

# Inferential Machine Comprehension: Answering Questions by Recursively Deducing the Evidence Chain from Text

Jianxing Yu<sup>1,2</sup>, Zheng-Jun Zha<sup>3</sup>, Jian Yin<sup>1,2</sup> \*

<sup>1</sup>School of Data and Computer Science, Sun Yat-sen University

<sup>2</sup>Guangdong Key Laboratory of Big Data Analysis and Processing, China

<sup>3</sup>School of Information Science and Technology, University of Science and Technology of China

{yujx26, issjyin}@mail.sysu.edu.cn

zhazj@ustc.edu.cn

## Abstract

This paper focuses on the topic of inferential machine comprehension, which aims to fully understand the meanings of given text to answer generic questions, especially the ones needed reasoning skills. In particular, we first encode the given document, question and options in a context aware way. We then propose a new network to solve the inference problem by decomposing it into a series of attention-based reasoning steps. The result of the previous step acts as the context of next step. To make each step can be directly inferred from the text, we design an operational cell with prior structure. By recursively linking the cells, the inferred results are synthesized together to form the evidence chain for reasoning, where the reasoning direction can be guided by imposing structural constraints to regulate interactions on the cells. Moreover, a termination mechanism is introduced to dynamically determine the uncertain reasoning depth, and the network is trained by reinforcement learning. Experimental results on 3 popular data sets, including MCTest, RACE and MultiRC, demonstrate the effectiveness of our approach.

## 1 Introduction

Machine comprehension is one of the hot research topics in natural language processing. It measures the machine's ability to understand the semantics of a given document via answering questions related to the document. Towards this task, many datasets and corresponding methods have been proposed. In most of these datasets, such as CNN/Daily Mail (Hermann et al., 2015), and SQuAD (Rajpurkar et al., 2016), the answer is often a single entity or a text span in the document. That leads to the fact that lots of questions can be solved trivially via word and context matching (Trischler et al., 2016a) instead of

\*Corresponding author.

**Passage:**  
(S<sub>1</sub>)It has not been much fun. (S<sub>2</sub>)Many papers stayed afloat by pushing journalists overboard. (S<sub>3</sub>)The American Society of News Editors reckons that 13,500 newsroom jobs have gone since 2007. (S<sub>4</sub>)Readers are paying more for slimmer products. (S<sub>5</sub>)Some papers even had the nerve to refuse delivery to distant suburbs. (S<sub>6</sub>)Yet these desperate measures have proved the right ones and, sadly for many journalists, they can be pushed further.

**Question:** Why some newspapers refused delivery to distant suburbs?  
**Options:**  
[A]. readers threatened to pay less  
[B\*]. newspapers wanted to reduce costs  
[C]. journalists reported little about these areas  
[D]. subscribers complained about slimmer products

Figure 1: Sample of question needed reasoning skill. Correct answer is marked with an asterisk

genuine comprehension on text. To alleviate this issue, some datasets are released, such as MCTest (Richardson et al., 2013), RACE (Lai et al., 2017) and MultiRC (Khashabi et al., 2018), where the answers are not restricted to be the text spans in the document; instead, they can be described in any words. Specially, a significant proportion of questions require reasoning which is a sophisticated comprehension ability to choose the right answers. As shown in Figure 1, the question asks the reason for the phenomenon on sentence S<sub>5</sub>. The answer has to be deduced over the logical relations among sentence S<sub>3</sub>, S<sub>4</sub> and S<sub>5</sub>, and then entailed from S<sub>3</sub> to the correct option B. Difficultly, such deduced chain is not explicitly given but expressed on text semantics. Existing methods primarily focus on document-question interaction to capture the context similarity for answer span matching (Wang et al., 2017). They have minimal capability to synthesize supported facts scattered across multiple sentences to form the evidence chain which is crucial for reasoning.

To support inference, mainstream methods can be summarized into three folds. One is converting the unstructured document to formal predicate expressions, on which to perform mathematical d-

education via Bayesian network or first-order logic. The conversion lacks of adequate robustness to be applicable. Another direction is to explicitly parse the document into a relation tree, on which to generate answers via hand-crafted rules (Sun et al., 2018b). However, the parser often has to cascade to the model, which is difficult to train globally and would suffer from the error propagation problem. The third method exploits memory network to imitate reasoning by multi-layer architecture and iterative attention mechanism (Weston et al., 2014). Nevertheless, the reasoning ability is insufficient due to the lack of prior structural knowledge to lead the inference direction.

We observe that when humans answer the inferential question, they often finely analyze the question details and comprehend contextual relations to derive an evidence chain step by step. Using the sample in Figure 1 for illustration, humans first investigate the question to find the useful details, such as the question type “why”, and the aspect asked, i.e. “some newspapers refused delivery to distant suburbs”. Such details often play a critical role for answering. For example, *why* question usually expects the *causal* relation that could indicate the reasoning direction. Based on question details, they then carefully read the document to identify the content on which the question aspect mentions, that is, the sentence S5. Based on the content, they would deduce new supported evidences step by step guided by question type and contextual relations, such as explainable relation between S5 and S4, and casual relation among S3 and S4. By considering the options, they would decide to stop when the observed information is adequate already to answer the question. For instance, by relevant paraphrase, S3 can entail option B that may be the answer. In this process, contextual relations and multi-step deduction are efficient mechanisms for deriving the evidence chain.

Based on above observations, we here propose an end-to-end approach to mimic human process for deducing the evidence chain. In particular, we first encode the given document, question and options by considering contextual information. We then tackle the inference problem by proposing a novel network that consists of a set of operational cells. Each cell is designed with structural prior to capture the inner working procedure of an elementary reasoning step, where the step can be directly inferred from the text without strong supervi-

sion. The cell includes the memory and three operating units that work in tandem. That is, master unit derives a series of attention-based operations based on the question; reader unit extracts relevant document content on the operation; and writer unit performs the operation to deduce a result and update the memory. The cells are recursively connected, where the result of the previous step acts as the context of next step. The interactions of cells are restricted by structural constraints, so as to regulate the reasoning direction. With such structural multi-step design, the network can integrate the supported facts by contextual relations to build the evidence chain in arbitrarily complex acyclic form. Since the reasoning depth is uncertain, a termination mechanism is exploited to adaptively determine the ending. Moreover, a reinforcement approach is employed for effective training. Experiments are conducted on 3 popular data sets that contain questions required reasoning skills, including MCTest, RACE and MultiRC. The results show the effectiveness of our approach.

