Hierarchical Reinforcement Learning With Guidance for Multi-Domain Dialogue Policy

Mahdin Rohmatillah, *Graduate Student Member, IEEE*, and Jen-Tzung Chien, *Senior Member, IEEE*

Abstract—**Achieving high performance in a multi-domain dialogue system with low computation is undoubtedly challenging. Previous works applying an end-to-end approach have been very successful. However, the computational cost remains a major issue since the large-sized language model using GPT-2 is required. Meanwhile, the optimization for individual components in the dialogue system has not shown promising result, especially for the component of dialogue management due to the complexity of multi-domain state and action representation. To cope with these issues, this article presents an efficient guidance learning where the imitation learning and the hierarchical reinforcement learning (HRL) with human-in-the-loop are performed to achieve high performance via an inexpensive dialogue agent. The behavior cloning with auxiliary tasks is exploited to identify the important features in latent representation. In particular, the proposed HRL is designed to treat each goal of a dialogue with the corresponding sub-policy so as to provide efficient dialogue policy learning by utilizing the guidance from human through action pruning and action evaluation, as well as the reward obtained from the interaction with the simulated user in the environment. Experimental results on ConvLab-2 framework show that the proposed method achieves state-of-the-art performance in dialogue policy optimization and outperforms the GPT-2 based solutions in end-to-end system evaluation.**

Index Terms—**Dialogue system, policy optimization, guidance learning, hierarchical reinforcement learning.**

I. Introduction

As THE dialogue system has become widespread due to its potential applications in real-world scenarios, designing a high-performance task-oriented dialogue system with low computation cost remains a crucial issue that must be addressed, especially in a multi-domain dialogue with a large number of possible combinations of user intents, semantic slots and values that must be correctly satisfied by the system. The common approaches to dialogue system based on the end-to-end method [1] and the pipeline method [2], [3] have been proposed with various strengths and shortcomings. Using end-to-end method, the dialogue task is formulated as a generative model where all dialogue system components including natural language understanding [4], dialogue state tracking, dialogue management and natural language generation are jointly optimized to provide appropriate response to the user. The recent end-to-end strategies built based on the generative pre-trained transformer 2 (GPT-2) [5] have achieved state-of-the-art (SOTA) results both in the automatic evaluation based on interaction with the bot as well as in the human evaluation which has been done by using the Amazon mechanical turk [6]. Unfortunately, such an approach suffers from two problems. First, the required computation cost is huge due to the usage of large-scaled language model via GPT-2. Second, several pre- and post-processing stages must be conducted as GPT-2 is not specially designed for solving dialogue task. On the other hand, the modular approach which optimizes individual components in the dialogue system offers simpler training and lower computation. However, the performance in [7], [8] only showed sub-optimal results.

Finding an ideal dialogue policy, which determines system response to the user, is extremely difficult in component-level optimization. Recent attempts reveal that the majority of dialogue policies have been formulated as a reinforcement learning (RL) task [9], [10], [11]. Unfortunately, the high dimensional state and action spaces may contain hundreds of entries which easily confuse the dialogue policy in determining appropriate action. The problem is aggravated by the fact that the exploration in this task is very limited, unlike in a common RL task like robotic control and Atari games where the agent needs to explore the environment more frequently to find more possible solutions. In the multi-domain dialogue task, making too much exploration during training may harm the performance and lead to out-of-domain response. As a consequence of the aforementioned constrains, the results revealed that good performance could only be obtained in the component-wise single-turn evaluation rather in the end-to-end system evaluation using the whole turns of each dialogue session [7] which indicates that the trained dialogue policy is not yet suitable for implementation in a real-world setting.

