A Target-Driven Planning Approach for Goal-Directed Dialog Systems

Jian Wang, Dongding Lin, and Wenjie Li

***Abstract*— Existing dialog systems mainly build social bonds reactively with users for chitchat or assist users with specific tasks. In this work, we push forward to a promising yet under-explored proactive dialog paradigm called goal-directed dialog systems, where the "goal" refers to achieving the recommendation for a predetermined target topic through social conversations. We focus on how to make plans that naturally lead users to achieve the goal through smooth topic transitions. To this end, we propose a target-driven planning network (TPNet) to drive the system to transit between different conversation stages. Built upon the widely used transformer architecture, TPNet frames the complicated planning process as a sequence generation task, which plans a dialog path consisting of dialog actions and topics. We then apply our TPNet with planned content to guide dialog generation using various backbone models. Extensive experiments show that our approach obtains the state-of-the-art performance in automatic and human evaluations. The results demonstrate that TPNet affects the improvement of goal-directed dialog systems significantly.**

***Index Terms*— Goal-directed dialog systems, dialog generation, target-driven planning.**

I. Introduction

DIALOGUE systems [1] are mainly developed for chatting with users for entertainment, i.e., open-domain dialogs [2], [3], or assisting users in accomplishing specific tasks, i.e., task-oriented dialogs [4], [5]. A particular type of task-oriented dialog system named recommendation dialog system [6], [7] has gained growing research interest in recent years. It reveals that recommendation-oriented tasks further activate the application potential of dialog systems [8].

Most existing recommendation dialog systems [6], [7], [9], [10] converse with users reactively. They mainly respond to users' utterances to better understand the expressed preferences or requirements and then provide recommendations accordingly. Such reactive dialog systems have their limitation in reality since people may not have clear preferences for unfamiliar new topics or items. With this in mind, we explore how to proactively recommend some target topics or items that possibly attract users through sociable conversations. Recently, the emergence of the DuRecDial [11] dataset has shed light on this research direction. As shown in the example of Fig. 1, suppose there is an explicit goal, i.e., to recommend a target movie named "*McDull, Prince de la Bun*," the system (i.e., Bot) is required to proactively and naturally lead the whole conversation (e.g., "*greeting*" → "*ask user*" → "*chat about the star*" → "*movie recommendation*") so as to recommend the target movie when appropriate. For the above process, the system needs to consider the user profile, the domain knowledge graph, and the designated target topic before generating each system utterance. Here, we take the term "system utterance" rather than "response" used in a lot of related work since the system needs to proactively lead the conversation in most cases. Specifically, the user profile helps the system take the initiative and warm up conversations since it reveals the user's attributes. The domain knowledge graph has domain-specific topics and associated attributes, which are crucial to enable smooth topic transitions (e.g., "*Running Out of Time*" → "*Andy Lau*" → "*McDull, Prince de la Bun*").

In this work, we move forward to goal-directed dialog systems, where the "goal" refers to achieving the recommendation for a predetermined target topic through social conversations. Given a target topic (e.g., a movie or a piece of music), we require a dialog system to lead the conversation proactively from the chitchat to achieving the goal. With this in mind, our key research question is "*How to make reasonable plans to drive the conversation to achieve the goal step by step?*." Compared to previous reactive recommendation dialog systems [6], [7], [9], [10], our problem is more challenging because: 1) the system should maintain an engaging conversation to attract the user's attention and naturally transit among relevant topics and 2) the system is required to arouse the user's interest in the target topic to be recommended rather than discovering user preferences alone.

To address the above challenges, we propose a target-driven planning network (TPNet) to guide a dialog system to generate appropriate utterances in a pipeline manner. First, since the system is grounded on complicated text input, we use different encoders to learn representations of different types of texts. We employ the widely used transformer [12] network to represent the domain knowledge graph, where we devise a target-aware graph attention transformer for knowledge encoding. We adopt an end-to-end memory network [13] and a pretrained language model (PLM) BERT [14] to encode

Manuscript received 1 July 2022; revised 1 November 2022 and 28 January 2023; accepted 30 January 2023. This work was supported in part by the Research Grants Council of Hong Kong under Grant 15207122, Grant 15207920, Grant 15207821, and Grant 15204018; in part by the National Natural Science Foundation of China under Grant 62076212; and in part by the PolyU internal under Grant ZVQ0 and Grant ZVVX. *(Corresponding author: Jian Wang.)*

The authors are with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong (e-mail: jian-dylan.wang@connect.polyu.hk; dongding88.lin@connect.polyu.hk; cswjli@comp.polyu.edu.hk).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2023.3242071.

Digital Object Identifier 10.1109/TNNLS.2023.3242071

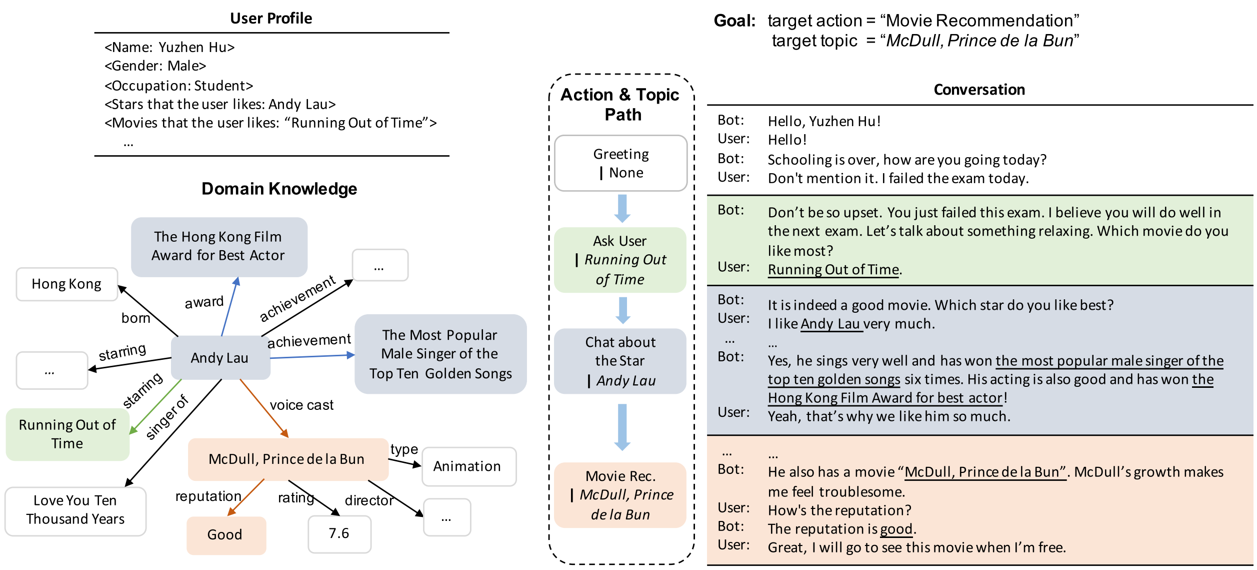

Fig. 1. Illustrative example from the repurposed DuRecDial [11] dataset. The whole conversation is grounded on the user profile and domain knowledge (displayed partially), and is directed by the target action and topic.

the user profile and the dialog history, respectively. We also propose a simple yet effective updating mechanism to update the user memory representation to cope with the user's input and feedback during the conversation. Second, we propose a target-driven conversation planner to make reasonable plans, which frames the planning process as a sequence generation task. It aims to plan (generate) a dialog path that consists of dialog topics and how the system delivers these topics (i.e., dialog actions). The planner is built on top of the transformer [12] decoder architecture, where we devise a novel knowledge-target mutual attention mechanism and a set-search decoding (SSD) strategy. In the end, we use the planned path to help extract necessary knowledge and, meanwhile, to explicitly guide the system to generate appropriate utterances.

Overall, our contributions are summarized in three folds.

1) We push forward from the reactive recommendation dialog paradigm toward the promising goal-directed dialog paradigm, where the "goal" refers to achieving the recommendation for a predetermined target topic through social conversations. The proposed paradigm is under-explored to the best of our knowledge.
2) We propose a TPNet to plan a dialog path consisting of dialog actions and topics, which helps the system lead the conversation to reach the target topic. It guides the system for utterance generation and achieving the goal step by step.
3) Experiments show that our method obtains the state-of-the-art results in both automatic and human evaluations. Our results and analysis indicate that target-driven planning is essential to improving goal-directed dialog systems.

II. Related Work

Our work is mainly related to target-guided dialog systems and recommendation-oriented dialog systems. We briefly review related work and clarify key differences compared with our work as follows.

A. Target-Guided Dialog

Target-guided dialog systems aim to proactively guide conversations by the given targets. Depending on the nature of targets, existing works have mainly focused on using keywords [15], [16], [17], and topics [18], [19] as the guided targets. For keyword-guided dialogs, [15] introduced coarse-grained keywords to control the intended content of the system response in open-domain conversations. As a follow-up study, [16] proposed a dynamic knowledge routing network (DKRN) to drive a conversation toward the target keyword with a discourse-level guiding strategy. Since commonsense knowledge is crucial to human conversations, [17] leveraged external commonsense knowledge graphs for keyword transition and response retrieval using graph neural networks (GNNs). For topic-guided dialogs, the DuConv [18] dataset is released for exploring an entity over a factual knowledge graph as the target topic to guide the conversation. It requires the system to plan over the knowledge graph to lead the conversation from an initial topic to the target topic. Another important research line is how to learn dialog strategies to achieve the target, including graph-grounded policies [20], [21], conversational lines [22], and topic transitions [23]. For example, [21] planned a high-level goal sequence with balance between dialog coherence and topic consistency by traversing over the knowledge graph. [24] presented a two-level policy using hierarchical reinforcement learning (HRL) to guide response generation.

