CS230

QuAMD: An end-to-end oracle for Question and Answering in the Medical Domain

Jacky Lin
Department of Computer Science
Stanford University
jackylin@stanford.edu

Amir Sahabi
Department of Computer Science
Stanford University
asahabi@stanford.edu

Richard Wang
Department of Computer Science
Stanford University
rcmwang@stanford.edu

1 Introduction

An effective question answering (QA) system presents significant opportunities, especially in the medical domain, where doctors can obtain answers to obscure medical questions. A question-answering pipeline consists of three main stages [Nogueira and Cho, 2019]. First, a large number possibly relevant documents to a given question are retrieved through a search engine. Next, each of these documents is scored and re-ranked by a more computationally-intensive method. Finally, the top documents will be the input for a QA system for extracting or generating the answer.

Significant advancement in QA models largely stems from the quality of datasets [Rajpurkar et al., 2016][Rajpurkar et al., 2018] and the success of BERT [Devlin et al., 2018]. Recently, Guu et al. [2020] augmented language model and allows it to retrieve documents from a large corpus in pre-training, fine-tuning and inference. Karpukhin et al. [2020] pointed out the importance of the document retrieval quality for successful answer extraction. The authors index the entire corpus with dense representation to allow users efficiently retrieving the relevant passages at run time. In this project, we build our work on pre-trained models and attempt to establish a BERT-based end-to-end QA system for the biomedical domain. For a biomedical question, our system first uses the PubMed search to retrieve the abstracts of relevant articles. Most of our effort in the retrieval stage is to understand the PubMed search and design queries that effectively retrieve the highly relevant documents. For the second stage, we begin with the BERT pre-trained model (PR) for passage re-ranking proposed in [Nogueira and Cho, 2019]. PR model was trained on MS MARCO constructed with Bing search results [Bajaj et al., 2016]. To improve the performance of the PR model for the biomedical domain, we construct a dataset (BioASQRV) from BioASQ 7b [Tsatsaronis et al., 2015]. BioASQRV includes both relevant and irrelevant question and context pairs. We fine-tune the PR model with BioASQRV and evaluate the model performance on a PubMedQA dataset (PMQAD) provided by [Jin et al., 2019]. For the final answer extraction stage, we use BERT as the baseline and compare the performance of BioBERT [Yoon et al., 2019] and BlueBERT [Peng et al., 2019] pre-trained models. [Yoon et al., 2019] conducted a detailed study and provide their pre-trained BioBERT + SQuAD model and pre-processed BioASQ 6/7b datasets in the SQuAD format. We essentially repeat a similar experiment by fine-tuning the pre-trained model with the BioASQ 7b dataset and evaluate the model on BioASQ 6b dataset with our metrics for comparison. [Peng et al., 2019] does not include studies on the QA task. We first train BlueBERT on SQuAD and then fine-tune and evaluate the resulting model with BioASQ 7b and BioASQ 6b datasets.

CS230: Deep Learning, Spring 2020, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

Our results show that the fine-tuning PR model on biomedical dataset improves the performance of revealing relevance between a question and a context. However, the performance difference among the three answer extraction models are less clear-cut. We find that hyperparameters play a vital role in the performance of the models. In general, we see that BioBERT has the best performance, and BlueBERT has the lowest performance after experimenting with different learning rates. However, our hyperparameter tuning is by no means exhaustive. We discuss the results of our experiment and potential research in the Analysis section and Future Work section.

2 Dataset

Passage Re-ranking Our datasets mainly come from two sources. For fine-tuning we use a dataset extracted from BIOASQ 7b provided by [Yoon et al., 2019]. To have a general overview for BIOASQ, see Tsatsaronis et al. [2015]. For testing, the dataset comes from PubMed provided by [Jin et al., 2019]. For fine-tuning the passage re-ranking model, we begin with the pre-processed dataset of BIOASQ 7b training set and construct a new dataset (BioASQRV) for the question and passage relevancy task in the biomedical domain. The pre-processed dataset includes different types such as "factoid" and "snippet", see [Yoon et al., 2019] for details. We use the "snippet" part, extract the question and context pairs and add a label "1" to form the positive part of BioASQRV.

To form a complete BioASQRV, we construct negative samples from the same source BIOASQ 7b training set. We describe the process as follows:

1. For each sample, we take the question q.
2. We randomly sample a context c that is not paired with the question q, and therefore context c is irrelevant to question q.
3. To further reduce the possibility of sampling a relevant context to the given question q, we compute the relevancy probability between c and q with the pre-trained PR model [Nogueira and Cho, 2019]. If the probability is higher than a certain threshold, we discard the sample c and continue the process until we find the sample context c that has probability less than the threshold.
4. We form the same number of negative samples as positive examples and label each of them with "0".

