

Article

# TodBR: Target-Oriented Dialog with Bidirectional Reasoning on Knowledge Graph

Zongfeng Qu <sup>1,2</sup>, Zhitong Yang <sup>3,4</sup>, Bo Wang <sup>1,\*</sup> and Qinghua Hu <sup>1</sup><sup>1</sup> College of Intelligence and Computing, Tianjin University, Tianjin 300072, China; quzf@cheari.com (Z.Q.)<sup>2</sup> China Household Electric Appliances Research Institute, Beijing 100037, China<sup>3</sup> School of New Media and Communication, Tianjin University, Tianjin 300072, China<sup>4</sup> State Grid Customer Service Center, Tianjin 300300, China

\* Correspondence: bo\_wang@tju.edu.cn

**Featured Application:** This article proposes a reasoning-based dialog agent to facilitate dialog goal accomplishment through natural language interaction. The model can be applied in the conversation recommendation, topic guidance, psychotherapy and education domains.

**Abstract:** Target-oriented dialog explores how a dialog agent connects two topics cooperatively and coherently, which aims to generate a “bridging” utterance connecting the new topic to the previous conversation turn. The central focus of this task entails multi-hop reasoning on a knowledge graph (KG) to achieve the desired target. However, current target-oriented dialog approaches suffer from inefficiencies in reasoning and the inability to locate pertinent key information without bidirectional reason. To address these limitations, we present a bidirectional reasoning model for target-oriented dialog implemented on a commonsense knowledge graph. Furthermore, we introduce an automated technique for constructing dialog subgraphs, which aids in acquiring multi-hop reasoning capabilities. Our experiments demonstrate that our proposed method attains superior performance in reaching the target while providing more coherent responses.

**Keywords:** target-oriented dialog; response generation; topic guidance; graph reasoning; natural language processing



**Citation:** Qu, Z.; Yang, Z.; Wang, B.; Hu, Q. TodBR: Target-Oriented Dialog with Bidirectional Reasoning on Knowledge Graph. *Appl. Sci.* **2024**, *14*, 459. <https://doi.org/10.3390/app14010459>

Academic Editor: Valentino Santucci

Received: 30 November 2023

Revised: 24 December 2023

Accepted: 30 December 2023

Published: 4 January 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Recently, dialog agents have been classified into three categories: open-domain [1–4], task-oriented [5,6], and target-oriented dialog systems [7–12]. Open-domain and task-oriented dialog systems focus on chit-chat dialog and task fulfillment, whereas their conversational guidance remains passive rather than active. However, target-oriented dialog systems aim to proactively introduce new topics and guide the dialog toward a predefined target topic. Specifically, when given a source and target topic expressed as two brief utterances, the target-oriented dialog (TOD) agent must generate a transitional response that connects the two topics seamlessly. Due to their purposeful and flexible nature, target-oriented dialog agents possess a wide range of applications, spanning conversational recommendation [13], psychotherapy [14], and education [15].

A typical example of a target-oriented dialog scenario is demonstrated in Table 1 as follows: the source topic is “I like going to concerts”, and the target topic is “I usually have to take a break during the week”. In this scenario, the dialog agent is expected to generate an appropriate transition response that links the source topic “concert” to the target topic “work”, and the transition is “I want to relax completely after a week’s work is done”. The transition response helps to seamlessly shift the focus of the conversation from the initial topic to the target topic.

**Table 1.** An example of target-oriented dialog.

User A	Source Topic:	I like going to concerts.
User B	Transition:	Me too! I usually have to take a break during the week.
	Target Topic:	I want to relax completely after a week's work is done.
	Entity Path:	concerts—break—work

In the domain of target-oriented dialog, the mainstream approach to address the topic transitions issue is using reasoning techniques based on a knowledge graph (KG). These methods can be categorized into two main types: supervised learning [10,16,17] and reinforcement learning [18,19]. Reinforcement learning (RL) methods employ a policy-based agent incrementally, extending its reasoning path until it reaches the target entity. The agent's state is typically defined as a tuple consisting of the query and the current entity, and its action involves traversing the knowledge graph by following the outgoing edges of the current entity. RL-based approaches heavily depend on the reward design strategy [20,21], so reward shaping and action dropout have been proposed to improve path searching and model learning. On the other hand, supervised-based methods heavily rely on costly annotations of reasoning signals [22]. These methods can be further divided into generating an entity path that connects the initial query to the target entity and ranking entities based on their relevance to the user's context.

While the mainstream methods mentioned above work well, there are still a couple of issues: (1) the current external knowledge graph, especially the common sense graph, has millions of nodes, affecting the efficiency and effectiveness of multi-hop reasoning; (2) current methods do not fully take into account the impact of the source and target information, leading to a dialog that feels unnatural or inefficient. Our approach tackles this problem by using bidirectional reasoning.

To this end, we propose **TodBR** (Target-oriented dialog with Bidirectional Reasoning on Knowledge Graph), aiming to effectively support the target-oriented process through bidirectional reasoning on a knowledge graph enclosing the target topic. Our approach considers two distinct perspectives when addressing the task setting: forward reasoning, which explores the path from source entities to target entities, and backward reasoning, which follows the reverse path from target entities to source entities. Our solution draws inspiration from bidirectional graph search algorithms like bidirectional BFS and DFS. While existing methods predominantly focus on forward reasoning, it is feasible to jointly model the two reasoning processes, as both topic entities and target entities are known at the start of the dialog. The bidirectional reasoning mechanism effectively reduces the search space and enhances the reasoning efficiency. In addition, to further improve both the performance and efficiency, we employ a pre-built context-relevant dialog subgraph and conduct reasoning on it. This approach demonstrates clear advantages over reasoning on the entire knowledge base.

Our main contributions are as follows.

- (1) We introduce an automated method of constructing dialog subgraphs that aid target-oriented dialog learning. These dialog subgraphs play a crucial role in this task as they provide logically organized entities and their relations, which are instrumental in reasoning the appropriate response keyword.
- (2) We take the first step towards bidirectional reasoning and propose a corresponding model. By incorporating bidirectional reasoning, we enable a more comprehensive and efficient exploration of the knowledge graph, leading to improved results.
- (3) We conduct evaluations using both automatic and human metrics, which demonstrate that our proposed approach, TodBR, outperforms the baselines in terms of the coherence of semantics and achieving the desired targets with a higher success rate.

The structure of this paper is as follows. Section 1 introduces the topic and the motivation behind target-oriented dialog systems. Section 2 explores existing mainstream

methods employed in target-oriented dialog systems and discusses multi-hop reasoning methods applied in conversation. Section 3 delves into the comprehensive description of the proposed algorithms. Then, Section 4 describes the dataset used for the task and the results obtained for both algorithms and conducts quantitative and qualitative analysis. Finally, Section 5 concludes the article by summarizing the essential findings and insights gained from the study and suggests potential directions for future research.

## 2. Related Work

### 2.1. Target-Oriented Dialog Systems

The existing research on target-oriented dialog can generally be categorized into two approaches: local target-oriented and global target-oriented methods. The local target-oriented [7,10,16] methods primarily focus on the next-turn target. For instance, Xu [18] proposed a hierarchical policy model that facilitates the planning and generation of responses at different levels; a high-level policy is responsible for planning a topic, while a low-level policy generates coherent replies to the selected topic. However, these methods do not explicitly approach the global target. Moreover, these approaches often employ a short-sighted and greedy strategy instead of incorporating explicit planning to optimize the process towards achieving the global target. Global target-oriented methods [23–26] employ a global target to guide the generation of responses at each turn. These methods introduce a keyword predictor to determine the keyword to be discussed in the next turn and generate a relevant response to the identified keyword.

### 2.2. Conversation with Knowledge Graph Reasoning

In recent years, there has been growing interest in incorporating commonsense knowledge and employing graph reasoning techniques to generate relevant and informative responses. Previous research [27] on knowledge graph (KG) reasoning can be broadly categorized into three main approaches: path-based models [28], embedding-based models [29], and models that combine both embedding and path-based techniques to predict missing links in the knowledge graph [30,31]. In our study, we align ourselves with the third category of research, as it best suits our objective of knowledge selection on a graph. Furthermore, our task differs from previous works due to considering the target topic vertices within the graph. This distinction is a key motivation for utilizing bidirectional reasoning in graph reasoning. By leveraging bidirectional reasoning, we aim to enhance the effectiveness of our approach.