Overall, the above studies mainly focus on open-domain dialog systems, where the target is regarded as achieved when either the human or the system mentions the target keyword or similar topical words in an utterance. In contrast, our work formulates a more challenging setting, i.e., achieving the recommendation for a predetermined target topic through social conversations. It requires the system to interact with the user with more actions to achieve the goal, such as chitchat, user exploration, topic elicitation, and recommendation.

B. Recommendation-Oriented Dialog

A recommendation-oriented dialog system encourages natural interactions with a user and make recommendations accordingly, which can be viewed as a special type of task-oriented dialog system. Recommendation dialog systems may have various forms of conversations, such as social chitchat, question answering, recommendation, and so on. It was the emergence of multiple datasets that helps push forward the research in this area, such as GoRecDial [7], TG-ReDial [19], INSPIRED [25], and DuRecDial [11]. Regarding dialog generation approaches, multi-goal driven conversation generation (MGCG) [11] and knowledge-enhanced multi-subgoal driven recommender system (KERS) [26] explored the transition policy from a non-recommendation dialog to a recommendation-oriented one. CR-Walker [9] performed tree-structured reasoning over knowledge graphs, obtaining hierarchical dialog acts to guide both item and response generations. More recently, [10] combined the advantage of slot filling and language generation [27]. Further explored knowledge-aware recommendation dialog systems. An attribute-guided framework [28] was proposed to keep track of item attributes and provide a more engaging chat experience. There is another similar research area called conversational recommender systems (CRSs) [29], [30]. Previous studies [8], [31] pointed out that CRS effectively handles the cold-start problem in recommender systems and it can provide personalized recommendations through natural language conversations. Compared with recommendation dialog systems, the main task of CRS lies in discovering user preferences [32], [33], asking clarifying questions about item attributes [34], [35], and searching for optimal candidate items [36], [37], [38].

Nonetheless, most existing models converse with users reactively, where they provide recommendations according to the user's expressed interests or requirements. However, people may not have clear preferences for unfamiliar new topics or items. There is still a lack of exploration to study how to enable a dialog system to proactively recommend target topics that possibly attract users. This work aims to explore such a task and refer to it as the goal-directed dialog.

III. Preliminaries

In this section, we mainly provide preliminaries to make this article more understandable. The notations frequently used in this article are listed in Table I. The problem formulation and important sub-tasks are introduced as follows.

Suppose we have a goal-directed dialog corpus $\mathcal{D} = \{(\mathcal{U}_i, \mathcal{K}_i, \mathcal{H}_i, \mathcal{P}_i)\}_{i=1}^{N}$, where N denotes the total number of conversations. $\mathcal{U}_i = \{u_{i,j}\}_{j=1}^{N_U}$ is a user profile with each entry $u_{i,j}$ in form of a ⟨key, value⟩ pair, $\mathcal{K}_i = \{k_{i,j}\}_{j=1}^{N_K}$ denotes a set of domain knowledge facts relevant to ith conversation with each element $k_{i,j}$ in form of a ⟨subject, relation, object⟩ triple, $\mathcal{H}_i = \{(X_{i,t}, Y_{i,t})\}_{t=1}^{T}$ denotes conversation content with a total number of T turns, $\mathcal{P}_i = \{(a_{i,l}, z_{i,l})\}_{l=1}^{L}$ denotes a sequence of annotated plans, and each plan specifies a dialog action $a_{i,l}$ and a dialog topic $z_{i,l}$. Here, the dialog topics are mainly constructed upon the domain knowledge $\mathcal{K}_i$. Each action-topic pair may affect multiple conversation turns. L denotes the number of distinct action-topic pairs.

TABLE I

List of Frequently Used Notations

Symbols	**Descriptions**
$\mathcal{U}$	User profile
$\mathcal{K}$	Domain knowledge
$\mathcal{H}$	Dialogue history
a_t	A dialogue action
z_t	A dialogue topic
$a_{T'}$	A target action
$z_{T'}$	A target topic
`[A]`	A special token to denote an action
`[T]`	A special token to denote a topic

Given a target consisting of a target action $a_{T'}$ and a target topic $z_{T'}$, a user profile $\mathcal{U}'$, a set of relevant domain knowledge $\mathcal{K}'$, and a dialog history $\mathcal{H}'$, our goal is to generate coherent utterances to engage the user in the conversation so as to recommend the designated target topic when appropriate. Due to the complexity, the problem can be decomposed into three sub-tasks: 1) *action planning*, i.e., plan actions to determine where the conversation should go to lead the conversation proactively; 2) *topic planning*, i.e., plan appropriate topics to move forward to the target topic; and 3) *dialog generation*, i.e., generate a proper system utterance to achieve the planned action and topic at each turn.

IV. Target-Driven Planning Network

To proactively lead the conversation to achieve the goal, we propose a TPNet to guide dialog generation in a pipeline manner, with the overview shown in Fig. 2. This section mainly describes the details of the TPNet, while the planning-enhanced dialog generation will be described in Section V. As shown in Fig. 2, our TPNet first employs different encoders to learn representations of different types of input texts, where the target-aware graph attention transformer and the user memory network are two key components. Then we propose a target-driven conversation planner to make reasonable plans, which frames the planning process as a sequence generation task. It aims to plan (generate) a dialog path consisting of dialog actions and topics.

A. Target-Aware Graph Attention Transformer

The domain knowledge is organized in a graph structure, while some nodes (i.e., *subject* or *object*) of this graph may contain long texts (e.g., topic-associated attributes). To efficiently represent the given domain knowledge $\mathcal{K}'$, we employ transformer [12] as the basic encoder, upon which additional encoding strategies are incorporated. Inspired by [39] and [40], we convert knowledge triples of the graph into unique relation-entity pairs instead of directly concatenating those triples to encode domain knowledge efficiently. As shown in Fig. 3, ⟨"voice cast," "Andy Lau"⟩ and ⟨"type," "Animation"⟩ are two different pairs of attributes of the same topic "*McDull, Prince de la Bun*." We use two new tokens `Entity` and `Relation` (`Rel.`) to differentiate between nodes and relations in the segment layer. We then use a special token `[T]`

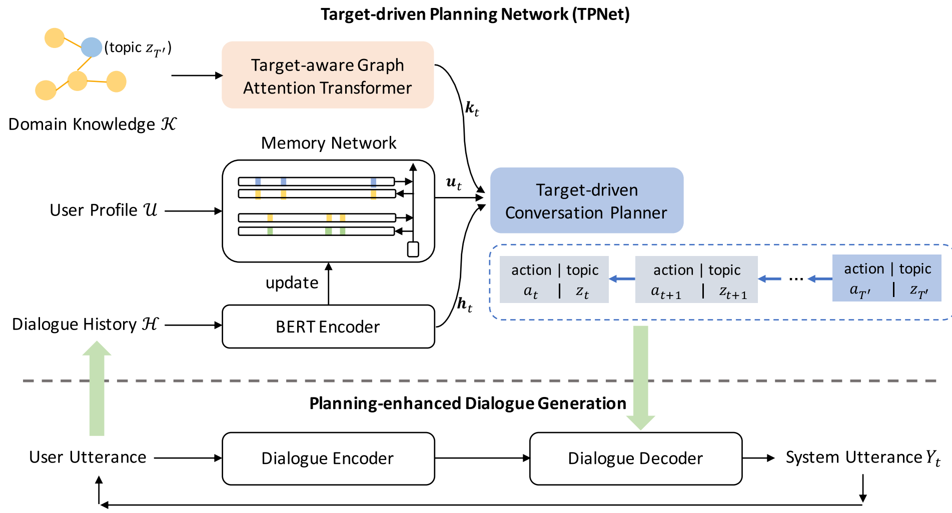

Fig. 2. Overview of our method, where the TPNet (top) and dialog generation (down) are trained in a pipeline manner.

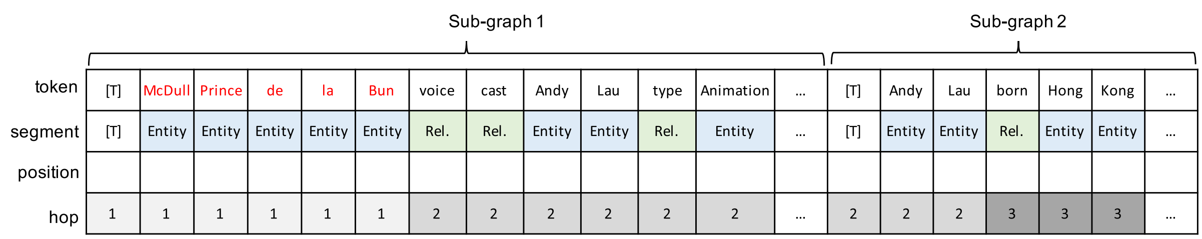

Fig. 3. Illustration of our encoding strategy for the target-aware graph attention transformer.

to denote a topic-centric (e.g., "*McDull, Prince de la Bun*") sub-graph in both the token layer and the segment layer. More importantly, the target topic itself is an entity node on the domain knowledge graph, which is of vital importance in our planning during the entire conversation. We attempt to encode such target-aware graph information to facilitate our planning. To this end, we add a new hop layer (see Fig. 3) into input embeddings to represent the number of hops from the target topic node to the current node or edge (relation) over the graph. We let the hops of the target topic tokens be 1, and the hops of the adjacent edge tokens and node tokens are computed by a simple breadth-first search (BFS). The embeddings of the hop layer are randomly initialized. The embeddings of the token, segment, and position layers can be initialized from PLMs, e.g., BERT [14]. After encoding, the final domain knowledge representation is denoted as $\mathbf{K} = (\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_l)$, where l is the length of the domain knowledge.

