

# Role of RNNs for Non-sequential Tasks in the Question Answering Context

Yuanyuan Shen

Department of Information  
Technology and Software  
Engineering  
Auckland University of  
Technology  
Auckland, New Zealand  
yuanyuan.shen@aut.ac.nz

Edmund M-K Lai

Department of Information  
Technology and Software  
Engineering  
Auckland University of  
Technology  
Auckland, New Zealand  
edmund.lai@aut.ac.nz

Mahsa Mohaghegh

Department of Information  
Technology and Software  
Engineering  
Auckland University of  
Technology  
Auckland, New Zealand  
mahsa.mohaghegh@aut.ac.nz

## ABSTRACT

Current state-of-the-art neural network-based Question Answering (QA) systems consist of both Recurrent Neural Networks (RNNs) and Feedforward Neural Networks (FFNNs). They generally performed well on 19 of the 20 tasks in the benchmark bAbI dataset. The only task that they failed badly is a task involving inductive reasoning where the order of the facts is not important in producing the correct answer. In this paper, we removed the RNNs from DMN+ QA system to form the ff-DMN system. The results demonstrate that ff-DMN improves the accuracy of the induction task significantly. Further experiments reveal that using RNNs is important if intra-sentence reasoning is required while it may adversely affect the performance if inter-sentence reasoning is involved. Finally, by incorporating ff-DMN and DMN+ our ensemble model outperforms the other QA systems on all the 20 tasks.

## CCS CONCEPTS

•Computing methodologies → Machine learning → Machine learning approaches → Neural networks •Computing methodologies → Artificial intelligence → Natural language processing → Information extraction

## KEYWORDS

Question Answering, Neural Networks, Recurrent Neural Networks, Inductive Reasoning

## ACM Reference format:

Yuanyuan Shen, Edmund M-K Lai and Mahsa Mohaghegh. 2020. Role of RNNs for Non-Sequential Tasks in the Question Answering Context. In *Proceedings of 2020 International Conference on Computational Linguistics and Natural Language Processing (CLNLP'20)*. Seoul, South Korea, 6 pages. <https://doi.org/10.1145/1234567890>

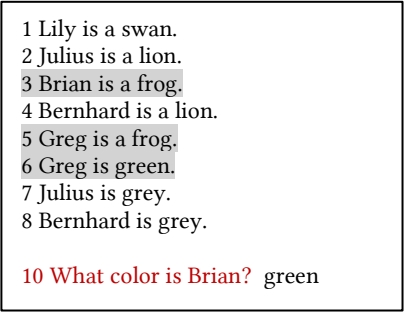
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

CLNLP'20, July, 2020, Seoul, South Korea

© 2020 Copyright held by the owner/author(s). 978-1-4503-0000-0/18/06...\$15.00  
<https://doi.org/10.1145/1234567890>

## 1 Introduction

In recent years question answering (QA), one of the most challenging areas in natural language processing (NLP), has experienced a significant growth due to the advances of Deep Neural Networks (DNNs) [1-3]. End-to-end neural network-based QA systems are trained to predict the answer to a question based on a set of statements of facts (the story). They can be incorporated into conversational systems and chatbots to provide replies to users when questions are raised.



1 Lily is a swan.  
2 Julius is a lion.  
3 Brian is a frog.  
4 Bernhard is a lion.  
5 Greg is a frog.  
6 Greg is green.  
7 Julius is grey.  
8 Bernhard is grey.  
  
10 What color is Brian? green

Figure 1: An example of the induction task

Most DNNs are either feedforward neural networks (FFNN) or recurrent neural networks (RNN). However, the most successful neural network-based QA systems typically consist of both FFNNs and RNNs. They include MemN2N [4], DMN+ [5], *t*-MEM-NN [6], EnDMN [7] and CAN [8]. They are able to achieve very good accuracies on the tasks in the benchmark bAbI dataset [9] that requires sequential and deductive reasoning skills. The RNNs in these systems play an important role in learning sequential and contextual information that are required for these tasks.