Our task often faces the challenge of response generation relying on external knowledge bases or corpora. This is a common issue in question-answering (QA) tasks [32]. In specific QA tasks, the fusion of knowledge graphs (KG) and text corpora is commonly approached through late or early fusion strategies [33,34]. These strategies aim to overcome the limited answer coverage of models that solely rely on KGs. Some works have also introduced advanced retrieval-augmented generative models for open-domain question-answering tasks [35,36]. We also employ a fusion mechanism to incorporate external knowledge and prevent zero-shot scenarios in our task. By leveraging the fusion strategy, we can introduce relevant external knowledge to improve the performance and coverage of our model. This approach ensures that our system can handle a broader range of topics and generate responses informed by external knowledge bases.

## 3. Methods

This section focuses on the solution developed, starting with describing the data used in the study. Subsequently, the employed methodology is outlined and discussed. It begins by presenting the formal definition of the tasks at hand. Then, the data preprocessing methods used to construct dialog subgraphs are introduced. Finally, the bidirectional reasoning model is presented in detail.

### 3.1. Task Definition

We define the target-oriented response generation task as follows:

$$\hat{y} = \operatorname{argmax}_y P(y | c, t, G) \quad (1)$$

Here,  $c$  represents the conversation context,  $G$  is a knowledge subgraph associated with the context,  $t$  is the dialog target, and  $y$  is the transition response representing the model's output that connects the conversation context  $c$  and the target  $t$ . Equation (1) signifies that  $\hat{y}$  is determined by selecting the response  $y$  that maximizes the conditional probability. In simpler terms, the argmax function is used to find the response with the highest probability of being the correct or most suitable transition response given the context, target, and associated knowledge subgraph. Specifically, our method generates a prompt keyword set  $z$  for transition based on subgraph  $G$ , which is extracted on the knowledge graph ConceptNet. Then, we integrate the keywords  $z$  and the conversation context  $c$  to generate a proper transition response  $y$ .

### 3.2. Method Overview

**Three Steps** Our method is divided into three steps: data preprocessing, training, and prediction. The data preprocessing involves dialog data cleaning, keyword extraction (the red font in Figure 1 is an example for keywords extraction), and dialog subgraph construction. During the training phase, a supervised learning method minimizes negative log-likelihood optimization. The prediction stage is based on the trained model and used with the beam-search [37] algorithm to generate the response.

**Main Components** Our model comprises three main components, as shown in Figure 1: an encoder and two decoders. The first component is the dialog encoder, which utilizes a Transformer encoder to encode both the source and target utterances into an embedding vector. The encoder plays a fundamental role in capturing the semantic information of the dialogs, which arises from its attention mechanism for long-range dependency capture, scalability, and parallelization, coupled with positional encoding and self-attention for semantic comprehension. Our method's following steps (bidirectional reasoning and response generation) are based on the output embedding generated by the dialog encoder. The second component is a bidirectional reasoning decoder. A graph neural network initially encodes the dialog subgraph constructed during the data preprocessing stage. Following this, the dialog embedding vector obtained from the dialog encoder is input into the decoder, and an attention mechanism [38] is applied to focus on the dialog subgraph and generate prompt keywords. The third component is a response generator based on GPT. It inputs the dialog context and prompt keywords, generating the final response.

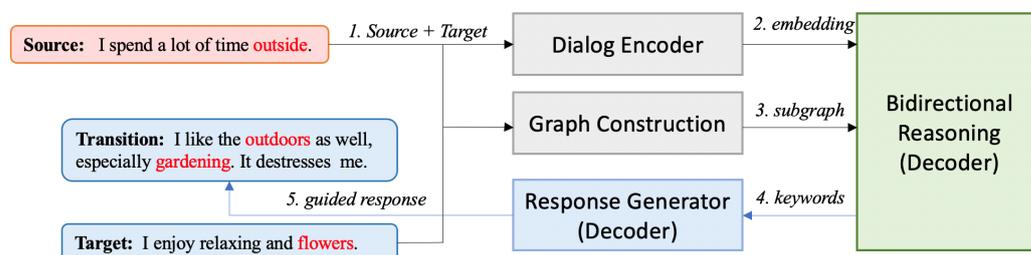


Figure 1. The process flowchart.

### 3.3. Data Cleaning

This section aims to improve the dataset quality and integrity. The specific process is as follows.

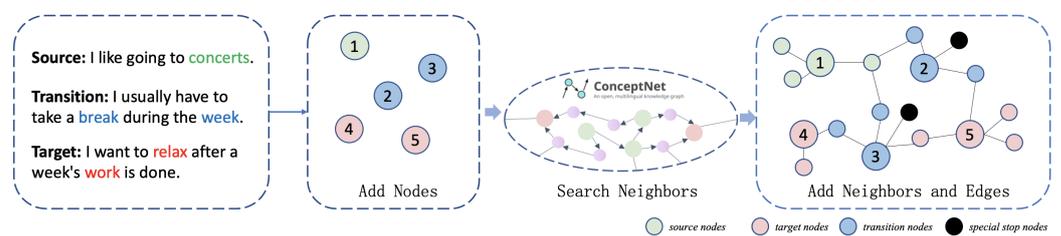
1. Token Limit Filter: Remove long text segments over 512 tokens and keep text concise and manageable.

2. Profanity Detection Filter: Use a filter to find offensive language in dialogs. Check for offensive words based on a list and patterns. The filter refers to Profanity-Check (<https://github.com/vzhou842/profanity-check>, accessed on 1 December 2023), which is a fast, robust Python library to check for profanities or offensive language in strings.
3. Check the filtered dataset manually. Ensure that there are no mistakes due to the automated filters.

Finally, this filtering process results in a refined dataset suitable for subsequent analysis and model training.

### 3.4. Subgraph Construction

We describe the construction of a dialog subgraph using  $c$  and  $t$ , represented as  $G = (V, E)$ . Here,  $V$  denotes the set of topic nodes, while  $E$  represents the edges connecting these topics. The specific details are provided in Figure 2, and the numbers in the nodes of Figure 2 indicate that the keywords are added to the dialog subgraph in order.



**Figure 2.** The process of dialog subgraph building.

**Node Selection** To determine the nodes in  $G$ , we employ a rule-based keyword extractor that combines TF-IDF [39] and Part-of-Speech [40] features to extract keywords from  $c$  and  $t$ . The keywords in  $c$  serve as the source topic nodes denoted as  $V_c = \{w_1, w_2, \dots, w_p\}$ , while the keywords in  $t$  serve as the target topic nodes denoted as  $V_t = \{w_1, w_2, \dots, w_q\}$ , where  $p$  and  $q$  represent the number of keywords in the source  $c$  and target  $t$ , respectively. Therefore,  $V = V_c \cup V_t$ . Afterward, we retrieve the neighboring nodes of keywords from ConceptNet, choosing  $N$  nodes to add to  $V$  and establish edges among them. The appropriate value of  $N$  will be determined in the forthcoming ablation experiment section. Furthermore, we add a particular neighbor node  $z^{stop}$  for each keyword within the subgraph. The  $z^{stop}$  serves as a condition to terminate decoding generation.

