Improving Multi-party Dialogue Generation via Topic and Rhetorical Coherence

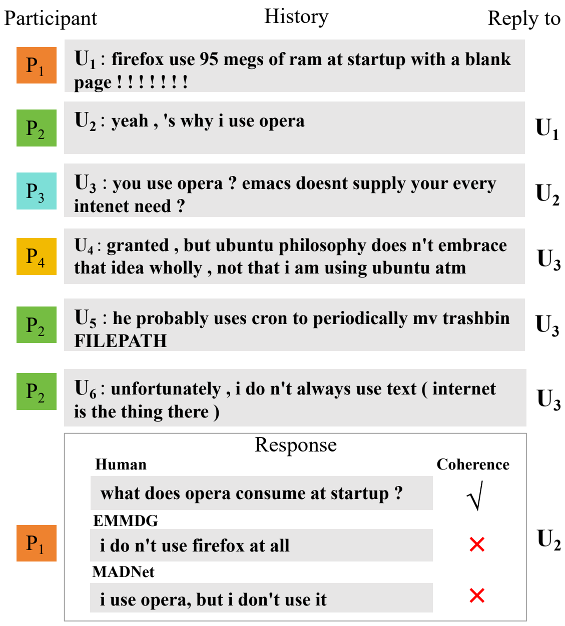
Yaxin Fan[1], **Peifeng Li**[1*], and **Qiaoming Zhu**[1]
[1]School of Computer Science and Technology, Soochow University, Suzhou, China
yxfansuda@stu.suda.edu.cn, {pfli, qmzhu}@suda.edu.cn

Abstract

Previous studies on multi-party dialogue generation predominantly concentrated on modeling the reply-to structure of dialogue histories, always overlooking the coherence between generated responses and target utterances. To address this issue, we propose a Reinforcement Learning approach emphasizing both Topic and Rhetorical Coherence (RL-TRC). In particular, the topic- and rhetorical-coherence tasks are designed to enhance the model's perception of coherence with the target utterance. Subsequently, an agent is employed to learn a coherence policy, which guides the generation of responses that are topically and rhetorically aligned with the target utterance. Furthermore, three discourse-aware rewards are developed to assess the coherence between the generated response and the target utterance, with the objective of optimizing the policy. The experimental results and in-depth analyses on two popular datasets demonstrate that our RL-TRC significantly outperforms the state-of-the-art baselines, particularly in generating responses that are more coherent with the target utterances.

1 Introduction

In a two-party dialogue, a response is always generated for the last utterance in the dialogue history. This contrasts with multi-party dialogue, which always involves multiple participants and each utterance can be uttered by any participant and reply to any participant else (Gu et al., 2022b). In a multi-party dialogue, the position of the target utterance (replied utterance) is not fixed. Figure 1 illustrates a multi-party dialogue involving four participants. In this example, the bot of Multi-party Dialogue Generation (MDG) is required to assume the role of participant P_1 and generate an appropriate response to target utterance U_2.



P_1 U_1 : firefox use 95 megs of ram at startup with a blank page ! ! ! ! ! ! !
P_2 U_2 : yeah , 's why i use opera — U_1
P_3 U_3 : you use opera ? emacs doesnt supply your every intenet need ? — U_2
P_4 U_4 : granted , but ubuntu philosophy does n't embrace that idea wholly , not that i am using ubuntu atm — U_3
P_2 U_5 : he probably uses cron to periodically mv trashbin FILEPATH — U_3
P_2 U_6 : unfortunately , i do n't always use text (internet is the thing there) — U_3
P_1 Response — U_2
Human
what does opera consume at startup ?
EMMDG
i do n't use firefox at all
MADNet
i use opera, but i don't use it

Figure 1: An example of multi-party dialogue from the Hu (Hu et al., 2019) dataset. EMMDG (Li and Zhao, 2023) and MADNet (Gu et al., 2023b) are two SOTA baselines.

Previous studies (Hu et al., 2019; Li and Zhao, 2023; Gu et al., 2022a, 2023b) on MDG have primarily focused on leveraging the reply-to structure within dialogue history to enhance generation. However, these methods do not adequately guarantee the coherence between the generated response and the target utterance.

Firstly, the modeling of reply-to structures does not guarantee that the model will comprehend the topics of the dialogue history, which is of the utmost importance for generating responses that align with the topic of the target utterance. Multi-party dialogue always has multiple entangled parallel dialogue flows, covering different topics. As shown in Figure 1, the example primarily concerns two topics, among which U_1, U_2, and U_7 (human response) discuss the browsers (Firefox and Opera) selection and resource usage, while U_3, U_4, U_5, and U_6 primarily discuss the usage of Emacs. Since the reply-to structure does not unravel the entangled

[*] Corresponding author

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 3240–3253
November 12-16, 2024 ©2024 Association for Computational Linguistics

topics in a multi-party dialogue, it is challenging for the model to consistently generate a response that aligns with the target utterance's topic. As illustrated by the response generated by EMMDG (Li and Zhao, 2023) in Figure 1, the response's topic does not relate to the target utterance U_2, despite EMMDG modeling the reply-to structure. Secondly, modeling reply-to structures does not guarantee logical and meaningful interactions between the generated response and the target utterance. As illustrated in Figure 1, while the response generated by MADNet (Gu et al., 2023b) remains relevant to the topic (Opera) of the target utterance, its internal contradictions undermine the logical coherence between it and the target utterance.

To address these challenges, we propose a Reinforcement Learning approach emphasizing both Topic and Rhetorical Coherence (RL-TRC), guiding the generation of responses that are both topically and logically consistent with the target utterance. We initially design tasks for topic and rhetorical coherence, utilizing the target utterance to predict the topic of the generated response and its rhetorical relation with the target utterance. Subsequently, our reinforcement learning agent learns a coherence policy to guide the model in generating responses that are topically and rhetorically consistent with the target utterance. To optimize this policy, we meticulously design two local rewards and one global reward. The local rewards guide policy learning by scoring the topic coherence and rhetorical coherence between the generated response and the target utterance, respectively. The global reward guides policy learning by measuring whether the target utterance in the dialogue history can be correctly recognized based on the generated response.

We conduct extensive experiments on two popular Ubuntu IRC channel datasets, and results from both automated and manual evaluations indicate that our RL-TRC significantly outperforms previous SOTA baselines. In-depth analysis reveals that RL-TRC can generate responses for target utterances that are more coherent in both topic and rhetoric.

2 Related Work

Multi-party Dialogue Generation Most previous studies on MDG have focused on modeling complex reply-to structures within dialogue histories. Hu et al. (2019) proposed a graph structure neural network that treated utterances as nodes and reply-to relations as edges to model the dialogue history. Gu et al. (2022a) introduced a heterogeneous graph-based neural network to simultaneously model the semantics of utterances and participants using two types of nodes and six types of node-edge relations. Li and Zhao (2023) and Gu et al. (2023b) proposed the adoption of the Expectation-Maximization (EM) algorithm to model the reply-to structure of dialogue history.

