้ำและเกลือคั่งง่าย และผู้ป่วยที่มีภาวะ hyponatremia จะแนะนำให้ผู้ป่วยดื่มน้ำไม่เกิน 1.5 ลิตรต่อวัน[8]

Chuck 3

*การเฝ้าระวังภาวะน้ำคั่งและการแก้ไขเบื้องต้น*

ผู้ป่วยควรรู้จักอาการต่าง ๆ ของภาวะน้ำคั่งเป็นอย่างดี ผู้ป่วยควรเข้าใจว่าอาการเหนื่อยมากขึ้น น้ำหนักขึ้น บวม นอนราบไม่ได้ หรือ ลุกหอบตอนกลางคืนเหล่านี้ ไม่ควรมีใน ผู้ป่วยที่ได้รับการรักษาเหมาะสมแล้ว หากเริ่มมีอาการดังกล่าวเพียงน้อยนิด ผู้ป่วยควรแจ้งให้ทีมผู้ดูแลแต่เนิ่น ๆ ก่อนอาการกำเริบรุนแรง ทั้งนี้เพื่อป้องกันภาวะแทรกซ้อนและการเข้ารักษาตัวในโรงพยาบาลโดยไม่จำเป็น

Figure 3.2 topic base preprocess

Finally each method will have a different word count and number of chunks, as shown in Table 3.1

| Method | Number of chunks | Words per chunks | Overlap words per chunks |
|---|---|---|---|
| **Topic-base** | **70** | **-** | **-** |
| **N500-O50** | **54** | **500** | **50** |
| **N500-O100** | **61** | **500** | **100** |
| **N1000-O50** | **26** | **1000** | **50** |
| **N1000-O100** | **27** | **1000** | **100** |

Table 3.1 Document preparation detail

## 3.2 Development

### 3.2.1 Developing mobile applications with authentication and version control systems.

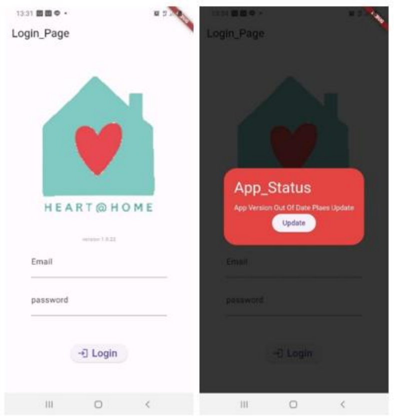


Figure 3.3 UI (left) Login page (right) Application Status

From Figure 3.3 on the left illustrates the authentication system in operation, while the picture on the right depicts the application update system.

### 3.2.2 Develop a Chat Room System

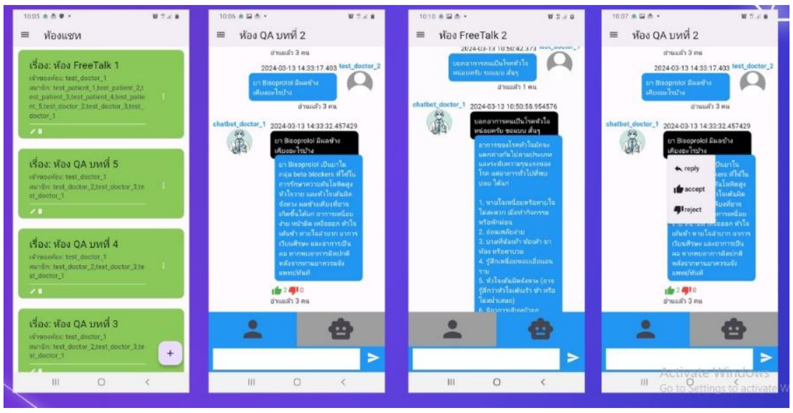


Figure 3.4 UI Application Chatroom

In Figure 3.4 from the left screen, users can view a list of chatrooms upon entering. This interface resembles the three left pictures. Within each chatroom, users can seamlessly switch between chatting with another user and interacting with a chatbot using a toggle tab. The right picture illustrates how a doctor user verifies answers provided by the chatbot.

### 3.2.3 Development of Chatbot System

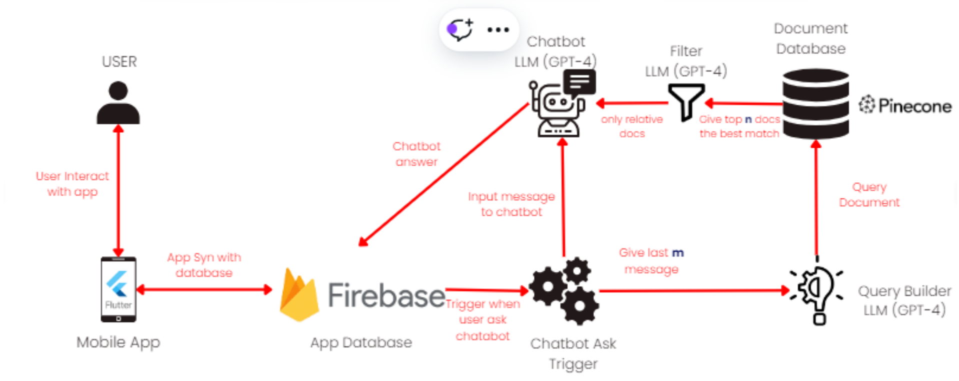


Figure 3.5 overview of chatbot system

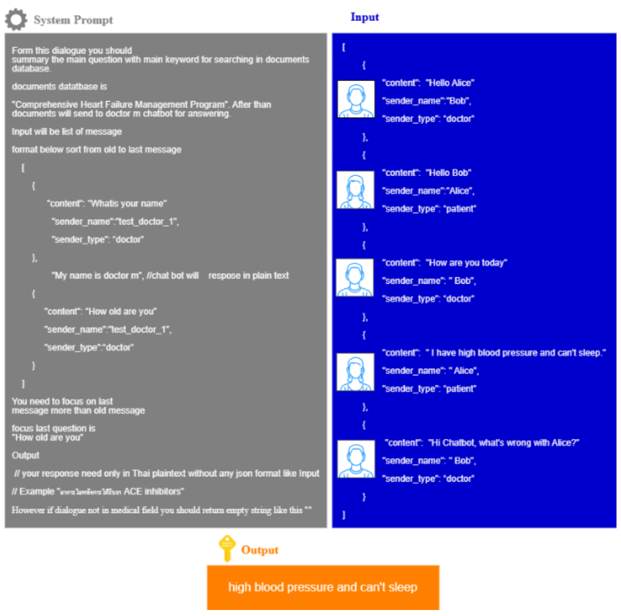
From Figure 3.5, when a user sends a question from the mobile application, the message is stored in Firebase Firestore. If triggered, the system retrieves the last m messages to extract keywords or the main sentence using a Query Builder. These key elements are used to query the top n related documents, which are sent to the Filter for rejecting non-related document. Finally, the chatbot generates answers based on this context and stores the reply message in Firebase Firestore as a response to the original question.

**Prompt Design**

The design of a prompt consists of three main components: Query Builder, Filter, and Answer Generator. The Query Builder is responsible for generating queries used to search for information from the Information Retrieval system. This component is detailed in Figure 3.6, which includes the system specifications for defining tasks and relevant details for the LLM, along with example inputs and outputs obtained from this prompt.