B. User Memory Network

As shown in Fig. 2, to model user attributes and preferences, we adopt an end-to-end memory network (MemNN) [13] to encode the user profile $\mathcal{U}'$, which is represented as a set of trainable embedding matrices $\mathbf{U}' = (\mathbf{U}^1, \mathbf{U}^2, \ldots, \mathbf{U}^K)$, where $\mathbf{U}^k = (\mathbf{u}_1^k, \mathbf{u}_2^k, \ldots, \mathbf{u}_m^k)$, $k \in [1, K]$ and K is the number of memory hops, m is the length of the user profile. This MemNN loops over K hops with adjacent weighted tying to obtain the memories of $\mathcal{U}'$, following previous studies [4], [41] that employ memory networks in dialog systems.

In course of the conversation, it is necessary to cope with the user's immediate feedback to track whether the user follows the system's plan. Especially, the system needs to make the conversation engaging first when the user deviates from the system's plan. The system must balance the goal and the user's feedback or expressed preference so that it can lead the conversation to reach the target topic. To this end, we use the dialog history to update user preferences by capturing relevant feedback from user utterances. First, we employ a BERT [14] encoder to encode the given dialog history $\mathcal{H}'$, obtaining its token-level hidden representation $\mathbf{H} = (\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_n)$, where n is the length of the dialog history. Then, we adopt the user memory embeddings $\mathbf{U}^k$ as the query to attend to the dialog history content $\mathbf{H}$, followed by an add operation to update the user memory representation. Similar to the multihop attention mechanism of the vanilla MemNN, here the user memory is also updated with K-hop iterations. The above computation is given by the following equation:

$$\mathbf{P}^k = \mathrm{softmax}(\mathbf{U}^k(\mathbf{H}^\mathsf{T})) \tag{1}$$

$$\mathbf{O}^k = \mathbf{P}^k\mathbf{H} \tag{2}$$

$$\mathbf{U}^{k+1} = \mathbf{U}^k + \mathbf{O}^k \tag{3}$$

where $k \in [1, K]$. We regard the memory representation at the last hop as the updated user memory representation, denoted as $\mathbf{U}$. Both $\mathbf{H}$ and $\mathbf{U}$ are then passed to the target-driven conversation planner for planning.

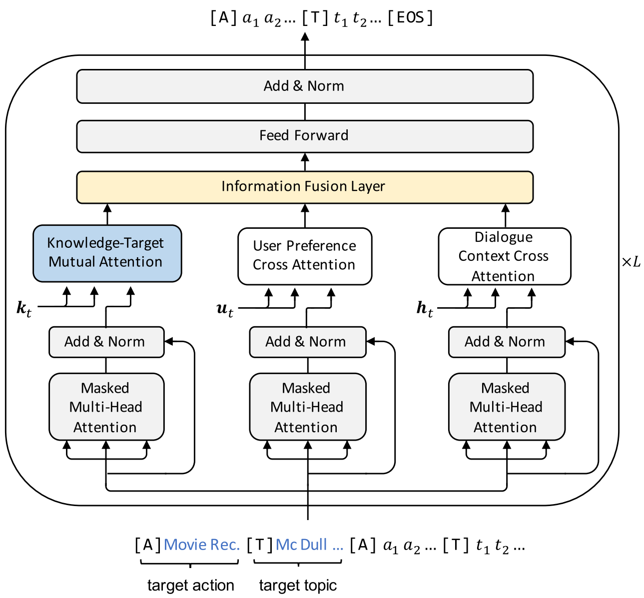

Fig. 4. Overview of the target-driven conversation planner.

C. Target-Driven Conversation Planner

Our target-driven conversation planner aims to plan a path consisting of dialog actions and topics in a generation-based manner. As a complete path to achieve the goal, both the target action $a_{T'}$ and the target topic $z_{T'}$ should be bounded at the end of the path to be planned. We expect that $a_{T'}$ and $z_{T'}$ can drive the conversation planner to generate a more reasonable path. Intuitively, we let the conversation planner generate the path from the target turn of the conversation to the present turn (see Fig. 2), which is of benefit to leverage more target-side information. With such intuition, we build our target-driven conversation planner based on the transformer [12] decoder architecture, which is shown in Fig. 4. It takes the tokens of the target action and the target topic as input and then generates a path sequence token by token, i.e., "[A] $a_1 a_2, \ldots,$ [T]$t_1 t_2, \ldots,$ [EOS]." Here, [A] is a special token to separate an action, [T] is a special token to separate a topic shared with our target-aware graph attention transformer encoder, and [EOS] denotes the end of the path sequence.

To train the conversation planner, we put the tokens of the target action and the target topic ahead of the path sequence as input (see Fig. 4) to let the planner condition on the target-side information during all generation steps. The path sequence is first passed to an embedding layer, in particular, the embedding representation of the target tokens is denoted as $\mathbf{T}$. Then the shifted token-level embedding representation of the plan path is used as the query, which is passed to three masked multihead attention layers followed by add and normalization layers. So far, we obtain the query representations $\mathbf{P}_k$, $\mathbf{P}_u$, and $\mathbf{P}_h$, which are used to attend to $\mathbf{K}$, $\mathbf{U}$, and $\mathbf{H}$, respectively.

Considering that the planned topics are mainly from the domain knowledge, and the target topic is essential to drive the entire conversation, we propose a knowledge-target mutual attention module (see Fig. 4). We use the encoded knowledge representation $\mathbf{K}$ and the planner's target representation $\mathbf{T}$ to calculate a relevance score via the scaled dot-product [12], the average of which can be viewed as a weight that the target influences the reasoning over the domain knowledge graph

$$\mathbf{K}_{\text{weight}} = \text{MeanPooling}\left(\frac{\mathbf{K}\mathbf{T}^{\top}}{\sqrt{d}}\right). \tag{4}$$

When using $\mathbf{P}_k$ to attend to $\mathbf{K}$, the computation can be further given by the following equation:

$$\mathbf{A}_k = \text{softmax}\left(\frac{\mathbf{P}_k\mathbf{K}^{\top}}{\sqrt{d}} * \mathbf{K}_{\text{weight}}\right)\mathbf{K} \tag{5}$$

where $\mathbf{A}_k$ is the attended representation, d is the hidden size. At the same time, it is also important to consider the user preferences and the conversation progress (i.e., dialog context) during planning. Therefore, we use query representations $\mathbf{P}_u$ and $\mathbf{P}_h$ to attend to $\mathbf{U}$ and $\mathbf{H}$, named "user preference cross attention" and "dialog context cross attention," respectively. Both attentions are computed as follows:

$$\mathbf{A}_u = \text{softmax}\left(\frac{\mathbf{P}_u\mathbf{U}^{\top}}{\sqrt{d}}\right)\mathbf{U} \tag{6}$$

$$\mathbf{A}_h = \text{softmax}\left(\frac{\mathbf{P}_h\mathbf{H}^{\top}}{\sqrt{d}}\right)\mathbf{H}. \tag{7}$$

To leverage different parts of the attended results strategically, we add an information fusion layer through gate control, which is formulated as follows:

$$\mathbf{A}_1 = \beta \cdot \mathbf{A}_u + (1-\beta) \cdot \mathbf{A}_h \tag{8}$$

$$\beta = \text{sigmoid}(\mathbf{W}_1[\mathbf{A}_u; \mathbf{A}_h] + \mathbf{b}_1) \tag{9}$$

$$\mathbf{A} = \gamma \cdot \mathbf{A}_k + (1-\gamma) \cdot \mathbf{A}_1 \tag{10}$$

$$\gamma = \text{sigmoid}(\mathbf{W}_2[\mathbf{A}_k; \mathbf{A}_1] + \mathbf{b}_2) \tag{11}$$

where $\mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{2d}$ are trainable weights, $\mathbf{b}_1, \mathbf{b}_2 \in \mathbb{R}$ are trainable biases. Here, $\mathbf{A}$ denotes the fused attended representation, which is then passed to a feed forward network followed by add and layer normalization. In general, such an architecture can be stacked to L layers for better planning capabilities [42], where L a hyperparameter.

D. Training and Inference

During training, we train our TPNet using the following cross-entropy loss:

$$\mathcal{L}_{\text{CE}}(\theta) = -\sum_{i=1}^{N} p(\mathbf{y}^{(i)}) \log p_\theta(\hat{\mathbf{y}}^{(i)}|\mathbf{k}^{(i)}, \mathbf{u}^{(i)}, \mathbf{h}^{(i)}) \tag{12}$$

where $p(\mathbf{y}^{(i)})$ is the distribution of the ground-truth plan path, while $p_\theta(\hat{\mathbf{y}}^{(i)}|\mathbf{k}^{(i)}, \mathbf{u}^{(i)}, \mathbf{h}^{(i)})$ is the distribution of the planner's output plan path conditioned on the inputs introduced before, θ denotes all trainable parameters.

For inference, we employ greedy search decoding to generate plan paths token by token. Additionally, we propose a simple SSD strategy (see Fig. 5) to facilitate to generate valid actions and topics. Suppose there is an action set containing all dialog actions in the dataset. At each step, the planner will first perform a prefix-based string search in the action set. Once a unique dialog action in the action set starts with the

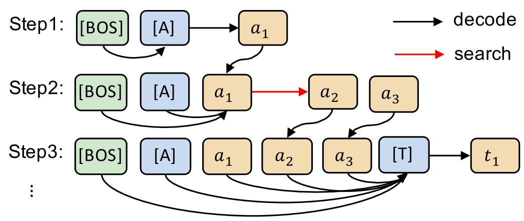

Fig. 5. Illustration of SSD.

decoded span, we will directly copy it without decoding for the following action tokens. A similar way is performed when decoding for topics, with the key difference that the planner will search in the topic set that contains all topics within the grounded domain knowledge graph.

