Transformer-Based Models for Question Answering on *COVID19*

A Preprint

Hillary Ngai [1][2] Yoona Park [1][2] John Chen [1][2]
Mahboobeh Parsapoor (Mah Parsa) [1][2]
[1]Vector Institute for Artificial Intelligence
[2]University of Toronto,
{hngai, ypark, johnc, mahparsa}@cs.toronto.edu

January 28, 2021

Abstract

In response to the Kaggle's COVID-19 Open Research Dataset (CORD-19) challenge, we have proposed three transformer-based question-answering systems using BERT, ALBERT, and T5 models. Since the CORD-19 dataset is unlabeled, we have evaluated the question-answering models' performance on two labeled questions answers datasets —CovidQA and CovidGQA. The BERT-based QA system achieved the highest F1 score (26.32), while the ALBERT-based QA system achieved the highest Exact Match (13.04). However, numerous challenges are associated with developing high-performance question-answering systems for the ongoing COVID-19 pandemic and future pandemics. At the end of this paper, we discuss these challenges and suggest potential solutions to address them.

***K*eywords** CORD-19 · COVID-19 · question-answering systems · ALBERT · BERT · T5

1 Introduction

Question Answering or **QA** systems may be useful for the medical research community to stay up-to-date when new literature is rapidly growing. However, due to the lack of labeled question-answer pairs, developing accurate QA systems is challenging. Thus, one solution is to fine-tune pre-trained transformer-based QA systems [1] [1, 2, 3].

In this paper, we propose three QA systems developed for the Kaggle *COVID-19 Open Research Dataset,* or **CORD-19** dataset [2]. We built the QA systems after carefully reviewing various QA systems submitted to the CORD-19 Kaggle competition. We have also presented our preliminary results obtained from evaluating the performance of three transformers: *Bidirectional Encoder Representations from Transformers* or **BERT** [4], ALBERT, and *Text-to-text transfer transformer* or **T5** model on two new QA datasets —CovidQA and CovidGQA. Finally, we discuss the challenges of developing a high-performance QA system for COVID-19-related research and suggest solutions to improve performance.

2 A Review of QA Systems for the Kaggle CORD-19 Dataset

More than 1,000 teams participated in the Kaggle CORD-19 dataset competition. They used various *Natural Language Processing)* or**NLP** approaches, including BERT [4]). Since most QA systems in the competition were developed based

[1] A typical transformer-based QA system uses a parallelizable architecture to efficiently find answers (i.e., a segment of text) to a query.

[2] They launched a competition to provide a chance for the NLP community to develop QA systems for medical experts to find answers to high-priority medical questions related to COVID-19

on BERT and *Latent Dirichlet Allocation* or (LDA), the following subsections just review LDA-based QA systems and BERT-based QA systems submitted to the competition.

2.1 LDA-based QA Systems

Using LDA to develop QA systems is considerably uncomplicated [5, 6, 7, 8]. Thus, for the CORD-19 Kaggle competition, various LDA-based QA systems were developed. For example, one approach [3] [9] combined the **whoosh** search engine and used Jensen-Shannon distance to discover topics related to *What do we know about COVID-19 risk factors?* and to find documents similar to those topics. Another interesting LDA-based QA [4] was proposed in [10]; It combined k-means clustering and LDA to cluster documents and discover topics from clusters to facilitate extracting articles from the CORD-19 dataset. In [11], a combination of LDA and Anserini (i.e., an open-source information retrieval approach) was proposed to develop a QA system that could find articles related to non-pharmaceutical intervention. The main aim of this work was to discover new interventions in specific environments by incorporating the context for each response in the search and guide policymakers to take appropriate actions to control spreading COVID-19 virus.

2.2 BERT-based QA systems for CORD-19

Since one of BERT's applications is in QA, different variations of BERT [12, 13, 14, 15] have been used. For the competition, many teams used BERT to develop QA systems [16, 17]. For instance, [16] used BERT to find relevant answers to keywords extracted from a question. The found solutions were ranked by *Universal Sentence Encoder Semantic Similarity* or **USESS** then *Bayesian Additive Regression Trees* or **BART** summarized the top results. Another team employed BERT as a semantic search engine to find answers. The QA system produced semantically meaningful sentence embedding on the paragraph extracted from the CORD-19 dataset and found five paragraphs and their corresponding papers' titles and abstracts. In [17], the QA systems were based on *A little BERT* or **ALBERT** [14] to find answers for questions related to COVID-19.

