Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Sheng-Chieh Lin, Jheng-Hong Yang, Rodrigo Nogueira, Ming-Feng Tsai, Chuan-Ju Wang, and Jimmy Lin. 2020. Query reformulation using query history for passage retrieval in conversational search. *CoRR*, abs/2005.02230.

Shayne Longpre, Yi Lu, Zhucheng Tu, and Chris DuBois. 2019. An exploration of data augmentation and sampling techniques for domain-agnostic question answering. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 220–227.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *Association for Computational Linguistics*, pages 280–290.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Chen Qu, Liu Yang, Cen Chen, Minghui Qiu, W. Bruce Croft, and Mohit Iyyer. 2020. Open-retrieval conversational question answering. In *SIGIR 2020: 43rd international ACM SIGIR conference on Research and Development in Information Retrieval*.

Jun Quan, Deyi Xiong, Bonnie Webber, and Changjian Hu. 2019. Gecor: An end-to-end generative ellipsis and co-reference resolution model for task-oriented dialogue. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 4547–4557.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. Coqa: A conversational question answering challenge. *TACL*, 7:249–266.

Pengjie Ren, Zhumin Chen, Zhaochun Ren, Evangelos Kanoulas, Christof Monz, and Maarten de Rijke. 2020. Conversations with search engines.

Marzieh Saeidi, Max Bartolo, Patrick Lewis, Sameer Singh, Tim Rocktäschel, Mike Sheldon, Guillaume Bouchard, and Sebastian Riedel. 2018. Interpretation of natural language rules in conversational machine reading. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2087–2097.

Svitlana Vakulenko, Shayne Longpre, Zhucheng Tu, and Raviteja Anantha. 2020. Question rewriting for conversational question answering. *CoRR*, abs/2004.14652.

Nikos Voskarides, Dan Li, Pengjie Ren, Evangelos Kanoulas, and Maarten de Rijke. 2020. Query resolution for conversational search with limited supervision. In *SIGIR 2020: 43rd international ACM SIGIR conference on Research and Development in Information Retrieval*.

Nick Webb, editor. 2006. *Proceedings of the Interactive Question Answering Workshop at HLT-NAACL 2006*. Association for Computational Linguistics, New York, NY, USA.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019a. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019b. Transfertransfo: A transfer learning approach for neural network based conversational agents. *CoRR*, abs/1901.08149.

Peilin Yang, Hui Fang, and Jimmy Lin. 2017. Anserini: Enabling the use of lucene for information retrieval research. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1253–1256.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-end open-domain question answering with bertserini. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 72–77.

A Reproducibility

A.1 Training Transformer++ for Question Rewriting

Details about training setup of Transformer++ for question rewriting task is provided in Table 7. The Transformer head is initialized with the pretrained weights of GPT-2 (medium) and further fine-tuned on the QReCC train set. We use PyTorch implementation from HuggingFace.[3] Transformer++ is trained using model parallelism on 5 Tesla V100 GPUs with hyperparameter search trial.

A.2 Building Document Collection

Here we provide further details for building the document collection. If the web page of the provenance link containing the answer was not archived by the Wayback Machine yet, we trigger the archiving through the Wayback Machine API whenever possible. Overall, 2% of the annotated web pages could not be archived by the Wayback Machine due to the restricted access (such as the Quora website).

For the Common Crawl data, we take the index files from November 2019 and filter URLs to only those that are retrieved with HTTP status code 200 and those that are identified as English. We extract the pages from the Common Crawl WET files that correspond to these filtered URLs, and sample the first link out of every 100 links in each filtered WET file.

Overall, we find that 97.8% of unique web pages found by human annotators to contain answer and has an associated archived copy on the Wayback Machine. The final collection consists of both these pages from the Wayback Machine and random web pages from the Common Crawl.

After downloading the pages we extract all text from the page using the Beautiful Soup library.[4] We iterate through the web page by newlines, and accumulate the tokens for every line. Whenever the number of tokens reaches 220 or more, we emit a paragraph, and reset the token counter to 0. Note the last paragraph on the page may have fewer than 220 tokens. After segmentation, we have a total of 54,241,550 passages which we index using Pyserini 0.10.0.1. Hence we treat each passage as a single document.

[3] https://github.com/huggingface/transformers

[4] https://www.crummy.com/software/BeautifulSoup/bs4/doc/

A.3 Training BERT-L for Reading Comprehension

Below we provide details about the training setup for BERT-L used of the reading comprehension task in our experiments, which is similar to the extractive reader setup in Longpre et al. (2019) but using BERT-L. We train on the full data of the QReCC training set, using Human rewritten questions. Our implementation of the BERT question answering modules follows that of the standard PyTorch (Paszke et al., 2019) implementations from HuggingFace, and are trained on 4 NVIDIA Tesla V100 GPUs. The model is trained to predict an answer span or abstain if the passage has "No Answer". For every query we obtain up to 25 paragraphs from the document that contains the gold answer as identified by a human grader. The paragraph with the answer is always used for training, and a portion of the other paragraphs are used in training as No Answer or "negative" examples. Using the development set we tune several hyperparameters, most importantly the percentage of negative examples to retain for training ("Pct Neg. Ratio"). Fixed parameters and tuning details are shown in Table 8.

B Annotation Guidelines

Instructions for question rewriting:

- Rewritten questions should be as close to the original as possible.
- Questions should not contain any references to the previous context of the conversation.
- Avoid using any pronouns in question rewrites.

Instructions for answering questions:

- Put the rewritten question (original question if it is already self-contained) in a web search engine to produce the correct answer.
- Produce an answer, which should be short and brief with minimum information required to answer the question.
- The answers should be grammatically correct, do not contain special symbols or any additional mark-up.
- Produce an answer that would be most natural for a human conversation.

Table 7: Hyperparameter selection and tuning ranges for TRANSFORMER++ used for question rewriting.

MODEL PARAMETERS	VALUE/RANGE
Fixed Parameters	
Batch Size	16
Optimizer	Adam
Vocabulary Size	150,263
Transformer Head	GPT-2 (medium)
Learning Rate Schedule	Exponential Decay
Output Attention	True
Max Input Sequence Length	1024
Max Output Sequence Length	30
Num Hyperparameter Search Trials	500
Tuned Parameters	
Num Epochs	[50, 100]
Initializer Range	[0.01, 0.1]
Dropout	[0.05, 0.2]
Attention Dropout	[0.05, 0.1]
Residual Dropout	[0.05, 0.1]
Learning Rate	$[1e-3, 1e-1]$
Decay Steps	[6000, 10000]
Decay Rate	[0.7 , 0.9]
Activation Functions	[ReLU, Leaky ReLU, GELU]
General	
Model Size (# params)	350M
Avg. Train Time (per epoch)	12 hours

Table 8: Hyperparameter selection and tuning ranges for BERT-L used for reading comprehension.

MODEL PARAMETERS	VALUE/RANGE
Fixed Parameters	
Batch Size	32
Optimizer	Adam
Learning Rate Schedule	Exponential Decay
Num Epochs	2
Max Input Sequence Length	512
Max Span Length	30
Num Hyperparameter Search Trials	32
Tuned Parameters	
Learning Rate	$[1e-5, 5e-5]$
Pct Neg. Ratio	[0.01, 0.5]
General	
Model Size (# params)	330M
Avg. Train Time (per epoch)	8 hours