The only task that these systems performs poorly is task 16 of the bAbI dataset which involves inductive reasoning. The prediction accuracy is typically around 50% only. An example of the induction task is shown in Figure 1, with the three supporting facts highlighted. It is interesting to note that for this task, the order by which the relevant facts are presented is not

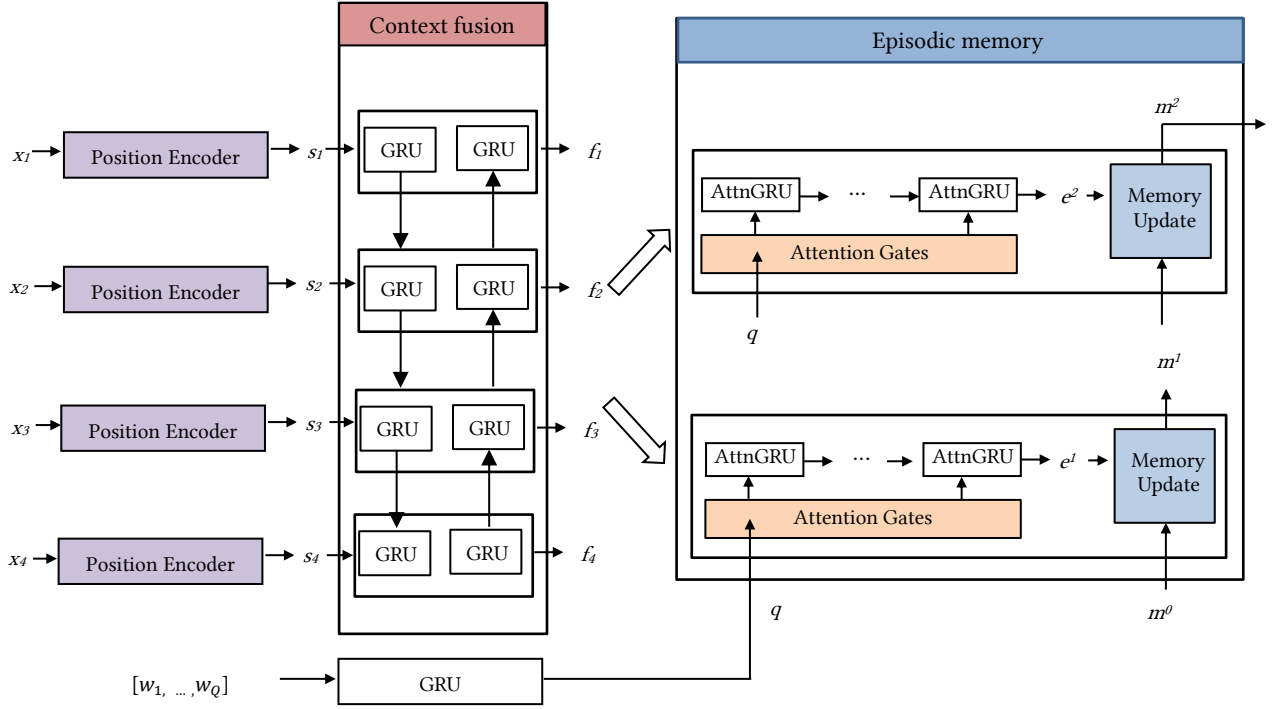


Figure 2: Block diagram of DMN+ QA System.

important. This raises the question whether the presence of RNNs in the system may actually adversely affect the performance of the system for such tasks.

In this paper, we investigate the performance of using a feedforward version of DMN+ for this particular task. This modified system will be referred to as ff-DMN and is obtained by removing all the RNNs from DMN+. Experimental results demonstrate that the accuracy for the induction task using ff-DMN improves significantly compared with using DMN+. For other tasks where that requires more inter-sentence reasoning, results indicate that the performance of ff-DMN is very competitive compared with that of DMN+. However, for the tasks that requires strongly intra-sentence reasoning, using RNNs is essential.

