REMED: Retrieval-Augmented Medical Document Query Responding with Embedding Fine-Tuning

Tianqi Pang[1,2], Kehui Tan[1,2], Yujun Yao[2], Xiangyang Liu[1], Fanlong Meng[4], Chenyou Fan[1,2], Xiaofan Zhang[2,3,*]
[1]South China Normal University, [2]Shanghai AI lab, [3]Shanghai Jiao Tong University, [4]Bytedance Inc.
Email: {pangtianqi.scnu@gmail.com, xiaofan.zhang@sjtu.edu.cn}

***Abstract*—While advanced Large Language Models (LLMs) exhibit considerable promise, their tendency to generate unreliable information poses significant challenges, particularly in high-risk domains like healthcare. However, the advent of Retrieval-Augmented Generation (RAG) offers a novel solution tailored for the medical realm. This study further enhances retrieval accuracy by introducing REMED, a specialized medical document retrieval framework designed to address the hallucination problem prevalent in LLMs. The REMED framework integrates dataset construction, an efficient embedding fine-tuning EM-FT model, retrieval-augmented generation, and human evaluation of LLM responses. The EM-FT model can end-to-end fine-tune the medical sentence representations in large pre-trained models through an efficient embedding fine-tuning method, thereby enhancing the performance of medical retrieval. We adopt contrastive learning as the loss function to optimize the performance of the EM-FT model, enabling it to accurately capture the similarity between query and relevant documents. This approach not only improves the retrieval accuracy of positively related contents but also effectively reduces the matching with negatively related contents. Compared to direct dense vector retrieval, fine-tuning query and content vectors first and then performing dense retrieval tasks significantly improved the performance. Through validation on two datasets, we demonstrate that our EM-FT method improves recall and precision on MMD by 3.2%-6.0% and on MPD by 14.4%-42.6% compared to using the embedding model directly for retrieval. Furthermore, through human evaluation on the PULSE-7Bv5 model, we further confirm the effectiveness of our retrieval results in improving the quality of generated text.**

***Index Terms*—Medical Document Retrieval, Medical Dataset, Large Language Models, Contrastive Learning**

I. Introduction

Despite the rapidly growing capabilities of large-scale language models (LLMs) such as GPT-4 [1], there are concerns about their tendency to generate unreliable information, referred to as "hallucinations" [2]. In critical domains such as healthcare and law, where accuracy and reliability are critical, even minor errors can have severe consequences, necessitating extremely cautious information processing.

Retrieval-Augmented Generation (RAG) has recently emerged as an effective approach to address hallucinations [3], [4]. RAG combines pre-trained language models with retrieval systems, leveraging external databases to enhance performance. In our study, we utilized self-collected datasets, the Medical Menu Dataset (MMD) and the Medical Paper Dataset (MPD), to integrate relevant information and improve RAG's performance. By incorporating access to large and reliable knowledge bases during the generation process, we significantly improve the reliability of the generated results by fact-checking against trusted content in the knowledge bases.

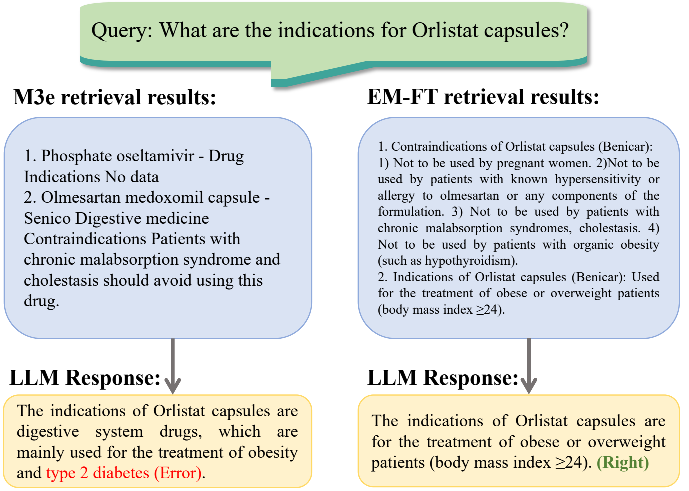
Query: What are the indications for Orlistat capsules?
M3e retrieval results:
1. Phosphate oseltamivir - Drug Indications No data
2. Olmesartan medoxomil capsule - Senico Digestive medicine Contraindications Patients with chronic malabsorption syndrome and cholestasis should avoid using this drug.
LLM Response:
The indications of Orlistat capsules are digestive system drugs, which are mainly used for the treatment of obesity and type 2 diabetes (Error).
EM-FT retrieval results:
1. Contraindications of Orlistat capsules (Benicar): 1) Not to be used by pregnant women. 2)Not to be used by patients with known hypersensitivity or allergy to olmesartan or any components of the formulation. 3) Not to be used by patients with chronic malabsorption syndromes, cholestasis. 4) Not to be used by patients with organic obesity (such as hypothyroidism).
2. Indications of Orlistat capsules (Benicar): Used for the treatment of obese or overweight patients (body mass index ≥24).
LLM Response:
The indications of Orlistat capsules are for the treatment of obese or overweight patients (body mass index ≥24). (Right)

Fig. 1: ***An example of retrieval-enhanced generation can be illustrated as follows.*** On the left side, we perform retrieval using only the embedding model and then utilize PULSE-7Bv5 to generate the answer. On the right side, we employ our method, EM-FT, for retrieval and then utilize PULSE-7Bv5 to generate the answer.

