CISC839 Project-20: A QUESTION ENTAILMENT APPROACH TO QUESTION ANSWERING

Nourhan Abdelkerim[1], Sondos Mahmoud[2], and Mostafa Awad[3]

[1]**21nmma1@queensu.ca**
[2]**21smam1@queensu.ca**
[3]**21mmaa1@queensu.ca**

1 BACKGROUND AND OBJECTIVE

One of the challenges in large-scale information retrieval (IR) is to develop domain-specific methods to answer natural language questions.That is when users tries to search online for an answer to their question they struggle to navigate through many links and web pages to find a complete sufficient answer.

Despite the availability of numerous sources and datasets for answer retrieval, Question Answering (QA) remains a challenging problem due to the difficulty of the question understanding and answer extraction tasks when tackling domain-specific searches. This is due to several factors, such as the lexical and semantic challenges of domain-specific data that often include advanced argumentation and complex contextual information, the higher sparseness of relevant information sources, and the more pronounced lack of similarities between users' searches.

The objective of the project is to help in building a system to answer questions based on Recognizing Question Entailment (RQE). This system retrieve answers to a premise question by retrieving entailed questions that already have associated answers. Therefore, the entailment relation between two questions is defined as: question A entails question B if every answer to B is also a correct answer to A.

Our main task is to provide solutions to improve the performance of the Question Answering task. The first thing that we will do is to build a model to measure the entailment score between two questions, this will improve the performance by retrieving questions with the highest similarity to the proposed one. The second contribution is a model that can predict the type of the question, using the type of the question can improve the performance by comparing the proposed question only with questions with the same type, this will reduce the run time and make the Question Answering system faster and provide better results.

To accomplish this task we will use two datasets the first one is MedQuAD dataset and the second one is dataset retrieved from the evaluation of LiveQAMed2017-TestQuestions Dataset.

1.1 Hypothesis Test

Is the ratio of treatment questions in the dataset significantly different from this ratio in the test dataset? When we want to test the question answering system, to achieve more accurate results so we need to know if the distribution of question types in train data will cover all users need by checking test data.

1.2 Regression question

What is the entailment score between two question? This will help question answering system to decide which question is similar to the proposed one and retrieve its answer.

1.3 predictive question

Can we predict any characteristics of the user question like question type? this can help the question answering system to provide a better answer which is related to the type of the question by filtering the dataset so that the model find the entailment within smaller size of dataset.

2 DATASET

2.1 MedQuAD Dataset

2.1.1 Train Data

The first dataset is MedQuAD. MedQuAD includes 47,457 medical question-answer pairs created from 12 NIH (National Institutes of Health) websites (e.g. cancer.gov, niddk.nih.gov, GARD, MedlinePlus Health Topics). The collection covers 37 question types (e.g. Treatment, Diagnosis, Side Effects) associated with diseases, drugs and other medical entities such as tests.

Data was crawled from websites from the National Institutes of Health7. Each web page describes a specific topic (e.g. name of a disease or a drug), and often includes synonyms of the main topic that we extracted during the crawl.

Hand-crafted patterns were constructed for each website to automatically generate the question-answer pairs based on the document structure and the section titles. Each question was annotated with the associated focus (topic of the web page) as well as the question type identified with the designed patterns.

To provide additional information about the questions that could be used for diverse IR and NLP tasks, the questions were automatically annotated with the focus, its UMLS Concept Unique Identifier (CUI) and Semantic Type. Two methods were combined to recognize named entities from the titles of the crawled articles and their associated UMLS CUIs: (i) exact string matching to the UMLS Metathesaurus8, and (ii) MetaMap Lite9. Then the UMLS Semantic Network was used to retrieve the associated semantic types and groups.

2.1.2 Test Data

Testset will be generated from the same dataset (MedQuAD Dataset), we may split the dataset into train and test and use the trainset to learn the model and the testset for evaluation. or we can use LiveQAMed2017-TestQuestions dataset as test set.

2.2 LiveQAMed2017-TestQuestions Dataset

2.2.1 Train Data

The second dataset is extraced from the manual evaluation of the results for the testset (TREC-2017-LiveQA-Medical-Test).