3 Transformer-Based QA Systems

For the competition, we aimed to develop a QA system using a high performance Transformer. To do so, we developed three QA systems using BERT-large, ALBERT-base and *Text-to-text transfer transformer* or (T5)-large, pre-trained them on various QA datasets and evaluated them on two labeled questions answers datasets —CovidQA, and CovidGQA as described below.

3.1 Datasets for Pre-training Transformers

We use various datasets to pre-train our QA systems: 1) **SQuAD v1.1** [18] —the *Stanford Question Answering Dataset* containing 100k question-answer pairs on more than 500 articles; 2) **SNLI** [19] —the *Stanford Natural Language Inference* corpus containing 570k human-written English sentence pairs manually labeled for balance classification with the labels entailment, contradiction and neutral; 3) **MultiNLI** [20] —the *Multi-Genre Natural Language Inference* corpus containing 433k crowd-sourced sentence pairs with the same format as SNLI except it includes a more diverse range of text and a test set for cross-genre transfer evaluation; 4) **STS** —the *Semantic Textual Similarity* benchmark is a careful selection of data from English STS shared tasks (2012-2017) comprising of 8.6k annotated examples of text from image captions, news headlines, and user forums; and 5) **BioASQ** [5] —the question-answering biomedical dataset consists of 1k questions with "exact" and "ideal" answers. We specifically use BioASQ factoid QA pairs, excluding yes/no or list QA pairs, because the factoid dataset has a similar structure as SQuAD v1.1 [18].

3.2 Datasets for Evaluating Transformers

After pre-training our models, we evaluate our QA systems on **CovidGQA** and **CovidQA**. The former [6] is a COVID-19 dataset created manually and encompasses 198 general question-text-answers related to COVID-19[7]. The question-text

[3] among 388 submitted kernels

[4] It was received a lot of attention among other teams and was one of the competition's kernels that obtained the highest number of votes around 783.

[5] http://bioasq.org/

[6] an example of a question in the CovidQA dataset is: What is the incubation period of the virus?

[7] an example of a question in the CovidGQA dataset is: How can I protect myself from getting COVID-19?

has been extracted from medical websites, and medical *subject-matter experts*or **SMEs** have provided answers. The latter is a COVID-19 question-answering dataset built by hand from knowledge gathered from the Kaggle CORD-19 dataset [21]. We merged the CovidQA dataset with the CORD-19 dataset to extract each article's relevant text to answer each question in the dataset. The final evaluation dataset contained 69 question-text-answer triplets.

3.3 BERT-large

Our BERT-large QA system is developed using a pre-trained QA BERT-large-uncased model with whole word masking fune-tuned on SQuAD v1.1 [18]. The model contains 24 Transformer blocks, 1024 hidden layers, 16 self-attention heads adding up to 340M parameters in total.

3.4 ALBERT-base

The ALBERT-base QA system (Figure 2) is forned using a pre-trained QA ALBERT-base-uncased model fine-tuned on SQuAD v1.1 [18]. The model contains 12 Transformer blocks, 768 hidden layers, 12 self-attention heads, adding up to 12M parameters in total.

3.5 T5-large

The T5-large QA system (see Figure 3) is based on the T5 model that is a modern, massive multitask model trained by uniting many NLP tasks in a unified text-to-text framework [22]. By leveraging extensive pre-training and transfer learning, it has achieved state-of-the-art performance on a variety of NLP benchmark tasks, including the GLUE benchmark [23]. Following work by [24], which explores the task of generative closed-book question answering, we explore the efficacy of generating (rather than extracting) COVID-19 answers directly from an input question, without context. Unlike our preceding two approaches, the T5 model explores generation of answers to questions, without context. Using the pre-trained T5 model with 770M parameters released by [22], we fine-tune for 25000 steps on an equal-proportions mixture of three QA tasks using the Natural Questions dataset, Trivia QA dataset and the train split of the COVID-19 QA dataset [25, 26]. Only the queries are given as input and answers are generated using simple greedy decoding. Evaluation is then performed on the test split of the COVID-19 dataset. We emphasize that these results are not directly comparable to the other frameworks, as the model is faced with the challenging task of jointly localizing relevant information and then generating a coherent answer. The advantage of such a framework is that it is context-free, meaning that it requires the least data preparation and human intervention.

