Reinforced Learning for History Selection in Machine Reading Comprehension

Minghui Qiu[1,2] , **Xinjing Huang**[1] , **Cen Chen**[2] , **Feng Ji**[2] and **Yin Zhang**[1]

[1]Zhejiang University
[2]Alibaba Group

{minghui.qmh,chencen.cc,zhongxiu.jf}@alibaba-inc.com,
{huangxinjing,zhangyin98}@zju.edu.cn

Abstract

Multi-turn interactions have frequently occurred in the dialogue and question answering systems, where the current query often refers to the conversation context. Thus modeling the context becomes a critical step for the task of multi-turn machine reading comprehension (MRC). To utilize the context, traditional methods rely on modeling the whole context as input, which may be confused by the irrelevant parts that appeared in the dialogue history. To alleviate the problem, in this paper, we employ a reinforcement learning based method to select the related conversation history to maximize the MRC model performance. We conduct extensive experiments on QuAC dataset, a large multi-turn MRC dataset, to examine the efficiency and effectiveness of our method.

1 Introduction

Multi-turn machine reading comprehension (MRC) has been an important task for building conversational question answering system. A key question for multi-turn MRC is to model the context history. Recent state-of-the-art studies append all dialogue history by using history answer embedding [Qu *et al.*, 2019a] or question attention [Qu *et al.*, 2019b], which can be viewed as a soft selection of the related history. However, considering the whole history in the single model will inevitably face some challenges. 1) *Resource Limitation*. It requires more computation resources to incorporate the representation of all the history, including both the relevant and unrelated ones, which may be unnecessary for understanding the query. Moreover, this issue gets even worse when we adopt a heavy model such as BERT large, as the whole history needs to be maintained. 2) *Information Conflict*. Existing works that modeling the whole history usually employ attention or gating based mechanisms to selectively attend to different history turns. However, those methods may still cause confusion due to the irrelevant parts that appeared in the dialogue history. In the other words, the goal of the reading comprehension task is to minimize the prediction loss, rather than finding the most relevant histories to the current query.

To alleviate the above problems, in this paper we work from a different perspective and try to make meaningful selections of conversation history. The advantage of our method is that it can avoid the negative impact of unimportant history turns from the source by not considering them. We model the multi-turn MRC task as two subtasks: a conversational QA task using a neural MRC model and a conversation history selection task with a reinforced selector. The reinforced selector is an agent that interacts with the environment constructed by the multi-turn MRC. More specifically, for each query, we view the process of finding the related history as a sequential decision making process. The agent acts on the available conversation history and backtracks the history question-answer pairs turn by turn to decide whether it is relevant/useful based on the observations. The MRC model then uses the selected history turns to help itself answer the current question and generates a reward to evaluate the utility of the history selection. As irrelevant history are filtered, the MRC model can be better trained with more sophisticate mechanism and concentrate on fitting the history turns with more confidence. Moreover, as the reinforced selector is a separate module, it can be flexibly adapted and further improved with techniques such as transfer learning in the future.

We summarize our contributions as follows:

1. We propose a novel solution for modeling the conversation history in the multi-turn MRC setting. We incorporate a reinforced selector in the traditional MRC model to filter the irrelevant history turns instead of evaluating them as a whole. As a consequence, the MRC model can concentrate more on the relevant history and obtain better performance.
2. We model the conversation history selection problem as a sequential decision making process, which can be solved by reinforcement learning (RL). By interacting with a pre-trained MRC model, the reinforced selector is able to generate good selection policies. We further propose a novel training scheme to address the sparse reward issue.
3. We conduct extensive experiments on a large question answering dataset QuAC, and the results show that the learned conversation history selection policy by RL could help boost answer prediction performance.

