Medical Dataset Expansion

Reza Dalir

Modern Information Retrieval Course Final Project, fall 2024

Project Description

Medical question summarization is an important task in medical information retrieval that aims to integrate patient questions for easier understanding by AI models and medical professionals. This project is related to the summarization of medical questions from the **MEQSUM** dataset and the improvement of data quality using Round-Trip Translation (RTT). We use several selection techniques, which include Frechét Question Distance (FQD), Precision Recall Question Distance (PRQD), and Question Semantic Volume (QSV), to choose the dataset. We use pre-trained summarization models like BART, T5, and **deepseek**, where performance is measured using ROUGE, BLEU, and METEOR metrics. It is aimed at enhancing summarization precision and question-answering optimization in medical practice.

Dataset Overview

The MEQSUM dataset is a specialized dataset designed for medical question summarization, aimed at improving the clarity and conciseness of patient queries in healthcare systems. It consists of medical questions written by patients and their corresponding summarized versions, making it an essential resource for developing and evaluating summarization models.

The dataset includes a diverse range of health-related questions, covering topics such as symptoms, diseases, medications, diagnoses, and treatment options.

Each entry consists of original questions which is a patient-generated query, often verbose and sometimes containing irrelevant details and also reference summary: a professionally curated or algorithmically generated summarized version of the question.

Project Phases

1- Load and Inspect the Dataset

In this step I have loaded the dataset using pandas library. To get more familiar with the dataset I printed some information about the dataset. This dataset contains 1000 rows and 3 columns. First column is file which contains the file address which is not important for our purpose, second and third columns are CHQ and Summary that contain question and the summary of the question respectively. Using isnull() function I noticed that there is no null or empty cell in the dataset. Also, I noticed that all the rows are unique.

2- Preprocessing

I used beautiful soup for removing HTML tags and use regular expression for remove all the other things. If there is SUBJECT and MESSAGE in the CHQ column I remove the SUBJECT and I only use the MESSAGE.

3- Handling Missing or Irregular Data

As we see in first step there is no missing values among the cells of the dataset, so, we don't need to do anything special for handling missing values. During the preprocessing stage, we observed that the length of questions and summaries varied significantly across the dataset. Some questions were excessively long, while others were very short. To ensure uniform input size for our summarization models, we applied truncation and padding techniques to both the original questions and their summarized versions. We used quantile function to find the first and last tenth of the data distribution and then we pad first tenth of the shortest questions and truncated the last tenth of the longest questions. We skip this step for summary column because firstly the distribution of the summary cells are almost normal and fine and secondly it reduces our accuracy in the final evaluation of the project and it's not the purpose of this project to do this step for the summary.

4- Translate Questions to a Pivot Language

To improve question summarization and introduce linguistic diversity, we implemented Round-Trip Translation (RTT) using a pivot language. This technique involves translating a question into an intermediate language and then back into the original language, which helps rephrase, simplify, or clarify the text. We used Spanish, French, German, Chinese and Italian languages for this step. To do this translation I have done two different methods:

a. MarianMTModel
b. Google Translate API

In Marian model I have defined a translate function and used the method for translating the questions into the destination languages. But the main method I used during this project was google translate API. It is faster and more accurate than the Marin model. Results are shown in the main code file of this project.

5- Translate Back to English

This step is almost like the previous step and I used google translate to translate back the translated questions into English. This completes the round-trip-translation and makes 5000 new questions, so we can extend the dataset using these questions. These two recent steps took too much time since we are using an online API or a huge model for translation. (Around 4 hours to complete the execution using GPU).

6- Use FQD to select a subset of the new dataset

After generating multiple variations of each question using Round-Trip Translation (RTT), we needed a method to select the most informative and diverse subset for training our summarization model. We employed Frechét Question Distance, a semantic similarity metric designed to measure how much a transformed question differs from the original while still preserving its meaning.