Once the output from Figure 3.6 is obtained, it is passed to the Information Retrieval system as shown in Figure 3.7 to retrieve documents relevant to the query for use as context. The retrieved documents are then sent to the Filter, as described in Figure 3.8, to reject irrelevant documents. This process enhances the chatbot’s ability to provide accurate responses, as failing to remove unrelated documents may cause the chatbot to hallucinate attempting to generate answers based on documents that lack the necessary information, resulting in responses containing unnecessary details.

Finally, the chat dialogue and the filtered documents are forwarded to the Answer Generator, as outlined in Figure 3.9, to generate the final response.

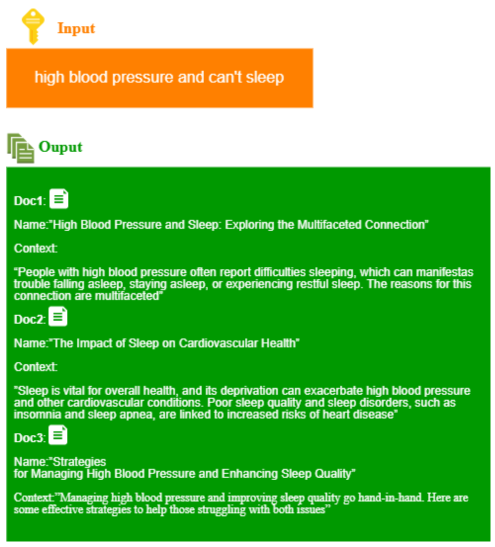


```
Form this dialogue you should
summary the main question with main keyword for searching in documents
database.

documents datatbase is

"Comprehensive Heart Failure Management Program". After than
documents will send to doctor m chatbot for answering.

Input will be list of message

format below sort from old to last message
  [
    {
        "content": "Whatis your name"
         "sender_name":"test_doctor_1",
         "sender_type": "doctor"
    },
         "My name is doctor m", //chat bot will   respose in plain text
    {
       "content": "How old are you"
       "sender_name":"test_doctor_1",
       "sender_type":"doctor"
    }
  ]
You need to focus on last
message more than old message

focus last question is
"How old are you"

Output

 // your response need only in Thai plaintext without any json format like Input

// Example "อาการไอหลังการได้รับยา ACE inhibitors"

However if dialogue not in medical field you should return empty string like this ""
```

```
[
    {
  "content":  "Hello Alice"
  "sender_name":"Bob",
  "sender_type": "doctor"
    },
    {
  "content":  "Hello Bob"
  "sender_name":"Alice",
  "sender_type": "patient"
    },
    {
  "content":  "How are you today"
  "sender_name": " Bob",
  "sender_type": "doctor"
    },
    {
  "content":  " I have high blood pressure and can't sleep."
  "sender_name": " Alice",
  "sender_type": "patient"
    },
    {
   "content":  "Hi Chatbot, what's wrong with Alice?"
  "sender_name": " Bob",
  "sender_type": "doctor"
    }
]
```

```
high blood pressure and can't sleep
```

Figure 3.6 Query Builder Prompt



high blood pressure and can't sleep

Doc1:

Name:"High Blood Pressure and Sleep: Exploring the Multifaceted Connection"

Context:

"People with high blood pressure often report difficulties sleeping, which can manifestas trouble falling asleep, staying asleep, or experiencing restful sleep. The reasons for this connection are multifaceted"

Doc2:

Name:"The Impact of Sleep on Cardiovascular Health"

Context:

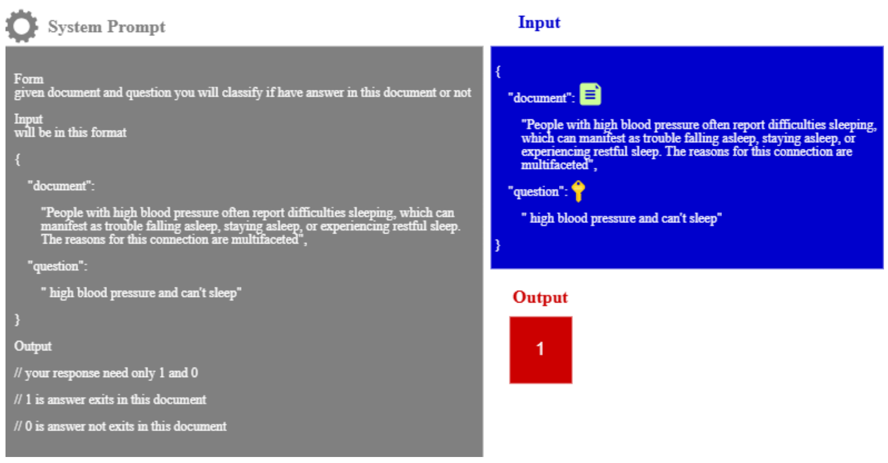
"Sleep is vital for overall health, and its deprivation can exacerbate high blood pressure and other cardiovascular conditions. Poor sleep quality and sleep disorders, such as insomnia and sleep apnea, are linked to increased risks of heart disease"

Doc3:

Name:"Strategies
for Managing High Blood Pressure and Enhancing Sleep Quality"

Context:"Managing high blood pressure and improving sleep quality go hand-in-hand. Here are some effective strategies to help those struggling with both issues"

Figure 3.7 IR example

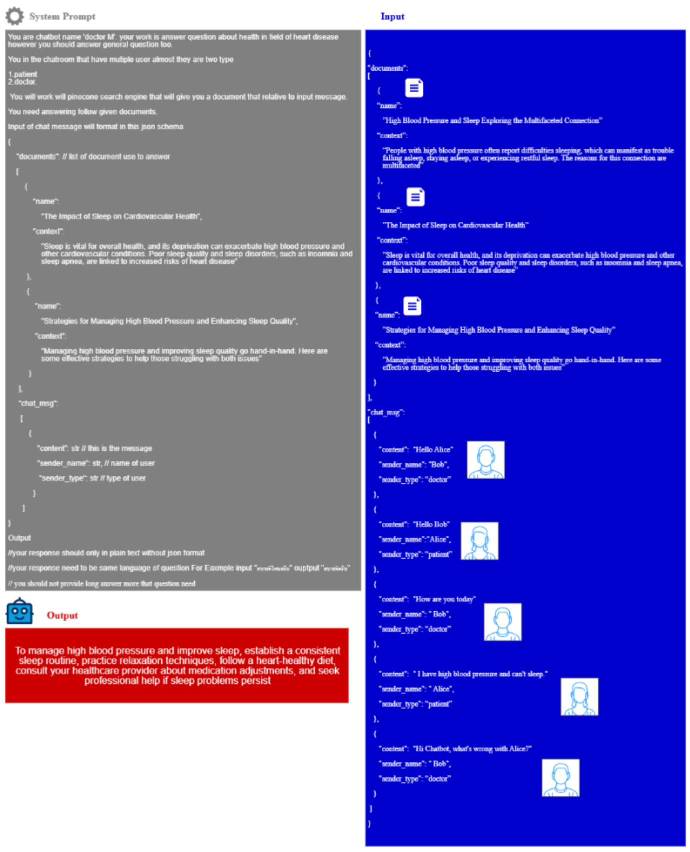


```
Form
given document and question you will classify if have answer in this document or not
Input
will be in this format
{
  "document":
    "People with high blood pressure often report difficulties sleeping, which can
    manifest as trouble falling asleep, staying asleep, or experiencing restful sleep.
    The reasons for this connection are multifaceted",
  "question":
    " high blood pressure and can't sleep"
}
Output
// your response need only 1 and 0
// 1 is answer exits in this document
// 0 is answer not exits in this document
```

```
{
  "document":
    "People with high blood pressure often report difficulties sleeping,
    which can manifest as trouble falling asleep, staying asleep, or
    experiencing restful sleep. The reasons for this connection are
    multifaceted",
  "question":
    " high blood pressure and can't sleep"
}
```