Furthermore, some studies have explored enhancing MDG from a discourse perspective. Chernyavskiy and Ilvovsky (2023) suggested using dialogue action planning to improve the consistency and quality of dialogue generation. Meanwhile, Chernyavskiy et al. (2024) proposed leveraging various fine-grained linguistic inputs, including Abstract Meaning Representation, discourse relations, sentiment, and grounding information, to facilitate multi-party dialogue generation. Recently, with the advent of Large Language Models (LLMs), Tan et al. (2023) evaluated the performance of ChatGPT and GPT-4 on MDG. Despite significant progress in MDG, incoherence between the generated response and the target utterance remains a challenge that needs to be addressed.

Dialogue Coherence Modeling Dialogue coherence modeling is primarily applied in persona-based dialogue systems, aiming to help models generate responses that are consistent with the given persona (Chen et al., 2024). Most previous work has focused on minimizing contradictions between responses and personas using natural language inference models (Song et al., 2021; Chen et al., 2023), conditional VAE (Lee et al., 2022), and over-sampling followed by post-evaluation (Zhou et al., 2023).

Unlike previous work on the coherence between the generated response and the persona, we seek to enhance the coherence between the generated response and the target utterance in multi-party dialogues. Uniquely, we model coherence from the perspectives of topic and rhetoric, addressing the multiple topic flows and logical structures in multi-party dialogues.

3 Method

We propose an approach RL-TRC for MDG, illustrated in Figure 2, that begins with using the dialogue history to initialize the sequence state of the Reinforcement Learning (RL) agent, which is

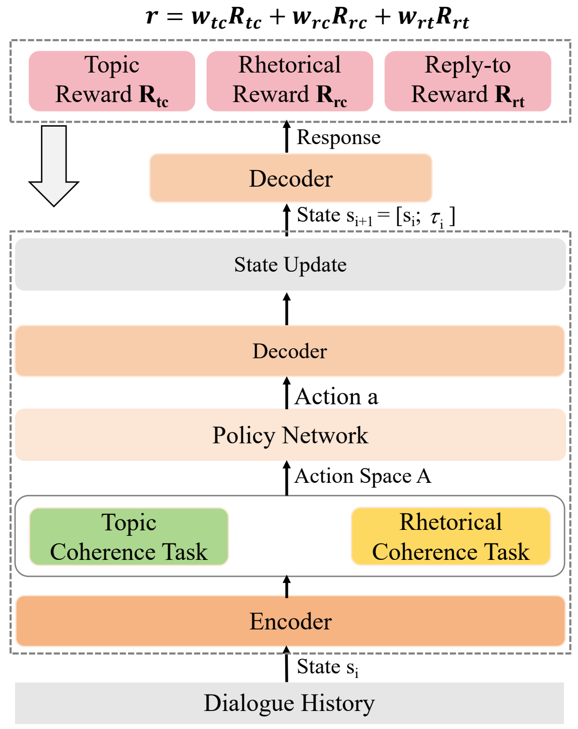


Figure 2: Framework of our method. The parameters of both decoders are shared.

encoded by a dialogue encoder. Then, the topic and rhetorical coherence tasks are devised to enhance the model's awareness of coherence with the target utterance. The coherence semantics are treated as action candidates for the coherence policy and are fed into the decoder to generate a trigger sequence that updates the state. The policy is optimized by evaluating the coherence between the generated response and the target utterance using carefully designed rewards.

3.1 Problem Formulation

Given a multi-party dialogue history $C = \{(p_1, u_1), ..., (p_i, u_i), ..., (p_n, u_n)\}$, where p_i and u_i are participant and utterance, respectively, the model requires to assume the role of a speaker p_r to generate an appropriate response y to the target utterance u_t, which is formulated as

$$y = argmax \sum_{t=1}^{m} logP(y_t|y_{<t}, C, p_r, u_t; \theta) \quad (1)$$

where θ is the trainable parameters, y_t and $y_{<t}$ are the t-th token and the previous t-1 tokens of the response y, respectively, and m is the length of y.

3.2 Encoder

Following Li and Zhao (2023) and Gu et al. (2023b), we use BART (Lewis et al., 2020) as encoder. The dialogue history is fed into the encoder in the form of "[CLS][SEP]$p_1 u_1$[SEP]$p_2 u_2$ $\cdots$ [SEP][RT] $p_t u_t$ $\cdots$ [SEP]$p_n u_n$[SEP]p_r" to obtain the hidden state $H \in \mathbb{R}^{l \times d}$, where l and d are the sequence length and the dimension of hidden state, respectively, and [RT] is the target utterance marker. We denote the semantic representation of the utterance as h_{ui}, which is derived from [SEP] preceding each utterance.

3.3 Topic Coherence Task

The purpose of the topic coherence task is to enable the model to be aware of the topic coherence with the target utterance when generating responses. To this end, we first construct the topic coherence matrix between the target utterance and the golden response. Subsequently, the target utterance and the topic coherence matrix are combined in order to predict the topic of the generated response, thereby ensuring coherence between the target utterance and the generated response.

To construct the topic coherence matrix, we calculate the Pointwise Mutual Information (PMI) (Church and Hanks, 1990) scores between the target utterance and the golden response keywords, and higher PMI scores are associated with stronger topic coherence. Specifically, we first extract the keywords of each utterance by using ChatGPT, which has shown a great topic understanding ability (Fan et al., 2024). The prompt is shown in Appendix A. Given two keywords w_i and w_j, the PMI score of keyword pairs is calculated as

$$PMI(w_i, w_j) = log \frac{p(w_i, w_j)}{p(w_i)p(w_j)} \quad (2)$$

where $p(w_i/w_j)$ is the co-occurrence frequency between w_i in the golden response and w_j in the target utterance, $p(w_i)$ and $p(w_j)$ are the frequency of w_i and w_j, respectively. For each keyword w_j in the target utterance, the w_i with the top 10 $PMI(w_i, w_j)$ scores is adopted to construct the topic coherence matrix. It is worth noting that all utterance pairs with the reply-to relation are utilized to calculate PMI scores.

