Unseen Filler Generalization In Attention-based Natural Language Reasoning Models

Chin-Hui Chen[§], Yi-Fu Fu[§], Hsiao-Hua Cheng and Shou-De Lin
Dept. of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan
chchen.johnson@gmail.com, yifu.arlj@gmail.com, hsiaohuacheng@utexas.edu, sdlin@csie.ntu.edu.tw

2020 IEEE Second International Conference on Cognitive Machine Intelligence (CogMI) | 978-1-7281-4144-2/20/$31.00 ©2020 IEEE | DOI: 10.1109/COGMI50398.2020.00016

***Abstract*—Recent natural language reasoning models have achieved human-level accuracy on several benchmark datasets such as bAbI. While the results are impressive, in this paper we argue by experiment analysis that several existing attention-based models have a hard time generalizing themselves to handle name entities not seen in the training data. We thus propose Unseen Filler Generalization (UFG) as a task along with two new datasets to evaluate the filler generalization capability of a natural language reasoning model.**

We also propose a simple yet general strategy that can be applied to various models to handle the UFG challenge through modifying the entity occurrence distribution in the training data. Such strategy allows the model to encounter unseen entities during training, and thus not to overfit to only a few specific name entities. Our experiments show that this strategy can significantly boost the filler generalization capability of three existing models including Entity Network, Working Memory Network, and Universal Transformers.

***Index Terms*—machine reasoning, attention, unseen filler, memory-augmented neural network, transformer.**

I. Introduction

Researchers of machine learning [1]–[3] have started to move on from perception tasks to cognition problems, which leads to a surging area called Natural Language Reasoning (NLR). Several memory augmented neural network models have been proposed to conquer NLR problems. Memory network [4] embeds memory matrix component into the existing neural network architecture, so as to efficiently act as a dynamic knowledge base to help with reasoning tasks. Recently, Transformer [5] has arisen to be another new strong competitor, which uses self-attention layers to replace conventional recurrent neural network (RNN), so as to achieve paralellizability and global receptive field. Based upon transformer, Universal Transformer (UT) [6] is claimed to gain even stronger reasoning ability by adding recurrent inductive bias of RNNs as well as a dynamic per-position halting mechanism. With UT, the state-of-the-art result achieves a mean error of less than 0.3% in bAbI [2], which is a commonly used benchmark dataset for machine reasoning composed of 20 fundamental reasoning tasks.

Recently, researchers have raised questions about whether the reasoning models really learn the expected behaviors. In visual question answering, [7] exposes the weakness of the models' pathological behaviors, such as "tend to fail on sufficiently novel instances" and "jump to conclusions". In Reading Comprehension, existing neural systems can be fooled by appending only one adversarial sentence [8]. In NLR, despite the high accuracy, we have found that the state-of-the-art models are trained overfittingly and can be attacked by simply replacing name entities. We argue that if the models indeed learn how to reason from the content, then it should still answer with high accuracy even if the names in the test data are replaced with novel ones. For example, if we modify the test data from "John travelled to the hallway. Where is John?" to "Alice travelled to the hallway. Where is Alice?", we would expect NLR models to still give the answer correctly. However, as will be shown later in Section III, we found that the performance drops about 30% when test data is composed of novel entities. That is to say, when confronting entities the machine has never seen, it performs reasoning poorly.

In cognitive linguistics, a widely known theory called tensor product representation [9]–[11] decomposes innate language structure into filler-role bindings. For example, in the statement "Mary journeyed to the bathroom", the filler (Mary) is bound to a specific role, which is the concept of someone journeying to somewhere. Specifically, no matter what filler, seen or unseen, is assigned to a specific role, the logic of "[Filler] journeyed to the bathroom" remains identical. Under the viewpoint of filler and role, we observe that current NLR models suffer from systematic generalization toward unseen fillers, which means modern models perform badly when encountering fillers that is not seen in the training data. Take language innateness into consideration, we would expect a perfect reasoning model to learn the hidden logic rules on role level instead of binding a role to some specific fillers.

In cognitive science, there is a long debate on language acquisition between Innatism and Behaviorism. Innatism advocates Chomsky's innate principle of language [12], while Behaviorism supports Skinner's behaviourist perspective [13], [14]. As the pioneer of Behaviorism, Skinner advocated that language learning proceeds with environmental factors. He argued that language learning is typically through classical or operant conditioning, which can be analogized to current data-driven AI trend that most of current AI models are purely trained by data starting from a blank slate. Chomsky, however,

[§]Equal contribution

978-1-7281-4144-2/20/$31.00 ©2020 IEEE
DOI 10.1109/CogMI50398.2020.00016

	Training/Validation Data	Test Data
Task 1	*story*: **Mary** moved to the bathroom. **John** went to the hallway. *query:* Where is **Mary**? *answer:* bathroom	*story:* **Carol** moved to the hallway. **Alice** travelled to the office. *query:* Where is **Alice**? *answer:* office
Name Entities	{Daniel, John, Mary, Sandra}	{Alice, Bob, Carol, Dave}

TABLE I: Training/validation instance and created test instance of task 1 of bAbI dataset are illustrated. The names in training and validation data is sampled from the same set of names while the names in the test data is sampled from unseen set of names.

proposed the idea that "human languages, as diverse as they are, do share some fundamental similarities, and that these similarities are attributable to unique innate principles" [15]. Recently, scholars [3], [16], [17] have attempted to look for the missing innateness in current AI models in order to develop more human-like AI. In this paper, we will largely emphasize on the innateness of filler and role in NLR, and expect NLR models to learn these important concept during training. In Section IV, we will furthermore propose a behavioral strategy as a preliminary method trying to capture this innate concept.

In this paper we demonstrate by experiment that three state-of-the-art, attention-based NLR models tend to overfit on fillers in training data, and they are unable to unbind role from filler in test data with unseen fillers. Thus, we propose a new challenge, Unseen Filler Generalization (UFG), for NLR models. We reconstruct two reasoning datasets by updating test data with unseen fillers, with the intention to evaluate the model's ability of filler generalization. After that, we take the first step toward solving this challenge by proposing a strategy of Stochastic Entity Replacement (SER). SER is a one-shot data-driven training approach to force model to unbind role from filler during training phase. We show that by adopting SER, the reasoning performance toward unseen fillers in test data can be improved significantly.