1

Figure 3.8 Filter Prompt



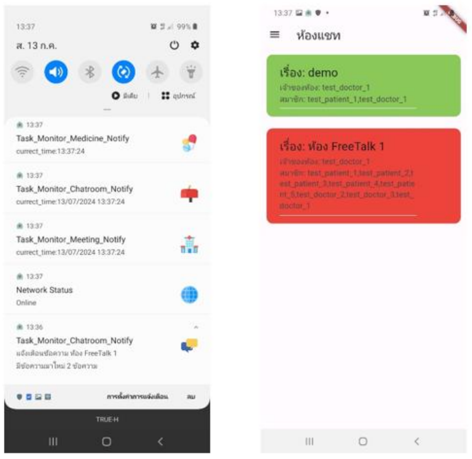
```
You are chatbot name 'doctor M'. your work is answer question about health in field of heart disease
however you should answer general question too.

You in the chatroom that have mutiple user almost they are two type

1.patient
2.doctor.

 You will work will pinecone search engine that will give you a document that relative to input message.

You need answering follow given documents.

Input of chat message will format in this json schema

{
   "documents": // list of document use to answer
   [
      {
         "name":
            "The Impact of Sleep on Cardiovascular Health",
         "context":
            "Sleep is vital for overall health, and its deprivation can exacerbate high blood pressure and
            other cardiovascular conditions. Poor sleep quality and sleep disorders, such as insomnia and
            sleep apnea, are linked to increased risks of heart disease"
      },
      {
         "name":
            "Strategies for Managing High Blood Pressure and Enhancing Sleep Quality",
         "context":
            "Managing high blood pressure and improving sleep quality go hand-in-hand. Here are
            some effective strategies to help those struggling with both issues"
      }
   ],
   "chat_msg":
   [
      {
         "content": str // this is the message
         "sender_name": str, // name of user
          "sender_type": str // type of user
      }
   ]
}
Output
//your response should only in plain text without json format
//your response need to be same language of question For Eaxmple input "[illegible]" ouptput "[illegible]"
// you should not provide long answer more that question need
```

```
{
"documents":
[
  {
  "name":
    "High Blood Pressure and Sleep Exploring the Multifaceted Connection"
  "context":
    "People with high blood pressure often report difficulties sleeping, which can manifest as trouble
    falling asleep, staying asleep, or experiencing restful sleep. The reasons for this connection are
    multifaceted"
  },
  {
  "name":
    "The Impact of Sleep on Cardiovascular Health"
  "context":
    "Sleep is vital for overall health, and its deprivation can exacerbate high blood pressure and other
    cardiovascular conditions. Poor sleep quality and sleep disorders, such as insomnia and sleep apnea,
    are linked to increased risks of heart disease"
  },
  {
  "name":
    "Strategies for Managing High Blood Pressure and Enhancing Sleep Quality"
  "context":
    "Managing high blood pressure and improving sleep quality go hand-in-hand. Here are some
    effective strategies to help those struggling with both issues"
  }
],
"chat_msg":
[
  {
   "content":  "Hello Alice"
   "sender_name": "Bob",
   "sender_type": "doctor"
  },
  {
   "content":  "Hello Bob"
   "sender_name":"Alice",
   "sender_type": "patient"
  },
  {
   "content":  "How are you today"
   "sender_name": " Bob",
   "sender_type": "doctor"
  },
  {
   "content":  " I have high blood pressure and can't sleep."
   "sender_name": " Alice",
   "sender_type": "patient"
  },
  {
   "content":  "Hi Chatbot, what's wrong with Alice?"
   "sender_name": " Bob",
   "sender_type": "doctor"
  }
 ]
}
```

```
To manage high blood pressure and improve sleep, establish a consistent
sleep routine, practice relaxation techniques, follow a heart-healthy diet,
consult your healthcare provider about medication adjustments, and seek
professional help if sleep problems persist
```

Figure 3.9 Answer Generate Prompt

### 3.2.4 Development of Chat Room Notification System



13:37
Task_Monitor_Medicine_Notify
currect_time:13:37:24

13:37
Task_Monitor_Chatroom_Notify
currect_time:13/07/2024 13:37:24

13:37
Task_Monitor_Meeting_Notify
currect_time:13/07/2024 13:37:24

13:37
Network Status
Online

13:36
Task_Monitor_Chatroom_Notify
แจ้งเตือนข้อความ ห้อง FreeTalk 1
มีข้อความมาใหม่ 2 ข้อความ

เรื่อง: demo
เจ้าของห้อง: test_doctor_1
สมาชิก: test_patient_1,test_doctor_1

เรื่อง: ห้อง FreeTalk 1
เจ้าของห้อง: test_doctor_1
สมาชิก: test_patient_1,test_patient_2,test_patient_3,test_patient_4,test_patient_5,test_doctor_2,test_doctor_3,test_doctor_1

Figure 3.10 UI (left) Application Notify (right) Chatroom Notify

In Figure 3.6 on the left picture displays the notification bar, which alerts users when the application is not in the foreground. Users have the option to configure notification settings in the settings page to hide these notifications.

On the right, notifications are shown within chat rooms when the application is in the foreground. A red background color on the room signifies unread messages

## 3.3 Deploy

During the deployment phase, the system will integrate three main components. First, the mobile applications will be deployed for Android users through Open Testing on the Play Store and for iOS users via TestFlight. Second, the chatbot will be packaged as a Docker container and deployed on a Virtual Machine (VM) in the Cloud environment to ensure scalability and accessibility. Third, data management will involve storing mobile application data in Firebase Firestore, while document vectors will be stored in Pinecone for efficient retrieval and management. This deployment strategy aims to optimize performance and user accessibility across different platforms while leveraging cloud infrastructure for robust operation

# 4 EVALUATION

This chapter provides a detailed overview of the experimental setup, including the configurations and scenario used in the study. It also explains the evaluation metrics in each experiment.

## 4.1 Experimental Settings

### 4.1.1 Experimental 1: Testing Chatbot Accuracy

In Experimental 1, we aim to assess the chatbot's accuracy in answering questions and retrieving relevant articles. We have prepared 100 questions categorized as follows:

**Question**

- **Question Type 1**: Medical questions directly from the provided document. ( 50 questions in total, with 10 questions allocated per chapter of the document.)
- **Question Type 2**: Medical questions referring to a third person from the provided document. (25 questions in total, with 5 questions allocated per chapter of the document.)
- **Question Type 3**: General questions referring to a third person.(25 questions in total.)

Each answer generated by the chatbot will be verified by three collaborators to ensure correctness and evaluate the chatbot's performance in understanding and responding to diverse types of queries. This experimental setup aims to validate the effectiveness of the chatbot in medical question answering and information retrieval scenarios

**Experimental 2: Role-playing Scenario Testing**

Experimental 2 involves testing with 28 participants, including 14 doctors and 14 patients. Each participant will role-play based on given scenarios, forming pairs consisting of 1 doctor, 1 patient, and 1 chatbot. The experiment includes the following steps:

- Each pair will engage in two chatrooms per scenario:
  1. The first chatroom for single chat interactions.
  2. The second chatroom for group chat interactions.
- The experiment will be conducted in two phases:
  1. The first phase involves the first 7 pairs using single chat first followed by group chat.
  2. The second phase involves the remaining 7 pairs using group chat first followed by single chat
- After completing both chat scenarios, all participants will provide feedback on their experiences with the project.