After obtaining the topic coherence matrix, we first deduce the coherence keywords $C_{ck} = \{w_1, w_2, \cdots, w_k\}$ from the topic coherence matrix according to the keywords of the target utterance, where k is the number of inferred coherence keywords. The semantics $\mathbf{E}_{ck}$ of the coherence keywords is obtained by feeding them into the dialogue encoder, where $\mathbf{E}_{ck} \in \mathbb{R}^{k \times d}$. Then, the semantics of the target utterance $\mathbf{h}_{ut}$ and $\mathbf{E}_{ck}$ are fused by the attention mechanism (Bahdanau et al., 2015) to predict the keywords of the generated response as

$$\mathbf{h}^a_{ut} = MLP(\mathbf{h}_{ut}) \tag{3}$$

$$\begin{aligned}\mathbf{h}_{ti} = softmax(\mathbf{W}_v tanh(\mathbf{W}_q \mathbf{h}^a_{ut} + \\ \mathbf{W}_m \mathbf{E}_{ck} + b))\mathbf{E}_{ck})\end{aligned} \tag{4}$$

$$P_a = softmax(\mathbf{W}_{ti} \mathbf{h}_{ti}) \tag{5}$$

where MLP is a multi-layer perceptron, $\mathbf{W}_v \in \mathbb{R}^d$, $\mathbf{W}_q \in \mathbb{R}^{d\times d}$, $\mathbf{W}_m \in \mathbb{R}^{d\times d}$ and $\mathbf{W}_{ti} \in \mathbb{R}^d$ are trainable parameters, and b is bias term. P_a is the probability of keywords prediction, which is supervised by the keywords $y_{kws} = \{w_i\}_{i=1}^{f}$ in the golden response with the cross-entropy loss as

$$\mathcal{L}_t = -\frac{1}{f}\sum_{i=1}^{f} logP_a(w_i) \tag{6}$$

3.4 Rhetorical Coherence Task

The task of rhetorical coherence primarily enables the model to maintain rhetorical coherence with the target utterance when generating responses. Given that rhetorical relations can explicitly reveal and enhance the understanding of logical and semantic coherence between utterances in multi-party dialogues (Asher and Lascarides, 2003), we utilize a discourse parsing tool (Wang et al., 2021) to analyze the dialogue and identify the rhetorical relation between the golden response and the target utterance. Details can be found in Appendix B. Similar to the topic coherence task, we feed $\mathbf{h}_{ut}$ into another MLP to perform the discourse relation prediction task as

$$\mathbf{h}^b_{ut} = MLP(\mathbf{h}_{ut}) \tag{7}$$

$$P_b = softmax(\mathbf{W}_{ub} \mathbf{h}^b_{ut}) \tag{8}$$

where $\mathbf{W}_{ub} \in \mathbb{R}^d$ is the trainable parameter, and P_b is the probability of relation prediction, which is supervised by the discourse relation r between the target utterance and the golden response with the cross-entropy loss as

$$\mathcal{L}_r = -logP_b(r) \tag{9}$$

3.5 Discourse-aware Reinforcement Learning

The RL agent is built upon the actor-critic framework (Konda and Tsitsiklis, 1999; Ye et al., 2020). In this section, we provide a detailed introduction to each component, including state, action, policy, and reward.

3.5.1 State

The dialogue history C is regarded as the initial state of the RL process denoted as s_1. At the step k, the policy updates the state by selecting the topic or rhetorical coherence semantics to generate the trigger sequence τ_k and the observed state is denoted as $s_k = \{(p_1, u_1), ..., (p_n, u_n), (p^\tau_1, \tau_1), ..., (p^\tau_{k-1}, \tau_{k-1})\}$. Then, s_k is fed into the dialogue encoder to obtain the dialogue semantics h_{sk} derived from the [CLS] token. Finally, the final semantics representation of the current state s_k is obtained by concatenating all the historical state semantics, denoted $\mathbf{s}^e_k = [\mathbf{h}_{s1} : \cdots : \mathbf{h}_{sk}]$.

3.5.2 Action

At the step k, the actor requires to take action a_k to select the topic coherence semantics ($\mathbf{h}^a_{ut}$) or rhetorical coherence semantics ($\mathbf{h}^b_{ut}$), where $a_k \in \mathcal{A} = \{0, 1\}$ (0 for the topic coherence semantics and 1 for the rhetorical coherence semantics). Then, the selected coherence semantics are fed into the BART decoder as a context embedding to generate the trigger sequence τ_k.

3.5.3 Policy

In addition to using a dialogue encoder as a semantic encoding policy network, we design a coherence policy network based on the actor-critic framework. The actor learns a coherence policy $\pi_\theta(s_k, a_k)$, which takes the appropriate action a_k to select the coherence semantics based on the current state s_k. The critic guides the actor to take the appropriate action a_k by evaluating the value $Q_\phi(s_k)$ of state s_k. The actor-critic network is as

$$\mathbf{o}_k = ELU(ELU(\mathbf{s}^e_k \mathbf{W}^o_1)\mathbf{W}^o_2) \tag{10}$$

$$\pi_\theta(a_k|s_k) = softmax(\mathbf{o}_k \mathbf{W}_\theta) \tag{11}$$

$$Q_\phi(s_k) = \mathbf{o}_k \mathbf{W}_\phi \tag{12}$$

where ELU is the activation function,$\mathbf{W}^o_1 \in \mathbb{R}^{(d\times k)\times(d\times k/2)}$, $\mathbf{W}^o_2 \in \mathbb{R}^{(d\times k/2)\times d}$, $\mathbf{W}_\theta \in \mathbb{R}^{d\times 2}$, and $\mathbf{W}_\phi \in \mathbb{R}^{d\times 1}$ are weight matrices.

3.5.4 Reward

To guide the policy learning, we measure the discourse coherence between the generated response and the target utterance. The rewards we designed include two local rewards and one global reward. The local rewards measure the topic and rhetorical coherence between the generated response and the target utterance, and the global reward measures whether the target utterance can be recognized based on the generated response.

Topic-coherence Reward The purpose of the topic-coherence reward is to guide the generation of responses by measuring the topic coherence between the generated response y with its keywords y_{kws}[1] and the target utterance u_t with its keywords u_{t_kws}. We reconstruct the Hu dataset (Hu et al., 2019) to construct the coherent pair (u_t, y) and incoherent pair (u_t, y^{neg}), in which y^{neg} is an utterance randomly selected from the current dialogue not reply to target utterance. Then, the topic evaluator f_{tc} is trained on these coherent and incoherent pairs by using RoBERTa-base (Liu et al., 2019), achieving 84% accuracy. More details are in Appendix C. We adopt the topic evaluator to evaluate the coherence between the generated response and the target utterance and treat the coherence probability as a topic-coherence reward, which is formulated as

$$R_{tc} = f_{tc}([u_t; u_{t\_kws}], [y_t; y_{kws}]) \cdot e^{(n/|y_{kws}|-1)} \tag{13}$$

where n is the number of words in y_{kws} that can be deduced through the topic coherence matrix using u_{t_kws}.