The main contributions of this study are summarized as below:

1) To our knowledge, this is the first work that directly points out the unseen filler generalization problem lying within the attention mechanism of NLR models. We investigate into three state-of-the-art models: Entity Network, Working Memory Network and Universal Transformers.
2) As most of the existing reasoning datasets are not suitable for testing the filler generation capability, here we release two modified NLR datasets (UFG-bAbI, UFG-CLUTRR) [1] to evaluate models' capability of filler generalization.
3) We propose a general one-shot learning strategy to solve this task, and show that it does improve the UFG ability.

The remainder of the paper is structured as follows. In Section II, we perform analyses on three state-of-the-art models to demonstrate the filler generalization problem. In Section III, two augmented datasets, UFG-bAbI and UFG-CLUTRR are detailedly introduced as well as the performance of state-of-the-art models. In Section IV, we introduce the details of SER along with the corresponding experiments. Finally, the conclusion is summarized in Section VI.

[1]https://github.com/ntumslab/ufg

II. Filler Generalization Problem

This section offers a deeper analysis about the overfitting of fillers in several existing models. Here we choose three state-of-the-art, attention-based models in NLR, Universal Transformers (UT) [6], Working Memory Network (W-MemNN) [18], and Entity Network (EntNet) [19]. Universal Transformers model is the leading representative from self-attentive recurrent sequence models. In addition, among all the attention-based memory models, Working Memory Network introduces relational components while entity modeling concept is proposed in Entity Network.

We use task 1 of the bAbI dataset to demonstrate the overfitting toward namesets in training data. Table I gives a simple example of the task. The task is composed of stories and queries regarding four persons and their action. Each sentence in a story states that one person moves to somewhere else, and the final query is a question simply asking the location of one of the four persons. In task 1 of the original bAbI dataset, there are only four constant names among all training, validation, and test data, which are John, Mary, Sandra, and Daniel. To examine the reasoning capability when encountering unseen names, we create new test data by substituting the names in the original test data into novel ones, for example, Alice, Bob, Carol, and Dave.

We then first train all three models with original training and validation data. And after training the models as noted in original papers, we separately test the models with original test data and new test data. We will show and analyze the phenomenon that the trained models tend to perform better on the original test data, and significantly worse on the new test data. Since the model design differs among all three models, below we provide diverse analyses to explain the observation. Nevertheless, the ideas behind the following three demonstrations are similar, which are to show that the models learn some neuron-level patterns that benefits the reasoning with seen names, but such patterns vanish when confronting unseen entities.

A. UT: Universal Transformers

Universal Transformer [6] is one of the latest state-of-the-art models in NLR. Fig. 1 presents the layout of UT. Based

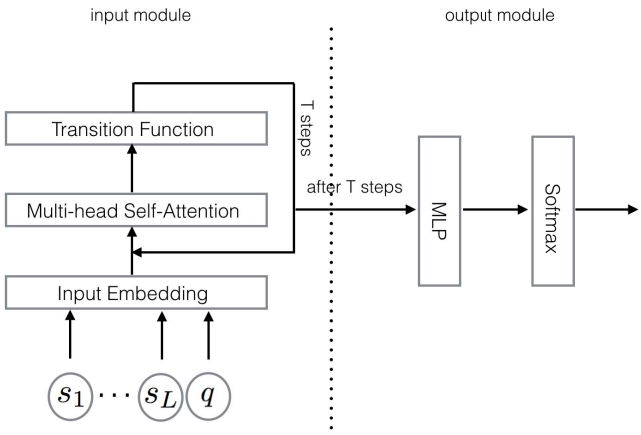


Fig. 1: The network diagram of UT.

upon vanilla Transformer [5], UT emphasizes recurrence over depth and also add dynamic per-position halting mechanism. The input story sentences $(s_1...s_L)$ along with query q are encoded using positional encoding first. Next, self-attention layer repeats in variable iterations determined by an Adaptive Computation Time algorithm proposed in [20]. In each iteration, queries Q, keys K, and values V of each hidden state are retrieved with parameter matrices. Each hidden state attends to others through a compatibility function of its own Q and other's K. For bAbI tasks, only the encoder part of the model and a succeeding multi-class classifier are used. The formula of the scaled dot-product attention is:

$$Attention(Q,K,V) = Softmax(\frac{QK^T}{\sqrt{d}})V$$

where Q,K and V are queries, keys and values of hidden states, and d is the number of columns of Q,K and V.

We first trained a UT model with original training data till it achieves error $< 5\%$. Next, we focus on the first iteration of self-attention layer, which directly operates self-attention mechanism among input sentences. We classify the input sentences and queries by the name they involve, into four groups. Then we examine the attention distribution of each group of query, to each group of input sentences.

Fig. 2(a) shows that after trained with original training data, when testing with the same nameset (John, Mary, Sandra, Daniel), queries can be perfectly attended to the exact sentences containing the same name, so as to perform succeeding reasoning task. However, this ability is only limited to the names appearing in the training data. In Fig. 2(b) we can see that when using unseen names in test data, model loses the ability to attend queries accurately, and thus may not perform the task well. We expect UT as a less entity-centered model, as well as possessing similarity-based self-attention mechanism to be free from problem of filler generalization. It turns out even the latest model still suffers from it.

B. W-MemNN: Working Memory Network

Working Memory Network [18] is a model that combines Memory Network and Relational Network [21]. It consists of three main modules: an input module, an attentional controller, and a reasoning module. Fig. 3 presents the layout of W-MemNN. The input story with sentences $(s_1...s_L)$ is first encoded by gated recurrent unit (GRU), into memories $m_1...m_L$. Next, several hubs with multi-head attention mechanism are sequentially arranged, to extract information from $m_1...m_L$. Each attention head has an independent projection matrix W to calculate the attention α toward every sentence. Finally, the hidden state of all hubs joint pairwisely together with query, and go through the reasoning module, to generate the answer to the query. With the same design of the original paper, we use four hubs and eight-headed attention mechanism in each hub. Within each attention head, the formula of the attention toward sentence memory m_i is:

$$m_i' = W_m m_i$$
$$\alpha_i = Softmax(\frac{u^T m_i'}{\sqrt{d}})$$

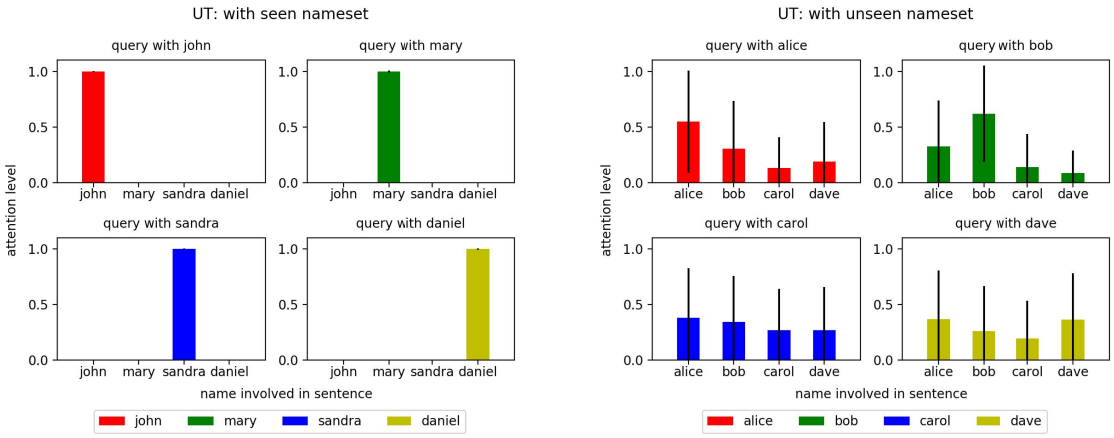
where W_m is the projection matrix of attention head m, u is the query embedding or previous hub's output, d is the dimension of m_i, which acts as a normalizer here.

We first trained a W-MemNN using original training data till early stopping is performed[2], and then observe the attention distribution toward each sentence. We can see that in Fig. 4(a), where the original test data is used, each attention hub (consisting of 8 consecutive attention heads, namely 1-8, 9-16, 17-24, 25-32) has high attention level to sentences containing one specific name. This phenomenon shows that W-MemNN probably performs reasoning in the original test data by assigning each attention hub to take charge of one name, and store related information of that name. However, in Fig. 4(b), this phenomenon vanishes when the new test data with unseen names are used. With new test data, attention hubs can not focus on specific name as they did with original test data. That is to say, machine possibly learned overfitting specialization toward seen names in training data, and is not capable of dealing with unseen names in the new test data.

C. EntNet: Entity Network

Entity Network [19] is the first memory network designed for learning and memorizing entities in latent space. Fig. 5 presents the layout of EntNet. The model consists of three main modules: an input encoder, a dynamic memory, and an output layer. The input story with sentences $(s_1...s_L)$ is first encoded using positional encoding and fed into recurrent dynamic memory component. As the sentence embedding feeding into model one by one, independent memory slots update their hidden state according to the input gate level g of each slot toward the sentence. Finally, the answer to the query is calculated according to the content h of each memory slot,

[2]W-MemNN experiment code is based on the github repository: https://github.com/jgpavez/Working-Memory-Networks



(a) UT with seen entities

(b) UT with unseen entities

Fig. 2: The analysis of self-attention mechanism of UT. Queries and sentences are classified into four groups according to the name involved. In Fig. 2(a), we can see query attends well to the sentences of same group. In Fig. 2(b), however, query attends disorderly.

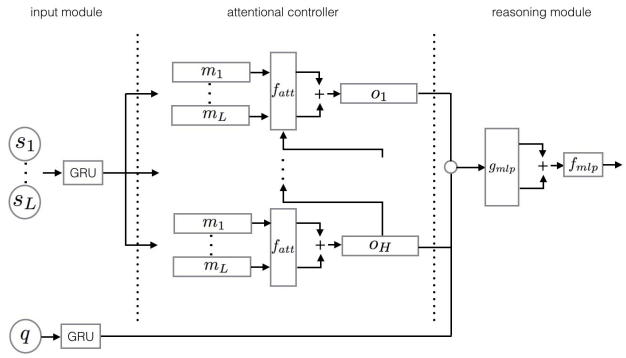


Fig. 3: The network diagram of W-MemNN.

query embedding q, and the attention level p of each slot to the query. Specifically, the update rule of the dynamic memory is:

$$g_j \leftarrow \sigma(s_t^T h_j + s_t^T w_j)$$
$$\tilde{h_j} \leftarrow \phi(U h_j + V w_j + W s_t)$$
$$h_j \leftarrow h_j + g_j \odot \tilde{h_j}$$
$$h_j \leftarrow \frac{h_j}{\|h_j\|}$$

where h_j is the hidden state stored in memory slot j, w_j is the "key" of memory slot j, and g_j is the gate level of memory slot j when encountering s_t. U, V, W are transition matrices in regard of updating dynamic memory. And the formula in output module is:

$$p_j = Softmax(q^T h_j)$$
$$u = \sum_j p_j h_j$$
$$y = R\phi(q + Hu)$$

where q is the question embedding, and p_j is the attention level that the model put on memory slot j, when confronting the query q. H, R are transition matrices to retrieve the output.

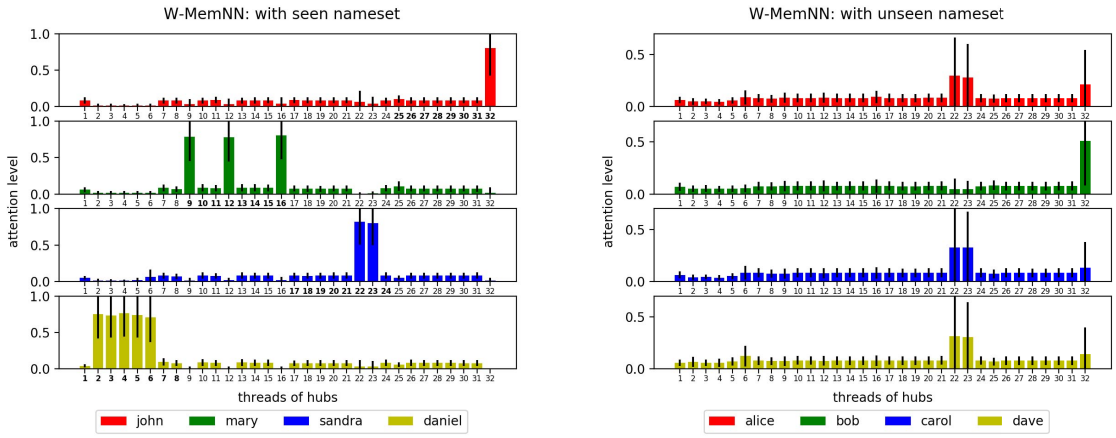
After an EntNet is trained with designed early stopping criteria[3], we probe the gate and attention distribution in each memory slot during different phases. Fig. 6(a) and Fig. 6(b) both consist of four subplots. Each subplot is related to the sentences containing one certain name. In each subplot, the left part shows the gate level of each memory slot during sentence feeding. The right part of the subplot is the attention level of each memory slot during answering. We can see that in Fig. 6(a), gate level during sentence feeding and attention level during answering align similarly in all four subplots. Moreover, in each subplot there are one or more dominant memory slots (notated in bold with *) that have at the same time both high gate level during sentence feeding and high attention level during answering. This means that the dominant memory slot is responsible for both storing and retrieving information related to the certain name. It is probable that EntNet learns to arrange memory slots to take charge of specific name entity. However, in Fig. 6(b), when confronting unseen entities, the phenomenon vanishes. In all four subplots of Fig. 6(b), no memory slot has at the same time both high gate level during sentence feeding and high attention level during answering. This means that the model may not perform reasoning task well in the way it did with seen name entities.

