

Unsupervised Context Rewriting for Open Domain Conversation

Kun Zhou^{1*}, Kai Zhang², Yu Wu³, Shujie Liu³, and Jingsong Yu¹

¹School of Software and Microelectronics, Peking University

²AI and Research Microsoft, Beijing, China

³Microsoft Research, Beijing, China

franciszhou@pku.edu.cn

{kaizh, Wu.Yu, shujliu}@microsoft.com

yjs@ss.pku.edu.cn

Abstract

Context modeling has a pivotal role in open domain conversation. Existing works either use heuristic methods or jointly learn context modeling and response generation with an encoder-decoder framework. **This paper proposes an explicit context rewriting method, which rewrites the last utterance by considering context history.** We leverage pseudo-parallel data and elaborate a context rewriting network, which is built upon the CopyNet with the reinforcement learning method. The rewritten utterance is beneficial to candidate retrieval, explainable context modeling, as well as enabling to employ a single-turn framework to the multi-turn scenario. The empirical results show that our model outperforms baselines in terms of the rewriting quality, the multi-turn response generation, and the end-to-end retrieval-based chatbots.

1 Introduction

Recent years have witnessed remarkable progress in open domain conversation (non-task oriented dialogue system) (Ji et al., 2014; Li et al., 2016a) due to the easy-accessible conversational data and the development of deep learning techniques (Bahdanau et al., 2014). One of the most difficult problems for open domain conversation is how to model the conversation context.

A conversation context is composed of multiple utterances, which raises some challenges not existing in the sentence modeling, including: 1) topic transition; 2) plenty of coreferences (he, him, she, it, they); and 3) long term dependency. To tackle these problems, existing works either refine the context by appending keywords to the last turn utterance (Yan et al., 2016), or learn a vector representation with neural networks (Serban et al.,

*This work was done when the first author was an intern at Microsoft Xiaolce.

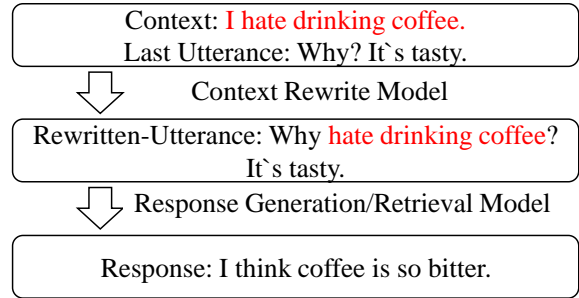


Figure 1: An Example of Context Rewriting.

2017b). However, these methods have drawbacks, for instance, correct keywords cannot be selected by heuristics rules, and a fix-length vector is not able to handle a long context.

We propose a context rewriting method, which explicitly rewrites the last utterance by considering the contextual information. Our goal is to generate a self-contained utterance, which neither has coreferences nor depends on other utterances in history. By this means, we change the input of chatbots from an entire conversation session to a rewritten sentence, which significantly reduces the difficulty of response generation/selection since the rewritten sentence is shorter and does not have redundant information. Figure 1 gives an example to further illustrate our idea.

The last utterance contains the word “it” which refers to the coffee in context. Moreover, “Why?” is an elliptical interrogative sentence, which is a shorter form of “Why hate drinking coffee?”. We rewrite the context and yield a self-contained utterance “Why hate drinking coffee? It’s tasty.” Compared to previous methods, our method enjoys the following advantages: 1) The rewriting process is friendly to the retrieval stage of retrieval-based chatbots. Retrieval-based chatbots consists of two components: candidate retrieval and candidate reranking. Traditional works (Yan et al., 2016; Wu et al., 2017) pay little attention to the

retrieval stage, which regards the entire context or context rewritten with heuristic rules as queries so noise is likely to be introduced; 2) It makes a step toward explainable and controllable context modeling, because the explicit context rewriting results are easy to debug and analyze. 3) Rewritten results enable us to employ a single-turn framework to solve the multi-turn conversation task. The single-turn conversation technology is more mature than the multi-turn conversation technology, which is able to achieve higher responding accuracy.

To this end, we propose a context rewriting network (CRN) to integrate the key information of the context and the original last utterance to build a rewritten one, so as to improve the answer performance. Our CRN model is a sequence-to-sequence network (Ilya Sutskever, 2014) with a bidirectional GRU-based encoder, and a GRU-based decoder enhanced with the CopyNet (Gu et al., 2016), which helps the CRN to directly copy words from the context. Due to the absence of the real written last utterance, unsupervised methods are used with two training stages, a pre-training stage with pseudo rewritten data, and a fine-tuning stage using reinforcement learning (RL) (Sutton et al., 1998) to maximize the reward of the final answer. Without the pre-training part, RL is unstable and slow to converge, since the randomly initialized CRN model cannot generate reasonable rewritten last utterance. On the other hand, only the pre-training part is not enough, since the pseudo data may contain errors and noise, which restricts the performance of our CRN.

We evaluate our method with four tasks, including the rewriting quality, the multi-turn response generation, the multi-turn response selection, and the end-to-end retrieval-based chatbots. Empirical results show that the outputs of our method are closer to human references than baselines. Besides, the rewriting process is beneficial to the end-to-end retrieval-based chatbots and the multi-turn response generation, and it shows slightly positive effect on the response selection.

