A Co-Matching Model for Multi-choice Reading Comprehension

Shuohang Wang[1], **Mo Yu**[2], **Shiyu Chang**[2], **Jing Jiang**[1]
[1]School of Information System, Singapore Management University [2]IBM Research
{shwang.2014,jingjiang}@smu.edu.sg
yum@us.ibm.com, shiyu.chang@ibm.com

Abstract

Multi-choice reading comprehension is a challenging task, which involves the matching between a passage and a question-answer pair. This paper proposes a new *co-matching* approach to this problem, which jointly models whether a passage can match both a question and a candidate answer. Experimental results on the RACE dataset demonstrate that our approach achieves state-of-the-art performance.

1 Introduction

Enabling machines to understand natural language text is arguably the ultimate goal of natural language processing, and the task of machine reading comprehension is an intermediate step towards this ultimate goal (Richardson et al., 2013; Hermann et al., 2015; Hill et al., 2015; Rajpurkar et al., 2016; Nguyen et al., 2016). Recently, Lai et al. (2017) released a new multi-choice machine comprehension dataset called RACE that was extracted from middle and high school English examinations in China. Figure 1 shows an example passage and two related questions from RACE. The key difference between RACE and previously released machine comprehension datasets (*e.g.*, the CNN/Daily Mail dataset (Hermann et al., 2015) and SQuAD (Rajpurkar et al., 2016)) is that the answers in RACE often cannot be directly extracted from the given passages, as illustrated by the two example questions (Q1 & Q2) in Figure 1. Thus, answering these questions is more challenging and requires more inferences.

Previous approaches to machine comprehension are usually based on pairwise sequence matching, where either the passage is matched against the sequence that concatenates both the question and a candidate answer (Yin et al., 2016), or the passage is matched against the question alone followed by a second step of selecting an answer using the matching result of the first step (Lai et al., 2017; Zhou et al., 2018). However, these approaches may not be suitable for multi-choice reading comprehension since questions and answers are often equally important. Matching the passage only against the question may not be meaningful and may lead to loss of information from the original passage, as we can see from the first example question in Figure 1. On the other hand, concatenating the question and the answer into a single sequence for matching may not work, either, due to the loss of interaction information between a question and an answer. As illustrated by Q2 in Figure 1, the model may need to recognize what "he" and "it" in candidate answer (c) refer to in the question, in order to select (c) as the correct answer. This observation of the RACE dataset shows that we face a new challenge of matching sequence triplets (*i.e.*, passage, question and answer) instead of pairwise matching.

In this paper, we propose a new model to match a question-answer pair to a given passage. Our *co-matching* approach explicitly treats the question and the candidate answer as two sequences and jointly matches them to the given passage. Specifically, for each position in the passage, we compute two attention-weighted vectors, where one is from the question and the other from the candidate answer. Then, two matching representations are constructed: the first one matches the passage with the question while the second one matches the passage with the candidate answer. These two newly constructed matching representations together form a *co-matching state*. Intuitively, it encodes the locational information of the question and the candidate answer matched to a specific context of the passage. Finally, we apply a hierar-

Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 746–751
Melbourne, Australia, July 15 - 20, 2018.

Passage: *My father wasn't a king, he was a taxi driver, but I am a prince-Prince Renato II, of the country Pontinha , an island fort on Funchal harbour. In 1903, the king of Portugal sold the land to a wealthy British family, the Blandys, who make Madeira wine. Fourteen years ago the family decided to sell it for just EUR25,000, but nobody wanted to buy it either. I met Blandy at a party and he asked if I'd like to buy the island. Of course I said yes, but I had no money-I was just an art teacher. I tried to find some business partners, who all thought I was crazy. So I sold some of my possessions, put my savings together and bought it. Of course, my family and my friends-all thought I was mad ... If l want to have a national flag, it could be blue today, red tomorrow. ... My family sometimes drops by, and other people come every day because the country is free for tourists to visit ...*	
Q1: *Which statement of the following is true?*	Q2: *How did the author get the island?*
a. *The author made his living by driving.*	a. *It was a present from Blandy.*
b. *The author's wife supported to buy the island.*	b. *The king sold it to him.*
c. *Blue and red are the main colors of his national flag.*	**c. *He bought it from Blandy.***
d. *People can travel around the island free of charge.*	d. *He inherited from his father.*