We set the threshold in the third step to 0.3. The newly constructed dataset BioASQRV consists of 54K examples. We shuffle all the examples and divide them with 80-20 for a train set and dev set. For the test set, we use the PMQAD dataset provided in [Jin et al., 2019].

Extractive Question Answering For stage 3 extractive QA model, we use two sources of datasets. To pre-train the QA model for BioBERT or BlueBERT, we use SQuAD 1.1 [Rajpurkar et al., 2016]. For fine-tuning the model for the biomedical domain, we use pre-processed BioASQ 6b/7b datasets provided by [Yoon et al., 2019]. Specifically, we use a full abstract BioASQ 7b training set as our training set (Train Set). Because the BioASQ test set does not provide the ground truth answer, we use a full abstract BioSAQ 6b training set as our development set (Dev Set). To have a general overview for BioASQ, see Tsatsaronis et al. [2015]. We choose SQuAD 1.1 instead of SQuAD 2.0 for pre-training because each question in the BioASQ Train Set and Dev Set for the fine-tuning task has an answer in the context.

3 Model

PubMed Search We first use the PubMed website to search for documents PMIDs, and then use Entrez[NCBI], a PubMed API, to retrieve the abstracts of the documents with the PMIDs. PubMed search results depend on query construction. We use ScispaCy [Neumann et al., 2019] to extract the keywords (name entities) from the input question for search. We construct our queries with the keywords and four rounds of operations during the PubMed website search until we reach the target number of article PMIDs. Each round loosens the query from the previous one and potentially yields more returns. We begin with search by phrases. If we do not hit the target number, then we break the phrases into words and search again. For these two rounds, we combine keywords by 'and'. If more

Context	Relevancy %	Percentage % MS MARCO	Percentage % MS MARCO + BioASQRV
Long	> 10	97.0	99.4
	> 50	96.3	99.2
	> 90	95.6	99.2
Short	> 10	85.0	98.9
	> 50	81.6	98.9
	> 90	76.0	98.6

Table 1: Percentage of relevancy for the original PR model and fine-tuned model on BioASQRV dataset

articles are required, the algorithm will relax again and search with only the keywords/keyphrases that have an upper case in the first letter. Our last resort is to search for keywords with the operation 'or'. All the searches are based on 'relevant matches' from the PubMed website. Moreover, we put a time delay when we need multiple searches to build the contexts for our test dataset.

Passage Re-ranking We use the pre-trained BERT-Base uncased model (PR) provided by [Nogueira and Cho, 2019]. The model was and fine-tuned on MS MARCO [Bajaj et al., 2016] constructed with real Bing search queries along with passages extracted from the top Bing search results. To improve the performance on the relevancy search in the biomedical domain, we propose to fine-tune the model on the biomedical dataset BioASQRV we constructed based on BioASQ 7b provided in [Yoon et al., 2019].

Answer Extaction In addition to the original BERT model [Devlin et al., 2018], Our pre-trained models come from two groups of researchers. The BioBERT + SQuAD 1.1 pre-trained model are provided by [Yoon et al., 2019], and the other two models BlueBERT PubMed (BlueBERT A) and BlueBERT PubMed + MIMIC III (BlueBERT B) are provided by [Peng et al., 2019]. All BERT, BioBERT and BlueBERT are base models. BERT and BioBERT are cased, and BlueBERT models are uncased.

4 Evaluation

We use accuracy for our evaluation during stage 1 and stage 2. For stage 3, we use the exact match (EM) and Macro-averaged F1 (F1) metrics to evaluate extractive QA model accuracy [Rajpurkar et al., 2016]. Exact match measures the percentage of predictions that match any one of the ground truth answers exactly. Macro-averaged F1 score measures the average overlap between the prediction and ground truth answer. We do not distinguish letter cases.

For the evaluation the overall QA system that goes through all three stages, we search for the contexts for each question in the Dev Set (BioASQ 6b), re-rank them, keep the top 5 documents and infer the answers to compute EM and F1 as described in [Chen et al., 2017]. For each question, we choose a single answer with the highest probability among the five documents for F1 and EM scores.

