BORT: Back and Denoising Reconstruction for End-to-End Task-Oriented Dialog

Haipeng Sun, Junwei Bao*, Youzheng Wu, Xiaodong He
JD AI Research, Beijing, China
{sunhaipeng6, baojunwei, wuyouzheng1, hexiaodong}@jd.com

Abstract

A typical end-to-end task-oriented dialog system transfers context into dialog state, and upon which generates a response, which usually faces the problem of error propagation from both previously generated inaccurate dialog states and responses, especially in low-resource scenarios. To alleviate these issues, we propose BORT, a **b**ack and den**o**ising **r**econs**t**ruction approach for end-to-end task-oriented dialog system. Squarely, to improve the accuracy of dialog states, *back reconstruction* is used to reconstruct the original input context from the generated dialog states since inaccurate dialog states cannot recover the corresponding input context. To enhance the denoising capability of the model to reduce the impact of error propagation, *denoising reconstruction* is used to reconstruct the corrupted dialog state and response. Extensive experiments conducted on MultiWOZ 2.0 and CamRest676 show the effectiveness of BORT. Furthermore, BORT demonstrates its advanced capabilities in the zero-shot domain and low-resource scenarios[1].

1 Introduction

Recently, task-oriented dialog systems, which aim to assist users to complete some booking tasks, have attracted great interest in the research community and the industry (Zhang et al., 2020c). Task-oriented dialog systems have been usually established via a pipeline system, including several modules such as natural language understanding, dialog state tracking, dialog policy, and natural language generation. The natural language understanding module converts user utterance into the structured semantic representation. The dialog state generated by the dialog state tracking module is used to query the database to achieve matched entities. The natural language generation module converts the action state estimated by the dialog policy module to the natural language response. This modular-based architecture is highly interpretable and easy to implement, used in most practical task-oriented dialog systems in the industry. However, every module is optimized individually and doesn't consider the entire dialog history, which affects the performance of the dialog system. Therefore, many researchers focus on end-to-end task-oriented dialog systems to train an overall mapping from user natural language input to system natural language output (Lei et al., 2018; Zhang et al., 2020b; Hosseini-Asl et al., 2020; Lin et al., 2020; Yang et al., 2021).

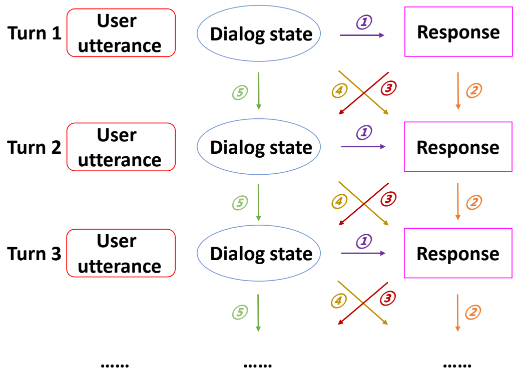

Figure 1: Illustration of different error propagation problem types, denoted by arrows in different colors, along the multi-turn task-oriented dialog flow. For example, the orange arrow indicates that the error in the previously generated response would affect the response generation in the current dialog turn.

However, all existing task-oriented dialog systems still suffer from one or more types of error propagation problems from both previously generated inaccurate dialog states and responses, as is illustrated in Figure 1. Firstly, the generated dialog state, which is crucial for task completion of task-oriented dialog systems, is usually inaccurate across the end-to-end task-oriented dialog system training. Secondly, the previously generated dialog state and response are encoded to create the current

*Corresponding author.
[1]The code is available at https://github.com/JD-AI-Research-NLP/BORT.

Findings of the Association for Computational Linguistics: NAACL 2022, pages 2156 - 2170
July 10-15, 2022

dialog state and response during inference, while the oracle previous dialog state and response are encoded during training. There exists a discrepancy between training and inference, affecting the quality of generated system responses.

We propose BORT, a back and denoising reconstruction approach for end-to-end task-oriented dialog systems to alleviate these issues. To improve dialog state learning ability, back reconstruction is used to reconstruct the generated dialog state back to the original input context to ensure that the information in the input side is completely transformed to the output side. In addition, to enhance the denoising capability of the task-oriented dialog system to reduce the impact of error propagation, denoising reconstruction is used to reconstruct the corrupted dialog state and response. It guarantees that the system learns enough internal information of the dialog context to recover the original version. Experimental results on MultiWOZ 2.0 and CamRest676 show that our proposed BORT substantially outperforms baseline systems. This paper primarily makes the following contributions:

- We propose two effective reconstruction strategies, i.e., back and denoising reconstruction strategies, to improve the performance of end-to-end task-oriented dialog systems.
- Extensive experiments and analysis on MultiWOZ 2.0 and CamRest676 show the effectiveness of BORT.
- BORT achieves promising performance in zero-shot domain scenarios and alleviates poor performance in low-resource scenarios.

2 Task-Oriented Dialog Framework

As illustrated in Figure 2(a), we construct an encoder-decoder framework for an end-to-end task-oriented dialog system via dialog state tracking and response generation tasks. One shared encoder encodes dialog context, and two different decoders decode dialog state and system response, respectively. The objective function $\mathcal{L}_{all}$ of the entire training process is optimized as:

$$\mathcal{L}_{all} = \mathcal{L}_B + \mathcal{L}_R, \tag{1}$$

where $\mathcal{L}_B$ is the objective function for dialog state tracking, and $\mathcal{L}_R$ is the objective function for response generation.

2.1 Dialog State Tracking

Motivated by Lin et al. (2020), we model the Levenshtein dialog state, which means the difference between the current dialog state and the previous dialog state, for dialog state tracking task to generate minimal dialog state and reduce the inference latency. The Levenshtein dialog state ΔB_t of dialog turn t, is generated based on the previous dialog state B_{t-1}, the previous system response R_{t-1}, and the current user utterance U_t via the encoder-decoder framework:

$$H_{eb} = encoder(B_{t-1}, R_{t-1}, U_t), \tag{2}$$

$$\Delta B_t = decoder_b(H_{eb}), \tag{3}$$

where H_{eb} denotes the hidden representation of the encoder for dialog state tracking. Therefore, the dialog state tracking objective function can be optimized by minimizing:

$$\mathcal{L}_B = \sum_{i=1}^{N}\sum_{t=1}^{n_i} -logP(\Delta B_t | B_{t-1}, R_{t-1}, U_t), \tag{4}$$

where N denotes the number of dialog sessions, n_i denotes the number of dialog turns in the dialog session i.

For inference, a predefined function $\Omega(\cdot)$ is used to generate the dialog state B_t as

$$B_t = \Omega(\Delta B_t, B_{t-1}). \tag{5}$$

The predefined function $\Omega(\cdot)$ deletes the slot-value pair in B_{t-1} when the `NULL` symbol appears in the ΔB_t, and it updates the B_{t-1} when new slot-value pair or new value for one slot appears in the ΔB_t. Refer to Lin et al. (2020) for more details. The generated dialog state B_t is used to query the corresponding database. The database state embedding DB_t represents the number of matched entities and whether the booking is available or not. The embedding DB_t is used as the start token embedding of the response decoder for response generation.