The main contributions of this paper include,

- We design a new network that can answer inferential question by recursively deducing the evidence chain from the text.
- We propose an effective termination mechanism which can dynamically determine the uncertain reasoning depth.
- We employ a reinforcement training approach and conduct extensive experiments.

The rest of this paper is organized as follows. Section 2 elaborates our approach on the inferential framework. Section 3 presents the experimental results. Section 4 reviews related work and Section 5 concludes this paper with future works.

## 2 Approach

As shown in Figure 2, our approach consists of three components, including input representation, inferential network composed out of multiple cells, and output. Next, we define some notations, and then elaborate the details on each component.

### 2.1 Notations and Problem Formulation

Given a document  $D$  in unstructured text, the task of machine comprehension is to answer the questions according to the semantics of the document. In this paper, multi-choice questions are our major focus. Thus, a set of plausible answer option-

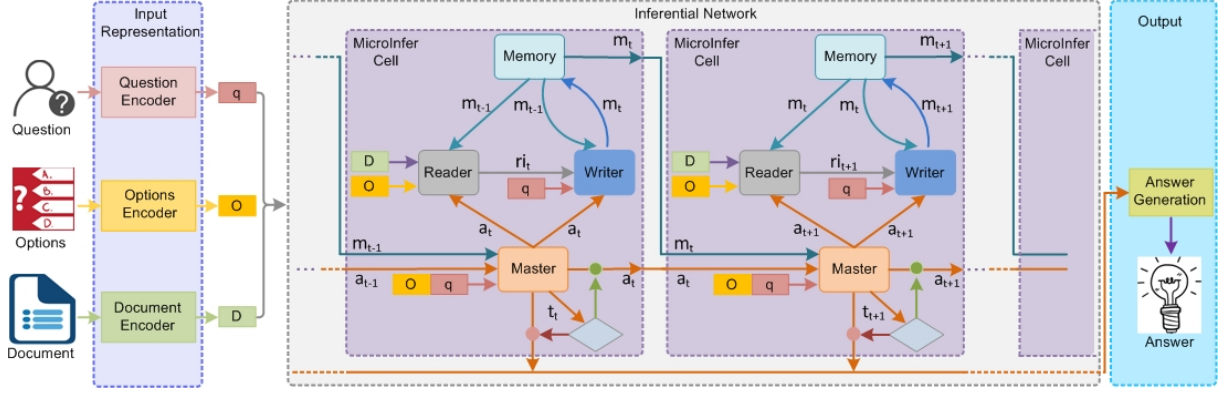


Figure 2: Overview of the our approach

s are assumed to be provided, and the task is reduced to select a correct option from the given set. Formally, let  $q$  represent the question, of length  $S$ , where  $\{w_1, \dots, w_S\}$  are the question words;  $O = \{o_1, \dots, o_L\}$  denotes an option set. For a given document and question  $x = (D, q)$ , a score  $h(x, y) \in \mathcal{R}$  is assigned for each candidate in the option set  $y = o \in O$ , so as to measure its probability of being the correct answer. The option with highest score is outputted as the answer  $\hat{y} = \text{argmax}_{y \in O} h(x, y)$ .

## 2.2 Input Representation

We first encode the input text into distributed vector representations by taking account the context.

**Question:** Two stages are conducted on the encoding. (1) We convert the question into a sequence of learned word embeddings by looking up the pre-trained vectors, such as GloVe (Pennington et al., 2014). By considering the question type would help inference, we customize an embedding to indicate such type via linguistic prior knowledge, such as the positions of interrogative words are often relatively fixed, and the corresponding parts of speech (POS) are mainly adverbs or conjunctions, etc. Practically, we utilize position and POS embedding (Li et al., 2018b) generated by word embedding tool. That is, the embedding layer is a  $W \in \mathbb{R}^{d \times v}$ , where  $d$  is the dimension and  $v$  denotes the number of instances. (2) We concatenate the embeddings of word, position and POS, and feed them into a bi-directional GRU (BiGRU) (Cho et al., 2014) to incorporate sequential context. Then we can yield two kinds of representations, including (a) *contextual words*: a series of output states  $cw_s|_{s=1}^S$  that represent each word in the context of the question, where  $cw_s = [\overleftarrow{h}_s, \overrightarrow{h}_s]$ ,

$\overleftarrow{h}_s$  and  $\overrightarrow{h}_s$  are the  $s^{\text{th}}$  hidden states in the backward and forward GRU passes respectively; and (b) *overall encoding*:  $q = [\overleftarrow{cw}_1, \overrightarrow{cw}_S]$ , the concatenation of the final hidden states.

**Options:** Each option word is embedded by pre-trained vectors and then option is contextually encoded by BiGRU to generate an overall vector.

**Document:** Three steps are performed on the encoding. (1) We encode each document sentence by considering context via BiGRU as aforementioned. The sentence is transformed into an  $n_i \times d$  matrix, where  $n_i$  is the size of words in the sentence  $i$ , and  $d$  is the dimension. (2) We conduct attention to compress the sentence encoding into a fixed size vector, and focus on the important components. Intuitively, long sentence may contain multiple significant parts, where each would help inference. For example, two clauses are linked by “or” with the *causal* relation in the sentence “The oil spill must be stopped or it will spread for miles.” The clauses and the contextual relation can assist answer the question “Why must the oil spill be stopped?” To model such situation, *structured self attention* technique proposed by Lin et al. (2017) is utilized. It can convert the sentence into a  $J \times d$  matrix, attending at  $J$  significant parts of the sentence in a context aware way. (3) All sentence matrices are fed into another BiGRU, so as to capture the context between the sentences. That is  $D^{H \times J \times d} = \{ds_{h,j}^d|_{h,j=1,1}^{H,J}\}$ , where  $H$  is the sentences size,  $ds$  is the sentence vector.