**Embedding Initialization** After identifying the nodes, we utilize ConceptNet [41] to obtain a node representation. Each topic node  $w_i$  is initially aligned with the corresponding node in ConceptNet and represented as  $h_i = ConceptNet(w_i)$ , where  $h_i$  denotes the initial representation of the node  $w_i$ . ConceptNet refers to the Numberbatch (<https://github.com/commonsense/conceptnet-numberbatch>, accessed on 1 December 2023) embeddings, and  $d$  represents the dimension of each node representation. Numberbatch is an embedding space for word vectors that leverages a combination of semantic information from diverse knowledge sources to enhance the representation of words. Developed by the team at ConceptNet, Numberbatch exhibits improved performance in various natural language processing tasks by capturing nuanced semantic relationships.

$$\begin{aligned}
 \mathbf{h}_{N_i^k} &= \frac{1}{|N_i^k|} \sum_{w_j \in N_i^k} ConceptNet(w_j), \\
 \bar{\mathbf{h}}_i &= \mathbf{h}_i + \sum_{k=1}^K (\mathbf{W}_k \mathbf{h}_{N_i^k} + \mathbf{b})
 \end{aligned}
 \tag{2}$$

Additionally, to capture topic relations effectively,  $h_i$  is updated by incorporating the representations of its  $K$ -hop neighbors in ConceptNet, known as  $K$ -hop neighboring representations.  $K$  represents the maximum number of hops considered, which is set to

2.  $N_i^k$  denotes the  $k$ -th hop neighboring nodes of  $w_i$  in the ConceptNet graph.  $\mathbf{W}_k$  and  $\mathbf{b}$  correspond to the weight matrix and bias vector, respectively.

### 3.5. Dialog Encoder

The dialog encoder comprehends the dialog context and outputs an embedding of the dialog. To obtain an embedded representation of the dialog, we concatenate the source and target utterances into a single line and input it into the dialog encoder. Our approach utilizes a multi-layer Transformer to encode the dialog context. Previous work [42,43] has proven that a multi-layer structure is highly effective in capturing semantic information, which offers parameter efficiency and feature abstraction, outperforming alternatives like single-layer Transformers or sequential models, such as RNNs or LSTMs, and proving well suited for tasks demanding comprehensive dialog comprehension.

Formally, given a dialog context  $C = \{c, t\}$ , where  $c_i = \{w_{i1}, w_{i2}, \dots, w_{in}\}$  is a sequence of words, the Transformer encoder will convert  $c_i$  into a sequence of hidden embeddings:

$$[\hat{\mathbf{h}}_1^{cls}, \hat{\mathbf{h}}_3^{cls}] = \text{Transformer}_{\theta_w}([\mathbf{e}_i^{cls}, \mathbf{e}_{i1}^w, \dots, \mathbf{e}_{in}^w]) \tag{3}$$

In the above equation,  $[\hat{\mathbf{h}}_1^{cls}, \hat{\mathbf{h}}_3^{cls}]$  represents a sequence of hidden embeddings. In this case, it appears that the output embeddings correspond to specific positions.  $\text{Transformer}_{\theta_w}$  refers to the Transformer encoder with parameters  $\theta_w$ . This is the function that processes the input sequence.  $([\mathbf{e}_i^{cls}, \mathbf{e}_{i1}^w, \dots, \mathbf{e}_{in}^w])$  is the input sequence to the Transformer encoder, where each  $\mathbf{e}_{ij}^w$  is the embedding vector of the  $j$ -th word in the  $i$ -th sequence.

$$h^{cls} = \text{Transformer}_{\theta_u}([\hat{\mathbf{h}}_1^{cls}, \hat{\mathbf{h}}_3^{cls}]). \tag{4}$$

Here,  $h^{cls}$  is the utterance representation embedding that incorporates source and target awareness.  $\text{Transformer}_{\theta_u}$  refers to another Transformer layer with parameters  $\theta_u$ . This layer is used to process the input sequence.  $[\hat{\mathbf{h}}_1^{cls}, \hat{\mathbf{h}}_3^{cls}]$  is the input sequence to the second Transformer layer, which is the output from the previous Transformer layer. The output embeddings can be further used for tasks such as guiding the generation of a keyword set in a bidirectional reasoning decoder.

### 3.6. Bidirectional Reasoning Decoder

The bidirectional reasoning module generates a keyword sequence as cue words to give an “intermediate” utterance. Bidirectional graph scoring is utilized to fuse graph node representations based on the dialog context representations, as shown in Figure 3.

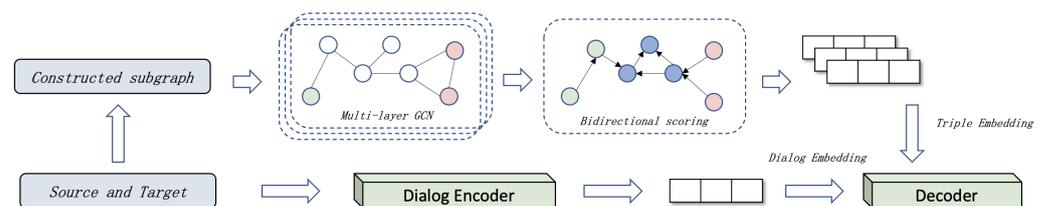


Figure 3. The bidirectional reasoning process.

**Graph Encoding** To encode the topic entities in the dialog graphs and obtain representations for concepts and relations, we employ multi-layer GCN encoders [44]. Inspired by the TransE [45] model, we also update the concept embedding by subtracting the corresponding relation embedding from each neighboring concept embedding to obtain the relation representation. At the  $(l + 1)$ th layer, we update the embedding of each entity  $v_i$

by aggregating its neighbors  $\mathcal{N}_i$ , which consist of pairs of concepts and relations that are connected to  $v_i$ :

$$h_i^{(l+1)} = \sigma \left( W_s^{(l)} h_i^{(l)} + \sum_{(j,r) \in \mathcal{N}_i} \frac{1}{|\mathcal{N}_i|} W_n^{(l)} (h_j^{(l)} - h_r^{(l)}) \right), \tag{5}$$

In the above equation,  $h_i^{(l)}$ ,  $h_j^{(l)}$  and  $h_r^{(l)}$  represent the embeddings of node  $v_i$  node  $v_j$ , and the relations between  $v_i$  and  $v_j$  at layer  $(l)$ th,  $W_s^{(l)}$  and  $W_n^{(l)}$  are specific trainable parameter matrices for layer  $l$ th;  $\sigma$  is a non-linear active function. The relation embedding is also updated at the  $(l + 1)$ th layer via  $h_r^{(l+1)} = W_R^{(l)} h_r^{(l)}$ . After  $L$  iterations, we are able to obtain  $\{h_{v_1}^{(L)}, \dots, h_{v_{|V|}}^{(L)}\}$ , a set of concept representations, and  $\{h_{r_1}^{(L)}, \dots, h_{r_{|R|}}^{(L)}\}$ , a set of relation representations.

**Bidirectional Reasoning** We employ bidirectional reasoning to compute concept distributions on graphs. This approach incorporates neighboring information and the current decoder state to adjust the weight of the bidirectional concept (source and target entities) in the graph at each decoding step. Initially, a score of 1 is assigned to source and target concepts, while others are given a score of 0. Subsequently, the information regarding the scored concepts is disseminated throughout the graph to update unvisited concepts in both directions. For an unvisited concept  $v_i \in V$ ,  $score(v)$  is computed by aggregating the evidence from its visited neighboring concepts  $\mathcal{N}_i^{in}$ :

$$R(v_j, r, v_i) = \sigma([h_j^{(L)}; h_r^{(L)}; h_i^{(L)}] W_{relation} s_t),$$

$$score(v_i) = \sum_{(v_j,r) \in \mathcal{N}_i^{in}} \frac{1}{|\mathcal{N}_i^{in}|} (\gamma \cdot score(v_j) + R(v_j, r, v_i)), \tag{6}$$

Here, in the triple relevance  $R(v_j, r, v_i)$ ,  $v_j, v_i$  denote nodes (or concepts) in the graph;  $[h_j^{(L)}; h_r^{(L)}; h_i^{(L)}]$  denote the representations of nodes  $v_j, v_i$  at the  $L$ -th layer of a neural network.  $W_{relation}$  is the weight matrix relating the triple  $v_j, r, v_i, s_t$  is the current decoder state at decoding step, and  $\sigma$  is the Sigmoid activation function.