2 Related Work

Recently, data-driven approaches for chatbots (Ritter et al., 2011; Ji et al., 2014) has achieved promising results. Existing work along this line includes retrieval-based methods (Hu et al., 2014; Ji et al., 2014; Wang et al., 2015; Yan et al., 2016; Zhou et al., 2016) and generation-based methods

(Shang et al., 2015; Serban et al., 2016; Vinyals and Le, 2015; Li et al., 2016a,b; Xing et al., 2017; Serban et al., 2017a).

Early research into retrieval-based chatbots (Wang et al., 2013; Hu et al., 2014; Wang et al., 2015) only considers the last utterances and ignores previous ones, which is also called Short Text Conversation (STC). Recently, several studies (Lowe et al., 2015; Yan et al., 2016; Wu et al., 2017, 2018b) have investigated multi-turn response selection, and obtained better results in a comparison with STC. A common practice for multi-turn retrieval-based chatbots first retrieve candidates from a large index with a heuristic context rewriting method. For example, (Wu et al., 2017) and (Yan et al., 2016) refine the last utterance by appending keywords in history, and retrieve candidates with the refined utterance. Then, response selection methods are applied to measure the relevance between history and candidates.

A number of studies about generation-based chatbots have considered multi-turn response generation. Sordoni et al. (2015) is the pioneer of this type of research, it encodes history information into a vector and feeds to the decoder. Shang et al. (2015) propose three types of attention to utilize the context information. In addition, Serban et al. (2016) propose a Hierarchical Recurrent Encoder-Decoder model (HRED), which employs a hierarchical structure to represent the context. After that, latent variables (Serban et al., 2017b) and hierarchical attention mechanism (Xing et al., 2018) have been introduced to modify the architecture of HRED. Compared to previous work, the originality of this study is that it proposes a principle way instead of heuristic rules for context rewriting, and it does not depend on parallel data. (Su et al., 2019) propose an utterance rewrite approach but need human annotation.

3 Model

Given a dialogue data set $\mathcal{D} = \{(U, r)_z\}_{z=1}^N$, where $U = \{u_0, \dots, u_n\}$ represents a sequence of utterances and r is a response candidate. We denote the last utterance as $q = u_n$ for simplicity, which is especially important to produce the response, and other utterances as $c = \{u_0, \dots, u_{n-1}\}$. The goal of our paper is to rewrite q as a self-contained utterance q^* using useful information from c , which can not only reduce the noise in multi-turn context but also lever-

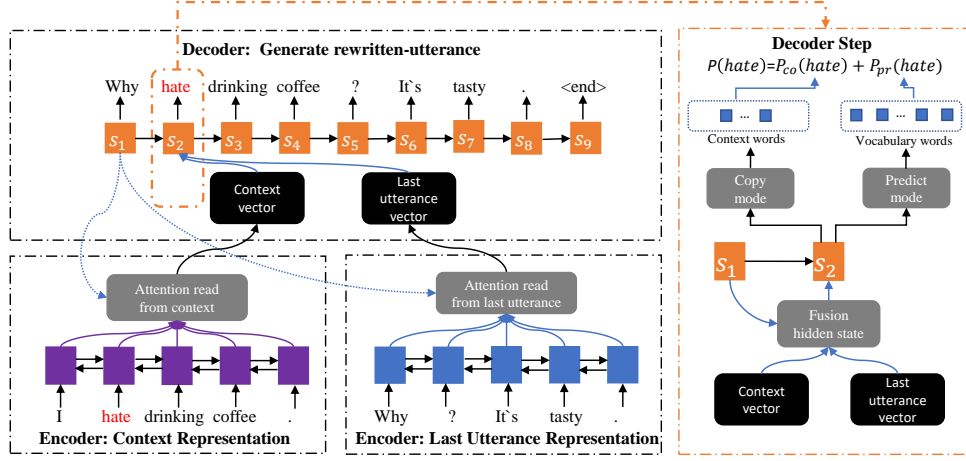


Figure 2: The Detail of CRN

age a more simple single-turn framework to solve the multi-turn end-to-end tasks. We focus on the multi-turn response generation and selection tasks.

To rewrite the last utterance q with the help of context c , we propose a context rewriting network (CRN), which is a popularly used sequence-to-sequence network, equipped with a CopyNet to copy words from the original context c (Section 3.1). Without the real paired data (pairs of the original and rewritten last utterance), our CRN model is firstly pre-trained with the pseudo data, generated by inserting extracted keywords from context into the original last utterance q (Section 3.2). To let the final response to influence the rewriting process, reinforcement learning is leveraged to further enhance our CRN model, using the rewards from the response generation and selection tasks respectively (Section 3.3).

3.1 Context Rewriting Network

As shown in Figure 2, our context rewriting network (CRN) follows the sequence to sequence framework, consisting of three parts: one encoder to learn the context (c) representation, another encoder to learn the last utterance (q) representation, and a decoder to generate the rewritten utterance q^* . Attention is also used to focus on different words in the last utterance q and the context c , and the copy mechanism is introduced to copy important words from the context c .

3.1.1 Encoder

To encode the context c and the last utterance q , bidirectional GRU is leveraged to take both the left and right words in the sentence into consideration, by concatenating the hidden states of two GRU networks in positive time direction and

negative time direction. With the bidirectional GRU, the last utterance q is encoding into $H_Q = [h_{q_1}, \dots, h_{q_{n_q}}]$, and the context c is encoding into $H_C = [h_{c_1}, \dots, h_{c_{n_c}}]$.