## 2.3 Micro-Infer Cell

Micro-infer is a recurrent cell designed to model the mechanism of an atomic reasoning step. The cell consists of one memory unit and three operational units, including the master unit, reader unit and writer unit. The memory independently

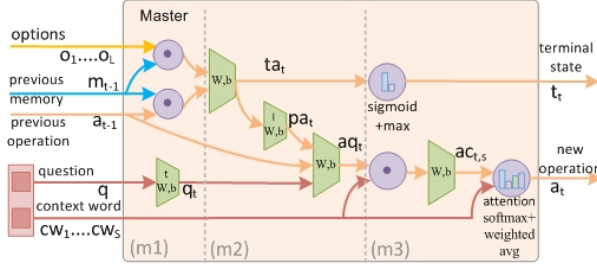


Figure 3: Flow chart of the master unit

stores the intermediate results obtained from the reasoning process up to the  $t^{th}$  step. Based on the memory state, three operational units work together in series to accomplish the reasoning process. In particular, master unit analyzes the question details to focus on certain aspect via self-attention; reader unit then extracts related content, guided by the question aspect and text context; and the writer unit iteratively integrates the content with preceding results from the memory to produce a new intermediate result. The interactions between the cell's units are regulated by structured constraints. Specially, the master outcome can only indirectly guide the integration of relevant content into the memory state by soft-attention maps and gating mechanisms. Moreover, a termination gate is introduced to adaptively determine ending of the inference. In the following, we detail the formal specifications of three operational units in the cell.

### 2.3.1 Master Unit

As presented in Figure 3, this unit consists of two components, involving the termination mechanism and question analysis.

#### Termination Mechanism

A maximum step is set to guarantee termination. Since the complexity of the questions is different, the reasoning depths are uncertain. To dynamically adjust to such depth, a terminated gate is designed by considering two conditions. That is, the correlation between the intermediate result  $m_{t-1}$  and the reasoning operation  $a_{t-1}$  in previous step, as well as  $m_{t-1}$  and candidate answer options  $o_l|_{l=1}^L$ . When both conditions are met, an acceptable answer is highly probable to obtain. Technically, the correlations are calculated by Eq.(1), i.e.  $m_{t-1} \odot a_{t-1}$ ,  $m_{t-1} \odot o_l$ , respectively. We then combine these two factors to get  $ta_{t,l}$ , and utilize a *sigmoid* layer to estimate the ending probability for a certain option. By maximizing over all the options, a termination function

$f_{ts}(m_{t-1}, a_{t-1}, o_l|_{l=1}^L; \theta_{ts})$  is generated, where  $\theta_{ts}$  is a parameter set, namely  $(W_{ta}^{d \times 2d}, b_{ta}^d)$ . Based on the function, a binary random variable  $t_t$  is probabilistically drawn as  $t_t \sim p(\cdot | f_{ts}(\cdot; \theta_{ts}))$ . If  $t_t$  is True, stop and execute the answer module accordingly; otherwise, continue the  $t^{th}$  reasoning step.

$$\begin{aligned} ta_{t,l} &= W_{ta}^{d \times 2d} [m_{t-1} \odot a_{t-1}, m_{t-1} \odot o_l] + b_{ta}^d \\ f_{ts}(\cdot; \theta_{ts}) &= \max\{sigmoid(ta_{t,l})|_{l=1}^L\} \end{aligned} \quad (1)$$

### Question Analysis

We design a soft-attention based mechanism to analyze the question and determine the basic operation performed at each step. Instead of grasping the complex meaning on the whole question at once, the model is encouraged to focus on certain question aspect at a time, making the reasoning operation can be directly inferred from the text. Three stages are performed as follows.

Firstly, we project the question  $q$  through a learned linear transformation to derive the aspect related to  $t^{th}$  reasoning step, as  $q_t = W_{qt}^{d \times d} q + b_{qt}^d$ .

Secondly, we use the previously performed operation  $a_{t-1}$  and memory result  $m_{t-1}$  as decision base to lead  $t^{th}$  reasoning operation. In details, we validate previous reasoning result by leveraging the terminated conditions in Eq.(1), that is,  $pa_t = W_{pa}^{d \times Ld} [ta_{t,1}, \dots, ta_{t,L}] + b_{pa}^d$ . We then integrate  $q_t$  with preceding operation  $a_{t-1}$  and validation  $pa_t$  through a linear transformation into  $aq_t$ , as  $W_{aq}^{d \times 3d} [q_t, a_{t-1}, pa_t] + b_{aq}^d$ .

Thirdly,  $aq_t$  is regulated by casting it back to original question words  $cw_s|_{s=1}^S$  based on attention in Eq.(2), so as to restrict the space of the valid reasoning operations and boost the convergence rate. In particular, we calculate the correlation  $ac_{t,s}$  and pass it through a *softmax* layer to yield a distribution  $av_{t,s}$  over the question words. By aggregation, a new reasoning operation  $a_t$  is generated, represented in terms of the question words.

$$\begin{aligned} ac_{t,s} &= W_{ac}^{1 \times d} [aq_t \odot cw_s] + b_{ac}^1 \\ av_{t,s} &= softmax(ac_{t,s}) \\ a_t &= \sum_{s=1}^S av_{t,s} \cdot cw_s; \end{aligned} \quad (2)$$

Briefly, the new reasoning operation  $a_t$  is modeled by a function  $f_{na}(q, a_{t-1}, cw_s; \theta_{na})$ , where  $\theta_{na}$  is a set of parameters, including  $(W_{qt}^{d \times d}, b_{qt}^d, W_{aq}^{d \times 3d}, b_{aq}^d, W_{pa}^{d \times Ld}, b_{pa}^d, W_{ac}^{1 \times d}, b_{ac}^1)$ .

### 2.3.2 Reader Unit

As shown in Figure 4, reader unit retrieves relevant document content that is required for per-



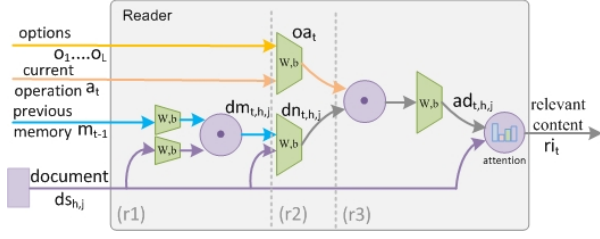


Figure 4: Flow chart of the reader unit

forming the  $t^{\text{th}}$  reasoning operation. The relevance is measured by the content context in a soft-attention manner, taking account of the current reasoning operation and prior memory. We do not rely on external tools to facilitate globally training.

To support transitive reasoning, we first extract the document content relevant to the preceding result  $m_{t-1}$ , resulting in  $dm_{t,h,j} = [W_m^{d \times d} m_{t-1} + b_m^d] \odot [W_{ds}^{d \times d} ds_{h,j} + b_{ds}^d]$ . The relevance often indicates a contextual relation in the distributed space. For instance, given a question aspect *why*, the contents with *causal* relation are highly expected and their relevant score is likely to be large.