2.2 Response Generation

The response R_t of dialog turn t is generated based on the previous system response R_{t-1}, the current user utterance U_t, the current dialog state B_t, and the database state embedding DB_t, which is formulated as:

$$H_{er} = encoder(R_{t-1}, U_t, B_t), \tag{6}$$

$$R_t = decoder_r(H_{er}, DB_t), \tag{7}$$

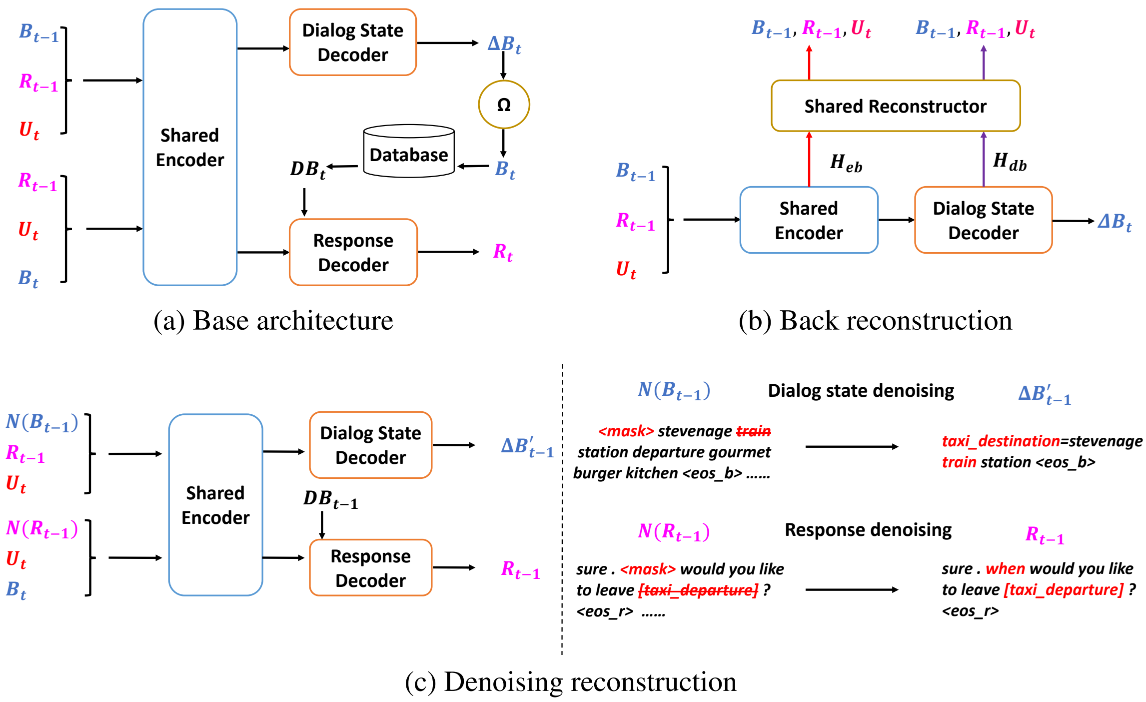

Figure 2: Illustration of the task-oriented dialog training process. We take turn t of a dialog session as an example.

where H_{er} denotes the hidden representation of the encoder for response generation. Therefore, the response generation objective function can be optimized by minimizing:

$$\mathcal{L}_R = \sum_{i=1}^{N}\sum_{t=1}^{n_i} -logP(R_t|R_{t-1}, U_t, B_t, DB_t). \quad (8)$$

3 Methodology

In this section, we proposes two reconstruction strategies, i.e., back reconstruction and denoising reconstruction, respectively. Generally, during task-oriented dialog training, objective functions $\mathcal{L}_{BR}$ and $\mathcal{L}_{DR}$ are added to enhance model learning ability. The general objective function of a task-oriented dialog system can be reformulated as follows:

$$\mathcal{L}_{all} = \mathcal{L}_B + \mathcal{L}_R + \lambda_1\mathcal{L}_{BR} + \lambda_2\mathcal{L}_{DR}, \quad (9)$$

where $\mathcal{L}_{BR}$ and $\mathcal{L}_{DR}$ denote the objective functions for back reconstruction and denoising reconstruction. λ_1 and λ_2 are hyper-parameters that adjust the weights of the objective functions.

3.1 Back Reconstruction

Dialog state is essential for the task completion of a task-oriented dialog system (Wang et al., 2022). As illustrated in Figure 2(b), we propose a back reconstruction strategy to mitigate the generation of inaccurate dialog states, including encoder-reconstructor and encoder-decoder-reconstructor modules. For the encoder-reconstructor module, the dialog context $C(t) = (B_{t-1}, R_{t-1}, U_t)$ could be reconstructed to enhance encoder information representation by the encoder hidden representation H_{eb}. For the encoder-decoder-reconstructor module, the decoder hidden representation H_{db} could be used to reconstruct the dialog context $C(t)$ to encourage the dialog state decoder to achieve complete information of dialog context.

The dialog state would be reconstructed back to the source input and the corresponding reconstruction score would be calculated to measure the adequacy of the dialog state. The objective function $\mathcal{L}_{BR}$ for the back reconstruction is optimized by minimizing:

$$\begin{aligned}\mathcal{L}_{BR} = &\sum_{i=1}^{N}\sum_{t=1}^{n_i} -logP(C(t)|H_{eb}) \\ &+\sum_{i=1}^{N}\sum_{t=1}^{n_i} -logP(C(t)|H_{db}).\end{aligned} \quad (10)$$

3.2 Denoising Reconstruction

To enhance the denoising capability of the task-oriented dialog system, we propose denoising reconstruction to guarantee that the system learns enough dialog context representation to recover the original version, as illustrated in Figure 2(c). Motivated by denoising auto-encoder strategy that maps a corrupted input back to the original version (Vincent et al., 2010), we introduce noise in the form of random token deleting and masking in the source input to improve the dialog model learning ability. Specifically, we delete or mask every token in the previous dialog state and system response with a probability α. More concretely, we propose two denoising reconstruction modules, i.e., dialog state denoising and response denoising modules.

For the dialog state denoising module, we reconstruct the new Levenshtein dialog state, which means the corrupted part of the dialog state rather than the complete dialog state in the original denoising auto-encoder. The Levenshtein dialog state $\Delta B'_{t-1}$ of dialog turn t, is generated based on the noisy dialog context $N_B(t) = (N(B_{t-1}), R_{t-1}, U_t)$. $N(B_{t-1})$ is the previous corrupted dialog state. For example, the Levenshtein dialog state '*taxi_destination=stevenage train station*' is reconstructed from the corrupted dialog state where '*taxi_destination*' is masked and '*train*' is deleted, as shown in Figure 2(c). For response denoising module, the previous system response R_{t-1} of dialog turn t is reconstructed based on the noisy dialog context $N_R(t) = (N(R_{t-1}), U_t, B_t, DB_{t-1})$. $N(R_{t-1})$ is the previous noisy system response. Therefore, the objective function $\mathcal{L}_{DR}$ for the denoising reconstruction is optimized by minimizing:

$$\begin{aligned}\mathcal{L}_{DR} = &\sum_{i=1}^{N}\sum_{t=1}^{n_i} -logP(\Delta B'_{t-1}|N_B(t)) \\ &+ \sum_{i=1}^{N}\sum_{t=1}^{n_i} -logP(R_{t-1}|N_R(t)).\end{aligned} \tag{11}$$

3.3 Training and Inference Details

There exists inconsistency between the lexicalized user utterance and delexicalized system response, which is used to reduce the influence of different slot values on evaluation (Zhang et al., 2020b). This adds an extra burden for the system to generate a delexicalized system response. To alleviate this issue, we introduce delexicalized user utterances for response generation while lexicalized user utterances are still used for dialog state tracking. For example, '*02:15*' is converted into delexicalized form '*[taxi_arriveby]*' for response generation, as shown in Figure 2(a). Different forms of user utterances take better training of both tasks, ultimately improving task completion.