The rest of this paper is organized as follows. A brief review of the DMN+ QA system is presented in Section 2. This is followed in Section 3 by our modified ff-DMN architecture. Experimental results using the bAbI dataset are presented in Section 4 with discussions on their significance. Finally, the conclusions are drawn in Section 5.

## 2 The DMN+ QA System

The architecture of DMN+ is illustrated in Figure 2. The vector  $x_i$  represents a concatenation of word embeddings  $[w_1^{(i)}, \dots, w_{L_i}^{(i)}] \in \mathbb{R}^V$  where  $L_i$  is the total number of words in the  $i^{th}$  sentence and  $V$  is the dimension of the embedding vector.

$x_i$  is encoded into an initial sentence vector  $s_i$  by a position encoder (PE):

$$s_i = PE(w_1^{(i)}, \dots, w_{L_i}^{(i)}) \quad (1)$$

A sequence of  $N$  sentences  $[s_1, \dots, s_N]$ , is fed into an RNN made up of bi-directional GRU (BiGRU) [10]. It forms the context fusion layer which enables the order of the sentences to be learnt. The outputs of this layer  $F = [f_1, f_2, \dots, f_N] \in \mathbb{R}^V$  forms the input to the episodic memory. Another input to the episodic memory is the last state of a forward GRU that encodes the vector of word embeddings  $[w_1, \dots, w_M] \in \mathbb{R}^V$  for the question, where  $M$  is the number of words in the question.

The episodic memory module consists of several hops to retrieve information from sentence representations by paying attention to sentences in the story. Each hop contains the following components: attention gates  $\{g_i\}_{i=1}^N$  calculation, attention based GRU to generate the context vector and update of the episodic memory vector for output to the next hop. The attention gate for  $f_i$  at hop  $t$  is given by

$$z_i^{(t)} = W_1(\varphi([q, m^t, s_i])) + b_1 \quad (2)$$

$$\tau_i^{(t)} = W_2 \tanh(z_i^{(t)}) + b_2 \quad (3)$$

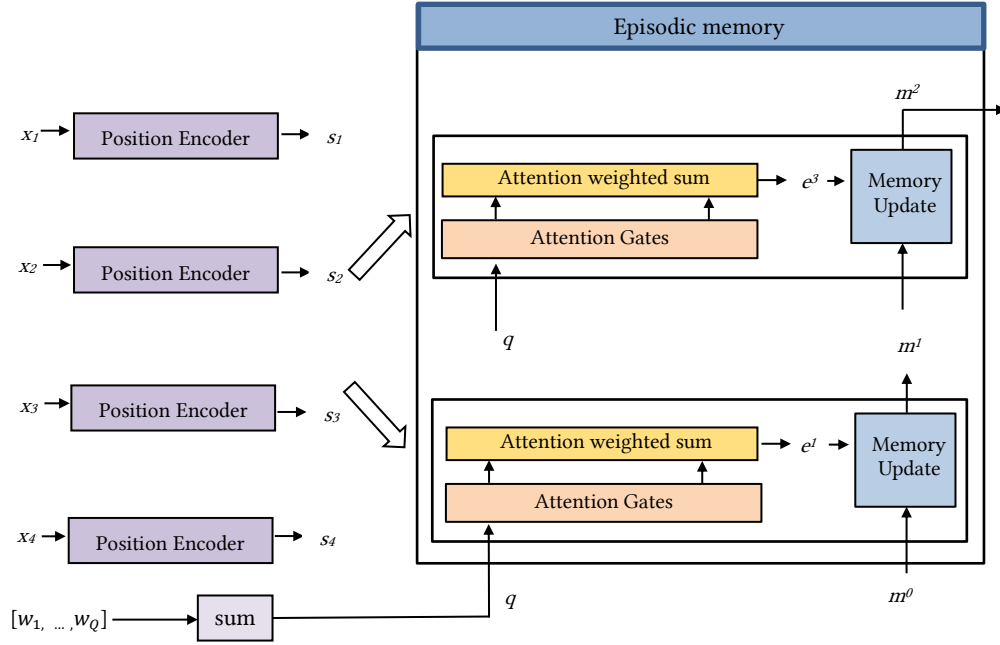


Figure 3: Block diagram of the ff-DMN QA System.

