Ranking Like Human: Global-View Matching via Reinforcement Learning for Answer Selection

Yingxue Zhang*, Ping Jian*†, Ruiying Geng*, Yuansheng Song* and Fandong Meng‡
*Beijing Institute of Technology, China
Email: {zhangyingxue, pjian, rygeng, yssong}@bit.edu.cn
†Beijing Engineering Research Center of High Volume Language Information Processing and Cloud Computing Applications, China
‡ Pattern Recognition Center, WeChat AI, Tencent Inc, China
Email: fandongmeng@tencent.com

***Abstract*—Answer Selection (AS) is of great importance for open-domain Question Answering (QA). Previous approaches typically model each pair of the question and the candidate answers independently. However, when selecting correct answers from the candidate set, the question is usually too brief to provide enough matching information for the right decision. In this paper, we propose a reinforcement learning framework that utilizes the rich overlapping information among answer candidates to help judge the correctness of each candidate. In particular, we design a policy network, whose state aggregates both the question-candidate matching information and the candidate-candidate matching information through a global-view encoder. Experiments on the benchmark of WikiQA and SelQA demonstrate that our RL framework substantially improves the ranking performance.**

***Keywords*-Answer Selection; Reinforcement Learning;**

I. Introduction

Answer Selection (AS) is an important component of open-domain Question Answering (QA). Given a question and a set of candidate answers, the objective is to select all the correct answers from the candidate set. It is also a ranking task, whose objective is to rank the correct answers in front of incorrect ones. Pre-deep learning methods address this task through modeling the question-candidate similarity (Q-C similarity), including syntactic or semantic features based methods [1–3], and structural kernels based methods [4, 5]. Recently, neural networks have shown outstanding performance on this task including attentive networks [6, 7] and compare-aggregate networks [8, 9] and so on.

These previous works typically model each pair of the question and the answer candidate independently. However, when modeled together with candidates, the question is usually too brief to provide adequate matching information, which makes it difficult to make the right decision. While in practice, a question usually corresponds to several correct candidates. These correct candidates carrying essential information are always similar to each other and easily distinguished from incorrect answers. Imagine when people encounter a complicate candidate answer, if they fail to extract enough meaningful information from the question for a safe decision, they will try to get strong hints from other easily identified answers.

Consider the question and its answer candidates given in Table I. The phrase "warehouse" in candidate C_1 also appears in the question, making C_1 an easily identified correct answer. While, other candidates have little matching information with the question, making it difficult to judge their correctness solely with the evidence provided by the question. Obviously, there is some overlapping information between C_1 and C_2/C_3, such as the *"operational"* of C_1 and C_2, the *"data"* of C_1 and C_3, while the incorrect answer C_4 is quite different from other answer candidates. Therefore, a comparison with C_1 can help distinguish other answer candidates. We observe that, when ranking the current candidate, comparing it with other candidates as humans always do can help make more accurate ranking decisions.

However, comparing the current candidate with all other candidates suffers from much noise from the incorrect answers in the set, making it challenging to leverage these overlapping information to improve ranking performance. To address the above problems, we construct a reinforcement learning (RL) framework [10] to model AS as a sequential ranking problem, where the RL agent compares the current candidate with the most reliable candidate predicted previously. In this way, we can utilize effective matching information between candidates while reducing much effect from noise of incorrect answers (We use "the highest-confidence candidate" to represent most reliable candidate predicted previously). On top of basic sentence modeling neural networks, two innovations are introduced in our framework. First, we propose a global-view comparison for answer candidates. In particular, for each answer candidate, we find the highest-confidence candidate from its previous candidates (which have been ranked in earlier time steps), then the system encodes the current candidate with both the question and the highest-confidence candidate as the global matching information. Second, we design a policy network for ranking incrementally, whose state is the global matching information mentioned above. The policy network decides whether the current candidate is correct (action) at each state, then obtains a reward for guiding the learning. The agent gets a positive reward only when the current decision improves the overall performance of ranking. Our RL framework can be easily applied to various basic neural networks,

Table I
A REAL CASE IN WIKIQA.