5 Experiment

PubMed Search We use the PubMed website to retrieve PMIDs and Entrez to retrieve the documents. The Entrez API interacts with the legacy PubMed system and requires a strict query pattern to search for the target documents effectively. For example, using a plural form of a noun sometimes results in no return. PubMed website enriches the search with Automatic Term Mapping and query translation that allow more flexible input and relevant returns [HelpDesk, 2020]. Even so, direct input of the entire question may return no result. We have first to pre-process the question to retrieve the keywords (name entities) with ScispaCy and systematically search with different queries as described in the Model section. We set our target to 50 PMIDs per question.

Metric	BERT	BioBERT	BlueBERT A	BlueBERT B
EM	85.77	86.59	85.18	85.56
F1	88.09	88.96	86.83	87.38

Table 2: F1 and EM for each model pre-trained on SQuAD and fine-tuned on BioASQ 7b (Train Set). The evaluation is on BioASQ 6b (Dev Set).

Passage Re-ranking We first compute the relevancy probability between the question and the context for each example with the PR model. We show that the relevancy score of test examples from the PMQAD dataset between a question and a "short context" and a question with a "long context". A "short context" is essentially a summary of the "long context", and is easier for humans to answer the question [Jin et al., 2019]. Our test results show that the PR model performs well for the question vs "long context" pairs. However, the relevancy between a question vs a "short context" pair does not show satisfying results. To improve the performance of the PR model for the biomedical problem, we use Hugging Face Classification implementation to fine-tune the PR model with our newly constructed BioASQRV dataset and test the relevance between a question and its context. Our results show that the fine-tuned model (BioPR) improve performance significantly in the "short context" case Table 1.

Answer Extraction We use Hugging Face Question Answering implementation to train and fine-tune the pre-trained models. We use SQuAD 1.1 for all training and BioASQ 7b for fine-tuning.

We begin with the BioBERT pre-trained model. Although Yoon et al. [2019] provides a pre-trained BioBERT + SQuAD, we pre-train the SQuAD and fine-tune it with the BioASQ 7b training set (Train Set). Because BioASQ 7b does not have a test set that contains ground truth, we use the BioASQ 6b training set as our development set (Dev Set) for performance evaluation. Note that Yoon et al. [2019] has conducted similar studies. We simply repeat the experiment and evaluate it with our metrics for performance comparison.

For BlueBERT, we experimented with both pre-trained models BlueBERT A and BlueBERT B. Peng et al. [2019] did not include QA studies. We again employ two steps to train and fine-tune the models. We first train them on SQuAD and subsequently fine-tune the results with the same BioASQ Train Set and Dev Set.

Finally, we train our baseline model BERT + SQuAD + BioASQ 7b in two steps similar to BlueBERT. We first train BERT on SQuAD, then fine-tune it on the BioASQ dataset.

The hyperparameters we use for SQuAD training are as follows. We use learning rate 3e-5, epochs 2, maximum sequence length 384, document stride 128, batch size 12. For BioASQ, we use learning rate 4e-5 and epochs 5. Other parameters are the same as SQuAD.

Table 2 shows the F1 and EM for each model. The column name shows the type of BERT model we use with SQuAD and BioASQ 6/7b. For example, "BERT" column refers to BERT + SQuAD + BioASQ 6/7b. Similarly, other columns. We are still in the process of understanding the results. Please see the Analysis section.

Combined QA System Once we obtain the fine-tuned model for each stage, we apply them and evaluate the performance of the overall QA system. For each question in the Dev Set, we search and select the top 5 abstracts with the PubMed search and BioPR passage re-ranking model to construct a test dataset (Test Set). The Test Set has the same question and answer as the Dev Set but has different contexts from the Dev Set. Each question has five different contexts.

We compute the F1 and EM scores on the Test Set for each of the fine-tuned models obtained from the Answering Extraction paragraph. For each question, we choose the context which has the highest probability of its predicted answer as our prediction. Table 3 shows the results. We see that the performance decreases significantly compared the results in Table 2. We discuss the results in the Analysis section.

Metric	BERT	BioBERT	BlueBERT A	BlueBERT B
EM	49.1	50.0	41.6	45.6
F1	56.8	57.4	48.0	52.4

Table 3: F1 and EM for each model pre-trained on SQuAD and fine-tuned on BioASQ 7b (Train Set). The evaluation is on the Test Set constructed by the top 5 abstracts of the PubMed search returns of Dev Set questions and passage re-ranking.