The rest of our paper is organized as follows. In Section 2

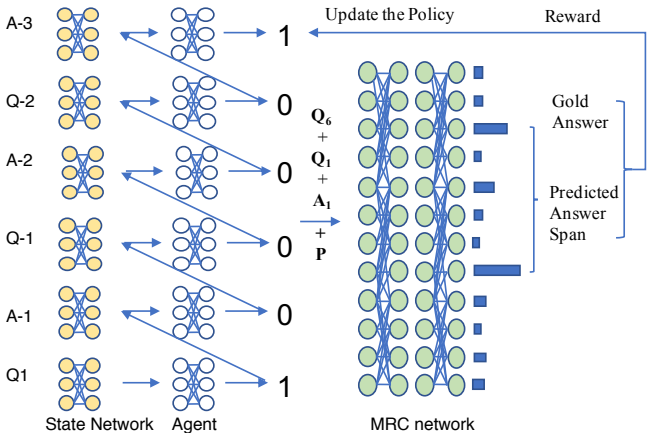

Figure 1: Overview of the our proposed MRC model with reinforced selector for the ConvQA task.

& 3, we formulate the conversation history selection problem in the multi-turn MRC setting and thoroughly elaborate our proposed approach that uses reinforcement learning to train backtracking policy for useful history-turn selection. In Section 4, we conduct the detailed experiments on the QuAC dataset. In Section 5, we present the related work regarding machine comprehension, conversational question answering, and reinforcement learning. Finally, we conclude the work in Section 6.

2 Task Definition

We define the conversation history selection task on top of the ConvQA task. We formulate our task into two subtasks: a conversational QA task and a conversation history selection task. Given the current question Q_k and dialogue history $H = \{(Q_i, A_i)|_{i=0}^{i=k-1}\}$, our reinforced backtracker aims to find a subset $H' \in H$ of most relevant history turns, to maximize the performance of the ConvQA task.

3 Models

In this section, we first present an overview of our proposed reinforced selecter, followed by detailed discussions on the state representation, the Reinforcement Learning (RL) agent, the environment and the reward.

3.1 Model Overview

As illustrated in Figure 1, we model the history selection problem as a sequential decision making process. Given the current query, the agent selects the history and obtains the state representation in each dialogue turn through the state network. The policy network takes the state representation and the last action to decide whether this history turn is related to the query. Subsequently, the MRC model uses the selected history turns and the passage as the inputs to predict the answer span. The history selection quality has a direct impact on the answer prediction performance. Thus, the MRC model is able to generate a reward to evaluate the utility of the history selection. Finally, the reward is used to update the policy network. We now introduce the state, agent, environment, and reward in detail in the following sections.

3.2 State

The state of a given history turn (Q_i, A_i) is denoted as a continuous real valued vector $\mathbf{S_i} \in R^l$, where l is the dimension of the state vector. The state vector S in i-th selection is the concatenation of the following features:

$$\mathbf{S_i} = [h_i \oplus V(a_{i-1}) \oplus V(i) \oplus \omega] \quad (1)$$

- **Sentence Vector** h_i**.** We adopt the average of the word's hidden representation generated by the vanilla BERT model as the sentence vector, where the input to the BERT is a sentence pair as follows: `[CLS]` $Q_i A_i$ `[SEP]`.
- **Last Action's Vector** $V(a_{i-1})$**.** We embed the last action into an action vector with length 20.
- **Position Vector** $V(i)$. We embed the current relative step into this vector, which is designed to inject the position information.
- **Segment Embedding** ω. This vector is defined as the average of past sentence embeddings whose corresponding action is 1 (denotes being selected), formally:

$$\omega = \sum_{m=1}^{i-1} h_m, \ where \ a_m = 1. \quad (2)$$

3.3 RL Agent

Policy Network. Given the state, our policy network is a fully connected neural network, defined as follows:

$$P = softmax(W \times S). \quad (3)$$

At the training stage, we calculate the action distribution and sample action from the distribution. At the evaluation stage, we select the action according to the max probability.

The policy gradient is calculated as followings:

$$\nabla_\theta J(\theta) = E\Big(\sum_{t=1}^{L}(R - b(\tau))\nabla_\theta \log \pi_\theta(a_t|s_t)\Big), \quad (4)$$

where b is the baseline, designed to reduce variance. We adopt the average return in the batch as our baseline. R is the cumulative reward, which will be discussed in the section 3.5.

