Reasoning in Dialog: Improving Response Generation by Context Reading Comprehension

Xiuying Chen[1,2], Zhi Cui[3], Jiayi Zhang[3], Chen Wei[3], Jianwei Cui[3], Bin Wang[3], Dongyan Zhao[1,2], and Rui Yan[4,5]*

[1]Wangxuan Institute of Computer Technology, Peking University,Beijing,China
[2]Center for Data Science, AAIS, Peking University,Beijing,China
[3]Xiaomi AI Lab
[4]Gaoling School of Artificial Intelligence, Renmin University of China
[5]Beijing Academy of Artificial Intelligence
xy-chen@pku.edu.cn

Abstract

In multi-turn dialog, utterances do not always take the full form of sentences (Carbonell 1983), which naturally makes understanding the dialog context more difficult. However, it is essential to fully grasp the dialog context to generate a reasonable response. Hence, in this paper, we propose to improve the response generation performance by examining the model's ability to answer a reading comprehension question, where the question is focused on the omitted information in the dialog. Enlightened by the multi-task learning scheme, we propose a joint framework that unifies these two tasks, sharing the same encoder to extract the common and task-invariant features with different decoders to learn task-specific features. To better fusing information from the question and the dialog history in the encoding part, we propose to augment the Transformer architecture with a memory updater, which is designed to selectively store and update the history dialog information so as to support downstream tasks. For the experiment, we employ human annotators to write and examine a large-scale dialog reading comprehension dataset. Extensive experiments are conducted on this dataset, and the results show that the proposed model brings substantial improvements over several strong baselines on both tasks. In this way, we demonstrate that reasoning can indeed help better response generation and vice versa. We release our large-scale dataset for further research[1].

Introduction

In recent years, text generation has made impressive progress (Chen et al. 2019; Li et al. 2020; Yu et al. 2020; Liu et al. 2020; Zhang et al. 2021), and open-domain dialogue generation has become a research hotspot in Natural Language Processing due to its broad application prospect, including chatbots, virtual personal assistants (Qiu et al. 2019; Debnath, Sengupta, and Wabgaonkar 2018; Li et al. 2019), etc. However, studies (Carbonell 1983) show that users of dialogue systems tend to use succinct language which often omits entities or concepts made in previous utterances. To make appropriate responses, dialogue systems must be equipped with the ability to understand these incomplete utterances. This naturally leads to the reading comprehension task, where correctly answering questions about the context requires understanding of natural language of the dialog context (Rajpurkar et al. 2016).

Take Example 2 in Table 1 for example, contents in parentheses are information omitted in the utterance. Humans are capable of comprehending such missing utterances dependent based on previous utterances and commonsense. For instance, A_3 means sending an *MV* to B instead of a *gift*. However, though of high importance, it is difficult for models to capture the implicit dependency between utterances without specific design, and that is why the reading comprehension task is proposed (Rajpurkar et al. 2016; Reddy, Chen, and Manning 2019). In this case, by reasoning and correctly answering the question with keyword "MV", the model learns that the dialog is focused on MV, which leads to a proper response that is also concentrated on music. Such cases that require dependency on the previous context to fully comprehend current utterance takes up about 60% according to a survey in Pan et al. (2019). This inspires us to come up with a multi-task framework that generates the response and answers reading comprehension question at the same time, which can boost the performance of each task.

Our *Multi-task Response Generator* (MRG) augments the previously proposed Transformer architecture (Vaswani et al. 2017) with the ability to encode multiple utterances in a question-aware fashion. The proposed model first uses a cross-attention mechanism (Vaswani et al. 2017) between the question and dialog words to identify representative words in dialog with the help of question. Concretely, we propose a memory updater, which updates its memory state using both the current inputs and previous memory state. The memory state can be interpreted as a container of the highly summarized dialog history information. During the cross-attention process, the current dialog representation is enhanced with the memory state from the previous step.

*Corresponding author: Rui Yan (email: ruiyan@ruc.edu.cn).

[1]https://github.com/yingtaomj/Reasoning-in-Dialog

	Example 1	Example 2	Example 3
A_1	求帮忙取名字姓程，俩男娃 Please help me decide how to name my two kids whose last name is Cheng	我最喜欢的歌手是MJ My favorite singer is MJ	那么我们即使不死,也在天堂 Then we are in heaven even if we don't die
B_1	程饭和程菜 Cheng fan and Cheng cai	你最喜欢他的什么歌呢? What's your favorite song?	这话哪抄的 Where did you copy that
A_2	哈哈哈哈哈 LOL	Thriller Thriller	三毛 Sanmao
B_2	请务必接受我的建议 Please accept my advice	我没听过呢，有这首歌的mv吗 I haven't heard of it. Is there an MV of this song?	想起以前豆瓣有个帅哥叫东门 Remember that there was a handsome man named Dongmen in Douban
A_3	咱俩一起生我就接受（取名程饭和程菜） I'll accept that (name as Cheng fan and Cheng cai) if they are our children	有(这首歌的MV)，我发给你看 Yes (I have the MV), I'll send it to you	那我（豆瓣）叫个南亭算了 Then my name (in Douban) will be Nanting.
Question	如果一起生娃那孩子叫什么 If the children are ours, how to call them?	准备发什么? What is going to be sent?	南亭是什么的ID? Nanting is ID of what
Answer	程饭和程菜 Cheng fan and Cheng cai	发thriller的的mv Thriller MV	豆瓣 Douban
B_3	我觉得这名字很好听啊 I think it's a nice name	好啊，我一直想看他的MV呢 Good, I've always wanted to see his MV	豆瓣就差你这个ID了 Douban is waiting for your ID
Reasoning Type	Paraphrasing (49.0%)	Lexical match (28.5%)	Pragmatics (22.5%)

Table 1: Examples from the dataset. Questions are concentrated on the omitted information of A_3 (which is shown in brackets), and reasoning type is the type of ability that is needed to answer the question.