In concept score update function  $score(v_i)$ ,  $v$  is a node or concept in the graph,  $\mathcal{N}_i^{in}$  is a set of visited neighboring concepts for  $v$ , and  $\gamma$  is a discount factor. This equation updates the concept score by aggregating information from its visited neighboring concepts. After  $L$ -hop interactions, the distribution over the concepts is as follows:

$$P(V | s_t, \mathcal{G}) = softmax_{v \in V}(score(v)). \tag{7}$$

Here,  $P(V | s_t, \mathcal{G})$  represents the distribution over concepts in the graph  $\mathcal{G}$  given the current decoder state and  $s_t$  at decoding step  $t$ . In other words, it is the probability of each concept in the graph being the next element in the sequence being generated.  $softmax_{v \in V}(score(v))$  is the softmax function applied to the scores assigned to each concept in the graph. The score for each concept  $v$  is obtained from the bidirectional reasoning mechanism described earlier.

The final generation distribution conjoins the distribution over the concepts in graphs and the distribution over the standard vocabulary with a soft gate as follows:

$$P(z_t | z_{<t}, \mathcal{G}, h^{cls}) = g_t \cdot P(V | s_t, \mathcal{G}) + (1 - g_t) \cdot P(\mathcal{V} | s_t, h^{cls}). \tag{8}$$

$P(z_t | z_{<t}, \mathcal{G}, h^{cls})$  represents the probability distribution of the token  $z_t$  at decoding step  $t$  given the previously generated tokens  $z_{<t}$ , the graph  $\mathcal{G}$ , and the classification state  $h^{cls}$ .  $g_t$  is a soft gate that determines whether to refer to the graph  $\mathcal{G}$  or not.  $P(V | s_t, \mathcal{G})$  is the distribution over concepts in the graph  $\mathcal{G}$  given the current decoder state, which represents

the likelihood of each concept being the next token.  $P(\mathcal{V} | s_t, h^{cls})$  is the distribution over the standard vocabulary  $V$  given the current decoder state  $s_t$  and the state  $h^{cls}$ , which represents the likelihood of each standard vocabulary token being the next token.

Furthermore, the decoding stop condition of the decoder is as follows. (1) When encountering a specially marked node  $z^{stop}$  added to the neighbor nodes of  $\mathcal{G}$ , the decoder will regard  $z^{stop}$  as a legal candidate node and automatically stop decoding when  $z^{stop}$  is selected. (2) The cue words set  $z$  exceeds  $N_{max}$ . Moreover, we set all probabilities extracted in step  $k(k < t)$  to 0 to avoid repeated extractions.

### 3.7. Response Decoder

As shown in Figure 1, the sequence of the keywords  $z$  generated by bidirectional reasoning is sent to the response generator to generate a response containing relevant keywords. Inspired by prompts using generative models [46], an explicit sequence of entity keywords  $z$  can be regarded as prior knowledge, or cue words, in the generation process [47].

$$\hat{y}_0 = z, \hat{\mathbf{E}}_t = [\mathbf{e}_k^w | w_k \in \hat{y}_t], \quad (9)$$

Here,  $\hat{y}_0$  represents the initial value of the variable  $\hat{y}$ . In this case, it is set equal to  $z$ . Meanwhile,  $z$  represents a sequence of keywords generated by bidirectional reasoning.  $\hat{\mathbf{E}}_t$  represents a set of word embedding vectors at decoding step  $t$ . It is defined as a list that includes the embedding vector  $\mathbf{e}_k^w$  for each word  $w_k$  in the sequence  $\hat{y}_t$ .

$$p(w_t, l_t | C, \hat{y}_{t-1}) = \text{Transformer}(\mathbf{h}^{cls}, \hat{\mathbf{E}}_{t-1}). \quad (10)$$

Subsequently,  $p(w_t, l_t | C, \hat{y}_{t-1})$  is a conditional probability distribution over the next word  $w_t$  and a length parameter  $l_t$ , given some context  $C$  and the previous sequence  $\hat{y}_{t-1}$ . Transformer refers to a Transformer model commonly used in natural language processing tasks.  $\mathbf{h}^{cls}$  is likely the output hidden state vector associated with the special token "[CLS]" in the Transformer model.  $\hat{\mathbf{E}}_{t-1}$  represents the set of word embedding vectors at the previous decoding step, i.e., at step  $t-1$ . At each decoding step  $t$ , the decoder combines  $\hat{y}_{t-1}$  and the probability distribution  $l_t \in [0, |\hat{y}_{t-1}|]$  to be generated, where  $\hat{\mathbf{E}}_t$  is a list of word embedding vectors for  $\hat{y}_t$ .

### 3.8. Training and Loss Functions

Given a list of entity keywords  $Z$  in the target response  $t$ , the bidirectional reasoning decoder is trained as a sequence generation model while minimizing the negative log-likelihood (NLL) loss as follows:

$$\mathcal{L}_Z = \frac{1}{|Z|} \sum_{t=1}^{|Z|} -\log p(z_t | z_{1:t-1}, c, t, \mathcal{G}) \quad (11)$$

We first sample a subsequence  $\hat{y}$  containing all target concepts in the target response  $t$  to train the response generator. Then, for each  $k+1$  position  $l = 0, 1, \dots, k$  in  $\hat{y}$ ,  $(w_{i_l}, w_{i_l+1}, \dots, w_{j_l})$  are word ranges in the target response not yet generated at position  $l$ . The loss function is finally defined as follows:

$$\mathcal{L}_R = \frac{1}{k+1} \sum_{l=0}^k \sum_{i=i_l}^{j_l} -\log p(w_i, l | c, t, \hat{y}) \quad (12)$$

## 4. Experiment

This section presents a comprehensive overview of the dataset, including a detailed analysis of its characteristics. We also describe the comparative methods employed and thoroughly present the obtained results. We also explore four prominent mainstream approaches. We specifically analyze the unique contributions of each component to develop

our proposed solution. In particular, we focus on the significant impact of bidirectional reasoning and the size of the dialog subgraph on the design of our model.

#### 4.1. Dataset

We evaluate our solution and other baseline methods on the latest public single-turn target-oriented dialog dataset, OTTers [48]. OTTers was created for target-oriented dialog, which requires the agent to proactively generate a “bridging” utterance from the source to the target topic. The dataset utilized in this study encompasses distinct train-dev-test splits, namely in-domain and out-of-domain splits. The initial split, identified as out-of-domain (ood), ensures the absence of shared topics between the test set and any topic pairs within the train set. Conversely, the subsequent split, known as the in-domain (id) split, deviates from this constraint, allowing for a partial overlap between a topic in each test set pair and the corresponding train set, albeit with varying second topics. The “ood” split closely aligns with a zero-shot scenario, wherein the model must generate outputs encompassing topic transitions that have yet to be finetuned. Consequently, our expectations regarding the achieved results for the “ood” split lie below those corresponding to the “id” split. The specific statistical data are shown in Table 2.

**Table 2.** The dataset split statistics.

Split	Train	Dev	Test
in-domain (id)	1929	1160	1158
out-of-domain (ood)	2033	1151	1129

#### 4.2. Data Statistics

A statistical analysis was conducted to understand the structure and content of the dialogs comprehensively. This analysis aimed to extract valuable insights and offer a comprehensive overview. Key metrics such as the number of words, entities, and paths were calculated to illuminate the dialogs’ characteristics. In this study, an “utterance” referred to an independent linguistic unit within a dialog. The statistical findings in Table 3 outline the average number of tokens and entities per utterance or dialog. These metrics provide quantitative information on the dialogs’ length and complexity.

**Table 3.** Sentence- and dialog-level statistical results.

	In-Domain	Out-of-Domain
Words per utterance	3–68	5–71
Average words per utterance	16.4	15.8
Entities per utterance	1–5	1–4
Average entities per utterance	3.3	3.8
Words per dialog	24–201	31–183
Average words per dialog	78.4	69.7
Entities per dialog	3–10	3–12
Average entities per dialog	5.4	6.1
Knowledge graph (KG) path per dialog	1–3	1–4
Average KG path per dialog	2.3	2.9

OTTers is a comprehensive collection of genuine dialogs encompassing various everyday situations. We have methodically categorized all conversation topics into eight distinct categories to enhance comprehension and coherence. Table 4 illustrates the distribution of topics across the out-of-domain and in-domain sections of the dataset. Daily life, interpersonal relationships, and workplace discussions consistently exhibit the highest occurrence within the dataset. This alignment in frequency distribution closely aligns with the communication demands encountered in real-world contexts.

