Readability Controlled Open-domain QA for COVID-19

Zhehan Qu[*]
Shanghai Jiao Tong University
Shanghai, China

Lumingyuan Tang[†] and **Xiao Liu**[‡]
NanKai University
Tianjin, China

Gary Bécigneul[§]
Gematria Technologies
London, U.K.

Abstract

The continuous spread of COVID-19 these days has been calling for a way to provide instant and readable knowledge for researchers, medical workers and the general public. Recent works on COVID related question answering (QA) systems have leveraged retrieval-based structures and yielded promising results, yet sometimes their answers could become unreadable for most users due to the highly technical nature of their knowledge source. We propose **RC-RAG** for Readability Controlled Retrieval Augmented Generation, an **open-domain question-answering** system on COVID related topics, which is capable of generating answers of a **chosen readability score** given a COVID-related question. Our model is proved to generate answers of different readability levels, and could meet the requirements of people with different levels of education. We hope our system will be able to aid researchers and the general public to go through this tough time together.

1 Introduction

Since the COVID-19 outbreak, the world has been demanding vast amount of information about this disease, and about how to guide our life in this new social context. Researchers, as well as the general public, are in desperate need of finding instant and reliable information regarding relevant topics about COVID. An NLP-based question-answering system, in this particular time, would be of great help to aid people in need, and help provide low-cost and accurate information to everybody.

A longstanding problem in NLP, information retrieval (IR) and related fields is **Open-domain Question Answering** (QA). Open-domain QA is a task consisting in answering factoid questions using a collection of documents. A good open-domain QA system must be able to retrieve and comprehend one or more knowledge sources to get the correct answer. Chen et al. (2017) considers answering factoid problems using several search methods such as TF-IDF matching and bigram hashing based on Wikipedia. Sun et al. (2018) aims at constructing the open-domain QA for generating answers from a question-specific subgraph containing text and Knowledge Bases of entities and relation. Wang et al. (2018) proposed an open-domain QA system which contains a Ranker component for ranking retrieved passages and a Reader component for extracting answers. Seo et al. (2019) introduces the query-agnostic indexable representation of document phrases that can speed up the open-domain QA system and allow us to reach long-tail targets.

RAG (Lewis et al., 2020) explores a general-purpose fine-tuning recipe for retrieval-augmented generation, and yields state-of-the-art in open-domain QA. The model contains a pretrained seq2seq model, which makes up the parametric memory, and a dense vector index of Wikipedia as the non-parametric memory, accessed with a pre-trained neural retriever. Despite its strong performance in open-domain QA, the answers generated from the model are strongly based on the knowledge base, therefore a very technical answer would be generated if the knowledge base is highly technical. Such behavior of retrieval-based systems is a strong fallback for users with relatively low educational levels, and is especially noticeable in works purposefully designed for COVID-related QA with a database of scientific articles (Su et al., 2020).

To resolve this issue, we first constructed a knowledge base [1] with both (1) technical texts from

[*] zhh_qu@sjtu.edu.cn
[†] tanglumy@mail.nankai.edu.cn
[‡] MerryXiao@mail.nankai.edu.cn
[§] gary@gematria.tech

[1] https://github.com/... released upon acceptance.