Question		Candidate	Label
what are **warehouse spreadsheets** used for?	C_1	The ***data stored*** in the **warehouse** are uploaded from the ***operational*** systems .(such as marketing , sales etc. , shown in the figure to the right)	**True**
	C_2	The ***data*** may pass through an ***operational data store*** for additional ***operations*** before they are used in the DW for reporting.	**True**
	C_3	The access layer helps users ***retrieve data***.	**True**
	C_4	The combination of facts and dimensions is sometimes called a star schema.	**False**

producing more accurate results than models that merely consider the current question-candidate pair and make local decisions.

Our major contributions are three-fold.

- We propose a RL framework for Answer Selection to utilize the matching information among candidates, which can be easily applied on various sentence modeling neural networks.
- The proposed RL framework allows models to extract global matching information from both the question-candidate pair and the highest-confidence candidate identified in earlier states through the global-view encoder;
- We apply the proposed RL framework to several sentence modeling neural networks. Experimental results show that it gains significant and consistent improvements on the benchmarks of WikiQA and SelQA.

II. RELATED WORK

A. Answer Selection

Answer Selection (AS) has enjoyed wide popularity in natural language processing. Most previous methods typically model the AS as a text matching problem, including conventional feature engineering based approaches [1–3] and deep learning models [6, 7]. Feature engineering based methods tend to design intricate linguistic features, such as syntactic features extracted from dependency and constituency tree [3], or semantic features extracted from WordNet [11], which is time-consuming and inefficient. Most rencent AS systems are based on deep neural networks, which have earned a much promising break. These DNN based models can be roughly divided into two categories, the attention-based models [12, 13] and the compare-aggregate based models [8, 14]. Some recent studies focus on employing external resource to improve the ranking performance. [15] propose the EviNets that extracts relevant evidence from external text corpora by a search system and retrieves KB triples based on entities tagged by a entity linking system before each prediction, which improves the performance of answer selection but is also time-consuming. [16] propose a kernel-based method that utilizes external resources such as constituency/dependency tree and semantic features from WordNet to model the similarity between question-candidate pairs, which requires intricate feature engineering work. In this paper, we propose a reinforcement learning algorithm where we assign the highest-confidence candidate and we utilize it to support answer selection. Compared with previous methods, our model needs neither intricate feature engineering work nor external resources.

B. Reinforcement Learning

Reinforcement Learning (RL) [17] has shown promising results in many natural language processing tasks [18]. [19] regard the pronoun resolution as a sequence decision problem. They argued that modeling useful information of preceding potential antecedents is crucial for classifying later zero pronoun-candidate antecedent pairs and used policy gradient to optimize the task. [20] explore a deep reinforcement learning strategy to generate the false-positive indicator for distant supervision relation extraction. [21] use reinforcement learning to optimize a new objective function with a reward defined by the property of the NLI datasets to make full use of the discourse information. In this work, we propose a RL framework for Answer Selection, which makes full use of the answer candidates to select correct answers. Our RL framework allows models to extract global matching information from both the Q-A pair and the answer candidates identified in earlier states through the global-view comparison mechanism.

III. MODEL

As figure 1 shows, the proposed framework consists of two components: (1) The global-view encoder, which contains two matching modules, the question-candidate matching module and the candidate-candidate matching module. We concatenated these two views of matching information as the RL state and feed it into the RL agent.(2) The RL agent, which follows a policy to decide which action is chosen at each state and obtains a reward for policy learning.

A. Global-View Encoder

The highest-confidence candidate We first define the highest-confidence candidate which is used to help identify the correctness of the current candidate. Suppose we are predicting the t-th candidate C_t, the highest-confidence candidate is the candidate which is most likely to be positive in $\{C_1, C_2, ...C_{t-1}\}$ predicted in earlier time steps. Since the first candidate C_1 has no antecedents, we use the question to initialize the highest-confidence candidate. After obtaining the highest-confidence candidate, we encode two views of matching information with two matching modules.