MRG then uses a hierarchical inner attention, first over different words in each utterance, and then over all utterances in dialog history, to successively learn the utterance-level features. Finally, MRG utilizes the utterance-level and question features to select the answer to the question while generating the response words.

Since there lacks large-scale dialog reading comprehension datasets, we hire an annotation team to construct a dialog reading comprehension dataset (DRCD). Concretely, based on the *Restoration-200K* dataset proposed by Pan et al. (2019), where the omitted word span is annotated by humans, we ask the annotators to write question where the answer is the missing phrase. We manually construct 10k cases, based on which we train a question generator and leverage the model to construct questions for the rest of the dataset. We benchmark several classic dialog generation and reading comprehension baselines on DRCD. We also conduct experiments to show that the proposed model brings substantial improvements over these baselines on both tasks. In this way, we demonstrate that reasoning can indeed help better response generation and vice versa.

Our contributions can be summarized as follows:

- We propose the multi-task learning framework, which jointly answers reading comprehension questions and generates a proper response in multi-turn dialog scenario.
- We augment the Transformer architecture with a memory updater, which helps selectively store and update history dialog information.
- We release a large scale dialog reading comprehension dataset. Experimental results on this dataset demonstrate the effectiveness of our proposed framework.

Related Work

Multi-turn Dialog. In recent years, text generation has made impressive progress (Li et al. 2018; Chan et al. 2019; Gao et al. 2020b; Xie et al. 2020), and multi-turn dialog model aims to take a message and utterances in previous turns as input and generates a response (Tao et al. 2019; Gao et al. 2020a). Several works (Zhang et al. 2019; Adiwardana et al. 2020; Chan et al. 2020) simplify the multi-turn dialog into single-turn problem by simply concatenating multiple sentences into one sentence, and utilized the basic Seq2seq based on RNN or Transformer to model long sequence. To make better use of multi-turn utterances, Xing et al. (2017) apply hierarchical attention on word-level and utterance-level information. There also various dialog datasets (Lowe et al. 2015; Zhang et al. 2018; Welleck et al. 2018; Reddy, Chen, and Manning 2019). However, these datasets do not contain reading comprehension question-answering pairs.

Machine Reading Comprehension. Machine reading comprehension (MRC) focuses on modeling semantic matching between a question and a reference document, which read the full text to select relevant text spans and then infer answers. Choi et al. (2017) propose hierarchical coarse-to-fine methods in order to mimic the reading mode of human. Huang et al. (2017) come up with a fully-aware fusion attention mechanism and apply it on MRC tasks. Large-scale datasets for MRC have also been proposed in parallel. CommonsenseQA (Talmor et al. 2018) is a dataset for commonsense question answering extracted from CONCEPTNET (Speer, Chin, and Havasi 2016). DROP (Dua et al. 2019) and COSMOS (Huang et al. 2019) focus on

factual understanding and commonsense comprehension, respectively. In this paper, we propose another MRC dataset focused on machine comprehension on dialog corpus.

Multi-task Learning. Multi-task learning (MTL) is a learning paradigm in machine learning and it aims to leverage useful information contained in multiple related tasks to help improve the generalization performance of all the tasks(Caruana 1997). There are a large quantity of natural language processing tasks based on multi-task learning, such as word segmentation, POS tagging, dependency parsing, and text classification (Bohnet and Nivre 2012; Hatori et al. 2012; Li et al. 2013; Liu, Qiu, and Huang 2016). Collobert and Weston (2008) describe a single convolutional network that jointly trained several NLP tasks, such as part-of-speech tags, chunks, named entity tags, semantic roles. Liu et al. (2015) develop a multi-task deep neural network combining tasks of multiple-domain classification and information retrieval to learn representations across multiple tasks. In this work, we apply multi-task learning on response generation and reading comprehension on dialog.

Problem Formulation

Before presenting our approach for the dialog reading comprehension multi-task, we first introduce our notations and key concepts.

We assume that a conversation is conducted between two users. Suppose there are already N^u turns in a dialogue, so we have historical utterances as $X = (X_1, X_2, \ldots, X_{N^u})$, where each utterance X_j is depicted as $X_j = (x_1^j, x_2^j, \ldots, x_{N_j}^j)$ and x_i^j denotes a word. Accordingly, MRG aims to predict the (N^u+1)-th utterance ,*i.e.*, the response, $Y = (y_1, y_2, \ldots, y_{N^y})$, according to the historical utterances X:

$$p(Y|X) = \prod_{i=1}^{N^y} p\left(y_i|X, y_1, \ldots, y_{i-1}\right) \tag{1}$$

Apart from the reponse generation, we also design a question-answering task for the model. That is, targeted at the N^u-th utterance, where some keywords are missing, there is a question $Q = (q_1, q_2, ..., q_{N^q})$ that asks about such missing information, and the answer is a score vector $A = (a_1, a_2, ...a_{N^a})$ that extracts the missing keywords from previous utterances. $N^a = \sum_{i=1}^{N^u} N_i$. Each score $a_i \in \{0, 1\}$ denotes whether the i-th word is selected (1) or not (0). The objective is to maximize the likelihood of all word labels A given the input:

$$p\left(A|X\right) = \prod_{i=1}^{N^a} p\left(a_i|X\right) \tag{2}$$

The Proposed MRG Model

Overview

In this section, we propose the *Multi-task Response Generator*, abbreviated as MRG. An overview of MRG is shown in Figure 1, which can be split into three main parts:

• *Cross-hierarchical encoder* first uses a memory-augmented cross-attention mechanism (Vaswani et al. 2017) between the question and dialog words to identify representative words in dialog with the help of question. It then uses a hierarchical inner attention, first over different words in an utterance, and then over all utterances in dialog history, to successively learn the utterance-level features.

• *Answer selecter* takes the question representation and utterance-level dialog features as input to predict the answer.

• *Response generator* produces the response by attending to the utterance-level features.