We calculated the FQD score for each translated question by comparing it with the original English version using an embedding-based distance measure. Instead of using all RTT-generated variations, we set a threshold to filter out translations that were too similar or too different from the original question. This selection step helped retain informative rephrasing while eliminating redundant or inaccurate translations, leading to a more efficient and effective training dataset for medical question summarization. This step was straight forward and just needed to pay attention to the main and normalized formula of FQD. Finally, I kept questions with normalized FQD score between 0.02 to 0.2 and removed other questions. Around 800 questions for each language remained and finally around 4000 new questions added to the list of questions.

7- Use PRQD to select a subset of the new dataset

This step is almost like the previous step, goal is the same but method is a bit different. Precision-Recall Question Distance is another selection method used to filter and refine the newly augmented dataset. Unlike FQD, which focuses on overall question similarity, PRQD evaluates the quality of generated questions by measuring their relevance to the original questions using precision and recall scores.

In this step, each generated question is compared to the original medical question, ensuring that it maintains key semantic components while avoiding unnecessary distortions. A high precision score ensures that the generated question retains important elements from the original, while a good recall score ensures it captures a sufficient amount of the original question's meaning. By applying PRQD, we effectively eliminate low-quality or misleading paraphrases, retaining only those that are both accurate and diverse. This helps maintain the integrity of the dataset and improves the effectiveness of the summarization model.

8- Use QSV to select a subset of the new dataset

Question Semantic Volume measures the diversity of the dataset by analyzing how much new semantic information the generated questions add. Using sentence embeddings, we compute a covariance matrix and estimate the dataset's semantic volume via its determinant. A higher value indicates greater question diversity. By applying QSV, we select a subset of questions that maximize variation while avoiding redundancy, ensuring a richer and more diverse dataset. During the execution of this method, I ran into a problem of underflowing which is happened because of the strong similarity between the generated questions and the original ones. To overcome this issue, I multiplied the covariance score to 100. This made the scores almost between 0 to 1. So, after the normalization step, I choose questions with normalized QSV score between 0.05 and 0.35 which adds almost around 4000 new questions to the dataset.

9- Use pre-trained models to summarize questions

This is one of the most important steps of this project, now that we have chosen some new questions, we need to summarize them and add them to the dataset. After cleaning the dataset with FQD, PRQD, and QSV, the next goal is to create clear summaries that preserve the most vital information and omit irrelevant details. The chosen pre-trained models, including BART, T5, and GPT-based models, are utilized to create these summaries.

It starts by placing the chosen questions into the summarization models, making sure the format is what the models expect. All the models receive the input and generate a summary of the question without losing the essential medical meaning. The generated summaries are then checked to make sure they are readable and provide useful information. The summarized questions are returned to the dataset, improving the overall quality of training data.

Some problems occurred during this step. One of the main problems was that some models would often create summaries that were too general or incomplete, without including specific medical details. In order to solve this issue, we had to fine-tune prompts and tweak settings like max length and beam search to improve the quality of the output. Handling various forms of languages was another challenge since some models struggled with specific sentence structures, particularly with questions that had undergone repeated back-and-forth translation. Computing cost was also high, as operating numerous summarization models on a large dataset required extensive processing power. I usually ran out of RAM or GPU and the runtime session crashes, so I had to start again at the beginning. To make these starts over faster I saved each step result into a file so I could read them instantly from the file and go to the next step rapidly. Some models were slow in providing responses, leading to delays in the summarization process. Enhancing the response speed using GPU acceleration solved some of these delays.

In spite of these challenges, the pre-trained models generated useful summaries that enhanced the dataset. Through choosing the best summaries, the project guaranteed the final dataset had brief, readable, and relevant medical questions, which increased its value for future application in accessing medical information. Result of the summarization between BART and T5 model were extremely poor in the contrast to DeepSeek LLM. But DeepSeek model uses too much graphical computing units to load and summarize a question.

In the last step we will compare some results between different language models and different languages.

