RAST: Domain-Robust Dialogue Rewriting as Sequence Tagging

Jie Hao[1]*, **Linfeng Song**[2]†, **Liwei Wang**[3]*, **Kun Xu**[2], **Zhaopeng Tu**[2], and **Dong Yu**[2]
[1]Florida State University, FL, USA
haoj8711@gmail.com
[2]Tencent AI Lab, Bellevue, WA, USA
{lfsong,kxkunxu,zptu,dyu}@tencent.com
[3]The Chinese University of Hong Kong
lwwang@cse.cuhk.edu.hk

Abstract

The task of dialogue rewriting aims to reconstruct the latest dialogue utterance by copying the missing content from the dialogue context. Until now, the existing models for this task suffer from the robustness issue, i.e., performances drop dramatically when testing on a different dataset. We address this robustness issue by proposing a novel sequence-tagging-based model so that the search space is significantly reduced, yet the core of this task is still well covered. As a common issue of most tagging models for text generation, the model's outputs may lack fluency. To alleviate this issue, we inject the loss signal from BLEU or GPT-2 under a REINFORCE framework. Experiments show huge improvements of our model over the current state-of-the-art systems when transferring to another dataset.

1 Introduction

Recent years have witnessed increasing attention in conversation-based tasks, such as conversational question answering (Choi et al., 2018; Reddy et al., 2019; Sun et al., 2019), dialogue response generation (Li et al., 2017; Zhang et al., 2018; Wu et al., 2019; Zhou et al., 2020), dialogue state tracking (Eric et al., 2020; Zeng et al., 2021) and dialogue understanding (Xu et al., 2021; Song et al., 2021; Yu et al., 2020), mainly due to increasing commercial demands. However, current models still face tremendous challenges in representing multi-turn dialogues, due to the frequent omission (a.k.a. ellipsis) and coreference that people naturally use in conversations for brevity. Specifically, recent work (Su et al., 2019) has shown that ellipsis and coreference can exist in more than 70% of dialogue utterances. To tackle this problem, people have proposed coreference resolution and zero pronoun recovery. But, the state-of-the-art performances on these tasks are still far from satisfactory, not to mention their uncovered situations, such as when a whole verb phrase is omitted.

Turn	Utterance with Translation
u_1	上海最近天气怎么样? (How is the recent weather in Shanghai?)
u_2	最近经常阴天下雨。 (It is always raining recently.)
u_3	冬天就是这样。 (Winter is like this.)
u_3'	上海冬天就是经常阴天下雨。 (It is always raining in winter Shanghai.)

Table 1: An example dialogue including the context utterances (u_1 and u_2), the latest utterance (u_3) and the rewritten utterance (u_3').

Recently, the task of dialogue utterance rewriting (Su et al., 2019; Pan et al., 2019; Elgohary et al., 2019) was proposed as for explicitly representing multi-turn dialogues. The task aims to reconstruct the latest dialogue utterance into a new utterance that is semantically equivalent to the original one and can be understood without referring to the context. In another point of view, it integrates the recovering of both coreference and omission. As shown in Table 1, the incomplete utterance u_3 omit "上海 (Shanghai)" and refer "经常阴天下雨 (always raining)" with pronoun "这样 (this)". By explicitly rewriting the dropped information into the latest utterance, the downstream dialogue model only needs to take the last utterance. Thus the burden on long-range reasoning can be largely relieved.

Most previous efforts (Su et al., 2019; Pan et al., 2019; Elgohary et al., 2019; Xu et al., 2020) consider this task as a standard text-generation problem, adopting a sequence-to-sequence model with a copy mechanism (Gulcehre et al., 2016; Gu et al., 2016; See et al., 2017). They have demonstrated almost *ready-to-use* performances on the test set from the same data source as the training set. However, they are not robust, as our experiments show

*Work done while J. Hao was interning and L. Wang was working at Tencent AI Lab.
†Corresponding author.

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 4913–4924
November 7–11, 2021.

that their performances can drop dramatically (by roughly 33 BLEU4 (Papineni et al., 2002) points and 44 percent of exact match) on another test set created from a different data source (not necessarily from a totally different domain). We argue that it may not be the best practice to model utterance rewriting as standard text generation. One main reason is that text generation introduces an overly large search space, while a rewriting output (e.g., u_3' in Table 1) always keeps the core semantic meaning of its input (e.g., u_3). Besides, exposure bias (Wiseman and Rush, 2016) can further exacerbate the problem for test cases that are not similar to the training set, resulting in outputs that convey different semantic meanings from the inputs.

In this paper, we propose a novel solution that treats utterance rewriting as multi-task sequence tagging. In particular, for each input word, we decide whether to *delete* it or not, and at the same time, we choose what span from the dialogue context need to be inserted to the front of the current word. In this way, our solution enjoys a far smaller search space than the generation based approaches.

Since our model does not directly take features from the word-to-word interactions of its output utterances, this may cause the lack of fluency. To encourage more fluent outputs, we propose to inject additional supervisions from two popular metrics, i.e., sentence-level BLEU (Chen and Cherry, 2014) and the perplexity of a pretrained GPT-2 (Radford et al., 2019) model, using the framework of "REINFORCE with a baseline" (Williams, 1992). Sentence-level BLEU is computationally efficient, but it requires references and thus may only provide domain-specific knowledge. Conversely, the perplexity by GPT-2 is reference-free, giving more guidance on open-domain scenarios benefiting from the large-scale pretraining.