Action. As our goal is to select the related history turns, the agent has two options for each turn: 0 (ignored) or 1 (selected).

3.4 Environment

Given the current question Q_k, a subset of the dialogue history H' and passage P, the environment reformulates the multi-turn ConvQA task to a unified single-turn machine reading comprehension (MRC) task by prepending the dialogue history to the current question. Then the model predicts the answer span and generates a reward to evaluate the utility of the history selection for predicting the answer.

In this paper, we adopt BERT [Devlin *et al.*, 2018] as our MRC model. The input for BERT is defined as

[CLS] Q_k [SEP] H' [SEP] P [SEP], where Q_k and P refer to the current k-th question and the passage, H' is the set of the selected history turns. We denote the output of BERT as $H_{rc} \in R^{L \times D_m}$, where L is the length of the input and D_m is the dimension of the vector. Formally, we predict the start and end positions of the answer as the following:

$$P_s = Softmax(W_s H_{rc}^T + b_s), \quad (5)$$

$$P_e = Softmax(W_e H_{rc}^T + b_e), \quad (6)$$

where W_s, $W_e \in R^{1 \times D_m}$, b_s, $b_e \in R^{1 \times L}$, and s stands for the start and e stands for the end.

3.5 Reward

Our goal is to maximize the accuracy of the MRC model's prediction through the inputs selected by the agent. So an intuitive way is to adopt the word-level F1 score between the predicted answer and the ground-truth as our reward R. If the input information is not sufficient for the model to predict the answer correctly, the F1 score can be low. Formally, the F1 score is defined as follows:

$$F1 = \frac{2 * P * R}{P + R} \quad (7)$$

where P is the overlap percentage of the gold answer that counts in the predicted answer, R is the overlap percentage that counts in the gold answer.

3.6 Algorithm

Details of the algorithm can be seen in Figure 1 and Algorithm 3.6. The agent is comprised of two modules: a vanilla BERT and policy network. The vanilla BERT aims to obtain the sentence representation. Here we directly adopt the released pre-trained model from github and freeze its weights. Given the current question Q_k and the history H, the agent repeatedly generates selection decisions. After the selecting procedure, we use REINFORCE algorithm [Williams, 1992] to update the policy.

Training Scheme

The idea of the training scheme is to gradually increase the difficulty of learning. The agent firstly learns policy from the episodes with only one history, which can be viewed as the simplified selection procedure. Then we increase the length of the episodes to make the agent learn more complicated strategy. On the other hand, the normal strategy that randomly selects an example from the dataset and can see the simple and difficult situations simultaneously. This kind of setting might confuse the agent.

Algorithm 1 RL Training Procedure

Require: Environment M: A pre-trained MRC model with latest 8 history turns;
Sentence Representation Model: (Vanilla) $BERT_s$
Policy Network: P_n

for j in range(MAX History Turn) **do**
  for $Q_k, H = \{(Q_i, A_i)|_{i=0}^{i=k-1}\}$ in training data **do**
    $V_q = BERT_s(Q_k)$
    if $len(H) > j$ **then**
      continue
    end if
    Actions=[1]
    for (Q_i, A_i) in H **do**
      $h_i = BERT_s(Q_i \oplus A_i)$
      $State = h_i \oplus \omega \oplus V(a_{i-1}) \oplus V_i$
      $a_i = P_n(State)$
      Actions.append(a_i)
    end for
    Obtain history subset H' according to $Actions$
  end for
  Obtain reward R according to Eq. 7
  Using R to update the policy network P_n
end for

4 Experiments

4.1 Datasets

We conduct experiments on the QuAC dataset. QuAC is a machine reading comprehension task with multi-turn interactions, where questions often refer to the dialogue history. Some dialogue behaviors often occur such as topic shift, drill down and topic return. There are mainly 100k questions and 10k dialogue in the dataset. The maximum round in the dialogue is 12.

Also, we conduct reinforcement learning on a rewritten dataset Canard, which partly selects around 31k questions from the QuAC dataset and rewrite them to well-formed questions.