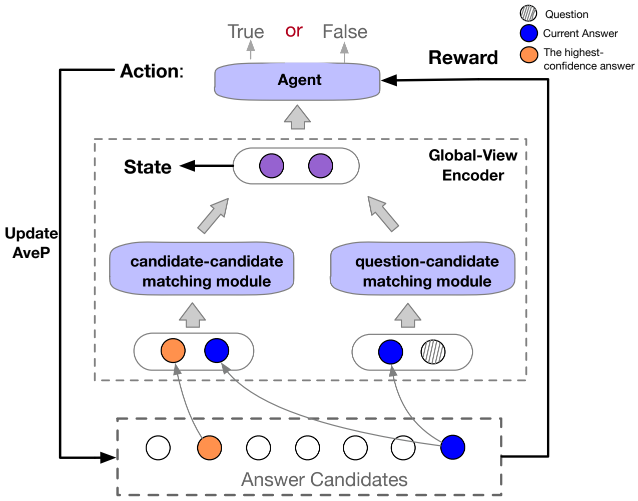

Figure 1. The architecture of proposed RL framework. The global-view encoder consists of two matching modules: the question-candidate matching module and the candidate-candidate matching module.

The matching Module The question-candidate matching module and the candidate-candidate matching module are two separate matching modules with the same network architecture. Each one has its own parameters. We apply the compare-aggregate network proposed by [9] as the matching module, which is effective for a wide range of sequence matching problems. We briefly introduce its components:

(1) The embedding module. We initialize the question Q as $\{\mathbf{x}_{q_1}, ..., \mathbf{x}_{q_{|Q|}}\}$ and initialize the current candidate X_C as $\{\mathbf{x}_{c_1}, \mathbf{x}_{c_2}, ..., \mathbf{x}_{c_{|C|}}\}$. Here, $\mathbf{x}_{q_1}, \mathbf{x}_{q_2}, ..., \mathbf{x}_{q_{|Q|}}$ and $\mathbf{x}_{c_1}, \mathbf{x}_{c_2}, ..., \mathbf{x}_{c_{|C|}}$ are pre-trained word embeddings. $|Q|$ and $|C|$ respectively represent the numbers of words in the question and the candidate.

(2) The attention module. It generates a question-aware candidate representation $H = \{\mathbf{h}_{c_1}, \mathbf{h}_{c_2}, ..., \mathbf{h}_{c_{|C|}}\}$ through the attention mechanism [22], where each $\mathbf{h}_c$ is an attention-weighted sum of the vectors of Q:

$$\alpha_{ij} = \mathbf{x}_{c_i} \cdot \mathbf{x}_{q_j} \tag{1}$$

$$p_{ij} = \frac{e^{\alpha_{ij}}}{\sum_{j=1}^{|Q|} e^{\alpha_{ij}}} \tag{2}$$

$$h_{c_i} = \sum_{j=1}^{|Q|} p_{ij}\mathbf{x}_{q_j} \tag{3}$$

where $\cdot$ represents dot product, and h_{c_i} represents the attention representation of the i-th word of the candidate answer.

(3) The comparison module. In this module, each pair of $\mathbf{x}_{c_i}$ and $\mathbf{h}_{c_i}$ are combined into a vector $\mathbf{v}_{c_i}$ through a comparison function $\mathbb{F}$, which can be "element-wise multiplication" and "element-wise subtraction". In this paper, we use "element-wise multiplication" as the comparison function:

$$\mathbf{v}_{c_i} = \mathbf{x}_{c_i} \odot \mathbf{h}_{c_i} \tag{4}$$

where the $\odot$ represents the element-wise multiplication.