10- Use evaluation metrics and compare the results

After generating summaries using pre-trained models, it is essential to evaluate their quality and effectiveness. This step involves using standard metrics such as ROUGE, BLEU, and METEOR to compare the generated summaries with reference summaries. These metrics help assess how well the models preserve key medical information while producing concise and readable outputs. Since summarization is not just about word overlap but also about meaning retention, careful analysis of the results is necessary to determine which model performs best for medical question summarization.

Since we used 3 different methods for selecting the questions, I evaluate each one of them separately. In evaluations, I calculate Rouge-1, Rouge-2, Rouge-L, bleu and meteor metrices for each one of the languages and show the result in a separate table.

Initially, I employed BART and T5 models for the summarization task. After executing these models and evaluating them using FQD selection method, I obtained the following results for gold questions using BART:

	Average Score
rouge-1	0.210705
rouge-2	0.071545
rouge-L	0.185902
bleu	6.231903
meteor	0.156663

Results for RTT-based summaries:

	ROUGE-1	ROUGE-2	ROUGE-L	bleu	meteor
chineese	0.120529	0.038993	0.107006	2.849950	0.084898
french	0.120785	0.037360	0.107497	2.885483	0.085473
german	0.175546	0.049065	0.155837	3.831917	0.119602
italian	0.127325	0.040392	0.111915	3.055146	0.092092
spanish	0.121179	0.039369	0.106658	2.783649	0.087510

While these results are not entirely poor, they also fall short of being optimal. To enhance the performance, I decided to experiment with another model, DeepSeek, a large language model with 7 billion parameters, utilizing its chat version for evaluation. The results for DeepSeek under the FQD selection method for gold questions are as below:

	rouge-1	rouge-2	rouge-L	bleu	meteor
0	0.545455	0.444444	0.545455	13.006502	0.284091
1	0.444444	0.000000	0.444444	9.930284	0.087719
2	0.300000	0.111111	0.300000	10.600313	0.258137
3	0.538462	0.083333	0.461538	3.030756	0.163399
4	0.250000	0.181818	0.250000	2.908318	0.342377
...	...	...	...	...	...
995	0.000000	0.000000	0.000000	0.000000	0.000000
996	0.400000	0.222222	0.400000	6.766165	0.220307
997	0.480000	0.260870	0.480000	16.188614	0.486772
998	0.545455	0.200000	0.272727	4.266505	0.433145
999	0.692308	0.333333	0.692308	21.464786	0.484375

	Average Score
rouge-1	0.295107
rouge-2	0.111745
rouge-L	0.259083
bleu	7.060116
meteor	0.173031

The better performance between these methods are obvious and Deepseek method is doing much better than other ones.

Here are the results for the RTT task using FQD:

	ROUGE-1	ROUGE-2	ROUGE-L	bleu	meteor
chineese	0.252149	0.088497	0.221494	5.719521	0.142994
french	0.258126	0.092993	0.227049	6.487052	0.150425
german	0.295847	0.107279	0.257683	6.979391	0.173467
italian	0.258471	0.090343	0.226120	6.105964	0.149357
spanish	0.246438	0.082258	0.215460	5.521652	0.142717

	Average Score
ROUGE-1	0.262206
ROUGE-2	0.092274
ROUGE-L	0.229561
bleu	6.162716
meteor	0.151792

At first glance, it is evident that different pivot languages yield comparable scores, though German outperformed others consistently across all metrics. These results suggest that the Round-Trip Translation (RTT) method, combined with FQD-based selection, successfully expanded the dataset while preserving meaningful question variations.

Original Summary vs. DeepSeek-Generated Summary (German RTT):

Here is a comparison between the original summary and the summary generated by the Deepseek model for summarization task in German RTT:

```
created summary:      Can I get genetic test for MM? Where? How much cost?
the original summary: Where can I get genetic testing for multiple myeloma, and what is the cost?
```

```
created summary:      Where can I order Nulytely, manufacturer, phone number?
the original summary: Who makes nulytely, and where can I buy it?
```

```
created summary:      Where can I get my daughter tested for Williams Syndrome in my area?
the original summary: Where can I get genetic testing for william's syndrome?
```

We can see Despite variations in word choice, the semantic meaning is preserved, and the summary remains concise and effective. This demonstrates the model's strong generalization ability in handling paraphrased medical questions.