Experiments on two dialogue rewriting benchmarks show that our model can give huge improvements (14.6 in BLEU4 score and 18.9 percent of exact match) over the current state-of-the-art model for cross-dataset evaluation. More analysis shows that the outputs of our model keep more semantic information from the inputs. Our code is available at `https://github.com/freesunshine0316/RaST-plus`.

2 Related Work

Initial efforts (Su et al., 2019; Elgohary et al., 2019) treat dialogue utterance rewriting as a standard text generation problem, adopting sequence-to-sequence models with copy mechanism to tackle this problem. Later work (Pan et al., 2019; Zhou et al., 2019; Huang et al., 2021) explores task-specific features for additional gains in performance. For instance, Pan et al. (2019) adopts a pipeline-based method, where all context words that need to be inserted during rewriting are identified in the first step. The second step adopts a pointer generator that takes the outputs of the first step as additional features to produce the output.

Xu et al. (2020) train a model of semantic role labeling (SRL) to highlight the core meaning (e.g., who did what to whom) of each input dialogue to prevent their rewriter from violating this information. To obtain an accurate SRL model on dialogues, they manually annotate SRL information for more than 27,000 dialogue turns, which is time-consuming and costly. Liu et al. (2020) casts this task into a semantic segmentation problem, a major task in computer vision. In particular, their model generates a word-level matrix, which contains the operations of substitution and insertion, for each original utterance. They adopt a heavy model that takes 10 convolution layers in addition to the BERT encoder. None of the existing efforts mention the robustness issue, a critical aspect for the usability of this task. Besides, they only compare performances under automatic metrics (e.g., BLEU). We take the first step to address this severe robustness issue, and we adopt multiple measures for comprehensive evaluation. Besides, we propose a novel model based on sequence tagging for solving this task, and our model takes a much smaller search space than previous models.

Sequence tagging for text generation Given the intrinsic nature of typical text-generation problems (e.g., machine translation), i.e. (1) the number of predictions cannot be determined by inputs, and (2) the candidate space for each prediction is usually very large, sequence tagging is not commonly adopted on text-generation tasks. Recently, Malmi et al. (2019) proposed a model based on sequence tagging for sentence fusion and sentence splitting, and they show that their model outperforms a vanilla sequence-to-sequence baseline. In particular, their model can decide whether to keep or delete each input word and what phrase needs to be inserted in front of it. As a result, they have to extract a large phrase table from the training data, causing inevitable computation for choosing

phrases from the table. Their approach also faces the issue on unseen cases where their phrase table has limited coverage. Though we also convert our original problem into a multi-task tagging problem, we predict what span to be inserted, avoiding the issues caused by using a phrase table. Besides, we study injecting richer supervision signals to improve the fluency of outputs, which is a common issue for tagging based approaches on text generation, as they do not directly model word-to-word dependencies. Finally, we are the first to apply sequence tagging on dialogue rewriting, showing much better performances than those of BERT-based strong baselines.

3 Baseline: TRANS-PG+BERT

Our baseline consists of a BERT (Devlin et al., 2019) encoder and a Transformer (Vaswani et al., 2017) decoder with a copy mechanism. Given input tokens $X = (x_1, \ldots, x_N)$ that is the concatenation of the current dialogue context $c = (u_1, \ldots, u_{i-1})$ and the latest utterance u_i, the BERT encoder is firstly adopted to represent the input with contextualized embeddings:

$$\boldsymbol{E} = \boldsymbol{e}_1, \ldots, \boldsymbol{e}_N = \text{BERT}(x_1, \ldots, x_N). \quad (1)$$

Next, the Transformer decoder with copy mechanism is adopted to generate a rewriting output $u' = (y_1, \ldots, y_M)$ one token at a time:

$$p(y_t|y_{<t}, X) = \theta_t p_t^{vocab} + (1-\theta_t) p_t^{attn} \quad (2)$$

$$p_t^{attn}, \boldsymbol{s}_t = \text{TransDecoder}(y_{<t}, \boldsymbol{E}) \quad (3)$$

$$p_t^{vocab} = \text{Softmax}(\text{Linear}(\boldsymbol{s}_t)) \quad (4)$$

where *TransDecoder* is the Transformer decoder that returns the attention probability distribution p_t^{attn} over the encoder states $\boldsymbol{E}$ and the latest decoder state $\boldsymbol{s}_t$ for each step t. Following See et al. (2017), the generation probability θ_t for timestep t is calculated from the weighted sum for the encoder-decoder cross attention distribution and the encoder hidden states.

$$\theta_t = \sigma(\boldsymbol{w}^\intercal \sum_{n \in [1..N]} (p_t^{attn}[n] \cdot \boldsymbol{e}_n)) \quad (5)$$

where $\boldsymbol{w}$ represents the model parameter. In this way, the copy mechanism encourages copying words from the input tokens. The TRANS-PG baseline is trained with standard cross-entropy loss:

$$\mathcal{L}_{gen} = -\sum_{t \in [1..M]} \log p(y_t|y_{<t}, X; \boldsymbol{\theta}) \quad (6)$$

where $\boldsymbol{\theta}$ represents all model parameters.