(4) The aggregate module. The vectors $\mathbf{V} = \{\mathbf{v}_{c_1}\mathbf{v}_{c_2}, ..., \mathbf{v}_{c_{|C|}}\}$ obtained after comparison are aggregated into a final vector $\mathbf{M}$ for the classification. The aggregate module can be various networks. In this paper, we apply the Bidirectional Gated Recurrent Unit network (BiGRU) [23] to the aggregate module. We feed $\mathbf{V}$ into the BiGRU and concatenate the last hidden states of the BiGRU in two directions to obtain the final vector M.

Global-View Matching The global-view encoder extracts adequate matching information through global-view comparison. It consists of two parts: the Q-C CompAgg module that compares the current candidate C_{cur} with the question, and the C-C CompAgg module compares C_{cur} with C_{pre}, where C_{pre} represents the highest-confidence candidate. When dealing with a specific candidate, we obtain the matching information as:

$$M_{qc} = \mathbf{CompAgg}(Q, C_{cur}) \tag{5}$$

$$M_{cc} = \mathbf{CompAgg}(C_{pre}, C_{cur}) \tag{6}$$

These two matching information are combined as follows and sent to the RL agent.

$$s_t = \mathbf{Concat}(M_{qc}, M_{cc}) \tag{7}$$

B. Policy Network for Answer Selection

From a global perspective of ranking, we design a policy network which can leverage the information of reliable candidates identified in the earlier state.

The policy network adopts a stochastic policy $\pi(a_t|s_t;\Theta)$, where s_t and a_t represent the state and the action at time t respectively. Here, we briefly introduce the state, the action, and the reward used to guide the policy learning at time *t*.

State The state is generated as Eq.(7), which combines the current Q-C matching information and the previous reliable candidate information.

Action The action space is defined as $\{pos, neg\}$, where the "*pos*" indicates the current candidate is correct while the "*neg*" indicates it is incorrect. The degree of "*pos*", i,e., $p(a_t = pos|s_t)$ corresponds to the confidence of the current candidate, which is used to determine the ranking position of the current candidate. If the current candidate reaches the highest confidence so far, it will replace the previous highest-confidence candidate C_{pre} and become a new C_{pre} for later time steps. Specifically, the agent makes an action based on the current state through an MLP.

$$h_1(s_t) = \tanh(W_1 s_t + b_1) \tag{8}$$

$$p(a_t|s_t) = \mathtt{softmax}(W_2 h_1(s_t) + b_2) \tag{9}$$

Rewards After sampling an action with the probability at each state, the agent obtains a reward for guiding the policy learning. Since the task is modeled as a sequential ranking problem, we design the reward as the difference of the Average Precision(AveP) before and after each action. We use AP_t to represent the AveP at time step t, then the function of the reward $R(a_{1:T})$ is as follows:

$$R(a_t) = \begin{cases} 0 & t = 1 \\ AP_t - AP_{t-1} & 1 < t \leq T \end{cases} \tag{10}$$

Table II
THE STATISTICS OF WIKIQA AND SELQA.

	WikiQA		**SelQA**	
	#Q	#QA	#Q	#QA
Train	873	8,627	5,529	66,438
Dev	126	1,130	785	9,377
Test	243	2,351	1,590	19,435

Table III
RESULTS ON WIKIQA AND SELQA.

Model	**WikiQA**		**SelQA**	
	MAP	**MRR**	**MAP**	**MRR**
Shen et al.v2017 [26]	71.07	73.04	89.14	89.83
Tay et al. 2018 [27]	71.20	72.70	-	-
Nicosia et al. 2018 [28]	72.24	73.91	-	-
Shao et al. 2019 [29]	69.41	70.77	-	-
our model	**73.39**	**74.45**	**90.39**	**90.95**

where,

$$AP_t = \frac{1}{N}\sum_{n=1}^{N}\frac{n}{index(n)} \tag{11}$$

Here, N represents the numbers of correct answers in the first t candidates. "$index(n)$" stands for the ranking position of the n-th correct answer.