This paper presents a guidance learning to handle three dialogue strategies where two learning stages are performed. The first stage is the imitation learning which implements the behavior cloning (BC) with auxiliary tasks (denoted by BCAux) to improve the generalization to unseen data which are likely observed once the BC agent is applied into the real environment to interact with the user. The generalization is improved by utilizing the learned features obtained from an auxiliary network. The

Manuscript received 13 March 2022; revised 3 December 2022; accepted 31 December 2022. Date of publication 9 January 2023; date of current version 13 January 2023. This work was supported by the National Science and Technology Council, Taiwan, under Contract NSTC 111-2634-F-A49-012. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Isabel Barbancho. *(Corresponding author: Jen-Tzung Chien.)*

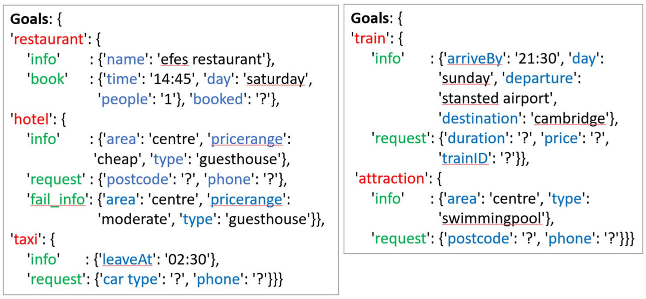
Mahdin Rohmatillah is with the EECS International Graduate Program, National Yang Ming Chiao Tung University, Hsinchu 30010, Taiwan (e-mail: mahdin.ee08@nycu.edu.tw).

Jen-Tzung Chien is with the Institute of Electrical and Computer Engineering, National Yang Ming Chiao Tung University, Hsinchu 30010, Taiwan (e-mail: jtchien@nycu.edu.tw).

Digital Object Identifier 10.1109/TASLP.2023.3235202

auxiliary tasks consist of predicting current belief state and user action which are selected due to their importance in determining the system action in every dialogue turn. Furthermore, the recent works [12], [13], [14] show that the introduction of auxiliary tasks can reduce the causal confusion phenomenon which cause the inability of agent to understand the true cause of expert action from a given dataset of state-action pairs. The strategy of finding reliable pre-trained weights for the subsequent stage is implemented. The second stage is built based on two strategies where the first is the hierarchical RL (HRL) and the second is the human-in-the-loop (HITL). The hierarchical strategy is implemented to simplify the multi-domain task by treating each base domain using its corresponding sub-policy in the low-level policy. The base domain is defined as the domain that firstly occurs in a dialogue session, which indicates the domain with the highest priority. The low-level policy is trained by using proximal policy optimization [15] through interaction with the simulated user in a given environment. The weights of policy network obtained from BCAux are set as the pre-trained weights for low-level policy. Meanwhile, the high-level policy in the hierarchical structure generates a latent vector which is used to activate a sub-policy in a given dialogue session. The hierarchical policies are trained by using the policy gradient method. Due to the fact that the dialogue agent is easily being trapped in the confounded states, HITL learning paradigm is utilized to provide the guidance to the agent by providing the action correction for the confounded state. Furthermore, the action evaluation in each dialogue turn is proposed to handle the dialogue policy which may result in sub-optimal actions in some states. An efficient guidance learning from human and environment is developed to fulfill the dialogue policy optimization with the performance close to the rule-policy which serves as a human in this work. Even with low human supervision, the hierarchical scheme still enhances the learning efficiency when compared to baseline systems with the maximal human guidance. The proposed work could achieve competitive result with low-computation cost.

The rest of this paper is organized as follows. In Section II, the multi-domain task-oriented dialogue system and dialogue management are surveyed. Section III presents the efficient guidance learning for dialogue management with the optimization process which includes imitation learning as well as hierarchical RL. Section IV addresses the experimental settings for evaluation of dialogue management. The experimental results to illustrate the learning efficiency of the proposed method relative to the recent methods are addressed. The summary of findings from this study is provided in Section V.



```
Goals: {
'restaurant': {
   'info'      : {'name': 'efes restaurant'},
   'book'      : {'time': '14:45', 'day': 'saturday',
                  'people': '1'}, 'booked': '?'},
'hotel': {
   'info'      : {'area': 'centre', 'pricerange':
                  'cheap', 'type': 'guesthouse'},
   'request' : {'postcode': '?', 'phone': '?'},
   'fail_info': {'area': 'centre', 'pricerange':
                  'moderate', 'type': 'guesthouse'}},
'taxi': {
   'info'     : {'leaveAt': '02:30'},
   'request': {'car type': '?', 'phone': '?'}}}
```

```
Goals: {
'train': {
   'info'      : {'arriveBy': '21:30', 'day':
                  'sunday', 'departure':
                  'stansted airport',
                  'destination': 'cambridge'},
   'request': {'duration': '?', 'price': '?',
                  'trainID': '?'}},
 'attraction': {
   'info'      : {'area': 'centre', 'type':
                  'swimmingpool'},
   'request': {'postcode': '?', 'phone': '?'}}}
```

Fig. 1. Example of user goals in a dialogue session in MultiWOZ 2.1 dataset which consist of three domains (left) and two domains (right) shown by red color. The sequence indicates the order of domain occurrence. The green and blue colored text are user intent and semantic slot, respectively with the corresponding value in black colored text.

