

Weakly-Supervised Open-Retrieval Conversational Question Answering

Chen Qu¹, Liu Yang¹, Cen Chen², W. Bruce Croft¹,
Kalpesh Krishna¹, and Mohit Iyyer¹

¹ University of Massachusetts Amherst
{chenqu,lyang,croft,kalpesh,m.iyyer}@cs.umass.edu

² Ant Financial Services Group
chencen.cc@antfin.com

Abstract. Recent studies on Question Answering (QA) and Conversational QA (ConvQA) emphasize the role of retrieval: a system first retrieves evidence from a large collection and then extracts answers. This open-retrieval ConvQA setting typically assumes that each question is answerable by a single span of text within a particular passage (a span answer). The supervision signal is thus derived from whether or not the system can recover an exact match of this ground-truth answer span from the retrieved passages. This method is referred to as *span-match weak supervision*. However, information-seeking conversations are challenging for this span-match method since long answers, especially freeform answers, are not necessarily strict spans of any passage. Therefore, we introduce a *learned weak supervision* approach that can identify a paraphrased span of the known answer in a passage. Our experiments on QuAC and CoQA datasets show that the span-match weak supervisor can only handle conversations with span answers, and has less satisfactory results for freeform answers generated by people. Our method is more flexible as it can handle both span answers and freeform answers. Moreover, our method can be more powerful when combined with the span-match method which shows it is complementary to the span-match method. We also conduct in-depth analyses to show more insights on open-retrieval ConvQA under a weak supervision setting.

Keywords: Weak Supervision · Open-Retrieval · Conversational Question Answering.

1 Introduction

Conversational search and Conversational Question Answering (ConvQA) have become one of the focuses of information retrieval research. Previous studies [5, 36] set up the ConvQA problem as to extract an answer for the conversation so far from a *given gold passage*. Recent work [30] has emphasized the fundamental role of retrieval by presenting an Open-Retrieval ConvQA (ORConvQA) setting. This setting requires the system to *learn* to retrieve top relevant passages from a large collection and then extract answers from the passages.

The open-retrieval setting presents challenges to training the QA/ConvQA system. Qu et al. [30] adopts a fully-supervised setting, which encourages the model to find the gold passage and extract an answer from it by manually including the gold passage in the retrieval results during training. This *full supervision* setting can be impractical since gold passages may not always be available. In contrast, other studies [2, 23, 8] assume no access to gold passages and identify weak answers in the retrieval results by finding a span that is an exact match to the known answer. We argue that the effectiveness of this *span-match weak supervision* approach is contingent on having only *span answers* that are short, or extractive spans of a retrieved passage. In information-seeking conversations, however, answers can be relatively long and are not necessarily strict spans of any passage. These *freeform answers* can be challenging to handle for span-match weak supervision.

In this work, we introduce a *learned weak supervision* approach that can identify a paraphrased span of the known answer in a retrieved passage as the weak answer. Our method is more flexible than span-match weak supervision since that it can handle both span answers and freeform answers. Moreover, our method is less demanding on the retriever since it can discover weak answers even when the retriever fails to retrieve any passage that contains an exact match of the known answer. By using a weakly-supervised training approach, our ConvQA system can discover answers in passages beyond the gold ones and thus can potentially leverage various knowledge sources. In other words, our learned weak supervision approach makes it possible for an ORConvQA system to be trained on natural conversations that can have long and freeform answers. The choice of the passage collection is no longer a part of the task definition. We can potentially combine different knowledge sources with these conversations since the weak answers can be discovered automatically.

Our learned weak supervisor is based on Transformers [41]. Due to the lack of training data to learn this module, we propose a novel training method for the learned weak supervisor by leveraging a diverse paraphraser [19] to generate the training data. Once the learned weak supervisor is trained, it is frozen and used to facilitate the training of the ORConvQA model.

We conduct experiments with the QuAC [5] and CoQA [36] datasets in an open-retrieval setting. We show that although a span-match weak supervisor can handle conversations with span answers, it is not sufficient for those with freeform answers. For more natural conversations with freeform answers, we demonstrate that our learned weak supervisor can outperform the span-match one, proving the capability of our method in dealing with freeform answers. Moreover, by combining the span-match supervisor and our method, the system has a significant improvement over using any one of the methods alone, indicating these two methods complement each other. Finally, we perform in-depth quantitative and qualitative analyses to provide more insight into weakly-supervised ORConvQA. Our data and model implementations will be available for research purposes.³

³ <https://github.com/prdwb/ws-orconvqa>

The rest of our paper is organized as follows. In Section 2, we present related work regarding question answering and conversational question answering. In Section 3, we formulate the research question of ORConvQA following previous work and present our weakly-supervised solution. In Section 4, we present our evaluation results on both span answers and freeform answers. Finally, Section 5 presents the conclusion and future work.

