Retrieval Data Augmentation Informed by Downstream Question Answering Performance

Anonymous ACL submission

Abstract

Training retrieval models to fetch contexts for Question Answering (QA) over large corpora requires labeling relevant passages in those corpora. Since obtaining exhaustive manual annotations of all relevant passages is not feasible, prior work uses text overlap heuristics to find passages that are likely to contain the answer, but this is not feasible when the task requires deeper reasoning and answers are not extractable spans (e.g.: multi-hop, discrete reasoning). We address this issue by identifying relevant passages based on whether they are useful for a trained QA model to arrive at the correct answers, and develop a search process guided by the QA model's loss. Our experiments show that this approach enables identifying relevant context for unseen data greater than 90% of the time on the IIRC dataset and generalizes better to the end QA task than those trained on just the gold retrieval data on IIRC and QASC datasets.

1 Introduction

Answering questions over a large text corpus typically requires retrieving information relevant to the question from the corpus, which is then used by a Question Answering (QA) model to arrive at the answer. Recent work (Guu et al., 2020; Lewis et al., 2020; Ni et al., 2020) relies on retrieval models that learn dense representations of questions and retrieval candidates (Karpukhin et al., 2020; Khattab and Zaharia, 2020) trained separately or jointly with the QA model. These learned retrieval models are more effective than those that use simple word overlap signals (Robertson and Zaragoza, 2009; Chen et al., 2017), but they require the positive retrieval targets for each question labeled. It is often difficult, if not impossible, to exhaustively label all the facts relevant to answering a question in a large corpus of text. Consequently, even when the datasets provide retrieval labels, it is often the case that there exist alternative paths to the answer that are not labeled (Jhamtani and Clark, 2020), an example of which is shown in Figure 1. The common heuristic of considering all contexts that contain mentions of the answer span (Clark and Gardner, 2018; Lee et al., 2019a) does not work when the QA task is not extractive (e.g.: when the answers are binary or require some numerical computation).

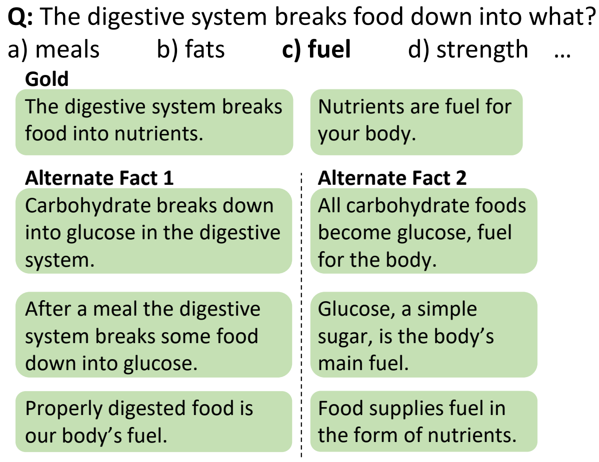


Figure 1: Retrieval annotations (gold) are often incomplete, only providing one of many relevant contexts. Alternative contexts can provide different views of the same information, providing more robust training data.

We propose to address this issue by augmenting the set of labeled retrieval targets with additional candidates that are not labeled as positive, but still provide sufficient information to answer the corresponding questions. Given question-answer pairs, and a QA model trained to maximize the likelihood of the correct answers conditioned on the labeled retrieval targets and the questions, we search for alternative contexts that also make the correct answers likely. Concretely, our search process finds those contexts not labeled as gold, that minimize the loss of the QA model. We consider these contexts as alternative retrieval targets, and train the retrieval model with the combination of these alternative contexts and the gold labeled contexts as positives. Our method is particularly effective for non-extractive QA tasks since it does not rely on answer-span overlaps.

We evaluate our approach on two multi-hop QA tasks, IIRC (Ferguson et al., 2020) and QASC (Khot et al., 2019), and show that our search for relevant contexts guided by the performance of the QA model correctly identifies a relevant context 91% of the time on IIRC and 84% of the time on QASC (Table 2a). Augmenting the retrieval training data with the results from our search process increases recall on unseen questions, leading to an improvement in the downstream QA performance by 0.5 F_1 points on IIRC and 2.1 accuracy points on QASC (Section 3.2).

2 Method

Overview and Problem Our approach uses the standard two-step pipeline for open-domain QA seen in prior work. We first run a retrieval model that takes as input a question, q, and a large corpus of passages, C, and outputs a small subset of those passages, $c \subset C$, that contains sufficient information to answer the question. This subset is then passed to the second step: the QA model. This model takes as input the same question, q, and subset of passages, c, from the first step, and outputs an answer, a. Depending on the data, this answer can take many forms, such as a span from the context, a number, yes/no, or none of these if the question is unanswerable.