3.1.2 Decoder

The GRU network is also leveraged as decoder to generate the rewritten utterance q^* , in which the attention mechanism is used to extract useful information from the context c and the last utterance q , and the copy mechanism is leveraged to directly copy words from the context c into q^* . At each time step t , we fuse the information from c , q and last hidden state s_t to generate the input vector z_t of GRU as following

$$z_t = W_f^T [s_t; \sum_{i=1}^{n_q} \alpha_{q_i} h_{q_i}; \sum_{i=1}^{n_c} \alpha_{c_i} h_{c_i}] + b \quad (1)$$

where $[\cdot]$ is the concatenation operation. W_f and b are trainable parameters, s_t is the last hidden state of decoder GRU in step t , α_q and α_c are the weights of the words in q and c respectively, derived by the attention mechanism as following

$$\alpha_i = \frac{\exp(e_i)}{\sum_{j=1}^n \exp(e_j)} \quad (2)$$

$$e_i = h_i W_a s_t \quad (3)$$

where h_i is the encoder hidden state of the i^{th} word in q or c , W_a is the trainable parameter.

The copy mechanism is used to predict the next target word according to the probability of $p(y_t | s_t, H_Q, H_C)$, which is computed as

$$p(y_t | s_t, H_Q, H_C) = p_{pr}(y_t | z_t) \cdot p_m(pr | z_t) + p_{co}(y_t | z_t) \cdot p_m(co | z_t) \quad (4)$$

where y_t is the $t - th$ word in response, pr and co stand for the predict-mode and the copy-mode,

$p_{pr}(y_t|z_t)$ and $p_{co}(y_t|z_t)$ are the distributions of vocabulary word and context word which are implemented by two MLP (multi layer perceptron) classifiers, respectively. And $p_m(\cdot|\cdot)$ indicates the probability to choose the two modes, which is a MLP (multi layer perceptron) classifier with softmax as the activation function:

$$p_m(pr|z_t) = \frac{e^{\psi_{pr}(y_t, H_Q, H_C)}}{e^{\psi_{pr}(y_t, H_Q, H_C)} + e^{\psi_{co}(y_t, H_Q, H_C)}} \quad (5)$$

where $\psi_{pr}(\cdot)$, $\psi_{co}(\cdot)$ are score functions for choosing the predict-mode and copy-mode with different parameters.

3.2 Pre-training with Pseudo Data

Instead of directly leverage RL to optimize our model CRN, which could be unstable and slow to converge, we pre-train our model CRN with pseudo-parallel data. Cross-entropy is selected as the training loss to maximize the log-likelihood of the pseudo rewritten utterance.

$$L_{MLE} = -\frac{1}{N} \sum_{i=1}^n \log(p(y_t|s_t, H_Q, H_C)) \quad (6)$$

The main challenge for the pre-training stage is how to generate good pseudo data, which can integrate suitable keywords from context and the original last utterance to form a better one to generate a good response. Given the context c , we extract keywords $w_{c1:n}^*$ using pointwise mutual information (PMI) (in Section 3.2.1). With the extracted keywords $w_{c1:n}^*$, language model is leveraged to find suitable positions to insert them into the original last utterance to generate rewritten candidates s^* , which will be re-ranked leveraging the information from following process (the response generation/selection) to get final pseudo rewritten utterance Q^* (in Section 3.2.2). In the following of this section, we will introduce our pseudo data creation method in detail.

3.2.1 Key Words Extraction

To penalize common and low frequent words, and prefers the “mutually informative” words, PMI is used to extract the keywords in the context c . Given a context word w_c , and a word w_r in response r , it is the divides the prior distribution $p_c(w_c)$ by the posterior distribution $p(w_c|w_r)$ as shown as:

$$\text{PMI}(w_c, w_r) = -\log \frac{p_c(w_c)}{p(w_c|w_r)} \quad (7)$$

In order to select the keywords which contribute to the response, and are suitable to be shown in the last utterance, we also calculate $\text{PMI}(w_c, w_q)$ between the context word w_c and any word w_q in the last utterance. The final contribution score $\text{PMI}(w_c, q, r)$ for the context word w_c to the last utterance q and the response r is calculated as

$$\text{norm}(\text{PMI}(w_c, q)) + \text{norm}(\text{PMI}(w_c, r)) \quad (8)$$

where $\text{norm}(\cdot)$ is the min-max normalization among all words in c , and $\text{PMI}(w_c, q)$ (similar for $\text{PMI}(w_c, r)$) is calculated as

$$\text{PMI}(w_c, q) = \sum_{w_q \in q} \text{PMI}(w_c, w_q). \quad (9)$$

The keywords w_c^* with top-20% contribution score $\text{PMI}(w_c, q, r)$ against r and q are selected to insert into the last utterance q .¹

3.2.2 Pseudo Data Generation

Together with the extract candidate keyword, the words nearby are also extracted to form a continuous span to introduce more information, of which, at most 2 words before and after are considered. For one keyword, there are at most $C_3^1 * C_3^1 = 9$ span candidates. We apply a multi-layer RNN language model to insert the extracted key phrase to a suitable position in the last utterance. Top-3 generated sentences with high language model scores are selected as the rewritten candidates s^* .