Evaluation Metrics

The QuAC challenge provides two evaluation metrics, the word-level F1, and the human equivalence score (HEQ). The word-level F1 evaluates the overlap of the prediction and the ground truth answer span. It is a classic metric used in (conversational) machine comprehension tasks [Rajpurkar *et al.*, 2016; Reddy *et al.*, 2018]. HEQ measures the percentage of examples for which system F1 exceeds or matches human F1. Intuitively, this metric judges whether a system can provide answers as good as an average human. This metric is computed on the question level (HEQ-Q) and the dialog level (HEQ-D).

4.2 Environment

We test our method on the following varied environments. We adopt the same model architecture but with different inputs and training corpus.

Env-ConvQA We denote the method of appending the latest k historical question-answer pairs to its current question as Env-ConvQA. Formally, the current question Q_k and its latest 8 history turns $H_8 = \{(Q_k - i, A_k - i)|_{i=1}^{i=min(8,k)}\}$ are concatenated to be a new long question then accepted as the input of the model. This method is a strong baseline as shown in [Zhu *et al.*, 2018; Ju *et al.*, 2019].

Env-ST (Single Turn) We denote the method of training a single turn MRC model on the first turn of dialogues in QuAC dataset as Env-ST. This is to avoid the negative impact caused by introducing dialogue history for the MRC model. Note the

Table 1: ConvQA means we append the past k history question answer pairs to the current question. Here we report the averaged results on development dataset for k in {0,4,8,12} in 5 runs.

Models	F1	HEQ-Q	HEQ-D	Total
ConvQA w/ no history	55.93	49.43	3.3	108.66
ConvQA w/ 4 avg	63.84	59.29	5.8	128.93
ConvQA w/ 8 avg	**64.02**	**59.59**	**6.3**	**129.91**
ConvQA w/ 12 avg	63.12	58.37	5.5	126.99

number of examples is the same with the number of dialogues in QuAC. The training dataset has 11,567 examples.

Env-Canard (Canard Dataset)) As the Canard dataset has re-written questions based on the history turns, it can serve as a perfect environment to examine different history modeling policies. We denote the method of training the MRC model on the re-written questions from Canard as Env-Canard. It has about 31k training examples.

4.3 Baselines

We consider to compare our method with rule based methods.

Rule Based Methods We define the rule-based policy where latest k history question-answer pairs are selected. This setting is adopted as baselines in recent studies [Choi *et al.*, 2018; Qu *et al.*, 2019b; Qu *et al.*, 2019a].

4.4 The Necessity of the Selection

As shown in this study [Yatskar, 2019], as topic shift and topic-return are common in conversations, thus it is necessary to prepend history turns in the model. However, there still remains an important question, *the more history turns appended, the better the performance will be?*

As shown in Table 1, we conduct experiments on training PQA model with various history turns, the performance will increase when we append 8 history turns instead of 4, but decrease when we append the latest 12 turns. The potential reasons behind this are: 1) Information conflict. It is hard for the model to automatically capture the dependices between related parts. 2) Length limitation. The more turns we appended, the less the passage words are included due to the length limitation of the BERT, which makes the model difficult to extract the key information in the input.

4.5 Reinforcement Learning vs Rule Policy

In this section we aim to examine the benefits of our reinforcement learning method. We conduct reinforcement learning on three environments Env-ConvQA, Env-ST and Env-Canard as discussed in Section 4.2.

As shown in Table 2, we compare the performance of different setting of our agent learning and rule policy. For Env-Canard obtained by training model on manually rewritten questions, it can be viewed as an good environment which can judge how good the selected history pairs are. We can see that the policy with no history can achieve the best performance. But with more histories appended, the performance start to drop dramatically. This means more history turns bring more noise information instead of useful information. When applied our method, the performance increases greatly. This shows our agent can help to backtrack helpful history turns to achieve better performance, i.e. dig out useful information from history turns.

Table 2: The performance of Env-Canard environment.