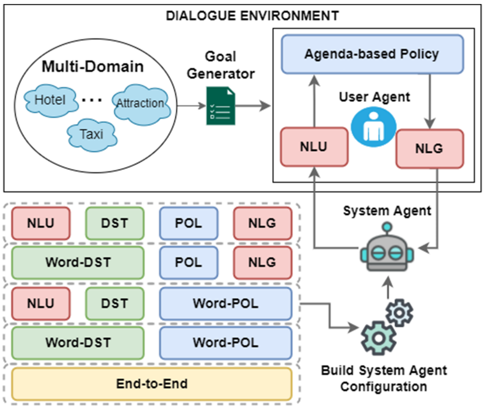


Fig. 2. Interaction between system agent and simulated user via an agenda-based policy in a multi-domain dialogue management using ConvLab-2 framework which can be built with different configurations.

II. Multi-Domain Task-Oriented Dialogue

The recent approaches to build dialogue system and handle dialogue management are surveyed and discussed.

A. Multi-Domain Dialogue System

Many academics have been devoted to work on the realistic scenario based on multi-domain task-oriented dialogue system. However, providing an appropriate solution to address this task is very challenging. Different from the previous dialogue tasks considering only limited domains like movie ticket booking [16], flight booking [17] and restaurant reservation [18], [19], multi-domain dialogue offers the involvement of various domains in a single user goal as shown by Fig. 1. As a result, the dialogue structure becomes complicated due to the increase of possible scenarios. In order to obtain a desirable performance, dialogue system needs to satisfy all of the user intentions in each domain concerning in the current goal in a limited number of time steps. Among various frameworks designed for multi-domain dialogue task, [20], [21], ConvLab-2 [22] is the most popular framework that is mainly designed for handling the MultiWOZ 2.1 dataset [21]. ConvLab-2 provides flexible structures of dialogue system for supporting various ways of optimization as illustrated in Fig. 2. Therefore, researchers are allowed to build their own dialogue system in a pipeline fashion that requires optimization of individual components including natural language understanding (NLU), dialogue state tracking (DST), dialogue policy (POL) and natural language generation (NLG) [23] or in an end-to-end manner that optimizes the overall components jointly. It is also possible to investigate the joint optimization that incorporates some pipeline system components such as word-level optimization scenarios like word-DST and word-POL that jointly optimize NLU-DST and policy-NLG, respectively. Another important benefit of ConvLab-2 framework is the end-to-end system evaluation which faithfully reflects the human evaluation in real-world application. In this evaluation, a system-wise evaluation is performed instead of component-wise evaluation which merely examines specific component of dialogue pipeline system by using a single-turn evaluation assuming that the model is provided with the ground truth from the other components or from the previous dialogue turn. Metrics of task success and inform rate are measured by using the current user utterance, dialogue state, and database query as has been done in previous works [24]. On the other hand, the system-wise evaluation considers all components in the dialogue pipeline system via an end-to-end system evaluation along with the multi-turn conversation done by utilizing the simulated user, which represents human as a user. All of the assessments in this study take a system-wise or end-to-end approach, which closely resembles a real-world scenario.