**Table 4.** Number of topic statistics.

Topics	Subtopics (The Most Common Are Listed)	Quantity
entertainment	cinema, museum, painting, concert	631
education	history, math, school, university	589
family	parent, friend, marriage, children	1143
workplace	job, stress, interview, employee	1053
health	sport, drinks, meal, vegetables	893
daily life	hobby, money, relax, food	1454
travel	transport, weather, outside, car	692

In particular, the daily life category involves informal interactions among individuals in families or friendships. The family category encompasses dialogs during social engagements, while the workplace category pertains to professional communication. The dataset demonstrates a diverse range of utterance counts, bolstering the system's ability to generalize across various conversational contexts. By incorporating dialogs with varying utterance counts, the model gains adaptability to handle a broad scope of conversational complexities effectively. It enhances the model's proficiency for real-world applications.

#### 4.3. Baselines

Given the existing mainstream approaches, predominantly end-to-end and graph-based methods, we have opted to employ four representative baselines for this study specifically. In addition, this paper only considers single-turn dialog; the multi-turn dialog model introduced in related work was not selected.

- DialoGPT [47]: DialoGPT is a SOTA large-scale pretrained dialog response generation model for multi-turn conversations. We select the medium version accessible via HuggingFace DialoGPT-medium (<https://huggingface.co/microsoft/DialoGPT-medium>, accessed on 1 December 2023).
- MultiGen [17]: This model extends GPT-2 with multi-hop reasoning on commonsense knowledge graphs.
- DKRN [16]: This model leverages a dynamic knowledge routing strategy for concept prediction. The concept is predicted based on closeness to the target.
- CODA [12]: This model breaks down the response generation process into two steps: generating explicit commonsense paths connecting the source and target concepts and then conditioning the response generation on these generated paths. It aims to replicate how humans bridge concepts and create conversation transitions by leveraging commonsense knowledge.

#### 4.4. Implementation Details

The experimentation conducted in this study utilized the publicly available target-oriented dialog dataset OTTers. The word embedding dimension and hidden layer dimension were both configured to 300. The batch size was set to 32, and the AdamW optimizer was employed. The initial learning rate for the model was established at  $1 \times 10^{-4}$ , complemented by dynamic learning rate decay. A dropout rate of 0.1 was implemented, along with 12 attention heads within the model architecture. Notably, the comparative model's experimental setup and parameter configurations closely mirrored those employed in TodBR.

#### 4.5. Metrics

##### 4.5.1. Automatic Evaluation

To assess the models' proficiency in generating effective bridging responses, we commence by conducting an automatic evaluation employing widely used text generation metrics, as follows.

(1) BLEU [49], commonly known as bilingual evaluation understudy, has emerged as a widely employed metric to assess the quality of machine-translated text. It proves

applicable in evaluating translations in any language, given the presence of discernible word boundaries within the text. Typically yielding a score ranging from 0 to 100, BLEU quantifies the degree of similarity between a reference text and a hypothesis text. A higher BLEU score indicates superior-quality translations.

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{1-\frac{r}{c}} & \text{if } c \leq r \end{cases} \quad (13)$$

$$BLEU = BP \cdot \exp\left(\sum_{n=1}^N w_n \log P_n\right)$$

As shown in the above formula,  $BP$  is a factor that penalizes excessive abbreviation in translations. It ensures that concise translations are appropriately penalized.  $c$  represents the length of the candidate translation (the machine-generated translation).  $r$  is the effective reference length, which is the length of the reference translation closest to the candidate translation in terms of the number of words.  $\exp$  is the exponential function, which is used to weigh the precision of different N-grams.  $\sum_{n=1}^N$  signifies that the BLEU score considers the precision of 1-g, 2-g, up to N-gram.  $w_n$  corresponds to the weight coefficients assigned to different N-grams.  $P_n$  is the precision of the N-gram.  $N$  denotes the characteristic length of the N-gram under examination. The BLEU score considers precision up to N-gram. Furthermore, we use SacreBLEU [50], which provides the hassle-free computation of shareable, comparable, and reproducible BLEU scores. It reports the total BLEU score that accounts for the overlap across 1–4 N-grams instead of only 4-g.

(2) ROUGE-L [51] relies on the longest common subsequence (LCS) approach, wherein a comparison is made between the model output and the reference. Specifically, it identifies the most extended sequence of words, which may not be consecutive but maintains the same order standard to both the model output and reference. A lengthier shared sequence signifies higher similarity between the two sequences.

$$R_{lcs} = \frac{LCS(X, Y)}{m}$$

$$P_{lcs} = \frac{LCS(X, Y)}{n} \quad (14)$$

$$F_{lcs} = \frac{(1 + \beta^2)R_{lcs}P_{lcs}}{R_{lcs} + \beta^2P_{lcs}}$$

$LCS(X, Y)$  is the length of the longest common subsequence of  $X$  and  $Y$ ;  $m$  and  $n$  represent the length of the reference text and generated text (usually the number of words contained);  $R_{lcs}$  and  $P_{lcs}$  represent recall and accuracy, respectively.  $F_{lcs}$  is the final calculated ROUGE-L.

(3) The evaluation of textual diversity incorporates Distinct-1 and Distinct-2 [52]. Distinct-1 measures the occurrence of distinct words or phrases within the generated text, while Distinct-2 focuses on the frequency of unique adjacent word or phrase pairs, commonly known as bigrams. Distinctness is quantified on a numerical scale from 0 to 1. A higher value approaching 1 signifies an augmented abundance of varied and unique words or bigrams in the generated text. Consequently, these elevated values indicate a more comprehensive and distinctive range of responses produced by the model. To establish a formal representation of the distinct metric, the following mathematical definition is provided:

$$\text{Distinct-1} = \frac{\text{count}(\text{distinct}_{w_i \in R}(w_i))}{\text{count}(\text{all}_{w_i \in R}(w_i))}$$

$$\text{Distinct-2} = \frac{\text{count}(\text{distinct}_{w_i w_{i+1} \in R}(w_i w_{i+1}))}{\text{count}(\text{all}_{w_i w_{i+1} \in R}(w_i w_{i+1}))} \quad (15)$$

The symbol  $R$  encompasses the entire set of outcomes obtained from the test dataset. The function `distinct()` denotes the elimination of duplicate elements within  $R$ , while `all()` implies the inclusive enumeration of all outcomes. Lastly, the variable count delineates the numerical representation of statistical instances.

#### 4.5.2. Human Evaluation

Given the considerable criticism surrounding the correlation between automatic metrics and human judgment [53], we supplement our evaluation with two dialog-level metrics assessed through human evaluation: “coherence” and “logic”. The coherence metric involves a manual examination of the fluency and target-oriented nature of the entire dialog. On the other hand, the logic metric measures the extent to which the target of the dialog is logically reached. For each model, one hundred dialogs are generated through simulation.

We recruited five evaluators to assess our work. They were undergraduate students studying science and engineering, aged 20 to 25, with an English vocabulary ranging from 4000 to 5000 words. We applied the mentioned conditions to minimize variations among scorers, aiming to lessen their influence on the final results. These annotators rated the generated dialogs on a scale of {0, 1, 2, 3}, with higher scores indicating superior performance. We also computed the kappa to check for consistency in the evaluations. The kappa calculation result is  $-1-1$ , but, usually, the kappa falls between 0 and 1. It can be divided into five groups to represent different levels of consistency: 0.0–0.20, very low consistency (slight); 0.21–0.40, average consistency (fair); 0.41–0.60, moderate consistency (moderate); 0.61–0.80 high consistency (substantial); and 0.81–1 almost perfect.