This experimental setup aims to evaluate the effectiveness of the chatbot in both single and group chat settings, as well as gather user feedback to assess usability and functionality from both medical professionals and patients.

### 4.1.3 Experimental 3: Dynamic Role-playing Testing

Experimental 3 involves testing with 3 patients and 6 doctors, where each patient interacts with 2 doctors for their care. Similar to Experiment 2, participants will engage in role-playing scenarios with the following approach:

- Each patient will interact with 2 doctors and 1 chatbot
- The experiment includes both single chat and group chat scenarios without fixed scenarios.
- Participants will use both chatroom formats interchangeably based on the needs of the interaction.
- The objective is to evaluate the chatbot's performance in dynamic healthcare settings with varying interactions and scenarios.

This experimental design aims to assess the chatbot's adaptability and effectiveness in supporting patient-doctor interactions across different healthcare contexts.

## 4.2 Evaluation Metrics

### 4.2.1 Experimental 1: Chatbot Performance Evaluation

**Question Answering Accuracy**

The correctness of answers provided by the chatbot will be assessed using a majority vote from 3 collaborators, ensuring accuracy in medical question responses.

**Information Retrieval System Accuracy [17]**

For document searching, Precision@K, Recall@K, Mean Average Precision (mAP), and Mean Reciprocal Rank (mRR) metrics will be employed to evaluate the effectiveness of the information retrieval system in retrieving relevant documents based on user queries.

- **Precision@K**
  This metric measures the proportion of relevant documents among the top K retrieved documents.

$$Precision@K = \frac{|\{relevant\_documents@K\}|}{K} \tag{4.1}$$

- **Recall@K**
  This measures the proportion of relevant documents retrieved in the top K results out of all relevant documents available.

$$Recall@K = \frac{|\{relevant\_documents@K\}|}{|\{all\_reevant\_documents\}|} \tag{4.2}$$

- **ReciprocalRank@K**
  This metric evaluates how early the first relevant document appears in the ranked list.

$$Reciprocal\ Rank@K = \frac{1}{rank\_of\_first\_relevant\_document@K} \tag{4.3}$$

- **F1-Score@K**
  is the harmonic mean of Precision@K and Recall@K, providing a balanced measure of accuracy.

$$F1@K = 2 \cdot \frac{Precision@K \cdot Recall@K}{Precision@K + Recall@K} \tag{4.4}$$

**Response Time Analysis**

Response times of chatbot messages will be measured from chat logs. Descriptive statistics including mean, standard deviation, minimum, and maximum will be calculated to analyze response time data. Correlation coefficients will be applied to examine relationships such as response time vs. length of input text, and response time vs. length of output text. Data visualization techniques such as histograms and scatter plots will be used to illustrate these correlations.

**Experimental 2: Hypothesis Testing**

To evaluate the impact of group chat on dialogue smoothness compared to single chat, we will employ a t-test hypothesis with a significance level of 0.05 using feedback data where each question is scored on a scale of 1 to 5. The dataset will be analyzed across two domains:

1. **User Type Comparison (Patient vs. Doctor)**: Questions will be analyzed to compare feedback scores between patients and doctors by using independent t-test.
2. **Chat Type Comparison (Single Chat vs. Group Chat)**: Questions will be analyzed to compare feedback scores between single chat and group chat by using paired t-test.

**Questions for Evaluation**

1. Is the UX/UI beautiful and modern?
2. Ease of use?
3. Speed of response time?
4. Accuracy of answers?
5. The usefulness of chatbots?
6. The novelty of chatbots?
7. In the future, do you want to return to using chatbots?
8. Comments (note: not used for analysis)

The t-test will provide statistical evidence to validate whether group chat enhances dialogue smoothness compared to single chat, based on user feedback across various aspects of usability and performance. This evaluation aims to support the assumption and provide insights into the preferred chat interaction mode in the experimental setting.

**Experimental 3: Real Patient and Doctor Interaction Testing**

involves evaluating real patient and doctor interactions, mirroring the setup of Experimental 2 but with actual healthcare professionals and patients. This iteration aims to validate findings from simulated scenarios by observing how real-world dynamics influence dialogue smoothness and user perceptions in both single and group chat settings. The study leverages feedback and statistical analysis to assess the effectiveness of these interactions in enhancing user experience and optimizing chatbot functionality within healthcare contexts.

## 5 RESULTS AND DISCUSSION

### 5.1 Experimental 1: Chatbot Performance Result

#### 5.1.1 Information Retrieval System Accuracy

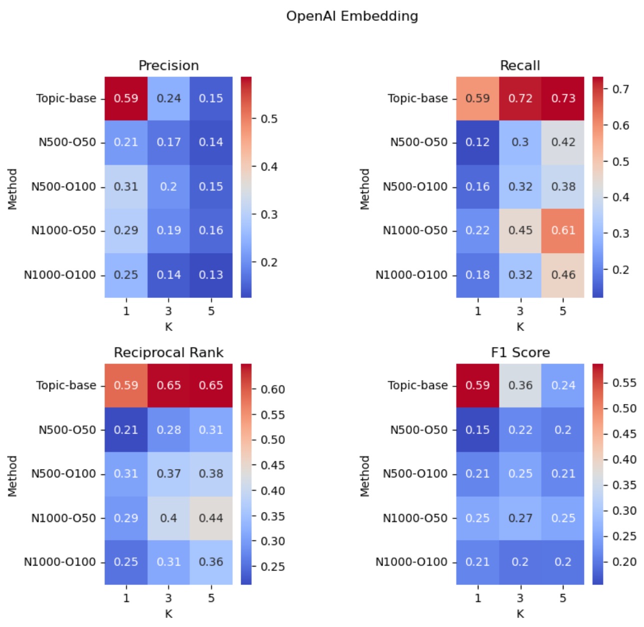


Figure 5.1 IR Metrics Result OpenAI

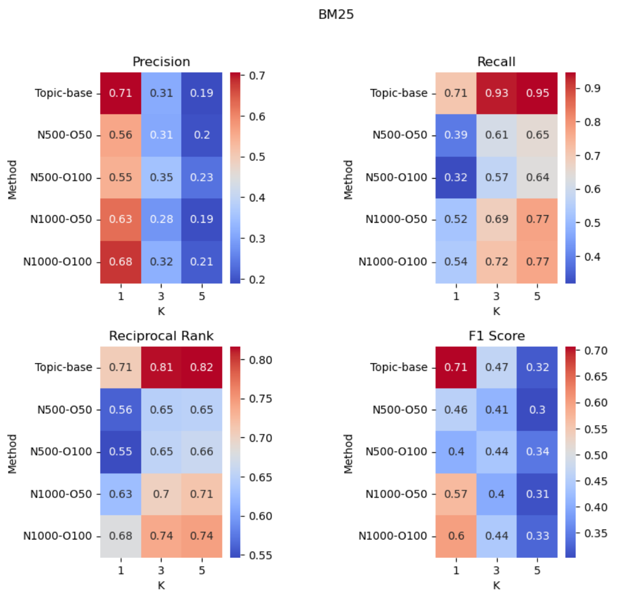


Figure 5.2 IR Metrics Result BM25

Form Figure 5.1 and Figure 5.2 the best document preparation and embedding method was **Topic-based BM25**, with the following performance metrics:

- **Precision@1**: 0.71
- **Recall@5**: 0.95
- **Reciprocal Rank@5**: 0.82
- **F1@1**: 0.71

For practical use, **K = 3** was chosen since increasing **K** to 5 did not significantly improve accuracy.