$$g_i^{(t)} = \text{softmax}(\tau_i^{(t)}) \quad (4)$$

where  $m^t$  is the memory vector at hop  $t$ , and  $\varphi([q, m^t, s_i]) = [s_i \cdot q; s_i \cdot m^t; |s_i - q|; |s_i - m^t|] \in R^{4V}$ .

The attention based GRUs are standard GRUs with the update gate replaced by attention gates. In this way, the positional and ordering information of the sentences is recorded, and the sentences with relatively higher attention gates are preserved in a context vector  $c^{(t)}$ . This latent context representation is then passed to the memory update component which is a feedforward neural network with the ReLU activation. The episodic memory vector of the last hop is used to predict the answer.

### 3 The Proposed ff-DMN model

The work in [11] shows that removing the RNN in the episodic memory module helps to improve the generalization capability on the induction task since in this task the order of the facts is not important. We conjecture that the RNNs used in the other places may also have a negative influence on the learning of this task. Hence, we propose to discard all the RNNs, in the form of BiGRU, from the architecture. More specifically, remove the GRUs from the context fusion module, the episodic memory module and question encoding module. Experiments confirms our speculation.

Figure 3 shows the architecture of the feedforward version of DMN+ which will be called ff-DMN. All RNNs in the DMN+ architecture shown in Figure 2 have been removed. The question

vector  $q$  is obtained by element-wise summation of the word embeddings in the question:

$$q = \sum_{i=1}^M w_{qi} \quad (5)$$

With the BiGRU in the context fusion module discarded, as a result, the sentence vectors  $[s_1, \dots, s_N]$  from the position encoders is directly passed to the episodic memory module. Instead of using an attention GRU to generate the context vector in each hop, a summation of the sentence vectors weighted by their corresponding attention weights is calculated as the episodic memory vector:

$$e^t = \sum_{i=1}^N g_i s_i \quad (6)$$

The memory update component is replaced by an identity function. The episodic memory vector and question vector are then passed into the linear feedforward answer module followed by the softmax operation to produce the predicted answer:

$$\hat{y} = \text{softmax}(W([q, m^T])) \quad (7)$$

## 4 Experiments

### 4.1 Dataset

The dataset used in our experiments is the bAbI 10k dataset. This dataset has 10k training samples and 1k testing samples. The 10k

<p>1 Mice are afraid of wolves.                  2 Gertrude is a mouse.                  3 Cats are afraid of sheep.                  4 Winona is a mouse.                  5 Sheep are afraid of wolves.                  6 Wolves are afraid of cats.                  7 Emily is a mouse.                  8 Jessica is a wolf.                  9 What is Gertrude afraid of?      wolf 2 1</p> <p style="text-align: center;">(a)</p>	<p>1 The suitcase fits inside the box.                  2 The chocolate fits inside the box.                  3 The container is bigger than the box of chocolates.                  4 The container is bigger than the suitcase.                  5 The box is bigger than the box of chocolates.                  6 The container is bigger than the chocolate.                  7 The chocolate fits inside the container.                  8 The chocolate fits inside the suitcase.                  9 The chocolate fits inside the chest.                  10 The suitcase fits inside the container.                  11 Does the box fit in the chocolate?    no 8 1</p> <p style="text-align: center;">(b)</p>
--	--

Figure 4: (a) An example of task 15; (b) an example of task 18

training samples are split into an 8:2 ratio for training and validation purposes respectively.