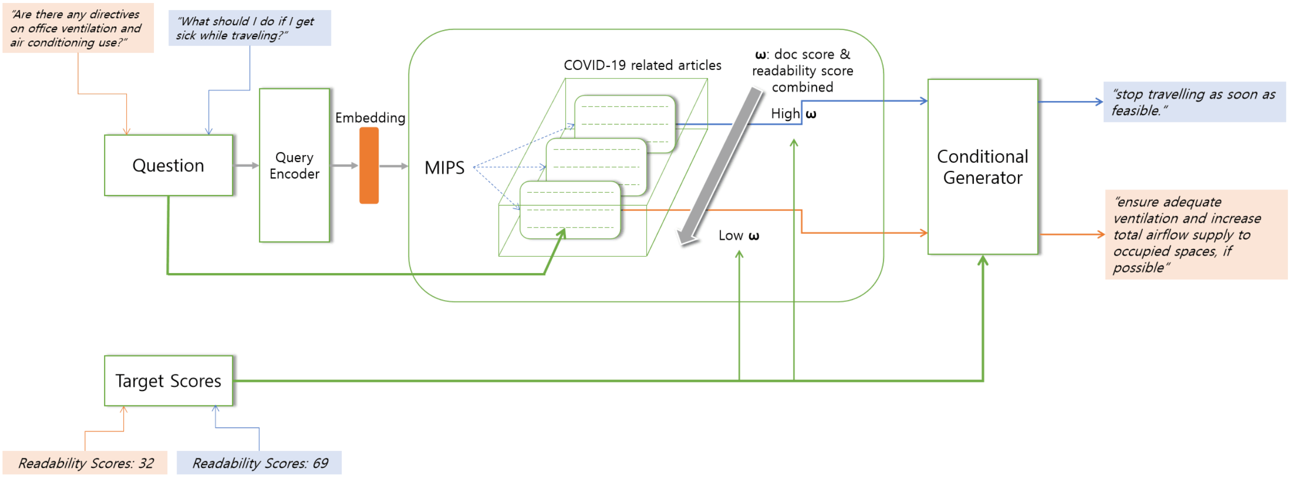
"Are there any directives on office ventilation and air conditioning use?"
"What should I do if I get sick while traveling?"
Question
Query Encoder
Embedding
MIPS
COVID-19 related articles
ω: doc score & readability score combined
High ω
Low ω
Conditional Generator
"stop travelling as soon as feasible."
"ensure adequate ventilation and increase total airflow supply to occupied spaces, if possible"
Target Scores
Readability Scores: 32
Readability Scores: 69

Figure 1: An overview of RC-RAG model. Target readability scores is used in the retriever after MIPS search to re-order the retrieved documents. We combine the retrieved documents with the target readability score to generate final answers.

CORD (Wang et al., 2020), a corpus of COVID-related scientific papers, and (2) a recent Wikipedia dump, curated using COVID-related keywords. To generate answers using this carefully designed database, we need to consider using additional factors to guide the answer-generation process of our model. Accordingly, we leverage a textual feature called readability (Coleman and Liau, 1975) to enhance our model, and plunge it to both the retriever and generator part of RAG, to control the reading ease of the answer obtained by the user.

To condition the generator of RAG with respect to readability scores, we introduce Readability-Controlled BART (RC-BART), following the idea of CTRL (Keskar et al., 2019), which introduces a way to generate text conditioned on chosen control codes, trained as a language model. CTRL has been used in question-answering tasks originally by adding "Question:" and "Answer:" as control codes, while in our case we instead want to control the readability of the answer generated. A detailed description of this model will be presented in Section 3.1.

To sum up, we proposed a readability-controlled retrieval-augmented open-domain question-answering system on COVID related topics, **RC-RAG**, using RAG as the basic structure and adding readability score as control code to tailor its output. Figure 1 shows the structure of our model.

Our model is able to generate answers that meet the chosen target readability score, which could be modified to give different answers of the same question. Our contribution is three-fold:

- We enhance BART with a readability controlled capacity via an efficient modification of its pre-training and architecture.
- We incorporate this readability controlled feature into RAG, turning it into the first model ever to provide controlled generation for open-domain QA.
- We train our RC-BART and RC-RAG on a new COVID-related dataset we built by combining several different data sources.

2 Background

RAG: Retrieval Augmented Generation

Lewis et al. (2020) recently proposed RAG, a general-purpose fine-tuning recipe for retrieval-augmented generation. RAG combines pretrained parametric and non-parametric memory for language generation, where the parametric memory is a pretrained seq2seq model and the non-parametric memory is a dense vector index of Wikipedia, accessed with a pretrained neural retriever. Two RAG formulations were proposed in this work, namely (1) **RAG-Sequence Model** which uses the same retrieved document to generate the complete sequence and (2) **RAG-Token Model** which uses a different latent passage for each target token and marginalize accordingly. For the retriever part, RAG uses DPR (Karpukhin et al., 2020), which uses two $\text{BERT}_{\text{BASE}}$ models for the encoding of the query and documents respectively, and employs a Maximum Inner Product Search (MIPS) index provided by the FAISS library (Johnson et al., 2019) to efficiently fetch the top-k documents from the knowledge base. The generator of RAG is a BART-large (Lewis et al., 2019) model, where the input and the retrieved documents are concatenated to fill as the input. RAG yields state-of-the-art in several areas, especially in open-domain question-answering. We use RAG as the basic structure of our model, and make several modifications to it to satisfy our conditional queries.