Then,  $dm_{t,h,j}$  is independently incorporated with the document content  $ds_{h,j}$  to produce  $dn_{t,h,j}$ , i.e.  $W_{dn}^{d \times 2d} [dm_{t,h,j}, ds_{h,j}] + b_{dn}^d$ . This allows us to also consider new information which is not directly related to the prior intermediate result, so as to assist parallel and inductive reasoning.

Lastly, we use soft attention to select content that is relevant to the reasoning operation  $a_t$  and candidate options  $o_l |_{l=1}^L$ . Precisely, we unify the  $a_t$  and  $o_l |_{l=1}^L$  by a linear transformation to obtain  $oa_t$ , i.e.  $W_{oa}^{d \times Ld} [a_t, o_l |_{l=1}^L] + b_{oa}^d$ , where the options size  $L$  is fixed and predefined. We then measure the correlation between  $oa_t$  and the extracted content  $dn_{t,h,j}$ , passing the result through *softmax* layer to produce an attention distribution. By taking weighted average over the distribution, we can retrieve related content  $ri_t$  by Eq.(3).

$$\begin{aligned} ad_{t,h,j} &= W_{ad}^{d \times d} [oa_t \odot dn_{t,h,j}] + b_{ad}^d \\ rv_{t,h,j} &= \text{softmax}(ad_{t,h,j}) \\ ri_t &= \sum_{h=1; j=1}^{H; J} rv_{t,h,j} \cdot ds_{h,j}; \end{aligned} \quad (3)$$

In short, the retrieved content  $ri_t$  is formulated by a function  $f_{ri}(m_{t-1}, ds_{h,j}, a_t, o_l |_{l=1}^L; \theta_{ri})$ , where  $\theta_{ri}$  is a parameter set, involving  $(W_m^{d \times d}, b_m^d, W_{ds}^{d \times d}, b_{ds}^d, W_{dn}^{d \times 2d}, b_{dn}^d, W_{oa}^{d \times Ld}, b_{oa}^d, W_{ad}^{d \times d}, b_{ad}^d)$ .

### 2.3.3 Writer Unit

As illustrated in Figure 5, writer unit is responsible to compute the intermediate result on the  $t^{\text{th}}$

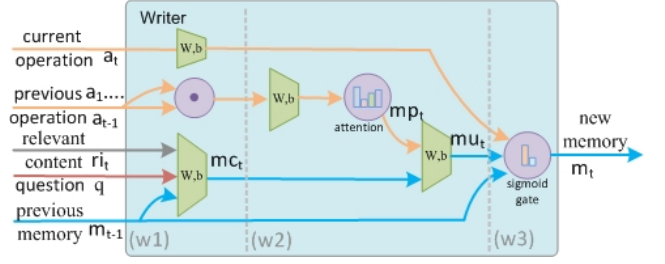


Figure 5: Flow chart of the writer unit

reasoning process and update the memory state. It integrates the retrieved content from the reader unit with the preceding intermediate result in the memory, guided by the  $t^{\text{th}}$  reasoning operation in the master unit. Details are presented as follows.

(1) Motivated by the work on relational reasoning (Santoro et al., 2017), we linearly incorporate the retrieved content  $ri_t$ , prior result  $m_{t-1}$ , and question  $q$  to get  $mc_t = W_{mc}^{d \times 3d} [ri_t, m_{t-1}, q] + b_{mc}^d$ , so as to measure their correlations.

(2) By considering non-sequential reasoning, such as tree or graph style, we refer to all previous memorized results instead of just the proceeding one  $m_{t-1}$ . Motivated by the work on scalable memory network (Miller et al., 2016), we compute the attention of the current operation  $a_t$  against all previous ones  $a_i |_{i=1}^{t-1}$ , yielding  $sa_{ti} = \text{softmax}(W_{sa}^{1 \times d} [a_t \odot a_i] + b_{sa}^1)$ . And then we average over the previous results  $m_i |_{i=1}^{t-1}$  to get preceding relevant support as  $mp_t$ , that is  $\sum_{i=1}^{t-1} sa_{ti} \cdot m_i$ . By combining  $mp_t$  with correlated result  $mc_t$  above, we can obtain a plausible result  $mu_t$ , namely  $W_{mp}^{d \times d} mp_t + W_{mc}^{d \times d} mc_t + b_{mu}^d$ .

(3) The operations on some question aspects such as *why* need multi-step reasoning and updating while others no need. In order to regulate the valid reasoning space, an update gate is introduced to determine whether to refresh the previous result  $m_{t-1}$  in the memory by the new plausible result  $mu_t$ . The gate  $\alpha_t$  is conditioned on the operation  $a_t$  by using a learned linear transformation and a *sigmoid* function. If the gate is open, the unit updates the new result to the memory, otherwise, it skips this operation and performs the next one.

$$\begin{aligned} \alpha_t &= \text{sigmoid}(W_a^{1 \times d} a_t + b_a^1) \\ m_t &= \alpha_t \cdot m_{t-1} + (1 - \alpha_t) \cdot mu_t; \end{aligned} \quad (4)$$

In brief, the new reasoning result  $m_t$  is modeled by a function  $f_{nm}(m_{t-1}, ri_t, q, a_t; \theta_{nm})$ , where  $\theta_{nm}$  is a parameter set, including  $(W_{mc}^{d \times 3d}, b_{mc}^d, W_{sa}^{1 \times d}, b_{sa}^1, W_{mp}^{d \times d}, W_{mc}^{d \times d}, b_{mu}^d, W_a^{1 \times d}, b_a^1)$ .

