A Constrained Long-form Question Answering (LFQA) Model

Jingsong Gao, jg2109

Qinren Zhou, qz142

Rui Qiu rq47

Introduction

The search engines nowadays enable us to ask any questions as a search query. However, when it comes to a scientific question, the answer to that question could be either not accurate enough or too specialized to understand. For instance, to fully apprehend a Wikipedia page, one might need much prior knowledge in a particular area.

Prior research on long-form question answering aims to fix this issue. ELI5: Long Form Question Answering by Facebook (2019) and Hurdles to Progress in Long-form Question Answering by Google (2021) are two representative papers in this area. Facebook created the ELI5 dataset and used the ROUGE-L metric to evaluate the long-form answers in their paper, which the dataset and metric widely used in later research. However, Google’ s paper pointed out that the dataset has severe train/validation overlapping and the metric showing a lack of distinction between random answers and gold answers.

Therefore, instead of training models on a dataset with potential issues and gaining higher grades using an invalid metric, our project will focus on recreating a categorized ELI5 dataset and proposing more discriminatory metrics for this LFQA task.

Data

The subreddit r/explainlikeim5 (ELI5) from Reddit seems to be a worthy training dataset. Users in this subreddit are known for their objective, thorough and intuitive explanations of various questions. ELI5 is appealing because answers are supposed to be entirely self-contained and thus rely less on pre-existing knowledge of the world and use more straightforward language that is easier to model.

The ELI5 dataset created by Facebook contains more than 270,000 posts in the subreddit from 2012 to 2019, and each post consists of a scientific question and some easy-to-understand answers. They also used the Wikipedia dataset as the ground truth to support the model answering those questions.

With the introduction of the thread tagging system in 2017, the questions that appeared in the subreddit are more organized in such a manner. So, we attempt to **build our abridged but categorized version of the ELI5 dataset using posts in the subreddit from 2017 to 2021**. By splitting the dataset with tags, we expect to obtain a non-overlapping training and validation set. Also, some stricter selection criteria will be applied to the dataset, e.g., a gold answer to the question should attain a score higher than 5; the maximal number of gold answers to a question will be limited to 5 during training. By setting these rules, we expect the dimensions of the training dataset, along with the support document, to be controlled under a reasonable capacity.

Evaluation

For long-form answers generated by the model, many aspects need to be evaluated, such as topical, accuracy, fluency, and coherence. Google’ s paper suggests the ROUGE-L metric has a small margin between the lower bound(question copies five times) and the upper bound(gold answer). So, we will examine some other metrics like **Precision, Recall, F1-score, BLEU, METEOR, BERTScore, BLEURT**. In addition, we will also introduce some human evaluations to see whether answers are fluent in presentation and coherent from start to end, although this probably highly depends on the de facto size of our homebrewed data set.

Other than that, if time allows, we can test run the sample answers generated by our model as a Reddit bot (of course, it should follow the subreddit’ s rules, pointing out that the answer is auto-generated.) In this way, we might acquire some extra human evaluation on the model. It would provide **some one-dimensional (the score) feedback on whether the answer makes sense**. Nevertheless, it will give us some reflection from the outsiders on freshly new questions.

Model

Since our project prioritizes recreating a tagged dataset and examining new metrics, we intend to only train some baseline models on the new dataset. These LFQA models are two-step models which contain a support document retriever and an answer generator.

First, we would like to use a **BERT model as a support document retriever**: Find related support documents according to the question.

Then the following models will be used as generators:

1. **Naïve TFIDF model**: Find seven sentences from the support document with the highest TFIDF similarity with the question.
2. **BART model**: Generate answers using questions, support documents as input.

Challenge

The main challenge of this project is the **computational resources**. As Google mentioned in their paper, their models were trained using 64 Google Cloud TPUs for a total of 32 hours. We have neither that much computing hardware nor that long period for training. Therefore, we will focus on creating a dataset, examining metrics, and only training a baseline model instead of a state-of-the-art model.

Expectation & Demonstration

We expect to obtain an NLP model that can give reliable, detailed, and easy-to-understand answers to scientific questions. We can build **a QA chatbot or an interactive web app** based on this model. Users can get answers either by asking the chatbot questions or by inputting questions on the web page. In addition, **a research report** will record the research process and conclusion, and this report will serve as documentation and guide for this project.