III. Dataset Design and Experiment

A. Dataset

In order to examine filler generalization capability of an NLR model, we propose a task called Unseen Filler Generalization (UFG). In UFG we randomly replace name entities in

[3]EntNet experiment code is based on the github repository: https://github.com/jimfleming/recurrent-entity-networks



(a) W-MemNN with seen entities

(b) W-MemNN with unseen entities

Fig. 4: The analysis of attention mechanism of W-MemNN. Sentences are classified into four groups according to the name involved. The attention level of each attention head toward each group of sentences are displayed. In Fig. 4(a), each attention hub (attention head 1-8, 9-16, 17-24, 25-32) is specialized to attend to one group of sentences. In Fig. 4(b), however, attention heads attend disorderly.

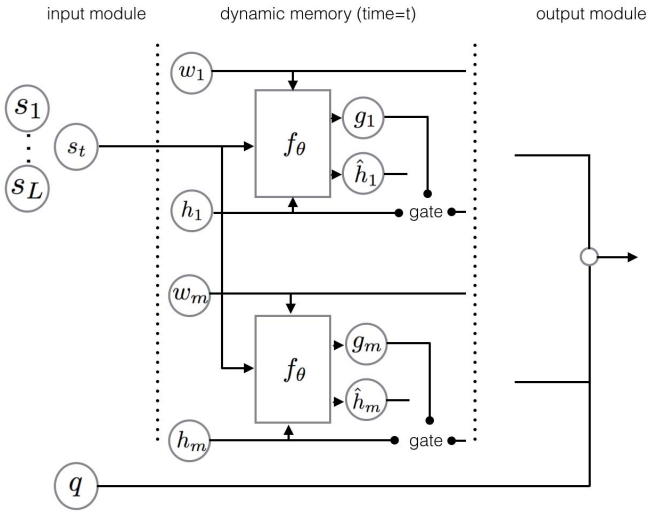


Fig. 5: The network diagram of EntNet.

the test data with unseen ones which are not existing in the training data. Formally, we create an unseen entity pool first, and UFG-modification means to pick certain number of name entities from the pool, and replace the name entities in test data respectively. In this way, while still evaluating the reasoning ability of models, UFG confines the evaluation toward unseen name entities, which makes the evaluation rely greatly on the generalization capability of models.

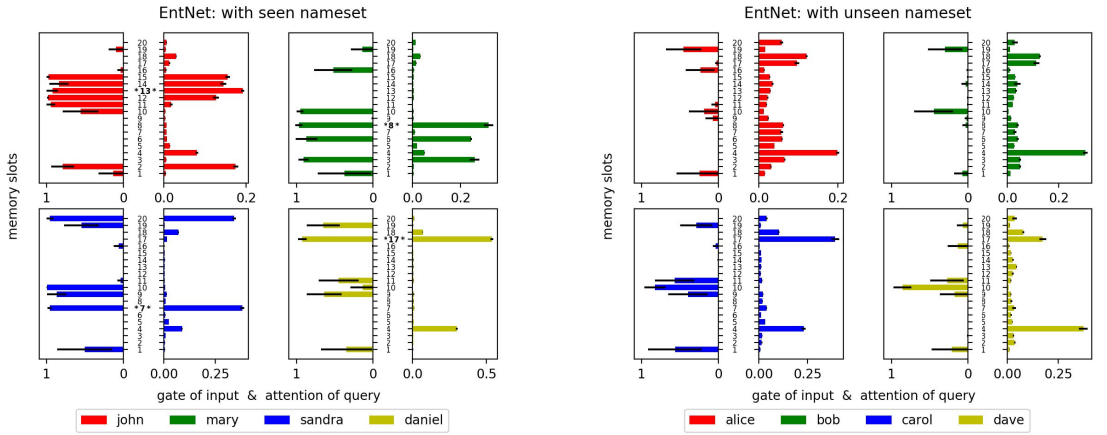
In this work, we propose two datasets, UFG-bAbI and UFG-CLUTRR as our benchmarks. UFG-bAbI and UFG-CLUTRR are variant versions of two widely representative tasks, bAbI [2] and CLUTRR [22], [23]. The former is composed of logic tasks that a reasoning system should require; the latter measures a reasoning system's systematic generalization of logic rules by evaluating on held-out combinations of relation.

1) UFG-bAbI Dataset: bAbI is designed to simulate human reasoning between entities and relations. A typical story and question of bAbI is depicted in the left part of Table I. We apply UFG-modification to bAbI dataset and yields the UFG-bAbI dataset to evaluate filler generalization capability. In this work, only tasks in bAbI that contains name entities in stories and queries are selected. We do not consider tasks that the name entities exist in answers because it means that models need to make prediction on a new class when meeting unseen filler, which is much more challenging. The final chosen tasks are Task 1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14, 20. Novel entities are randomly sampled in test data as illustrated in the right part of Table I.

2) UFG-CLUTRR Dataset: CLUTRR is proposed to evaluate combinatorial generalization ability on relational reasoning. Table II shows an example with k=3, where k stands for the length of the reasoning path. Given kinship description in story, the machine is asked to predict the relationship between the first entity and the last one. In this work, "k=2,3/k=2", "k=2,3/k=3", "k=2,3/k=4" tasks are chosen. For example, "k=2,3/k=2" means that the reasoning path may be 2 or 3 in training data, and would be 2 in test data. We apply UFG-modification to the dataset generation process to yield UFG-CLUTRR dataset.

B. Benchmarking the Current NLR Models

We then use the proposed UFG-bAbI and UFG-CLUTRR datasets as benchmarks to test the ability of filler generalization on the three NLR models. In addition to directly solving tasks with models, we also perform two embedding methods, GloVe [24] and BERT [25], to examine whether lexical and contextualized embeddings would give help. We will first elaborate the experiment setup, and then reveal our result and discussion.