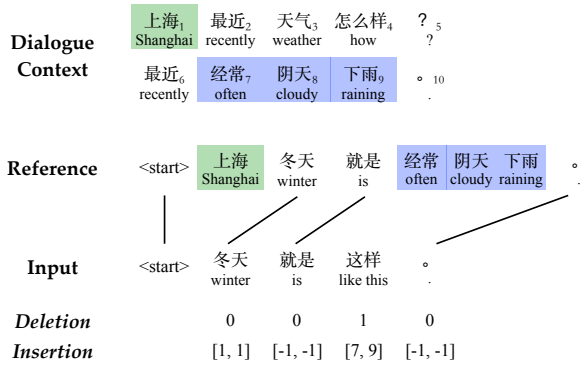

Figure 1: An example of annotating the *deletion* and *insertion* tags based on the word alignment between the input and reference utterances.

4 RAST: Rewriting as Sequence Tagging

In this section, we describe how to convert the dialogue rewriting task into a multi-task sequence tagging problem.

Task description Our analysis shows that dialogue rewriting mainly handles two linguistic phenomena: coreference and omission. To recover a coreference, it has to replace a pronoun in the current utterance with the phrase it refers to in the dialogue context. To recall an omission, it needs to insert the corresponding phrase into the omission position. Accordingly, we cast the dialogue rewriting as a sequence tagging task by introducing two types of tags for each word x_n:

- *Deletion* $\in \{0, 1\}$: the word x_n is deleted (i.e. 1) or not (i.e. 0);
- *Insertion*:[*start*, *end*]: a phrase ranging the span [*start*, *end*] in the dialogue context is inserted in front of the word x_n. If no phrase is inserted, the span is [-1, -1].

Recovering a coreference corresponds to the operation {Deletion:1, Insertion:[*start*, *end*]}, and recalling an omission corresponds to the operation {Deletion:0, Insertion:[*start*, *end*]}, where [*start*, *end*] denotes the corresponding phrase in the dialogue context. For the other words without any change, the operation is {Deletion:0, Insertion:[-1, -1]}. Figure 1 shows an example, where the word "这样 (like this)" corresponds to a coreference, and the word "冬天 (winter)" corresponds to an omission in front of it.[1]

[1]We use word-based annotations for easier visualization, while character-based annotations are adopted in reality.

Constructing annotated data The gold tags for dialogue utterance rewriting are not naturally available. In response to this problem, we construct the annotated data based on the alignment between the input and reference utterances. Specifically, we employ the *longest common sub-sequence* (LCS)[2] algorithm to generate the word alignments between the input utterance u_i and the reference utterance u_i' for each instance (the black lines in Figure 1). The LCS algorithm is based on dynamic programming, which takes a time complexity of $\mathcal{O}(|u_i| \times |u_i'|)$. For the words in the reference utterance that are not aligned, we search them from the dialogue context and obtain their span (e.g. the words in color highlighting).

Given the aligned instance, we construct the annotation tags by traversing the alignments in a left-to-right manner and comparing each alignment with its preceding one[3] under the following rules:

R1. If the two alignments are adjacent in both utterances (e.g. "就是 (is)"), there is no change for the current word, which is assigned the tags {Deletion:0, Insertion:[-1, -1]}.

R2. If the two alignments are only adjacent in the input utterance (e.g. "冬天 (winter)"), this generally corresponds to an omission. We insert the reference words between the two alignments (i.e. "上海 (Shanghai)") in front of the current input word. Accordingly, we assign the current word "冬天 (winter)" the tags {Deletion:0, Insertion:[1, 1]}.

R3. If the two alignments are only adjacent in the reference utterance, we simply delete the input words between the two alignments, and assign them the tags {Deletion:1, Insertion:[-1, -1]}. This situation is *rare* in the task of dialogue utterance rewriting.

R4. If the two alignments are not adjacent in either utterance (e.g. "。 (.)"), this generally corresponds to a coreference that requires a replacement. We first delete the input words between the two alignments (i.e. "这样 (like this)"), then insert the corresponding target phrase (i.e. "经常 阴天 下雨 (always raining)") in front of the left- most deleted input word (i.e. "这样 (like this)"). Accordingly, we assign the left-most deleted word "这样 (like this)" the tags {Deletion:1, Insertion:[7, 9]} (i.e. *replacement*), and assign the current word "。 (.)" the tags {Deletion:0, Insertion:[-1, -1]} (i.e. *no change*). If there exist more than one deleted words, we assign the other words beyond the left-most deleted word the tags {Deletion:1, Insertion:[-1, -1]} (i.e. *deletion*).

[2] https://en.wikipedia.org/wiki/Longest_common_subsequence_problem

[3] We insert a special flag "<start>" at the beginning of both utterances for processing the first alignment.

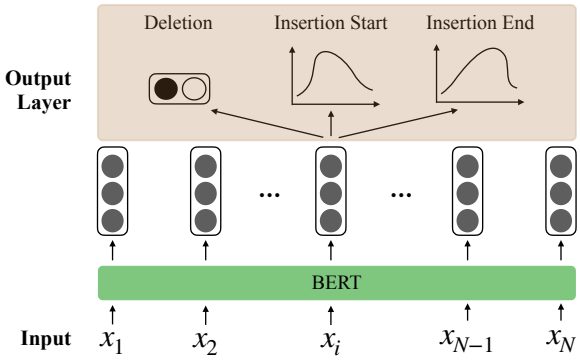

Figure 2: Our model architecture. For fair comparison, it takes the same encoder architecture as the baseline.