2 Related Work

Our work is closely related to question answering, conversational question answering, session search [27, 26, 56], and weak supervision and data augmentation [24, 3]. We highlight the related works on QA and ConvQA as follows.

Question Answering. Most of the previous work formulates question answering either as an answer selection task [54, 43, 13] or a machine comprehension (MC) task [35, 34, 20, 39]. These settings overlook the fundamental role of retrieval as articulated in the QA task of the TREC-8 Question Answering Track [42]. Another line of research on open-domain question answering addresses this issue by leveraging multiple documents or even the entire collection to answer a question [28, 16, 11, 10, 7]. When a large collection is given as a knowledge source, previous work [2, 53] typically uses TF-IDF or BM25 to retrieve a small set of candidate documents before applying a neural reader to extract answers. More recently, neural models are being leveraged to construct learnable rerankers [22, 14, 18, 44] or learnable retrievers [23, 8, 17] to enhance the retrieval performance. Compared to this work on single-turn QA, we focus on a conversational setting as a further step towards conversational search.

Conversational Question Answering. As an extension of the answer selection and MC tasks in single-turn QA, most research on conversational QA focuses on conversational response ranking [50, 25, 49, 48, 38, 47, 51, 52] and conversational MC [5, 36, 32, 31, 15, 57, 55, 4, 29]. A recent paper [30] extends conversational QA to an open-retrieval setting, where the system is required to learn to retrieve top relevant passages from a large collection before extracting answers from the passages. Although this research features a learnable retriever to emphasize the role of retrieval in ConvQA, it adopts a fully-supervised setting. This setting requires the model to have access to gold passages during training, and thus is less practical in real-world scenarios. Instead, we propose a learned weakly-supervised training approach that can identify good answers in any retrieved documents. In contrast to the span-match weak supervision [2, 23, 8] used in single-turn QA, our approach is more flexible since it can handle freeform answers that are not necessarily a part of any passage.

3 Weakly-Supervised ORConvQA

In this section, we first formally define the task of open-retrieval ConvQA under a weak supervision setting. We then describe an existing ORConvQA model [30] and explain how we train it with our learned weak supervision approach.

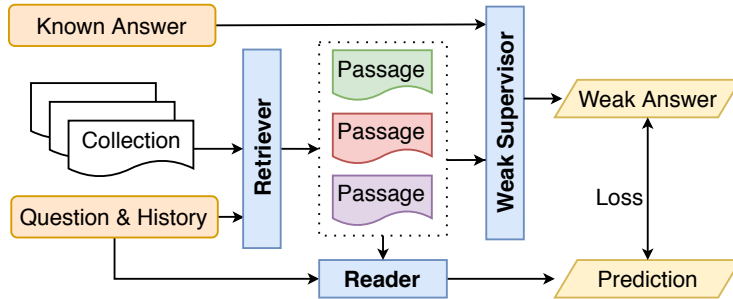


Fig. 1. Architecture of our full model. Given a question and its conversation history, the retriever first retrieves top-K relevant passages from the collection. The reader then reads the top passages and produces an answer. We adopt a weakly-supervised training approach. Given the known answer and one of the retrieved passages, the weak supervisor predicts a span in this passage as the weak answer to provide weak supervision signals for training the reader.

3.1 Task Definition

We define the ORConvQA task following Qu et al. [30]. Given the k -th question q_k in a conversation, and all history questions $\{q_i\}_{i=1}^{k-1}$ preceding q_k , the task is to predict an answer a_k for q_k using a passage collection C . Different from Qu et al. [30], we assume no access to gold passages when training the reader. The gold passage for q_k is the passage in C that is known to contain or support a_k .

3.2 An End-to-End ORConvQA System

We follow the same architecture of the ORConvQA model in Qu et al. [30].⁴ Our approach differs from theirs in how we train the model. They use full supervision while we adopt weak supervision. We briefly describe the architecture of this ORConvQA model before introducing our weakly-supervised training approach.

As illustrated in Figure 1, the ORConvQA model is composed of a passage retriever and a passage reader that are both learnable and based on Transformers [41]. Given a question and its history, the retriever first retrieves top-K relevant passages from the collection. The reader then reads the top passages and produces an answer. History modeling is enabled in both components by concatenating history questions. Since we do not have access to ground-truth history answers and gold passages, advanced history modeling approaches proposed in previous research [31, 32] does not apply here. The training contains two phases, a pretraining phase for the retriever, and a concurrent learning phase for the reader and fine-tuning the question encoder in the retriever. Our weakly-supervised training approach is applied to the concurrent learning phase.

⁴ We disable the reranker in Qu et al. [30] since our preliminary experiments indicated the weak supervision signals seem to lead to degradation for reranker and retriever.

