

# Smoothing Dialogue States for Open Conversational Machine Reading

Zhuosheng Zhang<sup>1,2,3,#</sup>, Siru Ouyang<sup>1,2,3,#</sup>, Hai Zhao<sup>1,2,3,\*</sup>, Masao Utiyama<sup>4</sup>, Eiichiro Sumita<sup>4</sup>

<sup>1</sup> Department of Computer Science and Engineering, Shanghai Jiao Tong University

<sup>2</sup> Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering, Shanghai Jiao Tong University

<sup>3</sup> MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University

<sup>4</sup> National Institute of Information and Communications Technology (NICT), Kyoto, Japan

{zhangzs, oysr0926}@sjtu.edu.cn, zhaohai@cs.sjtu.edu.cn

{mutiyama, eiichiro.sumita}@nict.go.jp

## Abstract

Conversational machine reading (CMR) requires machines to communicate with humans through multi-turn interactions between two salient dialogue states of decision making and question generation processes. In open CMR settings, as the more realistic scenario, the retrieved background knowledge would be noisy, which results in severe challenges in the information transmission. Existing studies commonly train independent or pipeline systems for the two subtasks. However, those methods are trivial by using hard-label decisions to activate question generation, which eventually hinders the model performance. In this work, we propose an effective gating strategy by smoothing the two dialogue states in only one decoder and bridge decision making and question generation to provide a richer dialogue state reference. Experiments on the OR-ShARC dataset show the effectiveness of our method, which achieves new state-of-the-art results.

## 1 Introduction

The ultimate goal of multi-turn dialogue is to enable the machine to interact with human beings and solve practical problems (Zhu et al., 2018; Zhang et al., 2018; Zaib et al., 2020; Huang et al., 2020; Fan et al., 2020; Gu et al., 2021). It usually adopts the form of question answering (QA) according to the user’s query along with the dialogue context (Sun et al., 2019; Reddy et al., 2019; Choi et al., 2018). The machine may also actively ask questions for confirmation (Wu et al., 2018; Cai et al., 2019; Zhang et al., 2020b; Gu et al., 2020).

In the classic spoken language understanding tasks (Tur and De Mori, 2011; Zhang et al., 2020a; Ren et al., 2018; Qin et al., 2021), specific slots and intentions are usually defined. According to these predefined patterns, the machine interacts

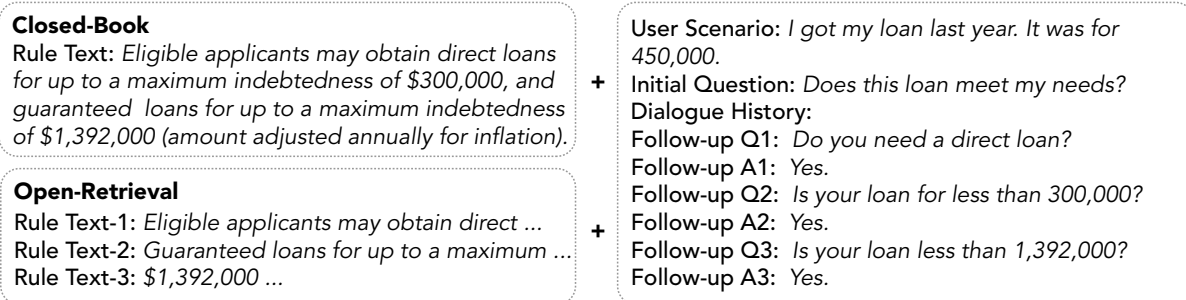
with people according to the dialogue states, and completes specific tasks, such as ordering meals (Liu et al., 2013) and air tickets (Price, 1990). In real-world scenario, annotating data such as intents and slots is expensive. Inspired by the studies of reading comprehension (Rajpurkar et al., 2016, 2018; Zhang et al., 2020c, 2021), there appears a more general task — conversational machine reading (CMR) (Saeidi et al., 2018): given the inquiry, the machine is required to retrieve relevant supporting rule documents, the machine should judge whether the goal is satisfied according to the dialogue context, and make decisions or ask clarification questions.

A variety of methods have been proposed for the CMR task, including 1) sequential models that encode all the elements and model the matching relationships with attention mechanisms (Zhong and Zettlemoyer, 2019; Lawrence et al., 2019; Verma et al., 2020; Gao et al., 2020a,b); 2) graph-based methods that capture the discourse structures of the rule texts and user scenario for better interactions (Ouyang et al., 2021). However, there are two sides of challenges that have been neglected:

1) Open-retrieval of supporting evidence. The above existing methods assume that the relevant rule documents are given before the system interacts with users, which is in a closed-book style. In real-world applications, the machines are often required to retrieve supporting information to respond to incoming high-level queries in an interactive manner, which results in an open-retrieval setting (Gao et al., 2021). The comparison of the closed-book setting and open-retrieval setting is shown in Figure 1.

2) The gap between decision making and question generation. Existing CMR studies generally regard CMR as two separate tasks and design independent systems. Only the result of decision making will be fed back to the question generation module. As a result, the question generation mod-

\*Corresponding author. # Equal contribution. This paper was partially supported by Key Projects of National Natural Science Foundation of China (U1836222 and 61733011).