Both rules #2 and #4 require finding phrases from the dialogue context, as highlighted in color in Figure 1. If there are multiple candidates in the dialogue context, we choose the one that is closest to the input utterance to avoid long-range dependency. If no candidate can be found, we consider such instances cannot be covered by our approach. In the REWRITE and RESTORATION datasets used in this work, we respectively found that 6.0% and 6.5% instances are not covered, and deleted them from the datasets.

5 Approach

5.1 Model Architecture

Figure 2 shows the architecture of our model. For a fair comparison, it takes the same BERT-based encoder (Equation 1) as the baseline to represent each input. For simplicity, we directly apply classifiers to predict the corresponding tags for each input word. In particular, to determine whether each word x_n in the current utterance u_i should be kept or deleted, we use a binary classifier:

$$p(d_n|X, n) = \text{Softmax}(\boldsymbol{W}_d \boldsymbol{e}_n + \boldsymbol{b}_d) \quad (7)$$

where $\boldsymbol{W}_d$ and $\boldsymbol{b}_d$ are learnable parameters, d_n is the binary classification result, and $\boldsymbol{e}_n$ is the BERT embedding for x_n.

Moreover, we cast span prediction as machine reading comprehension (MRC) (Rajpurkar et al., 2016), where a predicted span corresponds to an MRC target answer. For each input token $x_n \in u_i$, we follow the previous work on MRC to predict the start position s_n^{st} and end position s_n^{ed} for the target span s_n, performing separate self-attention mechanisms for them:

$$p(s_n^{st}|X, n) = \text{Attn}_{start}(\boldsymbol{E}, \boldsymbol{e}_n) \quad (8)$$

$$p(s_n^{ed}|X, n) = \text{Attn}_{end}(\boldsymbol{E}, \boldsymbol{e}_n) \quad (9)$$

where Attn_{start} and Attn_{end} are the self-attention layers for predicting the start and end positions of a span. We use the standard additive attention mechanism (Bahdanau et al., 2014) to perform the attention function. The probability for the whole span s_n is:

$$p(s_n|X, n) = p(s_n^{st}|X, n)p(s_n^{ed}|X, n) \quad (10)$$

where s_n^{st} is *no greater than* s_n^{ed}.

Given an example of (c, u_i) pair and $X = [c; u_i]$, the overall loss function for the multi-task sequence labeling is defined as the standard cross-entropy loss over gold tags:

$$\mathcal{L}_{tagging} = -\sum_{x_n \in u_i} \Big(\log p(d_n|X, n) + \log p(s_n|X, n) \Big) \quad (11)$$

where the terms are defined in Equation 7 and 10.

5.2 Enhancing Fluency with Additional Supervision

By converting dialogue utterance rewriting into a sequence tagging task, our model enjoys better efficiency and lower search space. However, a potential side effect is that our outputs may lack fluency, because our approach does not directly model word-to-word dependencies.

We explore sentence-level BLEU (Chen and Cherry, 2014) and GPT-2 (Radford et al., 2019) as additional training signal to improve the fluency of our generated outputs, adopting the framework of "REINFORCE with a baseline" to inject these supervision signals. For more detail, we first generate two candidate sentences: one is by *sampling* the tags at each position of the input utterance according to the distributions in Equations 7 and 10, the other is by greedily choosing the *model-considered best* tags. Next, the RL objective for sample (c, u_i) is calculated by:

$$\mathcal{L}_{rl} = (r(\hat{u}_i^g, u_i) - r(\hat{u}_i^s, u_i)) \log p(\hat{u}_i^s|X) \quad (12)$$

where $\hat{u}_i^s$ and $\hat{u}_i^g$ represents the two candidate sentences by sampling and greedy "argmax", respectively. $r(\cdot, \cdot)$ is the reward function, which can correspond to either sentence-level BLEU or the perplexity by the GPT-2 model. Finally, we follow previous work by combining this additional loss with the tagging loss:

$$\mathcal{L} = (1 - \lambda)\mathcal{L}_{tagging} + \lambda\mathcal{L}_{rl}, \quad (13)$$

where the λ is a constant weighting factor that is empirically set to 0.5.

Dataset	REWRITE	RESTORATION
Train/dev/test size	18k/1k/1k	194k/5k/5k
No. turns	3	6
No. con. tokens (μ, σ)	(18.63, 8.50)	(38.36, 11.71)

Table 2: Statistics of two datasets, where "No. turns" denotes number of turns for each example, and "No. con. tokens (μ, σ)" denotes the mean and standard deviation for number of tokens in the context.

6 Experiments

We study the robustness of our tagging-based model on two benchmarks for dialogue rewriting.

6.1 Setup

Datasets We conduct experiments on two popular dialogue rewriting datasets: REWRITE (Su et al., 2019) and RESTORATION (Pan et al., 2019). Both datasets are created by first crawling multi-turn dialogues from popular Chinese social media platforms, before asking human annotators to generate the rewriting result for the last turn of each dialogue. Table 2 lists some statistics of both datasets. In addition to the difference on data scale (20K vs 204K), it shows additional disparities on other aspects. For instance, the number of turns for RESTORATION is twice as much as REWRITE, and the dialogue context size for RESTORATION is larger and more variant than that of REWRITE. Besides, around 40% instances in RESTORATION do not require any changes for rewriting, while all instances require rewriting for REWRITE. These differences make transferring between the two datasets a good test bed for evaluating model robustness.