For inference of dialog state tracking, generated previous dialog state, oracle previous system response, and current user utterance are used as dialog context to generate the current Levenshtein dialog state. For inference of response generation, motivated by Yang et al. (2021), we use generated previous system response instead of oracle previous system response to generate the current system response to maintain coherence throughout the whole dialog session.

4 Experiments

4.1 Datasets and Evaluation Metrics

To establish our proposed end-to-end task-oriented dialog system, we consider two task-oriented dialog datasets, MultiWOZ 2.0 (Budzianowski et al., 2018) and CamRest676 (Wen et al., 2017).

MultiWOZ 2.0 is a large-scale human-to-human multi-domain task-oriented dialog dataset. The dataset consists of seven domains: attraction, hospital, police, hotel, restaurant, taxi, and train. It contains 8438, 1000, and 1000 dialog sessions for training, validation, and testing datasets. Each dialog session covers 1 to 3 domains, and multiple different domains might be mentioned in a single dialog turn. Particularly, there are no hospital and police domains in the validation and testing dataset. To make our experiments comparable with previous work (Zhang et al., 2020b; Lin et al., 2020; Yang et al., 2021), we use the pre-processing script released by Zhang et al. (2020b) and follow the automatic evaluation metrics to evaluate the response quality for the task-oriented dialog system. **Inform** rate measures if a dialog system has provided a correct entity; **Success** rate measures if a dialog system has provided a correct entity and answered all the requested information; **BLEU** score (Papineni et al., 2002) measures the fluency of the generated response; the **combined score**, which is computed by $(Inform+Success)\times 0.5+BLEU$, measures the overall quality of the dialog system. To evaluate the performance of dialog state tracking, we use the **joint goal accuracy** to measure the accuracy of generated dialog states.

CamRest676 is a small-scale restaurant-domain dataset. It contains 408, 136, 132 dialog sessions for training, validation, and testing datasets. To make our experiments comparable with previous work (Lei et al., 2018; Wu et al., 2021), we use the same delexicalization strategy and use **BLEU** score and **Success F1** to evaluate the response qual-

Model	Pre-trained	Inform	Success	BLEU	Combined
DAMD (Zhang et al., 2020b)	n/a	76.3	60.4	16.6	85.0
SimpleTOD (Hosseini-Asl et al., 2020)	DistilGPT2	84.4	70.1	15.0	92.3
MinTL-T5-small (Lin et al., 2020)	T5-small	80.0	72.7	**19.1**	95.5
SOLOIST (Peng et al., 2020)	GPT-2	85.5	72.9	16.5	95.7
MinTL-BART (Lin et al., 2020)	BART-large	84.9	74.9	17.9	97.8
LAVA (Lubis et al., 2020)	n/a	91.8	81.8	12.0	98.8
UBAR* (Yang et al., 2021)	DistilGPT2	91.5	77.4	17.0	101.5
SUMBT+LaRL (Lee et al., 2020)	BERT-base	92.2	85.4	17.9	106.7
Baseline (mask=0)	T5-small	89.0	78.8	17.9	101.8
Baseline (mask=0.15)	T5-small	88.0	77.6	17.7	100.5
BORT	T5-small	**93.8++**	**85.8++**	18.5	**108.3++**

Table 1: Comparison of end-to-end models evaluated on MultiWOZ 2.0. "++" after a score indicates that the proposed BORT is significantly better than Baseline (mask=0) at significance level p <0.01. * denotes the re-evaluated result by the author-released model since the result reported in this original paper (Yang et al., 2021) was evaluated using the ground truth dialog state instead of generated dialog state to query the database entities.

ity for the task-oriented dialog system. The success rate measures if the system answered all the requested information to assess recall while Success F1 balances recall and precision.

4.2 Settings

In the training process for the task-oriented dialog system, we select two backbone models. BORT is a transformer-based system initialized by a pre-trained model. BORT_G is a GRU-based system without a pre-trained model. The detailed training settings and results of the BORT_G backbone are provided in Appendix A.1.

For the BORT backbone, we use pre-trained T5-small (Raffel et al., 2020) to initialize the dialog system, based on the HuggingFace Transformers library (Wolf et al., 2020) and follow the settings of Lin et al. (2020). There are six layers for the encoder and the decoder. The dimension of hidden layers is set to 512, and the head of attention is 8. The batch size is set to 96. The AdamW optimizer (Loshchilov and Hutter, 2019) is used to optimize the model parameters. The learning rate is 0.0025, and the learning rate decay is 0.8. The hyper-parameters λ_1 and λ_2 are set to 0.05 and 0.03, respectively. For the denoising reconstruction strategy, the noise probability α is 0.15. The hyper-parameter analysis is provided in Appendix A.2. Training early stops when no improvement on the combined score of the validation set for five epochs. All results in the low-resource scenario are the average scores of three runs. One P40 GPU is used to train all dialog systems.

4.3 Baselines

Compared with other previous work, our proposed BORT is evaluated in two context-to-response settings: end-to-end modeling to generate dialog state and system response, and policy optimization to generate system response based on ground truth dialog state. Policy optimization results are provided in Appendix A.3.

Sequicity (Lei et al., 2018) and DAMD (Zhang et al., 2020b) are GRU-based end-to-end task-oriented dialog systems with a copy mechanism. Decoder based pre-trained model GPT-2 (Radford et al., 2019) is used in SimpleTOD (Hosseini-Asl et al., 2020), SOLOIST (Peng et al., 2020), and UBAR (Yang et al., 2021). Encoder-decoder based pre-trained model T5 (Raffel et al., 2020) and BART (Lewis et al., 2020) is used in MinTL (Lin et al., 2020). Reinforcement learning is used in LAVA (Lubis et al., 2020) and SUMBT+LaRL (Lee et al., 2020). Especially, SUMBT+LaRL merges a dialog state tracking model SUMBT (Lee et al., 2019) and a dialog policy model LaRL (Zhao et al., 2019) and fine-tune them via reinforcement learning, achieving the state-of-the-art performance.

In addition, we implement two baseline systems. One baseline is the base architecture of a task-oriented dialog system, as illustrated in Figure 2(a). The other is a noise-based baseline system, just masking 15% tokens in the dialog context for dialog training.