The dynamic nature of medical knowledge requires retrieval systems to integrate the in-depth expertise of domain specialists with real-time updated data to ensure the accuracy and timeliness of retrieval information. For example, pharmaceutical databases or medical literature databases can be used to obtain the latest drug instructions or research findings. In the case of drug instructions, the commercial names of drugs in practical applications usually vary based on the naming conventions of manufacturers, and people often tend to use abbreviated names when making inquiries. If only keyword-based retrieval or non-specialized medical knowledge retrieval systems are used, it may result in recommending incorrect drugs or inaccurate search results. Similarly, when retrieving medical literature, the model needs to understand complex medical concepts and research findings so that users can quickly and accurately locate relevant documents when asking questions about the content of articles.

To enhance the retrieval of drug information from our

[*]Corresponding Author.

databases, we employ self-collected datasets: Medical Menu Dataset (MMD) and Medical Paper Dataset (MPD). Domain experts have meticulously annotated MMD to include critical details like generic and brand names, usage, dosage, indications, contraindications, and drug interactions. Utilizing these annotated datasets, we fine-tune our embedding model to facilitate precise identification and retrieval of drug-related information. Specifically, our approach trains the embedding model in the retrieval stage using supervised datasets and incorporates contrastive learning methods to improve model performance at a low cost. In the field of dense retrieval, embedding models like M3e [5] and E5 [6] play a crucial role. Building upon these models, we introduce a new module called the Gate Linear Unit (GLU) module. As illustrated in Figure 2 on the right, EM-FT encompasses the entire Embedding model along with the GLU module. The EM-FT model is fine-tuned using a supervised dataset, where the embedding model is treated as a black box and its parameters are frozen, focusing the training process entirely on the GLU module. This approach ensures that medical data can be updated in real-time while preventing the leakage of private data.

Moreover, by incorporating contrastive learning into the fine-tuning of the EM-FT model, it becomes more effective in capturing the relevance between queries and documents. This allows the query embeddings to be closer to the positive text embeddings and farther from the negative text embeddings in the dense vector storage. As a result, compared to current vector retrieval methods, our approach leads to improvements in both accuracy and recall rates in the final retrieval outcomes. Additionally, our method offers advantages over full-parameter fine-tuning of M3e (M3e-FPFT) by saving time and resource costs while enhancing the overall accuracy of the retrieval model.

Figure 1 illustrates the comparison between the retrieval method proposed in this study and standard baseline methods on the MMD regarding their impact on the generation performance of LLMs. The results indicate that the baseline methods, due to providing inaccurate retrieval results, lead to erroneous answers generated by the LLMs. This finding emphasizes the criticality of optimizing retrieval accuracy in improving the generation quality of LLMs and reducing hallucination issues. Therefore, in summary, we make the following contributions:

1) **We have collected two large-scale supervised datasets, namely the MMD and the MPD.** MMD and MPD consist of 100 and 2,219 queries, respectively, sourced from doctors and LLMs. These datasets serve as two benchmark datasets for training and evaluating the medical document retrieval capability.
2) **We propose a medical document retrieval framework called REMED.** This framework consists of dataset construction, embedding model fine-tuning, retrieval-augmented generation, and human evaluation. EM-FT is an efficient embedding fine-tuning method that enables end-to-end fine-tuning of medical sentence representations in large pre-trained models, resulting in improved medical retrieval performance.
3) **We leverage LLM-Aided Query Generation approach for query generation.** We utilize LLMs to generate queries, as described in Section III-C, which are then used as user queries to construct supervised datasets. We subsequently use these supervised datasets to fine-tune the model and evaluate its performance. By utilizing LLMs itself to enhance the retrieval model's capability, we ultimately strengthen the query generation ability of LLMs.

II. Related Work

A. Retrieval-augmented Models

Retrieval-augmented generation (RAG) has become a key approach to improve the quality of language model generation. This technology can effectively integrate external knowledge sources into the model by retrieving relevant information and incorporating it into the model input or context, providing richer external knowledge support for the model, and enabling the model to make more accurate and comprehensive judgments in prediction and generation tasks. Recent studies have shown that by retrieving and enhancing the model's input with similar vocabulary or text fragments to the current task, its generalization ability and handling of unknown data can significantly improved [3], [4], [7], [8]. In a comparative study of knowledge injection in LLMs [9], RAG consistently outperformed unsupervised fine-tuning for both existing knowledge encountered during training and entirely new knowledge. More broadly, retrieval-augmented models have been explored in various modes and tasks. In text generation, retrieval has provided prototypes or examples to improve dialogue, translation, and summarization systems [10]–[12].

Currently, hotspots in the RAG field are mainly focused on improving retrieval accuracy, optimizing generation quality, exploring multimodal knowledge fusion methods, and leveraging self-supervised learning to enhance model performance [13]–[15]. However, while retrieval-augmented techniques have made significant progress in multiple areas, issues such as data scarcity and domain adaptation, explainability and controllability, as well as resource efficiency and inference costs still need further study. Our research is dedicated to exploring lighter-weight retrieval-augmented mechanisms aimed at combining the powerful generation capabilities of large language models (LLMs) to achieve higher efficiency and scalability in model performance improvement.