B. Multi-Domain Dialogue Policy

Many studies have been devoted to develop multi-domain dialogue policy, which is regarded as a critical component in dialogue systems. Based on the current benchmark result, both of word-level [2], [3], [25] and end-to-end optimization strategies [1], [26] resulted in sub-optimal performance in the end-to-end system evaluation although good performance was achieved in the component-wise evaluation. Accordingly, many attempts have been designed to improve dialogue policy by using reinforcement learning (RL) [27] as shown in the two most popular dialogue framework benchmarks, ConvLab-2 [22] and PyDial [20], [28]. To build an RL agent, the first important step is to train the dialogue policy using the behaviour cloning (BC) which is seen as a type of imitation learning by utilizing the state and action pairs from a dialogue dataset. Those pairs are commonly formed by using the pre-defined vectorized functions that convert the sentences in dialogue dataset to the vectors that are suitable for RL training. In case of MultiWOZ 2.1 dataset, the vectorization process yields a state vector with size of 340 consisting of six different partitions which are user action, system action, belief state information, booking information, database pointer and state termination. In addition, the action represented as a vectorized version of dialogue act with a dimension of 209, which consists of four information sources including domain, action type, slot and value. By using the state-action pairs $\mathcal{D} = \{\mathbf{s}_n, \mathbf{y}_n\}_{n=1}^{N}$ where $\mathbf{s}_n$ and $\mathbf{y}_n$ denote the state and the target corresponding to an expert action, respectively, the policy network $\pi_\theta(\cdot)$ for finding action $\mathbf{a}$ given by the optimal parameter θ^* is estimated by maximizing the log likelihood or minimizing the mean squared error or cross entropy error $\mathcal{L}_{\text{BC}}(\cdot)$ for regression or classification, respectively, from the training data $\mathcal{D}$ via

$$\theta^* = \underset{\theta}{\operatorname{argmin}}\, \mathbb{E}_{(\mathbf{s},\mathbf{y})\sim\mathcal{D}} \left[\mathcal{L}_{\text{BC}}\left(\pi_\theta(\mathbf{s}), \mathbf{y}\right)\right]. \quad (1)$$

Because the dataset $\mathcal{D}$ only contains successful trajectories, the RL agent that uses BC weights is prone to produce failed trajectories owing to the unobserved trajectories if the agent takes an incorrect action in the environment. Therefore, while BC is a simple strategy, achieving acceptable outcomes in real-world application is still very difficult.

Some sophisticated approaches have been proposed, such as training the agent by incorporating the learned reward function based on the adversarial inverse reinforcement learning [29] by using expert trajectories [30], [31], [32]. The training process was done similar to that of generative adversarial networks [33]. Another approach was developed by a model-based RL [34], [35] where the model was trained to replicate the user behavior so that the agent might progress through the planning phase with sample efficiency. Unfortunately, such an approach either only worked in a somewhat simple setting or only performed well in component-wise evaluations that solely looked at single turns. As a result, when the learned agents were evaluated in an end-to-end system evaluation via multi-turn dialogues, the desired results were not achieved.

Due to the success of transformer [36] in natural language processing tasks, many attempts have been proposed to apply it for multi-domain dialogue system [37], [38] in an end-to-end optimization manner. Recent work has also introduced offline RL optimization to improve the performance of the transformer-based system [39]. However, the significant results have been reported in the latest DSTC9 track 2 challenge in which end-to-end optimization based on the GPT-2 model [40], [41] has successfully achieved SOTA performance. That was the first time that end-to-end approach outperformed the component-level optimization. However, because a large-scaled language model (LM) using GPT-2 was used as the default component, the end-to-end method required a high computation cost. Furthermore, the enhanced data from other datasets as well as the extensive pre- and post-processing were needed to make LM operate in a task-oriented conversation system. In this paper, a new method is proposed by preserving low-cost computation while significantly improving multi-domain dialogue policy. The proposed strategy is designed by addressing the shortcomings of previous policies by means of hierarchical RL which can simplify the problem formulation in multi-domain dialogue system. This strategy is efficient and does not require data enhancement during the pre- and post-processing stages.

III. Hierarchical RL With Guidance

Overview of the proposed method is depicted in Fig. 3. The first step involves data pre-processing to obtain the inputs and the corresponding labels for training neural networks for BCAux. Next, the weights of policy network in BCAux is used as the pre-trained weights for low-level policy in the hierarchical RL (HRL). HRL is trained according to the hierarchical policy gradient with the help of human to provide additional guidance during training.