(a) EntNet with seen entities

(b) EntNet with unseen entities

Fig. 6: The analysis of attention mechanism of EntNet. Sentences and queries are classified into four groups according to the name involved. The gate level of each slot toward each group of sentence during sentence feeding is displayed in the left part of each subplot. The attention level during answering is displayed in the right part of each subplot. In Fig. 6(a), we can see in each subplot the left and right part align similarly, and there are one or more dominant memory slots (notated in bold with *) with both high level of left and right part at the same time, which means to be responsible for both storing and retrieving information related to the certain name. However, in Fig. 6(b), this phenomenon vanishes.

	Training/Validation Data	Test Data
$k = 3$	**Timothy** is **Amy**'s son. **Timothy** has a wife who is **Rose**. ... **Elbert** is a son of **Rose**. Predict: **Elbert** is **Amy**'s grandson.	**Kaiden** has a daughter called **Matilda**. **Lennox** is **Matilda**'s husband. ... **Gage** is **Lennox**'s son. Predict: **Kaiden** has a grandson who is **Gage**
Name Entities	{Timothy, Amy, Rose, Elbert}	{Kaiden, Matilda, Lennox, Gage}

TABLE II: Training/validation instance of CLUTRR and test instance created by UFG-modification are illustrated. The training and validation data is sampled from the same set of names while the test data is sampled from novel entities. Here k=3 means the length of relation path is three, and therefore four name entities are involved.

1) Experiment Setup: For UFG-bAbI, we follow the design of bAbI that each story is composed of four name entities, and the size of seen entity pool and unseen entity pool are exactly four. For UFG-CLUTRR, following the settings from original paper, each story contains k+1 name entities, while the size of seen entity pool and unseen entity pool are both 300. Both datasets are generated with UFG-modification depicted above.

We train the models using the training data and decide the stopping criteria using validation data, both of which are with seen entities. Model performance are evaluated using test data, which is with unseen entities.

We first examine the performance of all three models on the original bAbI and CLUTRR dataset. Next, we test the model with UFG-bAbI and UFG-CLUTRR, to demonstrate the inherent performance decay with unseen entities. The result serves as a primary baseline. Next, a simple solution one might come up with to tackle with unseen entities, is to make use of pre-trained word embeddings, since pre-trained embeddings might be able to solve the out-of-vocabulary nature of unseen fillers. Thus, we conduct two naive embedding methods as baseline solutions. The first one is to use pre-trained GloVe embeddings [24]. The other one is to make use of the contextualized sentence embeddings, BERT [25], such that not only to solve the out-of-vocabulary problem, but also to catch the contextualized meaning surrounding the unseen fillers. Two embedding methods are tried on UT, which is the best model among three.

C. Results and Discussion

1) UFG-bAbI: Table III shows the performance of three models in original bAbI and UFG-bAbI, on different tasks. We can see that all three models perform well in original bAbI, while the performance drops significantly in UFG-bAbI. The value in the parentheses indicates the performance degradation of UFG-bAbI compared to original bAbI tasks. We can also observe a trend that the filler generalization capability improves from EntNet to W-MemNN then to UT, as the degree of entity-centered characteristic decreases. However, even in UT, there is still a large gap of the accuracy between seen and unseen fillers. In addition, we experiment on the pre-trained embedding method with UT as showed in the last two columns. The result tells that the pre-trained embedding method gives limited help, and even worse to have negative effect in some tasks. With the experiment result, we want to

point out that the essence to solving UFG problem may not lie in the way to do with fixed, pre-trained word or sentence embeddings.

It is worth noting that Task 20 performs strongly and doesn't degrade as expected. The UFG-bAbI worst case is still 0.882 at least and in UT the accuracy is nearly perfect. If we look into the data, we can find that each question sentence is followed by an informative story line immediately which is sufficient to answer the question. And there is no multi-hop case. For example, in story/question: "Yann went to the kitchen. Why did Yann go to the kitchen? thirsty," model can easily infer answer by memorizing kitchen and thirsty pair. Althought Yann is unseen filler here, the model is trained to only look for the last noun (kitchen) in the previous sentence instead of name entities (Yann).

2) UFG-CLUTRR: Table IV shows the performance of the three models in the original CLUTRR and UFG-CLUTRR dataset under different task settings. In the left part of the table, all models perform relatively well on original CLUTRR dataset. The error arises when k in the denominator increases, which is consistent with the finding in the original paper that task difficulty increases with longer reasoning path. The "Unseen" part of the table shows the performance degradation due to unseen filler generalization challenge. We can see that all models have a hard time to handle the unseen entities, and the accuracy drops even more significantly than that in UFG-bAbI. It is probably because CLUTTR is inherently more difficult than bAbI. Again, we observe that UT suffers from filler generalization less than the other two models, and neither GloVe nor BERT provides beneficial effect to the problem.

IV. Stochastic Entity Replacement

Seeing that pre-trained embeddings method such as GloVe and BERT provide limited help to UFG problem, here we propose a data-driven one-shot learning approach called Stochastic Entity Replacement (SER). SER aim to prevent model from overfitting to name entities in training data, and guide model to treat the fillers more generally. Simply saying, we replace the name entities in training data stochastically. To elaborate, the name entities in each training sample are replaced with unseen ones during every batch iteration. That is to say, models now deal with every unseen filler for "one-shot" in every iteration, and will never encounter the same entity again during the whole process. In this way, SER is actually providing infinite novel entities for the training phase, until the early stop condition is met. The main purpose of creating diversity of entities in training data is to make sure the model does not learn to embed information in the specific entities.

By applying SER, training difficulty will increase because of facing distinct name entities in each batch iteration. However, it is exactly the increasing of training difficulty that force models not to overfit to specific name entities, and enhance the generalization ability. Models trained with this method may generalize better since they are facing same distribution of entities between training and test data. With a broader viewpoint, we force the model during training phase to optimize toward the actual goal, the generalization toward unseen fillers.

In the cognitive science perspective, SER is the hybrid training strategy that bridging behavior learning and language innateness. By designing environmental factors (SER data-driven training), neural models learn to develop language innateness properties (unseen filler capability).

In our implementation, as to replace the entities with unseen ones in every iteration, what we technically do is to randomize the embedding of name entities at the beginning of each batch iteration. In this way, novel random embedding is treated as a brand-new entity just like using a novel name entity in high level. And model will not bind role to specific entity since the optimization of name embedding during training is nullified.

A. Performance of SER on UFG

SER is a general strategy that can be applied to an arbitrary NLR model. Here we apply it to the three models and test on UFG-bAbI and UFG-CLUTRR datasets. The experiment setup and training details follow the same configurations as depicted in Section III.