Retriever The learnable retriever follows a dual-encoder architecture [1, 23, 8] that has a passage encoder and a question encoder. Both encoders are based on ALBERT [21] and can encode a question/passage into a 128-dimensional dense vector. The question is enhanced with history by prepending the initial question and other history questions within a history window. The retriever score is defined as the dot product of the representations of the question and the passage. The retriever pretraining process ensures the retriever has a reasonable initial performance during concurrent learning. A pretraining example contains a question and its gold passage. Other passages in the batch serve as sampled negatives. Using the passage encoder in the pretrained retriever, we encode the collection of passages to a collection of vectors. We then use Faiss⁵ to create an index of these vectors for maximum inner product search [37] on GPU. The question encoder will be fine-tuned during concurrent learning using the retrieved passages. We refer our readers to Qu et al. [30] for further details.

Reader The reader adapts a standard BERT-based extractive machine comprehension model [9] to a multi-document setting by using the shared-normalization mechanism [6] during training. First, the retrieved passages are encoded independently. Then, the reader maximizes the probabilities of the true start and end tokens among tokens from all the top passages. This step enables the reader to produce comparable token scores across all the retrieved passages for a question. The reader score is defined as the sum of the scores of the start token and the end token. The answer score is then the sum of its retriever score and reader score.

3.3 Weakly-Supervised Training

The reader component in Qu et al. [30] is trained with access to gold passages while our model is supervised by the conversation only. Our weakly-supervised training approach is *more practical* in real-world scenarios. Figure 1 illustrates the role the weak supervisor plays in the system. Given a known answer a_k and one of the retrieved passages p_j , the weak supervisor predicts a span in p_j as the *weak answer* a_k^{weak} . This weak answer is the weak supervision signal for training the reader. The weak supervisor can also indicate there is no weak answer contained in p_j . A question is skipped if there are no weak answers in any of the retrieved passages.

Inspirations Our learned weak supervision method is inspired by the classic span-match weak supervision. This method has been the default and only weak supervision method in previous open-domain QA research [23, 2, 8]. These works mainly focus on factoid QA, where answers are short. A span-match weak supervisor can provide accurate supervision signals since the weak answers are exactly the same as the known answers. In addition, the short answers can find matches

⁵ <https://github.com/facebookresearch/faiss>

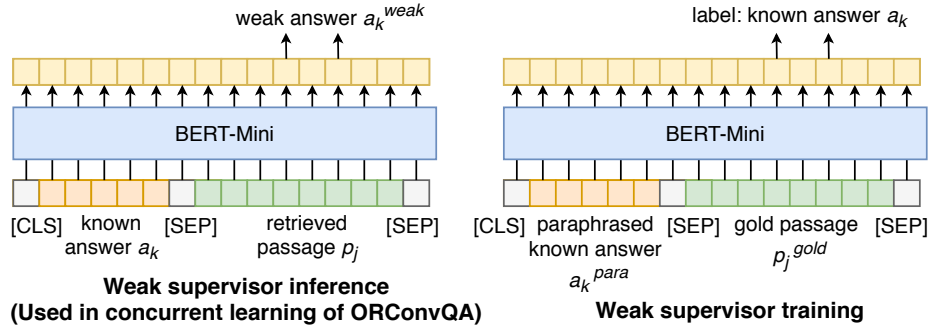


Fig. 2. Learned weak supervisor. During the concurrent learning phase of ORConvQA, the weak supervisor conducts inference on a retrieved passage p_j (the left figure) to predict a passage span that is a paraphrase of the known answer a_k . When training of the weak supervisor (the right figure), the model is trained to predict the known answer a_k in the passage given a paraphrase of the known answer a_k^{para} and the passage.

easily in passages other than the gold ones. In information-seeking conversations, however, the answers can be long and freeform, and thus are more difficult to get an exact match in retrieved passages. Although the span-match weak supervisor can still provide accurate supervision signals in this scenario, it renders many training examples useless due to the failure to find exact matches. A straightforward solution is to find a span in a retrieved passage that has the maximum overlap with the known answer. Such overlap can be measured by word-level F1. This overlap method, however, can be intractable and inefficient since it has to enumerate all spans in the passage. This method also requires careful tuning for the threshold to output “no answer”. Therefore, we introduce a learned weak supervisor based on Transformers [41] to predict a weak answer span directly in a retrieved passage given the known answer. This supervisor also has the ability to indicate that the retrieved passage does not have a good weak answer.

Learned Weak Supervisor Given the known answer a_k and one of the retrieved passages p_j , the weak supervisor predicts a span in p_j as the weak answer a_k^{weak} . Intuitively, a_k^{weak} is a paraphrase of a_k . We use a standard BERT-based extractive MC model [9] here as shown in Figure 2, except that we use a_k for the question segment. The best weak answer for all top passages is the one with the largest sum of start and end token scores.