With the information from the response, a re-rank model is used to select the best one from the candidates s^* . For end-to-end generation task, the quality of candidates is measured with the cross-entropy of a single-turn attention based encoder-decoder model M_{s2s} , hoping that the good rewritten utterance can help to generate the proper response. For the end-to-end response selection task, the quality of the candidates is measured by the rank loss of a single-turn response selection model M_{ir} , hoping that the good one can distinguish the positive and negative responses.

$$L_{M_{s2s}}(r|s^*) = -\frac{1}{n} \sum_{i=1}^n \log p(r_1, \dots, r_n|s^*) \quad (10)$$

$$L_{M_{ir}}(po, ne, s^*) = M_{ir}(po, s^*) - M_{ir}(ne, s^*) \quad (11)$$

In equation 10, r_i is the i -th word in response. In equation 11 po is the positive response, and ne

¹This threshold of PMI is based on the observation on the development set.

are the negative one.

3.3 Fine-Tuning with Reinforcement Learning

Since the generated pseudo data inevitably contains errors and noise, which limits the performance of the pre-trained model, we leverage the reinforcement learning method to build the direct connection between the context rewrite model CRN and different tasks. We first generate rewritten utterance candidates q_r with our pre-trained model, and calculate the reward $R(q_r)$ which will be maximized to optimize the network parameters of our CRN. Due to the discrete choices of words in sequential generation, the policy gradient is used to calculate the gradient.

$$\nabla_{\theta} J(\theta) = E[R \cdot \nabla \log(P(y_t|x))] \quad (12)$$

For reinforcement learning in sequential generation task, instability is a serious problem. Similar to other works (Wu et al., 2018a), we combine MLE training objective with RL objective as

$$L_{com} = L_{rl}^* + \lambda L_{MLE} \quad (13)$$

where λ is a harmonic weight.

By directly maximizing the reward from end tasks (response generation and selection), we hope that our CRN can correct the errors in the pseudo data and generate better rewritten last utterance. Two different rewards are used to fine-tune our CRN for the tasks of response generation and selection respectively. We will introduce them in detail in the following.

3.3.1 End-to-end response generation reward

Similar as we do in Section 3.2.2, for end-to-end response generation task, we use the cross-entropy loss of a single-turn attention based encoder-decoder model M_{s2s} to evaluate the quality of rewritten last utterance q_r as

$$R_g(r, q^*, q_r) = L_{M_{s2s}}(r|q^*) - L_{M_{s2s}}(r|q_r) \quad (14)$$

where $L_{M_{s2s}}$ is defined in Equation 10, r is the response candidate, q_r is the generated candidate of our CRN, and q^* is the pseudo rewritten candidate as introduced in Section 3.2.2. If q_r can bring more useful information from context, it will get lower cross-entropy to generate r than the original pseudo rewritten one q^* .

3.3.2 End-to-end response selection reward

For end-to-end response selection task, we use a single-turn response selected model M_{ir} to evalu-

ate the quality of the generated candidate q_r by the rank loss, it is calculated as

$$R_{ir}(po, ne, q^*, q_r) = L_{M_{ir}}(po, ne, q_r) - L_{M_{ir}}(po, ne, q^*) \quad (15)$$

where $L_{M_{ir}}$ is defined in Equation 11, q_r is the generated candidate of our model, and q^* is the pseudo candidate as introduced in Section 3.2.2. Similar to Equation 14, if q_r can bring more useful information, it will do better to distinguish the negative and positive responses.

4 Experiment

We conduct four experiments to validate the effectiveness of our model, including the rewriting quality, the multi-turn response generation, the multi-turn response selection, and the end-to-end retrieval-based chatbots.

We crawl human-human context-response pairs from Douban Group which is a popular forum in China and remove duplicated pairs and utterances longer than 30 words. We create pseudo rewritten context as described in Section 3.1. Because most of the responses are only relevant with the last two turn utterances, following Li et al. (2016c), we remove the utterances beyond the last two turns. We finally split 6,844,393 (c_i, q_i, q_i^*, r_i) quadruplets for training², 1000 for validation and 1074 for testing, and the last utterance in test set are selected by human and they all require rewriting to enhance information³. In the data set, the ratio between rewritten last utterance and un-rewritten⁴ the last utterance is 1.426 : 1. The average length of context c_i , last utterance q_i , response r_i , and rewritten last utterance q_i^* are 12.69, 11.90, 15.15 and 14.27 respectively.

We pre-train the CRN with the pseudo-parallel data until it coverages, then we use the reinforcement learning technique described in Section 3.3 to fine-tune the CRN. The specific details of the model hyper-parameters and optimization algorithms can be found in the **Supplementary**.

4.1 Rewriting Quality Evaluation

The detail of the training process is the same as Section 4.2.1. We evaluate the rewriting quality

²The data in the training set do not overlap with the test data of the four tasks.

³Our human-annotated test set is available at <https://github.com/lovelife/chat>

⁴Un-rewritten ones can handle utterances do not rely on their contexts.

	BLEU-4
Last Utterance	34.2
Last Utterance + Context	37.1
Last Utterance + Keyword	49.8
CRN	50.9
CRN + RL	54.2

Table 1: The result of rewriting quality.

by calculating the **BLEU-4 score** (Papineni et al., 2002), a sequence order sensitive metric, between the system outputs and human references. Such references are rewritten by a native speaker who considers the information in context. It is required that the rewritten last utterance is self-contained.