Objective Function We optimize the parameters of the policy network using REINFORCE algorithm[17], which aims to maximize the expected reward:

$$\begin{aligned} J(\theta) &= \mathbb{E}_{a_{1:T}\sim p_\theta(a_t|Q,C_{cur},C_{pre})}R(a_{1:T}) \\ &= \textstyle\sum_t \sum_a p_\theta(a_t|Q,C_{cur},C_{pre})R(a_t) \end{aligned} \tag{12}$$

Hence, we update the policy network with the following gradient:

$$\bigtriangledown_\theta J(\theta) = \bigtriangledown_\theta \sum_{t=1}^{T} R(a_t)\log p_\theta(a_t|Q,C_{cur},C_{pre}) \tag{13}$$

IV. EXPERIMENTS

A. Datasets and Evaluation Metric

We conduct experiments on two datasets: SelQA and WikiQA.

WikiQA Dataset The WikiQA dataset [24] is an open domain question-answering dataset constructed from real queries of Bing and Wikipedia. We remove all questions with no correct answers as predecessors [9, 24] do. The distribution statistics of the question and candidates are shown in table II.

SelQA Dataset The SelQA dataset [25] consists of questions generated through crowd-sourcing and sentence length answers that are drawn from the ten most prevalent topics in the English Wikipedia. SelQA is much larger than WikiQA. As table II shows, the numbers of its Q-A pairs is about nine times of WikiQA's.

Evaluation Metric We evaluate models by Mean Average Precision (MAP) and Mean Reciprocal Rank (MRR), which are commonly used metrics in this task.

B. Experimental Setup

We introduce experimental setups including baselines and implementation details in this section. We set up the following baselines and comparison systems to show the effectiveness of our method.

Shen et al. 2017 [26] propose an adaptive convolutional filter generation framework for question-answer sentence pairs modeling, by leveraging a meta network to generate input-aware filters.

Tay et al. 2018 [27] propose a novel deep learning architecture, HyperQA, for fast and efficient question-answer ranking and retrieval.

Nicosia et al. 2018 [28] take advantage of small amounts of labelled data that model semantic phenomena in text to encode matching features directly in the word representations.

Shao et al. 2019 [29] design a Transformer-based neural network for answer selection, where they deploy a bidirectional long short-term memory (BiLSTM) behind the Transformer to acquire both global information and sequential features in the question or answer sentence.

To further prove the effectiveness of our RL framework, we also build some matching neural networks to do a fully comparison:

ABCNN: An attention-based CNN proposed by [7].

CompAgg-GRU: The CompAgg-GRU is the matching neural network described in SectionIII-A, which is used as the matching module in our framework.

CompAgg-CNN: CompAgg-CNN is also based on the compare-aggregate framework described in SectionIII-A. But, we replace the BiGRU based aggregation module with CNN based one.

We conduct two experimental settings for these matching models to see whether they achieve higher performance after applying our RL framework. One experimental setting is using these models to simply encode the current question-candidate pair independently for answer selection and trained by supervised learning. The other is putting these models into our RL framework. The two settings use the same set of hyper-parameters. The comparison results are shown in Table III.

Implementation Details We initialize word embeddings with 300-dimensional-GloVe vectors [30]. The mini-batch contains Q-A pairs of 10 different questions. We set the hidden size of GRU to 128, the kernel window sizes of CNN in CompAgg-CNN to [1, 2, 3, 4, 5] with 150 hidden units, the learning rate to $1e^{-4}$ which decayed after every epoch by a factor of 0.95. All these hyper-parameters are optimized on the development set.

C. Main Results and Discussion

Table III shows the results on the benchmarks of WikiQA and SelQA. Our method achieves comparable performance to the recent powerful answer selection model. Compared with the baselines in Table III, we achieve 1.15% improvements of MAP and 0.54% improvements of MRR on WikiQA, and also achieves 1.25% improvements of MAP and 1.12% improvements of MRR on SelQA.