(a) Closed-book setting v.s. open-retrieval setting

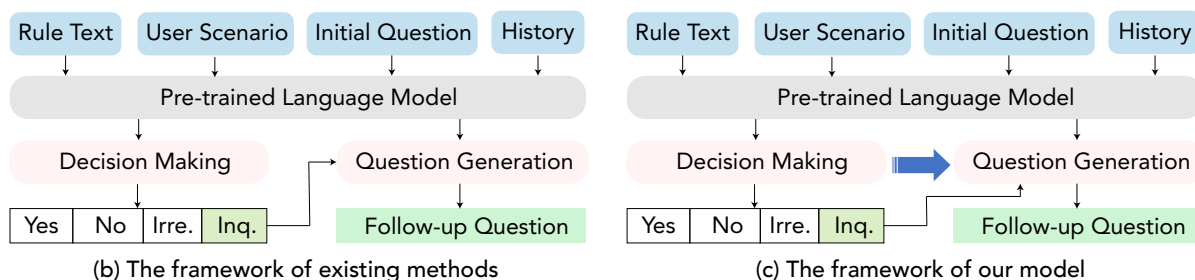


Figure 1: The overall framework for our proposed model (c) compared with the existing ones (b). Previous studies generally regard CMR as two separate tasks and design independent systems. Technically, only the result of decision making will be fed to the question generation module, thus there is a gap between the dialogue states of decision making and question generation. To reduce the information gap, our model bridges the information transition between the two salient dialogue states and benefits from a richer rule reference through open-retrieval (a).

ule knows nothing about the actual conversation states, which leads to poorly generated questions. There are even cases when the decision masking result is improved, but the question generation is decreased as reported in previous studies (Ouyang et al., 2021).

In this work, we design an end-to-end system by Open-retrieval of Supporting evidence and bridging deCision mAking and question geneRation (OSCAR),<sup>1</sup> to bridge the information transition between the two salient dialogue states of decision making and question generation, at the same time benefiting from a richer rule reference through open retrieval. In summary, our contributions are three folds:

1) For the task, we investigate the open-retrieval setting for CMR. We bridge decision making and question generation for the challenging CMR task, which is the first practice to our best knowledge.

2) For the technique, we design an end-to-end framework where the dialogue states for decision making are employed for question generation, in contrast to the independent models or pipeline systems in previous studies. Besides, a variety of

strategies are empirically studied for smoothing the two dialogue states in only one decoder.

3) Experiments on the ShARC dataset show the effectiveness of our model, which achieves the new state-of-the-art results. A series of analyses show the contributing factors.

## 2 Related Work

Most of the current conversation-based reading comprehension tasks are formed as either span-based QA (Reddy et al., 2019; Choi et al., 2018) or multi-choice tasks (Sun et al., 2019; Cui et al., 2020), both of which neglect the vital process of question generation for confirmation during the human-machine interaction. In this work, we are interested in building a machine that can not only make the right decisions but also raise questions when necessary. The related task is called conversational machine reading (Saeidi et al., 2018) which consists of two separate subtasks: decision making and question generation. Compared with conversation-based reading comprehension tasks, our concerned CMR task is more challenging as it involves rule documents, scenarios, asking clarification questions, and making a final decision.

Existing works (Zhong and Zettlemoyer, 2019;

<sup>1</sup>Our source codes are available at <https://github.com/ozyysr/OSCAR>.

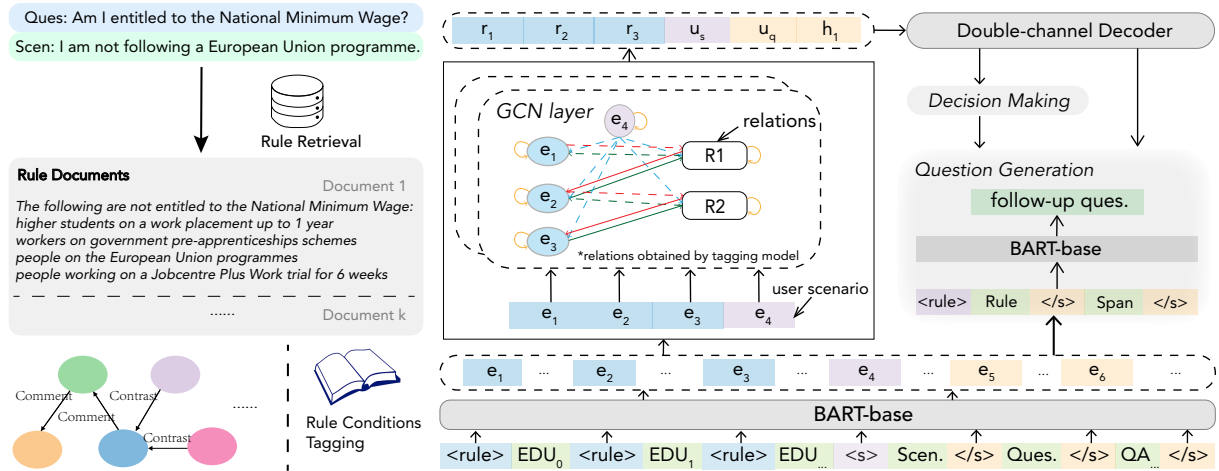


Figure 2: The overall structure of our model OSCAR. The left part introduces the retrieval and tagging process for rule documents, which is then fed into the encoder together with other necessary information.

Lawrence et al., 2019; Verma et al., 2020; Gao et al., 2020a,b; Ouyang et al., 2021) have made progress in modeling the matching relationships between the rule document and other elements such as user scenarios and questions. These studies are based on the hypothesis that the supporting information for answering the question is provided, which does not meet the real-world applications. Therefore, we are motivated to investigate the open-retrieval settings (Qu et al., 2020), where the retrieved background knowledge would be noisy. Gao et al. (2021) makes the initial attempts of open-retrieval for CMR. However, like previous studies, the common solution is training independent or pipeline systems for the two subtasks and does not consider the information flow between decision making and question generation, which would eventually hinder the model performance. Compared to existing methods, our method makes the first attempt to bridge the gap between decision making and question generation, by smoothing the two dialogue states in only one decoder. In addition, we improve the retrieval process by taking advantage of the traditional TF-IDF method and the latest dense passage retrieval model (Karpukhin et al., 2020).