3.6 Evaluation of QA Systems

We evaluate the above transformers using two datasets —CovidQA and CovidGQA. Using the same evaluation metrics as SQuAD v1.1, we calculate the macro-averaged F1 score and *Exact Match* or **EM** of the answer extraction methods of our QA systems on each dataset. Both BERT-large-uncased and ALBERT-base-uncased use whole word masking and are pre-trained on SQuAD v1.1, while T5-large was pre-trained on *Colossal Clean Crawled Corpus* or **C4**. Figure 1 shows the comparison of generated answer lengths on the CovidGQA dataset.

Dataset	QA System	F1 Score	EM
CovidQA	ALBERT-base	23.37	**13.04**
	BERT-large	**26.32**	11.59
	T5-large	5.16	0.00
CovidGQA	ALBERT-base	28.08	**9.64**
	BERT-large	**29.96**	5.08
	T5-large	5.95	0.00

Table 1: Preliminary results obtained using QA systems on CovidQA and CovidGQA.

BERT-large achieves the highest macro-averaged F1 score on both datasets. Furthermore, BERT-large outperforms ALBERT-base on GLUE, RACE, and SQuAD benchmarks [14]. However, ALBERT-base unexpectedly outperforms BERT-large on EM, achieving the highest EM of the three models. Further experimentation is required to investigate the cause of this. Despite T5-large having the most significant number of parameters (770M), both BERT-large and ALBERT-base outperform T5-large on all metrics on both datasets. This may be explained by the fact that T5-large was not pre-trained on a QA task. The reasonable performance of these three transformers, motivated us to use them to develop QA systems for the CORD-19 Kaggle competition. The next section discusses the QA systems in more detail.

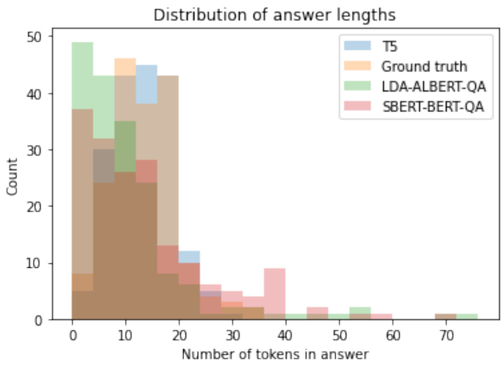

Figure 1: Comparison of generated answer lengths on the CovidGQA dataset.

4 QA Systems for COVID-19

Based on the results showed in the previous section, we have developed two QA systems on the CORD-19 dataset which aim to help the medical community answer high-priority scientific questions such as *“What is the efficacy of novel therapeutics being tested currently?"* or *“What is the best method to combat the hypercoagulable state seen in COVID-19?"*. We designed two BERT-based question-answering systems and a T5 question-answering system. Each QA system is pre-trained on different datasets and evaluated on two COVID-19 datasets.

4.1 SBERT-BERT-QA

The SBERT-BERT-QA system (see Figure 2 [8]) combines Sentence-BERT and BERT-large. We first filter the articles using a keyword search based on a set of pre-defined keywords such as **RNA virus, clinical, naproxen, clarithromycin**. Once filtered, the top n articles were extracted by embedding the query and the article titles using a pre-trained Sentence-BERT model (i.e., a BERT-base model with mean-tokens trained on SNLI and MultiNLI corpora and then on STS Benchmark training set) [27]. Sentence-BERT (SBERT) is a modification of the BERT network using Siamese and triplet network structures to derive semantically meaningful sentence embeddings. The top n articles were extracted by taking the articles with the highest cosine similarity scores between the query embedding and each article title embedding. Once the top article were extracted, each article's answer to the query was extracted using a pre-trained QA BERT-large-uncased model.

4.2 LDA-ALBERT-QA

The LDA-ALBERT-QA (see Figure 2[9]) combines LDA and pre-trained ALBERT-base. First, we filter out the irrelevant articles from the CORD-19 dataset using LDA to provide a dataset containing only the relevant articles to the query. Then, the filtered documents are fed into a pre-trained QA ALBERT-base-uncased model to extract an excerpt from articles that are relevant to the query. We have utilized **SQuAD v1.1** [18] and **BioASQ** 6b factoid QA pairs to develop the pre-trained ALBERT model. The primary reason for using SQuAD v1.1 is to overcome the data shortage of BioASQ 6b factoid, which contains less than 1k QA pairs. After pre-training the model on SQuAD v1.1, we further pre-train the model with BioASQ dataset to target biomedical domain. A single output layer on top of the pre-trained ALBERT model performs token-level classification to compute the start/end index of a predicted answer from each article.