Although theoretically simple, this model presents challenges in training because position labels of a_k^{weak} are not available. Therefore, we consider the known answer a_k as the weak answer we are seeking since we know the exact position of a_k in its gold passage p_j^{gold} . We then use a diverse paraphrase generation model (described in Section 3.3) to generate a paraphrase a_k^{para} for the known answer a_k . The paraphrase a_k^{para} simulates the known answer during the training of the weak supervisor, as shown in Figure 2. The weak supervisor is trained before

concurrent learning and kept frozen during concurrent learning. We train the weak supervisor to tell if the passage does not contain a weak answer by pairing a randomly sampled negative passage with the known answer.

We are aware of a dataset, CoQA [36], that provides both span answer and freeform answer for a given question q_k . In this case, we can take the freeform answer as a natural paraphrase a_k^{para} for the span answer (known answer) a_k when training the weak supervisor. For datasets that do not offer both answer types, our diverse paraphraser assumes the role of the oracle to generate the paraphrase answer. In other words, the use of the diverse paraphraser ensures that our weak supervision approach can be applied to a wide variety of conversation data that are beyond datasets like CoQA.

Diverse Paraphrase Model We now briefly describe the diverse paraphraser [19] used in the training process of the learned weak supervisor. This model is built by fine-tuning GPT2-large [33] using encoder-free seq2seq modeling [46]. As training data we use PARANMT-50M [45], a massive corpus of back translated data [45]. The training corpus is aggressively filtered to leave sentence pairs with high lexical and syntactic diversity so that the model can generate diverse paraphrases. We refer our readers to Krishna et al. [19] for further details.

4 Experiments

We now describe the experimental setup and report the results of our evaluations.

4.1 Experimental Setup

Dataset We select two ConvQA datasets, QuAC [5] and CoQA [36], with different answer types (span/freeform) to conduct a comprehensive evaluation of our weak supervision approach and to provide insights for weakly-supervised ORConvQA. We present the data statistics of both datasets in Table 1. We remove unanswerable questions in both datasets since there is no basis to find weak answers.⁶

OR-QuAC (span answers) We use the OR-QuAC dataset introduced in Qu et al. [30]. This dataset adapts QuAC to an open-retrieval setting. It contains information-seeking conversations from QuAC, and a collection of 11 million Wikipedia passages (document chunks).

OR-CoQA (freeform answers) We process the CoQA dataset [36] in the Wikipedia domain for the open-retrieval setting following Qu et al. [30], resulting in the OR-CoQA dataset. CoQA offers freeform answers generated by people in addition to span answers, resulting in more natural conversations. OR-CoQA and OR-QuAC

⁶ This difference in the data accounts for the discrepancies of the full-supervision results presented in Table 2.

Table 1. Data Statistics.

Items	OR-CoQA			OR-QuAC		
	Train	Dev	Test	Train	Dev	Test
# Dialogs	1,521	100	100	4,383	490	771
# Questions	23,027	1,494	1,611	25,824	2,808	4,406
# Avg. Question Tokens	5.8	5.7	5.8	6.8	6.6	6.8
# Avg. Answer Tokens	2.8	2.6	2.6	15.0	15.0	14.7
# Avg. Dialog Questions	15.1	14.9	16.1	5.9	5.7	5.7
# Avg./Max History	7.9/22	7.6/21	7.9/19	2.8/11	2.8/11	2.8/11
Turns per Question						

share the same passage collection. Similar to QuAC, many initial questions in CoQA are also ambiguous and hard to interpret without the given gold passage (e.g., “When was the University established?”). OR-QuAC deals with this by replacing the *first question* of a conversation with its context-independent *rewrite* offered by the CANARD dataset [12] (e.g., “When was the University of Chicago established?”). This makes the conversations self-contained. Since we are not aware of any CANARD-like resources for CoQA, we prepend the document title to the first question for the same purpose (e.g., “*University of Chicago* When was the University established?”). Since the CoQA test set is not publicly available, we take the original development set as our test set and 100 dialogs from the original training set as our development set.

Competing Methods Since this work focuses on weak supervision, we use the same ORConvQA model and vary the supervision methods. To be specific, the competing methods are:

- **Full supervision** (Full S): Manually add the gold passage to the retrieval results and use the ground-truth answer span [30]. This only applies to QuAC since we have no passage relevance for CoQA. This method serves as the upper bound of model performance and it is not comparable with other weak supervision methods that do not have access to the groundtruth answers in concurrent learning.
- **Span-match weak supervision** (Span-match WS): This method finds a weak answer span that is identical to the known answer in the retrieved passages. When there are multiple matched spans, we take the first one.
- **Learned weak supervision** (Learned WS): This is our method in Section 3.3 that finds a paraphrased span of the known answer as the weak answer.
- **Combined weak supervision** (Combined WS): This is the combination of the above two methods. We first use the span-match weak supervisor to try to find a weak answer. If it fails, we take the weak answer found by the learned weak supervisor.