### 3 Open-retrieval Setting for CMR

In the CMR task, each example is formed as a tuple  $\{R, U_s, U_q, C\}$ , where  $R$  denotes the rule texts,  $U_s$  and  $U_q$  are user scenarios and user questions, respectively, and  $C$  represents the dialogue history. For open-retrieval CMR,  $R$  is a subset retrieved from a large candidate corpus  $\mathcal{D}$ . The goal is to train a discriminator  $\mathcal{F}(\cdot, \cdot)$  for decision making,

and a generator  $\mathcal{G}(\cdot, \cdot)$  on  $\{R, U_s, U_q, C\}$  for question generation.

## 4 Model

Our model is composed of three main modules: retriever, encoder, and decoder. The retriever is employed to retrieve the related rule texts for the given user scenario and question. The encoder takes the tuple  $\{R, U_s, U_q, C\}$  as the input, encodes the elements into vectors and captures the contextualized representations. The decoder makes a decision or generates a question once the decision is ‘‘inquiry’’. Figure 1 overviews the model architecture, we will elaborate the details in the following part.

### 4.1 Retrieval

To obtain the supporting rules, we construct the query by concatenating the user question and user scenario. The retriever calculates the semantic matching score between the query and the candidate rule texts from the pre-defined corpus and returns the top- $k$  candidates. In this work, we employ TF-IDF and DPR (Karpukhin et al., 2020) in our retrieval, which are representatives for sparse and dense retrieval methods. TF-IDF stands for term frequency-inverse document frequency, which is used to reflect how relevant a term is in a given document. DPR is a dense passage retrieval model that calculates the semantic matching using dense vectors, and it uses embedding functions that can be trained for specific tasks.

## 4.2 Graph Encoder

One of the major challenges of CMR is interpreting rule texts, which have complex logical structures between various inner rule conditions. According to Rhetorical Structure Theory (RST) of discourse parsing (Mann and Thompson, 1988), we utilize a pre-trained discourse parser (Shi and Huang, 2019)<sup>2</sup> to break the rule text into clause-like units called elementary discourse units (EDUs) to extract the in-line rule conditions from the rule texts.

**Embedding** We employ pre-trained language model (PrLM) model as the backbone of the encoder. As shown in the figure, the input of our model includes rule document which has already be parsed into EDUs with explicit discourse relation tagging, user initial question, user scenario and the dialog history. Instead of inserting a [CLS] token before each rule condition to get a sentence-level representation, we use [RULE] which is proved to enhance performance (Lee et al., 2020). Formally, the sequence is organized as: {[RULE] EDU<sub>0</sub> [RULE] EDU<sub>1</sub> [RULE] EDU<sub>k</sub> [CLS] Question [CLS] Scenario [CLS] History [SEP]}. Then we feed the sequence to the PrLM to obtain the contextualized representation.

**Interaction** To explicitly model the discourse structure among the rule conditions, we first annotate the discourse relationships between the rule conditions and employ a relational graph convolutional network following Ouyang et al. (2021) by regarding the rule conditions as the vertices. The graph is formed as a Levi graph (Levi, 1942) that regards the relation edges as additional vertices. For each two vertices, there are six types of possible edges derived from the discourse parsing, namely, *default-in*, *default-out*, *reverse-in*, *reverse-out*, *self*, and *global*. Furthermore, to build the relationship with the background user scenario, we add an extra global vertex of the user scenario that connects all the other vertices. As a result, there are three types of vertices, including the rule conditions, discourse relations, and the global scenario vertex.

For rule condition and user scenario vertices,

<sup>2</sup>This discourse parser gives a state-of-the-art performance on STAC so far. There are 16 discourse relations according to STAC (Asher et al., 2016), including comment, clarification-question, elaboration, acknowledgment, continuation, explanation, conditional, question-answer, alternation, question-elaboration, result, background, narration, correction, parallel, and contrast.

we fetch the contextualized representation of the special tokens [RULE] and [CLS] before the corresponding sequences, respectively. For relation vertices, they are initialized as the conventional embedding layer, whose representations are obtained through a lookup table.

For each rule document that is composed of multiple rule conditions, i.e., EDUs, let  $h_p$  denote the initial representation of every node  $v_p$ , the graph-based information flow process can be written as:

$$h_p^{(l+1)} = \text{ReLU}\left(\sum_{r \in R_L} \sum_{v_q \in \mathcal{N}_r(v_p)} \frac{1}{c_{p,r}} w_r^{(l)} h_q^{(l)}\right), \quad (1)$$

where  $\mathcal{N}_r(v_p)$  denotes the neighbors of node  $v_p$  under relation  $r$  and  $c_{p,r}$  is the number of those nodes.  $w_r^{(l)}$  is the trainable parameters of layer  $l$ .

We have the last-layer output of discourse graph:

$$\begin{aligned} g_p^{(l)} &= \text{Sigmoid}(h_p^{(l)} W_{r,g}), \\ r_p^{(l+1)} &= \text{ReLU}\left(\sum_{r \in R_L} \sum_{v_q \in \mathcal{N}_r(v_p)} g_q^{(l)} \frac{1}{c_{p,r}} w_r^{(l)} h_q^{(l)}\right), \end{aligned} \quad (2)$$

where  $W_{r,g}^{(l)}$  is a learnable parameter under relation type  $r$  of the  $l$ -th layer. The last-layer hidden states for all the vertices  $r_p^{(l+1)}$  are used as the graph representation for the rule document. For all the  $k$  rule documents from the retriever, we concatenate  $r_p^{(l+1)}$  for each rule document, and finally have  $r = \{r_1, r_2, \dots, r_m\}$  where  $m$  is the total number of the vertices among those rule documents.