Cross-hierarchical Encoder

To begin with, we use an embedding matrix e to map a one-hot representation of each word in X, Q, into a high-dimensional vector space. We then employ a bi-directional recurrent neural network (Bi-RNN) to model the temporal interactions between words:

$$\begin{aligned} h_i^{x,j} &= \text{Bi-RNN}_x\left(e(x_i^j), h_{i-1}^{x,j}\right), \\ h_i^q &= \text{Bi-RNN}_y\left(e(q_i), h_{i-1}^q\right), \end{aligned} \tag{3}$$

where $h_i^{x,j}$ and h_i^q denote the hidden state of i-th step in Bi-RNN for X_j and Q, respectively. Following (Zhao, Zhao, and Eskénazi 2017; Chen et al. 2018), we choose long short-term memory (LSTM) as the cell for Bi-RNN.

Memory-augmented Cross Attention. This module grounds the conversation context by the question and fuses the information of the question into the dialog representation. Concretely, it has a stack of L identical layers. In each layer, we iteratively fuse the information from question words to the dialog words by Memory-augmented Cross Attention Module (MCAM). For convenience, we denote the output of l-th encoder layer as $m_i^{l,j}$ and the input for the first layer $m_i^{0,j}$ is initialized as $h_i^{x,j}$. Concretely, MCAM is based on the traditional Cross Attention Module (CAM) Transformer architecture (Vaswani et al. 2017). We first introduce the original CAM, and then introduce our modification.

The first input for CAM is for query Q and the second input is for keys K and values V for attention, which we denote as x_i and h_*^q respectively:

$$m_i^{l,j} = \text{CAM}(m_i^{l-1,j}, h_*^q). \tag{4}$$

Each output element, $m_i^{l,j}$, is computed as weighted sum of a linearly transformed input values:

$$m_i^{l,j} = \sum_{k=1}^{N_j} \alpha_{i,k}^{l,j}\left(h_k^q W^V\right). \tag{5}$$

Each weight coefficient, $\alpha_{i,k}^{l,j}$, is computed using a softmax function:

$$\alpha_{i,k}^{l,j} = \frac{\exp\left(\beta_{i,k}^{l,j}\right)}{\sum_{k=1}^{N_j}\exp\left(\beta_{i,k}^{l,j}\right)}. \tag{6}$$

And $\beta_{i,k}^{l,j}$ is computed using a compatibility function that compares two input elements:

$$\beta_{i,k}^{l,j} = \frac{\left(m_i^{l-1,j}W^Q\right)\left(h_k^q W^K\right)^T}{\sqrt{d}}, \tag{7}$$

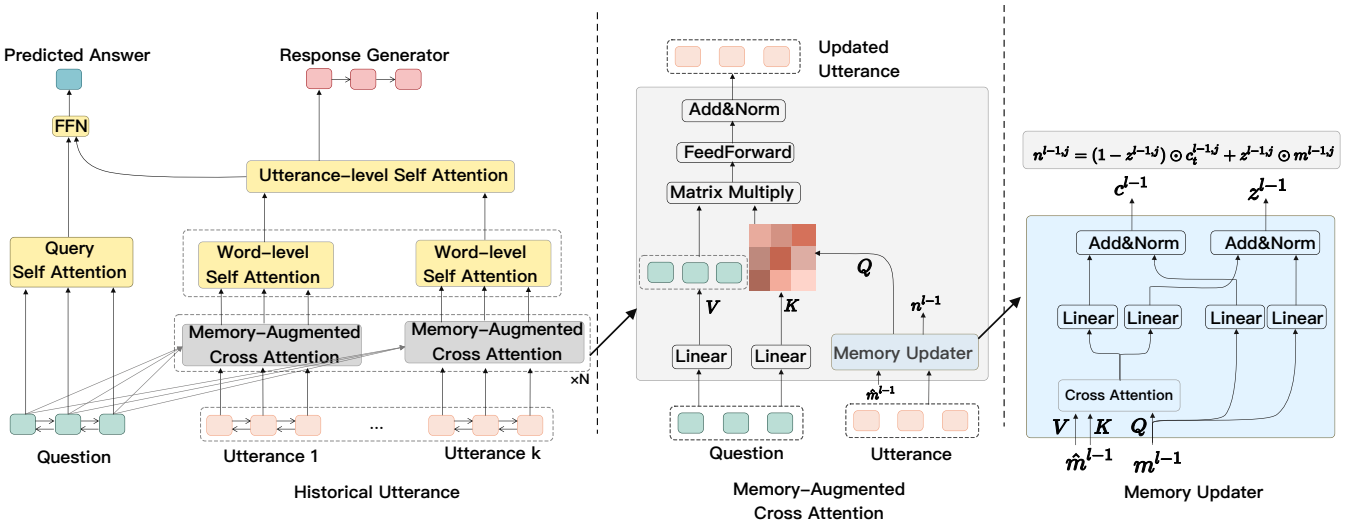

Figure 1: Overview of MRG. We divide our model into three parts: (1) Cross-hierarchical Encoder (which consists of memory-augment cross attention and two hierarchical self attentions); (2) Answer Selecter; (3) Response Generator.

where d stands for hidden dimension. $W^Q, W^K, W^V \in \mathbb{R}^{N_j \times N_j}$ are parameter matrices.