CTRL: Controlled Generation of Text

Keskar et al. (2019) introduced CTRL, a conditional transformer-based language model, which is trained to condition on control codes that govern style, content, and task-specific behavior. CTRL learns $p_\theta(x_i|x < i, c)$ by training on sequences of raw text prepended with control codes. Style domain, URLs, specific task indicators, and combinations of those have been tried as control codes for text generation, and yielded promising results. Some examples of conditional text generation via CTRL are shown in Appendix B.

Readability Scores

Readability is the ease with which a reader can understand a written text. In natural language, the readability of text depends on its content and its presentation. A popular readability judging criteria, flesch reading ease, is proposed by Flesch (1948), giving the provided text a score between 1 and 100, with 100 being the highest readability score. The higher the score is, the more readable the text is. A flesch reading ease score between 70 and 80 is roughly equivalent to school grade 7 to 8. Some examples are shown in Table 1. Other judging criterias include SMOG index (Mc Laughlin, 1969), Coleman–Liau index (Coleman and Liau, 1975), Automated readability index (Senter and Smith, 1967), etc, all approximating the U.S. grade level thought necessary to comprehend the text.

3 Method

We leverage RAG (Lewis et al., 2020) with careful adjustments to its retriever and generator parts, to adapt to the extra input of readability score and generate text accordingly. For the general structure, we use a typical RAG-Sequence Model, with its encoder unchanged. We then modify the retriever so that the passages it retrieves, besides a similarity check, has the closest readability scores with the target one. For the generator part (i.e. BART part), we prepend a special token representing the readability score to the input, so that it can adapt to this special constraint.

To train the model, we perform pre-training over the generator of RAG first, i.e. the BART model. We use the original BART pretraining strategy, but prepend a special token indicating the readability score to the sentence to be reconstructed. Then we train our RC-RAG model end-to-end, where we provide it with a question and the answer's readability score as source, and the answer as target.

3.1 Readability-Controlled BART

BART (Lewis et al., 2019) uses the standard sequence-to-sequence transformer architecture from Vaswani et al. (2017), and modify the **ReLU** activation to **GeLU**s (Hendrycks and Gimpel, 2016). A BART-large model is the normal choice of generator in RAG, and is fed with the documents retrieved, together with the encoded question, to generate an answer.

To further pre-train BART in a conditional behavior akin to CTRL, we prepend a special token indicating the sentence's readability score to each sample sentence. An illustration of how we obtained training samples for pretraining BART is shown in Figure 2. We then do standard *token masking* pre-training, based on the already pretrained BART, as a fine-tuning process towards the conditional end. Note that the special token indicating readability scores also has a probability to get masked in the training process, so as to train to model to pay attention to this score token and perform reconstruction with this information as well.

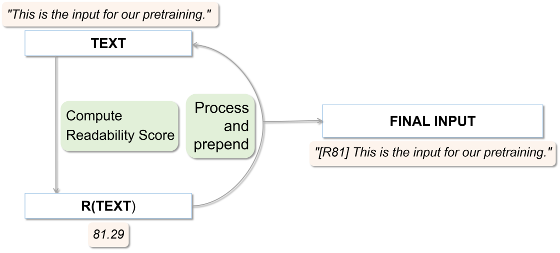
"This is the input for our pretraining."
TEXT
Compute Readability Score
Process and prepend
FINAL INPUT
"[R81] This is the input for our pretraining."
R(TEXT)
81.29

Figure 2: Illustration of how one sentence is processed before feeding to Readability-Controlled BART.

Table 1: Examples of flesch-reading-ease score

Text	RScore
Neo-virology is an emerging field engaged in cataloguing and characterising this biodiversity through a global consortium.	4.47
They act to stabilize the activation of aspects of the innate immune response and prevent excessive inflammation.	28.84
Most patients with A(H7N9) infections had contact with poultry or visited live animal markets.	48.81
All vaccine candidates for SARS and MERS were reported to be safe.	76.22
The time is ranging between 0 and 10 days with a mean of 3.7 days.	90.09

3.2 Readability Controlled RAG

To condition RAG on readability score, two modifications are made to the original structure.

Retriever

RAG uses DPR to retrieve documents from its knowledge base. DPR follows a bi-encoder architecture and indexes all the passages from the knowledge base in a low-dimensional and continuous space, tso retrieve efficiently the top k passages relevant to the input question. DPR uses a dense encoder to vectorize all the documents and index them for retrieval. Another dense encoder encodes the transmitted query and recalls top k articles as approximate nearest neighbors to this vector.