For each question, there may be many valid sets of context passages, where each set[1] contains all the information necessary to answer the question. We refer to individual sets as c_i^*, and the superset of all such sets as $c^* = \{c_1^* \dots c_n^*\}$. As seen in Figure 1, these different context sets may express different reasoning paths reaching the answer, or they may contain different ways of expressing the same reasoning path. However, most datasets just contain annotations of one such set per question, c_i^*. Our goal is to use these annotations to identify alternate, unannotated, relevant context, $\bar{c} \in c^* \setminus \{c_i^*\}$, for each question. These additional contexts is used to augment the retrieval training data.

Approach The goal of the retrieval model is to identify context that maximizes the probability of the correct answer when given to the QA model. When supervised data, c_i^*, is available, this is achieved by training the retrieval model to predict the input that the QA model is trained on i.e., $\theta_r = \arg\max_\theta P(c_i^*|q,\theta)$, and $\theta_q = \arg\max_\theta P(a|q,c_i^*,\theta)$, where the retriever and the QA models are parameterized by θ_r and θ_q. We refer to this initial QA model as the *base* QA model. When supervised data is not available, we can identify the retrieved contexts $\hat{c}$, by searching over the corpus for the contexts that maximize the probability of the correct answer under the base QA model:

$$\hat{c} = \arg\max_{c \subset C} P(a|q,c,\theta_q) \quad (1)$$

Based on this, for each question, we search over the corpus for the top k contexts, $\hat{c}_1 \dots \hat{c}_k$, and add them as additional data augmentation when training a new retrieval model:

$$\hat{\theta_r} = \arg\max_\theta P(c_i^*|q,\theta) + \sum_{j=1}^{k} P(\hat{c}_j|q,\theta) \quad (2)$$

Lastly, we train a final QA model using the gold context, including the results of this new retrieval model to incorporate the updated training and make it more robust to noise:

$$\begin{aligned} c_r &= \arg\max_{c \in C} P(c|q,\hat{\theta_r}) \\ \hat{\theta_q} &= \arg\max_\theta P(a|q,\{c_i^*, c_r\},\theta) \end{aligned} \quad (3)$$

Labeling sets of facts Because we apply our approach to datasets containing questions that require multiple facts to answer, we need to label *sets* of facts, not individual ones. For this reason, we train our base QA models conditioned on sets of facts, and while both labeling new contexts with the base QA model, and retrieving contexts, we use beam search to output sets of facts. In order to prevent the base QA model from memorizing the gold contexts, we use a 10-fold cross-labeling approach.[2]

3 Experiments

We show the effect of our approach on two multi-hop QA datasets: IIRC (Ferguson et al., 2020) and QASC (Khot et al., 2019).

3.1 Datasets and Setup

IIRC is a multi-hop QA open QA dataset, consisting of a mix of yes/no questions, span selection questions, unanswerable questions, and questions

[1] We apply our approach to datasets containing questions that require multiple facts to answer, so we label *sets* of facts.

[2] We train ten models, each on 90% of the data, and use them to label the remaining 10%.

requiring discrete reasoning such as arithmetic or counting. Each question is associated with a paragraph, and requires both information from that paragraph, as well as information from one or more pages linked to from within that paragraph.

QASC is a multiple-choice, multi-hop QA dataset constructed from a corpus of 17M facts. Each question is written by composing two facts from the corpus, and includes eight answer choices.

eQASC (Jhamtani and Clark, 2020) includes a more exhaustive annotation of relevant contexts for QASC questions and enables a more accurate evaluation of retrieval performance on QASC.

Evaluation We report recall@10 and the final QA performance results that provide a more reliable evaluation of the retrieval performance. For eQASC, we use mean-average precision (MAP) of the positive examples.

Implementation Details Following prior work on IIRC (Ni et al., 2020), we adopt a pipeline approach consisting of three steps: link selection using RoBERTa-base, retrieval, and answer selection using NumNet++ (Ran et al., 2019). For QASC, we initially filter the corpus using the two-step BM25 described in (Khot et al., 2019), selecting the top 1000 pairs of facts per answer choice. Similar to IIRC, we then select the top 10 pairs using a RoBERTa-base bi-encoder. Final QA model separately scores each answer choice using another RoBERTa-base model, and computes a softmax to get the final distribution over the choices.

3.2 Comparisons and Results

We compare our approach of identifying additional relevant context using QA loss with other retrieval baselines and alternate augmentation methods.

BM25: We use the top results from BM25 in lieu of training a supervised model with the annotated data. This is a commonly used heuristic when no retrieval annotations are available.