Rhetorical-coherence Reward Rhetorical-coherence reward is used to measure the similarity of the probability distribution of the rhetorical relation between (u_t, y) and (u_t, y^*), where u_t is the target utterance, y is the generated response, and y^* is the golden response. Since the Hu dataset did not annotate rhetorical relations, we reconstruct the Molweni dataset (Li et al., 2020) to construct the target utterance and golden response pairs, and then trained a relation classifier f_{rc} with the accuracy of 65%. More details are in Appendix C. We treat the KL divergence score between $f_{rc}(u_t, y)$ and $f_{rc}(u_t, y^*)$ as the rhetorical-coherence reward as

$$R_{rc} = -KL(f_{rc}(u_t, y^*)||f_{rc}(u_t, y)) \tag{14}$$

Reply-to Reward Reply-to reward aims to measure the similarity between the probability distribution of recognizing the target utterance based on the generated response and the probability distribution of recognizing the target utterance on the golden response. We train a reply-to model f_{rt} on Hu dataset (Hu et al., 2019) by using RoBERTa-base with the accuracy of 91%. More details are in Appendix C. Similar to the rhetorical-coherence reward, we adopt the KL score R_{rt} as the reply-to reward as

$$R_{rt} = -KL(f_{rt}(C, y^*)||f_{rt}(C, y)) \tag{15}$$

where C is the dialogue history.

Dataset	#Train	#Valid	#Test
Hu	311725	5000	5000
Ou5	461120	28570	32668

Table 1: Statistics of the two datasets evaluated in this paper.

Total Reward Finally, we weighted sum all the rewards as

$$r = w_{tc}R_{tc} + w_{rc}R_{rc} + w_{rt}R_{rt} \tag{16}$$

where w_{tc}, w_{rc}, and w_{rt} are weight coefficient.

3.6 Training

During training, we mainly optimize the parameters, including the RL agent, response generation, and coherence task learning. To optimize the learning of the RL agent, we maximize the expected cumulative reward $J(\delta) = \mathbb{E}_\delta[\sum_{k=1}^{K} \epsilon^k r_k]$, where δ is the parameters of actor-critic network, ϵ is the discount coefficient, which reduces the importance of rewards received in the future, and K is the optimization steps, indicating that taking K actions with one iteration of optimization. The agent is optimized by the following loss:

$$\mathcal{L}_{\text{ag}} = -\mathbb{E}_\delta[\log \pi_\theta(a_k|s_k)(\sum_{k=1}^{K} \epsilon^k r_k - Q_\phi(s_k))]. \tag{17}$$

To optimize the learning of the response generation, we feed the semantics $\mathbf{s}_k^e$ of the state s_k into the BART decoder as the context embedding to generate the response, which is optimized by the following loss:

$$\mathcal{L}_{\text{gen}} = -\sum_{t=1}^{m} logP(y_t|y_{<m}, \mathbf{s}_k^e) \tag{18}$$

In addition, the topic and rhetorical coherence tasks are optimized by $\mathcal{L}_t$ and $\mathcal{L}_r$ in Equations 6 and 9, respectively.

[1] The keywords of the generated response are extracted according to the vocabulary of the topic coherence matrix.

	Model	B1	B2	B3	B4	M	R_L
LLMs-based	ChatGPT† (Tan et al., 2023)	11.21	4.44	2.49	1.76	5.38	10.48
RNN-based	GSN (Hu et al., 2019)	10.23	3.57	1.70	0.97	4.10	9.91
BART-based	BART (Lewis et al., 2020)	11.25	4.02	1.78	0.95	4.46	9.90
	HeterMPC (Gu et al., 2022a)	12.26	4.80	2.42	1.49	4.94	11.20
	EMMDG (Li and Zhao, 2023)	12.31	5.39	3.34	2.45	5.52	11.71
	MADNet (Gu et al., 2023b)	12.73	5.12	2.64	1.63	5.31	11.74
	RL-TRC (Ours)	**13.66***	**6.58***	**4.10***	**2.93***	**6.20***	**12.72***

Table 2: Automatic evaluation results on the Hu dataset where * denotes that the improvements are statistically significant (t-test with p-value < 0.05) comparing with the SOTA baselines EMMDG and MADNet, and † represents that we reproduced the performance by running the publicly available code.

	Model	B1	B2	B3	B4	M	R_L
LLMs-based	ChatGPT† (Tan et al., 2023)	11.17	4.06	2.53	1.24	4.42	9.59
RNN-based	GSN (Hu et al., 2019)	6.32	2.28	1.10	0.61	3.27	7.39
BART-based	BART (Lewis et al., 2020)	11.13	3.95	2.11	1.44	4.45	10.20
	HeterMPC (Gu et al., 2022a)	11.40	4.29	2.43	1.74	4.57	10.44
	EMMDG† (Li and Zhao, 2023)	11.67	4.73	2.64	1.81	5.12	10.43
	MADNet (Gu et al., 2023b)	11.82	4.58	2.65	1.91	4.90	10.74
	RL-TRC (Ours)	**12.52***	**5.41***	**3.34***	**2.45***	**5.45***	**11.31***

Table 3: Automatic evaluation results on the Ou5 dataset.

In training, we first pre-train the model with $\mathcal{L}_{\text{gen}}$. Then, we jointly train the model as

$$\mathcal{L} = \mathcal{L}_{\text{ag}} + \mathcal{L}_{\text{gen}} + \mathcal{L}_t + \mathcal{L}_r \tag{19}$$

4 Experimentation

4.1 Datasets

To verify the effectiveness of our RT-TRC, we conduct experiments on two Ubuntu IRC channel datasets, Hu (Hu et al., 2019) and Ou5 (Ouchi and Tsuboi, 2016), following previous work (Gu et al., 2023b). The data statistics are shown in Table 1.

4.2 Baselines

we compare our method with the following baselines. **ChatGPT** (Tan et al., 2023): It directly uses ChatGPT for MDG. We re-ran their publicly available code and used the latest version of GPT-4o-2024-05-13. Notably, we used the same evaluation code as previous work(Gu et al., 2023b) for a fair comparison. **GSN** (Hu et al., 2019): it adopts the homogeneous graph to model the structure of dialogue history. **BART** (Lewis et al., 2020): it directly uses the BART-base model for MDG. **HeterMPC** (Gu et al., 2022a): it models the complicated interactions between utterances and interlocutors in dialogue history with a heterogeneous graph. **EMMDG** (Li and Zhao, 2023): it proposes an expectation-maximization approach that iteratively performs the expectation steps to generate addressee labels, and the maximization steps to optimize a response generation model. **MADNet** (Gu et al., 2023b): it maximizes addressee deduction expectation in heterogeneous graph neural networks for MDG.

4.3 Implementation Details

The pre-trained model we adopt is the BART-base [2] version. The maximum length of the dialogue history and generated response is set to 512 and 50, respectively. The epoch of pre-training and multi-task training are set to 3 and 10, respectively. The strategy of greedy search was adopted for decoding. The batch size is set to 128. The optimization step K of the RL agent is set to 2, which is obtained by using the grid search in $\{1, 2, 3, 4, 5\}$. The reward weights w_{tc}, w_{rc}, w_{rt} are set 0.5, 0.5, 1, respectively, which is obtained by using the grid-search in $\{0.5, 1\}$. The discount coefficient ϵ is set to 0.99. The Adam (Loshchilov and Hutter, 2018) optimizer with an initial learning rate of 2e-5 is adopted and the linear warmup step is set to 200.