Evaluation Metrics We use the word-level F1 and human equivalence score (HEQ) [5] to evaluate the performance of ConvQA. **F1** evaluates the overlap between the prediction and the ground-truth answer. **HEQ** is the percentage of examples for which system F1 \geq human F1. This is computed on a question level (HEQ-Q) and a dialog level (HEQ-D).

In addition to the performance metrics described above, we define another set of metrics to reveal the impact of the weak supervisor in the training process as follows. **% Has Answer** is the percentage of training examples that have a weak answer (in the last epoch). **% Hit Gold** is the percentage of training examples that have a weak answer identified in gold passages (in the last epoch). **Recall** is the percentage of training examples that have the gold passage retrieved (in the last epoch). **% From Gold** is the percentage of predicted answers that are extracted from the gold passages.

Implementation Details Our models are based on the open-source implementation of ORConvQA⁷, Diverse Paraphrase Model⁸, and the HuggingFace Transformers repository.⁹ We use the same pretrained retriever in Qu et al. [30] for both datasets. For concurrent learning of ORConvQA, we set the number of training epochs to 5 (larger than [30]) to account for the skipped steps where no weak answers are found. We set the number of passages to update the retriever to 100, and the history window size to 6 since these are the best settings reported in [30]. The max answer length is set to 40 for QuAC and 8 for CoQA. The rest of the hyper-parameters and implementation details for the ORConvQA model are the same as in [30].

For the weak supervisor, we use BERT-Mini [40] for better efficiency. We set the number of training epochs to 4, the learning rate to 1e-4, and the batch size to 16. As discussed in Section 3.3, the diverse paraphraser is used for OR-QuAC only. For OR-CoQA, we use the freeform answer provided by the dataset as a natural paraphrase to the span answer.

4.2 Evaluation Results on Span Answers

Given the different properties of span answers and freeform answers, we study the performance of our weak supervision approach on these answers separately. We report the evaluation results on the span answers in Table 2. Our observations can be summarized as follows.

The full supervision setting yields the best performance, as expected. This verifies the supervision signals provided by the gold passages and the ground-truth answer spans are more accurate than the weak ones. Besides, all supervision approaches have similar performance on span answers. This suggests that span-match weak supervision is sufficient to handle conversations with span answers.

⁷ <https://github.com/prdwb/orconvqa-release>

⁸ <https://github.com/martiansideofthemoon/style-transfer-paraphrase>

⁹ <https://github.com/huggingface/transformers>

Table 2. Evaluation results on OR-QuAC (span answers). The learned weak supervisor causes no statistical significant performance decrease compared span match.

Methods		Full S	Span-match WS	Learned WS	Combined WS
Train	% Has Answer	100.00%	72.96%	75.98%	75.52%
Dev	F1	22.8	20.8	20.2	20.1
	HEQ-Q	8.1	6.8	6.0	6.4
	HEQ-D	0.6	0.6	0.2	0.6
Test	F1	23.9	23.6	23.1	23.2
	HEQ-Q	14.0	12.3	11.8	12.5
	HEQ-D	2.2	1.7	1.9	1.9

Ideally, if the known answer is part of the given passage, the learned weak supervisor should be able to predict the weak answer as exactly the same with the known answer. In other words, the learned weak supervisor should fall back to the span-match weak supervisor when handling span answers. In practice, this is not guaranteed due to the variance of neural models. However, our learned weak supervisor causes no statistical significant performance decrease compared with the span-match supervisor. This demonstrates that the learned weak supervision approach can cover span answers as well. Although we observe that the learned supervisor can identify more weak answers than span match, these weak answers could be false positives that do not contribute to the model performance. Finally, for the combined weak supervisor, our analysis shows that 96% of the weak answers are identified by span match, further explaining the fact that all weak supervision approaches have almost identical performance.

4.3 Evaluation Results on Freeform Answers

We then look at the evaluation results on freeform answers in Table 3. These are the cases where a span-match weak supervisor could fail. We observe that combining the learned weak supervisor with span match brings a statistically significant improvement over the span-match baseline on the test set, indicating these two methods complement each other. The test set has multiple reference answers per question, making the evaluation more practical. In addition, the learned supervisors can identify more weak answers than span match, these weak answers contribute to the better performance of our model. Further, for the combined weak supervisor, our analysis shows that 77% of the weak answers are identified by span match. This means that nearly a quarter of the weak answers are provided by the learned supervisor and used to improve the performance upon span match. This further validates the source of effectiveness of our model.