Env-Canard	Reward	F1	HEQQ	HEQD	Total
Rule-0	-	48.61	41.43	2.2	92.24
Rule-4	-	44.26	36.92	0.9	82.08
Rule-8	-	44.25	36.91	0.9	82.06
Rule-12	-	44.25	36.91	0.9	82.06
Agent	F1	**49.90**	**42.89**	**2.3**	**95.09**

Table 3: The reinforcement learning on single-turn environment.

Env-ST	Reward	F1	HEQQ	HEQD	Total
Rule-0	-	33.60	**27.58**	1.0	62.18
Rule-4	-	31.00	21.89	0.5	53.39
Rule-8	-	31.05	21.92	0.5	53.47
Rule-12	-	31.05	21.92	0.5	53.47
Agent	F1	**33.62**	27.49	**1.1**	**62.21**

Table 4: The reinforcement learning on Env-ConvQA with 8 history question answer pairs on the full corpus.

Env-ConvQA	Reward	F1	HEQQ	HEQD	Total
Rule-0	-	46.98	38.14	1.8	86.92
Rule-4	-	66.05	61.89	**7.3**	135.24
Rule-8	-	66.09	61.97	**7.3**	135.36
Rule-12	-	66.09	61.97	**7.3**	135.36
Agent	F1	**66.09**	**61.97**	**7.3**	**135.36**

Table 5: Different learning procedure of reinforcement learning on Env-Canard on the full corpus.

Methods	F1	HEQQ	HEQD	Total
Scheme	47.69	40.48	2.5	90.67
No scheme	43.26	35.98	1.1	80.34

We also test our method on Env-ST to examine the effectiveness if no rewritten dataset is provided. As shown in Table 3, all model performance drops drastically. The reason is that without perfect environment, the RL and rule-based agent have less satisfactory performance. But, our method can still boost the performance over all the rule-based agents, which shows the effectiveness of our policy.

We also report the results of our agent on Env-ConvQA. Note that for fair comparison, Env-ConvQA is trained on the same number of training datasets as Canard's but with no rewritten questions. As shown in Table 4, the more history turns are appended, the better performance the rule policy can obtain. But it stops increasing when we append 8 history turns. When applied our reinforcement learning scheme, the metrics can be further boosted.

Table 6: Comparison between our method and the state-of-art ConvQA methods.

Methods	F1	HEQQ	HEQD	Total
BiDAF++	51.8	45.3	2.0	99.1
BiDAF++ w/ 2-C	60.6	55.7	5.3	121,6
BERT+HAE	63.9	59.7	5.9	129.5
BERT+PosHAE	64.7	60.7	6.0	131.4
HAM	65.7	**62.1**	**7.3**	135.1
Our method	**66.1**	**62.1**	**7.3**	**135.5**

4.6 The Comparison of Training Scheme Learning and Episode Learning

Recall that our training scheme is to learn the policy from examples with only one history, followed by learning from examples with two history turns and so on so forth. As shown in Table 5, we conduct experiments with different learning methods. The agent with Episode learning interacts with environment with examples in the natural order appeared in datasets. We can see that training scheme performs much better than episode learning. A potential reason is that the training scheme can provide a warm start stage, as it firstly learns the policy from examples with less history turns, followed by learning from examples with more history turns. It can be viewed as a student learning from easy courses to the hard courses.

4.7 The Comparison of Our Method and other ConvQA Methods

We compare our method on Env-ConvQA with several state-of-the-art ConvQA methods, including: BiDAF++ [Choi *et al.*, 2018], BERT+HAE from [Qu *et al.*, 2019b], BERT+PosHAE and a strong baseline with history attention called HAM from [Qu *et al.*, 2019c]. Note that those methods consider transfer learning or data augmentation are not compared here as they used external data.

As shown in Table 6, our method can obtain the best performance in F1, HEQD and wins the first place in total score. Our method can exceed a recent history attended model called HAM method 0.5 scores in F1. Given the task is very challenging, this improvement is not small. This further shows our policy of selecting related history turns is better than other methods.

5 Related Work

Our method is related to the tasks of machine reading comprehension and conversational question answering. In this section, we briefly review the related works in these areas.