6 Analysis

Answer Extraction The F1 and EM results in Table 2 for the answering extraction models depends on hyperparameter-tuning. Our earlier findings show that BERT baseline has the highest performance, and BlueBERT B (trained with MIMIC-III) performs lower than BlueBERT A. After increasing the learning rate, we obtain higher scores for all of the models. Furthermore, BioBERT has the highest performance, and BlueBERT B performs better than BlueBERT A. Our hyperparameter tuning is by no means thorough. For example, the BioASQ 7b is a smaller data, and using a higher number of epochs could impact the final results. We leave a complete hyperparameter tuning as future work.

Combined QA System The quality of the overall QA system depends on the search returns. If many of the return abstracts in the Test Set contain no answer, overall F1 and EM scores will drop significantly. To understand the quality of models for the first and the second stages, we compute the proportion of the retrieved documents in the Test Set that contains correct answers. The metric offers feedback on the quality of the search and re-ranking. It is also an upper bound of the overall QA system.

We first examine the PubMed search returns. If we consider the top 5 PubMed return contexts for a question as a group, 67.6% of the questions contain at least one answer in the top 5 contexts. We perform a similar calculation for the Test Set, which we built by the top 5 re-ranking of the PubMed returns. We again view the five contexts for a question as a group. We find 73.8% of the questions have at least one abstract that contains the answer. The passage re-ranking increased by 7.2% from the PubMed search.

The performance of the re-ranking model (BioPR) may be improved if we enhance the fine-tune dataset BioASQRV. During the dataset construction, we randomly sample an unrelated context to each question to form a negative sample. However, many of the PubMed returns are relevant to the question but do not contain the gold answer. If we add such relevant contexts into the dataset to form negative examples, the fine-tuned model may be able to distinguish better a relevant context that does not contain the answer from one that contains the answer. Besides, we currently construct an equal number of negative samples and positive samples in the BioASQRV. PubMed returns are biased toward negative samples. The performance of our fine-tune model BioPR may improve if we use a sample distribution more closely to the PubMed returns.

7 Future work

We currently access the PubMed website to search article PMIDs, then use Entrez to retrieve the articles. We do this because the PubMed search highly depends on the keywords. Current Entrez API interacts with the legacy system [HelpDesk, 2020] and has minimal tolerance on the input. For example, using a plural form of a noun may yield no returns from Entrez. An efficient automatic query construction is crucial for using Entrez to search in our project. PubMed is in the process of developing a RESTful API. We will switch to the new API once it becomes available.

We are interested in improving the BioPR passage re-ranking model. The current negative samples come from irrelevant, random contexts in the PMQAD that do match the question. If we include relevant samples that contain no answer from the PubMed search returns as negative samples in the BioASQRV dataset, the model may be more sensitive to subtle differences and include the context that contains the true answer in the top list.

Finally, we plan to systematically tune the hyperparameters to understand the real performance of BERT, BioBERT and BlueBERT models in the Experiment Section.

8 Contribution

We have weekly meeting and work across different areas. Everyone contributes to the ideas and discussions. Jacky Lin works on Question Answer extraction. Amir Sahabi works on PubMed Retrieval and User Interface. Richard Wang works on all models and analysis.

References

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. MS MARCO: A human generated machine reading comprehension dataset, 2016.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions, 2017.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Realm: Retrieval-augmented language model pre-training, 2020.

PubMed HelpDesk. Email communication with PubMed Help Desk, 2020.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William W. Cohen, and Xinghua Lu. PubMedQA: A dataset for biomedical research question answering, 2019.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen tau Yih. Dense passage retrieval for open-domain question answering, 2020.

NCBI. Entrez programming utilities help [internet]. bethesda (md): National center for biotechnology information (us); 2010-. available from: https://www.ncbi.nlm.nih.gov/books/nbk25501/.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. ScispaCy: Fast and robust models for biomedical natural language processing. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 319–327, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5034. URL `https://www.aclweb.org/anthology/W19-5034`.

Rodrigo Nogueira and Kyunghyun Cho. Passage re-ranking with BERT, 2019.

Yifan Peng, Shankai Yan, and Zhiyong Lu. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets, 2019.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text, 2016.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD, 2018.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R. Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, Yannis Almirantis, John Pavlopoulos, Nicolas Baskiotis, Patrick Gallinari, Thierry Artières, Axel-Cyrille Ngonga Ngomo, Norman Heino, Éric Gaussier, Liliana Barrio-Alvers, Michael Schroeder, Ion Androutsopoulos, and Georgios Paliouras. An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition. *BMC Bioinformatics*, 16, 2015.

Wonjin Yoon, Jinhyuk Lee, Donghyeon Kim, Minbyul Jeong, and Jaewoo Kang. Pre-trained language model for biomedical question answering, 2019.