4.4 A Closer Look at the Training Process

We take a closer look at the training process, as shown in Table 4. We conduct this analysis on OR-QuAC only since we do not have the ground-truth passage

Table 3. Evaluation results on OR-CoQA (freeform answers). ‡ means statistically significant improvement over the span-match baseline with $p < 0.05$.

Methods		Span-match WS	Learned WS	Combined WS
Train	% Has answer	51.81%	65.75%	70.35%
Dev	F1	18.3	18.9	19.7
	HEQ-Q	11.6	9.0	12.7
	HEQ-D	0.0	0.0	0.0
Test	F1	24.3	26.0	28.8[‡]
	HEQ-Q	19.9	15.9	22.5
	HEQ-D	0.0	0.0	0.0

Table 4. A closer look at the training process for OR-QuAC.

Methods	Train			Dev	Test
	% Has Ans	% Hit Gold	Recall	% From Gold	% From Gold
Full S	100.00%	100.00%	1.0000	45.23%	27.46%
Span-match WS	72.96%	68.97%	0.7190	40.88%	28.80%
Learned WS	75.98%	67.24%	0.7187	39.89%	28.73%
Combined WS	75.52%	68.37%	0.7129	40.28%	28.39%

relevance for CoQA. We observe that, “% Has Ans” are higher than “% Hit Gold” for all weak supervision methods, indicating all of them can identify weak answers in passages beyond the gold passages. In particular, our method can identify more weak answers than span match. We also notice that “% Hit Gold” is only slightly lower than “Recall”, suggesting that most of the retrieved gold passages can yield a weak answer. This verifies the capability of weak supervisors. Finally, “% From Gold” are relatively low for all methods, indicating great potential for improvements.

4.5 Case Study and Error Analysis

We then conduct a qualitative analysis by presenting weak answers identified by the learned weak supervisor in Table 5 to better understand the weak supervision process. Example 1 and 2 show that our learned weak supervisor can find weak answers that are exactly the same or almost identical to the known answers when an exact match of the known answer exists, further validating our method can potentially cover span-match weak supervision. Example 3 shows that if an exact match does not exist, our method can find a weak answer that expresses the same meaning with the known answer. This is a case that a span-match weak supervisor would fail.

Example 4 shows that our method tends to focus on the lexical similarity only but get the fact wrong. Example 5 indicates our method sometimes finds a

Table 5. Case study. Weak answers are found by the learned weak supervisor. Boldface denotes discrepancies and italic denotes paraphrasing.

	#	Questions and Answers	
Good	1	Question Known answer Weak answer	Where was the album released? on online forums and music sites. on online forums and music sites.
	2	Question Known answer Weak answer	... mention anything else he starred in? After starring ... the film adaptation of The Music Man After starring ... film adaptation of The Music Man (1962) .
	3	Question Known answer Weak answer	Where did he distribute the Cocaine? flying out planes several times, mainly between Colombia and Panama, along smuggling routes into the United States. <i>He flew a plane himself several times, mainly between Colombia and Panama, in order to smuggle a load into the United States.</i>
Bad	4	Question Known answer Weak answer	how long have people had clothes? as long ago as 650 thousand years ago around 170,000 years ago.
	5	Question Known answer Weak answer	What is data compression called? reducing the size of a data file By using wavelets, a compression ratio

weak answer that is relevant to the known answer but cannot be considered as a good answer. These are the limitations of our method.

5 Conclusions and Future Work

In this work, we propose a learned weak supervision approach for open-retrieval conversational question answering. Extensive experiments on two datasets show that, although span-match weak supervision can handle span answers, it is not sufficient for freeform answers. Our learned weak supervisor is more flexible since it can handle both span answers and freeform answers. It is more powerful when combined with the span-match supervisor. For future work, we would like to enhance the performance of ORConvQA by studying more advanced history modeling methods and more effective weak supervision approaches.

Acknowledgments This work was supported in part by the Center for Intelligent Information Retrieval and in part by NSF IIS-1715095. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor. The authors would like to thank Minghui Qiu for his constructive comments on this work.