## 2.4 Output and Training

After the terminated condition is met, we can obtain the memory state  $m_{t-1}$ , which indicates the final intermediate result of the reasoning process. For the multi-choice questions focused in the paper, there is a fixed set of possible answers. We then leverage a classifier to predict an answer by referring to the question  $q$  and options  $o_l|_{l=1}^L$ . Precisely, we first measure the correlation of  $m_{t-1}$  against  $q$  and  $o_l|_{l=1}^L$ , to get  $m_{t-1} \odot q$ ,  $m_{t-1} \odot o_l$ . By concatenation, we pass the outcome through a 2-layer fully-connected *softmax* network to derive an answer option by Eq.(5), with *ReLU* activation function to alleviate over-fitting. In summary, the parameter set  $\theta_{ans}$  is  $(W_u^{d \times 2d}, b_u^d, W_{an}^{1 \times Ld}, b_{an}^1)$ .

$$\begin{aligned} u_l &= ReLU(W_u^{d \times 2d}[m_{t-1} \odot q, m_{t-1} \odot o_l] + b_u^d) \\ Ans_t &= softmax(W_{an}^{1 \times Ld}[u_1, \dots, u_L] + b_{an}^1) \end{aligned} \quad (5)$$

## Reinforcement Learning

Due to the discrete of the termination steps, the proposed network could not be directly optimized by back-propagation. To facilitate training, a reinforcement approach is used by viewing the inference operations as policies, including the reasoning operation flow  $G_{1:T}$ , termination decision flow  $t_{1:T}$  and answer prediction  $A_T$ , where  $T$  is the reasoning depth. Given  $i^{th}$  training instance  $\langle q_i; D_i; o_i \rangle$ , the expected reward  $r$  is defined to be 1 if the predicted answer is correct, otherwise 0. The rewards on intermediate steps are 0, i.e.  $\{r_t = 0\}_{t=1}^{T-1}$ . Each probable value pair of  $(G; t; A)$  corresponds to an episode, where all possible episodes denote as  $\mathbb{A}^\dagger$ . Let  $J(\theta) = \mathbb{E}_\pi \left[ \sum_{t=1}^T r_t \right]$  be the total expected reward, where  $\pi(G, t, A; \theta)$  is a policy parameterized by the network parameter  $\theta$ , involving the encoding matrices  $\theta_W$ , question network  $\theta_{na}$ , termination gate  $\theta_{ts}$ , reader network  $\theta_{ri}$ , writer network  $\theta_{nm}$ , and answer network  $\theta_{ans}$ . To maximize the reward  $J$ , we explore gradient descent optimization, with Monte-Carlo REINFORCE (Williams, 1992) estimation by Eq.(6).

$$\begin{aligned} \nabla_\theta J(\theta) &= \mathbb{E}_{\pi(G, t, A; \theta)} [\nabla_\theta \log \pi(G, t, A; \theta)(r - b)] \\ &= \sum_{(G, t, A) \in \mathbb{A}^\dagger} \pi(G, t, A; \theta) [\nabla_\theta \log \pi(G, t, A; \theta)(r - b)] \end{aligned} \quad (6)$$

where  $b$  is a critic value function. It is usually set as  $\sum_{(G, t, A)} \pi(G, t, A; \theta) r$  (Shen et al., 2016) and  $(r/b - 1)$  is often used instead of  $(r - b)$  to achieve stability and boost the convergence speed.

## 3 Evaluations

In this section, we extensively evaluate the effectiveness of our approach, including comparisons with state-of-the-arts, and components analysis.

### 3.1 Data and Experimental Setting

As shown in Table 1, experiments were conducted on 3 popular data sets in 9 domains, including MCTest, RACE and MultiRC. Different from data sets such as bAbI (Weston et al., 2015) that are synthetic, the questions in the evaluated data sets are high-quality to reflect real-world applications.

Data set	#doc	#q	#domain	ratio
MCTest	660	2,640	1	54.2%
MC160	160	640	1	53.3%
MC500	500	2,000	1	54.6%
RACE	27,933	97,687	1	25.8%
RACE-M	7,139	20,794	1	22.6%
RACE-H	28,293	69,394	1	26.9%
MultiRC	871	9,872	7	59.0%

Table 1: Statistics of the data sets. #doc, #q denote the size of the documents and questions accordingly; ratio means the proportion of the questions that require reasoning on multiple sentences; MC160 is a human double-check subset of MCTest, while MC500 is an unchecked one; RACE-M and RACE-H are the subsets of RACE on middle/ high school exams, respectively

Hyper-parameters were set as follows. For question encoding, the POS tags were obtained by using OpenNLP toolkit. Multiple cells were connected to form the network, where the cells were weight sharing. The maximum size of connected cells length was 16. The network was optimized via Adam (Kingma and Ba, 2014) with a learning rate of  $10^{-4}$  and a batch size of 64. We used gradient clipping with clipnorm of 8, and employed early stopping based on the validation accuracy. For word embedding, we leveraged 300-dimension pre-trained word vectors from GloVe, where the word embeddings were initialized randomly using a standard uniform distribution and not updated during training. The out-of-vocabulary words were initialized with zero vectors. The number of hidden units in GRU was set to 256, and the recurrent weights were initialized by random orthogonal matrices. The other weights in GRU were initialized from a uniform distribution between  $-0.01$  and  $0.01$ . We maintained the exponential moving averages on the model weights with a decay rate of 0.999, and used them at test time instead of the raw weights. Variational dropout of 0.15 was used across the network and

maximum reasoning step was set to 5. Training usually converged within 30 epochs.

### 3.2 Comparisons with the State-of-the-Arts

We compared our approach with all published baselines at the time of submission on the evaluated data sets. The baselines were summarized as follows. (1) On RACE data set, six baselines were employed, including three introduced in the release of the data set, that is Sliding Window (Richardson et al., 2013), Stanford AR (Chen et al., 2016), and GA (Dhingra et al., 2016); another three methods proposed recently, namely DFN (Xu et al., 2017), BiAttention 250d MRU (Tay et al., 2018), and OFT (Radford et al., 2018). (2) For MCTest data set, nine baselines were investigated, involving four on lexical matching, i.e. RTE, SWD, RTE+SWD Richardson et al. (2013), Linguistic (Smith et al., 2015); two methods used hidden alignment, that is Discourse (Narasimhan and Barzilay, 2015), Syntax (Wang et al., 2015); three approaches based on deep learning, i.e. EK (Wang et al., 2016), PH (Trischler et al., 2016b), and HV (Li et al., 2018a). (3) Regarding multi-choices questions in MultiRC data set, we replace *softmax* to *sigmoid* at the answer generation layer, so as to make prediction on each option. Accordingly, five baselines were exploited, including three used in the release of the data set, that is IR, SurfaceLR, and LR (Khashabi et al., 2018); two methods currently composed, namely OFT (Radford et al., 2018) and Strategies (Sun et al., 2018a).