## 4.3 Double-channel Decoder

Before decoding, we first accumulate all the available information through a self-attention layer (Vaswani et al., 2017a) by allowing all the rule conditions and other elements to attend to each other. Let  $[r_1, r_2, \dots, r_m; u_q; u_s; h_1, h_2, \dots, h_n]$  denote all the representations,  $r_i$  is the representation of the discourse graph,  $u_q$ ,  $u_s$  and  $h_i$  stand for the representation of user question, user scenario and dialog history respectively.  $n$  is the number of history QAs. After encoding, the output is represented as:

$$H_c = [\tilde{r}_1, \tilde{r}_2, \dots, \tilde{r}_m; \tilde{u}_q, \tilde{u}_s; \tilde{h}_1, \tilde{h}_2, \dots, \tilde{h}_n], \quad (3)$$

which is then used for the decoder.

**Decision Making** Similar to existing works (Zhong and Zettlemoyer, 2019; Gao et al., 2020a,b), we apply an entailment-driven approach for decision making. A linear transformation tracks the fulfillment state of each rule condition among *entailment*, *contradiction* and *Unmentioned*. As a result, our model makes the decision by

$$f_i = W_f \tilde{r}_i + b_f \in \mathbb{R}^3, \quad (4)$$

where  $f_i$  is the score predicted for the three labels of the  $i$ -th condition. This prediction is trained via a cross entropy loss for multi-classification problems:

$$\mathcal{L}_{entail} = -\frac{1}{N} \sum_{i=1}^N \log \text{softmax}(f_i)_r, \quad (5)$$

where  $r$  is the ground-truth state of fulfillment.

After obtaining the state of every rule, we are able to give a final decision towards whether it is *Yes*, *No*, *Inquire* or *Irrelevant* by attention.

$$\begin{aligned} \alpha_i &= w_\alpha^T [f_i; \tilde{r}_i] + b_\alpha \in \mathbb{R}^1, \\ \tilde{\alpha}_i &= \text{softmax}(\alpha)_i \in [0, 1], \\ z &= W_z \sum_i \tilde{\alpha}_i [f_i; \tilde{r}_i] + b_z \in \mathbb{R}^4, \end{aligned} \quad (6)$$

where  $\alpha_i$  is the attention weight for the  $i$ -th decision and  $z$  has the score for all the four possible states. The corresponding training loss is

$$\mathcal{L}_{decision} = -\log \text{softmax}(z)_l. \quad (7)$$

The overall loss for decision making is:

$$\mathcal{L}_d = \mathcal{L}_{decision} + \lambda \mathcal{L}_{entail}. \quad (8)$$

**Question Generation** If the decision is made to be *Inquire*, the machine needs to ask a follow-up question to further clarify. Question generation in this part is mainly based on the uncovered information in the rule document, and then that information will be rephrased into a question. We predict the position of an under-specified span within a rule document in a supervised way. Following Devlin et al. (2019), our model learns a start vector  $w_s \in \mathbb{R}^d$  and end vector  $w_e \in \mathbb{R}^d$  to indicate the start and end positions of the desired span:

$$span = \arg \min_{i,j,k} (w_s^T t_{k,i} + w_e^T t_{k,j}), \quad (9)$$

where  $t_{k,i}$  denote the  $i$ -th token in the  $k$ -th rule sentence. The ground-truth span labels are generated by calculating the edit distance between the rule

span and the follow-up questions. Intuitively, the shortest rule span with the minimum edit distance is selected to be the under-specified span.

Existing studies deal with decision making and question generation independently (Zhong and Zettlemoyer, 2019; Lawrence et al., 2019; Verma et al., 2020; Gao et al., 2020a,b), and use hard-label decisions to activate question generation. These methods inevitably suffer from error propagation if the model makes the wrong decisions. For example, if the made decision is not “inquiry”, the question generation module will not be activated which may be supposed to ask questions in the cases. For the open-retrieval CMR that involves multiple rule texts, it even brings more diverse rule conditions as a reference, which would benefit for generating meaningful questions.

Therefore, we concatenate the rule document and the predicted span to form an input sequence:  $x = [\text{CLS}] \text{ Span } [\text{SEP}] \text{ Rule Documents } [\text{SEP}]$ . We feed  $x$  to BART encoder (Dong et al., 2019) and obtain the encoded representation  $H_e$ . To take advantage of the contextual states of the overall interaction of the dialogue states, we explore two alternative smoothing strategies:

1. **Direct Concatenation** concatenates  $H_c$  and  $H_e$  to have  $H = [H_c; H_e]$ .
2. **Gated Attention** applies multi-head attention mechanism (Vaswani et al., 2017b) to append the contextual states to  $H_e$  to get  $\hat{H} = \text{Attn}(H_e, K, V)$  where  $\{K, V\}$  are packed from  $H_c$ . Then a gate control  $\lambda$  is computed as  $\text{sigmoid}(W_\lambda \hat{H} + U_\lambda H^e)$  to get the final representation  $H = H^e + \lambda \hat{H}$ .

$H$  is then passed to the BART decoder to generate the follow-up question. At the  $i$ -th time-step,  $H$  is used to generate the target token  $y_i$  by

$$P(y_i | y_{<i}, x; \theta) \propto \exp(W_d \tanh(W_w H)), \quad (10)$$

where  $\theta$  denotes all the trainable parameters.  $W_d$  and  $W_w$  are projection matrices. The training objective is computed by

$$\mathcal{L}_g = \arg \max \sum_{i=1}^I \log P(y_i | y_{<i}, x; \theta). \quad (11)$$