### 5.1.2 Chatbot Question Answering Accuracy

| **Model** | **Total ACC** | **Total Accept** | **Total Reject** | **ACC QT1** | **ACC QT2** | **ACC QT3** |
|---|---|---|---|---|---|---|
| **OpenAI** | **0.64** | **0.68** | **0.31** | **0.58** | **0.64** | **0.74** |
| **BM25** | **0.74** | **0.73** | **0.26** | **0.72** | **0.60** | **0.92** |
| **BM25+Filter** | **0.81** | **0.85** | **0.14** | **0.82** | **0.80** | **0.80** |

Table 5.1 Acc Chatbot

From Table 5.1, it can be observed that BM25+Filter is the best model at total accuracy at 0.81, despite the fact that its ACC for QT3 is lower than that of BM25.

### 5.1.3 Response Time Analysis

| **Model** | **Mean (second)** | **Std (second)** | **Min (second)** | **Max (second)** |
|---|---|---|---|---|
| **OpenAI** | **40.88** | **51.01** | **9.00** | **323.00** |
| **BM25** | **24.03** | **12.77** | **7.00** | **76.0** |
| **BM25+Filter** | **30.44** | **40.54** | **5.00** | **381.00** |

Table 5.2 Response Time Chatbot

From Table 5.2, it is evident that, on average, BM25 provides the fastest response at average response time 24.03 $\pm$ 12.77 second. However, in practical applications, BM25+Filter is preferred due to its higher emphasis on accuracy over response speed.

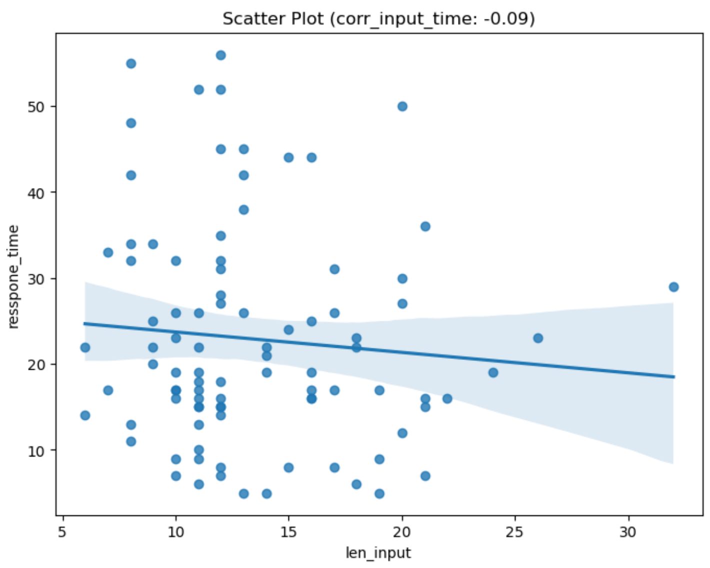


Figure 5.3 Correlation Input Token and Response Time

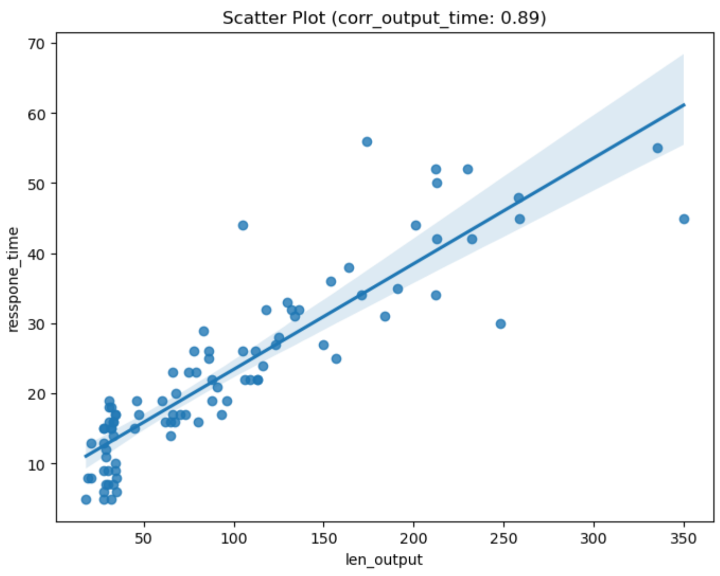


Figure 5.4 Correlation Output Token and Response Time

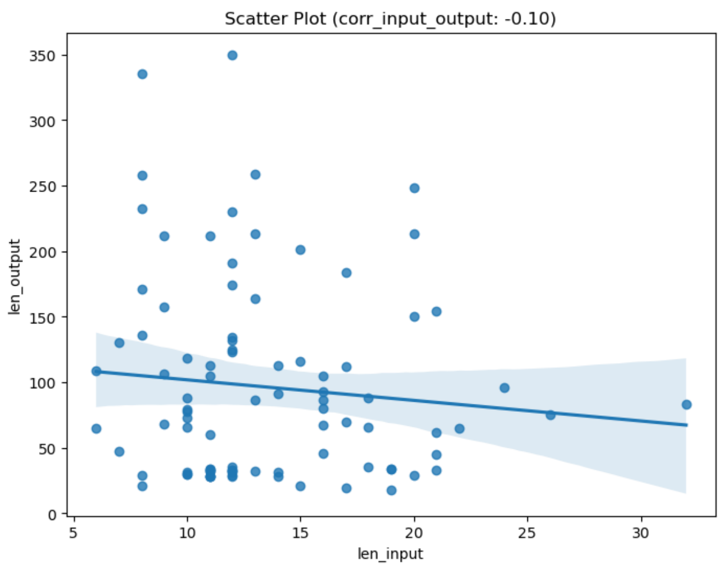


Figure 5.5 Correlation Input Token and Output Token

Figure 5.3 shows the correlation between the number of input tokens and response time, while Figure 5.5 presents the correlation between the number of input tokens and output, where no relationship is observed. However, Figure 5.4 reveals a correlation of 0.89 between the number of output token and response time. This is because when the output tokens are longer, the LLM takes more time to generate a response.