Table IV
COMPARISON OF MATCHING MODELS WITH AND WITHOUT OUR RL FRAMEWORK

Model	WikiQA		SelQA	
	MAP	MRR	MAP	MRR
ABCNN	68.71	70.28	82.14	82.93
ABCNN+RL	**69.63**	**71.02**	**83.08**	**83.71**
CompAgg-CNN	70.24	71.47	86.48	86.90
CompAgg-CNN+RL	**72.17**	**73.49**	**87.21**	**87.76**
CompAgg-GRU	71.28	72.59	89.90	90.40
CompAgg-GRU+RL (our model)	**73.39**	**74.45**	**90.39**	**90.95**

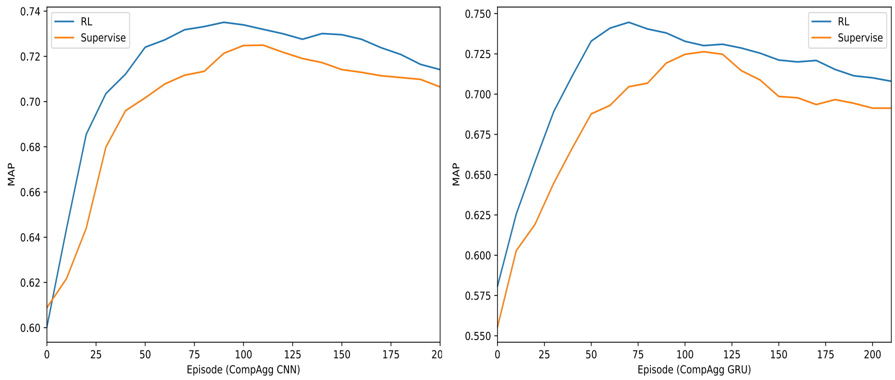

Figure 2. Comparison of training speed on development set of WikiQA in two experiment settings.

Our method not only utilizes the information of question, but also utilize the overlapping information between candidates, which fills the gap between the question and candidates. This is the main reason why our method outperforms them.

Table IV shows the results of models with and without our RL framework. We can see that the proposed RL framework enhances all the baselines both on the small-scale corpus WikiQA and the large-scale corpus SelQA. Under the same corpus, our RL framework gains stable improvement on different models, which proves its generalization.

To further confirm the validity of our model, we also visualize the training speed with and without the proposed RL framework for CompAgg-CNN and CompAgg-GRU under the same hyper-parameters set. Figure 2 shows the comparison of training speed on the development set of WikiQA. Models enhanced by the RL framework achieve better performance in less time. We believe that the global matching information, which is obtained by referring to the highest-confidence candidate ranked previously, fills the gap between the question and current candidate and makes training faster and easier.

V. CONCLUSION

We propose a reinforcement learning framework to model Answer Selection as a sequential ranking problem, which can refer to the high-confidence answer ranked previously and extract global matching information for better decisions. Experiments on several representation models show the effectiveness of our RL framework. For future work, we'd like to explore a more powerful way to better fuse the global matching information to improve the effectiveness of our RL framework.

ACKNOWLEDGMENT

The authors would like to thank the organizers of IALP 2019 and the reviewers for their helpful suggestions. This research work is supported by the National Key Research and Development Program of China under Grant No.2017YFB1002103, the National Natural Science Foundation of China under Grant No.61751201, and the Research Foundation of Beijing Municipal Science & Technology Commission under Grant No.Z181100008918002.

REFERENCES

[1] M. Heilman and N. A. Smith, "Tree edit models for recognizing textual entailments, paraphrases, and answers to questions," in *NAACL*, 2010.

[2] M. Wang, N. A. Smith, and T. Mitamura, "What is the jeopardy model? a quasi-synchronous grammar for qa," in *EMNLP-CoNLL*, 2007.