References

1. Ahmad, A., Constant, N., Yang, Y., Cer, D.M.: ReQA: An Evaluation for End-to-End Answer Retrieval Models. *ArXiv* (2019)
2. Chen, D., Fisch, A., Weston, J., Bordes, A.: Reading Wikipedia to Answer Open-Domain Questions. In: *ACL* (2017)
3. Chen, L., Tang, Z., Yang, G.: Balancing Reinforcement Learning Training Experiences in Interactive Information Retrieval. In: *SIGIR* (2020)
4. Chen, Y., Wu, L., Zaki, M.J.: GraphFlow: Exploiting Conversation Flow with Graph Neural Networks for Conversational Machine Comprehension. *ArXiv* (2019)
5. Choi, E., He, H., Iyyer, M., Yatskar, M., Yih, W.T., Choi, Y., Liang, P., Zettlemoyer, L.: QuAC: Question Answering in Context. In: *EMNLP* (2018)
6. Clark, C., Gardner, M.: Simple and Effective Multi-Paragraph Reading Comprehension. In: *ACL* (2017)
7. Cohen, D., Yang, L., Croft, W.B.: WikiPassageQA: A Benchmark Collection for Research on Non-factoid Answer Passage Retrieval. In: *SIGIR* (2018)
8. Das, R., Dhuliawala, S., Zaheer, M., McCallum, A.: Multi-step Retriever-Reader Interaction for Scalable Open-domain Question Answering. In: *ICLR* (2019)
9. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: *NAACL-HLT* (2019)
10. Dhingra, B., Mazaitis, K., Cohen, W.W.: Quasar: Datasets for Question Answering by Search and Reading. *ArXiv* (2017)
11. Dunn, M., Sagun, L., Higgins, M., Güney, V.U., Cirik, V., Cho, K.: SearchQA: A New Q&A Dataset Augmented with Context from a Search Engine. *ArXiv* (2017)
12. Elgohary, A., Peskov, D., Boyd-Graber, J.L.: Can You Unpack That? Learning to Rewrite Questions-in-Context. In: *EMNLP/IJCNLP* (2019)
13. Garg, S., Vu, T., Moschitti, A.: TANDA: Transfer and Adapt Pre-Trained Transformer Models for Answer Sentence Selection. In: *AAAI* (2020)
14. Htut, P.M., Bowman, S.R., Cho, K.: Training a Ranking Function for Open-Domain Question Answering. In: *NAACL-HLT* (2018)
15. Huang, H.Y., Choi, E., tau Yih, W.: Flowqa: Grasping flow in history for conversational machine comprehension. *ArXiv* (2018)
16. Joshi, M., Choi, E., Weld, D.S., Zettlemoyer, L.: TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In: *ACL* (2017)
17. Karpukhin, V., Ouguz, B., Min, S., Wu, L.Y., Edunov, S., Chen, D., tau Yih, W.: Dense Passage Retrieval for Open-Domain Question Answering. In: *EMNLP* (2020)
18. Kratzwald, B., Feuerriegel, S.: Adaptive Document Retrieval for Deep Question Answering. In: *EMNLP* (2018)
19. Krishna, K., Wieting, J., Iyyer, M.: Reformulating Unsupervised Style Transfer as Paraphrase Generation. In: *EMNLP* (2020)
20. Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A.P., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., Toutanova, K., Jones, L., Kelcey, M., Chang, M.W., Dai, A.M., Uszkoreit, J., Le, Q., Petrov, S.: Natural Questions: A Benchmark for Question Answering Research. *TACL* **7**, 453–466 (2019)
21. Lan, Z.Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. *ArXiv* (2019)
22. Lee, J., Yun, S., Kim, H., Ko, M., Kang, J.: Ranking Paragraphs for Improving Answer Recall in Open-Domain Question Answering. In: *EMNLP* (2018)