While the aforementioned vanilla CAM is a powerful method, it is less suitable for multi-turn dialog due to its inability to fully utilize dialog history information. Thus, we augment it with an external memory module, which helps to remember and update history dialog information in a multi-slot way as illustrated in Figure 1. The input for query, *i.e.*, $m_i^{l-1,j}$ is updated to $n_i^{l-1,j}$ through a memory updator, which will then be fed into CAM in Equation 4. Concretely, the memory updator aggregates the information from both its intermediate hidden states $\hat{m}_i^{l-1}$ ($\hat{m}^{l-1} \in \mathbb{R}^{N_j \times d}$) and the utterance (memory) states $m_i^{l-1,j}$ from the last layer, using a multi-head attention. Specifically, the input for query Q is $m_i^{l-1,j}$, and input for key K and value V is $[\hat{m}_i^{l-1}; m_i^{l-1,j}]$. The memory augmented hidden states are further encoded using a feed-forward layer and then merged with the intermediate hidden states $\hat{m}^{l-1}$ using a residual connection and layer norm. We summarize the procedure below:

$$
\begin{aligned}
s_i^{l-1,j} &= \text{CAM}(m_i^{l-1,j}, \hat{m}_i^{l-1}), \\
c_i^{l-1,j} &= \tanh(W_a^{l-1} m_i^{l-1,j} + W_b^{l-1} s_i^{l-1,j}), \\
z_i^{l-1} &= \text{sigmoid}(W_c^{l-1,j} m_i^{l-1,j} + W_d^{l-1} s_i^{l-1}), \\
n_i^{l-1,j} &= (1 - z_i^{l-1,j}) \odot c_t^{l-1,j} + z_i^{l-1,j} \odot m_i^{l-1,j},
\end{aligned}
$$

where $\odot$ denotes Hadamard product, W_a^{l-1}, W_b^{l-1}, W_c^{l-1}, and W_d^{l-1} are trainable weights, c_i^{l-1} is the internal cell state. z_i^{l-1} is the update gate that controls which information to retain from the previous memory state.

Hierarchical Self Attention. After utilizing question information to emphasize important keywords, $m_i^{L,j}$ (the output of last MCAM layer) is then processed by a hierarchical attentive module to encode long-term dependency among words into the representations. The first level in our hierarchical attention encodes each utterance independently from other utterances at word-level, resulting in a fixed-dimensional representation of each utterance. Concretely, the word-level attentive module simplifies the Multi-head Attention Module (MAM) in Transformer, which is similar to CAM, but takes the same input for query, key and value:

$$
h_i^{w,j} = \text{MAM}(m_i^{L,j}, m_*^{L,j}). \tag{8}
$$

A mean-pooling operation is then used over word vectors in each utterance to obtain a fixed-length utterance-level representation:

$$
h^{u',j} = \text{meanpool}\left(\left\{h_1^{w,j}, \cdots, h_{N_j}^{w,j}\right\}\right). \tag{9}
$$

Similar to word-level attention, an utterance-level MAM is applied on these representations to fuse information between different utterances:

$$
h^{u,j} = \text{MAM}(h^{u',j}, h^{u',*}). \tag{10}
$$

From the utterance representation, we can also obtain the overall dialog history representation, which will be used in the response decoder part:

$$
h^d = \text{meanpool}\left(\left\{h^{u,1}, \cdots, h^{u,N^u}\right\}\right). \tag{11}
$$

Answer Selector

After fusing information from question and dialog context, it is time to select words from context as the answer to the question. Since we have several utterance representations, and either taking the average or summing them together by specific weights is inappropriate and inelegant. Hence, we concatenate all utterance and question representations together and apply a multi-layer perceptron to them to generate the word extracting probabilities:

$$h^q = \text{meanpool}\left(\{h_1^q, \cdots, h_{N^q}^q\}\right),$$
$$\hat{A} = W_f \tanh\left(W_e\left[h^{u,1}; \ldots; h^{u,N^u}; h^q\right] + b^e\right) + b^f,$$

where $[;]$ denotes concatenation operation.

Response Generator

To generate a consistent and informative response, we propose an RNN-based decoder that incorporates outputs of utterance representations as illustrated in Figure 1.

We first apply a linear transform layer on the input document vector representation h^d and use the output of this layer as the initial state of decoder LSTM, shown in Equation 12. In order to reduce the burden of compressing document information into the initial state s_0, we use the attention mechanism (Bahdanau, Cho, and Bengio 2015a) to summarize the utterance representations into context vector f_{t-1} dynamically and we will show the detail of these in this section later. We then concatenate the context vector f_{t-1} with the embedding of previous step output $e(y_{t-1})$ and feed this into decoder LSTM, shown in Equation 13:

$$s_0 = W_g h^d + b_g, \tag{12}$$
$$s_t = \text{LSTM}\left(s_{t-1}, [f_{t-1}; e(y_{t-1})]\right). \tag{13}$$

Context vector f_{t-1} is the vector that stores the dialog context information at t-th step. Concretely, we use the decoder state s_{t-1} to attend to each utterance states $h^{u,i}$ and results in the attention distribution γ_t, shown in Equation 15. Then we use the attention distribution γ_t to weighted sum the document states as the context vector f_{t-1}.

$$\gamma'_{t-1,i} = W_n^\intercal \tanh\left(W_s s_{t-1} + W_h h^{u,i}\right), \tag{14}$$
$$\gamma_{t-1,i} = \exp\left(\gamma'_{t-1,i}\right) / \sum_{j=1}^{N_u} \exp\left(\gamma'_{t-1,j}\right), \tag{15}$$
$$f_{t-1} = \sum_{i=1}^{N_u} \gamma_{t-1,i} h^{u,i}. \tag{16}$$

Finally, an output projection layer is applied to get the final generating distribution P_t^v over vocabulary, as shown in Equation 17. We concatenate utterance context vector and the output of decoder LSTM s_t as the input of the output projection layer:

$$P_t^v = \text{softmax}\left(W_v[s_t; f_t] + b_v\right), \tag{17}$$

We use the negative log-likelihood as the loss function:

$$\mathcal{L}_g = -\sum_{t=1}^{N^y} \log P_t^v(y_t). \tag{18}$$

Experimental Setup

Dataset

To our best knowledge, no existing works consider MRC in response generation task. Hence, we first propose a dialog reading comprehension dataset (DRCD). DRCD is based on the *Restoration-200k* dataset proposed by Pan et al. (2019), where the utterance with omitted information is manually annotated. Such omitted information leads to a difficulty in fully understanding the dialog context and requires reasoning ability to for a model. Hence, we hire an annotation team to write questions that are focused on the missing information.