[3] W.-t. Yih, M.-W. Chang, C. Meek, and A. Pastusiak, "Question answering using enhanced lexical semantic models," in *ACL*, 2013.

[4] A. Severyn, M. Nicosia, and A. Moschitti, "Building structures from classifiers for passage reranking," in *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, 2013.

[5] K. Tymoshenko, A. Moschitti, and A. Severyn, "Encoding semantic resources in syntactic structures for passage reranking," in *EACL*, 2014.

[6] A. P. Parikh, O. Täckström, D. Das, and J. Uszkoreit, "A decomposable attention model for natural language inference," *arXiv preprint arXiv:1606.01933*, 2016.
[7] W. Yin, H. Schütze, B. Xiang, and B. Zhou, "Abcnn: Attention-based convolutional neural network for modeling sentence pairs," *Transactions of the Association for Computational Linguistics*, 2016.
[8] W. Bian, S. Li, Z. Yang, G. Chen, and Z. Lin, "A compare-aggregate model with dynamic-clip attention for answer selection," in *CIKM*, 2017.
[9] S. Wang and J. Jiang, "A compare-aggregate model for matching text sequences," *CoRR*, vol. abs/1611.01747, 2017.
[10] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," *IEEE Transactions on Neural Networks*, 1988.
[11] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, 1995.
[12] L. Sha, X. Zhang, F. Qian, B. Chang, and Z. Sui, "A multi-view fusion neural network for answer selection," in *AAAI*, 2018.
[13] B. Wang, K. Liu, and J. Zhao, "Inner attention based recurrent neural networks for answer selection," in *ACL*, 2016.
[14] Z. Wang, H. Mi, and A. Ittycheriah, "Sentence similarity learning by lexical decomposition and composition," *arXiv preprint arXiv:1602.07019*, 2016.
[15] D. Savenkov and E. Agichtein, "Evinets: Neural networks for combining evidence signals for factoid question answering," in *ACL*, 2017.
[16] K. Tymoshenko and A. Moschitti, "Cross-pair text representations for answer sentence selection," in *EMNLP*, 2018.
[17] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, 1992.
[18] D. Mollá, "Towards the use of deep reinforcement learning with global policy for query-based extractive summarisation," *arXiv preprint arXiv:1711.03859*, 2017.
[19] Q. Yin, Y. Zhang, W.-N. Zhang, T. Liu, and W. Y. Wang, "Deep reinforcement learning for Chinese zero pronoun resolution," in *ACL*, 2018.
[20] P. Qin, W. XU, and W. Y. Wang, "Robust distant supervision relation extraction via deep reinforcement learning," in *ACL*, 2018.
[21] B. Pan, Y. Yang, Z. Zhao, Y. Zhuang, D. Cai, and X. He, "Discourse marker augmented network with reinforcement learning for natural language inference," in *ACL*, 2018.
[22] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, 2014.
[23] K. Cho, B. van Merrienboer, aglar Gülehre, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," in *EMNLP*, 2014.
[24] Y. Yang, W.-t. Yih, and C. Meek, "Wikiqa: A challenge dataset for open-domain question answering," in *EMNLP*, 2015.
[25] T. Jurczyk, M. Zhai, and J. D. Choi, "Selqa: A new benchmark for selection-based question answering," *ICTAI*, 2016.
[26] D. Shen, M. R. Min, Y. Li, and L. Carin, "Adaptive convolutional filter generation for natural language understanding," *arXiv preprint arXiv:1709.08294*, 2017.
[27] Y. Tay, L. A. Tuan, and S. C. Hui, "Hyperbolic representation learning for fast and efficient neural question answering," in *WSDM*, 2018.
[28] M. Nicosia and A. Moschitti, "Semantic linking in convolutional neural networks for answer sentence selection," in *EMNLP*, 2018.
[29] T. Shao, Y. Guo, H. Chen, and Z. Hao, "Transformer-based neural network for answer selection in question answering," *IEEE Access*, 2019.
[30] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *EMNLP*, 2014.