Instead of a database constructed from a full Wikipedia dump, we establish a custom database of COVID-related articles. To make it adaptable to the readability-controlled generation, we first fetch documents by MIPS, then select the documents whose readability score have a smaller gap with the target one. To be more specific, we first fetch top-$3k$ documents by MIPS, then compute a weight sum of the inner product score by MIPS and the opposite of readability score gap of the document, to fetch the top-k documents base on this sum. After retrieval, doc scores used for generation do not take readability score into consideration.

Conditional Generator

We change the generator of RAG to the Conditional BART model previously mentioned, which could generate text based on given documents and a target readability score. We add readability scores as special tokens at the beginning of each sentence to indicate how easy to read the answer should be. Note that besides having the benefit of being simple to implement as a plug-and-play method, this parametrization also has the advantage of letting the attention mechanism capture dependencies between readability score and sentence tokens.

4 Experiments

4.1 Datasets

4.1.1 Pre-training & Knowledge Base

We use the same dataset for pre-training BART and the external knowledge-base of RAG. We chose CORD-19 (Wang et al., 2020), together with a curated version of a Wikipedia dump updated on May 2, 2021, and several pages of contents on WHO official website as our knowledge source.

• For the Wikipedia dump part, we use *WikiExtractor* (Attardi, 2015) to extract text from the Wikipedia dump. To make the training more efficient and consistent with our presets, we curated the dataset by retaining only Wikipedia articles containing at least one of a list of 18 COVID-related keywords such as “COVID”, “SARS-CoV-2”, “mask” and others (see Appendix A).

• CORD-19 (Wang et al., 2020) is a dataset of scientific papers relating to COVID-19. it is a free resource of tens of thousands of academic articles about COVID-19, SARS-CoV-2, and related coronaviruses for use. CORD helps us to gain a large amount of medical knowledge about such viruses.

• WHO official website[2] offers critical information about COVID-19. The content from its official website counts for a minor proportion in our knowledge base(about 0.1%), and is thus excluded from the following analysis.

We summarize the readability score distribution of sentences in our knowledge base in Figure 4. Both CORD and Wikipedia hold a diversity of readability scores, while as a scientific-paper-based dataset, CORD has more sentences with lower

[2] https://www.who.int/emergencies/diseases/novel-coronavirus-2019

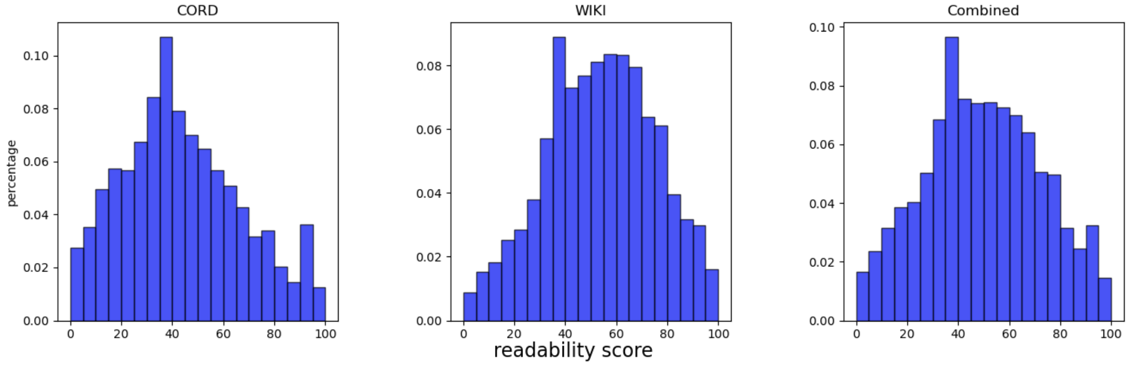

Figure 3: Scores histogram of Wikipedia and CORD.

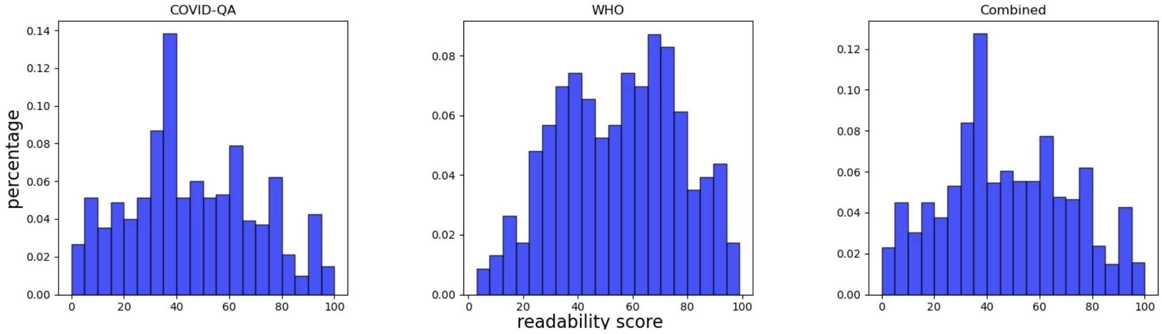

Figure 4: Scores histogram of QA pairs.