#	Model	BLEU1	BLEU2	BLEU3	BLEU4	R1	R2	R-L	EM
	Results on the REWRITE *Test Set*								
1	TRANS-PG+BERT	88.1	86.5	84.9	82.3	90.2	84.1	91.0	51.3
2	CSRL (Xu et al., 2020)	89.0	87.5	85.6	83.5	89.9	81.5	87.5	47.4
3	RUN (Liu et al., 2020)	**93.5**	**90.9**	88.2	85.5	**95.8**	**90.3**	91.3	**65.1**
4	RAST	90.6	90.2	**89.3**	**88.2**	94.0	88.9	**91.5**	64.3
5	RAST+RL-BLEU	89.9	89.6	88.7	87.7	93.7	88.7	91.2	64.4
6	RAST+RL-GPT2	89.2	88.8	87.9	86.9	93.5	88.2	90.7	63.0
	Results on the RESTORATION *Test Set (Robustness Examination)*								
7	TRANS-PG+BERT	65.8	61.4	58.5	55.3	66.9	55.1	69.8	7.4
8	CSRL (Xu et al., 2020)	70.0	66.1	63.2	60.1	67.4	55.9	67.6	8.6
9	RUN (Liu et al., 2020)	79.3	74.6	70.2	65.7	81.2	69.6	76.5	12.9
10	RAST	84.8	83.2	81.4	79.3	82.9	74.1	78.8	24.5
11	RAST+RL-BLEU	84.4	82.9	81.2	79.1	83.3	74.7	79.3	26.8
12	RAST+RL-GPT2	**85.2**	**83.8**	**82.2**	**80.3**	**84.3**	**76.0**	**80.5**	**31.8**

Table 3: Test results of all comparing models trained on the REWRITE dataset.

Model settings We implement the baseline and our model on top of a BERT-base model (Devlin et al., 2019), and we use Adam (Kingma and Ba, 2015) as the optimizer, setting the learning rate to $3e^{-5}$ as determined by a development experiment. For the reinforcement learning stage, we respectively use the sentence-level BLEU score with "Smoothing 3" (Chen and Cherry, 2014) or the perplexity score based on a Chinese GPT-2 model trained on massive dialogues (Zhang et al., 2020)[4] as the reward function. It is worth noting that the GPT-2 model is not fine-tuned during the reinforcement learning stage.

Comparing models In addition to the TRANS-PG+BERT baseline, we compare our approach with several state-of-the-art dialogue rewriting models that are also based on BERT. CSRL (Xu et al., 2020) leverages additional information on conversational semantic role labeling (CSRL) to enhance BERT representation, extra human efforts are required on CSRL annotation. RUN (Liu et al., 2020) treats this problem as semantic segmentation by predicting a word-level edit matrix for the input utterance. For fair comparison, we either run their released model or ask the authors to generate their outputs on our data.

Evaluation We use both automatic metrics and human evaluations to compare our proposed model with other approaches. For the automatic metrics, we follow previous work to use BLEU (Papineni et al., 2002), ROUGE (Lin, 2004) and the percent of sentence-level exact match (EM score).

[4]https://github.com/yangjianxin1/GPT2-chitchat

6.2 Main Results

Training on REWRITE Table 3 shows the results when all comparing models are trained on the REWRITE dataset, before evaluating on the in-domain REWRITE and the RESTORATION test data for robustness examination. On the REWRITE test set, our tagging-based models (Rows 4-6) are much better than the TRANS-PG+BERT baseline, and they can get comparable performances with RUN, the previous state-of-the-art model. RUN usually gets high numbers on BLEU1 without consistent improvements on higher-order BLEU scores. Our observation shows that it tends to insert context words into wrong places, hurting the number of matches regarding higher n-gram.

Among our tagging-based models, we find that injecting additional training signal (Row 5-6) does not help the in-domain performance, which is already very descent for practical use. The reason can be that optimizing with external rewards will dilute the main signal: the cross-entropy loss of the in-domain training data. This is especially evident on RAST+RL-GPT2, where the perplexity from an external GPT-2 model may not be fully aligned with the training data. Comparatively, sentence-level BLEU is better consistent with the main signal than GPT-2, explaining why RAST+RL-BLEU reports slightly higher in-domain numbers than RAST+RL-GPT2. Please note that our main focus is the robustness issue and that slight performance changes on in-domain data will not affect its practical use. Our observation shows that both types of rewards can improve the fluency of model outputs, especially on other non-in-domain datasets.