The overall loss function for end-to-end training is

$$\mathcal{L} = \mathcal{L}_d + \mathcal{L}_g. \quad (12)$$

Model	Dev Set				Test Set			
	Decision Making		Question Gen.		Decision Making		Question Gen.	
	Micro	Macro	F1 <sub>BLEU1</sub>	F1 <sub>BLEU4</sub>	Micro	Macro	F1 <sub>BLEU1</sub>	F1 <sub>BLEU4</sub>
<i>w/ TF-IDF</i>								
E <sup>3</sup>	61.8±0.9	62.3±1.0	29.0±1.2	18.1±1.0	61.4±2.2	61.7±1.9	31.7±0.8	22.2±1.1
EMT	65.6±1.6	66.5±1.5	36.8±1.1	32.9±1.1	64.3±0.5	64.8±0.4	38.5±0.5	30.6±0.4
DISCERN	66.0±1.6	66.7±1.8	36.3±1.9	28.4±2.1	66.7±1.1	67.1±1.2	36.7±1.4	28.6±1.2
DP-RoBERTa	73.0±1.7	73.1±1.6	45.9±1.1	40.0±0.9	70.4±1.5	70.1±1.4	40.1±1.6	34.3±1.5
MUDERN	78.4±0.5	78.8±0.6	49.9±0.8	42.7±0.8	75.2±1.0	75.3±0.9	47.1±1.7	40.4±1.8
<i>w/ DPR++</i>								
MUDERN	79.7±1.2	80.1±1.0	50.2±0.7	42.6±0.5	75.6±0.4	75.8±0.3	48.6±1.3	40.7±1.1
OSCAR	80.5±0.5	80.9±0.6	51.3±0.8	43.1±0.8	76.5±0.5	76.4±0.4	49.1±1.1	41.9±1.8

Table 1: Results on the validation and test set of OR-ShARC. The first block presents the results of public models from Gao et al. (2021), and the second block reports the results of our implementation of the SOTA model MUDERN, and ours based on DPR++. The average results with a standard deviation on 5 random seeds are reported.

Model	Seen		Unseen	
	F1 <sub>BLEU1</sub>	F1 <sub>BLEU4</sub>	F1 <sub>BLEU1</sub>	F1 <sub>BLEU4</sub>
MUDERN	62.6	57.8	33.1	24.3
OSCAR	64.6	59.6	34.9	25.1

Table 2: The comparison of question generation on the seen and unseen splits.

## 5 Experiments

### 5.1 Datasets

For the evaluation of open-retrieval setting, we adopt the OR-ShARC dataset (Gao et al., 2021), which is a revision of the current CMR benchmark — ShARC (Saeidi et al., 2018). The original dataset contains up to 948 dialog trees clawed from government websites. Those dialog trees are then flattened into 32,436 examples consisting of *utterance\_id*, *tree\_id*, *rule document*, *initial question*, *user scenario*, *dialog history*, *evidence* and the *decision*. The update of OR-ShARC is the removal of the gold rule text for each sample. Instead, all rule texts used in the ShARC dataset are served as the supporting knowledge sources for retrieval. There are 651 rules in total. Since the test set of ShARC is not public, the train, dev and test are further manually split, whose sizes are 17,936, 1,105, 2,373, respectively. For the dev and test sets, around 50% of the samples ask questions on rule texts used in training (seen) while the remaining of them contain questions on unseen (new) rule texts. The rationale behind seen and unseen splits for the validation and test set is that the two cases mimic the real usage scenario: users may ask questions about rule text which 1) exists in the training data (i.e., dialog history, scenario) as well as 2) completely newly added rule text.

### 5.2 Evaluation Metrics

For the decision-making subtask, ShARC evaluates the Micro- and Macro- Acc. for the results of classification. For question generation, the main metric is F1<sub>BLEU</sub> proposed in Gao et al. (2021), which calculates the BLEU scores for question generation when the predicted decision is “inquire”.

### 5.3 Implementation Details

Following the current state-of-the-art MUDERN model (Gao et al., 2021) for open CMR, we employ BART (Dong et al., 2019) as our backbone model and the BART model serves as our baseline in the following sections. For open retrieval with DPR, we fine-tune DPR in our task following the same training process as the official implementation, with the same data format stated in the DPR GitHub repository.<sup>3</sup> Since the data process requires hard negatives (*hard\_negative\_ctxs*), we constructed them using the most relevant rule documents (but not the gold) selected by TF-IDF and left the *negative\_ctxs* to be empty as it can be. For discourse parsing, we keep all the default parameters of the original discourse relation parser<sup>4</sup>, with F1 score achieving 55. The dimension of hidden states is 768 for both the encoder and decoder. The training process uses Adam (Kingma and Ba, 2015) for 5 epochs with a learning rate set to 5e-5. We also use gradient clipping with a maximum gradient norm of 2, and a total batch size of 16. The parameter  $\lambda$  in the decision making objective is set to 3.0. For BART-based decoder for question generation, the beam size is set to 10

<sup>3</sup><https://github.com/facebookresearch/DPR>

<sup>4</sup><https://github.com/shizhouxing/DialogueDiscourseParsing>

Model	Dev Set				Test Set			
	Top1	Top5	Top10	Top20	Top1	Top5	Top10	Top20
TF-IDF	53.8	83.4	94.0	96.6	66.9	90.3	94.0	96.6
DPR	48.1	74.6	84.9	90.5	52.4	80.3	88.9	92.6
TF-IDF + DPR	66.3	90.0	92.4	94.5	79.8	95.4	97.1	97.5

Table 3: Comparison of the open-retrieval methods.

TF-IDF	Top1	Top5	Top10	Top20
Train	59.9	83.8	94.4	94.2
Dev	53.8	83.4	94.0	96.6
Seen Only	62.0	84.2	90.2	93.2
Unseen Only	46.9	82.8	90.7	83.1
Test	66.9	90.3	94.0	96.6
Seen Only	62.1	83.4	89.4	93.8
Unseen Only	70.4	95.3	97.3	98.7

Table 4: Retrieval Results of TF-IDF.

DPR	Top1	Top5	Top10	Top20
Train	77.2	96.5	99.0	99.8
Dev	48.1	74.6	84.9	90.5
Seen Only	77.4	96.8	98.6	99.6
Unseen Only	23.8	56.2	73.6	83.0
Test	52.4	80.3	88.9	92.6
Seen Only	76.2	96.1	98.6	99.8
Unseen Only	35.0	68.8	81.9	87.3

Table 5: Retrieval Results of DPR.

for inference. We report the averaged result of five randomly run seeds with deviations.