As elaborated in Figure 6, our approach outperformed the individual baselines on all three data sets<sup>1</sup>. Specifically, for RACE data set, our approach achieved the best performance and outperformed the second one (i.e. *OFT*) in terms of average accuracy by over 4.12%, 5.00% on RACE-M and RACE-H, respectively. On MCTest data set, the outperformance was 5.55%, 7.14% over *PH* baseline which was the second best on MC160-*multi* and MC500-*multi*, respectively, where *multi* is a subset of the data set that is more difficult and needs understanding multiple sentences to answer. For MultiRC data set, our approach led to a performance boost against the second best one (i.e. *Strategies*) in terms of macro-average F1 by over 4.06%, while in terms of micro-average

<sup>1</sup>The leaderboard rankings were quickly refreshed, but our performance is still competitive at the camera-ready time.

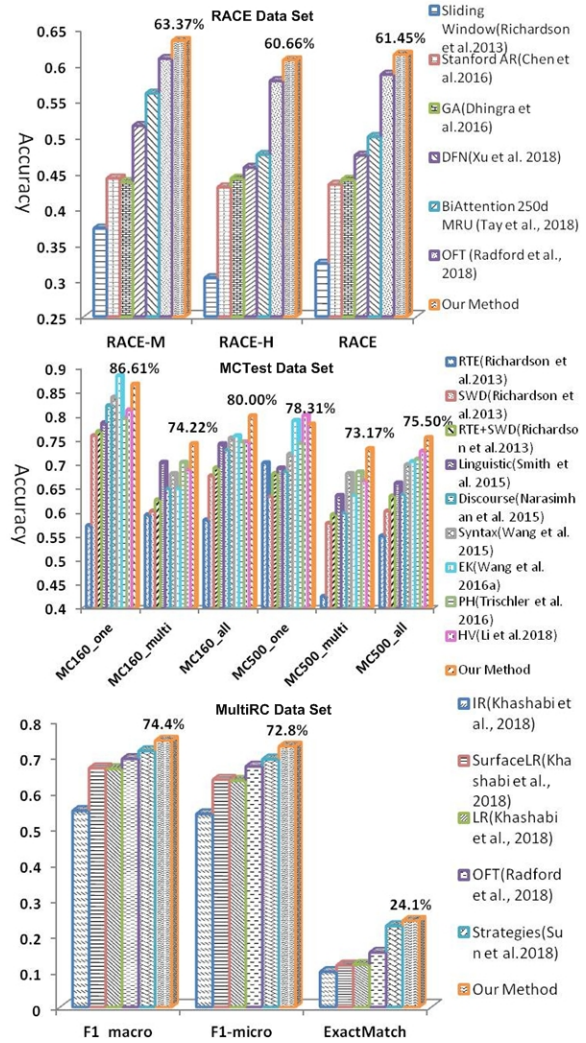


Figure 6: Comparisons of our approach against state-of-the-arts on the RACE, MCTest, and MultiRC data sets respectively. Statistical significant with p-values < 0.01 using two-tailed paired test

F1 and exact match accuracy by over 5.20% and 6.64%, respectively. Such results showed that our approach with structural multi-step design and context aware inference can correctly answer the questions, especially the non-trivial ones required reasoning, thus boost the overall performance.

### 3.3 Ablations Studies

To gain better insight into the relative contributions of various components in our approach, empirical ablation studies were performed on seven aspects, including (1) position and POS aware embedding on the question; (2) structural self-attention in document encoding; (3) two in the master unit, that is, guiding the reasoning operation by previous memory result, and casting back to original question words; (4) extract-



ing relevant content based on preceding memory result in reader unit; (5) two in the writer unit, namely, non-sequential reasoning and updating gate mechanisms. They were denoted as `pos_aware`, `doc_self_att`, `rsn_prior_mem`, `que_w_reg`, `prior_mem_res`, `non_seq_rsn`, and `udt_gate`, respectively.

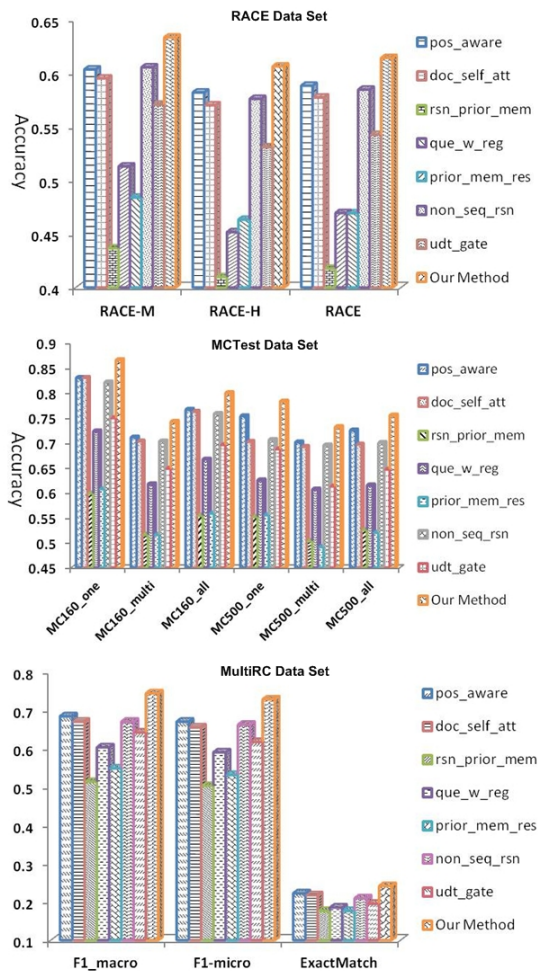


Figure 7: Ablation studies on various components of our approach for affecting the performance

As displayed in Figure 7, the ablation on all evaluated components in our approach led to the performance drop. The drop was more than 10% on four components, including (1) `rsn_prior_mem`; Lack of the memory guidance, the inferred result from previous step could not be served as context for the next. Losing such valuable context may lead to the misalignment of the reasoning chain. (2) `prior_mem_res`; Discard of the preceding memory result, the relevant content with contextual relations would not be identified. Such relations are the key for transitive reasoning. (3) `que_w_reg`; Without casting back to original question words,

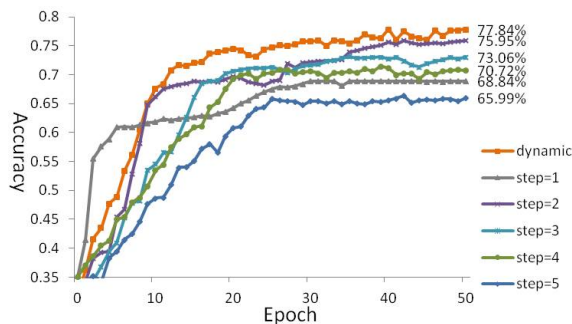


Figure 8: Evaluation on the termination mechanism

it is equivalent to processing the complex question at one step without identifying the details. Such coarse-grained processing fails to effectively regulate the space of the valid reasoning operations, and may confuse the reasoning direction. (4) `udt_gate`; The gate could help balance the complex and simple questions, and reduce long-range dependencies in the reasoning process by skipping, which would improve performance. These results further convinced us on the significant value of imposing strong structural priors to help the network derive the evidence chain from text.