[2] https://huggingface.co/facebook/bart-base

4.4 Metrics

We follow previous work (Li and Zhao, 2023; Gu et al., 2022a, 2023b) for automatic and human evaluation. The automatic metrics include BLEU-1 (B1), BLEU-2 (B2), BLEU-3 (B3), BLEU-4 (B4), METEOR (M), and ROUGE$_L$ (R$_L$). For human evaluation, the quality of the generated responses is assessed from three independent dimensions: 1) fluency, 2) informativeness, and 3) relevance. Each human evaluator assigns three binary scores for each response, which are then summed to produce a final score ranging from 0 to 3. Importantly, for the relevance dimension, we specifically measure the alignment of the generated response with the target utterance.

4.5 Results

Tables 2 and 3 present the automated results of our RL-TRC and baselines. It can be observed that the LLMs-based model ChatGPT can only achieve comparable performance to the other baselines, indicating that MDG still faces great challenges. Additionally, compared with the BART model, HeterMPC, EMMDG, and MADNet significantly enhance dialogue generation quality by incorporating reply-to structures. However, these methods still suffer from incoherence between the generated responses and the target utterances. Our RL-TRC outperforms all baselines in terms of all metrics, demonstrating its effectiveness in improving generation quality by enhancing discourse coherence between generated responses and target utterances.

Table 4 presents the results of the human evaluation on the Hu dataset. We randomly sampled 200 examples from the test set and recruited three computer science master students familiar with Ubuntu and Linux to score each response. The results demonstrate that all models exhibit comparable performance to humans in terms of fluency, indicating that generating fluent responses is no longer a significant challenge. Furthermore, our RL-TRC model demonstrates comparable performance to MADNet in terms of informativeness, although there remains a gap when compared to human performance. Notably, our RL-TRC model significantly improves the relevance score, generating responses that are more aligned with target utterances. This further substantiates the efficacy of our model.

Model	Flu	Infor	Rel	Overall
Human	0.95	0.94	0.84	2.73
EMMDG	0.94	0.71	0.58	2.23
MADNet	**0.97**	0.81	0.63	2.41
RL-TRC	0.95	**0.83**	**0.73**	**2.51**

Table 4: Human evaluation results of our RL-TRC and two SOTA baselines on a randomly sampled test set of Hu, where Rel, Flu, and Inf are short for Relevance, Fluency, and Informativeness, respectively. The agreement rate of the human evaluation outperforms 85% demonstrating the reliability of the human evaluation.

5 Analysis

In this section, we focus on an in-depth analysis on the Hu dataset, and the results on the Ou5 dataset are presented in Appendix E, G, and H.

5.1 Ablation Study

We perform ablation experiments to investigate the impact of each component within our RL-TRC. The results in Table 5 demonstrate that the discard of any component leads to a decline in performance, underscoring the significance of each part. A comparison between coherence tasks (TC vs. RC) reveals that discarding the topic coherence task causes a more substantial performance drop. This indicates the importance of understanding the topics of target utterances, as multi-party dialogues often involve multiple parallel topic flows. Furthermore, an analysis of coherence rewards (TCR vs. RCR vs. RTR) shows that the topic-coherence reward has a more pronounced impact on performance. Importantly, removing the coherence rewards tends to result in greater performance degradation compared to removing the coherence tasks, highlighting the effectiveness of reinforcement learning in enhancing coherence between generated responses and target utterances. It is noteworthy that the improvement of rhetorical coherence is less than that of topic coherence, primarily due to the low accuracy of the discourse parser.

5.2 Analysis of Topic Coherence

To evaluate whether our RL-TRC can generate responses that are more relevant to the topic of the target utterance compared to the baselines, we conduct a pairwise evaluation. Given that GPT-4 has been widely used for pairwise evaluations (Zheng et al., 2023; Dubois et al., 2024), we employ GPT-4 as a judge to determine which response is more relevant to the topic of the target utterance. Detailed

Model	B1	B2	B3	B4	M	R_L
RL-TRC	13.66	6.58	4.10	2.93	6.20	12.72
w/o TC	-1.04	-0.96	-0.72	-0.56	-0.60	-1.07
w/o RC	-0.16	-0.09	-0.06	-0.05	-0.06	-0.14
w/o TCR	-1.30	-1.24	-0.99	-0.82	-0.66	-1.44
w/o RCR	-0.23	-0.12	-0.25	-0.17	-0.24	-0.32
w/o RTR	-0.78	-0.54	-0.42	-0.33	-0.31	-0.45

Table 5: Ablation results on the Hu dataset. TC and RC refer to the topic-coherence and rhetorical-coherence tasks, respectively. TCR, RCR, and RTR stand for topic-coherence, rhetorical-coherence, and reply-to reward, respectively. '-' represents degraded performance.

Pair-wise Evaluation	Dataset	Win	Tie	Lose
RL-TRC v.s. EMMDG	Hu	75	70	55
	Ou5	71	77	52
RL-TRC v.s. MADNet	Hu	65	80	55
	Ou5	69	74	57

Table 6: Pairwise evaluation results of GPT-4 in terms of topic coherence.

information is provided in Appendix F.

Table 6 presents the pairwise evaluation results. Compared to EMMDG, RL-TRC wins 75 and 71 on the Hu and Ou5 datasets, respectively, while losing 55 and 52. Compared to MADNet, RL-TRC wins 65 and 69 on Hu and Ou5, respectively, while losing 55 and 57. These results further demonstrate that our approach generates responses more relevant to the target utterance topic, benefiting from our topic coherence task and reward.

5.3 Analysis of Rhetorical Coherence

To verify whether our RL-TRC can generate a more rhetorically coherent response with the target utterance, we present the accuracy of the rhetorical relation between the generated response and the target utterance. We first employ a trained discourse parser (Wang et al., 2021) to predict the rhetorical relation between the golden response and the target utterance, which is treated as the golden relation. Then, we use the parser to predict the relation between the generated response and the target utterance and calculate the relation accuracy, as shown in Table 7.

Notably, over 98% (4935 out of 5000) of the relations in the Hu dataset are classified as Comment, Question-Answer Pair, Continuation, or Clarification Question. Our RL-TRC model achieved accuracies of 59.17% on the Hu dataset, reflecting a significant improvement compared to the performances of EMMDG and MADNet. This suggests that our approach generates responses that exhibit greater rhetorical coherence with the target utterances, attributable to the incorporation of our rhetorical coherence task and reward.