## 5.2 Experimental 2: Hypothesis Testing Result

### 5.2.1 Paired T-Test

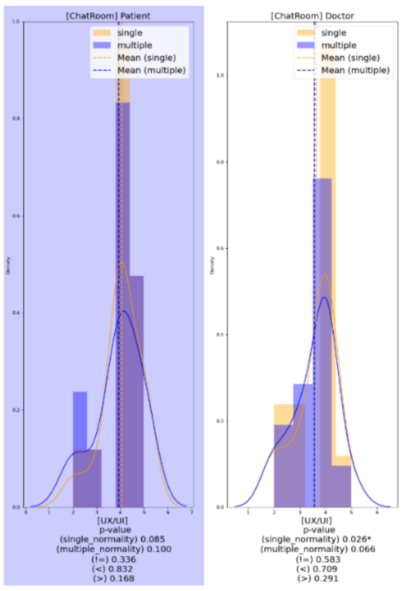


Figure 5.6 Paired t-test (UX/UI)

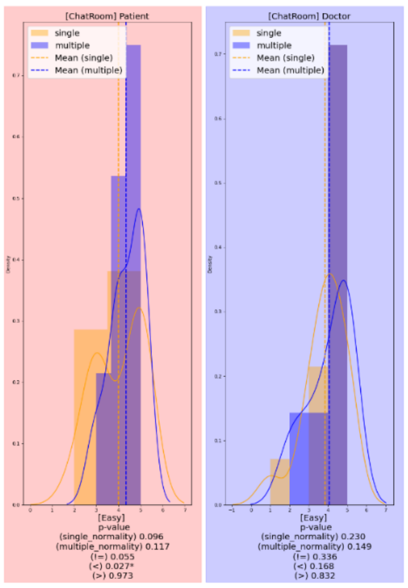


Figure 5.7 Paired t-test (Easy)

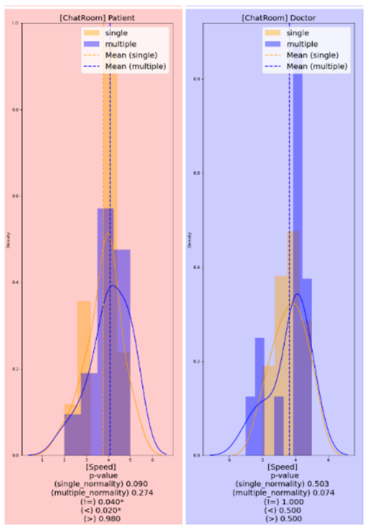


Figure 5.9 Paired t-test (Speed)

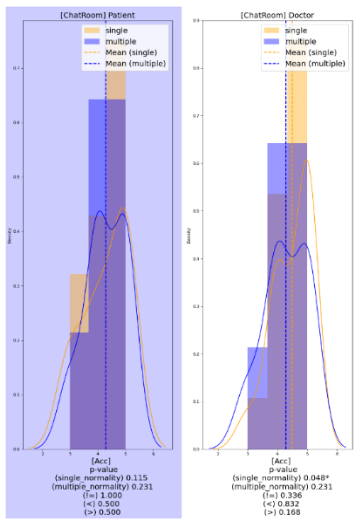


Figure 5.8 Paired t-test (Acc)

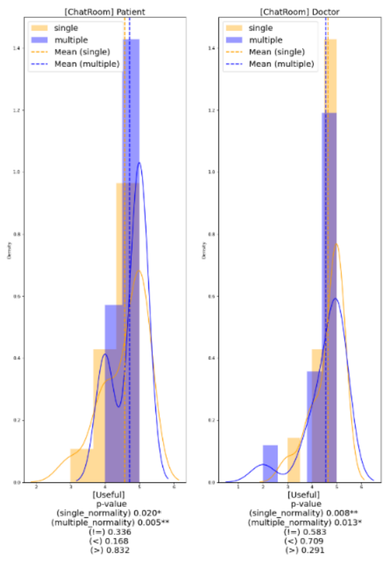


Figure 5.10 Paired t-test (Useful)

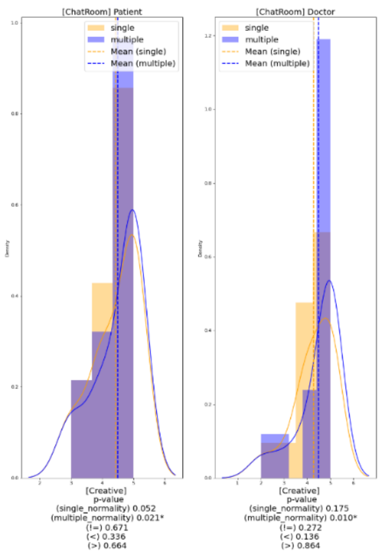


Figure 5.11 Paired t-test (Create)

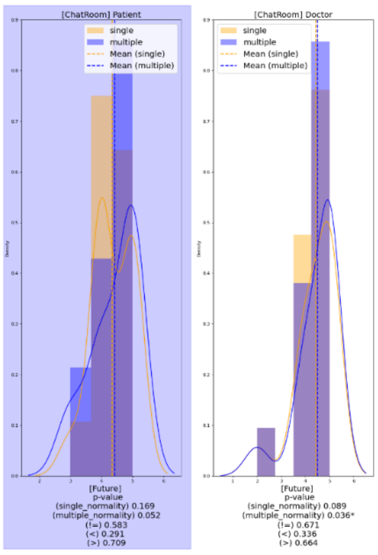


Figure 5.12 Paired t-test (Future)

Form Figure 5.7 to Figure 5.11, we conducted a Paired Samples T-Test to compare the mean values between the single-chat room and the group-chat room. The results show that, for the patient user group, the group-chat room is considered easier to use and faster in response compared to the single-chat room. The red-highlighted areas indicate these findings. The blue-highlighted areas represent cases where no significant difference in the mean values was found. However, the graphs that were not highlighted indicate that no conclusion could be drawn due to the data not passing the normality test using the Kolmogorov-Smirnov method.

### **5.2.1** Independent T-Test

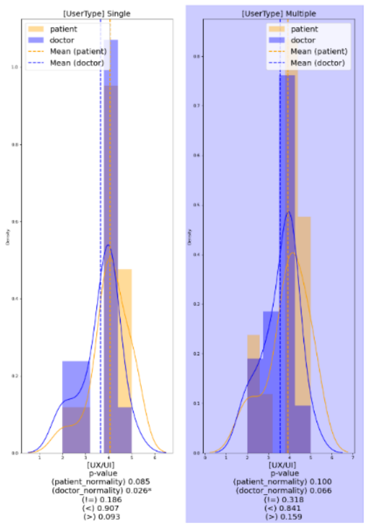


Figure 5.13 Independent t-test (UX/UI)

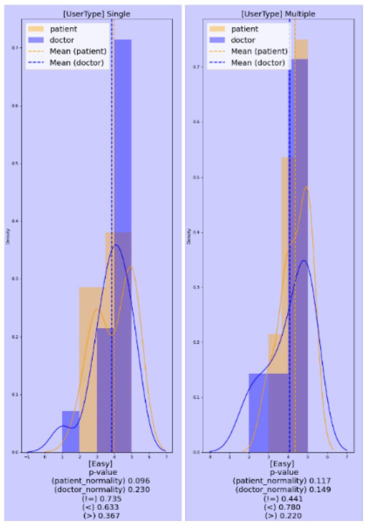


Figure 5.14 Independent t-test (Easy)

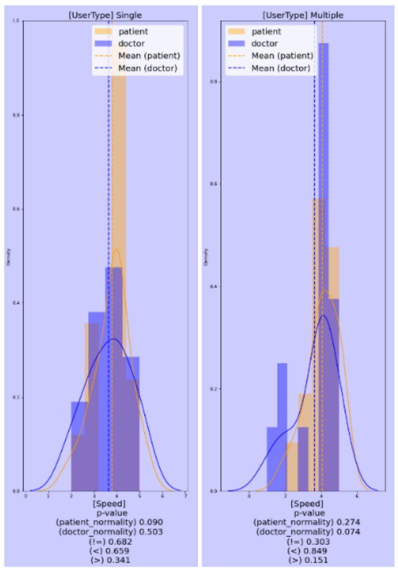


Figure 5.15 Independent t-test (Speed)