We compare our models CRN with three baselines. Firstly, we report the BLEU-4 scores of the origin last utterance and the combination of the last utterance and context. Additionally, following Wu et al. (2017), we append five keywords to the last utterance, where the keywords are selected from the context by TF-IDF weighting, which is named by Last Utterance + Keyword. The IDF score is computed on the entire training corpus.

Table 1 shows the experiment result, which indicates that our rewriting method outperforms heuristic methods. Moreover, a 54.2 BLEU-4 score means that the rewritten sentences are very similar to the human references. CRN-RL has a higher score than CRN-Pre-train on BLEU-4, it proves reinforcement learning promotes our model effectively.

4.2 Multi-turn Response Generation

Section 4.1 demonstrates the outputs of our model are more similar to the human rewritten references. In this part, we will show the influence of the context rewriting for response generation.

We use the same test data in Section 4.1 to evaluate our model in the multi-turn response generation task. The multi-turn response generation is defined as, given an entire context consisting of multiple utterances, a model should generate informative, relevant, and fluent responses. We compare against the following previous works:

S2SA: We adopt the well-known Seq2Seq with attention (Bahdanau et al., 2014) model to generate responses by feeding the last utterances q as source sentences.

HRED: Serban et al. (2016) propose using a hierarchical encoder-decoder model to handle the multi-turn response generation problem, where each utterance and the entire session are repre-

sented by different networks.

Dynamic, Static: Zhang et al. (2018) propose two state-of-the-art hierarchical recurrent attention networks for response generation. The dynamic model dynamically weights utterances in the decoding process, while the static model weights utterances before the decoding process.

4.2.1 Implementation Details

Given a context c and last utterance q , we first rewrite them with the CRN. Then the rewritten last utterance q' is fed to a single-turn generation model. The details of the model can be found in **Supplementary** and we set the same sizes of hidden states and embedding in all models. We regard two adjacent utterances in our training data to construct the training dataset (5,591,794 utterance-response pairs) for the single-turn generation model. We do not use the rewritten context as the input in the training phase, since we would like to guarantee the gain only comes from the rewriting mechanism at the inference stage.

4.2.2 Evaluation Metrics

We regard the human response as the ground truth, and use the following metrics:

Word overlap based metrics: We report BLEU score (Papineni et al., 2002) between model outputs and human references.

Embedding based metrics: As BLEU is not correlated with the human annotation perfectly, following (Liu et al., 2016), we employ embedding based metrics, **Embedding Average (Average)**, **Embedding Extrema (Extrema)**, and **Embedding Greedy (Greedy)** to evaluate results. The word2vec is trained on the training data set, whose dimension is 200.

Diversity: We evaluate the response diversity based on the ratios of distinct unigrams and bigrams in generated responses, denoted as Distinct-1 and Distinct-2 (Li et al., 2016a).

Human Annotation: We ask three native speakers to annotate the quality of generated responses. We compare the quality of our model with HRED and S2SA. We conduct 5-scale rating: +3, +2, +1, 0 and -1. +3: the response is natural, informative and relevant with context; +2: the response is natural, informative, but might not be relevant enough; +1: the response is natural, but might not be informative and relevant enough (e.g., I don't know); 0: The response makes no sense, irrelevant, or grammatically broken; -1:

	BLEU-1	BLEU-2	BLEU-3	Average	Extrema	Greedy	Distinct-1	Distinct-2
S2SA	5.72	2.80	1.37	11.14	8.58	13.15	25.55	58.89
HRED	10.10	5.53	2.75	27.45	21.71	27.43	15.22	32.19
Dynamic	7.05	3.54	1.75	17.77	14.20	18.94	6.22	15.08
Static	9.31	5.01	2.77	21.32	17.36	22.49	6.59	17.31
CRN	13.26	8.43	4.64	32.31	26.97	33.59	31.48	67.02
CRN + RL	13.63	8.69	4.88	33.14	27.49	34.68	31.42	65.10

Table 2: Automatic evaluation results.

	3	2	1	0	-1	avg
S2SA	14.48%	48.56%	31.01%	5.48%	0.46%	1.72
HRED	13.28%	16.90%	65.00%	3.99%	0.46%	1.40
CRN+S2SA	37.05%	31.01%	25.07%	6.04%	0.46%	1.99
CRN+S2SA+RL	42.43%	29.25%	15.04%	12.63%	0.46%	2.02

Table 3: The distribution of human evaluation in response generation model.

The response or utterances cannot be understood.

4.2.3 Evaluation Results

Table 2 presents the automatic evaluation results, showing that our models outperform baselines on relevance and diversity. Table 3 gives the human annotation results, which also demonstrates the superiority of our models. Our models significantly improve response diversity, mainly because the rewritten sentence contains rich information that is capable of guiding the model to generate a specific output. After reinforcement learning our model promotes on BLEU and embedding metrics, it is because reinforcement learning can build the connection between the utterance-rewrite model with the response generation model for exploring better rewritten-utterance. But our model drops a little on the diversity metrics after reinforcement learning, this owes to the fact that the reward is biased to relevance rather than diversity. The similar phenomenon can be observed in the comparison of HRED and S2SA, which means that although relevance can increase by considering context information, general responses become more frequently concurrently.