scores. The combination of them shows a diversity of readability levels, with the richest at about 40-70, which has "fairly difficult" or "standard" readability.

For pre-training BART, we see each sentence in this dataset as a separate sample, and do pre-processing as mentioned in Section 3.1. The knowledge base for RAG to retrieve is constructed by the standard way mentioned in Lewis et al. (2020), where each passage is a 100-word split from the original text.

4.1.2 Question-Answer Pairs

Two data sources are combined to make our question-answering dataset.

• COVID-QA (Möller et al., 2020). COVID-QA is a Question Answering dataset consisting of 2,019 question/answer pairs annotated by volunteer biomedical experts on scientific articles related to COVID-19. A total of 147 scientific articles from the CORD-19 dataset were annotated by 15 experts.

• WHO QA pairs. WHO official website[3] provides answers to all kinds of questions that might be asked by people concerning their life during the COVID pandemic. However the answers are generally too long and contains much extension beyond the question asked, and therefore cannot be used for question-answering directly. We remove irrelevant pairs and shorten/summarize the answers manually to get a total of 271 valid QA pairs[4], which contains mostly useful information.

We also plot the readability score histogram of answers in our QA-pairs dataset. The score of answers are computed as a whole, without separating them into single sentences. COVID-QA takes the most part of our obtained dataset, with a high proportion of hard-to-read answers whose scores range from 30 to 40. WHO QA pairs have a almost-even distribution in the range of 30 to 80, and could serve as a compensation of COVID-QA's lack of easy-to-read answers. Due to the gap in scale, their combination still has more answers with low readability scores, but we can leverage this dataset to tune our RC-RAG model for a wider range of readability scores.

[3] https://www.who.int/emergencies/diseases/novel-coronavirus-2019/question-and-answers-hub

[4] `https://github.com/...`, released upon acceptance.

Table 2: Examples of answers generated by original RAG.

Question	Answer
How can I reduce my risk of getting COVID-19?	Wash your hands frequently, cough or sneeze into a bent elbow or tissue.
What is lockdown?	regulations/legislations regarding strict face-to-face social interaction.
What may viral infections of the respiratory epithelium by viruses such as IFV, RV, RSV and HSV do?	may trigger the further production of ROS as an antiviral mechanism.

4.2 Training configurations

4.2.1 RC-BART Pretraining

We add additional special tokens as [R1], [R2], etc, where the number here indicates the round of the sample's readability score. We apply the pre-training strategy of token masking, with each token begin given a 15% chance to be masked, *including the readability special token*. To obtain each training sample, we separate each sentence in the knowledge base we obtained. The readability score of a given sentence is computed using the python package *textstat*[5]. We use a per device batch size of 2 to train it on 4 15GB GPU, with AdamW (Loshchilov and Hutter, 2018) using a learning rate of 5e-5. We did a full 1 epoch of training in total.

4.2.2 Fine-tuning RC-RAG

We feed extra input of target readability score to RAG, and in the forwarding process, the score is passed to the retriever to get a jointly-sorted set of documents. We set the number of final retrieved documents as 5. After retrieving the documents, we prepend a special token representing the target readability as mentioned in Section 4.2.1 to the generator's input. The model is trained end-to-end using our QA pairs dataset, using a 3.7GB knowledge base, which, after processing results in 22GB for faiss index and 20GB for passages. We use a batch size of 1 on each 15GB GPU, with a total number of 4 GPUs to train our model. The training goes for 120 epochs in total. We use code provided on transformers (Wolf et al., 2020) to train our model.