4.4 Main Results

Table 1 presents the detailed inform rates, success rates, BLEU scores, and combined scores of end-to-end dialog models on the MultiWOZ 2.0. Our re-implemented baseline system performs better than MinTL (Lin et al., 2020), using the same pre-trained T5-small model. This indicates that the baseline is a strong system. Our proposed BORT significantly outperforms our re-implemented baseline system by 6.5 combined scores, while the simple noise-based method (Baseline (mask=0.15)) doesn't achieve better performance. Moreover, BORT outperforms the previous state-of-the-art SUMBT+LaRL by 1.6 combined scores, achieving the best performance in terms of inform rate, success rate, and combined score. This demonstrates the effectiveness of our proposed BORT.

Model	Success F1	BLEU
Sequicity (Lei et al., 2018)	85.4	25.3
ARDM (Wu et al., 2021)	86.2	25.4
SOLOIST (Peng et al., 2020)	87.1	25.5
BORT	**89.7**	**25.9**

Table 2: Comparison of end-to-end task-oriented dialog systems on CamRest676.

To better assess the generalization capability of BORT, we fine-tune BORT on the CamRest676. The detailed Success F1 and BLEU scores on the CamRest676 are presented in Table 2. Our proposed BORT outperforms the previous state-of-the-art SOLOIST by 2.6 Success F1, achieving the best performance in terms of Success F1. This demonstrates the generalization capability of our proposed BORT.

4.5 Further Evaluation Analysis

Nekvinda and Dušek (2021) identify inconsistencies between previous task-oriented dialog methods in data preprocessing and evaluation metrics and introduce a standalone standardized evaluation script. BLEU score is computed with references, which have been obtained from the delexicalized MultiWOZ 2.2 span annotations.

Model	Inform	Success	BLEU	Combined
DAMD (Zhang et al., 2020b)	57.9	47.6	16.4	69.2
LABES (Zhang et al., 2020a)	68.5	58.1	18.9	82.2
AuGPT (Kulhánek et al., 2021)	76.6	60.5	16.8	85.4
MinTL-T5-small (Lin et al., 2020)	73.7	65.4	**19.4**	89.0
SOLOIST (Peng et al., 2020)	82.3	72.4	13.6	91.0
DoTS (Jeon and Lee, 2021)	80.4	68.7	16.8	91.4
UBAR (Yang et al., 2021)	83.4	70.3	17.6	94.5
BORT	**85.5**	**77.4**	17.9	**99.4**

Table 3: Comparison of end-to-end task-oriented dialog systems evaluated on the standardized setting (Nekvinda and Dušek, 2021).

To get a more complete picture of the effectiveness of reconstruction strategies, we also use this evaluation script to evaluate our proposed BORT which is trained on MultiWOZ 2.0. As shown in Table 3, BORT also substantially outperforms the previous state-of-the-art UBAR by a large margin (4.9 combined scores), achieving the best performance in terms of inform rate, success rate, and combined score. This further demonstrates the effectiveness of our proposed BORT.

4.6 Ablation Study

We empirically investigate the performance of the different components of BORT as shown in Table 4. Our introduced user utterance delexicalization strategy gains 1.9 combined scores, indicating the effectiveness of the user utterance delexicalization strategy. Back reconstruction performs slightly better than denoising reconstruction by 1 combined score regarding the two proposed reconstruction strategies. Moreover, the combination of both reconstruction strategies can complement each other to further improve the performance of the dialog system. The detailed analysis on different modules of every reconstruction strategy is provided in Appendix A.4.

Model	Inform	Success	BLEU	Combined
BORT	93.8	85.8	18.5	108.3
w/o DR	92.9	84.0	18.8	107.3
w/o BR	92.0	84.4	18.1	106.3
w/o BR & DR	90.4	81.4	17.8	103.7
w/o BR & DR& UD	89.0	78.8	17.9	101.8

Table 4: The performance of the different components of our proposed BORT on MultiWOZ 2.0. BR denotes back reconstruction, DR denotes denoising reconstruction, UD denotes user utterance delexicalization.

4.7 Dialog State Tracking

Table 5 reports the dialog state tracking performance of the end-to-end task-oriented dialog systems on MultiWOZ 2.0. BORT substantially outperforms MinTL (Lin et al., 2020) using the same pre-trained T5-small model by 2.8 points, achieving 54.0 joint goal accuracy. Moreover, BORT achieves the highest joint goal accuracy among the end-to-end task-oriented dialog systems. This indicates that our proposed reconstruction strategies could improve dialog state learning ability.

4.8 Case Study and Human Evaluation

Moreover, we analyze translation examples and conduct a human evaluation to further explore the

Model	Joint Accuracy
MinTL-T5-small (Lin et al., 2020)	51.2
SUMBT+LaRL (Lee et al., 2020)	51.5
MinTL-BART (Lin et al., 2020)	52.1
UBAR (Yang et al., 2021)	52.6
SOLOIST (Peng et al., 2020)	53.2
BORT	**54.0**

Table 5: The dialog state tracking performance of end-to-end task-oriented dialog systems on MultiWOZ 2.0.

effectiveness of BORT. Figure 3 shows an example generated by MinTL and BORT, respectively. More examples are provided in Appendix A.5. MinTL generates the response to request for the preferred area about college since it generates an inaccurate dialog state '*attraction_type=college*' rather than correct dialog state '*attraction_name=jesus college*'. In contrast, BORT generates an accurate dialog state, achieving the appropriate response that provides the information of *jesus college*. These further demonstrate the effectiveness of our proposed reconstruction strategies.

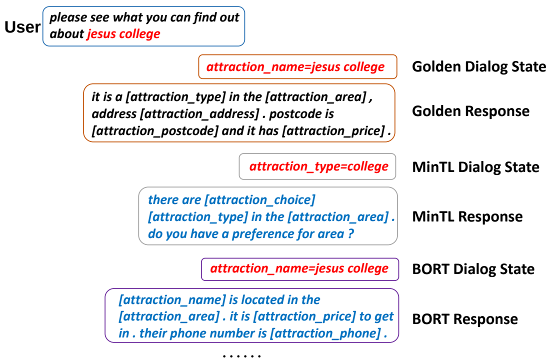

Figure 3: An example of the task-oriented dialog systems in dialog session PMUL4025.

For human evaluation, we manually evaluate the quality of generated responses on 50 dialog sessions, which are randomly extracted from the MultiWOZ 2.0 testing set. We consider the fluency and appropriateness of the generated response based on scores ranging from 1 to 5. The fluency metric measures whether the generated response is fluent. The appropriateness metric measures whether the generated response is appropriate and the system understands the user's goal. Three fluent English speakers are asked to evaluate these generated responses. The average scores evaluated by them are shown in Table 6. The results are consistent with the automatic evaluation, indicating that BORT could improve the quality of generated response.

Model	Fluency	Appropriateness
MinTL-T5-small	4.50	3.88
UBAR	4.50	3.81
BORT	**4.55**	**3.98**

Table 6: The human evaluation of the end-to-end task-oriented dialog systems on MultiWOZ 2.0.

4.9 Domain Adaptation Analysis

To investigate the domain adaptation ability of BORT to generalize to some unseen domains, we simulate zero-shot experiments by excluding one domain and training BORT on other domains. As shown in Table 7, the train and taxi domains achieve the highest combined scores because they have a high overlap in ontology with other domains. In addition, BORT and MinTL with an encoder-decoder-based pre-trained model achieve significantly better domain adaptation performance than DAMD without a pre-trained model and UBAR with a decoder-based pre-trained model, which demonstrates the encoder-decoder based pre-trained model have better domain transferability. Moreover, our proposed reconstruction strategy could further improve combined scores in the zero-shot domain scenario.