### 4.6. Results

#### 4.6.1. Automatic Evaluation

The results reported in Table 5 exemplify the outcomes of the automated evaluation conducted on both the out-of-domain (ood) and in-domain (id) test datasets. It becomes evident from the results that our approach surpasses other models across all evaluation metrics, implying that the generated responses produced by our method exhibit excellent proximity to the ground truth responses within the dataset. Notably, within the realm of existing prevalent methodologies, despite its standing as a classic end-to-end generative model, DialoGPT performs poorly in terms of the BLEU, distinct, and ROUGE metrics. Conversely, multiple approaches rooted in graph reasoning exhibit superior performance. These observations support the claim that knowledge-based reasoning methods are more suitable for this particular task.

**Table 5.** Automatic evaluation.

	In-Domain				Out-of-Domain			
	BLEU	Distinct-1	Distinct-2	ROUGE-L	BLEU	Distinct-1	Distinct-2	ROUGE-L
DialoGPT	3.61	0.15	0.54	23.41	3.51	0.21	0.43	21.62
Multi-Gen	6.12	0.21	0.64	28.22	<b>4.90</b>	0.28	0.59	25.36
DKRN	3.44	0.14	0.56	23.34	3.84	0.21	0.53	22.27
CODA	4.95	0.17	0.60	24.13	4.39	0.20	0.61	23.03
<b>TodBR</b>	<b>6.26</b>	<b>0.23</b>	<b>0.67</b>	<b>28.56</b>	4.88	<b>0.32</b>	<b>0.65</b>	<b>25.98</b>

In addition, our proposed methodology exhibited a modest improvement in the BLEU and ROUGE-L metrics compared to the baseline models. This can be attributed to the inherent nature of openness and diversity that characterizes the dialog scenarios in this particular task, resulting in substantial differences in language and grammar between the reference answers and the generated responses. Consequently, the effectiveness of the BLEU and ROUGE-L metrics is limited by such disparities. However, significant advancements were seen in the distinct metric, showcasing our model’s ability to avoid redundant content and thoroughly comprehend logical information in the context. It generates responses that

are both fluent and diverse. These findings were further supported by the subsequent human evaluation.

#### 4.6.2. Human Evaluation

Although the automatic evaluation compared the generated responses with the ground truth in the dataset, there is a major challenge in that there can be many suitable responses in a dialog context. The ground truth can only be used as a reference. Therefore, human evaluation must be introduced. The specific results are shown in Table 6. It can be seen that our model performs better than other models. It indicates that, compared to the baseline model, our model generated responses that exhibited heightened consistency within the semantic and logical context. This effect was particularly significant when considering the logic metric, which emphasized the enhanced capability of our model to grasp the intricate logical differences that underlie conversation and generate responses that align contextually. Consequently, our model surpassed the baseline model's ability to generate contextually relevant and cohesive responses. Moreover, we calculated the kappa scores of the manual evaluators' scores on various indicators. For details, see the last row of the Table 6. Through consistency verification, we found that the human evaluators' scores had moderate consistency, and, at the same time, the logical evaluation was more consistent.

**Table 6.** Human evaluation.

	In-Domain		Out-of-Domain	
	Coherence	Logic	Coherence	Logic
DialoGPT	1.34	1.55	1.34	1.85
Multi-Gen	1.88	1.98	1.53	1.99
DKRN	2.03	2.22	1.58	2.21
CODA	2.01	2.45	1.61	2.34
<b>TodBR</b>	<b>2.13</b>	<b>2.47</b>	<b>1.74</b>	<b>2.38</b>
<i>kappa</i>	<i>0.48</i>	<i>0.50</i>	<i>0.45</i>	<i>0.51</i>

#### 4.7. Ablation Studies

We conducted ablation studies to analyze the relative contributions of the main components in TodBR. These studies involved removing the bidirectional reasoning module and investigating the optimal number of prompt words and graph nodes. Our findings indicate that the model's performance exhibited varying degrees of decrease when these components were ablated. Specifically, the absence of bidirectional reasoning, lack of prompt words, and unsuitable dialog subgraph size resulted in observable declines in the model's performance.

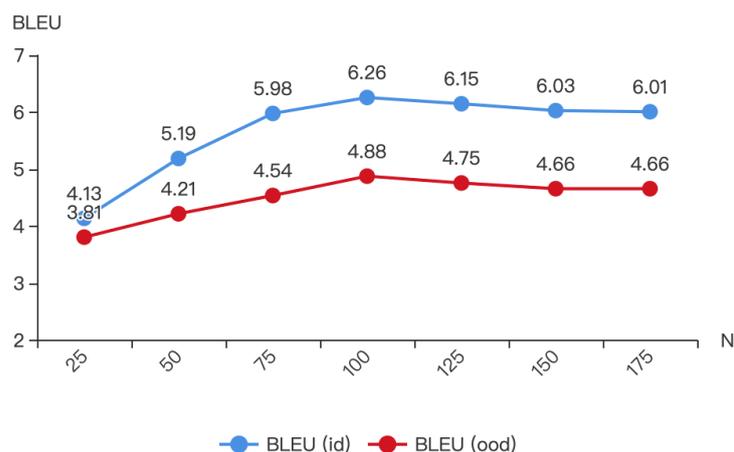
**Does the bidirectional reasoning work?** The importance and indispensability of both directions, namely the source direction and target direction reasoning, are exemplified in the first two rows of Table 7, where "w/o source" and "w/o target" denote the removal of source direction and target direction reasoning, respectively, such that only reasoning from one direction is considered. The results from these experiments demonstrate that, regardless of the direction, incorporating reasoning in the model contributes positively to its overall performance. Additionally, the performance degradation resulting from removing reasoning from the source and target is relatively similar, further highlighting the equal significance of both directions in improving the model's final performance.

**Table 7.** The result of ablation experiment in in-domain dataset.

Model	BLEU	Distinct-1/2	ROUGE-L
<b>TodBR</b>	<b>6.26</b>	<b>0.23/0.67</b>	<b>28.56</b>
w/o source reasoning	5.5	0.17/0.60	21.01
w/o target reasoning	5.4	0.12/0.61	21.87
w. fewer prompt words ( $N_{max} = 1$ )	4.3	0.19/0.65	23.63
w. more prompt words ( $N_{max} = 5$ )	5.8	0.11/0.64	23.19

**How many prompt words are suitable?** The terms “w. fewer prompt words” and “w. more prompt words” refer to the maximum number of prompt word sets generated during the bidirectional reasoning and decoding processes. The analysis of the experimental results reveals that the performance is significantly compromised regardless of whether there are fewer or more prompt words. This can be attributed to the fact that fewer words decrease the likelihood of obtaining the correct answer, while more words introduce noise during the decoding process. However, the impact on performance is more pronounced when fewer cue words are utilized. Notably, the degradation in performance is more evident when employing more prompt word sets due to the response generator’s filtering capabilities during decoding. Based on these findings, an average of three entity keywords per sentence is a more suitable size for achieving optimal performance.

**How much graph information do we need?** While constructing a dialog subgraph, our approach involved the utilization of pertinent keywords extracted from the dialog utterance to search within the commonsense knowledge graph. The resulting neighboring nodes were subsequently added to our evolving dialog subgraph. Notably, the precise number of neighboring nodes incorporated was not predetermined, prompting an investigation into the optimal quantity. To explore this matter, we conducted a series of systematic experiments. As illustrated in Figure 4, augmenting the quantity of neighbor node information only sometimes yields improved performance. Excess information can introduce noise and adversely impact the model’s performance. Nonetheless, it is noteworthy that the decline in performance as the number of nodes increases is not profoundly significant. Following thorough experimental validation, we determined that the most favorable trade-off between performance and efficiency is achieved by considering up to 100 neighbor nodes during graph reasoning.

**Figure 4.** The results of neighbors (denoted as N) and BLEU.

#### 4.8. Case Study

To further substantiate the effectiveness of our methodology, a comprehensive instance analysis was conducted, focusing on two specific cases extracted from the dataset employed in our experimental assessment. The outcomes of this analysis are meticulously documented and organized in Table 8.