1) UFG-bAbI: Table V shows the performance of models trained with SER in UFG-bAbI. The value in parentheses indicates the difference of accuracy comparing to performance without SER in Table III. In general, the average performance of SER-augmented models has well improved. Among three models, SER-UT performs best, but SER-EntNet gains the most improvement comparing to the result without SER. It is reasonable that EntNet benefit most from SER, since it is the most entity-centered model among three. However, we also notice that in some tasks, applying SER does not gain improvement and even worsen the performance. It is probably because those tasks are relatively difficult to be learned with SER, so that models could not converge before early stopping was met. This phenomenon is prominent especially in task 2 and 3, which is consistent with previous results [19] that the models might have a hard time solving multiple supports problem in stories.

2) UFG-CLUTRR: Table VI presents the experimental results of SER method in UFG-CLUTRR. In general, all SER-augmented models achieve better accuracy regardless of the length of k. SER-UT again obtains the best result among three. In addition, the accuracy decreases as the underlying reasoning path increases, confirming with the design of logic rules generalization [23] that task gets harder with longer reasoning path. We also notice that the improvement by SER is much greater in UFG-CLUTRR than in UFG-bAbI. However, even though SER-augmented models greatly improve the result toward UFG, the gap between performance of seen and unseen entities remains noticeable, which means there are still room to improve for this filler generalization problem.

B. Analysis of SER

To further investigate the effect of SER to the models, we conduct the same analysis as in Section II. We want to verify that whether the failure of attention mechanism toward unseen

Task	Seen (bAbI)			Unseen (UFG-bAbI)				
	EntNet	W-MemNN	UT	EntNet	W-MemNN	UT	GloVe-UT	BERT-UT
1	1.000	1.000	1.000	0.337 (-66%)	0.526 (-47%)	0.557 (-44%)	0.564 (-44%)	0.502 (-50%)
2	0.970	0.998	0.992	0.355 (-63%)	0.478 (-52%)	0.563 (-43%)	0.408 (-59%)	0.482 (-51%)
3	0.966	0.835	0.971	0.531 (-45%)	0.508 (-39%)	0.579 (-40%)	0.516 (-47%)	0.372 (-62%)
6	1.000	1.000	0.998	0.728 (-27%)	0.787 (-21%)	0.844 (-15%)	0.813 (-19%)	0.832 (-17%)
7	1.000	0.979	0.973	0.657 (-34%)	0.702 (-28%)	0.745 (-23%)	0.748 (-23%)	0.772 (-21%)
8	1.000	0.996	0.984	0.427 (-57%)	0.693 (-30%)	0.749 (-24%)	0.699 (-29%)	0.723 (-27%)
9	1.000	0.999	1.000	0.712 (-29%)	0.779 (-22%)	0.894 (-10%)	0.789 (-21%)	0.873 (-13%)
10	1.000	0.995	0.999	0.898 (-10%)	0.667 (-33%)	0.947 (-5%)	0.929 (-7%)	0.901 (-10%)
11	1.000	0.999	1.000	0.549 (-45%)	0.748 (-25%)	0.772 (-2%)	0.779 (-22%)	0.627 (-37%)
12	1.000	0.999	1.000	0.758 (-24%)	0.772 (-23%)	0.772 (-2%)	0.774 (-23%)	0.770 (-23%)
13	1.000	1.000	0.998	0.820 (-18%)	0.880 (-12%)	0.940 (-6%)	0.938 (-6%)	0.927 (-7%)
14	1.000	0.987	0.982	0.803 (-20%)	0.481 (-51%)	0.931 (-5%)	0.826 (-16%)	0.912 (-7%)
20	1.000	1.000	1.000	0.978 (-2%)	0.882 (-12%)	1.000 (0%)	0.999 (0%)	0.999 (0%)
Avg	0.995	0.983	0.992	0.658 (-34%)	0.685 (-30%)	0.792 (-20%)	0.752 (-24%)	0.746 (-25%)

TABLE III: Test accuracy of bAbI and UFG-bAbI are reported when using `EntNet`, `W-MemNN`, `UT` as reasoning models. The left part of table presents models tested with seen entities while the right part with unseen entities. `GloVe-UT` and `BERT-UT` stand for using GloVe and BERT for pre-trained word embedding with UT as the model. Value inside parentheses indicates the performance degradation of UFG-bAbI compared to original bAbI tasks.

Task	Seen (CLUTRR)			Unseen (UFG-CLUTRR)				
	EntNet	W-MemNN	UT	EntNet	W-MemNN	UT	Glove-UT	BERT-UT
k=2,3/k=2	0.962	0.923	1.000	0.399 (-59%)	0.302 (-67%)	0.616 (-38%)	0.491 (-51%)	0.495 (-51%)
k=2,3/k=3	0.868	0.874	0.973	0.422 (-51%)	0.417 (-67%)	0.455 (-53%)	0.408 (-58%)	0.507 (-48%)
k=2,3/k=4	0.757	0.741	0.710	0.363 (-52%)	0.389 (-48%)	0.372 (-48%)	0.382 (-46%)	0.302 (-57%)
Avg	0.862	0.846	0.894	0.395 (-54%)	0.369 (-56%)	0.481 (-46%)	0.427 (-52%)	0.435 (-51%)

TABLE IV: Test accuracy of CLUTRR and UFG-CLUTRR are reported when using `EntNet`, `W-MemNN`, `UT` as reasoning models. The left part of table presents models tested with seen entities while the right part with unseen entities. `Glove-UT` and `BERT-UT` stand for using GloVe and BERT for pre-trained word embedding and UT as the model. Value inside parentheses indicates the performance degradation of UFG-CLUTRR compared to the original CLUTRR tasks.

name entities are resolved by SER. Here we conduct the analysis on UT and EntNet, which gain notable improvement with SER.

Fig. 7(a) shows the self-attention distribution in UT, of each group of query to each group of sentence. While in Fig. 2(b) we observe that UT cannot attend query to the sentence of same group when facing unseen names, here we show that by training with SER, UT can attend correctly in the new test data, and thus gain the capability of UFG. On the other side, Fig. 7(b) shows that EntNet trained with SER also exhibit similar pattern we argue in Fig. 6(a), that for each unseen name entity, one or more memory slots are dynamically allocated for retrieving and storing information of that name. These two demonstrations provide an insight that SER indeed causes substantial influence to model training, and really can make models into a more generalizable structure.