A. Imitation Learning With Auxiliary Tasks

Typically, introducing the auxiliary objectives in construction of a target model is promising to regularize the model when dealing with unseen data [12], [42], [43]. The main purpose of auxiliary tasks is to share the learned representations from the auxiliary objectives to the primary model to boost primary

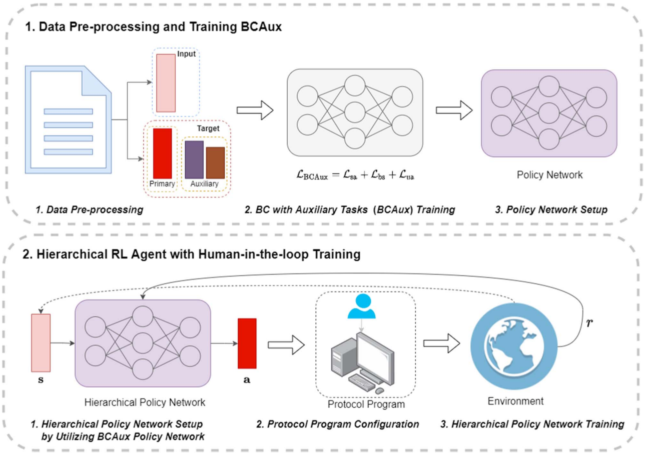

1. Data Pre-processing and Training BCAux
Input
Target
Primary
Auxiliary
$\mathcal{L}_{\text{BCAux}} = \mathcal{L}_{\text{sa}} + \mathcal{L}_{\text{bs}} + \mathcal{L}_{\text{ua}}$
Policy Network
1. Data Pre-processing
2. BC with Auxiliary Tasks (BCAux) Training
3. Policy Network Setup
2. Hierarchical RL Agent with Human-in-the-loop Training
s
Hierarchical Policy Network
a
Protocol Program
Environment
r
1. Hierarchical Policy Network Setup by Utilizing BCAux Policy Network
2. Protocol Program Configuration
3. Hierarchical Policy Network Training

Fig. 3. Overview of the proposed neural policy optimization for training a BCAux and a hierarchical RL agent.

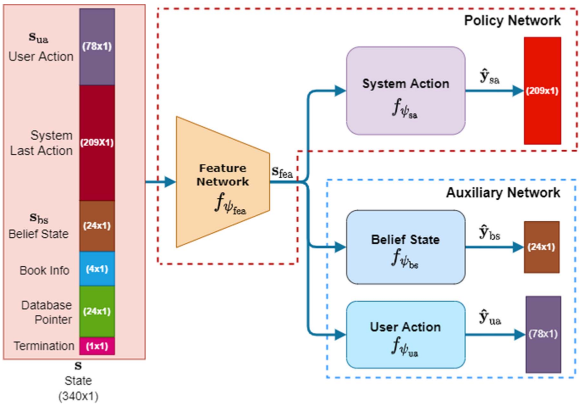


Fig. 4. Architecture of the proposed BCAux consisting of the policy network for handling primary task and predicting system action $\widehat{\mathbf{y}}_{\text{sa}}$ and the auxiliary networks for predicting belief state $\widehat{\mathbf{y}}_{\text{bs}}$ and user action $\widehat{\mathbf{y}}_{\text{ua}}$.

task performance. In case of dialogue policy optimization, the auxiliary tasks help the primary model to understand the true cause of an expert action given a certain state. This study presents the behavior cloning with auxiliary tasks as a specialized imitation learning to provide reliable pre-trained weights for low-level policy in subsequent HRL optimization. At the beginning, the input states and the corresponding targets, both for primary and auxiliary tasks must be formed from dialogue dataset $\mathcal{D} = \{\mathbf{s}_n, \mathbf{y}_n\}_{n=1}^{N}$. The input state and the primary target for system action are denoted by $\mathbf{s} \in \mathbb{R}^{340}$ and $\mathbf{y}_{\text{sa}} \in \mathbb{R}^{209}$, respectively. The auxiliary tasks consist of predicting the targets of belief state $\mathbf{y}_{\text{bs}} \in \mathbb{R}^{24}$ and user action $\mathbf{y}_{\text{ua}} \in \mathbb{R}^{78}$ which are selected due to their importance in determining appropriate system response. Since all tasks are seen as the multi-label binary classification, the binary cross-entropy losses are modified by considering the balanced parameters $\{\beta_{\text{sa}}, \beta_{\text{bs}}, \beta_{\text{ua}}\}$ due to the class imbalance between positive and negative samples. In addition to one-hot target vector $\mathbf{y}_n$, we calculate the ratios of the numbers of negative samples over all samples corresponding to the classification labels for system actions, belief states and user actions to determine β_{sa}, β_{bs} and β_{ua}, respectively. Such ratios are popular to handle the class imbalance [44] in multi-class classification which is fitted to the setting in this work.