**Table 8.** Case study.

Case 1—id	Context	I have prior agriculture experience.
	DialoGPT	a lot of cooking. I am a lot of cooking.
	Multi-Gen	My mom taught me how to cook.
	DKRN	My parents taught me how to grow.
	CODA	My mom loves to grow vegetables
	<b>TodBR</b>	<b>I have been a farmer for many years.</b>
Case 1—id	Ground truth	My parents were farmers.
	Target	I am a mom.
Case 2—ood	Context	Me and them hang out and specate sports.
	DialoGPT	This is great, and I like it too.
	Multi-Gen	So I decided to play football.
	DKRN	Cool! I like sport too.
	CODA	I like reading comics.
	<b>TodBR</b>	<b>I like comics and I also enjoy outside.</b>
Case 2—ood	Ground truth	I want to create cartoons for a major studio.
	Target	I like to read comics about sport.

Table 8 illustrates the results of a case analysis conducted on two split datasets, namely “ood” (out-of-domain) and “id” (in-domain). This analysis enables a comprehensive understanding of the strengths and weaknesses inherent in each model under scrutiny. Notably, when examining the performance of DialoGPT across both datasets, it becomes evident that the generated sentences are too simple. This observation highlights the challenge for end-to-end models and emphasizes the need for external information to generate coherent and contextually relevant topics in target-oriented dialogs. In contrast, models based on knowledge reasoning, such as Multi-Gen and DKRN, notably increase the amount of information conveyed in their generated responses, despite occasional errors. This enhanced information content significantly aids in the task of guiding topics. Additionally, our proposed model demonstrates robustness under challenging dialogs, such as dialogs with excessively long pathways or incomplete graph structures. This resilience further highlights the effectiveness of our bidirectional reasoning approach.

## 5. Conclusions, Limitations, and Future Research Directions

### 5.1. Conclusions

In this research, we introduce TodBR, an approach that leverages bidirectional reasoning on a knowledge graph (KG) to facilitate the effective steering of dialogs towards a specified topic. Initially, we construct a dialog subgraph enriched with commonsense knowledge to enable future reasoning. Subsequently, we train a bidirectional decoder using the dialog subgraph, which effectively guides our model to achieve the desired target word on ConceptNet. The efficacy of our proposed method is evaluated through both automatic metrics and human judgment, surpassing the performance of baseline models from both perspectives. Notably, bidirectional reasoning plays a crucial role in driving these improvements.

### 5.2. Limitations

The limitations of the current method are as follows.

1. Existing methods struggle to understand more complex targets: The current method’s understanding of the target only exists at the word level, and the semantics of the entire sentence must be considered, as well as more comprehensive goals, such as psychotherapy.
2. Existing methods are challenging to apply to limited resource languages: The current method requires rich external commonsense knowledge graphs for reasoning. However, except for English, the knowledge graphs of languages are too sparse and

lack structured knowledge, and it is difficult for our method to perform on limited language resources. We can resolve this problem by introducing an extra step to construct a knowledge graph for low-resource languages.

Future developments could explore methods to enhance the sentence-level comprehension within the current framework to address these limitations. Additionally, efforts to adapt the technique for languages with sparse knowledge graphs may involve strategies like leveraging domain-specific data or exploring alternative knowledge representation approaches.

### 5.3. Future Work

The future research directions may be as follows.

1. More diverse and complex conversation goals. Currently, goals are based on word semantics, but higher-level objectives like psychotherapy and educational guidance have yet to be explored. These areas have received little attention. Achieving such goals could significantly enhance the capabilities of target-oriented dialog systems.
2. Introducing large-scale language models and integrating supervised learning and reinforcement learning. With the popularity of large models, there is growing interest in reinforcement learning for goal-oriented dialog systems. These systems could be enhanced through reinforcement learning, and a simple yet effective approach is to use large models to provide reinforcement signals, improving the active dialog and guidance capabilities.
3. Introduce multi-modal models. Multi-modal methods can identify more appropriate conversation guidance opportunities by identifying users' facial expressions and tone, thereby significantly improving the goal-oriented conversation capabilities.

To summarize, we advise exploring these directions collaboratively, ensuring a balanced approach considering diverse conversational goals, leveraging large-scale language models, and incorporating multi-modal information for a more comprehensive understanding of user interactions. Embracing supervised learning and reinforcement learning is key to advancing target-oriented dialog systems.

**Author Contributions:** Conceptualization, Z.Q., Z.Y. and B.W.; Methodology, Z.Q. and Z.Y.; Software, Z.Y.; Writing—original draft preparation, Z.Q., Z.Y. and B.W.; Writing—review and editing, B.W. and Q.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by National Natural Science Foundation of China (Grant 62376188) and National Natural Science Foundation of China (Grant 62272340).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The used data are openly available in GitHub: <https://github.com/karinseve/OTTERS> (accessed on 1 December 2023).

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Adewumi, T.; Liwicki, F.; Liwicki, M. State-of-the-art in Open-domain Conversational AI: A Survey. *Information* **2022**, *13*, 298. [[CrossRef](#)]
2. Huang, M.; Zhu, X.; Gao, J. Challenges in Building Intelligent Open-domain Dialog Systems. *ACM Trans. Inf. Syst.* **2020**, *3*, 21. [[CrossRef](#)]
3. Shuster, K.; Xu, J.; Komeili, M.; Ju, D.; Smith, E.M.; Roller, S.; Ung, M.; Chen, M.; Arora, K.; Lane, J.; et al. BlenderBot 3: A deployed conversational agent that continually learns to responsibly engage. *arXiv* **2022**, arXiv:2208.03188.
4. OpenAI. GPT-4 Technical Report. *arXiv* **2023**, arXiv:2303.08774.
5. Balaraman, V.; Sheikhalishahi, S.; Magnini, B. Recent neural methods on dialogue state tracking for task-oriented dialogue systems: A survey. In Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue, Singapore, 29–31 July 2021; pp. 239–251.