Out of the 20 tasks in the dataset, there are eight tasks, listed in Table 1, in which the order of the relevant facts is unimportant. These 8 tasks can be categorized into two groups. The first group consists of four tasks requires mainly the inter-sentence reasoning. An example is task 15 and an instance of that is shown in Figure 4 (a). The second group consists of the other four tasks where intra-sentence reasoning is crucial. An example is task 18 with an instance shown in Figure 4 (b).

Table 1: The 8 tasks for which the order of facts is unimportant

Reasoning types	tasks
inter-sentence	5: Three Argument Relations
	15: Basic Deduction
	16: Basic Induction
	20: Agent’s Motivations
intra-sentence	4: Two Argument Relations
	17: Positional Reasoning
	18: Size Reasoning
	19: Path Finding

### 4.2 Training Setup

We use the Adam optimizer [12] for training with a learning rate of 0.001. The Glorot uniform initialization [13] procedure is used for all the trainable weights. The word embeddings are randomly initialized with a uniform distribution in the range  $[-\sqrt{3}, \sqrt{3}]$  and trained along with the other weights. Three hops are used in the episodic memory to balance the predictive accuracy and the computational efficiency. The training objective is to minimize the cross-entropy loss  $\mathcal{L}(\hat{y}, y; \omega) = -\sum_i^K y_i \log(\hat{y}_i)$ , where  $K$  is the number of categories.

In order to compare the performance of the systems for individual tasks, they are trained independently with the data for each task.

### 4.3 Results

The prediction accuracies of the trained ff-DMN and the other QA systems benchmarked on the induction task in bAbI dataset are shown in Figure 5, which clearly depicts that our proposed ff-DMN outperforms all the other systems significantly. It improves upon the basic LSTM by 74.2% and upon DMN+ by 42.5%. It suggests that incorporating RNNs may hinder the system to learn to reason inductively.

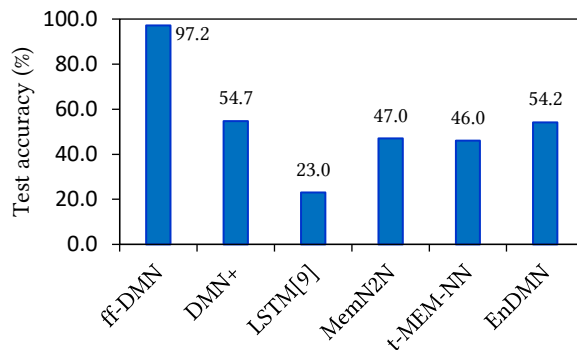


Figure 5: Test accuracies of different QA systems on task 16 induction

Figure 6 shows the validation losses of the two models on task 16 during training. As training progresses, the loss for ff-DMN continues to trend downwards. However, for DMN+, after the first few epochs the validation loss starts to increase. This suggests that DMN+ has a serious overfitting problem on this task.

Table 3: Prediction accuracies of ff-DMN and DMN+ on non-sequential tasks

Reasoning types	tasks	ff-DMN	DMN+	LSTM	MemN2N	t-MEM-NN	EnDMN
Inter-sentence	5: Three Argument Relations	91.4	99.5	70.0	85.9	88.0	<b>99.4</b>
	15: Basic Deduction	<b>100.0</b>	<b>100.0</b>	21.0	100.0	<b>100.0</b>	<b>100.0</b>
	16: Basic Induction	<b>97.2</b>	54.7	23.0	47.9	46.0	54.2
	20: Agent’s Motivations	<b>100.0</b>	<b>100.0</b>	91.0	100.0	<b>100.0</b>	<b>100.0</b>
Intra-sentence	4: Two Argument Relations	77.9	<b>100.0</b>	61.0	96.2	96.0	<b>100.0</b>
	17: Positional Reasoning	55.3	95.8	51.0	49.9	53.0	94.9
	18: Size Reasoning	52.7	97.9	52.0	86.4	91.0	98.5
	19: Path Finding	51.4	<b>100.0</b>	8.0	12.6	14.0	<b>100.0</b>