Furthermore, we evaluated the efficiency of the termination mechanism by replacing it with fixed steps from 1 up to 5. The results on the RACE and MCTest data sets showed the replacement would lead to drop on average accuracy and slowdown on the convergence rate. As demonstrated in Figure 8, for fixed size reasoning, more steps performed well at first, but deteriorated soon, while dynamic strategy can adaptively determine the optimal termination, that may help boost the accuracy.

### 3.4 Case Study

Due to the use of soft attention, the proposed network offers a traceable reasoning path which can interpret the generation of the answer based on the attended words. To better understand the reasoning behavior, we plotted the attention map over the document, question and options in Figure 9 with respect to the sample on Figure 1. From the sequence of the maps, we observed that the network adaptively decided which part of an input question should be analyzed at each hop. For example, it first focused on the question aspect “some newspapers refused delivery to distant suburbs.” Then it generated evidence attended at S5 regarding to the focused aspect by similarity. Subsequently, the aspect “why” was focused and evidence attended at S4 was identified. We may infer that since



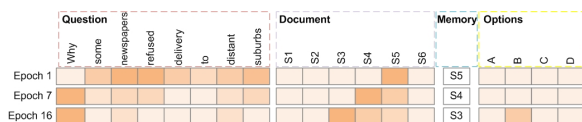


Figure 9: Visualized attention map on figure 1 sample

S4 and previous intermediate result S5 contain the explainable relation, they would most likely be correlated in the distributed space with sentence-level context aware encoding. Later, “why” was re-focused, the evidence attended at S3 was derived. Finally, option B was attended and the process ended due to termination unit may be triggered to work. Such results showed the network can derive the answer by capturing underlying semantics of the question and sequentially traversing the relations on document based on the context.

#### 4 Related Work

Earlier studies on machine comprehension mainly focused on the text span selection question. It is often transformed into a similarity matching problem and solved by feature engineering-based methods (Smith et al., 2015) or deep neural networks. The classical features include lexical features (e.g. overlapping of words, N-gram, POS tagging) (Richardson et al., 2013), syntactic features (Wang et al., 2015), discourse features (Narasimhan and Barzilay, 2015), etc. Besides, the typical networks involve Stanford AR (Chen et al., 2016), AS Reader (Kadlec et al., 2016), BiDAF (Seo et al., 2016), Match-LSTM (Wang and Jiang, 2017), etc, which used distributed vectors rather than discrete features to better compute the contextual similarity.

To support inference, existing models can be classified into three categories, including predicate based methods (Richardson and Domingos, 2006), rule-based methods relied on external parser (Sun et al., 2018b) or pre-built tree (Yu et al., 2012), and multi-layer memory networks (Hill et al., 2015), such as gated attended net (Dhingra et al., 2016), double-sided attended net (Cui et al., 2016), etc. These models either lack end-to-end design for global training, or no prior structure to subtly guide the reasoning direction. On the topic of multi-hop reasoning, current models often have to rely on the predefined graph constructed by external tools, such as interpretable network (Zhou et al., 2018) on knowledge graph. The graph plainly links the facts, from which the intermediate re-

sult in the next hop can be directly derived. However, in this paper, the evidence graph is not explicitly given by embodied in the text semantics.

Another related works are on Visual QA, aiming to answer the compositional questions with regards to a given image, such as “What color is the matte thing to the right of the sphere in front of the tiny blue block?” In particular, Santoro et al. (2017) proposed a relation net, yet the net was restricted to relational question, such as comparison. Later, Hudson and Manning (2018) introduced an iterative network. The network separated memory and control to improve interpretability. Our work leverages such separated design. Different from previous researches, we dedicate to inferential machine comprehension, where the question may not be compositional, such as *why* question, but requires reasoning on an unknown evidence chain with uncertain depth. The chain has to be inferred from the text semantics. To the best of our knowledge, no previous studies have investigated an end-to-end approach to address this problem.

#### 5 Conclusions and Future Works

We have proposed a network to answer generic questions, especially the ones needed reasoning. We decomposed the inference problem into a series of atomic steps, where each was executed by the operation cell designed with prior structure. Multiple cells were recursively linked to produce an evidence chain in a multi-hop manner. Besides, a terminated gate was presented to dynamically determine the uncertain reasoning depth and a reinforcement method was used to train the network. Experiments on 3 popular data sets demonstrated the efficiency of the approach. Such approach is mainly applied to multiple-choice questions now. In the future, we will expand it to support the questions on text span selection by using the relation type rather than the option as the terminated condition. For example, given the *why* question, reasoning process should be stopped when unrelated relation is met, such as transitional relation.

#### Acknowledgments

This work is supported by the National Key R&D Program of China (2018YFB1004404), Key R&D Program of Guangdong Province (2018B010107005, 2019B010120001), National Natural Science Foundation of China (U1711262, U1401256, U1501252, U1611264, U1711261).