Relation	Golden	EMMDG	MADNet	RL-TRC
#Comment	1817	1321	1355	1390
#QAP	1057	844	863	880
#Cont	504	209	210	195
#Clar_Q	1557	351	402	455
Total	4935	2725	2830	2920
Accuracy		55.22	57.35	**59.17**

Table 7: Accuracy of the relations between the generated response and the target utterance on the Hu dataset, where QAP, Cont, and Clar_Q are short for the relations Question-answer Pair, Continuation, and Clarification_question, respectively.

Model	Hu	Ou5
Golden	95.68	84.09
EMMDG	78.89	66.13
MADNet	81.02	70.80
RL-TRC	**85.38**	**75.51**

Table 8: Accuracy of target utterances that are recognized based on the generated response, where we adopt the MPC-BERT (Gu et al., 2021) as the evaluator.

5.4 Analysis of Target Utterance Recognition

We analyze the performance of correctly recognizing the target utterance in the dialogue history based on the generated response. Intuitively, the stronger the coherence between the generated responses and the target utterance, the higher the performance. As shown in Table 8, the golden responses achieve the highest performance, with an accuracy of 95.68% and 84.09% for Hu and Ou5, respectively, demonstrating strong coherence between the golden response and the target utterance. In addition, our RL-TRC achieves an accuracy of 85.38% and 75.51% on Hu and Ou5, respectively,

Epoch	Accuracy of Topic Reward Model	B1	B2	B3	B4	M	R_L
1	72%	13.17	6.21	3.86	2.68	5.99	12.39
2	79%	13.54	6.48	3.96	2.89	6.18	12.58
3	84%	13.66	6.58	4.10	2.93	6.20	12.72

Table 9: Impact of varying performances of topic coherence reward model on dialogue generation in the Hu dataset.

Epoch	Accuracy of Rhetorical Reward Model	B1	B2	B3	B4	M	R_L
1	50%	13.48	6.46	3.97	2.77	6.04	12.37
2	59%	13.60	6.54	4.07	2.80	6.07	12.69
3	65%	13.66	6.58	4.10	2.93	6.20	12.72

Table 10: Impact of varying performances of rhetorical coherence reward model on dialogue generation in the Hu dataset.

significantly surpassing the current SOTA baselines. This demonstrates the effectiveness of our method in enhancing the coherence between the generated responses and the target utterances, thereby facilitating multi-party dialogue generation.

5.5 Importance of Accuracy in Topic and Rhetorical Reward Models

We analyzed the impact of varying performances of the topic coherence and rhetorical coherence reward models on dialogue generation. The results for the Hu dataset are presented in Tables 9 and 10, respectively. In these tables, 'epoch' refers to the number of training iterations for the reward models. The experimental setup remains consistent with our state-of-the-art (SOTA) model, with the only modification replacing the topic coherence or rhetorical coherence reward model at different training epochs. Our observations indicate that the improved performance of both the topic and rhetorical coherence reward models correlate with enhanced generation performance. This suggests that more effective reward models are better at evaluating the coherence between generated responses and target utterances, thereby promoting a more coherent response to the target utterance.

5.6 Case Study

We conduct a case study to showcase the effectiveness of our RL-TRC model in Table 11. The dialogue history is depicted in Figure 1, with the target utterance being U_2. We observe that the topic coherence between the response generated by EMMDG and U_2 is not strong. U_2 pertains to "opera" but the response generated by EMMDG is about "Firefox", which is more topically coherent with U_1. Additionally, although the response generated by MADNet is related to "opera", it lacks a logical connection with the target utterance. In contrast, our RL-TRC produces a response that is both topic-related and logical, further demonstrating the effectiveness of our approach. In addition, we explore the impact of different topic extraction methods, as shown in Appendix H.

Model	Response
Golden	what does opera consume at startup?
EMMDG	i do n't use firefox at all
MADNet	i use opera, but i don't use it
RL-TRC	i do n't use opera , i use firefox

Table 11: Responses generated by our model and two SOTA baselines. The dialogue history is shown in Figure 1.

6 Conclusion

In this paper, we propose a reinforcement learning method based on discourse coherence for multi-party dialogue generation. By designing tasks centered on topic coherence and rhetorical coherence, we enable the model to perceive coherence with the target utterance. Furthermore, a reinforcement agent is employed to guide the model to generate responses that are topically and rhetorically aligned with the target utterances. To optimize the agent, three types of discourse-aware rewards are designed to guide the model to maintain coherence with the target utterance. Experimental results validate the effectiveness of our method. Our future work will focus on how to optimize rhetorical coherence.

Limitations

We discuss the limitations of RL-TRC as follows: 1) Our limited computational resources prevented us from verifying the effectiveness of our method on larger model sizes. Despite conducting parameter-efficient fine-tuning on LLaMA, the results were not satisfactory. As large language models continue to gain prominence, we aim to perform full parameter fine-tuning in the future. 2) Rhetorical coherence does not contribute to multi-party dialogue generation as significantly as topic coherence. The main reason may be the low performance of the discourse parser. Thus, optimizing rhetorical coherence to facilitate multi-party dialogue generation is a challenge that needs to be addressed. 3) The inherent complexity of reinforcement learning algorithms can lead to instability and a tendency to get stuck in local optima during training. Consequently, careful tuning and adjustment of hyperparameters are essential.

Acknowledgements

The authors would like to express their gratitude to the three anonymous reviewers for their insightful comments on this paper. We also extend our heartfelt thanks to Xiang Li, Huiyao Wang, and Didi Zhang for their contributions to the human evaluation. This research was supported by the National Natural Science Foundation of China (Nos. 62276177 and 62376181), and Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

References

Nicholas Asher and Alex Lascarides. 2003. *Logics of conversation*. Cambridge University Press.

Dzmitry Bahdanau, KyungHyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation By Jointly Learning To Align And Translate. In *The Thrid International Conference on Learning Representations*.

Ruijun Chen, Jin Wang, Liang-Chih Yu, and Xuejie Zhang. 2023. Learning to Memorize Entailment and Discourse Relations for Persona-consistent Dialogues. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 12653–12661.

Yi-Pei Chen, Noriki Nishida, Hideki Nakayama, and Yuji Matsumoto. 2024. Recent Trends in Personalized Dialogue Generation: A Review of Datasets, Methodologies, and Evaluations. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, pages 13650–13665.

Alexander Chernyavskiy and Dmitry Ilvovsky. 2023. Transformer-based multi-party conversation generation using dialogue discourse acts planning. In *Proceedings of the 24th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 519–529.

Alexander Chernyavskiy, Lidiia Ostyakova, and Dmitry Ilvovsky. 2024. GroundHog: Dialogue generation using multi-grained linguistic input. In *Proceedings of the 5th Workshop on Computational Approaches to Discourse (CODI 2024)*, pages 149–160.