To create this dataset, the test questions of the medical task at TREC-2017 LiveQA were passed to a model that can measure the entailment degree between two questions (questions from this testset and questions from MedQuAD Dataset) then retrieve the answers of the top 10 candidates. Then an interface to perform the manual evaluation of the retrieved answers was developed.

The evaluation was done by 3 assessors: a medical doctor, a medical librarian and a researcher in medical informatics. They gave a score for each answer that varies from 1 to 4 where Correct and Complete Answer takes rank 4, Correct but Incomplete takes rank 3, Incorrect but Related takes rank 2, Incorrect takes rank 1. Then these scores are used to tell the entailment degree between the questions.

Medical questions at TREC-2017 LiveQA are randomly selected from the consumer health questions that the NLM receives daily from all over the world. The test questions cover different medical entities and have a wide list of question types such as Comparison, Diagnosis, Ingredient, Side effects and Tapering.

2.2.2 Test Data

To generate a testset LiveQAMed2017-TestQuestions Dataset will be splitted into two parts: the first 50% (for example) of the file (question 1 to 52) (row 1 to 1296) to train a learning model that can predict their judgement score: 1-Incorrect

2-Related

3-Incomplete

4-Excellent

and the remained 50% of the file (question 53 to 104) (row 1297 to 2479) as a testset. We may use different splitting percentages depending on the behaviour of the model.

2.2.3 Extra dataset

Our data is very small, so we maybe gain a low accuracy. In this case we may use another dataset. The other dataset contains only two labels (Entail or Not-Entail). In this case our problem will not be regression, it will be classification.

3 BASIC DATA EXPLORATION

3.1 General view of MedQuAD-dataset.

The important features are the following :

1-Qtype-Fine contains question type (e.g Causes , treatment ,...).

2-Questions contains different questions.

3-Qtype-Coarse is a new feature, contain type of question if it is 'Drugs', 'Tests' or 'Disease'.

```
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Answer        16359 non-null  object
 1   Qtype-Fine    46856 non-null  object
 2   Focus         46842 non-null  object
 3   Question      46856 non-null  object
 4   Qtype-Coarse  46856 non-null  object
```

Figure 1. Information of dataset.

3.2 General view of pair-question dataset

we will split this dataset into two parts :

1-Train part contains 70 percent of the dataset.

2-Test set contains 30 percent of the dataset.

The important features are the following :

1-Question.

2-normal Ques.

3-Judge score.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2479 entries, 0 to 2478
Data columns (total 4 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   ID           2479 non-null   int64
 1   Question_1   2479 non-null   object
 2   normal_Ques  2479 non-null   object
 3   Judge_score  2479 non-null   object
dtypes: int64(1), object(3)
memory usage: 77.6+ KB
```

Figure 2. Information of pair-question dataset.

3.2.1 Rank for Judge score

Return a Series containing counts of unique values. The resulting object will be in descending order so that the first element is the most frequently-occurring element.