Additional Evaluation: QSV and PRQD Methods

To further assess the impact of different selection techniques, I applied Question Semantic Volume (QSV) and Precision-Recall Question Distance (PRQD) methods. The results for PRQD method with Deepseek in RTT task are presented below:

	ROUGE-1	ROUGE-2	ROUGE-L	bleu	meteor
chineese	0.250239	0.085318	0.221812	5.838075	0.144736
french	0.237092	0.078468	0.207524	5.654251	0.138687
german	0.279595	0.097910	0.244278	6.280108	0.158930
italian	0.238510	0.078810	0.205584	5.467073	0.132074
spanish	0.233923	0.075426	0.203687	5.396213	0.135073

	Average Score
ROUGE-1	0.247872
ROUGE-2	0.083186
ROUGE-L	0.216577
bleu	5.727144
meteor	0.141900

And The results for QSV method with Deepseek in RTT task are presented below:

	ROUGE-1	ROUGE-2	ROUGE-L	bleu	meteor
chineese	0.243706	0.080095	0.211939	4.394625	0.156385
french	0.230054	0.069549	0.200502	4.340036	0.142792
german	0.278045	0.095346	0.243765	5.327750	0.179086
italian	0.242372	0.077016	0.209199	4.686008	0.151896
spanish	0.233010	0.077947	0.205836	4.426099	0.153246

	Average Score
ROUGE-1	0.245437
ROUGE-2	0.079991
ROUGE-L	0.214248
bleu	4.634904
meteor	0.156681

As we can see The FQD method consistently achieves the highest scores across most metrics, while PRQD and QSV perform similarly, with slight variations in different languages. Also German achieves the best scores across all three methods, suggesting that it serves as the most effective pivot language for RTT-based summarization. PRQD and QSV have very close performance, but PRQD slightly outperforms QSV in BLEU and METEOR scores, whereas QSV has a slightly higher ROUGE-1 score.

So, we use FQD method with deepseek and RTT task to expand the dataset. It adds around 4000 new questions and their corresponding summaries to the dataset, these questions are selected so they are neither too close not too far from the original questions in the dataset.

Conclusion

In this project, we aimed to enhance the quality and diversity of the MEQSUM dataset for medical question summarization. By implementing Round-Trip Translation (RTT) and leveraging selection techniques such as FQD, PRQD, and QSV, we generated an extended dataset that maintains both semantic integrity and variation. Our selection criteria ensured that the newly introduced questions were not only linguistically diverse but also preserved the essential medical meaning of the original patient queries.

The preprocessing phase involved cleaning the dataset, handling inconsistencies, and ensuring uniform input lengths. RTT was applied using multiple languages, with German emerging as the most effective pivot language. The generated dataset was then refined using FQD, PRQD, and QSV to filter out redundant or irrelevant questions. Among these, FQD provided the best balance between diversity and meaning retention, making it the most effective approach for dataset expansion.

To summarize the expanded dataset, we employed pre-trained models such as BART, T5, and DeepSeek. While BART and T5 provided baseline performances, DeepSeek significantly outperformed them in summarization accuracy, though at the cost of higher computational demands. The generated summaries were evaluated using standard metrics (ROUGE, BLEU, and METEOR), confirming that the extended dataset maintained high-quality medical question summaries.

Overall, this project successfully expanded the MEQSUM dataset by approximately 4000 new question-summary pairs, improving its utility for medical information retrieval. The combination of RTT, FQD-based selection, and DeepSeek summarization offers a robust approach for dataset augmentation in the medical domain. Future work could explore further fine-tuning of summarization models or alternative linguistic techniques to improve both efficiency and accuracy.