4.3 Results and Analysis

4.3.1 RC-BART

There are currently no widely accepted way to measure how well-conditioned one model is on readability score, so we evaluate our model's performances by calculating the sum of squares of differences between the input (unmasked) sentence's and the reconstructed sentence's flesch-reading-ease score, namely:

$$\mathcal{L} = \frac{1}{n}\sum_{i}^{n}(\text{Readability\_Score}(\mathbf{S}_{\text{in}}) - \text{Readability\_Score}(\mathbf{S}_{\text{reconstructed}}))^2. \tag{1}$$

As the training proceeds, we observe a decreasing trend of $\mathcal{L}$, from an initial value of about 134 to an ending point around 80. Figure 5 depicts this trend during one epoch of training. This indicates that RC-BART is learning to reconstruct the sentence with a similar readability score. Note that sentences within a flesch-reading-ease score gap of 10 are often considered to have similar readability.

We also tested RC-BART's capability to fill masks by computing the top-1 accuracy of the filled token. We compare it with a BART model further pre-trained on our dataset without adding readability scores, using exactly the same settings. The accuracy reported is 52.06% versus 53.48%, where RC-BART suffers a small setback due to the difficulty of reconstructing the readability special token. Note that the reconstruction performance can still be considered to be on-par, despite the greater difficulty of the new task. Furthermore, we believe that with enough computational resources this tiny gap could be resolved eventually.

4.3.2 RC-RAG

We did fine-tuning on an original version of RAG as well as the proposed RC-RAG. After fine-tuning, both of them are able to provide reasonable answers for COVID-related questions. A few example answers generated by original RAG model after fine-tuning is shown in Table 2. This shows RAG's natural ability to adapt to different domains, if provided with a rich knowledge base containing the information needed. We report a EM (exact match) ratio of 23.4% for original BART, while for RC-RAG this ratio is 22.0% when using the same read-

[5] https://pypi.org/project/textstat/

Table 3: Examples of readability-controlled question-answering.

Question	Target Score	Answer	Actual Score
How does PEDV spread?	36	fecal-oral contact	35.61
	66	from person to person	75.88
Gemcitabine has been shown to have antiviral activity against which viruses?	20	Middle East respiratory syndrome coronavirus (MERS-CoV), severe acute respiratory syndrome coronavirus (SARS-CoV), Zika virus (ZIKV)	5.49
	50	citabine has been shown to protect against several influenza A virus	68.77
What precautions should everyone take in a hotel or other accommodation establishment?	80	Wash your hands frequently. Maintain at least a 1 metre distance from staff and other.	80.78
	40	hands should be frequently cleaned using either alcohol-based sanitizers or disinfect	18.01
How can I reduce my risk of getting COVID-19?	46	Avoid crowded places, poorly ventilated, indoor locations and avoid prolonged contact with others.	41.36
	76	Wash your hands frequently and thoroughly with soap and water and dry them thoroughly.	65.73
What is required to establish a secondary immune response to a viral infection?	12	innate immune responses and the generation of antigen-specific "memory" B and T lymphocytes	7.52
	52	B and T lymphocytes	75.88
What is hepatitis C?	48	positive-sense single stranded RNA enveloped virus	48.47
	78	virus	100

[1] for each question, score in the first row indicates the readability score of its ground truth answer, and score in the second row (marked red) is the tuned target score for conditional testing

[2] Some extremely simple sentences(e.g. sentence with one single word) might have a score greater than 100. In this case we simply write 100 to indicate its extreme simplicity.

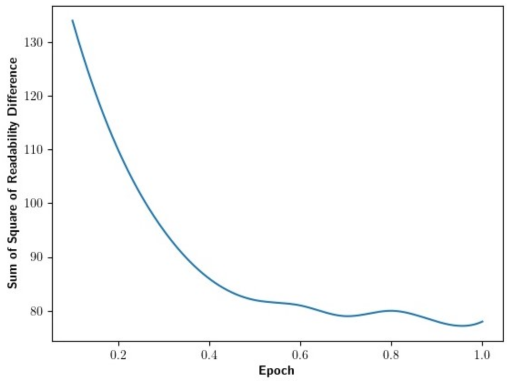

Figure 5: Performance of RC-BART in 1 epoch.

ability score of the target sentence as input target score. This shows that despite the additional goal of reading-score matching of RC-RAG – which makes the task harder – it is able to preserve on-par performance with RAG on the original QA task.

We then evaluate the conditional performance of RC-RAG. To do so, we choose some questions from the dataset, first using the readability score of its ground truth answer as input to check its generated answer, and then change the target readability score to see what happens. Some questions, naturally, don't really possess answers with different readability levels, especially those real professional and in-depth questions on specific medical topics, or those regarding a certain, unshakable fact. Furthermore, questions such as "Could the virus be transmitted from humans to food animals or vice versa?" don't have confirmed answers yet, so its answer "No evidence so far" couldn't be used to tune readability scores either.