Table 1: An example passage and two related multi-choice questions. The ground-truth answers are in **bold**.

chical LSTM (Tang et al., 2015) over the sequence of co-matching states at different positions of the passage. Information is aggregated from word-level to sentence-level and then from sentence-level to document-level. In this way, our model can better deal with the questions that require evidence scattered in different sentences in the passage. Our model improves the state-of-the-art model by 3 percentage on the RACE dataset. Our code will be released under https://github.com/shuohangwang/comatch.

2 Model

For the task of multi-choice reading comprehension, the machine is given a passage, a question and a set of candidate answers. The goal is to select the correct answer from the candidates. Let us use $\mathbf{P} \in \mathbb{R}^{d\times P}$, $\mathbf{Q} \in \mathbb{R}^{d\times Q}$ and $\mathbf{A} \in \mathbb{R}^{d\times A}$ to represent the passage, the question and a candidate answer, respectively, where each word in each sequence is represented by an embedding vector. d is the dimensionality of the embeddings, and P, Q, and A are the lengths of these sequences.

Overall our model works as follows. For each candidate answer, our model constructs a vector that represents the matching of $\mathbf{P}$ with both $\mathbf{Q}$ and $\mathbf{A}$. The vectors of all candidate answers are then used for answer selection. Because we simultaneously match $\mathbf{P}$ with $\mathbf{Q}$ and $\mathbf{A}$, we call this a *co-matching* model. In Section 2.1 we introduce the word-level co-matching mechanism. Then in Section 2.2 we introduce a hierarchical aggregation process. Finally in Section 2.3 we present the objective function. An overview of our co-matching model is shown in Figure 2.

2.1 Co-matching

The co-matching part of our model aims to match the passage with the question and the candidate answer at the word-level. Inspired by some previous work (Wang and Jiang, 2016; Trischler et al., 2016), we first use bi-directional LSTMs (Hochreiter and Schmidhuber, 1997) to pre-process the sequences as follows:

$$\mathbf{H}^{\mathrm{p}} = \text{Bi-LSTM}(\mathbf{P}), \mathbf{H}^{\mathrm{q}} = \text{Bi-LSTM}(\mathbf{Q}),$$
$$\mathbf{H}^{\mathrm{a}} = \text{Bi-LSTM}(\mathbf{A}), \quad (1)$$

where $\mathbf{H}^{\mathrm{p}} \in \mathbb{R}^{l\times P}$, $\mathbf{H}^{\mathrm{q}} \in \mathbb{R}^{l\times Q}$ and $\mathbf{H}^{\mathrm{a}} \in \mathbb{R}^{l\times A}$ are the sequences of hidden states generated by the bi-directional LSTMs. We then make use of the attention mechanism to match each state in the passage to an aggregated representation of the question and the candidate answer. The attention vectors are computed as follows:

$$\begin{aligned}
\mathbf{G}^{\mathrm{q}} &= \text{SoftMax}\left((\mathbf{W}^{\mathrm{g}}\mathbf{H}^{\mathrm{q}} + \mathbf{b}^{\mathrm{g}} \otimes \mathbf{e}_Q)^{\mathrm{T}}\mathbf{H}^{\mathrm{p}}\right), \\
\mathbf{G}^{\mathrm{a}} &= \text{SoftMax}\left((\mathbf{W}^{\mathrm{g}}\mathbf{H}^{\mathrm{a}} + \mathbf{b}^{\mathrm{g}} \otimes \mathbf{e}_Q)^{\mathrm{T}}\mathbf{H}^{\mathrm{p}}\right), \\
\overline{\mathbf{H}}^{\mathrm{q}} &= \mathbf{H}^{\mathrm{q}}\mathbf{G}^{\mathrm{q}}, \\
\overline{\mathbf{H}}^{\mathrm{a}} &= \mathbf{H}^{\mathrm{a}}\mathbf{G}^{\mathrm{a}}, \quad (2)
\end{aligned}$$

where $\mathbf{W}^{\mathrm{g}} \in \mathbb{R}^{l\times l}$ and $\mathbf{b}^{\mathrm{g}} \in \mathbb{R}^{l}$ are the parameters to learn. $e_Q \in \mathbb{R}^{\mathrm{Q}}$ is a vector of all 1s and it is used to repeat the bias vector into the matrix. $\mathbf{G}^{\mathrm{q}} \in \mathbb{R}^{Q\times P}$ and $\mathbf{G}^{\mathrm{a}} \in \mathbb{R}^{A\times P}$ are the attention