B. Large Language Models (LLMs)

Large language models (LLMs) like GPT-4 [1] and Claude [16] has significantly improved the quality of text generation, benefiting natural language processing (NLP). However, the performance of LLMs is still limited in specific vertical domains due to the lack of specialized knowledge. To address this issue, researchers have adopted methods such as fine-tuning [17], [18], retrieval augmentation [4], post-pretraining [19] and prompt optimization [20]–[22] to improve

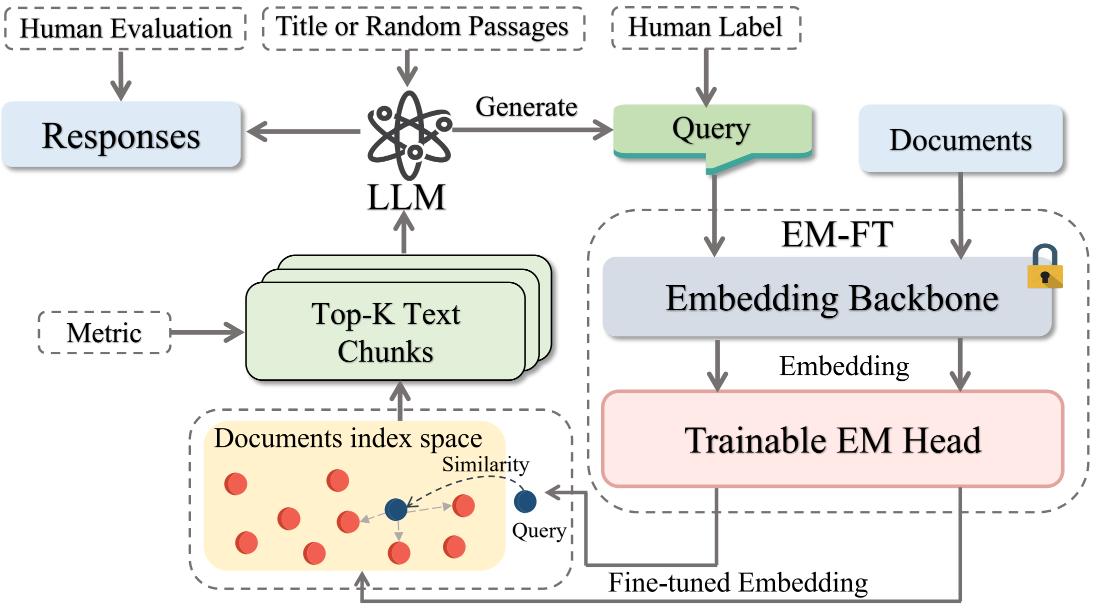

Fig. 2: **REMED framework.** The REMED framework integrates dataset construction (Section III-A III-B), an efficient embedding fine-tuning EM-FT model (Section IV-D), retrieval-augmented generation, and human evaluation of LLM responses (Section V-D).

the output of LLMs in vertical domains and make them more professional and precise.

Meanwhile, some researchers [23]–[26] have attempted to use LLMs to assist data annotation by automatically generating a large amount of annotated data with their text generation capabilities, enriching datasets and improving the effect of model training. This method reduces annotation costs and provides new ways to solve data scarcity problems.

In our research, by leveraging LLMs' ability to deeply analyze documents and craft retrieval queries, along with integrating their generative skills with our enhanced retrieval approach, we achieved more precise and relevant answers, improving question answering systems' performance and laying the groundwork for LLMs' use in more sophisticated tasks.

III. Supervised Medical Document Datasets

In this section, we specifically introduced two self-collected supervised medical document datasets: the Chinese Medical Document dataset (MMD) and the English Medical Document dataset (MPD). We provided detailed explanations of the methodology for generating queries using the LLM-Aided Query Generation approach.

A. *Medical Menu dataset (MMD)*

MMD aims to be a comprehensive and reliable benchmark for evaluating medical information retrieval systems. We source data from the authoritative "WHO Medicine"[*] and cover all drug information in the "National Pharmacopoeia", comprising over 200,000 records. This provides solid data support for key fields such as medical research and drug development. MMD takes into account factors such as positive screening, negative screening, and noisy data to comprehensively evaluate system performance. As depicted in Figure 3, manual annotations were carried out for each medical question to assess the relevance of drug names. These drug names were categorized into two primary directions: medication instructions and light medical inquiries. A total of 1,276 reference recall data points were annotated, spanning 100 distinct medical questions. In our experiments, the training set comprises data from the initial 70 questions, totaling 573 instances, while the remaining 30 questions constitute the test set, with a total of 205 instances.

B. *Medical Paper dataset (MPD)*

We construct a sampled medical document dataset by sampling 1,000 papers from the well-known National Center for Biotechnology Information (NCBI)[*] in the United States. We abbreviate this dataset as Medical Paper dataset (MPD). As shown in Figure 4, we pre-process and clean the MPD to ensure the accuracy and reliability of our analysis. At first, we performed a series of filtering operations to exclude literature that did not meet our research criteria, such as informal conference speeches and non-peer-reviewed reports. Additionally, to focus on the analysis of textual content, we removed tabular data and non-standard mathematical formulas from the papers.

Furthermore, for the cleaned documents, we implemented a series of processing steps to adapt them to the requirements