For questions that possess answers with a range of readability levels, we sample a few answers generated by our model and present examples in Table 3. This shows our model's capability to generate a correct, understandable answer that has the smallest gap against the target readability score. Extra examples are provided in Appendix C.

5 Related Work

Retrieval method for Open-domain QA

Recent development of retrieval methods has helped open domain QA system in getting improvements. Karpukhin et al. (2020) proposed a new dense retriever which is different from other conventional sparse retrieval methods such as TF-IDF (Aizawa, 2003) and large pretrained model (Nogueira and Cho, 2019) for ranking retrieved passages and documents. Our work combines retrieved document score from RAG and readability score together to rank the passage and finetune single retrieval-based architecture using our QA pairs, achieving strong performance.

Generative QA

Generative QA generates answers typically with a seq2seq model instead of extracting answers from large corpora. For question generation, Lewis and Fan (2018) introduce generative models leveraging the joint distribution between questions and answers to explain the target question. The approach Heilman and Smith (2010) used is to generate questions and then rank them statistically. Duan et al. (2017) use neural networks to generate questions from given passages. There are also tasks aiming to generate question-answer pairs such as the work of Sachan and Xing (2018), who trained a model from unlabelled text. These generative models show great robustness to biased training data. Our work focuses on generating answers according to an target readability score as control code, and shows different controlled generated outputs according to the chosen score of the user.

Conditional Text Generation

Conditional Text Generation (CTG) is the task of generating text according to some pre-specified conditioning. Supervised (Wang and Wan, 2018; Sohn et al., 2015) as well as semi-supervised methods (Hu et al., 2017) have been proposed for this task. Significant studies have been conducted in CTG since the release of CTRL (Keskar et al., 2019). Xia et al. (2020) used CG-BERT to leverage a large pre-trained language model to generate text conditioned on the intent label. Duan et al. (2019) used PPVAE in flexible conditional text generation. Chen et al. (2020) used CTGAN which can generate diverse text content of variable length with customizable emotion labels. In our work, we prepend special tokens that represent the sentence's readability to a BART model directly, which differs from PPVAE and CG-BERT.

COVID-19 related contributions

COVID-19 broke out on December 2019, and quickly spread around the world. In an effort to end the pandemic, a significant amount of research has been produced about COVID-19, including its origin (Morens et al., 2020; Zhang et al., 2020), treatment (Felsenstein et al., 2020; Beigel et al., 2020), vaccine (Graham, 2020), etc.
There also exist Q&A systems designed specifically for COVID-19 related tasks which are document-based, such as a real-time question answering (QA) and multi-document summarization system (Su et al., 2020), and a web understanding Q&A (Zhang et al., 2021).

6 Discussion

In this paper, we present readability controlled RAG, a retrieval augmented generation model which is specialized to do COVID-related open domain question-answering tasks, with the ability to generate answers for different readability levels. We showed that with the help of a readability-controlled BART as generator, our RC-RAG model could generate answers with different reading ease well, and is usable by people of different educational backgrounds. In future work, it would be nice to consider encoding also the question, as well as the documents to be retrieved, to perform a retrieval process with better readability control. Also, with a larger knowledge set and richer computation resources, this model could be further applied beyond only COVID-related areas, and make more contributions to people from all walks of live. We also point out that certain textual features, such as readability, could be used in Seq2Seq models for controlled generation, which is a promising direction of future efforts. We look forward to further explorations of our model.

References

Akiko Aizawa. 2003. An information-theoretic perspective of tf–idf measures. *Information Processing & Management*, 39(1):45–65.

Giuseppe Attardi. 2015. Wikiextractor. https://github.com/attardi/wikiextractor.

John H Beigel, Kay M Tomashek, Lori E Dodd, Aneesh K Mehta, Barry S Zingman, Andre C Kalil, Elizabeth Hohmann, Helen Y Chu, Annie Luetkemeyer, Susan Kline, et al. 2020. Remdesivir for the treatment of covid-19. *New England Journal of Medicine*, 383(19):1813–1826.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*.

Jinyin Chen, Yangyang Wu, Chengyu Jia, Haibin Zheng, and Guohan Huang. 2020. Customizable text generation via conditional text generative adversarial network. *Neurocomputing*, 416:125–135.