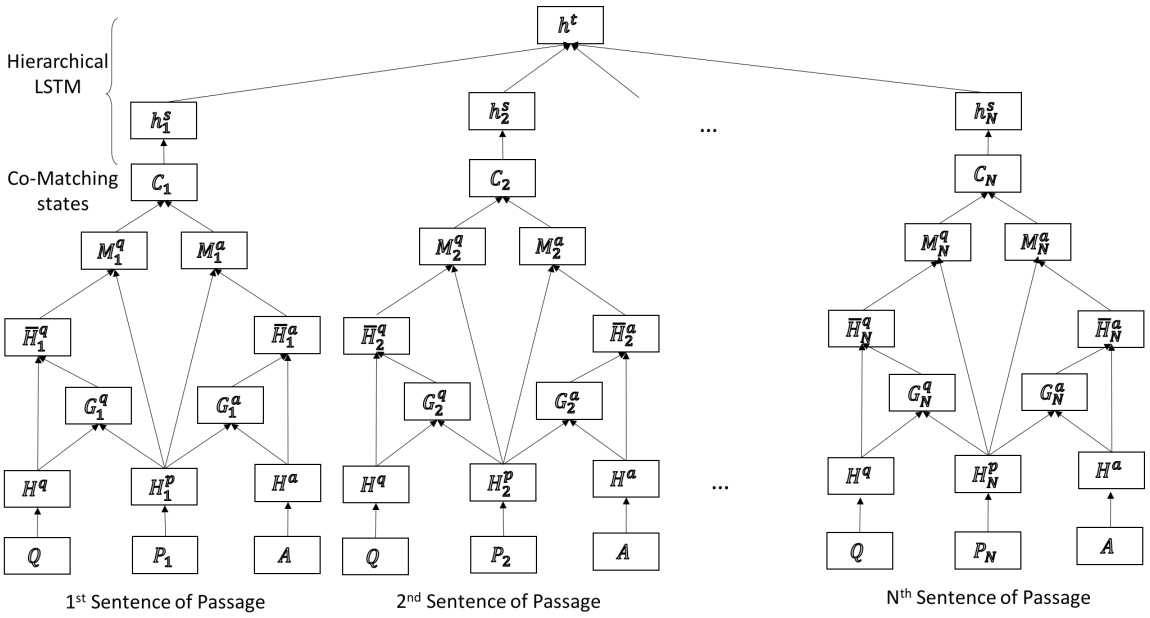

Figure 1: An overview of the model that builds a matching representation for a triplet $\{\mathbf{P}, \mathbf{Q}, \mathbf{A}\}$ (*i.e.*, passage, question and candidate answer).

weights assigned to the different hidden states in the question and the candidate answer sequences, respectively. $\overline{\mathbf{H}}^{\mathrm{q}} \in \mathbb{R}^{l \times P}$ is the weighted sum of all the question hidden states and it represents how the question can be aligned to each hidden state in the passage. So is $\overline{\mathbf{H}}^{\mathrm{a}} \in \mathbb{R}^{l \times P}$. Finally we can co-match the passage states with the question and the candidate answer as follows:

$$\begin{aligned}
\mathbf{M}^{\mathrm{q}} &= \mathrm{ReLU}\left(\mathbf{W}^{\mathrm{m}}\begin{bmatrix}\overline{\mathbf{H}}^{\mathrm{q}} \ominus \mathbf{H}^{\mathrm{p}} \\ \overline{\mathbf{H}}^{\mathrm{q}} \otimes \mathbf{H}^{\mathrm{p}}\end{bmatrix} + \mathbf{b}^{\mathrm{m}}\right), \\
\mathbf{M}^{\mathrm{a}} &= \mathrm{ReLU}\left(\mathbf{W}^{\mathrm{m}}\begin{bmatrix}\overline{\mathbf{H}}^{\mathrm{a}} \ominus \mathbf{H}^{\mathrm{p}} \\ \overline{\mathbf{H}}^{\mathrm{a}} \otimes \mathbf{H}^{\mathrm{p}}\end{bmatrix} + \mathbf{b}^{\mathrm{m}}\right), \\
\mathbf{C} &= \begin{bmatrix}\mathbf{M}^{\mathrm{q}} \\ \mathbf{M}^{\mathrm{a}}\end{bmatrix}, \qquad (3)
\end{aligned}$$

where $\mathbf{W}^{\mathrm{g}} \in \mathbb{R}^{l \times 2l}$ and $\mathbf{b}^{\mathrm{g}} \in \mathbb{R}^{l}$ are the parameters to learn. $\begin{bmatrix}\cdot \\ \cdot\end{bmatrix}$ is the column-wise concatenation of two matrices, and $\cdot \ominus \cdot$ and $\cdot \otimes \cdot$ are the element-wise subtraction and multiplication between two matrices, which are used to build better matching representations (Tai et al., 2015; Wang and Jiang, 2017). $\mathbf{M}^{\mathrm{q}} \in \mathbb{R}^{l \times P}$ represents the matching between the hidden states of the passage and the corresponding attention-weighted representations of the question. Similarly, we match the passage with the candidate answer and represent the matching results using $\mathbf{M}^{\mathrm{a}} \in \mathbb{R}^{l \times P}$. Finally $C \in \mathbb{R}^{2l \times P}$ is the concatenation of $\mathbf{M}^{\mathrm{q}} \in \mathbb{R}^{l \times P}$ and $\mathbf{M}^{\mathrm{a}} \in \mathbb{R}^{l \times P}$ and represents how each passage state can be matched with the question and the candidate answer. We refer to $\mathbf{c} \in \mathbb{R}^{2l}$, which is a single column of $\mathbf{C}$, as a *co-matching state* that concurrently matches a passage state with both the question and the candidate answer.

2.2 Hierarchical Aggregation

In order to capture the sentence structure of the passage, we further modify the model presented earlier and build a hierarchical LSTM (Tang et al., 2015) on top of the co-matching states. Specifically, we first split the passage into sentences and we use $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_N$ to represent these sentences, where N is the number of sentences in the passage. For each triplet $\{\mathbf{P}_n, \mathbf{Q}, \mathbf{A}\}, n \in [1, N]$, we can get the co-matching states $\mathbf{C}_n$ through Eqn. (1-3). Then we build a bi-directional LSTM followed by max pooling on top of the co-matching states of each sentence as follows:

$$\mathbf{h}^{\mathrm{s}}_n = \mathrm{MaxPooling}\left(\text{Bi-LSTM}\left(\mathbf{C}_n\right)\right), \quad (4)$$

where the function $\mathrm{MaxPooling}(\cdot)$ is the row-wise max pooling operation. $\mathbf{h}^{\mathrm{s}}_n \in \mathbb{R}^{l}, n \in [1, N]$ is the sentence-level aggregation of the co-matching states. All these representations will be further integrated by another Bi-LSTM to get the final triplet matching representation.

$$\begin{aligned}
\mathbf{H}^{\mathrm{s}} &= [\mathbf{h}^{s}_1; \mathbf{h}^{s}_2; \ldots; \mathbf{h}^{s}_N], \\
\mathbf{h}^{\mathrm{t}} &= \mathrm{MaxPooling}\left(\text{Bi-LSTM}\left(\mathbf{H}^{\mathrm{s}}\right)\right), \quad (5)
\end{aligned}$$

	RACE-M	RACE-H	RACE
Random	24.6	25.0	24.9
Sliding Window	37.3	30.4	32.2
Stanford AR	44.2	43.0	43.3
GA	43.7	44.2	44.1
ElimiNet	-	-	44.7
HAF	45.3	47.9	47.2
MUSIC	51.5	45.7	47.4
Hier-Co-Matching	**55.8***	**48.2***	**50.4***
- Hier-Aggregation	54.2	46.2	48.5
- Co-Matching	50.7	45.6	46.4
Turkers	85.1	69.4	73.3
Ceiling	95.4	94.2	94.5