Next, the architecture of the proposed BCAux is depicted in Fig. 4. The primary and auxiliary networks share a common feature extractor $\mathbf{s}_{\text{fea}} \triangleq f_{\psi_{\text{fea}}}(\mathbf{s})$ with parameter ψ_{fea} that is designed to provide meaningful features for primary task by taking advantage of auxiliary tasks. The loss function of this scheme $\mathcal{L}_{\text{BCAux}} = \mathcal{L}_{\text{sa}} + \mathcal{L}_{\text{bs}} + \mathcal{L}_{\text{ua}}$ is integrated by three losses for classification prediction in a policy network and an auxiliary network where four parameters $\psi = [\psi_{\text{fea}}, \psi_{\text{sa}}, \psi_{\text{bs}}, \psi_{\text{ua}}]$ are included. The first loss is devoted to the primary task which predicts the expert action given an input state. The remaining two losses belong to auxiliary tasks for prediction of belief state and user action. Given the samples of input feature $\mathbf{s}_{\text{fea}}$ and target $\mathbf{y}$ ($\mathbf{y}_{\text{sa}}$, $\mathbf{y}_{\text{bs}}$ or $\mathbf{y}_{\text{ua}}$) from dataset $\mathcal{D}$, and the ratios β (β_{sa}, β_{bs} or β_{ua}), the balanced cross-entropy loss $\mathcal{L}$ ($\mathcal{L}_{\text{sa}}$, $\mathcal{L}_{\text{bs}}$ or $\mathcal{L}_{\text{ua}}$) between true target $\mathbf{y}$ and predicted target $\widehat{\mathbf{y}} \triangleq f_{\psi}(\mathbf{s}_{\text{fea}})$ ($\widehat{\mathbf{y}}_{\text{sa}}$, $\widehat{\mathbf{y}}_{\text{bs}}$ or $\widehat{\mathbf{y}}_{\text{ua}}$) with mapping parameter ψ (ψ_{sa}, ψ_{bs} or ψ_{ua}) is yielded by

$$\begin{aligned}\mathcal{L}(f_{\psi}(\mathbf{s}_{\text{fea}}), \mathbf{y}) = & -\mathbb{E}_{(\widetilde{\mathbf{s}}, \mathbf{y}) \sim \mathcal{D}} \left[\beta \mathbf{y}^{\top} \log f_{\psi}(\mathbf{s}_{\text{fea}})\right. \\ & \left. + (1-\beta)(\mathbf{1}-\mathbf{y})^{\top} \log\left(\mathbf{1} - f_{\psi}(\mathbf{s}_{\text{fea}})\right)\right] \triangleq \mathcal{L} \qquad (2)\end{aligned}$$

which is consistently applied for three auxiliary tasks.

B. Hierarchical RL With Sub-Policies

Instead of dealing with each dialogue session using standard RL, in this work, a hierarchical RL (HRL) based on policy gradient method is proposed to elaborate the response to the user's goals by treating them uniquely based on their base domain. Base domain is defined as the domain that becomes the main concern in a dialogue which always occurs in the beginning of the dialogue. Different from the common HRL that high-level policy chooses an action in every pre-determined period of time or after reaching a specific sub-goal [45], in this work, the high-level policy only outputs an action at the beginning of a dialogue session to activate a sub-policy in the low-level policy that corresponds to the base domain of current dialogue. This scenario is reasonable to build the task-oriented dialogue system where dialogue policy needs to satisfy a user's goal in a very limited time step, ideally less than 15 time steps in average, which are significantly lower than standard HRL tasks like maze or robotic tasks which require hundreds to thousands of time steps to achieve the goal. By implementing HRL in this setting, the complexity of the task which involves huge state and action spaces can be reduced so that the dialogue policy training can be optimized. As hierarchical approach is used, standard PG [46] is calculated over the accumulated reward with the trajectory of states and actions $\tau = \{\mathbf{s}_t, \mathbf{a}_t\}_{t=0}^{T-1}$ drawn by a policy $\pi_{\theta}(\cdot)$ with the hierarchical parameters $\theta = \{\theta_h, \theta_l\}$ in different levels in a form of