Table 3 presents the distribution of score in human evaluation, we can observe that most of the responses generated by HRED and S2SA get 1 or 2 in human evaluation, while most of the responses generated by our model can get 2 or 3. It proves that our model can reduce noisy from context and construct an informative utterance to generate high-quality response. However, after reinforcement learning our model gets more 0 and 3 score, that is because after reinforcement learning, our model becomes unstable and prefers to extract

more words from context. The score of one candidate will increase or decrease a lot if useful keywords or wrong keywords are inserted into the last utterance, respectively. In fact, more utterances are rewritten better after reinforcement learning so the average evaluation score improves.

4.3 Multi-turn Response Selection

We also evaluate the multi-turn response selection task of retrieval-based chatbots, which aims to select proper responses from a candidate pool by considering the context. We use the Douban Conversation Corpus released by Wu et al. (2017), which is created by crawling a popular Chinese forum, the Douban Group⁵, covering various topics. Its training set contains 0.5 million conversational sessions, and the validation set contains 50,000 sessions. The negative instances in both sets are randomly sampling with a 1:1 positive-negative ratio. The test set contains 1000 conversation contexts, and each context has 10 response candidates with human annotations.

We split the last utterance from each context in the training data, and forms 0.5 million of (q, r) pairs. Subsequently, we train a single-turn Deep Attention Matching Network (Zhou et al., 2018) consuming the pair as an input, which is denoted as DAM_{single} . The DAM_{single} model is treated as a rank model in Section 3.2.2 and a reward function in Section 3.3.2. In the testing stage, we use the CRN and the DAM_{single} to assign a score for each candidate. Notably, the original DAM takes a context-response pair as an input, which is set as a baseline method. The parameters of the DAM is the same as its original paper.

⁵<https://www.douban.com/group/explore>

	MAP	MRR	P@1	R ₁₀ @1	R ₁₀ @2	R ₁₀ @5
TF-IDF (Lowe et al., 2015)	0.331	0.359	0.180	0.096	0.172	0.405
RNN (Lowe et al., 2015)	0.390	0.422	0.208	0.118	0.223	0.589
CNN (Lowe et al., 2015)	0.417	0.440	0.226	0.121	0.252	0.647
LSTM (Lowe et al., 2015)	0.485	0.527	0.320	0.187	0.343	0.720
BiLSTM (Lowe et al., 2015)	0.479	0.514	0.313	0.184	0.330	0.716
Multi-View (Zhou et al., 2016)	0.505	0.543	0.342	0.202	0.350	0.729
DL2R (Yan et al., 2016)	0.488	0.527	0.330	0.193	0.342	0.705
MV-LSTM (Pang et al., 2016)	0.498	0.538	0.348	0.202	0.351	0.710
Match-LSTM (Wang and Jiang, 2017)	0.500	0.537	0.345	0.202	0.348	0.720
Attentive-LSTM (Tan et al., 2016)	0.495	0.523	0.331	0.192	0.328	0.718
SMN (Wu et al., 2017)	0.529	0.569	0.397	0.233	0.396	0.724
DAM (Zhou et al., 2018)	0.550	0.601	0.427	0.254	0.410	0.757
DAM _{single}	0.543	0.592	0.414	0.255	0.427	0.725
CRN + DAM _{single}	0.548	0.603	0.428	0.262	0.439	0.727
CRN + RL + DAM _{single}	0.552	0.605	0.431	0.267	0.445	0.729

Table 4: Evaluation results on multi-turn response selection. The numbers of baselines are copied from (Zhou et al., 2018)

	Win	Loss	Tie
CRN vs baseline	51.3%	29.7%	19.0%
CRN + RL vs baseline	35.7%	21.8%	42.5%

Table 5: The result of end-to-end response selection subjective evaluation.

4.3.1 Evaluation Results

Table 4 shows the response selection performances of different methods. We can see that our model achieves a comparable performance with state-of-the-art DAM model, but only consuming a rewritten utterance rather than the whole context. This indicates that our model is able to recognize important content in context and generate a self-contained sentence. This argument is also verified by 1 point promotion compared with DAM_{single} which only uses the last utterance as an input. Additionally, DAM_{single} just underperforms DAM 1 point, meaning that the last utterance is very important for response selection. It supports our assumption that the last utterance is important which is a good prototype for context rewriting.

4.4 End-to-End Multi-turn Response Selection

In practice, a retrieval-based chatbot first retrieves a number of response candidates from an index, then re-ranks the candidates with the aforementioned response selection methods. Previous works pay little attention to the retrieval stage, which just appends some keywords to the last utterance to collect candidates (Wu et al., 2017).

Because our model is able to rewrite context and generate a self-contained sentence. We expect it could retrieve better candidates at the first step, benefiting to the end-to-end performance.

Since it is hard to evaluate the retrieval-stage, we evaluate the end-to-end response selection performance. Specifically, we first rewrite the contexts in the test set with CRN, and then retrieve 10 candidates with the rewritten context from the index⁶. DAM_{single} is employed to compute relevance scores with the rewritten utterance and the candidates. The candidate with the top score is selected as the final response. The baseline model appends keywords from context to the last utterance for retrieval and use the original DAM with all context as the input to select final response.

We recruit three annotators to do a side-by-side evaluation, and the model outputs are shuffled before human evaluation. The majority of the three judgments are selected as a result. If both outputs are hard to distinguish, we choose Tie as the result.