5 Conclusion

We presented two transformer-based QA systems to compete in the competition. We selected transformers based on the preliminary results obtained from BERT, ALBERT, and T5 model on two QA datasets. As our results indicated, BERT-large could achieve the highest F1-score for both datasets and one of our first candidates to develop a QA system. Our results also showed that ALBERT-base could achieve the highest EM score for both datasets and was our second nominee to establish a QA system. We consider that one of the significant limitations of transformers is that they require

[8] The system diagram of SBERT-BERT-QA and LDA-ALBERT-QA. The dotted arrows and the white rectangles represent the path of SBERT-BERT-QA.

[9] The dashed arrows and the dark rectangles represent the path of LDA-ALBERT-QA.

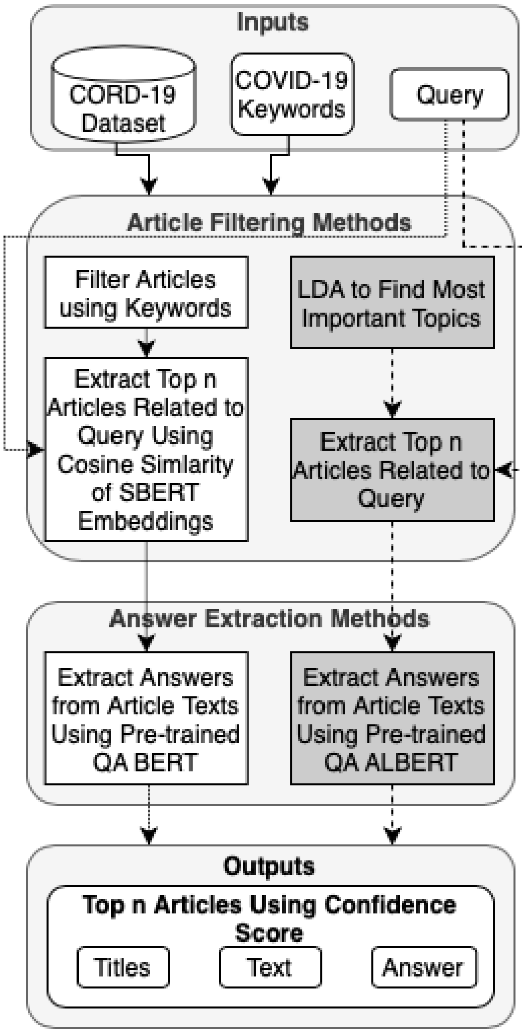
Inputs
CORD-19 Dataset
COVID-19 Keywords
Query
Article Filtering Methods
Filter Articles using Keywords
LDA to Find Most Important Topics
Extract Top n Articles Related to Query Using Cosine Simlarity of SBERT Embeddings
Extract Top n Articles Related to Query
Answer Extraction Methods
Extract Answers from Article Texts Using Pre-trained QA BERT
Extract Answers from Article Texts Using Pre-trained QA ALBERT
Outputs
Top n Articles Using Confidence Score
Titles
Text
Answer

Figure 2: BERT-based QAs

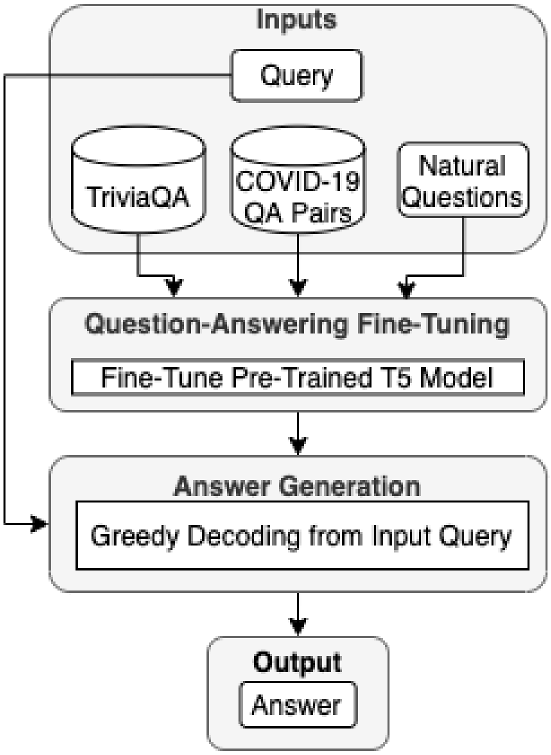
Inputs
Query
TriviaQA
COVID-19 QA Pairs
Natural Questions
Question-Answering Fine-Tuning
Fine-Tune Pre-Trained T5 Model
Answer Generation
Greedy Decoding from Input Query
Output
Answer