## 5.4 Results

Table 1 shows the results of OSCAR and all the baseline models for the End-to-End task on the dev and test set with respect to the evaluation metrics mentioned above. Evaluating results indicate that OSCAR outperforms the baselines in all of the metrics. In particular, it outperforms the public state-of-the-art model MUDERN by 1.3% in Micro Acc. and 1.1% in Macro Acc for the decision making stage on the test set. The question generation quality is greatly boosted via our approaches. Specifically,  $F1_{BLEU1}$  and  $F1_{BLEU4}$  are increased by 2.0% and 1.5% on the test set respectively.

Since the dev set and test set have a 50% split of user questions between seen and unseen rule documents as described in Section 5.1, to analyze the performance of the proposed framework over seen and unseen rules, we have added a comparison of question generation on the seen and unseen splits as shown in Table 2. The results show consistent gains for both of the seen and unseen splits.

## 6 Analysis

### 6.1 Comparison of Open-Retrieval Methods

We compare two typical retrievals methods, TF-IDF and Dense Passage Retrieval (DPR), which are widely-used traditional models from sparse vector space and recent dense-vector-based ones for open-domain retrieval, respectively. We also present the results of TF-IDF+DPR (denoted DPR++) follow-

ing Karpukhin et al. (2020), using a linear combination of their scores as the new ranking function.

The overall results are present in Table 3. We see that TF-IDF performs better than DPR, and combining TF-IDF and DPR (DPR++) yields substantial improvements. To investigate the reasons, we collect the detailed results of the *seen* and *unseen* subsets for the dev and test sets, from which we observe that TF-IDF generally works well on both the seen and unseen sets, while DPR is degraded on the unseen set. The most plausible reason would be that DPR is trained on the training set, it can only give better results on the *seen* subsets because *seen* subsets share the same rule texts for retrieval with the training set. However, DPR may easily suffer from over-fitting issues that result in the relatively weak scores on the *unseen* sets. Based on the complementary merits, combining the two methods would take advantage of both sides, which achieves the best results finally.

### 6.2 Decision Making

By means of TF-IDF + DPR retrieval, we compare our model with the previous SOTA model MUDERN (Gao et al., 2021) for comparison on the open-retrieval setting. According to the results in Table 1, we observe that our method can achieve a better performance than DISCERN, which indicates that the graph-like discourse modeling works well in the open-retrieval setting in general.

### 6.3 Question Generation

**Overall Results** We first compare the vanilla question generation with our method with encoder

DPR++	Top1	Top5	Top10	Top20
Train	84.2	99.0	99.9	100
Dev	66.3	90.0	92.4	94.5
Seen Only	84.6	98.0	99.8	100
Unseen Only	51.2	83.3	86.3	100
Test	79.8	95.4	97.1	97.5
Seen Only	83.7	98.5	99.9	100
Unseen Only	76.9	93.1	95	95.6

Table 6: Retrieval Results of DPR++.

Model	Dev Set		Test Set	
	F1 <sub>BLEU1</sub>	F1 <sub>BLEU4</sub>	F1 <sub>BLEU1</sub>	F1 <sub>BLEU4</sub>
OSCAR	51.3±0.8	43.1±0.8	49.1±1.1	41.9±1.8
w/o GS	50.9±0.9	43.0±0.7	48.7±1.3	41.6±1.5
w/o SS	50.6±0.6	42.8±0.5	48.1±1.4	41.4±1.4
w/o both	49.9±0.8	42.7±0.8	47.1±1.7	40.4±1.8

Table 7: Question generation results on the OR-ShARC dataset. SS and GS denote the sequential states and graph states, respectively.

states. Table 7 shows the results, which verify that both the sequential states and graph states from the encoding process contribute to the overall performance as removing any one of them causes a performance drop on both F1<sub>BLEU1</sub> and F1<sub>BLEU4</sub>. Especially, when removing *GS/SS*, those two matrices drops by a great margin, which shows the contributions. The results indicate that bridging the gap between decision making and question generation is necessary.<sup>5</sup>

**Smoothing Strategies** We explore the performance of different strategies when fusing the contextual states into BART decoder, and the results are shown in Table 8, from which we see that the gating mechanism yields the best performance. The most plausible reason would be the advantage of using the gates to filter the critical information.

**Upper-bound Evaluation** To further investigate how the encoder states help generation, we construct a “gold” dataset as the upper bound evaluation, in which we replace the reference span with the ground-truth span by selecting the span of the rule text which has the minimum edit distance with the to-be-asked follow-up question, in contrast to the original span that is predicted by our model. We find an interesting observation that the BLEU-1 and BLEU-4 scores drop from 90.64 → 89.23, and

<sup>5</sup>Our method is also applicable to other generation architectures such as T5 (Raffel et al., 2020). For the reference of interested readers, we tried to employ T5 as our backbone, achieving better performance: 53.7/45.0 for dev and 52.5/43.7 for test (F1<sub>BLEU1</sub>/F1<sub>BLEU4</sub>).

Model	Dev Set		Test Set	
	F1 <sub>BLEU1</sub>	F1 <sub>BLEU4</sub>	F1 <sub>BLEU1</sub>	F1 <sub>BLEU4</sub>
Concatenation	51.3±0.8	43.1±0.8	49.1±1.1	41.9±1.8
Gated Attention	51.6±0.6	44.1±0.5	49.5±1.2	42.1±1.4

Table 8: Question generation results using different smoothing strategies on the OR-ShARC dataset.