Model	Attraction	Hotel	Restaurant	Taxi	Train
DAMD	28.7	26.9	24.4	52.3	51.4
UBAR	28.3	29.5	23.5	59.5	53.9
MinTL	33.4	37.3	31.5	60.4	77.1
BORT	**33.6**	**38.7**	**32.0**	**62.7**	**85.6**

Table 7: Comparison of combined scores on the MultiWOZ 2.0 in the zero-shot domain scenario.

4.10 Low Resource Scenario Analysis

To better assess the robustness of our proposed BORT, we choose 5%, 10%, 20%, and 30% of training dialog sessions to investigate the performance of task-oriented dialog systems in the low resource scenario. As shown in Table 8, BORT substantially outperforms other methods in these low-resource scenarios. This is because the error propagation problem in the low resource scenario is more serious, while BORT could effectively alleviate the error propagation problem. Moreover, our proposed BORT trained on the 30% dataset performs comparably to some baseline systems trained on all datasets as shown in Table 1. These further demonstrate that our proposed BORT is robust, alleviating poor performance in the low-resource

Model	5%				10%				20%				30%			
	Inform	Success	BLEU	Combined	Inform	Success	BLEU	Combined	Inform	Success	BLEU	Combined	Inform	Success	BLEU	Combined
DAMD	49.1	23.7	11.3	47.7	57.6	32.6	12.0	57.1	64.7	45.0	15.3	70.2	64.5	47.3	15.5	71.4
UBAR	35.7	21.2	11.0	39.5	62.4	43.6	12.7	65.7	76.2	58.3	14.1	81.4	81.2	65.4	14.7	88.0
MinTL	55.2	40.9	**13.9**	62.0	67.7	55.7	15.3	77.0	66.7	57.9	**17.3**	79.6	74.9	66.5	**17.3**	88.0
BORT	**69.8**	**45.9**	11.0	**68.9**	**74.5**	**60.6**	**15.5**	**83.1**	**82.1**	**65.5**	14.3	**88.1**	**83.8**	**69.9**	17.2	**94.1**

Table 8: Comparison of task-oriented dialog systems on the MultiWOZ 2.0 in the low resource scenarios.

scenario.

4.11 Error Propagation Analysis

To investigate the denoising capability of our proposed BORT, we perform the simulated experiments, where noise is added in the oracle dialog state for the policy optimization evaluation. In detail, we replace every token in the oracle dialog state with the masked token with a probability to generate synthetic noise. As shown in Figure 4, BORT performs substantially better than the baseline system in the noisy scenario. In particular, as the noise ratio in the oracle dialog state increases, the performance gap between the baseline system and BORT increases. When noise proportion is 0, BORT still performs better than the baseline system because BORT generates more appropriate response via the denoising reconstruction strategy. These demonstrate that our proposed BORT is robust and effective.

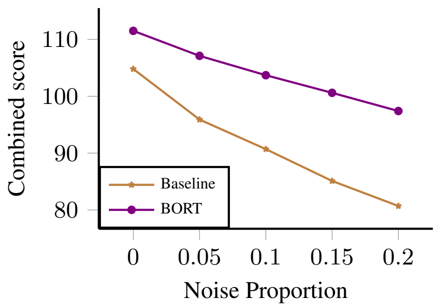

Figure 4: The policy optimization performance (combined score) of baseline and BORT as the noise in the oracle dialog state increases on the MultiWOZ 2.0.

5 Related Work

End-to-end task-oriented dialog system has attracted much attention in the dialog community. A two-stage copynet framework was proposed to establish an end-to-end task-oriented dialog system based on a single sequence-to-sequence model (Lei et al., 2018). Zhang et al. (2020b) proposed a multi-action data augmentation framework to improve the diversity of dialog responses. Recently, large-scale language model pre-training has been effective for enhancing many natural language processing tasks (Peters et al., 2018; Radford et al., 2018; Devlin et al., 2019). Decoder-based pre-trained language model such as GPT-2 (Radford et al., 2019) was used to improve the performance of end-to-end task-oriented dialog system (Budzianowski and Vulić, 2019; Hosseini-Asl et al., 2020; Peng et al., 2020; Yang et al., 2021). The Levenshtein dialog state instead of the dialog state was generated to reduce the inference latency (Lin et al., 2020). In addition, they used encoder-decoder-based pre-trained model such as T5 (Raffel et al., 2020) and BART (Lewis et al., 2020) to establish a dialog system. In contrast with previous work, in which system response was generated, Wu et al. (2020) used encoder-based pre-trained model such as BERT (Devlin et al., 2019) for task-oriented dialog system, aiming to retrieve the most relative system response from a candidate pool. Reinforcement learning could also be used to enable task-oriented dialog systems to achieve more successful task completion (Lubis et al., 2020; Lee et al., 2020).

Tu et al. (2017) proposed an encoder-decoder-reconstructor framework for neural machine translation to alleviate over-translation and under-translation problems. Reconstruction strategy was used to moderate dropped pronoun translation problems (Wang et al., 2018). In contrast, we considered the adequacy of semantic representations rather than natural language sentences to build the reconstruction model. Vincent et al. (2010) proposed a denoising autoencoder, in which random noise is added to enhance the robustness of the model, alleviating the overfitting problem of traditional auto-encoder. The denoising auto-encoder strategy was used as the language model to generate more fluent translation candidates for the unsupervised neural machine translation (Artetxe et al., 2018; Lample et al., 2018; Sun et al., 2019). In addition, a denoising auto-encoder was used to pre-train sequence-to-sequence models on the large scale corpus (Lewis et al., 2020; Liu et al., 2020). In contrast, we proposed a denoising reconstruction mechanism to alleviate the error propagation problem along the multi-turn conversation flow.

6 Conclusion

This paper proposes back and denoising reconstruction strategies for the end-to-end task-oriented dialog system. Back reconstruction strategy is proposed to mitigate the generation of inaccurate dialog states, achieving better task completion of the task-oriented dialog system. Denoising reconstruction is used to train a robust task-oriented dialog system, further alleviating the error propagation problem. Our extensive experiments and analysis on MultiWOZ 2.0 and CamRest676 demonstrate the effectiveness of our proposed reconstruction strategies.

Acknowledgments

We are grateful to the anonymous reviewers and the area chair for their insightful comments and suggestions. This work is supported by the National Key Research and Development Program of China under Grant No. 2020AAA0108600.

References

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised neural machine translation. In *Proceedings of the 6th International Conference on Learning Representations*. OpenReview.net.

Paweł Budzianowski and Ivan Vulić. 2019. Hello, it's GPT-2 - how can I help you? towards the use of pretrained language models for task-oriented dialogue systems. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 15–22. Association for Computational Linguistics.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026. Association for Computational Linguistics.

Wenhu Chen, Jianshu Chen, Pengda Qin, Xifeng Yan, and William Yang Wang. 2019. Semantically conditioned dialog response generation via hierarchical disentangled self-attention. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3696–3709. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. In *Advances in Neural Information Processing Systems 33*, pages 20179–20191. Curran Associates, Inc.