$\mathrm{Sup}_{\mathrm{A}}$ Models are trained using just the annotated training data with no additional data provided.

$\mathrm{Sup}_{\mathrm{A+BM25}}$ We augment the annotated training data with the top results from querying the corpus using BM25 with the question and answer.

$\mathrm{Sup}_{\mathrm{A+R}}$ We augment the annotated training data with the top retrieval results conditioned on the question and correct answer. As in the QA-loss labeling approach, we use a 10-fold labeling procedure to prevent memorizing the annotated context.

Approach	QASC		IIRC		eQASC
	R@10	Acc	R@10	F1	MAP
BM25	45.1	71.9	18.0	42.0	36.0
$\mathrm{Sup}_{\mathrm{A}}$	46.1	71.8	39.5	51.1	41.9
$\mathrm{Sup}_{\mathrm{A+BM25}}$	41.7	69.3	38.0	49.2	40.3
$\mathrm{Sup}_{\mathrm{A+R}}$	46.2	71.5	39.3	51.0	35.4
$\mathrm{Sup}_{\mathrm{A+QA}}$	**47.8**	**73.9**	**40.3**	**51.6**	**43.7**
Prior Work	-	71.9	-	50.6	-

Table 1: Comparison of different retrieval models. R@10 and MAP are direct evaluations of retrieval performance, Acc is the performance of the final QA model trained given retrieval results. For IIRC, prior work is the state-of-the-art model (Ni et al., 2020) that uses the same QA model as our work. For QASC, prior work is RoBERTa-base model that uses the same model size as ours and is trained and evaluated on the same data used by (Khashabi et al., 2020).

Main Results Table 1 compares our approach, $\mathrm{Sup}_{\mathrm{A+QA}}$, with the baselines and prior work.[3] Our approach results in improved performance on both datasets with a larger improvement on QASC over the baseline compared to IIRC. This is likely due to the fact that QASC has a much larger number of alternate contexts per question compared to IIRC (discussed below in oracle analysis). We generally see a correlation between retrieval recall of the gold annotations, performance on eQASC, and downstream accuracy, indicating that providing more accurate context to the downstream model does help with QA performance.

We manually labeled the accuracy of the top result for 100 questions for each approach (results in table 2a). We can see that using the QA model to label data significantly outperforms the other two approaches. In table 2b we also further break down the accuracy based on the different types of questions in IIRC. Our approach works well on *Binary* and *Numeric* questions, where the span heuristic cannot be applied. Our approach also outperforms the it on *Span Selection* questions, where the answer is a span from the context. Although the heuristic can be applied on these questions, it often returns false positives. Our approach struggles with *Span Compare* questions, as discussed in more detail in Error Analysis below.

Oracle Analysis Figure 2c shows an oracle study of the same 100 questions from the previous section to determine how many alternate contexts were available in each dataset. For IIRC, we considered

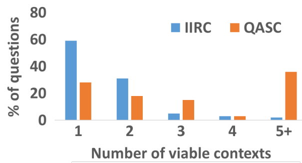
[3]The state-of-the-art model (Khashabi et al., 2020) for QASC uses roughly 100x more parameters than us (with the results 89.6), but the same model with a comparable size as ours is significantly worse, 50.8. Therefore, we use the best-performing model that has the same size as ours.

Approach	IIRC	QASC
BM25	38	41
Retrieval	39	45
QA Loss	91	84

(a)

Question type	QA	Span
Binary	100	-
Numeric	78	-
Span Selection	97	77
Span Compare	50	-

(b)



(c)

Table 2: (a) Manual analysis Accuracy of different approaches based on manual analysis on 100 examples for different context labeling approaches, (b) comparing span-selection retrieval baseline with our approach for different question types, and (c) Comparison of the number of relevant contexts in each dataset.

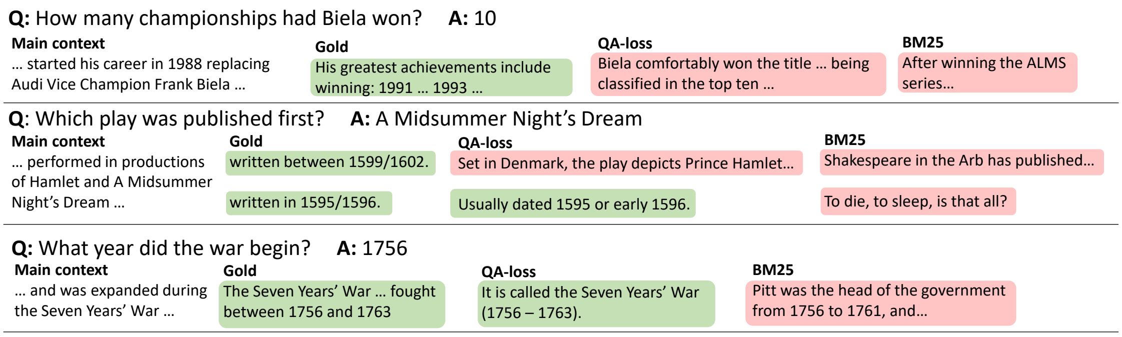


Figure 2: Example errors of our approach in IIRC. Relevant context is highlighted in green, and irrelevant context is in red.