[*] https://whoyx.com/service

[*] https://www.ncbi.nlm.nih.gov/

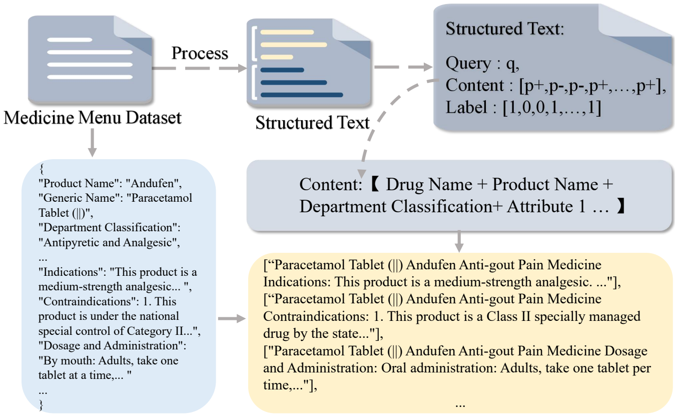
Process
Medicine Menu Dataset
Structured Text
Structured Text:
Query : q,
Content : [p+,p-,p-,p+,…,p+],
Label : [1,0,0,1,…,1]
Content:【 Drug Name + Product Name + Department Classification+ Attribute 1 … 】
{
"Product Name": "Andufen",
"Generic Name": "Paracetamol Tablet (||)",
"Department Classification": "Antipyretic and Analgesic",
...
"Indications": "This product is a medium-strength analgesic... ",
"Contraindications": 1. This product is under the national special control of Category II...",
"Dosage and Administration": "By mouth: Adults, take one tablet at a time,... "
...
}
[“Paracetamol Tablet (||) Andufen Anti-gout Pain Medicine Indications: This product is a medium-strength analgesic. ..."],
[“Paracetamol Tablet (||) Andufen Anti-gout Pain Medicine Contraindications: 1. This product is a Class II specially managed drug by the state..."],
["Paracetamol Tablet (||) Andufen Anti-gout Pain Medicine Dosage and Administration: Oral administration: Adults, take one tablet per time,..."],
...

Fig. 3: ***The structured details of MMD.*** This figure illustrates the process of transforming the original medical menu dataset into structured text. The key focus is on the content, which is amalgamated with drug names, product names, and department classifications for each data point, and the attributes vary accordingly.

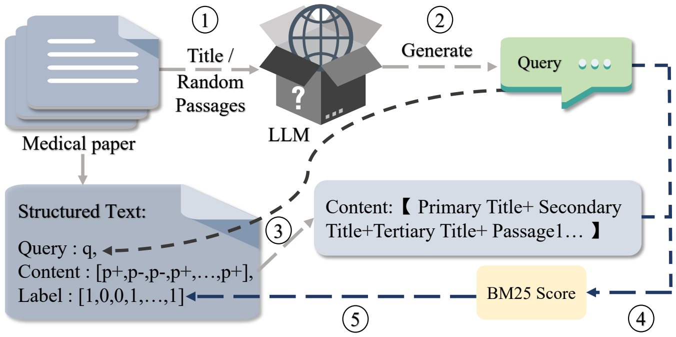

Fig. 4: ***The structured details of MPD.*** (1) Using the paper's title or random passages as input for the LLM. (2) The LLM generates relevant queries based on the paper's title or passages. (3)Structured text. (4) Computing the BM25 scores for each query and passage. (5) Using the BM25 scores to assign labels accordingly.

of the embedding model. Considering the length limitation for model input, we employed a text segmentation strategy to divide longer texts into fixed-length segments (with a maximum sequence length of 768). This approach aims to ensure that each data segment can be effectively processed by the model while preserving sufficient contextual information. To enhance the information density of the data, we appended key information, including main and subheadings, at the beginning of each data entry to improve the model's understanding and retrieval capabilities. In the end, the processed dataset consisted of a total of 886 papers with 79,966 data entries. In our experiments, we divided the MPD into training and test sets, with the training set accounting for 70% of the data and using a random state of 42. The MPD, used for generating queries based on paper titles, was split into a training set consisting of 392 instances (approximately 60,000 entries) and a test set consisting of 169 instances (approximately 20,000 entries). Similarly, the MPD used for generating queries based on random paragraphs was divided into a training set with 658 instances (approximately 100,000 entries) and a test set with 282 instances (approximately 45,000 entries).

C. LLM-Aided Query Generation

To overcome the limitations of the MPD containing only document data, we adopted a novel LLM-Aided Query Generation approach to construct a supervised dataset. This approach leverages LLM (such as GPT-3.5) to automatically generate user queries based on medical literature titles. This step intends to simulate the query intent of potential users in the real world, thereby creating a more realistic training environment. However, queries generated solely based on titles may not capture the richness of the article content, affecting the quality of the dataset and the performance of the retrieval system.

To address this limitation, we further propose a query generation strategy based on the article content. By analyzing randomly selected passages from the literature, LLM is able to generate queries that reflect the details of the literature. This approach provides advantages in terms of topic coverage, accuracy, and diversity of generated queries, while promoting the adaptability and accuracy of the retrieval system in handling complex and diverse information needs. Moreover, this query generation strategy serves as an effective means for evaluating the performance of the retrieval system, as high-quality simulated queries can be generated in the absence of real user inputs. By integrating LLM-Aided Query Generation approach into the training process, we not only reduce human involvement in the dataset construction phase but also improve the quality and efficiency of the dataset and model training.

Furthermore, queries generated by LLM are also used to evaluate the performance of the retrieval model. By comparing the literature returned by the model with the expected retrieval results, we can quantify the model's ability to understand and match user query intent. In this process, the generation capability and retrieval capability of LLM are seen as mutually reinforcing. On one hand, the generated queries help the retrieval model understand and adapt to diverse user needs. On the other hand, the fine-tuned retrieval model improves the ability of LLM to generate more accurate and relevant answers.

In our experiments, we utilized GPT-3.5 as our LLM. For paper titles, we instructed the LLM to generate three questions for each title, and for random passages, we extracted five from each paper, asking the LLM to create five queries based on these passages. Some of the generated questions had issues, such as incompleteness or errors. Consequently, we filtered out these problematic questions, resulting in a final dataset of 210 papers. Next, we compute the BM25 score [27] for every query and passage to create a similarity ranking metric. Afterward, we categorize the label as 1 if it exceeds the average score, and as 0 if it equals or falls below the average score.