Kenneth Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29.

Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S Liang, and Tatsunori B Hashimoto. 2024. AlpacaFarm: A Simulation Framework for Methods that Learn from Human Feedback. *Advances in Neural Information Processing Systems*.

Yaxin Fan, Feng Jiang, Peifeng Li, and Haizhou Li. 2024. Uncovering the Potential of ChatGPT for Discourse Analysis in Dialogue: An Empirical Study. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, pages 16998–17010.

Jia-Chen Gu, Zhenhua Ling, Quan Liu, Cong Liu, and Guoping Hu. 2023a. GIFT: Graph-Induced Fine-Tuning for Multi-Party Conversation Understanding. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11645–11658.

Jia-Chen Gu, Chao-Hong Tan, Caiyuan Chu, Zhen-Hua Ling, Chongyang Tao, Quan Liu, and Cong Liu. 2023b. MADNet: Maximizing Addressee Deduction Expectation for Multi-Party Conversation Generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7681–7692.

Jia-Chen Gu, Chao-Hong Tan, Chongyang Tao, Zhen-Hua Ling, Huang Hu, Xiubo Geng, and Daxin Jiang. 2022a. HeterMPC: A Heterogeneous Graph Neural Network for Response Generation in Multi-Party Conversations. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5086–5097.

Jia-Chen Gu, Chongyang Tao, and Zhen-Hua Ling. 2022b. Who Says What to Whom: A Survey of Multi-Party Conversations. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, pages 5486–5493.

Jia-Chen Gu, Chongyang Tao, Zhenhua Ling, Can Xu, Xiubo Geng, and Daxin Jiang. 2021. MPC-BERT: A Pre-Trained Language Model for Multi-Party Conversation Understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational*

Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 3682–3692.

Wenpeng Hu, Zhangming Chan, Bing Liu, Dongyan Zhao, Jinwen Ma, and Rui Yan. 2019. GSN: A Graph-Structured Network for Multi-Party Dialogues. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence*, pages 5010–5016.

Vijay Konda and John Tsitsiklis. 1999. Actor-Critic Algorithms. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 1008–1014.

Jing Yang Lee, Kong Aik Lee, and Woon Seng Gan. 2022. Improving Contextual Coherence in Variational Personalized and Empathetic Dialogue Agents. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 7052–7056.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pretraining for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Jiaqi Li, Ming Liu, Min-Yen Kan, Zihao Zheng, Zekun Wang, Wenqiang Lei, Ting Liu, and Bing Qin. 2020. Molweni: A Challenge Multiparty Dialogues-based Machine Reading Comprehension Dataset with Discourse Structure. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2642–2652.

Yiyang Li and Hai Zhao. 2023. EM Pre-training for Multi-party Dialogue Response Generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pages 92–103.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Pptimized Bert Pretraining Approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

Hiroki Ouchi and Yuta Tsuboi. 2016. Addressee and Response Selection for Multi-Party Conversation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2133–2143.

Haoyu Song, Yan Wang, Kaiyan Zhang, Weinan Zhang, and Ting Liu. 2021. BoB: BERT Over BERT for Training Persona-based Dialogue Models from Limited Personalized Data. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 167–177.

Chao-Hong Tan, Jia-Chen Gu, and Zhen-Hua Ling. 2023. Is ChatGPT a Good Multi-Party Conversation Solver? In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4905–4915.

Jianheng Tang, Tiancheng Zhao, Chenyan Xiong, Xiaodan Liang, Eric Xing, and Zhiting Hu. 2019. Target-Guided Open-Domain Conversation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5624–5634.

Ante Wang, Linfeng Song, Hui Jiang, Shaopeng Lai, Junfeng Yao, Min Zhang, and Jinsong Su. 2021. A Structure Self-Aware Model for Discourse Parsing on Multi-Party Dialogues. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence*, pages 3943–3949.

Zhiquan Ye, Yuxia Geng, Jiaoyan Chen, Jingmin Chen, Xiaoxiao Xu, SuHang Zheng, Feng Wang, Jun Zhang, and Huajun Chen. 2020. Zero-shot Text Classification via Reinforced Self-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3014–3024.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, pages 46595–46623.

Junkai Zhou, Liang Pang, Huawei Shen, and Xueqi Cheng. 2023. SimOAP: Improve Coherence and Consistency in Persona-based Dialogue Generation via Over-sampling and Post-evaluation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pages 9945–9959.

Relation	Description
Comment	Arg2 comments Arg1.
Clarification question	Arg2 clarifies Arg1.
Question-answer pair	Arg1 is a question and Arg2 is the answer of Arg1.
Continuation	Arg2 is the continuation of Arg1.
Acknowledgement	Arg2 acknowledges Arg1.
Q-Elab	Arg1 is a question and Arg2 tries to elaborate Arg1.
Result	Arg2 is the effect brought about by the situation described in Arg1.
Elaboration	Arg2 elaborates Arg1.
Explanation	Arg2 is the explanation of Arg1.
Correction	Arg2 corrects Arg1.
Contrast	Arg1 and Arg2 share a predicate or property and a difference on shared property.
Conditional	Arg1 is the condition of Arg2 or Arg2 is the condition of Arg1.
Background	Arg2 is the background of Arg1.
Narration	Arg2 is the narration of Arg1.
Alternation	Arg1 and Arg2 denote alternative situations.
Parallel	Arg2 and Arg1 are parallel and present almost the same meaning.

Table 12: Discourse relations and their descriptions, cited from Li et al. (2020).

Reward	#Train	#Valid	#Test
Topic	2775454	44644	44520
Rhetoric	70454	3880	3911
Reply-to	311725	5000	5000

Table 13: Number of samples for training the reward models.

A Prompt of Extracting Topics with ChatGPT

We feed the dialogue to ChatGPT[3] and ask ChatGPT to extract no more than five keywords for each utterance, the prompt is as follows:

> The following is a conversation with multiple participants. Please extract the key words from each utterance. The number of key words should not exceed five, and each key word should consist of only one word. Please return a dictionary, where the key is the index of the utterance and the value is a list of key words. Do not return anything else.
> U_1:
> U_2:
> ...
> U_n

B Details of Discourse Parsing Tool

In this paper, we adopted the discourse parser trained by Wang et al. (2021) and provide golden links to parser to predict discourse relations. There are mainly 16 types of discourse relations, the relation types and descriptions are shown in Table 12.

C Reward Model

Topic-coherence model The topic-coherence model is a binary classifier that determines whether an utterance pairs is coherent or not. We reconstructed the Hu dataset (Hu et al., 2019) to construct the coherent pair (u_t, y) and incoherent pair (u_t, y^{neg}), in which u_t and y are target utterance and golden response, respectively, y^{neg} is an utterance randomly selected from the current dialogue not reply to target utterance, the statistics are shown in Table 13. We feed the utterance pair and their keywords into the BART-base (Lewis et al., 2020) in the form of "$[CLS]u_tu_{tkws}[SEP]yy_{kws}[SEP]$". To train the classifier, we adopted the Trainer function of transformer library [4]. The epoch, batch size, learning rate, and weight decay are set to 3, 192, 2e-5, and 0.02, respectively, and other hyperparameters are set default.