```
1-Incorrect     1339
2-Related        630
3-Incomplete     215
4-Excellent      132
Name: Judge_score, dtype: int64
```

Figure 3. Related rank for two questions with Judge score.

3.3 Statistics summary for MedQuAD categorical data

There are no missing values for the MedQuAD dataset

	Question	qtype
count	47441	47441
unique	44603	39
top	What causes Causes of Diabetes ?	information
freq	20	9214

Figure 4. Statistics summary.

3.4 Statistics summary for pair questions categorical data

There are no missing values for pair questions dataset.

	Question_1	normal_Ques	Judge_score
count	2479	2479	2479
unique	1805	103	4
top	What is (are) Shingles ?	My question to you is: what is the reason that...	1-Incorrect
freq	15	45	1436

Figure 5. Statistics summary.

3.5 Quality of data

Data quality is the measure of how fit a data set is to serve its specific purpose and how reliable it is to make trusted decisions. With too many missing values it is going to be difficult to go through the correct analysis to answer specific questions.

Dataset	MedQuAD	pair questions
	561	163

Table 1. Number of duplicated.

4 DATA ANALYSIS PIPELINE

4.1 Data Exploration

4.1.1 MedQuAD Data

As we knew, the first file of dataset(MedQuAD) includes 47,457 medical question-answer pairs but there are 3 subsets have no Answers, they are removed to respect the MedlinePlus copyright. The remaining data only is 16,407 rows, but we don't need to use answers in our classification task, so there is no effect. Data contains lots of scientific questions and it will make classification task easier, but if the user adds unscientific question it will not be accurate in classification.

4.1.2 QA-TestSet-LiveQA-Med-Qrels-2479-Answers Data

Our data that will be used to measure entailment is unbalanced data.

4.2 Data Quality

4.2.1 MedQuAD Data

The data contains a lot of special characters, so we will apply cleaning and preprocessing functions. There are 2838 duplicates.

4.2.2 QA-TestSet-LiveQA-Med-Qrels-2479-Answers Data

Data contains a lot of special characters. Also it contains two questions paired, one of them is medical question and other written by people doesn't have a medical background, so it will be challenging to find entailment with them. Secondly we will solve unbalanced problem by using resample function or SMOTE function.

4.3 Feature engineering

At the first we will extract the features from XML file then apply feature selection to select the best feature and remove useless ones. Then apply some preprocessing techniques to handle text data like questions and answers by using tokenization, Lemmatization, embedding, resample data, remove stop words and special characters to prepare our data for training. Here we need to create new feature, we'll add 'Qtype-Coarse' feature, it will contain type of question if it is 'Drugs', 'Tests' or 'Disease'.

4.4 Potential Models

Our goal is a study of machine learning and deep learning approaches of RQE, so we will train our data through machine learning and deep learning models, then compare between the performance of each and apply the one that has best performance.

4.4.1 MedQuAD Data

1-Logistic Regression. 2-Bidirectional Encoder Representations from Transformers (BERT).

4.4.2 QA-TestSet-LiveQA-Med-Qrels-2479-Answers Data

1-Linear Regression. 2-Bidirectional Encoder Representations from Transformers (BERT).

4.5 Framework and tools

We will use TensorFlow and Scikit-learn frameworks.

4.6 Model Evaluation

4.6.1 MedQuAD Data

Our model will predict the type of each question we have in our dataset, we will evaluate it by comparing the result with the main type in the dataset to compute the Accuracy, Recall, Precision, F1 score and the confusion matrix.

4.6.2 QA-TestSet-LiveQA-Med-Qrels-2479-Answers Data

Our QE system is comparing the the questions from QA-TestSet-LiveQA-Med-Qrels-2479-Answers with questions from TREC-2017-LiveQA-Medical-Test then predict the rank of entailment between them, we will evaluate it by comparing the rank from our model with the main rank to calculate the Accuracy, Recall, Precision, F1 score and the confusion matrix.

4.7 Potential Challenges

4.7.1 MedQuAD Data

Some potential challenges we may face in deploying our model in production is that the model doesn't cover all the topics so there may be some questions by the users that the model can't provide a reliable type to them. The data that will be classified contains scientific questions and the users need to write their question in a scientific way using scientific terms to find accurate type.

4.7.2 QA-TestSet-LiveQA-Med-Qrels-2479-Answers Data

The data contains questions in unscientific way and other in scientific way, the challenge will be how to find entailment between them. But the benefit now in this part, the users don't need to write their question in a scientific way using scientific terms.

5 DISTRIBUTION OF WORKLOAD

5.1 MedQuAD Data

5.1.1 First Member will work on:

Data Cleaning and Feature Engineering.

5.1.2 Second Member will work on:
Build model and Data training.

5.1.3 Third Member will work on:
Model Evaluation.

5.2 QA-TestSet-LiveQA-Med-Qrels-2479-Answers Data
5.2.1 First Member will work on:
Data Cleaning and Feature Engineering.

5.2.2 Third Member will work on:
Build model and Data training.

5.2.3 Second Member will work on:
Model Evaluation.

6 REFERENCES

@ARTICLEBenAbacha-BMC-2019, author= Asma Ben Abacha and Dina Demner-Fushman, title= A Question-Entailment Approach to Question Answering, journal=BMC Bioinform., volume= 20, number= 1, pages= 511:1–511:23, year= 2019, url = https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-019-3119-4