Table 2: Experiment Results. * means it's significant to the models ablating either the hierarchical aggregation or co-matching state.

where $\mathbf{H}^{\mathrm{s}} \in \mathbb{R}^{l \times N}$ is the concatenation of all the sentence-level representations and it is the input of a higher level LSTM. $\mathbf{h}^{\mathrm{t}} \in \mathbb{R}^{l}$ is the final output of the matching between the sequences of the passage, the question and the candidate answer.

2.3 Objective function

For each candidate answer $\mathbf{A}_i$, we can build its matching representation $\mathbf{h}_i^t \in \mathbb{R}^l$ with the question and the passage through Eqn. (5). Our loss function is computed as follows:

$$L(\mathbf{A}_i|\mathbf{P},\mathbf{Q}) = -\log \frac{\exp(\mathbf{w}^T\mathbf{h}_i^t)}{\sum_{j=1}^{4}\exp(\mathbf{w}^T\mathbf{h}_j^t)}, \quad (6)$$

where $\mathbf{w} \in \mathbb{R}^l$ is a parameter to learn.

3 Experiment

To evaluate the effectiveness of our hierarchical co-matching model, we use the RACE dataset (Lai et al., 2017), which consists of two subsets: RACE-M comes from middle school examinations while RACE-H comes from high school examinations. RACE is the combination of the two.

We compare our model with a number of baseline models. We also compare with two variants of our model for an ablation study.

Comparison with Baselines We compare our model with the following baselines:

• **Sliding Window** based method (Richardson et al., 2013) computes the matching score based on the sum of the tf-idf values of the matched words between the question-answer pair and each subpassage with a fixed a window size.

• **Stanford Attentive Reader (AR)** (Chen et al., 2016) first builds a question-related passage representation through attention mechanism and then compares it with each candidate answer representation to get the answer probabilities.

• **GA** (Dhingra et al., 2017) uses gated attention mechanism with multiple hops to extract the question-related information of the passage and compares it with candidate answers.

• **ElimiNet** (Soham et al., 2017) tries to first eliminate the most irrelevant choices and then select the best answer.

• **HAF** (Zhou et al., 2018) considers not only the matching between the three sequences, namely, passage, question and candidate answer, but also the matching between the candidate answers.

• **MUSIC** (Xu et al., 2017) integrates different sequence matching strategies into the model and also adds a unit of multi-step reasoning for selecting the answer.

Besides, we also report the following two results as reference points: **Turkers** is the performance of Amazon Turkers on a randomly sampled subset of the RACE test set. **Ceiling** is the percentage of the unambiguous questions with a correct answer in a subset of the test set.

The performance of our model together with the baselines are shown in Table 2. We can see that our proposed complete model, **Hier-Co-Matching**, achieved the best performance among all the public results. Still, there is a huge gap between the best machine reading performance and the human performance, showing the great potential for further research.

Ablation Study Moreover, we conduct an ablation study of our model architecture. In this study, we are mainly interested in the contribution of each component introduced in this work to our final results. We studied two key factors: (1) the co-matching module and (2) the hierarchical aggregation approach. We observed a 4 percentage performance decrease by replacing the co-matching module with a single matching state (*i.e.*, only $\mathbf{M}^a$ in Eqn (3)) by directly concatenating the question with each candidate answer (Yin et al., 2016). We also observe about 2 percentage decrease when we treat the passage as a plain sequence, and run a two-layer LSTM (to ensure the numbers of parameters are comparable) over the whole passage instead of the hierarchical LSTM.