all sentences from the gold articles, and for QASC we considered the top twenty sentences according to BM25. QASC has a much higher ceiling for this form of data augmentation, as can be seen by the fact that 70% of questions have multiple relevant contexts, compared to IIRC where many questions have only a single context. Additionally, many of the questions in IIRC with exactly 2 contexts share a similar structure, seen in the third example in Figure 2. Although our approach is often able to identify this alternate context, using it to augment the data does not add much new information.

Error Analysis Figure 2 shows examples of problems our approach encounters in IIRC. The first question requires the model to count occurrences of an event, but the QA model instead selects context containing a textual expression of the answer. The second question is a *span compare* example. The model has to identify context containing attributes of two entities mentioned in the original paragraph, but takes a shortcut and and only selects context for the correct answer.

4 Related Work

Most similar to our work are recent approaches using weak supervision for learning to retrieve for QA, using only questions and answers. Lee et al. (2019b) pretrain a retrieval model using an inverse cloze task. Zhao et al. (2021) more recently proposed to iteratively improve a retrieval model using hard-EM. Both approaches filter the data using the answer span heuristic. This heuristic breaks down on multi-hop questions, as well as questions that are not answerable by spans, such as true/false or discrete reasoning questions. Izacard and Grave (2021) and Yang and Seo (2021) propose using knowledge distillation to incorporate QA information into a supervised retriever, and while assuming access to retrieval annotations, Ni et al. (2020) jointly learn retrieval and QA by marginalizing over potential contexts. All three of these approaches require encoding all potential contexts together with the question, whereas ours does not have that requirement, making ours more memory-efficient.

5 Conclusion

This work shows that using the loss of a QA model trained on a partial set of labeled contexts to search for alternative contexts for retrieval is an effective method for augmenting the retriever's training data. Our results present a more label-efficient training scheme for building supervised retrievers for QA. They also suggest that creators of datasets for open QA tasks that require supervised retrievers can better allocate their annotation budgets by obtaining retrieval labels for a small set of questions while maximizing the number of question-answer annotations.

References

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

Christopher Clark and Matt Gardner. 2018. Simple and effective multi-paragraph reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 845–855, Melbourne, Australia. Association for Computational Linguistics.

James Ferguson, Matt Gardner, Hannaneh Hajishirzi, Tushar Khot, and Pradeep Dasigi. 2020. Iirc: A dataset of incomplete information reading comprehension questions. In *ACL*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR.

Gautier Izacard and Edouard Grave. 2021. Distilling knowledge from reader to retriever for question answering. In *ICLR*.

Harsh Jhamtani and Peter Clark. 2020. Learning to explain: Datasets and models for identifying valid reasoning chains in multihop question-answering. In *EMNLP*.

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

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. Unifiedqa: Crossing format boundaries with a single qa system. In *EMNLP*.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *SIGIR*.

Tushar Khot, Peter Clark, Michael Guerquin, Petre Jansen, and Shish Sabharwal. 2019. Qasc: A dataset for question answering via sentence composition. In *AAAI*.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019a. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019b. Latent retrieval for weakly supervised open domain question answering. In *ACL*.

Patrick Lewis, Ethan Perez, Aleksandara Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *ArXiv*, abs/2005.11401.

Ansong Ni, Matt Gardner, and Pradeep Dasigi. 2020. Mitigating false-negative contexts in multi-document question answering with retrieval marginalization. In *arXiv preprint arXiv:2103.12235*.

Qiu Ran, Yankai Lin, Peng Li, Jie Zhou, and Zhiyuan Liu. 2019. NumNet: Machine reading comprehension with numerical reasoning. In *EMNLP*.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. In *Foundations and Trends in Information Retrieval*.

Sohee Yang and Minjoon Seo. 2021. Is retriever merely an approximator of reader? In *arXiv preprint arXiv:2010.10999*.

Chen Zhao, Chenyan Xiong, Jordan Boyd-Graber, and Hal Daumé III. 2021. Distantly-supervised evidence retrieval enables question answerwing without evidence annotation. In *EMNLP*.