V. Planning-Enhanced Dialog Generation

As shown in Fig. 2, our TPNet and dialog generation are trained in a pipeline manner, where the generated plan paths aim at guiding the system to generate more reasonable utterances. Since each planned path is in the "target-to-present" order, we take the last action a_t and the last topic z_t in a path as the guiding prompt. Here, z_t is further taken as the center topic to extract the corresponding triples (i.e., topics and topic-centric attributes) within one hop from the domain knowledge graph, denoted as $\mathcal{K}_t$. The $\mathcal{K}_t$ is expected to provide necessary knowledge beneficial to dialog generation. Note that we assume no domain knowledge is required when z_t is "NULL," i.e., a_t is "chit-chat." Accordingly, we set the extracted knowledge $\mathcal{K}_t$ as empty if this is the case.

Motivated by previous work employing prompt-based learning for dialog systems [43], [44], we adopt an important type of task-specific prompt named *natural language prompt* for our planning-enhanced dialog generation. In this work, we define our prompts as follows:

$$\begin{aligned}\mathcal{G}_A &= \text{“}\texttt{Next action is:}\text{”}\\ \mathcal{G}_T &= \text{“}\texttt{Next topic is:}\text{”}\\ \mathcal{G}_K &= \text{“}\texttt{Relevant knowledge include:}\text{”.}\end{aligned}$$

Finally, the concatenated text of the given dialog history $\mathcal{H}'$, the guiding prompts, and corresponding values are concatenated as the input for dialog generation, denoted as follows:

$$X = [\mathcal{H}';\ \mathcal{G}_A;\ a_t;\ \mathcal{G}_T;\ z_t;\ \mathcal{G}_K;\ \mathcal{K}_t]$$

where ";" denotes the concatenate operation. We then leverage various PLMs as backbone models to generate system utterances. We will describe the backbone models used for our experiments in Section VI-B.

VI. Experimental Setup

A. Datasets

We conduct experiments using the DuRecDial [11] and DuRecDial 2.0 [45] datasets, which are suitable for evaluating goal-directed dialog systems to the best of our knowledge. The system in the two datasets often leads conversations proactively instead of passively responding to users, with various interactive actions such as chitchat, question answering, and recommendation. We first briefly introduce the two datasets and then describe how to repurpose the datasets.

TABLE II
Statistics of the System's Dialog Actions

Category	System's Dialogue Action	Percentage
Chitchat	Say goodbye Greetings	12.90 % 7.00 %
User Exploration	Respond Q&A Respond to user's query about weather Respond to user's query about date Respond to music on demand Respond to news on demand	3.61 % 2.40 % 1.60 % 0.92 % 0.70 %
Topic Elicitation	Chat about movie/music stars Ask questions	15.79 % 2.57 %
Rec.	Movie recommendation Music recommendation Play music News recommendation Point-of-Interest recommendation Food recommendation	14.29 % 11.57 % 10.52 % 7.18 % 5.04 % 3.93 %

The original DuRecDial [11] and DuRecDial 2.0 [45] datasets are collected from crowdsourced human-to-human conversations. One person is defined as the seeker (the user's role) and the other as the recommender (the system's role) in a conversation. The recommender is required to proactively lead the conversation and make recommendations by introducing new topics. Each seeker is equipped with a user profile, which contains user attributes (e.g., name, age range) and his or her past preference information. All user attributes in their profiles are randomly chosen from specific candidate templates to preserve privacy. To achieve smooth conversations with the seeker, the recommender has a domain knowledge graph consisting of domain-specific topics (e.g., movies, music, and food) with related attributes. More importantly, a specific dialog action-topic pair is annotated with respect to the recommender at each turn of the conversation. A specific amount of domain knowledge relevant to each topic that appeared in the conversation is grounded in each conversation. The DuRecDial dataset is in Chinese and contains about 10 k multiturn conversations. The DuRecDial 2.0 dataset has about 8.2 k bilingual multiturn conversations, where we adopt the English version for experiments.

Since no explicit targets (i.e., target actions and topics) are annotated in the original datasets, we repurpose the two datasets through automatic target construction. We regard the topic the user has accepted at the end of each conversation as the target topic and treat the system's corresponding action as the target action. Each target topic is guaranteed to appear in the domain knowledge grounded on each conversation. We filter out those conversations without introducing any recommendation topics. For our TPNet training, we follow the "target-to-present" order to construct each golden plan path, which consists of a sequence of distinct action-topic pairs from the target turn of the conversation to the present turn. We use two special tokens [A] and [T] to separate an action and a topic in the constructed plan path, respectively. In total, the number of topics in the DuRecDial and DuRecDial 2.0 is 678 (including a NULL topic) and 628 (including a NULL

TABLE III
STATISTICS OF THE PROCESSED DATASETS, WHERE "CONV." DENOTES "CONVERSATION," "UTTER." DENOTES "UTTERANCE"

Dataset		#Conv.	#Utter.	Conv. Turn #Max.	Conv. Turn #Avg.	Knowledge Triples #Max.	Knowledge Triples #Avg.
DuRecDial	Train	5,400	84,869	14	7.9	71	27.1
	Dev	800	12,508	12	7.8	57	26.9
	Test	1,804	28,809	13	8.0	71	26.0
DuRecDial 2.0	Train	410,4	66,716	13	8.1	37	27.5
	Dev	608	9,918	14	8.2	36	27.7
	Test	1,368	22,085	13	8.1	36	27.5

topic), respectively. The system's dialog actions across the two datasets are almost identical, with the statistics reported in Table II. On average, each conversation has 4.5 different "action and topic" transitions from the start to the target. Following the splitting criterion in [11], our processed datasets are split into train/dev/test sets, with the statistics shown in Table III.

B. Baseline Methods

To validate the performance of our method on the end task, we first compare it with several competitive models for general dialog generation.

1) *Transformer [12]:* It is an encoder–decoder model widely used for text generation. We use OpenNMT's [46] implementation with its suggested parameters for training.
2) *DialoGPT [2]:* It is an autoregressive generation model pretrained using large-scale English dialog corpora. We employ the *CDial-GPT* [47] for fine-tuning our dataset in Chinese.
3) *BART [48]:* It is an encoder–decoder pretrained model with denoising for natural language generation.
4) *GPT-2 [49]:* It is a pretrained autoregressive generation model for language generation. We use the GPT2-base version and fine-tune it for dialog generation.

Note that for a fair comparison, the above models concatenate all given input texts as the model input and generate system utterances directly. To explore to what extent our TPNet affects dialog generation, we also employ the above models as our backbone models, then follow the description in Section V to generate utterances.

We also compare our planning-enhanced method with the following the state-of-the-art recommendation-oriented dialog generation models, where they follow a predict-then-generate paradigm.

1) *MGCG_G [11]:* It employs the next predicted dialog action and topic to guide system utterance generation. Following our problem setting, we rerun the released code[1] on the repurposed datasets.
2) *KERS [26]:* It has a knowledge-enhanced mechanism for recommendation dialog generation. Similarly, we rerun the released code[2] on the repurposed datasets.

[1]https://github.com/PaddlePaddle/Research/tree/master/NLP/ACL2020-DuRecDial

[2]https://github.com/z562/KERS

To further explore the effect of planning for goal-directed dialog systems, we compare our TPNet with the following planning methods.

1) *MGCG [11]:* It employs a convolutional neural network [50] to perform multitask predictions for the next dialog action and topic. However, it assumes that ground-truth historical dialog actions and topics are known for a system. In our problem formulation, only the target (i.e., a target action paired with a target topic) is provided. The system needs to plan all interim dialog actions and topics to achieve the goal. For a fair comparison, we take the same input as our problem formulation to predict the next dialog action and topic for MGCG.
2) *KERS [26]:* It aims to generate the next dialog action and topic based on a transformer [12] network. Similarly, we take the same input as our problem formulation for KERS.
3) *BERT [14]:* Based on the intuition of multitask predictions, we add two fully connected layers upon BERT to predict the system's next dialog action and topic jointly. We adopt the BERT-base version model for fine-tuning.
4) *GPT-2 [49]:* Apart from predictions, planning can also be performed in a generation manner. We directly generate the system's next dialog action and topic token by token by fine-tuning the pretrained GPT-2 base version model.

For all involved PLMs, we adopt the pretrained models released in the Huggingface [51] library for experiments.

C. Evaluation Metrics

1) Automatic Evaluation: Following many previous studies in dialog generation, we adopt widely used metrics including *perplexity* (PPL), *word-level F1*, BLEU [52], *distinct* (DIST) [53], and *knowledge F1* (Know. F1) [11]. In detail, the PPL and DIST measure the fluency and the diversity of generated system utterances, respectively. The *F1* score estimates the precision and recall of the generated utterance at the word level. The BLEU score calculates n-gram overlaps between generated and gold utterances. The Know. F1 evaluates the performance of generating correct knowledge (e.g., topics, attributes) from the domain knowledge triples. However, no knowledge is labeled for gold system utterances in original datasets. We first conduct strict string matching to search for the entities from the domain knowledge that also appear in each gold system utterance as the knowledge label. Some knowledge entries (*object* in the triple ⟨*subject, relation, object*⟩) are in the form of long texts (e.g., topic-associated attributes), and they are paraphrased during conversations. We thereby compute word-based recall scores between knowledge entries and gold system utterances. We take the knowledge entries whose recall scores exceed the threshold of 0.55 as the pseudo label. For knowledge F1 evaluation, we take the same threshold (i.e., 0.55) to examine whether a knowledge entry is hit in the generated utterances.