6. Zhang, Z.; Takanobu, R.; Zhu, Q.; Huang, M.; Zhu, X. Recent advances and challenges in task-oriented dialog systems. *Sci. China Technol. Sci.* **2020**, *10*, 2011–2027. [[CrossRef](#)]
7. Tang, J.; Zhao, T.; Xiong, C.; Liang, X.; Xing, E.; Hu, Z. Target-Guided Open-Domain Conversation. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 5624–5634.
8. Ni, J.; Pandelea, V.; Young, T.; Zhou, H.; Cambria, E. Hitkg: Towards goal-oriented conversations via multi-hierarchy learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2022; pp. 11112–11120.
9. Tang, Z.H.; Yeh, M.-Y. EAGLE: Enhance Target-Oriented Dialogs by Global Planning and Topic Flow Integration. In Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM'23), Birmingham, UK, 21–25 October 2023; Association for Computing Machinery: New York, NY, USA, 2023; pp. 2402–2411.
10. Zhong, P.; Liu, Y.; Wang, H.; Miao, C. Keyword-guided neural conversational model. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 8 February 2021; pp. 14568–14576.
11. Yang, Z.; Wang, B.; Zhou, J.; Tan, Y.; Zhao, D.; Huang, K. Topkg: Target-oriented dialog via global planning on knowledge graph. In Proceedings of the 29th International Conference on Computational Linguistics, Gyeongju, Republic of Korea, 12–17 October 2022; pp. 745–755.
12. Gupta, P.; Jhamtani, H.; Bigham, J. Target-Guided Dialogue Response Generation Using Commonsense and Data Augmentation. In Proceedings of the Findings of the Association for Computational Linguistics: NAACL, Seattle, WA, USA, 10–15 July 2022; pp. 1301–1317.
13. Jannach, D.; Chen, L. Conversational recommendation: A grand AI challenge. *AI Mag.* **2022**, *43*, 151–163. [[CrossRef](#)]
14. Venturo-Conerly, K.E.; Reynolds, R.; Clark, M.; Fitzpatrick, O.M.; Weisz, J.R. Personalizing youth psychotherapy: A scoping review of decision-making in modular treatments. *Clin. Psychol. Sci. Pract.* **2023**, *30*, 45–62. [[CrossRef](#)]
15. Okonkwo, C.W.; Ade-Ibijola, A. Chatbots applications in education: A systematic review. *Comput. Educ. Artif. Intell.* **2021**, *2*, 100033. [[CrossRef](#)]
16. Qin, J.; Ye, Z.; Tang, J.; Liang, X. Dynamic knowledge routing network for target-guided open-domain conversation. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 8657–8664.
17. Ji, H.; Ke, P.; Huang, S.; Wei, F.; Zhu, X.; Huang, M. Language Generation with Multi-Hop Reasoning on Commonsense Knowledge Graph. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 16–20 November 2020; pp. 725–736.
18. Xu, J.; Wang, H.; Niu, Z.; Wu, H.; Che, W. Knowledge graph grounded goal planning for open-domain conversation generation. In Proceedings of the AAAI conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 9338–9345.
19. Xu, J.; Wang, H.; Niu, Z.Y.; Wu, H.; Che, W.; Liu, T. Conversational graph grounded policy learning for open-domain conversation generation. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 1835–1845.
20. Rohmatillah, M.; Chien, J.T. Corrective guidance and learning for dialogue management. In Proceedings of the 30th ACM International Conference on Information, Gold Coast, Australia, 1–5 November 2021; pp. 1548–1557.
21. Liu, Z.; Wang, H.; Niu, Z.Y.; Wu, H.; Che, W.; Liu, T. Towards Conversational Recommendation over Multi-Type Dialogs. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 1036–1049.
22. Deng, Y.; Lei, W.; Lam, W.; Chua, T.S. A survey on proactive dialogue systems: Problems, methods, and prospects. *arXiv* **2023**, arXiv:2305.02750.
23. Deng, Y.; Lei, W.; Huang, M.; Chua, T.S. Goal Awareness for Conversational AI: Proactivity, Non-collaborativity, and Beyond. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, Toronto, ON, Canada, 9–14 July 2023; pp. 1–10.
24. Wang, J.; Lin, D.; Li, W. Dialogue Planning via Brownian Bridge Stochastic Process for Goal-directed Proactive Dialogue. *arXiv* **2023**, arXiv:2305.05290.
25. Tan, Y.; Wang, B.; Liu, A.; Zhao, D.; Huang, K.; He, R.; Hou, Y. Guiding Dialogue Agents to Complex Semantic Targets by Dynamically Completing Knowledge Graph. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2023, Toronto, ON, Canada, 9–14 July 2023; pp. 6506–6518.
26. Liu, A.; Wang, B.; Tan, Y.; Zhao, D.; Huang, K.; He, R.; Hou, Y. MTGP: Multi-turn Target-oriented Dialogue Guided by Generative Global Path with Flexible Turns. In Proceedings of the Findings of the Association for Computational Linguistics: ACL 2023, Toronto, ON, Canada, 9–14 July 2023; pp. 259–271.
27. Wang, H.; Guo, B.; Wu, W.; Liu, S.; Yu, Z. Towards information-rich, logical dialogue systems with knowledge-enhanced neural models. *Neurocomputing* **2021**, *1*, 248–264. [[CrossRef](#)]
28. Puduppully, R.; Dong, L.; Lapata, M. Data-to-text generation with content selection and planning. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 6908–6915.
29. Hua, X.; Wang, L. Sentence-level content planning and style specification for neural text generation. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; pp. 591–602.
30. Moryossef, A.; Goldberg, Y.; Dagan, I. Step-by-step: Separating planning from realization in neural data-to-text generation. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; Volume 1, pp. 2267–2277.

31. Su, Y.; Vandyke, D.; Wang, S.; Fang, Y.; Collier, N. Plan-then-generate: Controlled data-to-text generation via planning. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2021, Punta Cana, Dominican Republic, 7–11 November 2021; pp. 895–909.
32. Lan, Y.; He, G.; Jiang, J.; Jiang, J.; Zhao, W.X.; Wen, J.R. Complex knowledge base question answering: A survey. *IEEE Trans. Knowl. Data Eng.* **2022**, *35*, 11196–11215. [[CrossRef](#)]
33. Lao, M.; Guo, Y.; Pu, N.; Chen, W.; Liu, Y.; Lew, M.S. Multi-stage hybrid embedding fusion network for visual question answering. *Neurocomputing* **2021**, *1*, 541–550. [[CrossRef](#)]
34. Ye, M.; You, Q.; Ma, F. Qualifier: Question-guided self-attentive multimodal fusion network for audio visual scene-aware dialog. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2022; pp. 248–256.
35. Siriwardhana, S.; Weerasekera, R.; Wen, E.; Kaluarachchi, T.; Rana, R.; Nanayakkara, S. Improving the domain adaptation of retrieval augmented generation (RAG) models for open domain question answering. *Trans. Assoc. Comput. Linguist.* **2023**, *11*, 1–17. [[CrossRef](#)]
36. Yu, W. Retrieval-augmented generation across heterogeneous knowledge. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 10–15 July 2022; pp. 52–58.
37. Wiseman, S.; Rush, A.M. Sequence-to-Sequence Learning as Beam-Search Optimization. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 1296–1306.
38. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–8.
39. Ramos, J. Using tf-idf to determine word relevance in document queries. In Proceedings of the First Instructional Conference on Machine Learning, Piscataway, NJ, USA, 3–8 December 2003; pp. 29–48.
40. Brill, E. A simple rule-based part of speech tagger. In Proceedings of the third conference on Applied natural language processing (ANLC '92), Harriman, NY, USA, 23–26 February 1992; pp. 152–155.
41. Speer, R.; Chin, J.; Havasi, C. Conceptnet 5.5: An open multilingual graph of general knowledge. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI'17), San Francisco, CA, USA, 4–9 February 2017; pp. 4444–4451.
42. Zhao, L.; Yu, B.; Shang, W. Investigation into Layer Number Effect on Breakdown Strength of Multi-Layer Polymer Films. *Polymers* **2022**, *14*, 1653. [[CrossRef](#)] [[PubMed](#)]
43. Zhang, L.; Sun, H. ESA-GCN: An Enhanced Graph-Based Node Classification Method for Class Imbalance Using ENN-SMOTE Sampling and an Attention Mechanism. *Appl. Sci.* **2024**, *14*, 111. [[CrossRef](#)]
44. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
45. Yang, H.; Liu, J. Knowledge graph representation learning as groupoid: Unifying TransE, RotatE, QuatE, ComplEx. In Proceedings of the 30th ACM International Conference on Information, Gold Coast, Australia, 1–5 November 2021; pp. 2311–2320.
46. Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; Neubig, G. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.* **2023**, *55*, 195. [[CrossRef](#)]
47. Zhang, Y.; Sun, S.; Galley, M.; Chen, Y.C.; Brockett, C.; Gao, X.; Dolan, W.B. DIALOGPT: Large-Scale Generative Pre-training for Conversational Response Generation. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Online, 5–10 July 2020; pp. 270–278.
48. Sevegnani, K.; Howcroft, D.M.; Konstas, I.; Rieser, V. OTTers: One-turn Topic Transitions for Open-Domain Dialogue. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Virtual Event, 1–6 August 2021; pp. 2492–2504.
49. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. Bleu: A method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA, USA, 6–12 July 2002; pp. 311–318.
50. Keenan, W.J. *Sacre Bleu: Faith, fashion and freedom: Marist foundation garments 1817–1862*. In *Materializing Religion*; Routledge: London, UK, 2017; pp. 132–153.
51. Lin, C.Y. Rouge: A package for automatic evaluation of summaries. In Proceedings of the Workshop on Text Summarization Branches Out, Barcelona, Spain, 21–26 July 2004; pp. 74–81.
52. Li, J.; Galley, M.; Brockett, C.; Gao, J.; Dolan, B. A diversity-promoting objective function for neural conversation models. *arXiv* **2015**, arXiv:1510.03055.
53. Sai, A.B.; Mohankumar, A.K.; Khapra, M.M. A survey of evaluation metrics used for NLG systems. *ACM Comput. Surv. (CSUR)* **2022**, *1*, 26. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.