5.1 Machine Reading Comprehension (MRC) and Conversations.

MRC is at the central part of natural language understanding. Many high-quality challenges and datasets [Rajpurkar *et al.*, 2016; Rajpurkar *et al.*, 2018; Nguyen *et al.*, 2016; Joshi *et al.*, 2017; Kwiatkowski *et al.*, 2019] have greatly boosted the research progress in this field, resulting in a wide range of model architectures [Seo *et al.*, 2016; Hu *et al.*, 2018; Wang *et al.*, 2017; Huang *et al.*, 2017; Clark and Gardner, 2018]. The MRC task is typically conducted in a single-turn QA manner. The goal is to answer the question by predicting an answer span in the given passage. The ConvQA task formulated in CoQA [Reddy *et al.*, 2018] and QuAC [Choi *et al.*, 2018] is closely related to the MRC task. A major difference is that the questions in ConvQA are organized in conversations. Thus we need to incorporate the conversation history to better understand the current question. Most methods seek to incorporate modeling the dialogue history into the process of the passage representation. FlowQA [Huang *et al.*, 2018] adopts RNN to convert the passage representation from the past. FlowDelta [Yeh and Chen, 2019] seek to employ delta operation to model the change in relative turns. GraphFlow [Chen *et al.*, 2019] views each word in the passage as node and use the attention score as their connections. Then it adopts a gating mechanism to fuse the representation of the past and the current. MC^2 [Zhang, 2019] propose to use CNN in multiple perspectives to capture the semantic changes based on FlowQA. In the other hand, methods that adopt history answer embedding is also competitive. HAE [Qu *et al.*, 2019a] employs answer embedding to indicate the position the history answers. HAM [Qu *et al.*, 2019b] further adopts attention mechanism to select related history questions.

5.2 Reinforcement Learning.

Reinforcement Learning is a series of goal-oriented algorithms that has been studied for many decades in many disciplines [Sutton and Barto, 1998; Arulkumaran *et al.*, 2017; Li, 2017]. The recent development in deep learning has greatly contributed to this area and has delivered amazing achievements in many domains, such as playing games against humans [Mnih *et al.*, 2013; Silver *et al.*, 2017]. There are two lines of work in RL: value based methods and policy based methods. Value based methods, including SARSA [Rummery and Niranjan, 1994] and the Deep Q Network [Mnih *et al.*, 2015], take actions based on estimations of expected long-term return. On the other hand, policy based methods optimize for a strategy which can map states to actions that promise for the highest reward. Finally, hybrid methods, such as the actor-critic algorithm [Konda and Tsitsiklis, 2003], integrate a trained value estimator into policy based methods to reduce variance in rewards and gradients. We mainly experiment with hybrid methods in our work.

The nature of RL problems is making a sequence of actions based on certain observations in order to achieve a long-term goal. This nature has made RL suitable to deal with data selection problems in many areas [Fang *et al.*, 2017; Wu *et al.*, 2018; Fan *et al.*, 2017; Patel *et al.*, 2018; Wang *et al.*, 2018; Feng *et al.*, 2018]. The study in [Takanobu *et al.*, 2018] adopts reinforcement learning in the topic segmentation task. The study in [Buck *et al.*, 2018] adopts reinforcement learning to generate better quality of the question. It freeze the QA model and regard the seq2seq model as the agent.

Our proposed method does not make selections on the training data but aims to identify helpful conversation history to construct a better training data.

To the best of our knowledge, our work is the first research to study the problem of backtracking the helpful dialgoue his-

tories by reinforcement learning in ConvQA setting. Our proposed method is an end-to-end trainable approach that shows better results than the competitive baselines.

6 Conclusion

We proposed an unsupervised method using reinforcement learning to select related history turns for multi-turn machine reading comprehension model. Compared with modeling history in one single model, our reinforcement learning can select helpful history turns to boost the performance of MRC model. For each question in the dialogue, the learned policy can select the related history turns and performs better than rule based and episode learning policies. Extensive experiments on public datasets show our method yields consistently better performance than the competing methods.