## References

- D. Chen, J. Bolton, and C.D. Manning. 2016. A thorough examination of the cnn/daily mail reading comprehension task. In *Proceedings of the 54th ACL*, volume abs/1606.02858.
- K. Cho, B.V. Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Proceedings of EMNLP*, volume abs/1704.04683.
- Y. Cui, Z. Chen, S. Wei, S. Wang, T. Liu, and G. Hu. 2016. Attention-over-attention neural networks for reading comprehension. In *Proceedings of the 55th ACL*, volume abs/1607.04423.
- B. Dhingra, H. Liu, W. W.Cohen, and R. Salakhutdinov. 2016. Gated-attention readers for text comprehension. In *Proceedings of the 55th ACL*, volume abs/1606.01549.
- K.M. Hermann, T. Kociský, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of NIPS*, volume abs/1506.03340.
- F. Hill, A. Bordes, S. Chopra, and J. Weston. 2015. The goldilocks principle: Reading children’s books with explicit memory representations. In *Journal of Computer Science*, abs/1511.02301.
- D.A. Hudson and C.D. Manning. 2018. Compositional attention networks for machine reasoning. In *Proceedings of ICLR*, volume abs/1803.03067.
- R. Kadlec, M. Schmid, O. Bajgar, and J. Kleindienst. 2016. Text understanding with the attention sum reader network. In *Proceedings of the 54th ACL*, volume abs/1603.01547.
- D. Khashabi, S. Chaturvedi, M. Roth, S. Upadhyay, and D. Roth. 2018. Looking beyond the surface: A challenge set for reading comprehension over multiple sentences. In *Proceedings of NAACL-HLT*, pages 252–262.
- D.P. Kingma and J. Ba. 2014. Adam: A method for stochastic optimization. In *Proceedings of ICLR*, volume abs/1412.6980.
- G. Lai, Q. Xie, H. Liu, Y. Yang, and E.H. Hovy. 2017. RACE: large-scale reading comprehension dataset from examinations. In *Proceedings of EMNLP*, volume abs/1704.04683.
- C. Li, Y. Wu, and M. Lan. 2018a. Inference on syntactic and semantic structures for machine comprehension. In *Proceedings of AACL*, pages 5844–5851.
- L. Li, Y. Liu, and A. Zhou. 2018b. Hierarchical attention based position-aware network for aspect-level sentiment analysis. In *Proceedings of CoNLL*, volume abs/1704.04683, page 181189.
- Z. Lin, M. Feng, C.N. Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio. 2017. A structured self-attentive sentence embedding. In *Proceedings of ICLR*, volume abs/1703.03130.
- A.H. Miller, A. Fisch, J. Dodge, A. Karimi, A. Bordes, and J. Weston. 2016. Key-value memory networks for directly reading documents. In *Proceedings of the 54th ACL*, volume abs/1606.03126.
- K. Narasimhan and R. Barzilay. 2015. Machine comprehension with discourse relations. In *Proceedings of the 53rd ACL*, pages 1253–1262.
- J. Pennington, R. Socher, and C.D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of EMNLP*, pages 1532–1543.
- A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. 2018. Improving language understanding by generative pre-training. In *eprint*.
- P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. volume abs/1606.05250.
- M. Richardson, C. Burges, and E. Renshaw. 2013. M-CTest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of EMNLP*, pages 193–203.
- M. Richardson and P. Domingos. 2006. Markov logic networks. In *Journal of Machine Learning*, 62(1-2):107–136.
- A. Santoro, D. Raposo, D.G.T. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T.P. Lillicrap. 2017. A simple neural network module for relational reasoning. In *eprint arXiv:1706.01427*, abs/1706.01427.
- M.J. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. In *Proceedings of ICLR*, volume abs/1611.01603.
- Y. Shen, P. Huang, J. Gao, and W. Chen. 2016. Reasonet: Learning to stop reading in machine comprehension. In *Proceedings of the 23rd ACM SIGKDD*, volume abs/1609.05284.
- E. Smith, N. Greco, M. Bosnjak, and A. Vlachos. 2015. A strong lexical matching method for the machine comprehension test. In *Proceedings of EMNLP*, pages 1693–1698.
- K. Sun, D. Yu, D. Yu, and C. Cardie. 2018a. Improving machine reading comprehension with general reading strategies. In *eprint arXiv:1810-13441*, abs/1810.13441.
- Y. Sun, G. Cheng, and Y. Qu. 2018b. Reading comprehension with graph-based temporal-casual reasoning. In *Proceedings of COLING*, pages 806–817.
- Y. Tay, L.A. Tuan, and S.C. Hui. 2018. Multi-range reasoning for machine comprehension. In *eprint arXiv:1803.09074*, abs/1803.09074.

- A. Trischler, Z. Ye, X. Yuan, and K. Suleman. 2016a. [Natural language comprehension with the epireader](#). In *Proceedings of EMNLP*, volume abs/1606.02270.
- A. Trischler, Y. Zheng, X. Yuan, H. Jing, and P. Bachman. 2016b. [A parallel-hierarchical model for machine comprehension on sparse data](#). In *Proceedings of the 54th ACL*, pages 432–441.
- B. Wang, S. Guo, L.L. Kang, S. He, and J. Zhao. 2016. [Employing external rich knowledge for machine comprehension](#). In *Proceedings of IJCAI*.
- H. Wang, M. Bansal, K. Gimpel, and D. McAllester. 2015. [Machine comprehension with syntax, frames, and semantics](#). In *Proceedings of the 53rd ACL*, pages 700–706.
- S. Wang and J. Jiang. 2017. [Machine comprehension using match-1stm and answer pointer](#). In *Proceedings ICLR*, volume abs/1608.07905.
- W. Wang, N. Yang, F. Wei, B. Chang, and M. Zhou. 2017. [Gated self-matching networks for reading comprehension and question answering](#). In *Proceedings of the 55th ACL*, pages 189–198.
- J. Weston, A. Bordes, S. Chopra, and T. Mikolov. 2015. [Towards ai-complete question answering: A set of prerequisite toy tasks](#). In *Proceedings ICLR*, volume abs/1502.05698.
- J. Weston, S. Chopra, and A. Bordes. 2014. [Memory networks](#). In *eprint arXiv:1503.08895*, abs/1410.3916.
- R.J. Williams. 1992. [Simple statistical gradient-following algorithms for connectionist reinforcement learning](#). In *Journal Machine Learning*, 8(3-4):229–256.
- Y. Xu, J. Liu, J. Gao, Y. Shen, and X. Liu. 2017. [Dynamic fusion networks for machine reading comprehension](#). In *eprint arXiv:1711.04964*, abs/1711.04964.
- J. Yu, Z.J. Zha, and T.S. Chua. 2012. [Answering opinion questions on products by exploiting hierarchical organization of consumer reviews](#). In *Proceedings of EMNLP*, volume abs/1704.04683.
- M. Zhou, M. Huang, and X. Zhu. 2018. [An interpretable reasoning network for multi-relation question answering](#). In *Proceedings COLING*, volume abs/1801.04726.