When switching from the in-domain REWRITE

#	Model	BLEU1	BLEU2	BLEU3	BLEU4	R1	R2	R-L	EM
	Results on the RESTORATION *Test Set*								
1	TRANS-PG+BERT	88.0	87.0	85.9	84.4	90.1	84.5	89.8	49.0
2	CSRL (Xu et al., 2020)	90.6	**89.7**	**88.6**	**87.2**	91.1	85.0	**90.0**	49.1
3	RUN (Liu et al., 2020)	**92.0**	89.1	86.4	83.6	**92.1**	**85.4**	89.5	**49.3**
4	RAST	89.7	88.8	87.6	86.1	91.1	84.2	87.8	48.7
5	RAST+RL-BLEU	90.4	**89.6**	**88.5**	**87.0**	91.2	84.3	87.9	48.8
6	RAST+RL-GPT2	89.7	88.9	87.7	86.2	90.9	84.0	87.6	47.8
	Results on the REWRITE *Test Set (Robustness Examination)*								
7	TRANS-PG+BERT	76.6	74.9	73.0	70.8	79.8	70.7	79.7	25.7
8	CSRL (Xu et al., 2020)	75.5	73.8	71.8	69.6	80.0	69.9	79.0	23.2
9	RUN (Liu et al., 2020)	80.6	76.0	71.1	65.9	84.5	73.6	80.6	27.2
10	RAST	81.2	80.0	78.2	75.9	**84.9**	**75.2**	**81.1**	28.5
11	RAST+RL-BLEU	80.9	79.6	77.8	75.5	84.8	75.1	80.7	**29.6**
12	RAST+RL-GPT2	**82.4**	**81.0**	**79.1**	**76.7**	**85.0**	74.8	80.8	**29.4**

Table 4: Test results of all comparing models trained on the RESTORATION dataset.

test set to the non-in-domain RESTORATION test set, all comparing systems (Rows 7-9) get much worse performances than the in-domain situation, where the drops are 27, 23 and 20 BLEU4 points for TRANS-PG+BERT, CSRL and RUN, respectively. Conversely, our models are much more robust regarding this change, resulting in large advantages of 14.6 points in BLEU4, 4.0 points in Rough-L and 9.2 points in exact match over RUN, the previous state-of-the-art model.

Training on RESTORATION As shown in Table 4, we also conduct experiments in the opposite direction by training all models on the RESTORATION training set to further verify the above conclusions. Similarly, our models achieve comparable performances with all comparing systems on the in-domain test set, they are more advantageous on the test set for robustness examination. The advantages (10.8 in BLEU4, 2.2 in exact match) are smaller than the previous direction, with the reason being that the RESTORATION training set is nearly 11-time larger than the REWRITE training set. This shows that our model is less data hunger than the comparing systems, and please note that data annotation is usually very costly.

For other interesting facts, the advantage of RUN on BLEU1 still does not benefit high order situations. This is consistent with the situation in Table 3, where RUN tends to insert words into wrong contexts. Comparing the two types of extra signals, sentence-level BLEU is better on the in-domain test set, while GPT-2 is better on helping the robustness on other datasets. This is intuitive as our GPT-2 model has been pretrained on massive data.

Model	Win	Tie	Loss
Ours v.s. Trans-PG+BERT	23.0%	63.2%	13.8%
Ours v.s. CSRL	28.4%	57.0%	14.6%
Ours v.s. RUN	23.6%	64.0%	12.4%

Table 5: Human evaluation on 500 randomly selected test samples from RESTORATION. "Ours" denotes the RAST+RL-GPT2 model.

6.3 Human Evaluation

In addition to these automatic metrics, we also conduct a human evaluation for each system pair to further compare their rewriting quality, and we focus on the *robustness examination* scenario. Specifically, we first train the comparing systems on REWRITE, before using them to decode 500 randomly selected test examples from RESTORATION. Finally, we ask 3 graders to choose a winner from each pair of rewriting outputs. The evaluation criteria is based on fluency and adequacy, where the adequacy mainly considers two aspects: 1) how much meaning is retained; and 2) how many coreference and omission situations are recovered. All graders agree on 88.8% cases.

As shown in Table 5, the number of winning cases for RAST+RL-GPT2 is much more than the number of losing cases when comparing with any other system. This further confirms the effectiveness of our model. Many losing cases are due to the lack of fluency caused by object fronting, while most of this type of situations are understandable by human. Some examples are discussed in Section 6.5. Our model loses the least number of samples against RUN, because RUN also lack fluency due to improper context word insertion (as mentioned in Section 6.2).

Model	REWRITE → RESTORATION			RESTORATION → REWRITE		
	Precision	Recall	F-score	Precision	Recall	F-score
TRANS-PG+BERT	25.74	24.72	25.22	41.16	36.81	38.86
CSRL	24.67	26.13	25.38	40.36	35.97	38.04
RUN	26.52	27.62	27.06	41.09	37.78	39.36
RAST+RL-GPT2	**37.40**	**36.90**	**37.15**	**41.85**	**39.41**	**40.59**

Table 6: Results by comparing the semantic roles of rewriting outputs and references on both transfer scenarios.

	Case1	Case2	Case3
Contexts (Translation)	U1: 我意见很大 (I have a lot of complaints) U2: 有意见保留 (keep it yourself if there's any)	U1: 帮我找一下西安到商洛的顺风车 (Can you help me find a free ride from Xi'an to Shangluo) U2: 哪的 (where is it)	U1: 你帮我考雅思 (Please help me on IELTS) U2: 雅思第一项考什么 (What is tested first for IELTS)
Current utterance (Translation)	U3: 不想保留 (don't want to keep it myself)	U3: 能不能找到 (can you find any)	U3: 考口语啊 (it's oral test)
Reference (Translation)	不想保留意见 (don't want to keep the complaints myself)	能不能找到西安到商洛的顺风车 (can you find any free ride from Xi'an to Shangluo)	雅思第一项考口语啊 (it's oral test for IELTS)
TRANS-PG+BERT (Translation)	不想保留 (don't want to keep to myself)	能不能找到商洛的顺风车 (can you find any free ride to Shangluo)	考口语考口语啊 (it's oral test it's oral test)
RUN (Translation)	意见不想意保留 (the complains, I don't want to meaning keep to myself)	能不能找到商洛的顺风车 (can you find any free ride to Shangluo)	雅思第一项考口语啊 (it's oral test for IELTS)
RAST+RL-GPT2 (Translation)	意见不想保留 (the complains, I don't want to keep to myself)	能不能找到西安到商洛的顺风车 (can you find any free ride from Xi'an to Shangluo)	雅思第一项考口语啊 (it's oral test for IELTS)

Table 7: Case Study, where mistakenly predicted redundant phrases and their translations are in red.