Since it is time-consuming to write questions for the whole dataset, and based on the labeled answer it is rather easy to construct the question, we ask the team to write questions for 10k cases, and then automatically generate questions for the rest of the dataset. Concretely, we utilize PG (See, Liu, and Manning 2017) to generate questions due to its good performance in many tasks including summarization and dialog completion (Pan et al. 2019; Chen et al. 2019). We then conduct a human evaluation to examine the generation quality. Concretely, we randomly sample 200 cases and asked three annotators to state how well they agree with the following two statements, on a scale of one to five (strongly disagree, disagree, neutral, agree, or strongly agree): 1) The generated question asks about the omitted phrase. 2) The generated question is written in fluent Chinese. The result shows that generated questions that score over 3 takes up 76.5%, showing that most of the generated questions are of good quality. The kappa statistics indicate the moderate agreement between annotators.

We randomly split the dataset with question-answer pair to 113,116 training, 3,000 validation, and 3,000 test cases. The average character-level context length and utterance length of the dataset is and 43.4 and 9.05. Note that in the validation and test datasets the questions are all written by human, ensuring that the testing results are convincing.

Comparison Methods

To evaluate the performance of our proposed model, we compare it with the following response generation and MRC baselines:

Seq2Seq (Bahdanau, Cho, and Bengio 2015b): the vanilla schema of the sequence to sequence model with attention mechanism.

HRED (Serban et al. 2016): extends the hierarchical recurrent encoder-decoder neural network to the dialogue domain.

VAE (Zhao, Zhao, and Eskénazi 2017): uses latent variables to learn a distribution over potential conversational intents and generates diverse responses.

Transformer (Vaswani et al. 2017): is based solely on attention mechanisms.

PAC (Pan et al. 2019): is a "pick-and-combine" model to restore the incomplete utterance from its context, and then use the restored utterance to generate the next response.

MemN2N (Sukhbaatar et al. 2015): is an extension of RNNsearch to the case with multiple computational hops.

DMN (Kumar et al. 2016): processes input sequences and questions, forms episodic memories, and generates relevant answers.

DMN+ (Xiong, Merity, and Socher 2016): proposes several improvements to memory and input modules of DMN.

QRN (Seo et al. 2017): is a variant of RNN that effectively handles both short-term (local) and long-term (global) sequential dependencies to reason over multiple facts.

Model	BLEU1	BLEU2	BLEU3	BLEU4	Average	Extrema	Greedy
Seq2Seq	0.2260	0.1566	0.0876	0.0671	0.4341	0.6695	0.7759
HRED	0.2273	0.1559	0.0871	0.0667	0.4320	0.6601	0.7885
VAE	0.2316	0.1586	0.0886	0.0680	0.4350	0.6396	0.7808
Transfomer	0.2181	0.1482	0.0825	0.0631	0.4407	0.6500	0.7920
PAC	0.2413	0.1624	0.0902	0.0689	0.4396	0.6447	0.7909
MRG	**0.2632**	**0.1735**	**0.0968**	0.0741	**0.4513**	**0.6769**	**0.8025**
MRG w/o MCAM	0.2224	0.1533	0.0857	0.0656	0.4436	0.6630	0.7837
MRG w/o MAM	0.2404	0.1616	0.0946	0.0665	0.4343	0.6740	0.7798
MRG w/o MemUpd	0.2498	0.1585	0.0894	**0.0747**	0.4419	0.6551	0.7884
MRG w/o MT	0.2231	0.1541	0.0862	0.0661	0.4343	0.6734	0.7645

Table 2: Automatic evaluation results on response generation task. The best results are bold.

Model	Accuracy(%)	
	Mean	Best
MemN2N	37.85	38.22
DMN	40.83	42.21
QRN	40.80	43.71
DMN+	43.97	45.02
MRG	**45.43**	**47.17**
MRG w/o MT	44.89	46.34

Table 3: Automatic evaluation results on MRC task. Best accuracy over 10 runs.

Implementation Details

We implement our experiments in TensorFlow (Abadi et al. 2016) on an NVIDIA GTX 1080 Ti GPU. We truncate input dialog to 100 words, with 20 words in each utterance. We chose 100 as our truncation size as we did not find significant improvement when increasing input length from 100 to 200 tokens. The minimum decoding step is 10, and the maximum step is 20. The word embedding dimension is set to 128 and the number of hidden units is 256. We initialize all of the parameters randomly using a Gaussian distribution. The batch size is set to 16, and we limit the vocabulary size to 50K. We use Adagrad optimizer (Duchi, Hazan, and Singer 2010) as our optimizing algorithm. We also apply gradient clipping (Pascanu, Mikolov, and Bengio 2013) with a range of $[-2, 2]$ during training. During the inference stage, the checkpoint with smallest validation loss is chosen and the beam-search size is set to 4 for all methods. Note that when evaluating the response generation performance, we use the generated questions as input instead of the ground truth human-written questions for the sake of fairness.

Evaluation Metrics

To evaluate the results of the generated responses, we adopt the following metrics widely used in existing research.

Overlap-based Metric. Following Li et al. (2021); Xu et al. (2020), we utilize BLEU score (Papineni et al. 2002), an algorithm which has been widely used in machine translation and dialogue system, to measure n-grams overlaps between ground-truth and generated response. Specifically, we follow the conventional setting in previous work (Gu et al. 2019) to compute BLEU scores using smoothing techniques (smoothing 7).

Embedding Metrics. To capture the semantic matching degrees between generated responses and ground-truth, we perform evaluations on embedding space. In consistent with previous study (Gu et al. 2019), we compute the similarity between the bag-of-words (BOW) embeddings representations of generated results and reference. In particular, we calculate three metrics:1) *Greedy* (BOW-Greedy), i.e., greedily matching words in two utterances based on the cosine similarities; 2) *Average* (BOW-Average), cosine similarity between the averaged word embeddings in the two utterances (Mitchell and Lapata 2008); 3) *Extrema* (BOW-Extrema), cosine similarity between the largest extreme values among the word embeddings in the two utterances (Forgues et al. 2014).