For conversation planning, we adopt *accuracy* (Acc.) to evaluate the predicted action or topic for prediction-based models, following [11]. For generation-based models, we take

TABLE IV
RESULTS OF DIALOG GENERATION ON DURECDIAL. * DENOTES SIGNIFICANT IMPROVEMENT OVER THE BACKBONE MODEL (t-TEST, $p < 0.05$)

Model		PPL (↓)	F1 (%)	BLEU-1 / 2	DIST-1 / 2	Know. F1 (%)
Generation	Transformer	22.83	27.95	0.224 / 0.165	0.001 / 0.005	17.73
	CDial-GPT	5.45	29.60	0.287 / 0.213	0.005 / 0.036	27.26
	BART	6.29	34.07	0.312 / 0.242	**0.008** / 0.067	38.16
	GPT-2	4.93	38.93	0.367 / 0.291	0.007 / 0.058	43.83
Predict-then-generate	MGCG_G	18.76	33.48	0.279 / 0.203	0.007 / 0.043	35.12
	KERS	12.55	34.04	0.302 / 0.220	0.005 / 0.030	40.75
Ours	Transformer w/ TPNet	20.28	28.59*	0.252 / 0.213	0.002 / 0.008	24.39*
	CDial-GPT w/ TPNet	5.87	31.98*	0.305 / 0.262	0.006 / 0.041	35.68*
	BART w/ TPNet	5.23	37.22*	0.338 / 0.255	**0.008** / **0.083**	44.52*
	GPT-2 w/ TPNet	**4.22**	**41.53***	**0.379** / **0.301**	0.007 / 0.075	**48.81***

the generated action or topic at the evaluating turn and calculate accuracy by an exact match for a fair comparison. Due to the nature of conversations, multiple temporary planning strategies might be reasonable before completing the goal. Following [54], we also adopt *bigram accuracy* (Bi. Acc.) for evaluation. It expands labels by counting the system's gold actions (or topics) within the previous turn and the subsequent turn.

2) Human Evaluation: Similar to [11], we conduct human evaluation from both turn-level and conversation-level aspects. During turn-level evaluation, we randomly select 100 samples from test sets and ask each model to produce system utterances according to the input. Three well-educated annotators are required to mark scores for different models from *appropriateness* and *informativeness*. The *appropriateness* measures if a generated system utterance can complete the current plan and respond to the context appropriately. The *informativeness* measures if a model can fully use domain knowledge to generate an informative utterance. For fairness, all model names are masked to annotators during the evaluation process.

For conversation-level evaluation, we let each model interact with our human annotators, which indicates that the model will take its generated utterance in the previous turn as a part of the dialog history in the current turn. To ensure the evaluation covers a wide range of targets, we randomly sample five different target actions from the test dataset, with each action consisting of ten different target topics. In total, we have 50 different dialog targets for evaluation. To examine whether a model can lead the conversation naturally and proactively to reach the designated target, we do not expose the target action and topic to human annotators during human-model conversations. Besides, human annotators are asked to be consistent with the given user profile. All human-model conversations are limited to no more than 12 turns. At the end of each conversation, we expose the designated target to human annotators and ask them to mark scores for different models from the following perspectives: 1) *proactivity*, which measures if a model can proactively lead new actions/topics in the conversation; 2) *coherence*, which measures the overall fluency and naturalness of the whole dialog generation; and 3) *goal success*, which estimates whether the designated target is achieved.

For all the above metrics, human evaluation scores are settled from {0, 1, 2}, where a higher score denotes better performance. The averaged score of different human annotators is reported as the evaluation result for each model. The agreement among the annotators is measured by Fleiss's kappa [55].

D. Implementation Details

We implement our TPNet based on the Huggingface's transformers [51] codebase. We employ character-based tokenization for the Chinese DuRecDial dataset and the default BERT tokenizer for the English DuRecDial 2.0 dataset. The pretrained BERT-base model is used during encoding, with a vocabulary size of 21 128 and a hidden size of 768. The user memory network adopts the vocabulary shared by BERT and randomly initializes the memory embeddings with the number of memory hops $K = 3$. The target-driven conversation planner is stacked into 12 layers with eight attention heads, similarly using the vocabulary shared by BERT. All hidden sizes are set to 768. We adopt the Adam [56] optimizer with an initial learning rate of $2e$-5. We train TPNet for ten epochs and warm up over the first 3000 training steps with linear decay. We select the best model based on the performance of the validation set. For TPNet inference, we adopt greedy search decoding with our SSD strategy, with a maximum decoding length of 256. During planning-enhanced dialog generation, we employ transformer, DialogGPT/CDial-GPT, BART, and GPT-2 as our backbone models. Each backbone model adopts the same parameter setting as that in baseline experiments. During generation, the maximum decoding length is set to 80.

VII. EXPERIMENTAL RESULTS AND ANALYSIS

A. Evaluation Results of Dialog Generation

Our automatic evaluation results of dialog generation on the two datasets are reported in Tables IV and V, respectively. The best result in terms of the corresponding metric is highlighted in boldface. We observe that the vanilla transformer performs inferior compared with other models since it has neither conversation planning nor pretraining. As pretrained models, CDialGPT/DialoGPT, BART, and GPT-2 achieve much better performance over various metrics, which shows they are powerful to generate fluent and diverse utterances. For dialog models based on the "predict-then-generate" paradigm, we observe that MGCG_G and KERS are able to achieve better results than transformer and CDialGPT over multiple

TABLE V

RESULTS OF DIALOG GENERATION ON DURECDIAL 2.0. * DENOTES SIGNIFICANT IMPROVEMENT OVER THE BACKBONE MODEL (t-TEST, $p < 0.05$)

Model		PPL (↓)	F1 (%)	BLEU-1 / 2	DIST-1 / 2	Know. F1 (%)
Generation	Transformer	32.56	25.24	0.216 / 0.157	0.014 / 0.050	19.35
	DialoGPT	5.44	33.12	0.295 / 0.201	0.021 / 0.070	34.86
	BART	7.62	35.83	0.273 / 0.168	0.029 / 0.091	36.16
	GPT-2	5.95	33.58	0.302 / 0.205	0.023 / 0.079	35.88
Predict-then-generate	MGCG_G	26.68	32.26	0.293 / 0.182	0.016 / 0.051	29.35
	KERS	22.60	30.11	0.282 / 0.178	0.017 / 0.060	33.08
Ours	Transformer w/ TPNet	28.35	26.11*	0.228 / 0.163	0.015 / 0.050	25.50*
	DialoGPT w/ TPNet	**5.36**	34.45*	0.303 / 0.214	0.022 / 0.074	36.15*
	BART w/ TPNet	6.33	**36.28***	0.296 / 0.204	**0.030 / 0.093**	**40.22***
	GPT-2 w/ TPNet	5.87	34.62*	**0.308 / 0.217**	0.025 / 0.082	38.80*

metrics. In view of the fact that MGCG_G and KERS are trained without using PLMs, their improvements are mainly from the planning of the next dialog action and topic, which guides the model to generate more informative and reasonable utterances.

As shown in Table IV, we observe that all backbone models achieve significant improvements with the benefit of our TPNet. For example, transformer with TPNet obtains a remarkably improved Know. F1 (from 17.73% to 24.39%). Compared to the vanilla transformer, our TPNet provides reasonable dialog actions and appropriate topics with the necessary knowledge to guide the generation of system utterances. Though BART and GPT-2 are powerful for dialog generation over many metrics, they achieve much better performance with the help of our TPNet, especially in terms of word-level F1 and Know. F1. It demonstrates that our TPNet-enhanced method effectively generates more appropriate utterances. We observe a similar trend on the DuRecDial 2.0 dataset as shown in Table V. We find that almost all models' Know. F1 scores are much lower than the results on the DuRecDial dataset. Our analyses indicate that in the DuRecDial 2.0 dataset, the topics and topic-related attributes in the grounded domain knowledge for each conversation are much noisier, making it more challenging to distinguish different topics and generate appropriate knowledge-rich words accordingly.

B. Evaluation Results of Conversation Planning

To further validate the effect of conversation planning for the formulated goal-directed dialog systems, we compare TPNet with different conversation planning methods. The experimental results on DuRecDial and DuRecDial 2.0 datasets are reported in Tables VI and VII, respectively. We observe that it is more difficult for all baseline methods to predict or generate dialog topics correctly than dialog actions. The reason is that the number of topics is much larger than the actions in the two datasets. As a generation method, KERS achieves significantly higher topic accuracy than MGCG. A similar trend is also observed between GPT-2 and BERT, as shown in Table VI. It indicates the effectiveness of generation for conversation planning. According to Table VII, it is more difficult for all models to achieve high topic accuracy scores. Our analyses reveal that in the DuRecDial 2.0 dataset, the topics and topic-related attributes in the grounded domain knowledge for each conversation are much noisier, making it challenging to plan appropriate topics with smooth transitions.