IV. Scenarios and Approaches

In this section, we introduce our proposed EM-FT model. Specifically, we describe the embedding model, gated linear unit (GLU) network architecture, and key formulas used. We also outline the experimental setup and training procedure.

A. Embedding Backbone

Referring to the MTEB Leaderboard [28], we have selected two baseline embedding backbone models: the m3e-base (M3e) [5] for exceptional performance on Chinese datasets and the e5-base-v2 (E5) [6] for its outstanding performance on English datasets.

M3e, short for "Moka Massive Mixed Embedding", is an advanced natural language processing model developed and released by MokaAI. This model has been meticulously trained using the UniEM framework and rigorously evaluated against the MTEB-zh benchmark [28]. "Massive" signifies its extensive training data, comprising over 22 million Chinese sentence pairs, enabling proficiency across a wide range of language understanding tasks.

E5 is a state-of-the-art family of text embedding models celebrated for their exceptional adaptability across diverse tasks. E5 serves as a universal embedding model, effortlessly integrating into a wide spectrum of tasks that rely on single-vector text representations. This versatility extends to tasks like retrieval, clustering, and classification, and E5 consistently excels in both zero-shot and fine-tuned scenarios.

The design of our Embedding backbone model emphasizes a high degree of modularity, allowing for easy replacement based on specific requirements without impacting subsequent phases of model training. This design approach significantly enhances the flexibility and adaptability of the framework.

B. Loss Function

We design a contrast loss as supervision to optimize the embedding space so that documents relevant to the query are closer than irrelevant documents as shown in Eq. 1 and Eq. 2:

$$L(W) = L(q, p_1^+, p_2^+, ..., p_n^+, p_1^-, p_2^-, ..., p_m^-), \tag{1}$$

$$L(W) = -log \frac{\sum_{i=1}^{n} e^{(sim(q,p_i^+))}}{\sum_{i=1}^{n} e^{(sim(q,p_i^+))} + \sum_{j=1}^{m} e^{(sim(q,p_j^-))}}, \tag{2}$$

where $L(W)$ represents maximizing the relevance probability of positive passages and minimizing the relevance probability of negative passages with respect to the query q, by training the model parameters W, and q represents the input query. p_i^+ are positive passages relevant to the query. p_j^- are negative passages irrelevant to the question. We employ cosine similarity $sim(q,p) = Cos(E(q), E(p))$ as the scoring function to measure the match between query q and passage p.

To mitigate model overfitting, we augment the L2 term in the contrast loss. As depicted in Eq. 3, the final loss, denoted as L, is the sum of $L(W)$ and $L2$.

$$L = L(W) + \lambda \sum_{i=1}^{n} w_i^2, \tag{3}$$

where, w_i represents the model's parameters, and λ is a scaling factor for regularization.

C. Activation Function

In our methods, we primarily employ two activation functions: GELU [29] and Swish [30]. The distinctions between the two methods are illustrated in Eq. 4 and Eq. 5:

$$\text{GELU}(x) = x \cdot \Phi(x) = x \cdot \frac{1}{2}[1 + \text{erf}(x/\sqrt{2})], \tag{4}$$

$$\text{Swish}(x) = x \cdot \sigma(\beta x), \tag{5}$$

where $f(x)$ is a linear transformation of input x. In Swish function, β is a constant or a trainable parameter, we make β equlal to 1, i.e., it becomes the Sigmoid Linear Unit (SiLU) activation function.

D. EM-FT Model

The EM-FT model architecture integrates two core components, the Embedding Backbone and the Trainable EM Head, aiming to achieve efficient text similarity retrieval. As shown in Figure 5, our compact Trainable EM Head design contains three main components: Layer Normalization (LayerNorm), two linear layers (LinearLayer) with an activation function in between. Drawing inspiration from recent advancements in LLM design [31], we offer two choices for the activation function: GELU and Swish. Both functions are formulated to introduce nonlinearity while maintaining smooth gradient flow during the backpropagation process, as demonstrated in Eq. 4 and Eq. 5 respectively.

The choice of using GLU is motivated not only by its computational efficiency but also by its ability to effectively preserve the complexity and fine-grained information of the original text. Additionally, GLU helps maintain the density of embedding vectors and handles the dimensions of text data. This concise and efficient network structure enables us to achieve enhanced accuracy in similarity retrieval during supervised fine-tuning, all while ensuring the richness of the text content is preserved.

In the entire process of EM-FT, the query and target texts are first encoded through the embedding model, which transforms natural language text into high-dimensional dense vector representations to reveal their underlying semantic features. Subsequently, the vector representations processed by the embedding model are fed into the Trainable EM Head, which further optimizes and adjusts the vector space to better capture the relevance between the query and target content. The EM-FT optimization process is based on the loss function Eq. 2, where the objective of these loss functions is to adjust the distances between positive and negative samples in the multidimensional space, enabling the model to differentiate between relevant and irrelevant texts more accurately.

Through this approach, the EM-FT model continuously optimizes its parameters through iterative training to achieve more accurate text retrieval. The process emphasizes not only parameter optimization but also the model's ability to generalize when handling complex queries. Ultimately, the

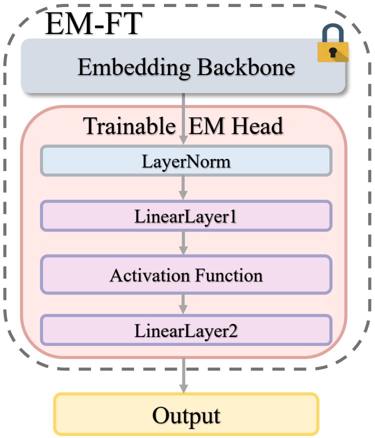

Fig. 5: ***The structured details of EM-FT model.*** EM-FT contains an embedding backbone and trainable EM head.

goal of the EM-FT model is to strike a balance where the model can improve retrieval accuracy and efficiency while maintaining the richness of the text content.