[3] The version is 'gpt-3.5-turbo-0301'.

[4] https://huggingface.co/docs/transformers/index

Model	Flu	Infor	Rel	Overall
Human	0.87	0.95	0.78	2.60
EMMDG	0.73	0.63	0.48	1.84
MADNet	0.73	0.78	0.50	2.01
RL-TRC	**0.75**	**0.81**	**0.57**	**2.13**

Table 14: Human evaluation results of our RL-TRC and two SOTA baselines on a randomly sampled test set of Ou5. Rel, Flu, and Inf are short for Relevance, Fluency, and Informativeness, respectively. The agreement rate of the human evaluation outperforms 75% on all three metrics.

Rhetorical-coherence model The rhetorical-coherence model is a multi-class classifier that recognizes the rhetorical relation between an utterance pair. We extract the utterance pairs with a rhetorical relation from the Molweni (Li et al., 2020) dataset and the data statistics are shown in Table 13. All hyperparameters are set to the same value as the topic-coherence model.

Reply-to model Given a multi-party dialogue history $C = \{(p_1, u_1), ..., (p_i, u_i), ..., (p_n, u_n)\}$, where p_i and u_i are participant and utterance, respectively, the reply-to model aims to recognize a target utterance u_t for the generated response y, where $1 < t \leq n$. Following the previous work (Gu et al., 2023a), we trained the reply-to model on the Hu (Hu et al., 2019) dataset, and the data statistics are shown in Table 13. The pre-trained model we adopted is RoBERTa-base. The epoch, batch size, learning rate and weight decay are set to 3, 32, 2e-5, and 0.02, respectively.

D Human evaluation results on the Ou5 Dataset

The human evaluation results on the Ou5 dataset is shown in Table 14. Similarly, our RL-TRC can significantly improve the relevance score, demonstrating the effectiveness of our method in enhancing the coherence between generated responses and target utterances.

E Ablation results on the Ou5 Dataset

Ablation results on the Ou5 (Ouchi and Tsuboi, 2016) dataset is shown in Table 15. The phenomenon on the Ou5 dataset is consistent with that on the Hu dataset, i.e., topic coherence task and reward have a more pronounced effect on performance.

F Pairwise Evaluation with GPT-4

To evaluate which of the responses generated by the two models is more relevant to the target utterance in terms of topic, we follow previous work (Zheng et al., 2023; Dubois et al., 2024) to conduct pairwise evaluation using GPT-4 [5]. The 200 samples evaluated are the same as those used in human evaluation of Section 4.5. GPT-4 is instructed to compare the outputs of two models and determine which one exhibited a stronger relevance with the topic of the target utterance in the dialogue history. The model names remained anonymous, and the positions of the model outputs were randomly swapped. The prompt is as follows:

> This is a conversation history consisting of multiple speakers.
> **[The beginning of dialogue history]**
> U_1
> U_2
> $\cdots$
> u_t
> $\cdots$
> U_n
> **[The end of dialogue history]**
>
> Here are two bots generating two responses to target utterance U_t.
>
> **A: response1**
>
> **B: response2**
>
> Please determine which of the above two responses is more closely related to the target utterance U_t in terms of topic.
>
> Please return the option directly.

G Relation Performance on the Ou5 Dataset

The relation accuracy on the Ou5 (Ouchi and Tsuboi, 2016) dataset is shown in Table 18. Our RL-TRC achieves an accuracy of 55.92%, significantly outperforming EMMDG and MADNet, which suggests that our method can further enhance the rhetorical coherence between generated response and the target utterance.

[5] The GPT-4 version is gpt-4-0613.

Model	B1	B2	B3	B4	M	R_L
RL-TRC	12.52	5.41	3.34	2.45	5.45	11.31
w/o TC	-1.04	-1.35	-1.42	-1.40	-0.54	-0.88
w/o RC	-0.27	-0.08	-0.15	-0.46	-0.16	-0.53
w/o TCR	-1.26	-1.31	-1.40	-1.33	-0.78	-1.39
w/o RCR	-0.61	-0.77	-1.05	-1.07	-0.28	-0.77
w/o RTR	-0.75	-0.88	-1.01	-0.98	-0.33	-1.00

Table 15: Ablation results on the Ou5 (Ouchi and Tsuboi, 2016) dataset. TC and RC means the topic and rhetorical coherence tasks, respectively. TCR, RCR, and RTR stand for topic-coherence, rhetorical-coherence, and reply-to reward, respectively.

Method	B1	B2	B3	B4	M	R_L
ChatGPT	13.66	6.58	4.10	2.93	6.20	12.72
KeyBERT	12.42	5.38	3.17	2.10	5.48	11.25
Rule-based	13.00	5.99	3.60	2.50	5.84	11.89

Table 16: Results comparison of topic extraction methods on the Hu dataset.

Method	B1	B2	B3	B4	M	R_L
ChatGPT	12.52	5.41	3.34	2.45	5.45	11.31
KeyBERT	11.44	5.00	2.90	2.06	5.19	10.38
Rule-based	11.87	5.06	2.06	2.38	5.24	10.62

Table 17: Results comparison of topic extraction methods on the Ou5 dataset.

Relation	Golden	EMMDG	MADNet	RL-TRC
#Comment	13821	9767	9518	9879
#QAP	3642	2340	2497	2629
#Cont	2903	1106	1163	1232
#Clar_Q	12065	4131	4265	4395
Total	32431	17344	17443	18135
Accuracy		53.48	53.78	**55.92**

Table 18: Accuracy of the relation between the generated response and the target utterance on the Ou5 (Ouchi and Tsuboi, 2016) dataset. QAP, Cont and Clar_Q are short for Question-answer Pair, Continuation, and Clarification_question, respectively.

H Comparison of Topic Extraction Methods

Since the extracted topic serves as the foundation for enhancing topic coherence, we investigate the impact of various topic extraction methods on generation. The results for the Hu and Ou5 datasets are presented in Tables 16 and 17. KeyBERT[6] leverages BERT embeddings to extract keywords most similar to a document, while the rule-based method (Tang et al., 2019) combines TF-IDF and part-of-speech features to score word salience. Our findings indicate that ChatGPT achieves the best performance, demonstrating its strong generalization ability in topic extraction. Additionally, KeyBERT performs worse than the rule-based method, likely due to its limited generalization capabilities, which hinder its effectiveness in extracting topics from the Ubuntu operating system corpus.

[6] https://github.com/MaartenGr/KeyBERT