Human Metrics. We also employ human evaluation to assess the responses generated by our model and the baselines. Three well-educated annotators are hired to evaluate the quality of generated responses, where the evaluation is conducted in a double-blind fashion. Totally, 200 randomly sampled responses generated by each model are rated by each annotator with two different aspects, *i.e.,* readability (Is the response grammatically formed and smooth?), informativeness (Does the response contains informative words?). Criteria are scored from 1 to 3, *i.e.,* bad, normal, and good.

Experimental Results

Overall Performance

Automatic evaluation. We first examine whether our MRG outperforms generative baselines as listed in Table 2. Our model outperforms baselines on all automatic metrics. This demonstrates that our model generates more appropriate responses by reading comprehension, and understands the dialog context better by predicting response. Especially, our model improves approximately 16.46% over seq2seq on BLEU1, and outperforms PAC by 9.07%. We also list the results of ablation study in Table 2, aiming to assess the contribution of individual components. Our experiments confirmed that interacting between dialog and question by Memory-augmented Cross Attention Module is beneficial (see row w/o MCAM), as well as self-attention module (see row w/o MAM) memory updator (see row w/o MemUpd).

	Example 1	Example 2	Example 3
A_1	有木有人带妹子吃喝玩乐在杭州 Is there anyone to take girls to eat, drink and have fun in Hangzhou	天蝎座不自恋真的就能死 Scorpio will die without narcissism	lz女坐标杭州心情不nice The author is a girl, located in Hangzhou, has a bad mood
B_1	没真相不敢带 I dare not bring a girl without a photo	挺准最近就是被一个天蝎虐 That's right. I've been abused by a Scorpio recently.	那怎么办 Then what to do
A_2	有了真相更不敢带哈 With a photo, you will dare not bring her more	嗯 Yes	睡觉么 What about sleep
B_2	犀利 Sharp	你是摩羯啊 So you are a Capricorn	也可以不过不是长久之计 That's good, but not a long-term solution
A_3	一般(犀利)啦 Generally (sharp)	(摩羯被天蝎虐)这就是宿命 This (Capricorn abused by Scorpio) is fate.	哈哈那(心情不nice)怎么办 Ha-ha, then what to do (if in bad mood)
Question	什么很一般 What is general	什么是宿命 What is fate	什么出问题了 What's wrong
Answer	犀利程度一般 The degree of sharpness	摩羯被天蝎虐 Capricorn abused by Scorpio	心情 Mood
Reference	一般都这么犀利男人咋办 If that is general sharp then what to do with man	所以我喜欢上天蝎就是个错误 So it was a mistake for me to like Scorpio	找个心灵的驿站 Find a post station for the soul
HRED	是你的错觉 It's your delusion	洗洗睡了 Wash and sleep	生命健康更加可贵 Life and health are more valuable
VAE	好吧,改天可以一起去嗨皮了 Well, we can play the other day	我帮同事发帖子勾搭上了 I helped my colleagues post and hook up	年龄大了就想睡 When you get older, you want to sleep
PAC	你也要去对不啦 So you will go there, aren't you?	那你还不如玩手机 You might as well play with your cell phone	我闲了就读书 I read when I'm free
MRG	是过于犀利啦 It's too sharp	还是要看人，怎么就宿命了哈哈哈 It still depends on people. You should not believe in fate lol.	出门旅游，逛逛豆瓣 Travel outside, visit Douban

Table 4: Responses generated by baselines and our model along with the QA pairs.

We next examine whether our MRG outperforms MRC baselines in Table 3. Generally, these baselines perform similar to the experiment on bAbI dataset (Bordes and Weston 2017). Specifically, DMN+ is the strongest baseline, which achieves 43.97% accuracy on average. QRN, however, does not perform as well as it does on bAbI dataset, obtaining lower accuracy than DMN+. Our model obtains highest mean accuracy and best accuracy over 10 runs among all baselins.

Human evaluation. The results of human evaluation are listed in Table 5. Our model significantly outperforms most of the baselines in terms of all the metrics. Particularly, our model increases informativeness approximately 4.1% over PAC. This demonstrates that trying to answer reading comprehension question about dialog history if beneficial for improving and enriching the responses.

Analysis of Multi-task Learning

Our model aims to generate response as well as answering MRC questions, which can be regarded as a multi-task. Hence, in this subsection, we examine whether these two tasks can complement each other. We list the performance on two single tasks by 'MRG w/o MT' in Table 2 and Table 3, which solely generates response and answers MRC question, respectively. It can be seen that by answering reading comprehension question, the performance of dialog generation increases by 12.1% in terms of BLEU4 score, and by generating responses at the same time, MRC accuracy increases by 1.2%.

Model	Readability	Informativeness
HRED	1.43	1.46
VAE	1.60	1.58
PAC	1.56	1.72
MRG	**1.63**	**1.75**

Table 5: Human evaluation on two aspects: Readability and informativeness.

Conclusion

In this paper we propose the multi-task framework to generate response and answer reading comprehension questions about multi-turn dialog. Concretely, the two tasks share the same encoder to extract the common and task-invariant features with different decoders to learn specific features. To better fusing information from the question and the dialog history in the encoding part, we propose to augment the Transformer architecture with a memory updater, which is designed to selectively store and update the history dialog information. Experimental results show that our proposed model outperforms classic baselines. In the future we would like to apply our model to other multi-task scenarios.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (No. 2020YFB1406702), the National Science Foundation of China (NSFC No. 61876196 and No. 61672058), Beijing Outstanding Young Scientist Program No. BJJWZYJH012019100020098. Rui Yan is supported as a young fellow at Beijing Academy of Artificial Intelligence (BAAI).