TABLE VI

RESULTS OF CONVERSATION PLANNING ON DURECDIAL

Model	Dialogue Action		Dialogue Topic	
	Acc. (%)	Bi. Acc. (%)	Acc. (%)	Bi. Acc. (%)
MGCG	84.78	86.52	64.31	66.65
KERS	89.17	90.49	76.34	79.33
BERT	90.19	91.35	83.53	85.61
GPT-2	91.76	93.03	86.24	87.47
Ours (TPNet)	**93.58**	**95.11**	**91.92**	**93.53**

TABLE VII

RESULTS OF CONVERSATION PLANNING ON DURECDIAL 2.0

Model	Dialogue Action		Dialogue Topic	
	Acc. (%)	Bi. Acc. (%)	Acc. (%)	Bi. Acc. (%)
MGCG	85.23	88.05	57.68	58.29
KERS	86.15	88.75	63.18	65.06
BERT	91.32	93.26	66.87	67.77
GPT-2	91.64	93.88	65.63	66.80
Ours (TPNet)	**93.21**	**94.62**	**83.26**	**84.49**

Our TPNet achieves substantial improvements on all the metrics compared to the baseline methods. For example, TPNet improves the topic accuracy from 70%∼80% to over 90% on the DuRecDial dataset. Even though for the more challenging DuRecDial 2.0 dataset, our TPNet still achieves remarkable improvements in topic planning. Our TPNet generates a path consisting of dialog actions and topics in a target-driven manner, which considers the essential role of the designated target. It verifies that TPNet effectively makes proper plans consisting of dialog actions and topics.

C. Ablation Study of TPNet

To verify the effectiveness of each module in TPNet, we also conduct an ablation study. We focus on the following proposed modules or mechanisms and set them for ablation experiments accordingly: 1) without the hop encoding strategy in target-aware graph attention transformer (w/o Hop-E); 2) without the user memory updating mechanism (w/o UMU); 3) without the knowledge-target mutual attention mechanism (w/o K-T MA), which denotes we directly use the query representation $\mathbf{P}_k$ to attend to the domain knowledge representation

TABLE VIII

ABLATION STUDY RESULTS OF TPNET

Model	Dialogue Action Acc. (%)	Dialogue Action Bi. Acc. (%)	Dialogue Topic Acc. (%)	Dialogue Topic Bi. Acc. (%)
TPNet (full)	**93.58**	**95.11**	**91.92**	**93.53**
w/o Hop-E	92.70	94.78	90.63	92.15
w/o UMU	90.04	91.84	90.21	92.38
w/o K-T MA	93.06	94.14	88.38	89.56
w/o SSD	93.55	95.08	90.93	92.56

K in a simple cross attention manner; and 4) without the SSD strategy (w/o SSD). From the ablation study results shown in Table VIII, we observe that each module or mechanism contributes to planning better actions and topics. In particular, the performance of TPNet w/o K-T MA deteriorates rapidly in topic planning. It shows that our knowledge-target mutual attention is essential since it explicitly leverages the influence of the target information that drives the whole planning process.

D. Analysis of Goal Success

For goal-directed dialog systems, it is essential to validate a model of how well it achieves the expected goal, such as to measure whether a model generates the target topic correctly when appropriate. Due to the nature of conversations, different models might arrive at the target topic with different turns. To this end, we take each model's generated utterances of all turns during automatic evaluation on test sets for analysis. We compute the ratio of generating the target topic correctly without restricting how many turns a model consumes, and define it as the *goal success rate* (Succ.). We define the ratio of never generating the target topic successfully among all turns in a conversation as the *never success rate* (*Never*). Our evaluation results of goal success on the two datasets are reported in Table IX. Among the baselines, MGCG_G and KERS still struggle to achieve high *Succ.* scores although they conduct dialog planning. It shows that their "predict-then-generate" paradigm is not effective enough to achieve the goal, mainly because the paradigm ignores the crucial role of target actions and topics since it concerns more about predicting the next dialog action and topic. In comparison, our TPNet employs a target-driven manner to plan a path, which considers the target action and topic during planning throughout the conversation. Performance gains from our TPNet provide the backbone dialog generation models with more appropriate prompts, making them more likely to generate the target topic when appropriate. As shown in Table IX, our methods outperform all baseline models with significantly higher *Succ.* scores and lower *Never* scores.

E. Human Evaluation Results

We select several representative models for human evaluation, including MGCG_G, CDial-GPT (w/ and w/o TPNet), and GPT-2 (w/ and w/o TPNet). Our turn-level and conversation-level evaluation results are shown in Figs. 6 and 7, respectively. The Fleiss's kappa scores are mainly distributed in [0.4, 0.6], which denotes moderate

TABLE IX

EVALUATION RESULTS OF GOAL SUCCESS. SIGNIFICANT DIFFERENCES OVER BASELINES ARE MARKED WITH * (t-TEST, $p < 0.05$)

Model	DuRecDial Succ. (↑)	DuRecDial Never (↓)	DuRecDial 2.0 Succ. (↑)	DuRecDial 2.0 Never (↓)
Transformer	36.7%	63.3%	34.0%	66.0%
MGCG_G	46.8%	53.2%	32.2%	67.8%
KERS	50.5%	49.5%	40.6%	59.4%
BART	55.8%	44.2%	52.8%	47.2%
GPT-2	61.2%	38.8%	51.5%	48.5%
Ours (BART w/ TPNet)	71.5%*	28.5%*	**63.6%***	**36.4%***
Ours (GPT-2 w/ TPNet)	**74.7%***	**25.3%***	60.7%*	39.3%*

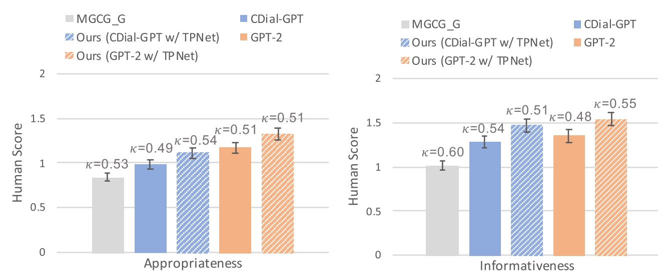

Fig. 6. Turn-level human evaluation results. (a) Appropriateness. (b) Informativeness. κ denotes Fleiss's kappa.

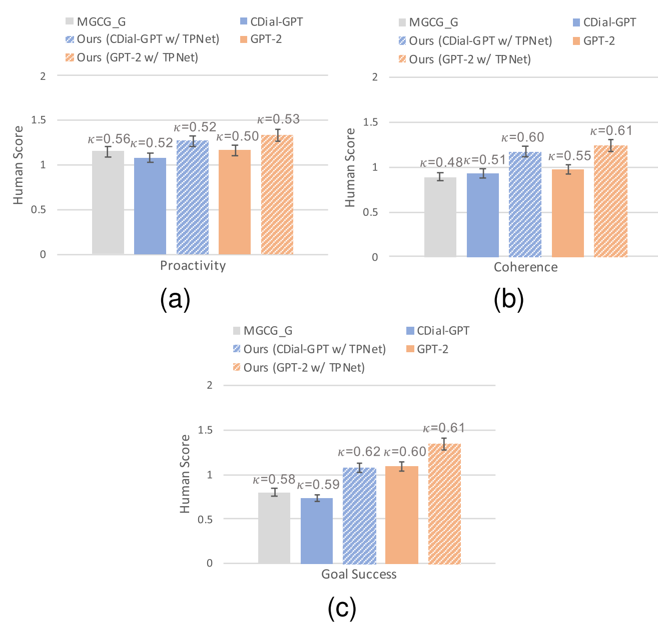

Fig. 7. Conversation-level human evaluation results. (a) Proactivity. (b) Coherence. (c) Goal success. κ denotes Fleiss's kappa.

interannotator agreement. We observe that from the turn-level aspect, though almost all models obtain comparable informativeness scores, they vary remarkably in terms of appropriateness. It shows that with the guidance of TPNet, both CDial-GPT and GPT-2 can generate more appropriate system utterances in response to dialog context. According to the conversation-level evaluation results shown in Fig. 7, we find that our models with TPNet obtain significantly higher coherence scores and goal success scores than the original backbone models. It indicates that TPNet effectively enables

Goal: target action = 音乐推荐 (Music Recommendation) target topic =『最好的幸福』("The Gift of Happiness")

	MGCG_G	Predicted Action | Topic
User: Bot:	你好，请放一首『不要害怕』。(Hello, please play the song "Do Not Be Afraid".) 正在为你播放哦！(It is playing now!)	回复音乐点播|『不要害怕』(Respond to user's order about music |"Do Not Be Afraid")
User: Bot:	谢谢了，我主要现在心情不太好。(Thank you. I'd like to listen to it since I'm in a bad mood.) 你喜欢何炅老师就好。(It's good if you like Jiong He.) [Not coherent enough]	关于明星的聊天|何炅 (Chat about the star | Jiong He)
User: Bot:	我是挺喜欢的！(I sincerely love him!) 你喜欢就好。(It's good if you like.) [Not informative]	关于明星的聊天|何炅 (Chat about the star | Jiong He)
User: Bot:	好吧。(OK, fine.) 现在就可以为你放栀子花开哦，要听吗？(I can play the song "Gardenia in Blossom" for you now. Would you like to have a try?) [Not coherent]	音乐推荐|『栀子花开』 (Music rec. | "Gardenia in Blossom")
User: Bot:	算了，你有其他的歌可以推荐吗？(Um, do you have any other song to recommend?) 正在为你播放哦。(It is playing now!) [Not achieve the goal]	音乐推荐|『栀子花开』 (Music rec. | "Gardenia in Blossom")
User: Bot:	嗯，再见！(Well, goodbye!) 再见！(Goodbye!)	再见|NULL (Say goodbye | NULL)

(a)

Goal: target action = 音乐推荐 (Music Recommendation) target topic =『最好的幸福』("The Gift of Happiness")

	GPT-2	(No Planning)
User: Bot:	你好，请放一首『不要害怕』。(Hello, please play the song "Do Not Be Afraid".) 好的，已经播放，请你欣赏。(OK. It is playing now. Hope you appreciate it!)	
User: Bot:	谢谢了，我主要现在心情不太好。(Thank you. I'd like to listen to it since I'm in a bad mood.) 这样啊，那你喜欢何炅吗？我给你推荐一首何炅的『时光匆匆』 (Well, do you like Jiong He? I'd like to recommend his another song "How Time Flies".) [Not coherent enough]	
User: Bot:	我是挺喜欢的！(I sincerely love him!) 那『一路走过』这首歌也不错哦，还是电影『无问西东』的主题曲呢！ (The song "All the Way Through" is also good. It is the theme song of the movie "Forever Young".)	
User: Bot:	这首歌确实不错。(Yeah, it's really a good song.) 还有『思念的距离』呢？这首歌太温柔了，温柔得想哭。(How about "The Distance of Miss"? This song is so gentle and soft that I want to cry.)	
User: Bot:	是的。(Yeah.) 嗯嗯好的，你听听看，很好听的。(Well, you can have a try. It's pretty good.) [Not achieve the goal]	
User: Bot:	嗯，再见！(Well, goodbye!) 再见！(Goodbye!)	