23. Lee, K., Chang, M.W., Toutanova, K.: Latent Retrieval for Weakly Supervised Open Domain Question Answering. In: *ACL* (2019)
24. Li, J., Qiu, L., Tang, B., Chen, D., Zhao, D., Yan, R.: Insufficient Data Can Also Rock! Learning to Converse Using Smaller Data with Augmentation. In: *AAAI* (2019)
25. Lowe, R., Pow, N., Serban, I., Pineau, J.: The Ubuntu Dialogue Corpus: A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems. In: *SIGDIAL* (2015)
26. Luo, J., Dong, X., Yang, G.: Learning to Reinforce Search Effectiveness. In: *ICTIR* (2015)
27. Luo, J., Zhang, S., Yang, G.: Win-win search: dual-agent stochastic game in session search. In: *SIGIR* (2014)
28. Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., Deng, L.: MS MARCO: A Human Generated MACHine Reading COmprehension Dataset. *ArXiv* (2016)
29. Qiu, M., Huang, X., Chen, C., Feng Ji, C.Q., Wei, W., Huang, J., Zhang, Y.: Reinforced History Backtracking for Conversational Question Answering. In: *AAAI* (2021)
30. Qu, C., Yang, L., Chen, C., Qiu, M., Croft, W.B., Iyyer, M.: Open-Retrieval Conversational Question Answering. In: *SIGIR* (2020)
31. Qu, C., Yang, L., Qiu, M., Zhang, Y., Chen, C., Croft, W.B., Iyyer, M.: Attentive History Selection for Conversational Question Answering. In: *CIKM* (2019)
32. Qu, C., Yang, L., Qiu, M., Croft, W.B., Zhang, Y., Iyyer, M.: BERT with History Answer Embedding for Conversational Question Answering. In: *SIGIR* (2019)
33. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language Models are Unsupervised Multitask Learners. *OpenAI Blog* (2019)
34. Rajpurkar, P., Jia, R., Liang, P.: Know What You Don't Know: Unanswerable Questions for SQuAD. In: *ACL* (2018)
35. Rajpurkar, P., Zhang, J., Lopyrev, K., Liang, P.: SQuAD: 100, 000+ Questions for Machine Comprehension of Text. In: *EMNLP* (2016)
36. Reddy, S., Chen, D., Manning, C.D.: CoQA: A Conversational Question Answering Challenge. *TACL* **7**, 249–266 (2018)
37. Shrivastava, A., Li, P.: Asymmetric LSH (ALSH) for Sublinear Time Maximum Inner Product Search (MIPS). In: *NIPS* (2014)
38. Tao, C., Wu, W., Xu, C., Hu, W., Zhao, D., Yan, R.: Multi-Representation Fusion Network for Multi-Turn Response Selection in Retrieval-Based Chatbots. In: *WSDM* (2019)
39. Trischler, A., Wang, T., Yuan, X., Harris, J., Sordoni, A., Bachman, P., Suleman, K.: NewsQA: A Machine Comprehension Dataset. In: *Rep4NLP@ACL* (2016)
40. Turc, I., Chang, M.W., Lee, K., Toutanova, K.: Well-Read Students Learn Better: On the Importance of Pre-training Compact Models. *ArXiv* (2019)
41. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention Is All You Need. In: *NIPS* (2017)
42. Voorhees, E.M., Tice, D.M.: The TREC-8 Question Answering Track Evaluation. In: *TREC* (1999)
43. Wang, M., Smith, N.A., Mitamura, T.: What is the Jeopardy Model? A Quasi-Synchronous Grammar for QA. In: *EMNLP-CoNLL* (2007)
44. Wang, S., Yu, M., Guo, X., Wang, Z., Klinger, T., Zhang, W., Chang, S., Tesauro, G., Zhou, B., Jiang, J.: R3: Reinforced Ranker-Reader for Open-Domain Question Answering. In: *AAAI* (2018)

45. Wieting, J., Gimpel, K.: ParaNMT-50M: Pushing the Limits of Paraphrastic Sentence Embeddings with Millions of Machine Translations. In: ACL (2018)
46. Wolf, T., Sanh, V., Chaumond, J., Delangue, C.: TransferTransfo: A Transfer Learning Approach for Neural Network Based Conversational Agents. In: NeurIPS CAI Workshop (2018)
47. Wu, Y., Wu, W.Y., Zhou, M., Li, Z.: Sequential Match Network: A New Architecture for Multi-turn Response Selection in Retrieval-based Chatbots. In: ACL (2016)
48. Yan, R., Song, Y., Wu, H.: Learning to Respond with Deep Neural Networks for Retrieval-Based Human-Computer Conversation System. In: SIGIR (2016)
49. Yan, R., Song, Y., Zhou, X., Wu, H.: "Shall I Be Your Chat Companion?": Towards an Online Human-Computer Conversation System. In: CIKM (2016)
50. Yang, L., Qiu, M., Qu, C., Guo, J., Zhang, Y., Croft, W.B., Huang, J., Chen, H.: Response Ranking with Deep Matching Networks and External Knowledge in Information-seeking Conversation Systems. In: SIGIR (2018)
51. Yang, L., Hu, J., Qiu, M., Qu, C., Gao, J., Croft, W.B., Liu, X., Shen, Y., Liu, J.: A Hybrid Retrieval-Generation Neural Conversation Model. In: CIKM (2019)
52. Yang, L., Qiu, M., Qu, C., Chen, C., Guo, J., Zhang, Y., Croft, W.B., Chen, H.: IART: Intent-aware Response Ranking with Transformers in Information-seeking Conversation Systems. In: WWW (2020)
53. Yang, W., Xie, Y., Lin, A., Li, X., Tan, L., Xiong, K., Li, M., Lin, J.: End-to-End Open-Domain Question Answering with BERTserini. In: NAACL-HLT (2019)
54. Yang, Y., Yih, W.T., Meek, C.: WikiQA: A Challenge Dataset for Open-Domain Question Answering. In: EMNLP (2015)
55. Yeh, Y.T., Chen, Y.N.: FlowDelta: Modeling Flow Information Gain in Reasoning for Conversational Machine Comprehension. ArXiv (2019)
56. Zhou, J., Agichtein, E.: RLIRank: Learning to Rank with Reinforcement Learning for Dynamic Search. In: WWW (2020)
57. Zhu, C., Zeng, M., Huang, X.: SDNet: Contextualized Attention-based Deep Network for Conversational Question Answering. ArXiv (2018)