Question Type Analysis We also conducted an analysis on what types of questions our model can handle better. We find that our model obtains similar performance on the "wh" questions such as "why," "what," "when" and "where" questions, on which the performance is usually around 50%. We also check statement-justification questions with the keyword "true" (*e.g.*, "Which of the following statements is true"), negation questions with the keyword "not" (*e.g.*, "which of the following is not true"), and summarization questions with the keyword "title" (*e.g.*, "what is the best title for the passage?"), and their performance is 51%, 52% and 48%, respectively. We can see that the performance of our model on different types of questions in the RACE dataset is quite similar. However, our model is only based on word-level matching and may not have the ability of reasoning. In order to answer questions that require summarization, inference or reasoning, we still need to further explore the dataset and improve the model. Finally, we further compared our model to the baseline, which concatenates the question with each candidate answer, and our model can achieve better performance on different types of questions. For example, on the subset of the questions with pronouns, our model can achieve better accuracy of 49.8% than 47.9%. Similarly, on statement-justification questions with the keyword "true", our model could achieve better accuracy of 51% than 47%.

4 Conclusions

In this paper, we proposed a co-matching model for multi-choice reading comprehension. The model consists of a co-matching component and a hierarchical aggregation component. We showed that our model could achieve state-of-the-art performance on the RACE dataset. In the future, we will adapt the idea of co-matching and hierarchical aggregation to the standard open-domain QA setting for answer candidate reranking (Wang et al., 2017). We will also further study how to explicitly model inference and reasoning on the RACE dataset.

5 Acknowledgement

This work was partially supported by DSO grant DSOCL15223.

References

Danqi Chen, Jason Bolton, and Christopher D. Manning. 2016. A thorough examination of the CNN/Daily Mail reading comprehension task. In *Proceedings of the Conference on Association for Computational Linguistics*.

Bhuwan Dhingra, Hanxiao Liu, Zhilin Yang, William W Cohen, and Ruslan Salakhutdinov. 2017. Gated-attention readers for text comprehension. In *Proceedings of the Conference on Association for Computational Linguistics*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of the Conference on Advances in Neural Information Processing Systems*.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015. The goldilocks principle: Reading children's books with explicit memory representations. *arXiv preprint arXiv:1511.02301* .

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation* 9(8):1735–1780.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale reading comprehension dataset from examinations. *Proceedings of the Conference on Empirical Methods in Natural Language Processing* .

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268* .

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Matthew Richardson, Christopher JC Burges, and Erin Renshaw. 2013. MCTest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Parikh Soham, Sai Ananya, Nema Preksha, and M Khapra Mitesh. 2017. Eliminet: A model for eliminating options for reading comprehension with multiple choice questions. *Openreview* .

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the Conference on Association for Computational Linguistics*.

Duyu Tang, Bing Qin, and Ting Liu. 2015. Document modeling with gated recurrent neural network for sentiment classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Adam Trischler, Zheng Ye, Xingdi Yuan, Jing He, Phillip Bachman, and Kaheer Suleman. 2016. A parallel-hierarchical model for machine comprehension on sparse data. In *Proceedings of the Conference on Association for Computational Linguistics*.

Shuohang Wang and Jing Jiang. 2016. Learning natural language inference with LSTM. In *Proceedings of the Conference on the North American Chapter of the Association for Computational Linguistics*.

Shuohang Wang and Jing Jiang. 2017. A compare-aggregate model for matching text sequences. In *Proceedings of the International Conference on Learning Representations*.

Shuohang Wang, Mo Yu, Jing Jiang, Wei Zhang, Xiaoxiao Guo, Shiyu Chang, Zhiguo Wang, Tim Klinger, Gerald Tesauro, and Murray Campbell. 2017. Evidence aggregation for answer re-ranking in open-domain question answering. *arXiv preprint arXiv:1711.05116* .

Yichong Xu, Jingjing Liu, Jianfeng Gao, Yelong Shen, and Xiaodong Liu. 2017. Towards human-level machine reading comprehension: Reasoning and inference with multiple strategies. *arXiv preprint arXiv:1711.04964* .

Wenpeng Yin, Sebastian Ebert, and Hinrich Schütze. 2016. Attention-based convolutional neural network for machine comprehension. *arXiv preprint arXiv:1602.04341* .

Haichao Zhou, Wei Furu, Qin Bing, and Liu Ting. 2018. Hierarchical attention flow for multiple-choice reading comprehension. In *Proceedings of AAAI Conference on Artificial Intelligence*.