6.4 Evaluating with Semantic Role Labeling

We also compare our model with the baselines regarding the "semantic corectness" of their rewriting outputs, taking semantic role labeling (SRL) as the form of semantics on their outputs. By doing so, we can focus on comparing the core meaning, ignoring other functional words and phrases. Specifically, we choose a state-of-the-art SRL system (Che et al., 2020) to annotate rewriting outputs and references. Next, the precision, recall and F1 scores for each system are calculated by comparing the SRL results of its outputs with these of the references.

Table 6 lists the performances on both transfer (robustness examination) scenarios, where our model reports consistently higher numbers than all other systems. This indicates that our model also makes improvement regarding the core semantic meaning. The relative differences on recall, precision and F1 are also consistent with the differences under automatic metrics (Table 3, 4) and human evaluation (Table 5).

6.5 Case Study

Table 7 gives 3 test examples that indicate the representative situations we find. The first example illustrates the cases when RUN inserts context words (e.g. "意 (meaning)") into wrong places. This hurts the fluency and high-order BLEU score as mentioned in Section 6.2. The third example shows the situations when the TRANS-PG+BERT baseline messes up by word repeating (e.g. "考口语考口语 (it's oral test it's oral test)"). This is a common situation for generation-based models, especially on unseen data samples. Conversely, this situation rarely happens to our model, as it is based on sequence tagging. Lastly, the second example corresponds to the situation of referring to a complex concept (e.g. "西安到商洛的顺风车 (a free ride from Xi'an to Shangluo)"). For these cases, it is easier for our model to get the correct span. This is because our model directly predicts the span boundaries, thus it has a smaller search space than other previous approaches, like generating the concept word by word.

6.6 Evaluation on Uncovered Examples

As mentioned earlier, a few examples may not be covered by our model that treats rewriting as sequence tagging. To get a more comprehensive evaluation, we further compare the TRANS-PG+BERT baseline and our model on the *uncovered* test examples of both REWRITE and RESTORATION datasets.

Model	BLEU4	R-L	EM
TRANS-PG+BERT	56.6	67.6	0.6
RAST+RL-GPT2	56.4	70.7	0.0

Table 8: Results on the combination of the *uncovered* test examples from both datasets.

Table 8 compares their performances on the transfer scenario. Comparing with the baseline, our model is comparable on precision (indicated by BLEU4) and is better at content recall (indicated by Rough-L). Regarding EM, our model achieves none, because all testing examples are *not covered* by our model. On the other hand, the EM score for the baseline is also close to 0.0. Our investigation finds that most of these examples are very challenging, we list some examples in the Appendix.

7 Conclusion

In this paper, we addressed the robustness issue of dialogue utterance rewriting, which is crucial for its usability on real applications. We proposed a novel tagging-based approach that results in a significantly smaller search space than the existing methods on this task, and we introduced additional supervision (e.g. by GPT-2) to improve the fluency of model outputs. Experiments with automatic metrics, human evaluation and semantic matching show that our model is much more robust than the previous state-of-the-art system without sacrificing its in-domain performances.

Future work includes evaluating this tagging framework on other English benchmarks, such as SMCalFlow (Andreas et al., 2020) and TreeDST (Cheng et al., 2020).

References

Jacob Andreas, John Bufe, David Burkett, Charles Chen, Josh Clausman, Jean Crawford, Kate Crim, Jordan DeLoach, Leah Dorner, Jason Eisner, et al. 2020. Task-oriented dialogue as dataflow synthesis. *Transactions of the Association for Computational Linguistics*, 8:556–571.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Wanxiang Che, Yunlong Feng, Libo Qin, and Ting Liu. 2020. N-ltp: A open-source neural chinese language technology platform with pretrained models. *arXiv preprint arXiv:2009.11616*.

Boxing Chen and Colin Cherry. 2014. A systematic comparison of smoothing techniques for sentence-level bleu. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 362–367.

Jianpeng Cheng, Devang Agrawal, Héctor Martínez Alonso, Shruti Bhargava, Joris Driesen, Federico Flego, Dain Kaplan, Dimitri Kartsaklis, Lin Li, Dhivya Piraviperumal, et al. 2020. Conversational semantic parsing for dialog state tracking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8107–8117.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau 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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

Ahmed Elgohary, Denis Peskov, and Jordan Boyd-Graber. 2019. Can you unpack that? learning to rewrite questions-in-context. 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 5920–5926.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, Adarsh Kumar, Anuj Goyal, Peter Ku, and Dilek Hakkani-Tur. 2020. Multiwoz 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 422–428.

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 (Volume 1: Long Papers)*, pages 1631–1640.

Caglar Gulcehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 140–149.

Mengzuo Huang, Feng Li, Wuhe Zou, and Weidong Zhang. 2021. Sarg: A novel semi autoregressive generator for multi-turn incomplete utterance restoration. *AAAI*.