Table 2: Test accuracies of our ensemble model and the other QA systems

Tasks	Ensemble	DMN+	LSTM	MemN2N	t-MEM-NN	EnDMN
1: Single Supporting Fact	<b>100.0</b>	<b>100.0</b>	50.0	<b>100.0</b>	<b>100.0</b>	<b>100.0</b>
2: Two Supporting Facts	<b>99.8</b>	99.7	20.0	99.7	82.0	98.6
3: Three Supporting Facts	<b>98.9</b>	<b>98.9</b>	20.0	97.9	57.0	88.8
4: Two Argument Relations	<b>100.0</b>	<b>100.0</b>	61.0	96.2	96.0	<b>100.0</b>
5: Three Argument Relations	<b>99.7</b>	99.5	70.0	85.9	88.0	<b>99.4</b>
6: Yes/No Questions	<b>100.0</b>	<b>100.0</b>	48.0	99.9	78.0	<b>100.0</b>
7: Counting	<b>98.1</b>	97.6	49.0	98.0	81.0	97.7
8: Lists/Sets	<b>100.0</b>	<b>100.0</b>	45.0	99.1	89.0	<b>100.0</b>
9: Simple Negation	<b>100.0</b>	<b>100.0</b>	64.0	99.7	86.0	<b>100.0</b>
10: Indefinite Knowledge	<b>100.0</b>	<b>100.0</b>	44.0	<b>100.0</b>	84.0	<b>100.0</b>
11: Basic Coreference	<b>100.0</b>	<b>100.0</b>	72.0	99.9	97.0	<b>100.0</b>
12: Conjunction	<b>100.0</b>	<b>100.0</b>	74.0	<b>100.0</b>	99.0	<b>100.0</b>
13: Compound Coreference	<b>100.0</b>	<b>100.0</b>	94.0	<b>100.0</b>	90.0	<b>100.0</b>
14: Time Reasoning	<b>100.0</b>	99.8	27.0	99.9	89.0	98.5
15: Basic Deduction	<b>100.0</b>	<b>100.0</b>	21.0	<b>100.0</b>	<b>100.0</b>	<b>100.0</b>
16: Basic Induction	<b>97.6</b>	54.7	23.0	47.9	46.0	54.2
17: Positional Reasoning	<b>96.7</b>	95.8	51.0	49.9	53.0	94.9
18: Size Reasoning	<b>98.5</b>	97.9	52.0	86.4	91.0	<b>98.5</b>
19: Path Finding	<b>100.0</b>	<b>100.0</b>	8.0	12.6	14.0	<b>100.0</b>
20: Agent’s Motivations	<b>100.0</b>	<b>100.0</b>	91.0	<b>100.0</b>	<b>100.0</b>	<b>100.0</b>
Mean test accuracy	<b>99.5</b>	97.2	51.9	88.7	81.0	96.5

In order to see the performance of ff-DMN on the other non-sequential tasks demonstrated in Table 1, we report the results in Table 2. Furthermore, the results for the other tasks in this group (tasks 5, 15 and 20) show that the performance of ff-DMN is comparable to DMN+.

The results of the intra-sentence group of tasks in Table 2 show that DMN+ has much higher prediction accuracy in all these four tasks. For tasks 17, 18 and 19, it seems without RNNs,

ff-DMN has difficulty in learning properly. This suggests that incorporating RNNs is crucial for the system to learn this type of reasoning ability.

We also build the ensemble model with DMN+ and ff-DMN. The results shown in Table 3 shows that the ensemble model outperforms the other systems on all the 20 tasks. The mean test accuracy 99.5% is the highest among all the systems.