Model	ShARC		OR-ShARC	
	BLEU1	BLEU4	F1 <sub>BLEU1</sub>	F1 <sub>BLEU4</sub>
BASELINE	62.4±1.6	47.4±1.6	50.2±0.7	42.6±0.5
OSCAR	63.3±1.2	48.1±1.4	51.6±0.6	44.4±0.4

Table 9: Performance comparison on the dev sets of the closed-book and open-retrieval tasks.

89.61 → 85.81 after aggregating the DM states on the constructed dataset. Compared with the experiments on the original dataset, the performance gap shows that using embeddings from the decision making stage would well fill the information loss caused by the span prediction stage, and would be beneficial to deal with the errors propagation.

**Closed-book Evaluation** Besides the open-retrieval task, our end-to-end unified modeling method is also applicable to the traditional CMR task. We conduct comparisons on the original ShARC question generation task with provided rule documents to evaluate the performance. Results in Table 9 show the obvious advantage on the open-retrieval task, indicating the strong ability to extract key information from noisy documents.

## 6.4 Case Study

To explore the generation quality intuitively, we randomly collect and summarize error cases of the baseline and our models for comparison. Results of a few typical examples are presented in Figure 3. We evaluate the examples in term of three aspects, namely, *factualness*, *succinctness* and *informativeness*. The difference of generation by OSCAR and the baseline are highlighted in green, while the blue words are the indication of the correct generations. One can easily observe that our generation outperforms the baseline model regarding factualness, succinctness, and informativeness. This might be because that the incorporation of features from the decision making stage can well fill in the gap of information provided for question generation.



Types	Gold Snippet & Scenario	Predicted Span	Our Gen. & Ori. Gen.
<b>Factualness</b> (associates with the correct facts)	<b>Snippet:</b> You can still get Statutory Maternity Leave and SMP if <b>your baby</b> : (1) is born early; (2) is stillborn after the start of your 24th week of pregnancy (3) dies after being born <b>Scenario:</b> (empty)	is born early	<b>Ours:</b> Was <b>your baby</b> born early? <b>Original:</b> Was <b>you</b> born early?
<b>Succinctness</b> (does not contain redundant information)	<b>Snippet:</b> ..., In general, loan funds may be used for normal operating expenses, machinery and equipment, minor real estate repairs or improvements, and refinancing debt. <b>Scenario:</b> I have no intentions of using the loan for <b>operating expenses...</b>	expenses, machinery, equipment	<b>Ours:</b> Will it be used for <b>machinery</b> and <b>equipment</b> ? <b>Original:</b> Will it be used for <b>expenses, machinery</b> and <b>equipment</b> ?
<b>Informativeness</b> (covers the most important content)	<b>Snippet:</b> The eligible <b>items</b> include: (1) medical, veterinary and scientific equipment (2) ambulances (3) goods for disabled people (4) motor vehicles for medical use. <b>Scenario:</b> (empty)	goods for disabled people	<b>Ours:</b> Is <b>it</b> for disabled people? <b>Original:</b> Is <b>the item goods</b> for disabled people?

Figure 3: Question generation examples of OSCAR and the original model. “Our Gen.” stands for the question generated by OSCAR; “Ori. Gen.” stands for the question generated by the baseline model.

## 7 Conclusion

In this paper, we study conversational machine reading based on open-retrieval of supporting rule documents, and present a novel end-to-end framework OSCAR to enhance the question generation by referring to the rich contextualized dialogue states that involve the interactions between rule conditions, user scenario, initial question and dialogue history. Our OSCAR consists of three main modules including retriever, encoder, and decoder as a unified model. Experiments on OR-ShARC show the effectiveness by achieving a new state-of-the-art result. Case studies show that OSCAR can generate high-quality questions compared with the previous widely-used pipeline systems.

## Acknowledgments

We thank Yifan Gao for providing the sources of MUDERN (Gao et al., 2021) and valuable suggestions to help improve this work.

## References

- Nicholas Asher, Julie Hunter, Mathieu Morey, Benamara Farah, and Stergos Afantenos. 2016. **Discourse structure and dialogue acts in multiparty dialogue: the STAC corpus**. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 2721–2727, Portorož, Slovenia. European Language Resources Association (ELRA).
- Deng Cai, Yan Wang, Wei Bi, Zhaopeng Tu, Xiaojiang Liu, Wai Lam, and Shuming Shi. 2019. **Skeleton-to-**

**response: Dialogue generation guided by retrieval memory**. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1219–1228, Minneapolis, Minnesota. Association for Computational Linguistics.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentaoh Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. **QuAC: Question answering in context**. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2174–2184, Brussels, Belgium. Association for Computational Linguistics.

Leyang Cui, Yu Wu, Shujie Liu, Yue Zhang, and Ming Zhou. 2020. **MuTual: A dataset for multi-turn dialogue reasoning**. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1406–1416, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. **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: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. **Unified language model pre-training for natural language understanding and generation**. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019*,

- NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 13042–13054.
- Yifan Fan, Xudong Luo, and Pingping Lin. 2020. A survey of response generation of dialogue systems. *International Journal of Computer and Information Engineering*, 14(12):461–472.
- Yifan Gao, Jingjing Li, Michael R Lyu, and Irwin King. 2021. Open-retrieval conversational machine reading. *arXiv preprint arXiv:2102.08633*.
- Yifan Gao, Chien-Sheng Wu, Shafiq Joty, Caiming Xiong, Richard Socher, Irwin King, Michael Lyu, and Steven C.H. Hoi. 2020a. **Explicit memory tracker with coarse-to-fine reasoning for conversational machine reading**. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 935–945, Online. Association for Computational Linguistics.
- Yifan Gao, Chien-Sheng Wu, Jingjing Li, Shafiq Joty, Steven C.H. Hoi, Caiming Xiong, Irwin King, and Michael Lyu. 2020b. **Discern: Discourse-aware entailment reasoning network for conversational machine reading**. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2439–2449, Online. Association for Computational Linguistics.
- Jia-Chen Gu, Chongyang Tao, Zhenhua Ling, Can Xu, Xiubo Geng, and Daxin Jiang. 2021. **MPC-BERT: A pre-trained language model for multi-party conversation understanding**. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3682–3692, Online. Association for Computational Linguistics.
- Xiaodong Gu, Kang Min Yoo, and Jung-Woo Ha. 2020. Dialogbert: Discourse-aware response generation via learning to recover and rank utterances. *arXiv:2012.01775*.
- Minlie Huang, Xiaoyan Zhu, and Jianfeng Gao. 2020. Challenges in building intelligent open-domain dialog systems. *ACM Transactions on Information Systems (TOIS)*, 38(3):1–32.
- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. **Dense passage retrieval for open-domain question answering**. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.
- Diederik P. Kingma and Jimmy Ba. 2015. **Adam: A method for stochastic optimization**. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.
- Carolin Lawrence, Bhushan Kotnis, and Mathias Niepert. 2019. **Attending to future tokens for bidirectional sequence generation**. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1–10, Hong Kong, China. Association for Computational Linguistics.
- Haejun Lee, Drew A. Hudson, Kangwook Lee, and Christopher D. Manning. 2020. **SLM: Learning a discourse language representation with sentence unshuffling**. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1551–1562, Online. Association for Computational Linguistics.
- Friedrich Wilhelm Levi. 1942. *Finite geometrical systems: six public lectures delivered in February, 1940, at the University of Calcutta*. University of Calcutta.
- Jingjing Liu, Panupong Pasupat, Scott Cyphers, and Jim Glass. 2013. Asgard: A portable architecture for multilingual dialogue systems. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8386–8390. IEEE.
- William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.
- Siru Ouyang, Zhuosheng Zhang, and Hai Zhao. 2021. Dialogue graph modeling for conversational machine reading. In *The Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*.
- P. J. Price. 1990. **Evaluation of spoken language systems: the ATIS domain**. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27, 1990*.
- Libo Qin, Tianbao Xie, Wanxiang Che, and Ting Liu. 2021. A survey on spoken language understanding: Recent advances and new frontiers. In *the 30th International Joint Conference on Artificial Intelligence (IJCAI-21: Survey Track)*.
- Chen Qu, Liu Yang, Cen Chen, Minghui Qiu, W. Bruce Croft, and Mohit Iyyer. 2020. **Open-retrieval conversational question answering**. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 539–548. ACM.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

- Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. [Know what you don't know: Unanswerable questions for SQuAD](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. [SQuAD: 100,000+ questions for machine comprehension of text](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.
- Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. [CoQA: A conversational question answering challenge](#). *Transactions of the Association for Computational Linguistics*, 7:249–266.
- Liliang Ren, Kaige Xie, Lu Chen, and Kai Yu. 2018. [Towards universal dialogue state tracking](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2780–2786, Brussels, Belgium. Association for Computational Linguistics.
- Marzieh Saeidi, Max Bartolo, Patrick Lewis, Sameer Singh, Tim Rocktäschel, Mike Sheldon, Guillaume Bouchard, and Sebastian Riedel. 2018. [Interpretation of natural language rules in conversational machine reading](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2087–2097, Brussels, Belgium. Association for Computational Linguistics.
- Zhouxing Shi and Minlie Huang. 2019. [A deep sequential model for discourse parsing on multi-party dialogues](#). In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7007–7014. AAAI Press.
- Kai Sun, Dian Yu, Jianshu Chen, Dong Yu, Yejin Choi, and Claire Cardie. 2019. [DREAM: A challenge data set and models for dialogue-based reading comprehension](#). *Transactions of the Association for Computational Linguistics*, 7:217–231.
- Gokhan Tur and Renato De Mori. 2011. *Spoken language understanding: Systems for extracting semantic information from speech*. John Wiley & Sons.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017a. [Attention is all you need](#). In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017b. [Attention is all you need](#). In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.
- Nikhil Verma, Abhishek Sharma, Dhiraj Madan, Danish Contractor, Harshit Kumar, and Sachindra Joshi. 2020. [Neural conversational QA: Learning to reason vs exploiting patterns](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7263–7269, Online. Association for Computational Linguistics.
- Yu Wu, Wei Wu, Dejian Yang, Can Xu, and Zhoujun Li. 2018. [Neural response generation with dynamic vocabularies](#). In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5594–5601. AAAI Press.
- Munazza Zaib, Quan Z Sheng, and Wei Emma Zhang. 2020. A short survey of pre-trained language models for conversational ai-a new age in nlp. In *Proceedings of the Australasian Computer Science Week Multiconference*, pages 1–4.
- Linhao Zhang, Dehong Ma, Xiaodong Zhang, Xiaohui Yan, and Houfeng Wang. 2020a. Graph lstm with context-gated mechanism for spoken language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9539–9546.
- Yizhe Zhang, Siqu Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020b. [DIALOGPT : Large-scale generative pre-training for conversational response generation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278, Online. Association for Computational Linguistics.
- Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. 2018. [Modeling multi-turn conversation with deep utterance aggregation](#). In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3740–3752, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
- Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. 2020c. [Semantics-aware BERT for language understanding](#). In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI*

2020, New York, NY, USA, February 7-12, 2020, pages 9628–9635. AAAI Press.

Zhuosheng Zhang, Junjie Yang, and Hai Zhao. 2021. Retrospective reader for machine reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14506–14514.

Victor Zhong and Luke Zettlemoyer. 2019. [E3: Entailment-driven extracting and editing for conversational machine reading](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2310–2320, Florence, Italy. Association for Computational Linguistics.

Pengfei Zhu, Zhuosheng Zhang, Jiangtong Li, Yafang Huang, and Hai Zhao. 2018. [Lingke: a fine-grained multi-turn chatbot for customer service](#). In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 108–112, Santa Fe, New Mexico. Association for Computational Linguistics.