Hyunmin Jeon and Gary Geunbae Lee. 2021. Domain state tracking for a simplified dialogue system. *CoRR*, abs/2103.06648.

Jonás Kulhánek, Vojtech Hudecek, Tomás Nekvinda, and Ondrej Dusek. 2021. Augpt: Dialogue with pre-trained language models and data augmentation. *CoRR*, abs/2102.05126.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049. Association for Computational Linguistics.

Hwaran Lee, Seokhwan Jo, HyungJun Kim, Sangkeun Jung, and Tae-Yoon Kim. 2020. Sumbt+larl: End-to-end neural task-oriented dialog system with reinforcement learning. *CoRR*, abs/2009.10447.

Hwaran Lee, Jinsik Lee, and Tae-Yoon Kim. 2019. SUMBT: Slot-utterance matching for universal and scalable belief tracking. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5478–5483. Association for Computational Linguistics.

Wenqiang Lei, Xisen Jin, Min-Yen Kan, Zhaochun Ren, Xiangnan He, and Dawei Yin. 2018. Sequicity: Simplifying task-oriented dialogue systems with single sequence-to-sequence architectures. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1437–1447. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880. Association for Computational Linguistics.

Zhaojiang Lin, Andrea Madotto, Genta Indra Winata, and Pascale Fung. 2020. MinTL: Minimalist transfer learning for task-oriented dialogue systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 3391–3405. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of the 7th International Conference on Learning Representations*. OpenReview.net.

Nurul Lubis, Christian Geishauser, Michael Heck, Hsien-chin Lin, Marco Moresi, Carel van Niekerk, and Milica Gasic. 2020. LAVA: Latent action spaces via variational auto-encoding for dialogue policy optimization. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 465–479. International Committee on Computational Linguistics.

Tomáš Nekvinda and Ondřej Dušek. 2021. Shades of BLEU, flavours of success: The case of MultiWOZ. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics*, pages 34–46. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Baolin Peng, Chunyuan Li, Jinchao Li, Shahin Shayandeh, Lars Liden, and Jianfeng Gao. 2020. Soloist: Building task bots at scale with transfer learning and machine teaching. *arXiv preprint arXiv:2005.05298*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf*.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Haipeng Sun, Rui Wang, Kehai Chen, Masao Utiyama, Eiichiro Sumita, and Tiejun Zhao. 2019. Unsupervised bilingual word embedding agreement for unsupervised neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1235–1245. Association for Computational Linguistics.

Zhaopeng Tu, Yang Liu, Lifeng Shang, Xiaohua Liu, and Hang Li. 2017. Neural machine translation with reconstruction. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 3097–3103. AAAI Press.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11:3371–3408.

Jianhong Wang, Yuan Zhang, Tae-Kyun Kim, and Yunjie Gu. 2021. Modelling hierarchical structure between dialogue policy and natural language generator with option framework for task-oriented dialogue system. In *Proceedings of the 9th International Conference on Learning Representations*. OpenReview.net.

Longyue Wang, Zhaopeng Tu, Shuming Shi, Tong Zhang, Yvette Graham, and Qun Liu. 2018. Translating pro-drop languages with reconstruction models. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pages 4937–4945. AAAI Press.

Yifan Wang, Jing Zhao, Junwei Bao, Chaoqun Duan, Youzheng Wu, and Xiaodong He. 2022. Luna: Learning slot-turn alignment for dialogue state tracking. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45. Association for Computational Linguistics.

Chien-Sheng Wu, Steven C.H. Hoi, Richard Socher, and Caiming Xiong. 2020. TOD-BERT: Pre-trained natural language understanding for task-oriented dialogue. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 917–929. Association for Computational Linguistics.

Qingyang Wu, Yichi Zhang, Yu Li, and Zhou Yu. 2021. Alternating recurrent dialog model with large-scale pre-trained language models. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1292–1301. Association for Computational Linguistics.

Yunyi Yang, Yunhao Li, and Xiaojun Quan. 2021. UBAR: towards fully end-to-end task-oriented dialog system with GPT-2. In *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence*, pages 14230–14238. AAAI Press.

Yichi Zhang, Zhijian Ou, Min Hu, and Junlan Feng. 2020a. A probabilistic end-to-end task-oriented dialog model with latent belief states towards semi-supervised learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 9207–9219. Association for Computational Linguistics.

Yichi Zhang, Zhijian Ou, and Zhou Yu. 2020b. Task-oriented dialog systems that consider multiple appropriate responses under the same context. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, pages 9604–9611. AAAI Press.

Zheng Zhang, Ryuichi Takanobu, Minlie Huang, and Xiaoyan Zhu. 2020c. Recent advances and challenges in task-oriented dialog system. *CoRR*, abs/2003.07490.

Tiancheng Zhao, Kaige Xie, and Maxine Eskenazi. 2019. Rethinking action spaces for reinforcement learning in end-to-end dialog agents with latent variable models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1208–1218. Association for Computational Linguistics.

A Appendix

A.1 BORT_G Settings and Results

For the BORT_G backbone, we follow the settings of Zhang et al. (2020b). We use one layer bi-directional GRU for the encoder and the decoder. The dimension of hidden layers is set to 100. The batch size is 128. The AdamW optimizer (Loshchilov and Hutter, 2019) is used to optimize the model parameters, and the learning rate is 0.005.

Model	Inform	Success	BLEU	Combined
DAMD	76.3	60.4	16.6	85.0
MinTL-T5-small	80.0	72.7	19.1	95.5
BORT_G	87.3	75.8	18.4	100.0

Table 9: Comparison of end-to-end models evaluated on MultiWOZ 2.0.

As shown in Table 9, BORT_G performs better than DAMD without a pre-trained model, achieving the improvement of 15.0 combined scores, even though multi-action data augmentation is not used in BORT_G. Moreover, BORT_G outperforms MinTL (Lin et al., 2020), using the pre-trained model. This demonstrates the effectiveness and applicability of our proposed reconstruction strategies.

A.2 Hyper-parameter Analysis

In Figure 5, we empirically investigate how the hyper-parameters in Eq. 9 affects the dialog performance on the MultiWOZ 2.0 validation set. The selection of hyper-parameters λ_1 and λ_2 influence the role of the $\mathcal{L}_{BR}$ and $\mathcal{L}_{DR}$ across the entire end-to-end task-oriented dialog training process. Larger values of λ_1 or λ_2 cause the $\mathcal{L}_{BR}$ or $\mathcal{L}_{DR}$ to play a more important role than the original task-oriented dialog loss terms. The smaller the value of λ_1 or λ_2, the less important is the $\mathcal{L}_{BR}$ or $\mathcal{L}_{DR}$. As Figure 5 shows, λ_1 ranging from 0.01 to 0.5 nearly all enhances task-oriented dialog performance, and when λ_2 is larger than 0.3, the performance underperforms the baseline system. When $\lambda_1 = 0.05$ and $\lambda_2 = 0.03$, our proposed BORT achieves the best performance on the validation set.