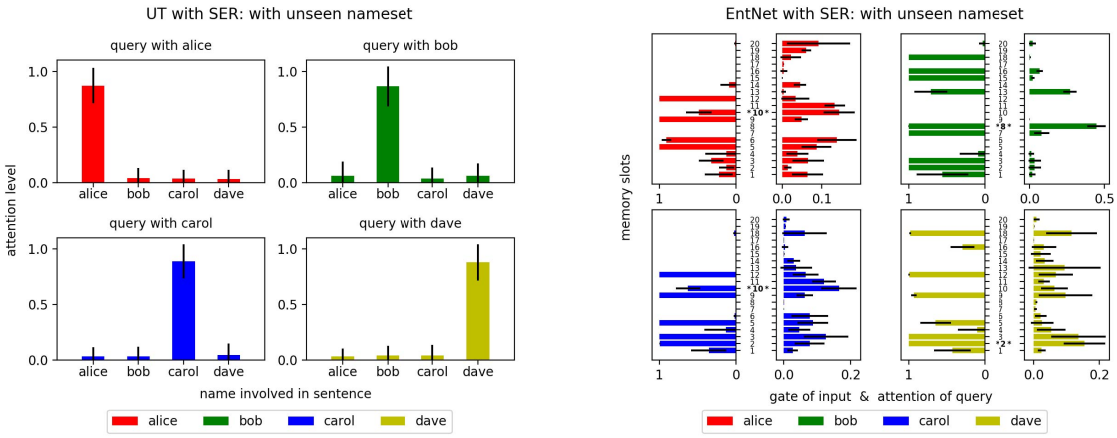
task	SER-EntNet	SER-W-MemNN	SER-UT
1	0.967 (187%)	0.510 (-3%)	1.000 (80%)
2	0.391 (10%)	0.455 (-5%)	0.415 (-26%)
3	0.523 (-2%)	0.505 (-1%)	0.513 (-11%)
6	0.953 (31%)	0.830 (5%)	0.931 (10%)
7	0.941 (43%)	0.798 (14%)	0.790 (6%)
8	0.984 (130%)	0.757 (9%)	0.975 (30%)
9	0.960 (35%)	0.849 (9%)	0.889 (-1%)
10	0.977 (9%)	0.752 (13%)	0.997 (5%)
11	0.706 (29%)	0.726 (-3%)	0.708 (-8%)
12	0.766 (1%)	0.765 (-1%)	0.973 (26%)
13	0.943 (15%)	0.944 (7%)	0.944 (0%)
14	0.869 (8%)	0.607 (26%)	0.971 (4%)
20	0.996 (2%)	0.980 (11%)	0.999 (0%)
Avg	0.844 (28%)	0.729 (6%)	0.854 (8%)

TABLE V: Performance of models training with SER on UFG-bAbI. Test accuracy of SER-augmented models has well improves. Value inside the parentheses indicates the difference of accuracy comparing to performance without SER in Table III.

V. Related Work

A. Machine Reasoning Datasets

Machine Reasoning is a recently emerging research direction in machine learning community [1]. Several natural language reasoning datasets are proposed to address this problem. bAbI [2] is the first synthetic textual reasoning on relational reasoning. Some of the variants are introduced such as dialog-based bAbI [26] and theory of mind [27]. CLUTRR [22], [23] is a bAbI-like benchmark which evaluates combinational generalization of relations and model robustness. On the other hand, for visual question reasoning, CLEVR [28] is proposed to grounded language and vision. NLVR [29], successor of CLEVR, presents strong challenge in current state-of-the-art models. Otherwise, GQA [30] claims to be the first real world image vision reasoning dataset. Note that, in Machine



(a) UT with SER and unseen entities

(b) EntNet with SER and unseen entities

Fig. 7: Improvement of attention mechanism toward unseen name entities after trained with SER. Fig. 7(a) shows that after UT trained with SER, query attends well to sentences of the same group. Fig. 7(b) shows that after EntNet trained with SER, memory slots display similar pattern as discussed in Fig. 6(a)

task	SER-EntNet	SER-W-MemNN	SER-UT
k=2,3/k=2	0.785 (97%)	0.761 (152%)	0.846 (37%)
k=2,3/k=3	0.675 (60%)	0.503 (21%)	0.663 (46%)
k=2,3/k=4	0.555 (53%)	0.624 (60%)	0.545 (47%)
Avg	0.672 (70%)	0.629 (70%)	0.685 (42%)

TABLE VI: Performance of models training with SER on UFG-CLUTRR. Test accuracy of SER-augmented models significantly improves. Value inside the parentheses indicates the difference of accuracy comparing to performance without SER in Table IV.

Reasoning community, all NLR datasets are synthetic and still the prevailing trend [18], [31], [32], since the language reasoning data need to be generated by handcrafted logic rules.

B. Machine Reasoning Models

Recently, several types of reasoning models are proposed. One of the main stream is attention-based memory-augmented neural network (MANN). Memory Networks (MemNNs) [4], [33], [34] and neuro-inspired Neural Turing Machines (NTMs) [35], [36] are the two representative categories of MANNs. Inspired by MemNNs, Entity Network (EntNet) [19] make network itself learn how to read and write the memory. To build relations between memory hops, Working Memory Network (W-MemNN) [18] augments relational reasoning module [21] to separate memory module from reasoning module. Lastly, by using self-attention layers and recurrent inductive bias of RNNs, Universal Transformer (UT) [6] is proved to gain even stronger reasoning capability. In addition to attention-based memory models, ILP-based [37] model is the promising knowledge representation and reasoning direction, which combines statistical NLP parser and Inductive Logic Programming module. The logical-based approach may treat the entities as variables, and the unseen filler generalization problem seems to be softened. However, it requires a robust and foreseeing NLP parser to translate text into AMR (Abstract Meaning Representation) structure and lacks the capability to train reasoning model end to end.

C. Generalization in Machine Reasoning

Despite gorgeous performance in field of Machine Reasoning, it is often questioned whether the neural model can predict examples out of training distribution correctly [7], [8]. Reference [38] proposes SCAN for evaluating systematic compositional skills of the model. In regard of exploring generalization ability of models, [39] examines systematic generalization in VQA domain, and [31] uses TPR-RNN model to improve generalization among different tasks. However, the above studies deal with generalization within seen entities, and our work mainly focuses on discussing and analyzing unseen filler generalization problem.