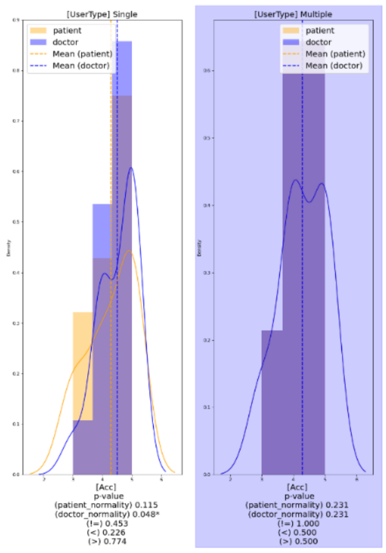


Figure 5.16 Independent t-test (Acc)

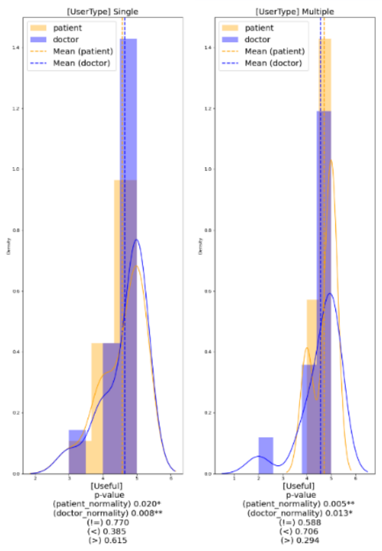


Figure 5.18 Independent t-test (Useful)

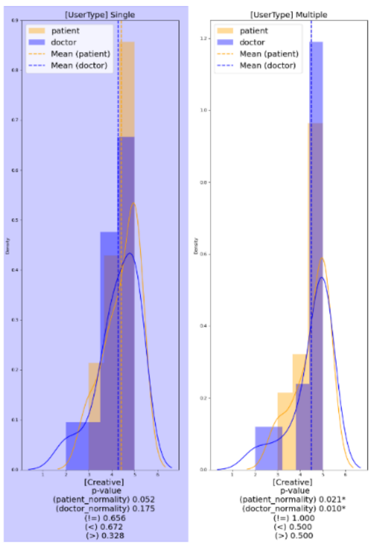


Figure 5.17 Independent t-test (Create)

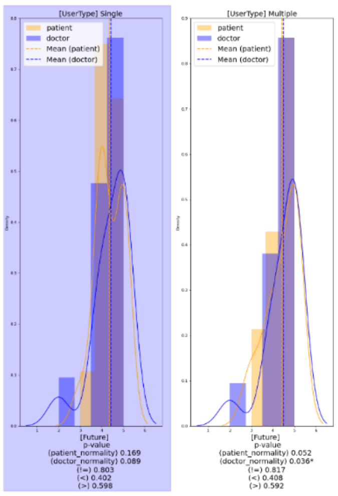


Figure 5.19 Independent t-test (Future)

From Figure 5.13 to Figure 5.19, we conducted an Independent Samples T-Test to compare the mean values between the patient and doctor user groups. The results indicate that, in most cases, there is no significant difference between the two groups.

## 5.3 Experimental 3: Feedback From Real Patient and Doctor

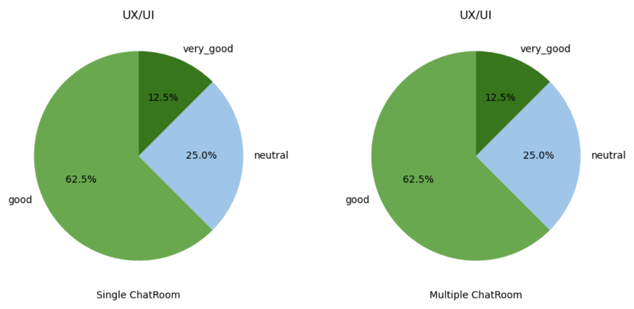


Figure 5.19 User Feedback (UX/UI)

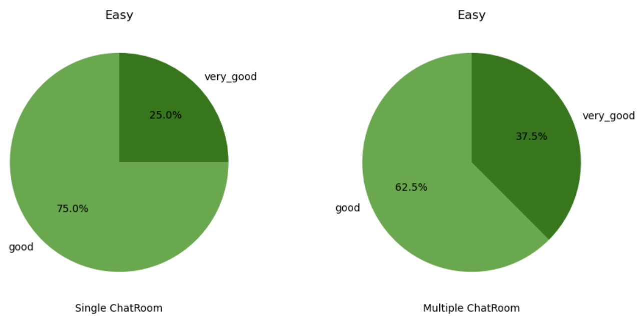


Figure 5.20 User Feedback (Easy)

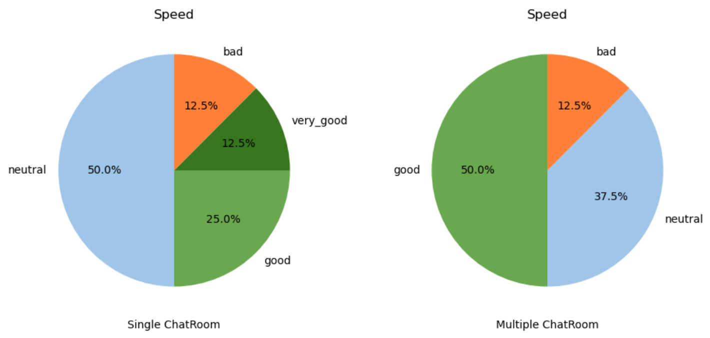


Figure 5.21 User Feedback (Speed)

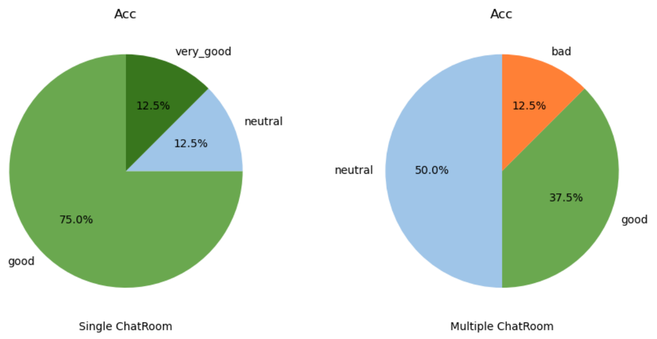


Figure 5.22 User Feedback (Acc)

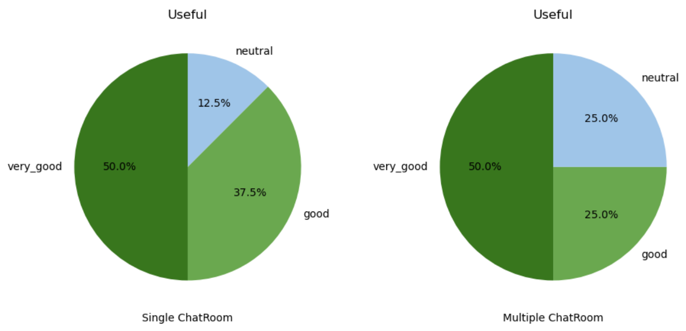


Figure 5.23 User Feedback (Useful)