References

Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; Kudlur, M.; Levenberg, J.; Monga, R.; Moore, S.; Murray, D. G.; Steiner, B.; Tucker, P. A.; Vasudevan, V.; Warden, P.; Wicke, M.; Yu, Y.; and Zhang, X. 2016. TensorFlow: A System for Large-Scale Machine Learning. In *OSDI*.

Adiwardana, D.; Luong, M.-T.; So, D. R.; Hall, J.; Fiedel, N.; Thoppilan, R.; Yang, Z.; Kulshreshtha, A.; Nemade, G.; Lu, Y.; et al. 2020. Towards a human-like open-domain chatbot. *arXiv preprint arXiv:2001.09977* .

Bahdanau, D.; Cho, K.; and Bengio, Y. 2015a. Neural Machine Translation by Jointly Learning to Align and Translate. In *ICLR*.

Bahdanau, D.; Cho, K.; and Bengio, Y. 2015b. Neural Machine Translation by Jointly Learning to Align and Translate. *ICLR* .

Bohnet, B.; and Nivre, J. 2012. A transition-based system for joint part-of-speech tagging and labeled non-projective dependency parsing. In *EMNLP*, 1455–1465.

Bordes, A.; and Weston, J. 2017. Learning End-to-End Goal-Oriented Dialog. *ICLR* .

Carbonell, J. 1983. Discourse Pragmatics and Ellipsis Resolution in Task-Oriented Natural Language Interfaces. In *ACL*.

Caruana, R. 1997. Multitask learning. *Machine learning* 28(1): 41–75.

Chan, Z.; Chen, X.; Wang, Y.; Li, J.; Zhang, Z.; Gai, K.; Zhao, D.; and Yan, R. 2019. Stick to the Facts: Learning towards a Fidelity-oriented E-Commerce Product Description Generation. In *EMNLP*, 4960–4969.

Chan, Z.; Zhang, Y.; Chen, X.; Gao, S.; Zhang, Z.; Zhao, D.; and Yan, R. 2020. Selection and Generation: Learning towards Multi-Product Advertisement Post Generation. In *EMNLP*, 3818–3829.

Chen, X.; Chan, Z.; Gao, S.; Yu, M.-H.; Zhao, D.; and Yan, R. 2019. Learning towards Abstractive Timeline Summarization. In *IJCAI*, 4939–4945.

Chen, X.; Gao, S.; Tao, C.; Song, Y.; Zhao, D.; and Yan, R. 2018. Iterative Document Representation Learning Towards Summarization with Polishing. *ArXiv* abs/1809.10324.

Choi, E.; Hewlett, D.; Uszkoreit, J.; Polosukhin, I.; Lacoste, A.; and Berant, J. 2017. Coarse-to-fine question answering for long documents. In *ACL*, 209–220.

Collobert, R.; and Weston, J. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *ICML*, 160–167.

Debnath, P.; Sengupta, S.; and Wabgaonkar, H. M. 2018. Identifying , Classifying and Resolving Non-Sentential Utterances in Customer Support Systems.

Dua, D.; Wang, Y.; Dasigi, P.; Stanovsky, G.; Singh, S.; and Gardner, M. 2019. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. *arXiv preprint arXiv:1903.00161* .

Duchi, J. C.; Hazan, E.; and Singer, Y. 2010. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *J. Mach. Learn. Res.* 12: 2121–2159.

Forgues, G.; Pineau, J.; Larchevêque, J.-M.; and Tremblay, R. 2014. Bootstrapping dialog systems with word embeddings. In *Nips, workshop*, volume 2.

Gao, S.; Chen, X.; Liu, C.; Liu, L.; Zhao, D.; and Yan, R. 2020a. Learning to Respond with Stickers: A Framework of Unifying Multi-Modality in Multi-Turn Dialog. In *Proceedings of The Web Conference 2020*, 1138–1148.

Gao, S.; Chen, X.; Ren, Z.; Zhao, D.; and Yan, R. 2020b. Meaningful Answer Generation of E-Commerce Question-Answering. *ACM Trans. Inf. Syst.* .

Gu, X.; Cho, K.; Ha, J.-W.; and Kim, S. 2019. DialogWAE: Multimodal Response Generation with Conditional Wasserstein Auto-Encoder. In *International Conference on Learning Representations*. URL https://openreview.net/forum?id=BkgBvsC9FQ.

Hatori, J.; Matsuzaki, T.; Miyao, Y.; and Tsujii, J. 2012. Incremental joint approach to word segmentation, POS tagging, and dependency parsing in Chinese. In *ACL*, 1045–1053.

Huang, H.-Y.; Zhu, C.; Shen, Y.; and Chen, W. 2017. Fusionnet: Fusing via fully-aware attention with application to machine comprehension. *preprint arXiv:1711.07341* .

Huang, L.; Bras, R. L.; Bhagavatula, C.; and Choi, Y. 2019. Cosmos qa: Machine reading comprehension with contextual commonsense reasoning. *preprint arXiv:1909.00277* .

Kumar, A.; Irsoy, O.; Ondruska, P.; Iyyer, M.; Bradbury, J.; Gulrajani, I.; Zhong, V.; Paulus, R.; and Socher, R. 2016. Ask Me Anything: Dynamic Memory Networks for Natural Language Processing. In *ICML*.

Li, J.; Qiu, L.; Tang, B.; Chen, D.; Zhao, D.; and Yan, R. 2019. Insufficient data can also rock! learning to converse using smaller data with augmentation. In *AAAI*, volume 33, 6698–6705.

Li, J.; Song, Y.; Zhang, H.; Chen, D.; Shi, S.; Zhao, D.; and Yan, R. 2018. Generating classical chinese poems via conditional variational autoencoder and adversarial training. In *EMNLP*, 3890–3900.