Meri Coleman and Ta Lin Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283.

Nan Duan, Duyu Tang, Peng Chen, and Ming Zhou. 2017. Question generation for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 866–874.

Yu Duan, Canwen Xu, Jiaxin Pei, Jialong Han, and Chenliang Li. 2019. Pre-train and plug-in: Flexible conditional text generation with variational autoencoders. *arXiv preprint arXiv:1911.03882*.

Susanna Felsenstein, Jenny A Herbert, Paul S McNamara, and Christian M Hedrich. 2020. Covid-19: Immunology and treatment options. *Clinical Immunology*, page 108448.

Rudolph Flesch. 1948. A new readability yardstick. *Journal of applied psychology*, 32(3):221.

Barney S Graham. 2020. Rapid covid-19 vaccine development. *Science*, 368(6494):945–946.

Michael Heilman and Noah A. Smith. 2010. Good question! statistical ranking for question generation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, page 609–617, USA. Association for Computational Linguistics.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *International Conference on Machine Learning*, pages 1587–1596. PMLR.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

Mike Lewis and Angela Fan. 2018. Generative question answering: Learning to answer the whole question. In *International Conference on Learning Representations*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Patrick Lewis, Ethan Perez, Aleksandara Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *arXiv preprint arXiv:2005.11401*.

Ilya Loshchilov and Frank Hutter. 2018. Fixing weight decay regularization in adam.

G Harry Mc Laughlin. 1969. Smog grading-a new readability formula. *Journal of reading*, 12(8):639–646.

Timo Möller, Anthony Reina, Raghavan Jayakumar, and Malte Pietsch. 2020. Covid-qa: A question answering dataset for covid-19. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*.

David M Morens, Joel G Breman, Charles H Calisher, Peter C Doherty, Beatrice H Hahn, Gerald T Keusch, Laura D Kramer, James W LeDuc, Thomas P Monath, and Jeffery K Taubenberger. 2020. The origin of covid-19 and why it matters. *The American journal of tropical medicine and hygiene*, 103(3):955–959.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with bert. *arXiv preprint arXiv:1901.04085*.

Mrinmaya Sachan and Eric Xing. 2018. Self-training for jointly learning to ask and answer questions. 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 629–640.

RJ Senter and Edgar A Smith. 1967. Automated readability index. Technical report, CINCINNATI UNIV OH.

Minjoon Seo, Jinhyuk Lee, Tom Kwiatkowski, Ankur P Parikh, Ali Farhadi, and Hannaneh Hajishirzi. 2019. Real-time open-domain question answering with dense-sparse phrase index. *arXiv preprint arXiv:1906.05807*.

Kihyuk Sohn, Honglak Lee, and Xinchen Yan. 2015. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28:3483–3491.

Dan Su, Yan Xu, Tiezheng Yu, Farhad Bin Siddique, Elham J Barezi, and Pascale Fung. 2020. Caire-covid: A question answering and multi-document summarization system for covid-19 research. *arXiv preprint arXiv:2005.03975*.

Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William W Cohen. 2018. Open domain question answering using early fusion of knowledge bases and text. *arXiv preprint arXiv:1809.00782*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Ke Wang and Xiaojun Wan. 2018. Sentigan: Generating sentimental texts via mixture adversarial networks. In *IJCAI*, pages 4446–4452.

Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Darrin Eide, Kathryn Funk, Rodney Kinney, Ziyang Liu, William Merrill, et al. 2020. Cord-19: The covid-19 open research dataset. *ArXiv*.

Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerry Tesauro, Bowen Zhou, and Jing Jiang. 2018. R 3: Reinforced ranker-reader for open-domain question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Congying Xia, Chenwei Zhang, Hoang Nguyen, Jiawei Zhang, and Philip Yu. 2020. Cg-bert: Conditional text generation with bert for generalized few-shot intent detection. *arXiv preprint arXiv:2004.01881*.

Tao Zhang, Qunfu Wu, and Zhigang Zhang. 2020. Probable pangolin origin of sars-cov-2 associated with the covid-19 outbreak. *Current biology*, 30(7):1346–1351.

Yuan Zhang, Xiaoqing Zhang, Yichuan Hu, Guanchun Wang, and Rui Yan. 2021. Wulai-qa: Web understanding and learning with ai towards document-based question answering against covid-19. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, WSDM '21, page 898–901, New York, NY, USA. Association for Computing Machinery.