Compared to full fine-tuning of the embedding model, the EM-FT model provides a more efficient strategy. This strategy allows for incremental updates to the model, adjusting only the parts affected by newly added data. This approach significantly reduces computational resources and shortens the model update time. By locally adjusting parameters instead of retraining the entire model with each database update, the EM-FT model can adapt to new information more agilely, ensuring real-time and accurate retrieval in the search system.

E. Evaluation metrics

In our experiments, we assess the retrieval results using four metrics:

$$Recall = \frac{TP}{TP + FN}, \quad Precision = \frac{TP}{TP + FP}, \qquad (6)$$

where TP (True Positive) counts the correctly predicted positive examples, FN (False Negative) counts the positive examples incorrectly predicted as negative. Similarly,FP (False Positive) tallies the negative examples incorrectly predicted as positive, and TN (True Negative) represents the correctly predicted negative examples.

In information retrieval, Average Precision (AP) serves as an evaluation metric for assessing the quality of ranking in response to a single query. AP calculates the average precision values of the top-K results for a query. Each precision value is weighted by the relevance of the result. The mean of the APs across all queries is known as the mean Average Precision (mAP).

$$mAP = \frac{\sum_{i=1}^{K} AP_i}{K}, \quad Hit\_rate = \frac{\sum_{i=1}^{N} hit(i)}{N}, \qquad (7)$$

in our experiments, $hit(i)$ is present when at least one positive passage has been retrieved within the Top-K results, in which case $hit(i) = 1$; otherwise, $hit(i)$ is set to 0. $\sum_{i=1}^{N} hit(i)$ is the sum of $hit(i)$, and the N is the sum of the query.

F. Training details

The model was trained using the SGD optimizer with initial learning rate of 0.01 which decays by a factor of 0.95 for every epoch. The training process encompassed 50 epochs, with each epoch representing one complete iteration through the training data. To mitigate overfitting and enhance generalization, a regularization coefficient of 1e-6 was applied.

V. Experiment

In this section, we present the state-of-the-art (SOTA) baseline we utilized, share the results obtained using our method, conduct a detailed analysis of these results, and perform additional ablation studies to support our approach.

A. Baselines

Retrieval with LLM. Following the recent state-of-the-art work REPLUG [4], we propose the baseline based on the LM-Supervised Retrieval method, which we call **LSR**. The LSR directly prompts the LLM to score the similarity between query and documents, such as:

> *PROMP_TEXT*="Here are a query and a list of documents. Please provide the similarity score between these two sentences. The first is [QUERY] and the other is [Doc1, Doc2, ..., DocN]. Score the query with each of the document in range from 0 to 10 in which 10 means the most similar while 0 means the most different.

Embedding-based baselines. For "M3e-base" method, we directly use the pre-trained M3e embedding model to embed queries and contents, and then employ a FAISS index to enable medical document retrieval. For "M3e-FPFT" (M3e Full-Parameter Fine-Tuning), we utilize our MMD to fine-tune the M3e pre-trained model, then employ the fine-tuned model for retrieval. E5-base-v2 (Title) and E5-base-v2 (RP) also use E5 directly for embedding without fine-tuning.

B. Results and analysis

In Table I, we present the performance of the MMD using various evaluation metrics with a focus on the Top-K=10 recommendations. The evaluation metrics include Recall, Precision, Hit Rate, and mAP (mean Average Precision), as shown in Section IV-E.

- ***Incorporating L2 regularization in the loss function significantly improves model performance,*** with recall increasing by around 6%. This validates that regularization helps avoid overfitting and enhances generalization capability.
- ***Using the EM-FT yields superior performance compared to EM-FT(s).*** The design of swiglu in EM-FT(s) may result in excessively intricate embedding vectors, leading to overfitting. EM-FT, on the other hand, yields improved vector representations, thereby enhancing retrieval quality.

TABLE I: Performance evaluation of different methods on MMD (Top-$K = 10$).

	Recall	Precision	Hit-rate	mAP
LSR [4]	0.387	0.300	**1.000**	0.492
M3e-base	0.503	0.548	0.862	0.513
M3e-FPFT	0.391	0.292	**1.00**	0.366
EM-FT(s) (w/o L2)	0.524	0.532	0.897	**0.589**
EM-FT(s) (w/ L2)	0.533	0.54	0.931	0.570
EM-FT (w/o L2)	0.554	0.560	0.897	0.564
EM-FT (w/ L2)	**0.563**	**0.580**	0.897	0.587

- ***Our model performs better than M3e-FPFT.*** M3e-FPFT incurs a tenfold rise in time cost, while recall and precision decrease by 17.2% and 28.8% respectively, compared to EM-FT (W/L2). This indicates our model is more computationally efficient.

In Table II, we have included the performance metrics for MPD using different evaluation criteria. “Title” refers to the mean LLM-generated queries based on the paper title, while “RP” represents the mean LLM-generated queries based on random passages from the paper.

TABLE II: Performance evaluation of different methods on MPD (Top-$K = 10$).