VI. Conclusions

We are the first work that points out the UFG problem by showing deeper analysis to demonstrate the overfitting phenomenon in modern attention-based reasoning models. We show that while models seemingly perform superbly in reasoning tasks, the attention mechanism in fact memorizes the name entities in training data, and can not generalize to new name entities. We expect a reasoning model should learn the hidden logic rules rather than overfitting to specific name entities. To evaluate filler generalization capability, we release two NLR datasets (UFG-bAbI, UFG-CLUTRR), expecting to facilitate this research direction. Futhermore, a one-shot training strategy, SER, is also proposed as a simple, general, yet promising solution to solve this task. Experiments show that it provides decent improvement as a preliminary strategy. The deeper insight within SER, and practical combination with other NLR datasets will be fascinating future work to focus on.

REFERENCES

[1] L. Bottou, "From machine learning to machine reasoning," *Machine learning*, vol. 94, no. 2, pp. 133–149, 2014.

[2] J. Weston, A. Bordes, S. Chopra, and T. Mikolov, "Towards ai-complete question answering: A set of prerequisite toy tasks," in *Proceedings of the 4th International Conference on Learning Representations*, 2016.

[3] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman, "Building machines that learn and think like people," *Behavioral and brain sciences*, vol. 40, 2017.

[4] J. Weston, S. Chopra, and A. Bordes, "Memory networks," in *International Conference on Learning Representations*, 2015.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[6] M. Dehghani, S. Gouws, O. Vinyals, J. Uszkoreit, and Ł. Kaiser, "Universal transformers," in *International Conference on Learning Representations*, 2019.

[7] A. Agrawal, D. Batra, and D. Parikh, "Analyzing the behavior of visual question answering models," in *Empirical Methods in Natural Language Processing*, 2016.

[8] R. Jia and P. Liang, "Adversarial examples for evaluating reading comprehension systems," in *Empirical Methods in Natural Language Processing*, 2017.

[9] P. Smolensky, "Tensor product variable binding and the representation of symbolic structures in connectionist systems," *Artif. Intell.*, vol. 46, no. 1-2, pp. 159–216, 1990.

[10] P. Smolensky and G. Legendre, *The harmonic mind: From neural computation to optimality-theoretic grammar (Cognitive architecture), Vol. 1*. MIT press, 2006.

[11] P. Smolensky, M. Goldrick, and D. Mathis, "Optimization and quantization in gradient symbol systems: A framework for integrating the continuous and the discrete in cognition," *Cognitive Science*, vol. 38, no. 6, pp. 1102–1138, 2014.

[12] N. Chomsky, *Knowledge of language: Its nature, origin, and use*. Greenwood Publishing Group, 1986.

[13] B. F. Skinner, *The behavior of organisms: an experimental analysis*. Appleton-Century, 1938.

[14] B. Skinner, *Verbal behavior*. New York: Appleton-Century-Crofts, 1957.

[15] N. Chomsky, *The architecture of language*. New Delhi: Oxford University Press, 2000.

[16] G. F. Marcus, *The birth of the mind: How a tiny number of genes creates the complexities of human thought*. Basic Civitas Books, 2004.

[17] P. Marler, "Innateness and the instinct to learn," *Anais da Academia Brasileira de Ciências*, vol. 76, no. 2, pp. 189–200, 2004.

[18] J. Pavez, H. Allende, and H. Allende-Cid, "Working memory networks: Augmenting memory networks with a relational reasoning module," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2018.

[19] M. Henaff, J. Weston, A. Szlam, A. Bordes, and Y. LeCun, "Tracking the world state with recurrent entity networks," in *International Conference on Learning Representations*, 2017.

[20] A. Graves, "Adaptive computation time for recurrent neural networks," *arXiv preprint arXiv:1603.08983*, 2016.

[21] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap, "A simple neural network module for relational reasoning," in *Advances in Neural Information Processing Systems*, 2017, pp. 4967–4976.

[22] K. Sinha, S. Sodhani, W. L. Hamilton, and J. Pineau, "Compositional language understanding with text-based relational reasoning," in *Advances in neural information processing systems - Relational Representation Learning Workshop*, 2018.

[23] K. Sinha, S. Sodhani, J. Dong, J. Pineau, and W. L. Hamilton, "Clutrr: A diagnostic benchmark for inductive reasoning from text," in *Empirical Methods in Natural Language Processing*, 2019.

[24] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing*, 2014, pp. 1532–1543.

[25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, 2018.

[26] J. Weston, "Dialog-based language learning," in *Advances in Neural Information Processing Systems*, 2016, pp. 829–837.

[27] A. Nematzadeh, K. Burns, E. Grant, A. Gopnik, and T. L. Griffiths, "Evaluating theory of mind in question answering," in *Empirical Methods in Natural Language Processing*, 2018, p. 2392–24007.

[28] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. L. Zitnick, and R. Girshick, "Clevr: A diagnostic dataset for compositional language and elementary visual reasoning," in *Computer Vision and Pattern Recognition*, 2017.

[29] A. Suhr, M. Lewis, J. Yeh, and Y. Artzi, "A corpus of natural language for visual reasoning," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2017, pp. 217–223.

[30] D. A. Hudson and C. D. Manning, "Gqa: A new dataset for real-world visual reasoning and compositional question answering," in *Computer Vision and Pattern Recognition*, 2019.

[31] I. Schlag and J. Schmidhuber, "Learning to reason with third order tensor products," in *Advances in Neural Information Processing Systems*, 2018, pp. 9981–9993.

[32] D. A. Hudson and C. D. Manning, "Compositional attention networks for machine reasoning," in *International Conference on Learning Representations*, 2018.

[33] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, "Weakly supervised memory networks," in *arXiv preprint arXiv:1503.08895*, vol. 412, 2015.

[34] S. Sukhbaatar, J. Weston, R. Fergus *et al.*, "End-to-end memory networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 2440–2448.

[35] A. Graves, G. Wayne, and I. Danihelka, "Neural turing machines," in *arXiv preprint arXiv:1410.5401*, vol. abs/1410.5401, 2014.

[36] A. Graves, G. Wayne, M. Reynolds, T. Harley, I. Danihelka, A. Grabska-Barwińska, S. G. Colmenarejo, E. Grefenstette, T. Ramalho, J. Agapiou *et al.*, "Hybrid computing using a neural network with dynamic external memory," *Nature*, vol. 538, no. 7626, p. 471, 2016.

[37] A. Mitra and C. Baral, "Addressing a question answering challenge by combining statistical methods with inductive rule learning and reasoning." in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[38] B. M. Lake and M. Baroni, "Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks," in *International Conference on Machine Learning*, 2017.

[39] D. Bahdanau, S. Murty, M. Noukhovitch, T. H. Nguyen, H. de Vries, and A. Courville, "Systematic generalization: What is required and can it be learned?" in *International Conference on Learning Representations*, 2019.