Acknowledgment

This work was supported by the NSFC projects (No. 61402403, No. U19B2042), DAMO Academy (Alibaba Group), Alibaba-Zhejiang University Joint Institute of Frontier Technologies, Chinese Knowledge Center for Engineering Sciences and Technology, MoE Engineering Research Center of Digital Library, and the Fundamental Research Funds for the Central Universities.

References

[Arulkumaran *et al.*, 2017] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath. Deep Reinforcement Learning: A Brief Survey. *IEEE Signal Processing Magazine*, 2017.

[Buck *et al.*, 2018] Christian Buck, Jannis Bulian, Massimiliano Ciaramita, Wojciech Gajewski, Andrea Gesmundo, Neil Houlsby, and Wei Wang. Ask the Right Questions: Active Question Reformulation with Reinforcement Learning. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

[Chen *et al.*, 2019] Yu Chen, Lingfei Wu, and Mohammed J. Zaki. GraphFlow: Exploiting Conversation Flow with Graph Neural Networks for Conversational Machine Comprehension. *CoRR*, abs/1908.00059, 2019.

[Choi *et al.*, 2018] Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen tau Yih, Yejin Choi, Percy Liang, and Luke S. Zettlemoyer. Quac: Question answering in context. In *EMNLP*, 2018.

[Clark and Gardner, 2018] Christopher Clark and Matt Gardner. Simple and effective multi-paragraph reading comprehension. In *ACL*, 2018.

[Devlin *et al.*, 2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, abs/1810.04805, 2018.

[Fan *et al.*, 2017] Y. Fan, F. Tian, T. Qin, J. Bian, and T. Liu. Learning What Data to Learn. *CoRR*, 2017.

[Fang *et al.*, 2017] M. Fang, Y. Li, and T. Cohn. Learning how to Active Learn: A Deep Reinforcement Learning Approach. In *EMNLP*, 2017.

[Feng *et al.*, 2018] J. Feng, M. Huang, L. Zhao, Y. Yang, and X. Zhu. Reinforcement Learning for Relation Classification From Noisy Data. In *AAAI*, 2018.

[Hu *et al.*, 2018] Minghao Hu, Yuxing Peng, Zhen Huang, Xipeng Qiu, Furu Wei, and Ming Zhou. Reinforced mnemonic reader for machine reading comprehension. In *IJCAI*, 2018.

[Huang *et al.*, 2017] Hsin-Yuan Huang, Chenguang Zhu, Yelong Shen, and Weizhu Chen. Fusionnet: Fusing via fully-aware attention with application to machine comprehension. *CoRR*, abs/1711.07341, 2017.

[Huang *et al.*, 2018] Hsin-Yuan Huang, Eunsol Choi, and Wen-tau Yih. FlowQA: Grasping Flow in History for Conversational Machine Comprehension. *CoRR*, abs/1810.06683, 2018.

[Joshi *et al.*, 2017] Mandar S. Joshi, Eunsol Choi, Daniel S. Weld, and Luke S. Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*, 2017.

[Ju *et al.*, 2019] Ying Ju, Fubang Zhao, Shijie Chen, Bowen Zheng, Xuefeng Yang, and Yunfeng Liu. Technical report on Conversational Question Answering. *CoRR*, abs/1909.10772, 2019.

[Konda and Tsitsiklis, 2003] V. R. Konda and J. N. Tsitsiklis. On Actor-Critic Algorithms. *SIAM J. Control Optim.*, 2003.

[Kwiatkowski *et al.*, 2019] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*, 2019.

[Li, 2017] Yuxi Li. Deep reinforcement learning: An overview. *CoRR*, 2017.

[Mnih *et al.*, 2013] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. Riedmiller. Playing Atari with Deep Reinforcement Learning. *CoRR*, 2013.

[Mnih *et al.*, 2015] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. A. Riedmiller, A. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis. Human-level control through deep reinforcement learning. *Nature*, 2015.

[Nguyen *et al.*, 2016] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. Ms marco: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268, 2016.