Figure 3: Context-free T5 QA system

a lot of labeled QA pairs to reach acceptable performance. We aimed to resolve the limitation by developing a hybrid QA system that combines few-shot learning with a transformer. LDA also has some limitations; for example, we need to determine the number of topics and consider the articles that are not too short. To address these issues, we explore prediction-focused supervised LDA and identify topics in an online manner. Furthermore, a QA system should be explainable and trustworthy to medical users. Developing such a QA system is possible if medical experts and NLP researchers cooperate closely.

Author's contributions

The first three authors equally worked with technical parts of the paper that has been summarized in Section 3 and 4. Dr. Parsa wrote sections 1 and 2 and reviewed the manuscript and provided valuable feedback on the manuscript.

References

[1] KSD Ishwari, AKRR Aneeze, S Sudheesan, HJDA Karunaratne, A Nugaliyadde, and Y Mallawarrachchi. Advances in natural language question answering: A review. *arXiv preprint arXiv:1904.05276*, 2019.

[2] D. Lukovnikov, A. Fischer, and J. Lehmann. Pretrained transformers for simple question answering over knowledge graphs, 2020.

[3] Betty van Aken, Benjamin Winter, Alexander Löser, and Felix A. Gers. How does bert answer questions? *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, Nov 2019.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[5] Asli Celikyilmaz, Dilek Hakkani-Tur, and Gokhan Tur. Lda based similarity modeling for question answering. In *Proceedings of the NAACL HLT 2010 Workshop on Semantic Search*, pages 1–9, 2010.

[6] Lin Cui and Caiyin Wang. An intelligent q&a system based on the lda topic model for the teaching of database principles. *World Transactions on Engineering and Technology Education*, 12:26–30, 01 2014.

[7] Hamidreza Chinaei, Luc Lamontagne, François Laviolette, and Richard Khoury. A topic model scoring approach for personalized qa systems. In *International Conference on Text, Speech, and Dialogue*, pages 84–92. Springer, 2014.

[8] Zongcheng Ji, Fei Xu, Bin Wang, and Ben He. Question-answer topic model for question retrieval in community question answering. In *CIKM '12*, 2012.

[9] Daniel Wolffram. discovid.ai - a search and recommendation engine. `https://www.kaggle.com/danielwolffram/discovid-ai-a-search-and-recommendation-engine`, April 2020.

[10] SoloNick Maksim Ekin. Covid-19 literature clustering. `https://www.kaggle.com/maksimeren/covid-19-literature-clustering`, April 2020.

[11] Jonathan Smith, Borna Ghotbi, Seungeun Yi, and Mahboobeh Parsapoor. Non-pharmaceutical intervention discovery with topic modeling, 2020.

[12] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 2019.

[13] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[14] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations, 2020.

[15] Ying-Hong Chan and Yao-Chung Fan. A recurrent bert-based model for question generation. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 154–162, 2019.

[16] Peilin Yang, Hui Fang, and Jimmy Lin. Anserini: Reproducible ranking baselines using lucene. *Journal of Data and Information Quality (JDIQ)*, 10(4):1–20, 2018.

[17] Shivam Abhilash Sandyvarma et al. Covid-19: Bert + mesh enabled knowledge graph. `https://www.kaggle.com/sandyvarma/covid-19-bert-mesh-enabled-knowledge-graph`, April 2020.

[18] Rajpurkar, Zhang, Lopyrev, and Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250v3*, 2016.

[19] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[20] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics, 2018.

[21] Tang, Nogueira, Zhang, Gupta, Cam, Cho, and Lin. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:2004.11339v1*, 2020.

[22] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2019.

[23] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

[24] Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model? *arXiv preprint arXiv:2002.08910*, 2020.

[25] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: A benchmark for question answering research, 2019.

[26] Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, 2017.

[27] Reimers and Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.