In addition, the influence of noise type and noise proportion on the performance of our proposed BORT on the MultiWOZ 2.0 validation set is empirically investigated, as shown in Figure 6. Both deletion and masking noise strategies could improve the dialog performance. In particular, their

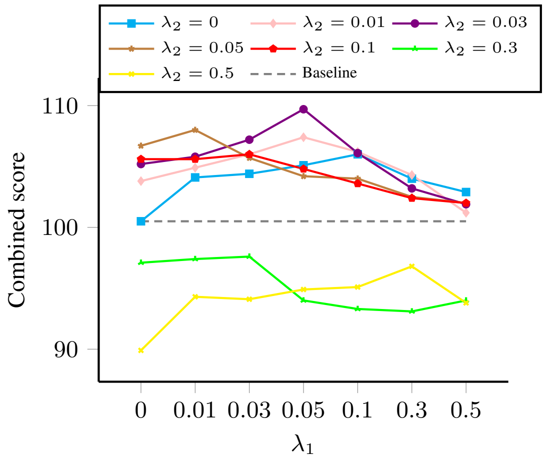

Figure 5: BORT performance (combined score) with different levels of hyper-parameters on the MultiWOZ 2.0 validation set.

combination is further better than both of them. This demonstrates that both noise strategies can complement each other to further improve the dialog performance. As shown in Figure 6, when the noise proportion is 0.15, our proposed BORT achieves the best performance on the validation set.

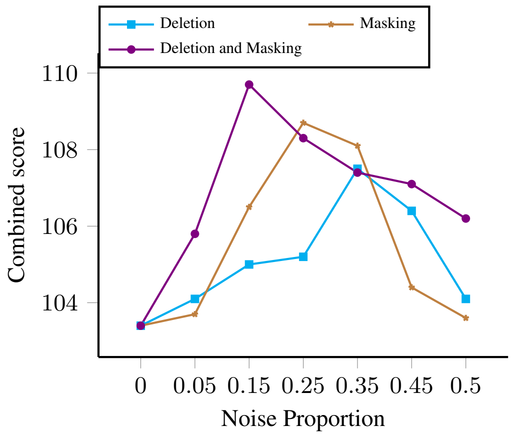

Figure 6: BORT performance (combined score) with different levels of noise type and noise proportion on the MultiWOZ 2.0 validation set.

A.3 Policy Optimization Evaluation

The detailed inform rates, success rates, BLEU scores, and combined scores of policy optimization dialog models on the MultiWOZ 2.0 are presented in Table 10. The ground truth dialog state is used for the policy optimization setting to query the database entities and generate system responses. Our proposed BORT achieves performance comparable to the state-of-the-art LAVA in terms of inform rate. In addition, compared with previous policy optimization methods, BORT achieves bet-

Model	Pre-trained	Inform	Success	BLEU	Combined
LaRL (Zhao et al., 2019)	n/a	82.8	79.2	12.8	93.8
SimpleTOD (Hosseini-Asl et al., 2020)	DistilGPT2	88.9	67.1	16.9	94.9
HDSA (Chen et al., 2019)	BERT-base	82.9	68.9	**23.6**	99.5
ARDM (Wu et al., 2021)	GPT-2	87.4	72.8	20.6	100.7
DAMD (Zhang et al., 2020b)	n/a	89.2	77.9	18.6	102.2
SOLOIST (Peng et al., 2020)	GPT-2	89.6	79.3	18.0	102.5
UBAR (Yang et al., 2021)	DistilGPT2	94.0	83.6	17.2	106.0
LAVA (Lubis et al., 2020)	n/a	**97.5**	**94.8**	12.1	108.3
HDNO (Wang et al., 2021)	n/a	96.4	84.7	18.9	109.5
BORT_G	n/a	89.6	80.5	19.1	104.2
BORT	T5-small	96.1	88.8	19.0	**111.5**

Table 10: Comparison of policy optimization models evaluated on MultiWOZ 2.0.

ter performance in terms of the combined score even though BORT has not modeled action learning.

Compared with previous works, BORT achieves much more significant improvement in the end-to-end setting rather than policy optimization setting because our proposed reconstruction strategies pay more attention to improving the quality of dialog state while the golden dialog state is used in the policy optimization setting.

A.4 Ablation Study

Moreover, we further investigate the performance of the different components of the two proposed reconstruction strategies, respectively. As shown in Table 11, encoder-decoder-reconstructor module for back reconstruction strategy significantly outperforms encoder-reconstructor module by 2.2 combined scores because dialog state decoder could achieve more dialog context information for encoder-decoder-reconstructor. In addition, regarding two denoising reconstruction modules, dialog state denoising and response denoising have achieved similar performance. These two modules could improve the denoising capability of the task-oriented dialog system.

Model	Inform	Success	BLEU	Combined
Back reconstruction	92.9	84.0	18.8	107.3
w/o enc-rec	92.2	83.5	19.0	106.9
w/o enc-dec-rec	92.1	81.2	18.0	104.7
Denoising reconstruction	92.0	84.4	18.1	106.3
w/o dialog state denoising	91.7	83.0	17.9	105.3
w/o response denoising	92.8	81.2	18.6	105.6

Table 11: The performance of the different components of the two proposed reconstruction strategies. enc-dec denotes encoder-reconstructor module, enc-dec-rec denotes encoder-decoder-reconstructor module.

A.5 More examples

Figures 7 - 11 show several examples generated by MinTL and BORT, respectively. As shown in Figure 7, MinTL generates the inadequate dialog state, which may provide the hotel without internet. Our proposed BORT reconstructs the generated dialog state back to the original input context to ensure the information in the input side is completely transformed to the output side to achieve an adequate dialog state via a back reconstruction strategy. Figure 8 shows that our proposed BORT generated the correct slot value '*european*' rather than the corrupted one '*europeon*' from the corrupted dialog context, indicating the robustness of the denoising reconstruction strategy. As shown in Figures 9 - 11, MinTL generates the inaccurate dialog state, leading to the inaccurate response. The results are consistent with our opinion that the generated dialog state, which is crucial for task completion of a task-oriented dialog system, has always been inaccurate across the end-to-end task-oriented dialog system training. Moreover, Figure 11 shows that MinTL faces the problem of error propagation from both previously generated inaccurate dialog states and responses. Our proposed BORT can alleviate these issues via reconstruction strategies, further demonstrating the effectiveness of BORT.

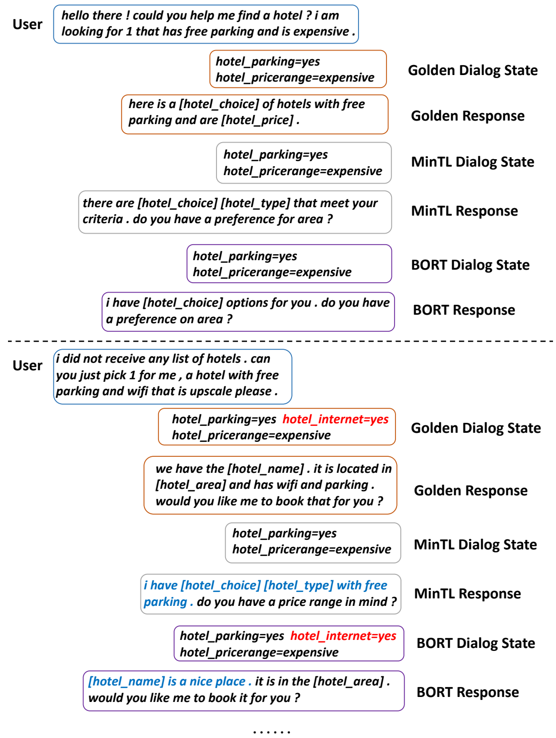

Figure 7: An example of the task-oriented dialog systems in dialog session MUL1139.