	Recall	Precision	Hit-rate	mAP
LSR [4]	0.390	0.250	0.750	0.350
E5-base-v2 (Title)	0.212	0.526	0.994	0.541
E5-base-v2 (RP)	0.229	0.562	0.954	0.600
EM-FT(s) (Title)	0.300	0.767	1.00	0.763
EM-FT(s) (RP)	0.342	0.849	1.00	0.913
EM-FT (Title)	**0.356**	**0.952**	1.00	**0.967**
EM-FT (RP)	0.334	0.831	1.00	0.881

- ***Both EM-FT and EM-FT(s) result in noticeable gains over the baseline.*** Specifically, EM-FT (Title) boosts recall, precision and mAP substantially, by 14.4%, 42.6% and 42.6% respectively.
- ***Using LLM-Aided Query Generation approach for query generation proves beneficial.*** In order to confirm whether the performance improvements are attributed to the method of query generation, we employed two distinct approaches: Title and RP. We observed that the model’s performance improved under both approaches, thus validating the generality of our method.

C. Ablation studies

In this section, we conducted a comparison to assess the impact of utilizing different labels for training on the model’s performance. To keep it concise, our analysis primarily concentrated on the evaluation results for the medical menu dataset with a Top-K value of 10, as presented in Table III. Here’s a simplified explanation of the various labels employed:

GT label: These labels were assigned manually, primarily based on drug names, to determine whether a query should be included or not.

GT $\vee$ M3e label: This label combines both the GT labels and the selected labels from “M3e label”. It retains the original labels while incorporating the labels chosen based on the additional criteria.

GT $\wedge$ mAP label: This label differs from “M3e label” in terms of how Top-K is defined. In “mAP label”, Top-K is defined as the data points with scores higher than the mAP threshold.

GT $\wedge$ M3e label: These labels were derived from the GT labels with additional criteria. If a data point meets both of the following conditions: a) Its similarity score is among the Top-K (where Top-K is defined as half the content length), and b) The original label is also 1, then the label for that data point is set as 1; otherwise, it is set as 0.

- ***“GT $\wedge$ M3e label” and “GT $\wedge$ mAP label” methods perform better.*** Compared to M3e-base, “GT $\wedge$ M3e label” and “GT $\wedge$ mAP label” methods show improvements of approximately 4.8%-9.6% and 3.4%-10.9%, respectively, in terms of recall and mAP.
- ***The performance is worse when using the GT label.*** Using the “GT label” for fine-tuning results in even worse performance than M3e-base, with a decrease of approximately 1.2% in recall and 0.3% in mAP, respectively. Furthermore, the performance of “GT $\vee$ M3e label” is also worse compared to using “GT $\wedge$ M3e label” or “GT $\wedge$ mAP label” individually.

D. Human Evaluations

As illustrated in Figure 1, we select the Top-2 retrieval results from the knowledge base as prefixes. The query is concatenated with each prefix in the format “prefix + query” as input to the PULSE-7Bv5 [32], a clinical language model (CLM). Since the quality of retrieval model’s results directly impacts the quality of outputs from the LLM, evaluation of the final responses allows us to indirectly measure whether the retrieval model has an enhancing effect. We randomly selected $t(t = 20)$ test sample pairs (M3e, EM-FT (w/L2)). Each answer is scored as “$+1$” if EM-FT (w/L2) generates better, “0” if EM-FT (w/L2) has the equal effectiveness as M3e, and “-1” if EM-FT (w/L2) is even less effective than M3e.

We gathered evaluations from 5 individuals, including both specialized healthcare team members and generalists, with scores of 0.5, 0.3, 0.1, 0.4, and 0.4. The final result is 0.34 with P-value 0.0041. So we reject the null hypothesis that “M3e gets better answers than EM-FT” with 1% significance. This proves that our optimized retrieval model (EM-FT) does provide better answers.

VI. Conclusion

In this study, we propose a medical document retrieval framework called REMED. The framework includes data set

TABLE III: Performance evaluation of different methods.(Top-K=10)

	Recall	Precision	Hit-rate	mAP
M3e	0.379 (+0.0%)	0.592 (+0.0%)	0.931 (+0.0%)	0.551 (+0.0%)
GT label	0.368 (-1.2%)	0.568 (-2.4%)	0.966 (+3.5%)	0.548 (-0.3%)
GT $\vee$ M3e label	0.395 (+1.6%)	0.596 (+0.4%)	0.966 (+3.5%)	0.638 (+8.8%)
GT $\wedge$ mAP label	0.413 (+3.4%)	0.632 (+4.0%)	0.931 (+0.0%)	0.660 (+10.9%)
GT $\wedge$ M3e label	0.428 (+4.8%)	0.632 (+4.0%)	0.931 (+0.0%)	0.646 (+9.6%)

construction, embedding model fine-tuning, retrieval enhancement generation, and human evaluation. The EM-FT is an efficient embedding fine-tuning method that enables end-to-end fine-tuning of medical sentence representations in large pre-trained models to improve medical retrieval performance. Additionally, we collected two supervised data sets, MMD and MPD, for fine-tuning the EM-FT model. In summary, our REMED framework for medical document retrieval ensures the privacy and security of the data sets. We have demonstrated that by improving the quality of the retrieval model's Top-K results, we can enhance the accuracy of LLM responses. However, our work has some limitations, such as the binary labeling system (0 and 1). In future efforts, refining the labeling to include a range from 0 to 4 would be a promising direction, potentially leading to more precise results.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (Projects 62106156 and 62301311), and the Shanghai Artificial Intelligence Laboratory.

References

[1] OpenAI, "Gpt-4 technical report," *ArXiv*, vol. abs/2303.08774, 2023.

[2] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin *et al.*, "A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions," *arXiv preprint arXiv:2311.05232*, 2023.