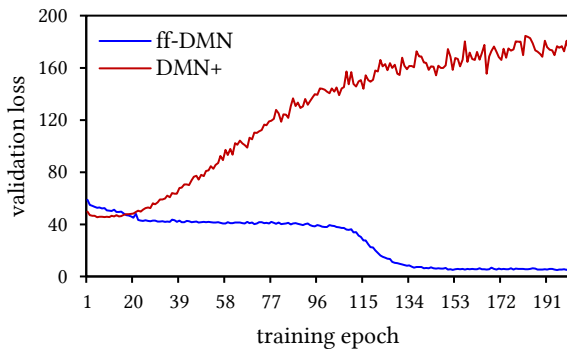


Figure 6: Validation losses of ff-DMN and DMN+ on task 16

## 4 Conclusions

By using two different architectures, one with a mixture of RNNs and FFNNs and another with only FFNNs, we studied the role RNN plays in a QA system. We used the tasks in the bAbI dataset for which the orders of facts are unimportant to compare the performance of the two architectures. The results show that for those tasks that involves inter-sentence reasoning, applying RNNs is not always beneficial. In fact, for task 16 that requires inductive reasoning, it is harmful. Removing RNN can remarkably improve the performance on this task. However, using RNN is crucial for the tasks that involves intra-sentence reasoning. By incorporating ff-DMN and DMN+, our ensemble model outperforms the other QA systems on all the 20 tasks. Future work will explore how the two types of reasoning could be solved by some mechanism in a single QA system.

## ACKNOWLEDGMENTS

The authors would like to thank Callaghan Innovation for the R&D Fellowship Grants and Ambit AI Limited for the research sponsorship.

## REFERENCES

- [1] Y. LeCun, Y. Bengio, and G. Hinton, *Deep learning* vol. 521, 2015.
- [2] Y. Sharma and S. Gupta, "Deep Learning Approaches for Question Answering System," *Procedia computer science*, vol. 132, pp. 785-794, 2018.
- [3] K. Ishwari, A. Aneez, S. Sudheesan, H. Karunaratne, A. Nugalayadde, and Y. Mallawarachchi, "Advances in Natural Language Question Answering: A Review," *arXiv preprint arXiv:1904.05276*, 2019.
- [4] S. Sukhbaatar, J. Weston, and R. Fergus, "End-to-end memory networks," in *Advances in neural information processing systems (NIPS)*, 2015, pp. 2440-2448.
- [5] C. Xiong, S. Merity, and R. Socher, "Dynamic memory networks for visual and textual question answering," in *International Conference on Machine Learning (ICML)*, 2016, pp. 2397-2406.
- [6] K. Tolias and S. P. Chatzis, "t-Exponential Memory Networks for Question-Answering Machines," *IEEE transactions on neural networks and learning systems*, 2018.
- [7] C. Yue, H. Cao, K. Xiong, A. Cui, H. Qin, and M. Li, "Enhanced question understanding with dynamic memory networks for textual question answering," *Expert Systems with Applications*, vol. 80, pp. 39-45, 2017.
- [8] H. Li, M. R. Min, Y. Ge, and A. Kadav, "A context-aware attention network for interactive question answering," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 927-935.
- [9] J. Weston, A. Bordes, S. Chopra, A. M. Rush, B. van Merriënboer, A. Joulin, *et al.*, "Towards ai-complete question answering: A set of prerequisite toy tasks," in *International Conference on Learning Representations (ICLR)*, 2016.
- [10] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *NIPS 2014 Workshop on Deep Learning, December 2014*, 2014.
- [11] Y. Shen, E. M.-K. Lai, and M. Mohaghegh, "The Role of RNNs for Context Representations: A Case Study Using DMN+," in *International Conference on Computational Linguistics and Natural Language Processing (CLNLP)*, 2020 (forthcoming).
- [12] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [13] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249-256.