Li, M.; Chen, X.; Gao, S.; Chan, Z.; Zhao, D.; and Yan, R. 2020. VMSMO: Learning to Generate Multimodal Summary for Video-based News Articles. In *EMNLP*, 9360–9369.

Li, M.; Chen, X.; Yang, M.; Gao, S.; Zhao, D.; and Yan, R. 2021. The Style-Content Duality of Attractiveness: Learning to Write Eye-Catching Headlines via Disentanglement. In *AAAI*.

Li, Z.; Zhang, M.; Che, W.; Liu, T.; and Chen, W. 2013. Joint optimization for Chinese pos tagging and dependency parsing. *IEEE/ACM transactions on audio, speech, and language processing* 22(1): 274–286.

Liu, D.; Li, J.; Yu, M.-H.; Huang, Z.; Liu, G.; Zhao, D.; and Yan, R. 2020. A Character-Centric Neural Model for Automated Story Generation. In *AAAI*, 1725–1732.

Liu, P.; Qiu, X.; and Huang, X. 2016. Recurrent neural network for text classification with multi-task learning. *arXiv preprint arXiv:1605.05101* .

Liu, X.; Gao, J.; He, X.; Deng, L.; Duh, K.; and Wang, Y.-y. 2015. Representation Learning Using Multi-Task Deep Neural Networks for Semantic Classification and Information Retrieval. In *NAACL*.

Lowe, R.; Pow, N.; Serban, I.; and Pineau, J. 2015. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. *arXiv preprint arXiv:1506.08909* .

Mitchell, J.; and Lapata, M. 2008. Vector-based models of semantic composition. *NAACL-HLT* 236–244.

Pan, Z.; Wang, Y.; Bai, K.; Zhou, L.; and Liu, X. 2019. Improving Open-Domain Dialogue Systems via Multi-Turn Incomplete Utterance Restoration. In *EMNLP*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL*, 311–318. ACL.

Pascanu, R.; Mikolov, T.; and Bengio, Y. 2013. On the difficulty of training recurrent neural networks. In *ICML*.

Qiu, L.; Li, J.; Bi, W.; Zhao, D.; and Yan, R. 2019. Are training samples correlated? learning to generate dialogue responses with multiple references. In *ACL*, 3826–3835.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. SQuAD: 100, 000+ Questions for Machine Comprehension of Text. *EMNLP* .

Reddy, S.; Chen, D.; and Manning, C. D. 2019. Coqa: A conversational question answering challenge. *TACL* 7: 249–266.

See, A.; Liu, P. J.; and Manning, C. D. 2017. Get To The Point: Summarization with Pointer-Generator Networks. In *ACL*.

Seo, M.; Min, S.; Farhadi, A.; and Hajishirzi, H. 2017. Query-Reduction Networks for Question Answering. *arXiv: Computation and Language* .

Serban, I.; Sordoni, A.; Bengio, Y.; Courville, A. C.; and Pineau, J. 2016. Building End-To-End Dialogue Systems Using Generative Hierarchical Neural Network Models. In *AAAI*.

Speer, R.; Chin, J.; and Havasi, C. 2016. Conceptnet 5.5: An open multilingual graph of general knowledge. *arXiv preprint arXiv:1612.03975* .

Sukhbaatar, S.; Szlam, A.; Weston, J.; and Fergus, R. 2015. End-To-End Memory Networks. In *NIPS*.

Talmor, A.; Herzig, J.; Lourie, N.; and Berant, J. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *preprint arXiv:1811.00937* .

Tao, C.; Wu, W.; Xu, C.; Hu, W.; Zhao, D.; and Yan, R. 2019. One time of interaction may not be enough: Go deep with an interaction-over-interaction network for response selection in dialogues. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1–11.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NIPS*, 5998–6008.

Welleck, S.; Weston, J.; Szlam, A.; and Cho, K. 2018. Dialogue natural language inference. *arXiv preprint arXiv:1811.00671* .

Xie, P.; Cui, Z.; Chen, X.; Hu, X.; Cui, J.; and Wang, B. 2020. Infusing Sequential Information into Conditional Masked Translation Model with Self-Review Mechanism. In *COLING*, 15–25.

Xing, C.; Wu, W.; Wu, Y.; Zhou, M.; Huang, Y.; and Ma, W.-Y. 2017. Hierarchical recurrent attention network for response generation. *arXiv preprint arXiv:1701.07149* .

Xiong, C.; Merity, S.; and Socher, R. 2016. Dynamic Memory Networks for Visual and Textual Question Answering. In *ICML*.

Xu, R.; Tao, C.; Jiang, D.; Zhao, X.; Zhao, D.; and Yan, R. 2020. Learning an Effective Context-Response Matching Model with Self-Supervised Tasks for Retrieval-based Dialogues. In *AAAI*.

Yu, M.-H.; Li, J.; Liu, D.; Zhao, D.; Yan, R.; Tang, B.; and Zhang, H. 2020. Draft and edit: Automatic storytelling through multi-pass hierarchical conditional variational autoencoder. In *AAAI*, volume 34, 1741–1748.

Zhang, J.; Cui, Z.; Xia, X.; Guo, Y.; Li, Y.; Wei, C.; and Cui, J. 2021. Writing Polishment with Simile: Task, Dataset and A Neural Approach. In *AAAI*.

Zhang, S.; Dinan, E.; Urbanek, J.; Szlam, A.; Kiela, D.; and Weston, J. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? *preprint arXiv:1801.07243* .

Zhang, Y.; Sun, S.; Galley, M.; Chen, Y.-C.; Brockett, C.; Gao, X.; Gao, J.; Liu, J.; and Dolan, B. 2019. Dialogpt: Large-scale generative pre-training for conversational response generation. *arXiv preprint arXiv:1911.00536* .

Zhao, T.; Zhao, R.; and Eskénazi, M. 2017. Learning Discourse-level Diversity for Neural Dialog Models using Conditional Variational Autoencoders. In *ACL*.