[3] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020.

[4] W. Shi, S. Min, M. Yasunaga, M. Seo, R. James, M. Lewis, L. Zettlemoyer, and W.-t. Yih, "Replug: Retrieval-augmented black-box language models," *arXiv preprint arXiv:2301.12652*, 2023.

[5] Y. Wang, Q. Sun, and S. He, "M3e: Moka massive mixed embedding model," 2023.

[6] L. Wang, N. Yang, X. Huang, B. Jiao, L. Yang, D. Jiang, R. Majumder, and F. Wei, "Text embeddings by weakly-supervised contrastive pre-training," *arXiv preprint arXiv:2212.03533*, 2022.

[7] O. Ram, Y. Levine, I. Dalmedigos, D. Muhlgay, A. Shashua, K. Leyton-Brown, and Y. Shoham, "In-context retrieval-augmented language models," *arXiv preprint arXiv:2302.00083*, 2023.

[8] U. Khandelwal, O. Levy, D. Jurafsky, L. Zettlemoyer, and M. Lewis, "Generalization through memorization: Nearest neighbor language models," *arXiv preprint arXiv:1911.00172*, 2019.

[9] O. Ovadia, M. Brief, M. Mishaeli, and O. Elisha, "Fine-tuning or retrieval? comparing knowledge injection in llms," *arXiv preprint arXiv:2312.05934*, 2023.

[10] J. Weston, E. Dinan, and A. H. Miller, "Retrieve and refine: Improved sequence generation models for dialogue," *arXiv preprint arXiv:1808.04776*, 2018.

[11] J. Gu, Y. Wang, K. Cho, and V. O. Li, "Search engine guided neural machine translation," in *AAAI*, vol. 32, no. 1, 2018.

[12] H. Peng, A. P. Parikh, M. Faruqui, B. Dhingra, and D. Das, "Text generation with exemplar-based adaptive decoding," *arXiv preprint arXiv:1904.04428*, 2019.

[13] A. Asai, Z. Wu, Y. Wang, A. Sil, and H. Hajishirzi, "Self-RAG: Learning to retrieve, generate, and critique through self-reflection," *arXiv preprint arXiv:2310.11511*, 2023. [Online]. Available: https://arxiv.org/abs/2310.11511

[14] S. Xue, C. Jiang, W. Shi, F. Cheng, K. Chen, H. Yang, Z. Zhang, J. He, H. Zhang, G. Wei *et al.*, "Db-gpt: Empowering database interactions with private large language models," *arXiv preprint arXiv:2312.17449*, 2023.

[15] W. Lin, J. Chen, J. Mei, A. Coca, and B. Byrne, "Fine-grained late-interaction multi-modal retrieval for retrieval augmented visual question answering," *arXiv preprint arXiv:2309.17133*, 2023.

[16] "Introducing 100k context windows," https://www.anthropic.com/index/100k-context-windows, 2023.

[17] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.

[18] X. Liu, K. Ji, Y. Fu, W. L. Tam, Z. Du, Z. Yang, and J. Tang, "P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks," *arXiv preprint arXiv:2110.07602*, 2021.

[19] C. Wu, Y. Gan, Y. Ge, Z. Lu, J. Wang, Y. Feng, P. Luo, and Y. Shan, "Llama pro: Progressive llama with block expansion," *arXiv preprint arXiv:2401.02415*, 2024.

[20] M. Besta, N. Blach, A. Kubicek, R. Gerstenberger, L. Gianinazzi, J. Gajda, T. Lehmann, M. Podstawski, H. Niewiadomski, P. Nyczyk *et al.*, "Graph of thoughts: Solving elaborate problems with large language models," *arXiv preprint arXiv:2308.09687*, 2023.

[21] S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. Narasimhan, "Tree of thoughts: Deliberate problem solving with large language models," *arXiv preprint arXiv:2305.10601*, 2023.

[22] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 24 824–24 837, 2022.

[23] Z. Chen, H. Mao, H. Wen, H. Han, W. Jin, H. Zhang, H. Liu, and J. Tang, "Label-free node classification on graphs with large language models (llms)," *arXiv preprint arXiv:2310.04668*, 2023.

[24] "Autolabel," https://github.com/refuel-ai/autolabel, 2023.

[25] Y. Ma, H. Jiang, and C. Fan, "Sci-cot: Leveraging large language models for enhanced knowledge distillation in small models for scientific qa," *arXiv preprint arXiv:2308.04679*, 2023.

[26] X. Liu, T. Pang, and C. Fan, "Federated prompting and chain-of-thought reasoning for improving llms answering," in *International Conference on Knowledge Science, Engineering and Management*, 2023.

[27] S. Robertson and H. Zaragoza, "The probabilistic relevance framework: Bm25 and beyond," *Found. Trends Inf. Retr.*, 2009.

[28] "Mteb leaderboard," https://huggingface.co/spaces/mteb/leaderboard, 2023.

[29] D. Hendrycks and K. Gimpel, "Gaussian error linear units (gelus)," *arXiv preprint arXiv:1606.08415*, 2016.

[30] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions," *arXiv preprint arXiv:1710.05941*, 2017.

[31] A. Zeng, X. Liu, Z. Du, Z. Wang, H. Lai, M. Ding, Z. Yang, Y. Xu, W. Zheng, X. Xia *et al.*, "Glm-130b: An open bilingual pre-trained model," *arXiv preprint arXiv:2210.02414*, 2022.

[32] I. Team, "Internlm: A multilingual language model with progressively enhanced capabilities," https://github.com/InternLM/InternLM, 2023.