4.4.1 Evaluation Results

We list the side-by-side evaluation results in Table 5. Human annotators prefer the outputs of our model. On account of the reranking modules are comparable, we can infer that the gain comes from the better retrieval candidates. However, reinforcement learning does not have a positive effect on this task. We find our reinforced model becomes more conservative, it tends to generate shorter rewritten utterance than our pre-training model. That may be beneficial for response re-rank, but if wrong keywords or noise words are extracted from context. It will reduce the quality of retrieved candidates, leading to an undesired end-to-end result.

⁶The authors (Wu et al., 2017) share their index data with us

Context	我十九号昆明飞厦门 I will fly from Kunming to Xiamen in 19th	上了豆瓣更无聊 Douban is so boring	你5点半下班? Do you go off duty in 5:30?
Last Utterance	我也想去 I want to go, too	好像是这样的 Yes	是的呀, 坐上班车了都 Yes, I am in bus now
Rewritten-Utterance	昆明飞厦门我也想去 I want to fly from Kunming to Xiamen, too	豆瓣好像是这样的 Yes, Douban is.	是的呀, 坐上班车了都下班? Yes, I am now in bus off duty?
Rewritten-Utterance+RL	我也想去厦门 I want to go to Xiamen, too	好像是这样的豆瓣更无聊 Yes, Douban is so boring	是的呀5点半下班, 坐上班车了都 Yes, go off duty in 5:30, I am now in bus
S2SA	那就出发吧 Let's go	你是双子 You are Gemini	5点半 5:30
HRED	我也是 Me too	是啊 Yes	好吧 Okay
Our Model	昆明大理丽江 Kunming, Dali, Lijiang	豆瓣毁一生 Douban can ruin whole life.	下班了吗 Do you go off?
Our Model+RL	厦门欢迎你 Welcome to Xiamen	无聊到爆 I'm bored to death	我也是坐班车 I am taking the bus, too

Table 6: The examples of end-to-end response generation.

4.5 Case Study

We list the generated examples of our models and baseline models for End-to-end Generation Chatbot and Retrieval Chatbot. Because the submission space is quite limited, we put the case study of Retrieval Chatbot in the Supplementary Material.

4.5.1 End-to-end Generation Chatbots

Table 6 presents the generated examples of our models and baselines, our model can extract the keywords from the context which is helpful to generate an informative response, but the HRED model often generates safe responses like “*Me too*” or “*Yes*”. It is because the input information from context and last utterance contain so much noise, some of the context words are useless for the last utterance to generate responses. Our model can extract important keywords from noisy context and insert them into the last utterance, it is not only easy to control and explain in a chat-bot system, but also transmit useful information directly to last utterance. The input of S2SA model is the last utterance, so it can generate diverse response due to less noise, but its relevancy with context is low. Our model succeeds fusing advantage from both models and get a significant promotion. Comparing the generated responses by our pre-training model and reinforced model, the rewritten-utterance inferred by

our pre-training model may be more informative, but the final generated response may be unrelated to context and last utterance. It is because reinforcement learning can build the connection between the utterance-rewrite model with response generation model for exploring better rewritten-utterance. A better rewritten-utterance should be helpful to generate a context-related response, Too much information inserted will add noise and too little will be useless.

5 Conclusion

This paper investigates context modeling in open domain conversation. It proposes an unsupervised context rewriting model, which benefits to candidate retrieval and controllable conversation. Empirical results show that the rewriting contexts are similar to human references, and the rewriting process is able to improve the performance of multi-turn response selection, multi-turn response generation, and end-to-end retrieval chatbots.

Acknowledgement

We are thankful to Yue Liu, Sawyer Zeng and Libin Shi for their supportive work. We also gratefully thank the anonymous reviewers for their insightful comments.

References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.
- Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O. K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.
- Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional neural network architectures for matching natural language sentences. In *Advances in Neural Information Processing Systems*, pages 2042–2050.
- Quoc V. Le Ilya Sutskever, Oriol Vinyals. 2014. Sequence to sequence learning with neural networks. In *Nips*, pages 3104–3112.
- Zongcheng Ji, Zhengdong Lu, and Hang Li. 2014. An information retrieval approach to short text conversation. *ArXiv*, abs/1408.6988.
- Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A diversity-promoting objective function for neural conversation models. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 110–119.
- Jiwei Li, Michel Galley, Chris Brockett, Georgios P. Spithourakis, Jianfeng Gao, and William B. Dolan. 2016b. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.
- Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016c. Deep reinforcement learning for dialogue generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1192–1202.
- Chia-Wei Liu, Ryan Lowe, Iulian Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2122–2132.
- Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of the SIGDIAL 2015 Conference, The 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 2-4 September 2015, Prague, Czech Republic*, pages 285–294.
- Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2016. Text matching as image recognition. In *Proceedings of the thirtieth AAAI Conference on Artificial Intelligence*, pages 2793–2799.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318. Association for Computational Linguistics.
- Alan Ritter, Colin Cherry, and William B Dolan. 2011. Data-driven response generation in social media. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 583–593. Association for Computational Linguistics.
- Iulian Vlad Serban, Tim Klinger, Gerald Tesauro, Karthik Talamadupula, Bowen Zhou, Yoshua Bengio, and Aaron C. Courville. 2017a. Multiresolution recurrent neural networks: An application to dialogue response generation. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.*, pages 3288–3294.
- Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C. Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA.*, pages 3776–3784.
- Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron C Courville, and Yoshua Bengio. 2017b. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI*, pages 3295–3301.
- Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. In *ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1577–1586.
- Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 196–205.
- Hui Su, Xiaoyu Shen, Rongzhi Zhang, Fei Sun, Pengwei Hu, Cheng Niu, and Jie Zhou. 2019. Improving multi-turn dialogue modelling with utterance