(b)

Goal: target action = 音乐推荐 (Music Recommendation) target topic =『最好的幸福』("The Gift of Happiness")

	Ours (GPT-2 w/ TPNet)	Planned Path by TPNet
User: Bot:	你好，请放一首『不要害怕』。(Hello, please play the song "Do Not Be Afraid".) 好的，正在播放，请你欣赏。(OK. It is playing now. Hope you appreciate it!)	回复音乐点播|『不要害怕』← 关于明星的聊天|何炅 ← 音乐推荐|『最好的幸福』 (Respond to user's order about music |"Do Not Be Afraid" ← Chat about the star | Jiong He ← Music Rec. | "The Gift of Happiness")
User: Bot:	谢谢了，我主要现在心情不太好。(Thank you. I'd like to listen to it since I'm in a bad mood.) 这样啊，那你心情不好的时候是喜欢听何炅的歌吗？(Well, do you like to listen to Jiong He's songs when you are in a bad mood?) [More coherent]	关于明星的聊天|何炅 ← 音乐推荐|『最好的幸福』 (Chat about the star | Jiong He ← Music Rec. | "The Gift of Happiness")
User: Bot:	是啊，他的歌很温暖。(Yeah. His songs often make me feel warm.) 那我给你推荐一首他的歌『最好的幸福』，还是电视剧『别爱我』的插曲呢。 (Let me recommend one of his songs named "The Gift of Happiness". It is also used as an interlude in the TV series "Don't Love Me".)	音乐推荐|『最好的幸福』 (Music Rec. | "The Gift of Happiness")
User: Bot:	这个我倒是没听过。(I've never listened before.) 那你可以听听看，也许会有不一样的感觉呢。(You can have a try. Maybe you will have a different feeling.)	音乐推荐|『最好的幸福』 (Music Rec. | "The Gift of Happiness")
User: Bot:	好的，谢谢。(OK. Thank you.) 现在为你播放『最好的幸福』。("The Gift of Happiness" is playing now.) [Succeed to achieve the goal]	音乐推荐|『最好的幸福』 (Music Rec. | "The Gift of Happiness")
User: Bot:	真好听，再见！(Very good. Goodbye!) 再见！(Goodbye!)	再见|NULL ← 音乐推荐|『最好的幸福』 (Say goodbye | NULL ← Music Rec. | "The Gift of Happiness")

(c)

Fig. 8. Generated cases from conversation-level human evaluation. The bot's utterances are generated by (a) MGCG_G, (b) GPT-2, and (c) Ours.

dialog models to generate more coherent utterances. More importantly, our TPNet-enhanced dialog generation method is more likely to drive the conversation to reach the target topic and then make successful recommendations.

F. Case Study and Visualization Analysis

To present the dialog generation quality of different models under the goal-directed setting, we conduct some case studies. Here, we designate the same target action and topic, and show generated utterances by three different models (i.e., MGCG_G, GPT-2, and GPT-2 w/TPNet) from conversation-level human evaluation in Fig. 8. As shown in Fig. 8(a), we observe that MGCG_G often fails to generate coherent utterances and is incapable to maintain an engaging conversation with the user. Though MGCG_G performs planning first, it fails to predict a correct topic when necessary, causing the model fails to achieve the goal (i.e., recommend the song "*The Gift*

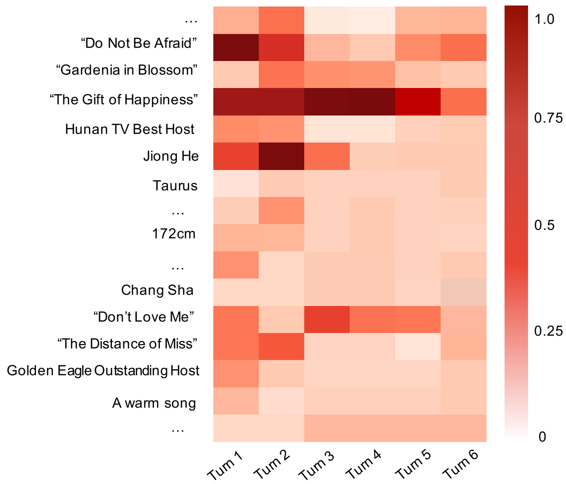

Fig. 9. Visualization of TPNet's attention weights over the domain knowledge.

of Happiness”) in the end. For the case shown in Fig. 8(b), GPT-2 can generate more fluent and informative utterances in general. However, it still fails to achieve the goal, which is might because it struggles to know when to generate the target action and topic. In comparison, the case shown in Fig. 8(c) demonstrates that our TPNet can plan a path consisting of reasonable dialog actions and topics (including the target action and topic). With the guidance of our TPNet, GPT-2 is able to generate more coherent utterances. More importantly, our TPNet can guide GPT-2 to achieve the goal successfully since TPNet generates the target topic “*The Gift of Happiness*” when necessary.

We further visualize the attention distribution of TPNet for the case in Fig. 8 to explain the planning process. We first aggregate token-level attention weights at each turn into node-level weights over the domain knowledge, followed by a normalization operation. We draw attentions varying by dialog turns, which are consistent with that in Fig. 8. The TPNet's attention distribution is visualized in Fig. 9. It shows that at the 1-st turn, the model focuses more on the song “*Do Not Be Afraid*” than the target topic “*The Gift of Happiness*” because it needs to respond to the user's query about this song. At the 2-nd turn, the model mainly attends to the topic “*Jiong He*” so it generates a planned path containing “*Jiong He*” as the next dialog topic. Subsequently, the model mainly attends to the target topic “*The Gift of Happiness*.” In such cases, it is time for the system to move forward toward recommending the target topic. Therefore, GPT-2 with our TPNet can generate reasonable utterances to complete this process (see Fig. 8).

VIII. Conclusion and Future Work

In this work, we push forward from the reactive recommendation dialog paradigm toward the promising goal-directed dialog paradigm. We propose a TPNet to plan a dialog path, which aims to lead a conversation with the user to achieve the goal step by step. The planned path provides our model with specific dialog actions and topics, and facilitates a dialog model to generate proper utterances in a pipeline manner. Experimental results and analyses demonstrate the effectiveness of our method.

This work still has some limitations for further improvement. First, our pipelined framework has error propagation, which might be a typical issue of most existing pipelined methods. We find that the performance of dialog generation is prone to drop when our TPNet fails to plan the path appropriately. We intend to alleviate this issue by introducing some techniques in the cascaded generation, such as noisy channel models [57], [58]. Second, our planning-enhanced method still suffers from a significant gap in fully achieving the goal. We leave this as our future work and will explore controllable generation techniques to further enhance dialog generation.

References

[1] F. Cui, Q. Cui, and Y. Song, “A survey on learning-based approaches for modeling and classification of human–machine dialog systems,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 4, pp. 1418–1432, Apr. 2021.

[2] Y. Zhang et al., “DIALOGPT: Large-scale generative pre-training for conversational response generation,” in *Proc. 58th ACL*, Jul. 2020, pp. 270–278.

[3] M. Yang, W. Huang, W. Tu, Q. Qu, Y. Shen, and K. Lei, “Multitask learning and reinforcement learning for personalized dialog generation: An empirical study,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 49–62, Jan. 2021.

[4] A. Madotto, C.-S. Wu, and P. Fung, “Mem2Seq: Effectively incorporating knowledge bases into end-to-end task-oriented dialog systems,” in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 1468–1478.

[5] C.-S. Wu, R. Socher, and C. Xiong, “Global-to-local memory pointer networks for task-oriented dialogue,” in *Proc. 7th ICLR*, 2019, pp. 1–25.

[6] Q. Chen et al., “Towards knowledge-based recommender dialog system,” in *Proc. EMNLP-IJCNLP*, Nov. 2019, pp. 1803–1813.

[7] D. Kang, A. Balakrishnan, P. Shah, P. Crook, Y.-L. Boureau, and J. Weston, “Recommendation as a communication game: Self-supervised bot-play for goal-oriented dialogue,” in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, Hong Kong, 2019, pp. 1951–1961.

[8] D. Jannach, A. Manzoor, W. Cai, and L. Chen, “A survey on conversational recommender systems,” *ACM Comput. Surv.*, vol. 54, no. 5, pp. 1–36, Jun. 2022.