[Patel *et al.*, 2018] Y. Patel, K. Chitta, and B. Jasani. Learning Sampling Policies for Domain Adaptation. *CoRR*, 2018.

[Qu *et al.*, 2019a] Chen Qu, Liu Yang, Minghui Qiu, W. Bruce Croft, Yongfeng Zhang, and Mohit Iyyer. BERT with History Answer Embedding for Conversational Question Answering. In *SIGIR*, pages 1133–1136, 2019.

[Qu *et al.*, 2019b] Chen Qu, Liu Yang, Minghui Qiu, Yongfeng Zhang, Cen Chen, W. Bruce Croft, and Mohit Iyyer. Attentive History Selection for Conversational Question Answering. In *CIKM*, pages 1391–1400, 2019.

[Qu *et al.*, 2019c] Chen Qu, Liu Yang, Minghui Qiu, Yongfeng Zhang, Cen Chen, W. Bruce Croft, and Mohit Iyyer. Attentive History Selection for Conversational Question Answering. *CoRR*, abs/1908.09456, 2019.

[Rajpurkar *et al.*, 2016] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100, 000+ questions for machine comprehension of text. In *EMNLP*, 2016.

[Rajpurkar *et al.*, 2018] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for squad. In *ACL*, 2018.

[Reddy *et al.*, 2018] Siva Reddy, Danqi Chen, and Christopher D. Manning. Coqa: A conversational question answering challenge. *CoRR*, abs/1808.07042, 2018.

[Rummery and Niranjan, 1994] G. A. Rummery and M. Niranjan. On-Line Q-Learning Using Connectionist Systems. Technical report, University of Cambridge, 1994.

[Seo *et al.*, 2016] Min Joon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. *CoRR*, abs/1611.01603, 2016.

[Silver *et al.*, 2017] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. R. Baker, M. Lai, A. Bolton, Y. Chen, T. P. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, and D. Hassabis. Mastering the game of Go without human knowledge. *Nature*, 2017.

[Sutton and Barto, 1998] R. S. Sutton and A. G. Barto. *Reinforcement Learning - An Introduction*. Adaptive Computation and Machine Learning. MIT Press, 1998.

[Takanobu *et al.*, 2018] Ryuichi Takanobu, Minlie Huang, Zhongzhou Zhao, Feng-Lin Li, Haiqing Chen, Xiaoyan Zhu, and Liqiang Nie. A Weakly Supervised Method for Topic Segmentation and Labeling in Goal-oriented Dialogues via Reinforcement Learning. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 4403–4410, 2018.

[Wang *et al.*, 2017] Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. Gated self-matching networks for reading comprehension and question answering. In *ACL*, 2017.

[Wang *et al.*, 2018] S. Wang, M. Yu, X. Guo, Z. Wang, T. Klinger, W. Zhang, S. Chang, G. Tesauro, B. Zhou, and J. Jiang. R^3: Reinforced Ranker-Reader for Open-Domain Question Answering. In *AAAI*, 2018.

[Williams, 1992] Ronald J. Williams. Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. *Machine Learning*, 8:229–256, 1992.

[Wu *et al.*, 2018] J. Wu, L. Li, and W. Y. Wang. Reinforced Co-Training. In *NAACL*, 2018.

[Yatskar, 2019] Mark Yatskar. A qualitative comparison of coqa, squad 2.0 and quac. In *NAACL-HLT*, pages 2318–2323, 2019.

[Yeh and Chen, 2019] Yi Ting Yeh and Yun-Nung Chen. FlowDelta: Modeling Flow Information Gain in Reasoning for Conversational Machine Comprehension. *CoRR*, abs/1908.05117, 2019.

[Zhang, 2019] Xuanyu Zhang. MCˆ2: Multi-perspective convolutional cube for conversational machine reading comprehension. In *ACL*, pages 6185–6190, 2019.

[Zhu *et al.*, 2018] Chenguang Zhu, Michael Zeng, and Xuedong Huang. Sdnet: Contextualized attention-based deep network for conversational question answering. *CoRR*, 2018.