Diederick P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. Dailydialog: A manually labelled multi-turn dialogue dataset. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Volume 1: Long Papers*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Qian Liu, Bei Chen, Jian-Guang Lou, Bin Zhou, and Dongmei Zhang. 2020. Incomplete utterance rewriting as semantic segmentation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

Eric Malmi, Sebastian Krause, Sascha Rothe, Daniil Mirylenka, and Aliaksei Severyn. 2019. Encode, tag, realize: High-precision text editing. 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 5057–5068.

Zhufeng Pan, Kun Bai, Yan Wang, Lianqiang Zhou, and Xiaojiang Liu. 2019. Improving open-domain dialogue systems via multi-turn incomplete utterance restoration. 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 1824–1833, Hong Kong, China.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

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*.

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.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

Linfeng Song, Chunlei Xin, Shaopeng Lai, Ante Wang, Jinsong Su, and Kun Xu. 2021. CASA: Conversational aspect sentiment analysisfor dialogue understanding. *Journal of Artificial Intelligence Research*.

Hui Su, Xiaoyu Shen, Rongzhi Zhang, Fei Sun, Pengwei Hu, Cheng Niu, and Jie Zhou. 2019. Improving multi-turn dialogue modelling with utterance ReWriter. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 22–31, Florence, Italy.

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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30:5998–6008.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.

Sam Wiseman and Alexander M Rush. 2016. Sequence-to-sequence learning as beam-search optimization. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1296–1306.

Wenquan Wu, Zhen Guo, Xiangyang Zhou, Hua Wu, Xiyuan Zhang, Rongzhong Lian, and Haifeng Wang. 2019. Proactive human-machine conversation with explicit conversation goal. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3794–3804.

Kun Xu, Haochen Tan, Linfeng Song, Han Wu, Haisong Zhang, Linqi Song, and Dong Yu. 2020. Semantic role labeling guided multi-turn dialogue rewriter. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6632–6639.

Kun Xu, Han Wu, Linfeng Song, Haisong Zhang, Linqi Song, and Dong Yu. 2021. Conversational semantic role labeling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Dian Yu, Kai Sun, Claire Cardie, and Dong Yu. 2020. Dialogue-based relation extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4927–4940.

Jiali Zeng, Yongjing Yin, Yang Liu, Yubin Ge, and Jinsong Su. 2021. Domain adaptive meta-learning for dialogue state tracking. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020. 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.

Hao Zhou, Chujie Zheng, Kaili Huang, Minlie Huang, and Xiaoyan Zhu. 2020. KdConv: A Chinese multi-domain dialogue dataset towards multi-turn knowledge-driven conversation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7098–7108, Online.

Kun Zhou, Kai Zhang, Yu Wu, Shujie Liu, and Jingsong Yu. 2019. Unsupervised context rewriting for open domain conversation. 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 1834–1844.

A Examples for Uncovered Cases

#1	
Contexts (Translation)	U1: 流浪地球有看吗 (Have you seen "The Wandering Earth") U2: 总感觉流浪地球写成短篇浪费了太精彩了 (I fill that it's a waste to make a short film out of "The Wandering Earth", the novel is awesome)
Current utterance (Translation)	U3: 其实导演剪辑版有两小时半 (In fact the director's cut version is 2.5 hour long)
Reference (Translation)	其实导演剪辑版的流浪地球有两小时半 (In fact the director's cut version of "The Wandering Earth" is 2.5 hour long)
TRANS-PG+BERT (Translation)	其实导演剪辑版有两小时半 (In fact the director's cut version is 2.5 hour long)
RAST+RL-GPT2 (Translation)	其实流浪地球导演剪辑版有两小时半 (In fact the director's cut version of "The Wandering Earth" is 2.5 hour long)
#2	
Contexts (Translation) (Translation) (Translation) (Translation)	U1: 有图有真相征男友 (Seeking for boyfriend. I'm with my real personal photographs) U2: 上盘真相先 (Upload your photographs first) U3: 头像就是 (See my profile photo) U4: 看不清 (It's unclear)
Current Utterance (Translation)	U5: 那肿么办 (What can I do)
Reference (Translation)	头像看不清那肿么办 (What can I do if it's unclear)
TRANS-PG+BERT (Translation)	像看不清男友肿 (No translation available as its semantic meaning is unclear.)
RAST+RL-GPT2 (Translation)	肿么办 (What can I do)
#3	
Contexts (Translation) (Translation) (Translation) (Translation)	U1: 一个人的孤单 (A person's loneliness) U2: 一个人吃饭看书写信到处走走停停 (I'm single and everyday I taste cuisine, read books, write letters, and travel around) U3: 这也太悠闲了点我还是有正经事要考虑的 (You are so relaxed, I'd do something regular) U4: 我这是写的文艺了点除了上班就这些 (I described that in an artistic way, I have a regular job)
Current Utterance (Translation)	U5: 那你日子过得也不错哦 (You have a very good life)
Reference (Translation)	除了上班就看书写信那你日子过得也不错哦 (You have a very good life, if you have time to read and write after work)
TRANS-PG+BERT (Translation)	那你的孤孤单单子过得也不错哦 (You have a very good and lonely life)
RAST+RL-GPT2 (Translation)	那你日子过得也不错哦 (You have a very good life)

Table 9: Case study for the uncovered examples.