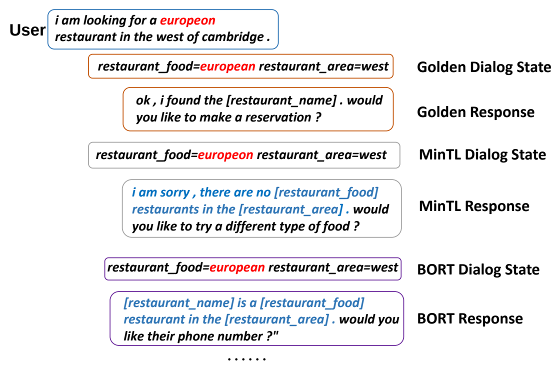

Figure 8: An example of the task-oriented dialog systems in dialog session PMUL0095.

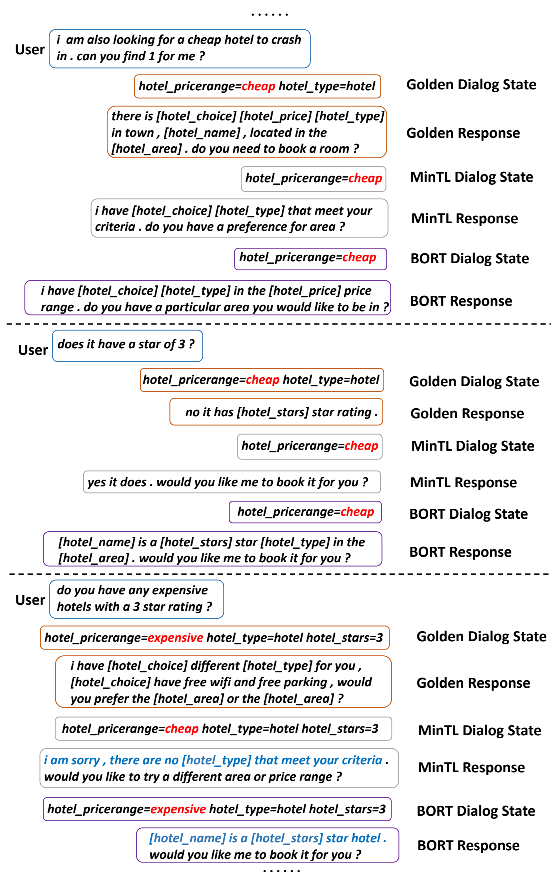

Figure 9: An example of the task-oriented dialog systems in dialog session PMUL3868.

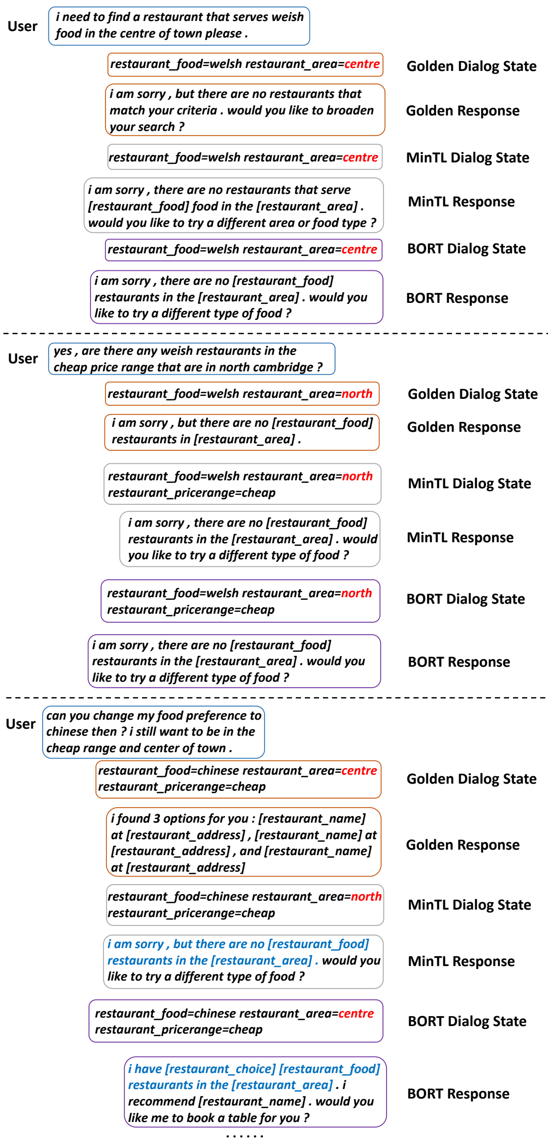

Figure 10: An example of the task-oriented dialog systems in dialog session MUL0286.

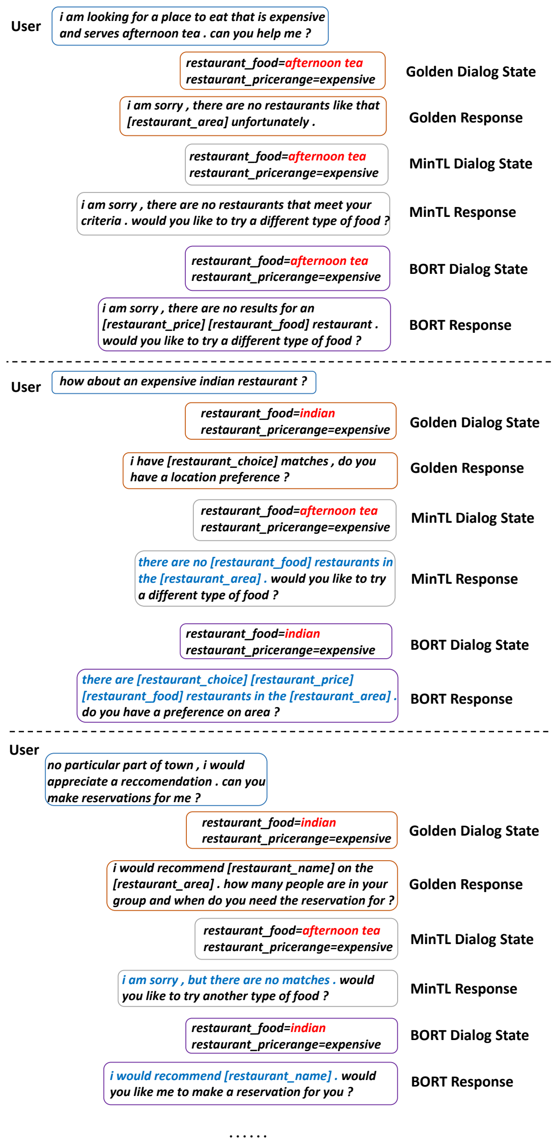

Figure 11: An example of the task-oriented dialog systems in dialog session PMUL3875.