- rewriter. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.
- Richard S Sutton, Andrew G Barto, et al. 1998. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge.
- Ming Tan, Cícero Nogueira dos Santos, Bing Xiang, and Bowen Zhou. 2016. Improved representation learning for question answer matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.
- Oriol Vinyals and Quoc V. Le. 2015. A neural conversational model. *CoRR*, abs/1506.05869.
- Hao Wang, Zhengdong Lu, Hang Li, and Enhong Chen. 2013. A dataset for research on short-text conversations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 935–945.
- Mingxuan Wang, Zhengdong Lu, Hang Li, and Qun Liu. 2015. Syntax-based deep matching of short texts. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.
- Shuohang Wang and Jing Jiang. 2017. Machine comprehension using match-lstm and answer pointer. *ICLR*.
- Lijun Wu, Fei Tian, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2018a. A study of reinforcement learning for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018, Brussels, Belgium, November 2-4, 2018*.
- Yu Wu, Furu Wei, Shaohan Huang, Yunli Wang, Zhoujun Li, and Ming Zhou. 2018b. Response generation by context-aware prototype editing. In *AAAI*.
- Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. 2017. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 496–505.
- Chen Xing, Wei Wu, Yu Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. 2017. Topic aware neural response generation. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 3351–3357.
- Chen Xing, Yu Wu, Wei Wu, Yalou Huang, and Ming Zhou. 2018. Hierarchical recurrent attention network for response generation. 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 5610–5617.
- Rui Yan, Yiping Song, and Hua Wu. 2016. Learning to respond with deep neural networks for retrieval-based human-computer conversation system. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, SIGIR 2016, Pisa, Italy, July 17-21, 2016*, pages 55–64.
- Wei-Nan Zhang, Yiming Cui, Yifa Wang, Qingfu Zhu, Lingzhi Li, Lianqiang Zhou, and Ting Liu. 2018. Context-sensitive generation of open-domain conversational responses. In *COLING*, pages 2437–2447.
- Xiangyang Zhou, Daxiang Dong, Hua Wu, Shiqi Zhao, Dianhai Yu, Hao Tian, Xuan Liu, and Rui Yan. 2016. Multi-view response selection for human-computer conversation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 372–381.
- Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. 2018. Multi-turn response selection for chatbots with deep attention matching network. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1118–1127.

Context	我7月20日才到奥克兰呢,你机票订了吗 I will arrive at Auckland in July 20th, have you book the air ticket?	那说明你很熟悉俄罗斯这个国家了啊还开贴干嘛 That means you are familiar with the country Russia, why you open new post
Last Utterance	那太遗憾了我七月二十号就结束旅途了 What a shame, I will finish my trip in July 20th	对啊你都知道 Yes, you have known that
Rewritten-Utterance	7月20日才到奥克兰那太遗憾了我七月二十号就结束旅途了 Arrive at Auckland in July 20th? What a shame, I will finish my trip in July 20th	对啊你都知道俄罗斯这个国家了 Yes, you have known the country Russia
Rewritten-Utterance-RL	机票订了那太遗憾了,我七月二十号就结束旅途了 Air ticket is booked? What a shame, I will finish my trip in July 20th	对啊你都知道国家了 Yes, you have known the country
baseline	这样啊去上学 Go to school like this?	你东北哪噶的啊 Where is your home in northeast of China
Our Model	请问你行程规划了几天 How many days have you planned your trip	俄罗斯不了解 I don't know about Russia
Our Model+RL	可惜了那你还要去吗 What a shame, then are you going there?	说得容易啊 Easier said

Table 7: The examples of end-to-end response selection.

A Model Details

A.1 Context Rewriting Model

We employ the Adam algorithm to optimize the objective function with a batch size of 200. We set the initial learning rate as 0.0004 and reduce it by half if perplexity on validation begins to increase. Both of the encoder and decoder are 1-layer GRU, the word embedding size and hidden vector of encoder GRU are 512, the hidden size of decoder GRU is 1024. We use BPE to do Chinese word segmentation because it can solve the out-of-vocabulary word problem and reduce the size of the vocabulary. The vocabulary size of input and output are 34687. Drop out mechanism is used and equals to 0.3. We use beam search in generating rewritten last utterance and response, the beam size is always 5. For reinforcement learning, λ is equals to 0.1.

A.2 Single-turn Response Generation Model

The model is pre-trained by an encoder-decoder framework with an attention mechanism, where the word embedding size is 512, the encoder is 1-layer bidirectional GRU with 1024 units in both

directions, and the decoder is 1-layer unidirectional GRU with 2048 units.

B Case Study

B.1 End-to-end Retrieval Chatbots

Table 7 presents the selected examples of our models and baselines, our model can extract the keywords from context to construct an informative utterance, which is helpful to retrieve response relative candidates from the index and select the correct response. But the baseline model retrieves candidates by concatenating extracted keywords from the context with the last utterance, the extracted keywords may be useless and even noisy, so some candidates with low correlation to utterances will be retrieved, such error will propagate to next step.

As shown in Table 7, after reinforcement learning, our model tends to generate shorter rewritten utterance than our pre-trained model. And the extracted keywords may be useless or lose important information. It will reduce the quality of retrieved candidates and the quality of the final selected response.