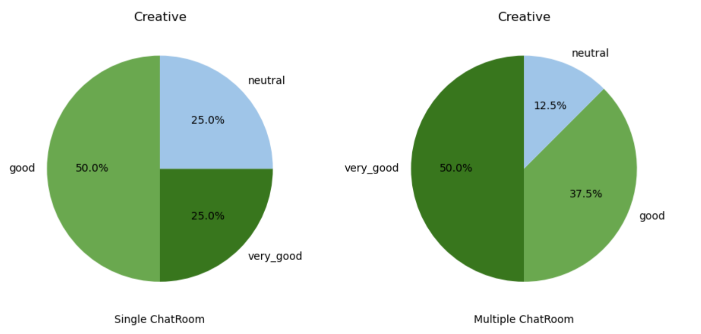


Figure 5.24 User Feedback (Create)

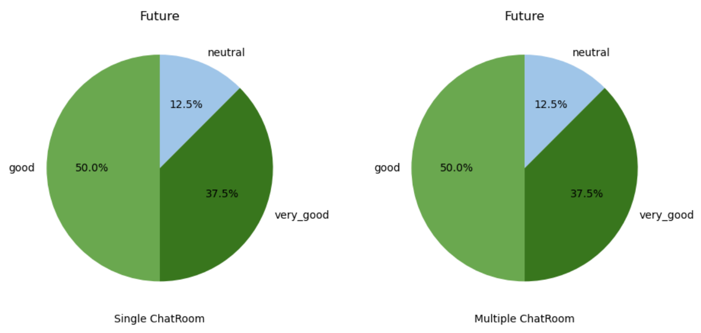


Figure 5.25User Feedback (Future)

From Figure 5.20 to Figure 5.26, the user review results revealed that most users rated the system as "Good" to "Very Good." However, there were only two term—speed and accuracy—where some users gave negative feedback. The following are some comments from users:

- “It’s good that it can respond even with misspellings. For doctors, if there were more drug information, it would be better. The response is slow.” - D1_single_chatroom
- “Still uses too much medical terminology. Some information is incorrect.” - D1_multiple_chatroom
- “There are parts where the answers are in medical terms that patients may not understand, such as Stage D.” - D3_multiple_chatroom

In terms of comparison between the single-chatroom and multiple-chatroom, it was observed that the multiple-chatroom was rated slightly better than the single-chatroom in terms of ease of use and creativity. The following are some comments from users:

- “Can inquire about initial symptoms.” - P1_multiple_chatroom
- “Innovative and easy to use. Very useful for heart failure patients.” - D2_multiple_chatroom

- “Good that it allows verification of the chatbot's answers.” - D3_multiple_chatroom

## 6 CONCLUSION

This project focuses on developing a chatbot for consulting heart disease patients, with doctors and nurses verifying the accuracy of responses. The document used as a reference for answering questions is Comprehensive Heart Failure Management Program[4]. The document is divided into chunks using various methods to build an Information Retrieval System.

In Experiment 1 From Table 3.1, along with Figure 5.1 and Figure 5.2, it can be observed that BM25 yields the best results compared to other methods, as it achieved the highest values for Precision@K, Recall@K, Reciprocal Rank@K, and F1@K across all values of K. Additionally, it also retrieved the largest number of documents. Regarding chatbot accuracy, Experiment 1 (Table 5.1) found that BM25+Filter performed best, achieving an Accuracy of 0.81. However, in Experiment 3 (Figure 5.23), users perceived that the chatbot in a multiple chatroom setting was slightly less accurate than in a single chatroom, which aligns with the results in Table 5.1, where Acc_QT2 was slightly lower than Acc_QT1. This suggests that the chatbot may perform worse when answering third-person reference questions compared to one-on-one questions.

For response speed, Experiment 1 (Table 5.2) showed that BM25 was the fastest method, with an average response time of 24 seconds. However, BM25+Filter was chosen for real-world use, prioritizing accuracy over speed. This aligns with Experiment 3 (Figure 5.22), where most users perceived the chatbot as slow.
Regarding the difference between single-chatroom and multiple-chatroom, Experiment 2 (Figure 5.8) indicated that users found multiple-chatroom significantly easier to use than single-chatroom, which was consistent with Experiment 3 (Figure 5.21).

For user satisfaction, Experiments 2 and 3 found that most users were satisfied with the chatbot, despite lower ratings on speed. Overall, users found both single-chatroom and multiple-chatroom chatbots beneficial and creative, and many expressed interest in using the chatbot again in the future.

However, in the future, a caching system and usage quota should be implemented to improve the chatbot’s response speed. In real-world usage, certain questions may be repeatedly asked. Retrieving answers from the cache instead of generating new responses is significantly faster and also helps reduce the number of API calls, ultimately lowering costs

## References

1. J. Lopatecki. "Kolmogorov Smirnov Test for AI: When and Where To Use It." https://arize.com/blog-course/kolmogorov-smirnov-test/ (accessed.
2. D. M. Jesussek. "t-Test." https://datatab.net/tutorial/t-test (accessed.
3. S. Sankar. "TF-IDF and BM25 for RAG— a complete guide." https://www.ai-bites.net/tf-idf-and-bm25-for-rag-a-complete-guide/ (accessed.

4. R. K. Arinthaya Phraminthikun, Onnga Omritkomon., Comprehensive Heart Failure Management Program. 2013.
5. P. Ramjee et al., "CataractBot: An LLM-Powered Expert-in-the-Loop Chatbot for Cataract Patients," p. arXiv:2402.04620doi: 10.48550/arXiv.2402.04620.
6. H. Kong, "Labeling supervised fine-tuning data with the scaling law," p. arXiv:2405.02817doi: 10.48550/arXiv.2405.02817.
7. A. Lozano, S. L. Fleming, C.-C. Chiang, and N. Shah, "Clinfo.ai: An Open-Source Retrieval-Augmented Large Language Model System for Answering Medical Questions using Scientific Literature," p. arXiv:2310.16146doi: 10.48550/arXiv.2310.16146.
8. A. Vaswani et al., "Attention is all you need," Advances in neural information processing systems, vol. 30, 2017.
9. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR, Las Vegas, NV, June 01, 2016 2016, p. 1, doi: 10.1109/cvpr.2016.90. [Online]. Available: https://ui.adsabs.harvard.edu/abs/2016cvpr.confE...1H
10. J. Lei Ba, J. R. Kiros, and G. E. Hinton, "Layer Normalization," p. arXiv:1607.06450doi: 10.48550/arXiv.1607.06450.
11. C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation Functions: Comparison of trends in Practice and Research for Deep Learning," p. arXiv:1811.03378doi: 10.48550/arXiv.1811.03378.
12. T. B. Brown et al., "Language Models are Few-Shot Learners," p. arXiv:2005.14165doi: 10.48550/arXiv.2005.14165.
13. L. Ouyang et al., "Training language models to follow instructions with human feedback," p. arXiv:2203.02155doi: 10.48550/arXiv.2203.02155.
14. e.-u. mrbullwinkle, kcpitt. "Chat Markup Language ChatML." https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/chat-markup-language (accessed.
15. T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," p. arXiv:1301.3781doi: 10.48550/arXiv.1301.3781.
16. Y. Gao et al., "Retrieval-Augmented Generation for Large Language Models: A Survey," p. arXiv:2312.10997doi: 10.48550/arXiv.2312.10997.
17. C. U. Press., Evaluation in information retrieval. 2009.