[9] W. Ma, R. Takanobu, and M. Huang, “CR-Walker: Tree-structured graph reasoning and dialog acts for conversational recommendation,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2021, pp. 1839–1851.

[10] Z. Liang et al., “Learning neural templates for recommender dialogue system,” in *Proc. EMNLP*, Nov. 2021, pp. 7821–7833.

[11] Z. Liu, H. Wang, Z.-Y. Niu, H. Wu, W. Che, and T. Liu, “Towards conversational recommendation over multi-type dialogs,” in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 1036–1049.

[12] A. Vaswani et al., “Attention is all you need,” in *Proc. NIPS*, 2017, pp. 5998–6008.

[13] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, “End-to-end memory networks,” in *Proc. 28th NIPS*, 2015, pp. 2440–2448.

[14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. ACL Conf. NAACL HLT*, Minneapolis, MN, USA, Jun. 2019, pp. 4171–4186.

[15] J. Tang, T. Zhao, C. Xiong, X. Liang, E. Xing, and Z. Hu, “Target-guided open-domain conversation,” in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, Italy, 2019, pp. 5624–5634.

[16] J. Qin, Z. Ye, J. Tang, and X. Liang, “Dynamic knowledge routing network for target-guided open-domain conversation,” in *Proc. AAAI*, 2020, pp. 8657–8664.

[17] P. Zhong, Y. Liu, H. Wang, and C. Miao, “Keyword-guided neural conversational model,” in *Proc. AAAI*, 2021, pp. 14568–14576.

[18] W. Wu et al., “Proactive human–machine conversation with explicit conversation goal,” in *Proc. 57th ACL*, Florence, Italy, Jul. 2019, pp. 3794–3804.

[19] K. Zhou, Y. Zhou, W. X. Zhao, X. Wang, and J.-R. Wen, "Towards topic-guided conversational recommender system," in *Proc. 28th Int. Conf. Comput. Linguistics*, Barcelona, Spain, 2020, pp. 4128–4139.

[20] J. Xu, H. Wang, Z.-Y. Niu, H. Wu, W. Che, and T. Liu, "Conversational graph grounded policy learning for open-domain conversation generation," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 1835–1845.

[21] J. Xu, H. Wang, Z. Niu, H. Wu, and W. Che, "Knowledge graph grounded goal planning for open-domain conversation generation," in *Proc. AAAI*, 2020, pp. 9338–9345.

[22] S. Ghazarian, Z. Liu, T. Chakrabarty, X. Ma, A. Galstyan, and N. Peng, "DiSCoL: Toward engaging dialogue systems through conversational line guided response generation," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol., Demonstrations*, 2021, pp. 26–34.

[23] K. Sevegnani, D. M. Howcroft, I. Konstas, and V. Rieser, "OTTers: One-turn topic transitions for open-domain dialogue," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics 11th Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 2492–2504.

[24] J. Liu, F. Pan, and L. Luo, "GoChat: Goal-oriented chatbots with hierarchical reinforcement learning," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 1793–1796.

[25] S. A. Hayati, D. Kang, Q. Zhu, W. Shi, and Z. Yu, "INSPIRED: Toward sociable recommendation dialog systems," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2020, pp. 8142–8152.

[26] J. Zhang, Y. Yang, C. Chen, L. He, and Z. Yu, "KERS: A knowledge-enhanced framework for recommendation dialog systems with multiple subgoals," in *Proc. Findings Assoc. Comput. Linguistics, EMNLP*, Punta Cana, Dominican Republic, 2021, pp. 1092–1101.

[27] D. Lin, J. Wang, and W. Li, "Target-guided knowledge-aware recommendation dialogue system: An empirical investigation," in *Proc. 3rd KaRS 5th ComplexRec Joint Workshop*, 2021, pp. 1–10.

[28] Y. Li, S. A. Hayati, W. Shi, and Z. Yu, "DEUX: An attribute-guided framework for sociable recommendation dialog systems," 2021, *arXiv:2105.00825*.

[29] R. Li, S. Kahou, H. Schulz, V. Michalski, L. Charlin, and C. Pal, "Towards deep conversational recommendations," in *Proc. 32nd NIPS*, 2018, pp. 9748–9758.

[30] Y. Sun and Y. Zhang, "Conversational recommender system," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 235–244.

[31] C. Gao, W. Lei, X. He, M. de Rijke, and T.-S. Chua, "Advances and challenges in conversational recommender systems: A survey," 2021, *arXiv:2101.09459*.

[32] H. Xu et al., "User memory reasoning for conversational recommendation," in *Proc. 28th COLING*, Barcelona, Spain, Dec. 2020, pp. 5288–5308.

[33] K. Xu, J. Yang, J. Xu, S. Gao, J. Guo, and J.-R. Wen, "Adapting user preference to online feedback in multi-round conversational recommendation," in *Proc. 14th ACM Int. Conf. Web Search Data Mining*, Mar. 2021, pp. 364–372.

[34] W. Lei et al., "Estimation-action-reflection: Towards deep interaction between conversational and recommender systems," in *Proc. 13th ACM WSDM*, 2020, pp. 304–312.

[35] W. Lei et al., "Interactive path reasoning on graph for conversational recommendation," in *Proc. 26th ACM SIGKDD*, 2020, pp. 2073–2083.

[36] K. Zhou, W. X. Zhao, S. Bian, Y. Zhou, J.-R. Wen, and J. Yu, "Improving conversational recommender systems via knowledge graph based semantic fusion," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 1006–1014.

[37] R. Sarkar, K. Goswami, M. Arcan, and J. P. McCrae, "Suggest me a movie for tonight: Leveraging knowledge graphs for conversational recommendation," in *Proc. 28th Int. Conf. Comput. Linguistics*, Barcelona, Spain, 2020, pp. 4179–4189.

[38] T. Zhang, Y. Liu, P. Zhong, C. Zhang, H. Wang, and C. Miao, "KECRS: Towards knowledge-enriched conversational recommendation system," 2021, *arXiv:2105.08261*.

[39] D. Ienco and R. G. Pensa, "Enhancing graph-based semisupervised learning via knowledge-aware data embedding," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 5014–5020, Nov. 2020.

[40] F. Galetzka, J. Rose, D. Schlangen, and J. Lehmann, "Space efficient context encoding for non-task-oriented dialogue generation with graph attention transformer," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics 11th Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 7028–7041.

[41] J. Wang et al., "Dual dynamic memory network for end-to-end multi-turn task-oriented dialog systems," in *Proc. 28th COLING*, Barcelona, Spain, Dec. 2020, pp. 4100–4110.

[42] A. Galassi, M. Lippi, and P. Torroni, "Attention in natural language processing," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 10, pp. 4291–4308, Oct. 2021.

[43] C. Zheng and M. Huang, "Exploring prompt-based few-shot learning for grounded dialog generation," 2021, *arXiv:2109.06513*.

[44] A. Madotto, Z. Lin, G. I. Winata, and P. Fung, "Few-shot bot: Prompt-based learning for dialogue systems," 2021, *arXiv:2110.08118*.

[45] Z. Liu, H. Wang, Z.-Y. Niu, H. Wu, and W. Che, "DuRecDial 2.0: A bilingual parallel corpus for conversational recommendation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2021, pp. 4335–4347.

[46] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. Rush, "OpenNMT: Open-source toolkit for neural machine translation," in *Proc. ACL, Syst. Demonstrations*, Vancouver, BC, Canada, 2017, pp. 67–72.

[47] Y. Wang et al., "A large-scale Chinese short-text conversation dataset," in *Proc. CCF NLPCC*. Cham, Switzerland: Springer, 2020, pp. 91–103.

[48] M. Lewis et al., "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proc. 58th ACL*, 2020, pp. 7871–7880.

[49] A. Radford et al., "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, p. 9, 2019.

[50] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. EMNLP*, Doha, Qatar, Oct. 2014, pp. 1746–1751.

[51] T. Wolf et al., "Transformers: State-of-the-art natural language processing," in *Proc. EMNLP, Sys. Demo*, Oct. 2020, pp. 38–45.

[52] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, Philadelphia, PA, USA, Jul. 2002, pp. 311–318.

[53] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan, "A diversity-promoting objective function for neural conversation models," in *Proc. NAACL-HLT*, San Diego, CA, USA, Jun. 2016, pp. 110–119.

[54] Y. Zhou, Y. Tsvetkov, A. W. Black, and Z. Yu, "Augmenting non-collaborative dialog systems with explicit semantic and strategic dialog history," in *Proc. ICLR*, 2020, pp. 1–15.

[55] J. L. Fleiss, "Measuring nominal scale agreement among many raters," *Psychol. Bull.*, vol. 76, no. 5, p. 378, 1971.

[56] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[57] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, Jul. 1948.

[58] Q. Liu, L. Yu, L. Rimell, and P. Blunsom, "Pretraining the noisy channel model for task-oriented dialogue," *Trans. Assoc. Comput. Linguistics*, vol. 9, pp. 657–674, Jul. 2021.

Jian Wang received the B.S. and M.S. degrees in computer science and engineering from the South China University of Technology, Guangzhou, China, in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong.

His main research interests include dialog systems and natural language generation.

Dongding Lin received the B.S. and M.S. degrees from the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China, in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong.

His research interests include dialog systems, multimodal learning, and natural language generation.

Wenjie Li received the Ph.D. degree in systems engineering and engineering management from the Chinese University of Hong Kong, Hong Kong, in 1997.

She is currently a Professor with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong. Her main research interests include natural language understanding and generation, machine conversation, and summarization and question answering.