$$\begin{aligned}\nabla_{\theta} J(\theta) &= \nabla_{\theta} \mathbb{E}_{\tau \sim \pi_{\theta}(\tau)} \left[R(\tau)\right] \\ &= \mathbb{E}_{\tau \sim \pi_{\theta}(\tau)} \left[\nabla_{\theta} \log \pi_{\theta}(\tau) R(\tau)\right] \qquad (3)\end{aligned}$$

where θ_h denotes the high-level policy and θ_l denotes the low-level policy. The reward $R(\tau) = \{R_h(s_0), R_l(\tau)\}$ involves the ones R_h and R_l for high-level and low-level policies using initial state s_0 and remaining trajectory τ, respectively. Given the output of high-level policy $\mathbf{z}$ from $\{\mathbf{z}_k\}_{k=1}^{K}$, the trajectory distribution of an agent with K sub-policies is yielded and expanded over a trajectory of T steps by

$$\pi_\theta(\tau) = p(\mathbf{s}_0)\prod_{k=1}^{K}\left[\pi_{\theta_h}(\mathbf{z}_k|\mathbf{s}_0)\prod_{t=0}^{T-1}\left(\pi_{\theta_l}(\mathbf{a}_t|\mathbf{s}_t,\mathbf{z}_k)\right.\right.$$
$$\left.\left.\times p(\mathbf{s}_{t+1}|\mathbf{s}_t,\mathbf{a}_t)\right)\right]. \tag{4}$$

Considering (3) and (4), the hierarchical policy gradient is accordingly calculated by

$$\nabla_\theta J_{\text{HRL}}(\theta) = \mathbb{E}_{\tau\sim\pi_\theta(\tau)}\left[\sum_{k=1}^{K}\nabla_\theta \log\left(\pi_{\theta_h}(\mathbf{z}_k|\mathbf{s}_0)\right.\right.$$
$$\left.\left.\times\prod_{t=0}^{T-1}\pi_{\theta_l}(\mathbf{a}_t|\mathbf{s}_t,\mathbf{z}_k)\right)R(\tau)\right] \tag{5}$$

where the terms independent of θ are disregarded and the hierarchical setting of $\{\mathbf{z}_k\}_{k=1}^{K}$ gives the gradient

$$\nabla_\theta \log \pi_\theta(\tau) = \frac{\nabla_\theta\left(\prod_{k=1}^{K}\pi_{\theta_h}(\mathbf{z}_k|\mathbf{s}_0)\prod_{t=0}^{T-1}\pi_{\theta_l}(\mathbf{a}_t|\mathbf{s}_t,\mathbf{z}_k)\right)}{\prod_{k=1}^{K}\pi_{\theta_h}(\mathbf{z}_k|\mathbf{s}_0)\prod_{t=0}^{T-1}\pi_{\theta_l}(\mathbf{a}_t|\mathbf{s}_t,\mathbf{z}_k)}. \tag{6}$$

Unfortunately, such a gradient is prone to be unstable during training if high-level policy outputs a wrong $\mathbf{z}_k$, i.e. the optimal output $\mathbf{z}^*$ is missed as $\mathbf{z}_k \neq \mathbf{z}^*$, which means high-level policy assigns an incorrect sub-policy to deal with user specific goal. Suppose $0 < \pi_{\theta_l}(\mathbf{a}_t|\mathbf{s}_t,\mathbf{z}_k) < \rho$ for a wrong $\mathbf{z}_k$ in trajectory τ is considered. Then, the probability of non-optimal trajectory for each low-level policy is upper bounded by ρ^T. The gradient for each non-optimal low-level policy using the output of high-level policy $\mathbf{z}_k$ can be derived and the computation complexity can be obtained by

$$\nabla_\theta\left(\pi_{\theta_h}(\mathbf{z}_k|\mathbf{s}_0)\prod_{t=0}^{T-1}\pi_{\theta_l}(\mathbf{a}_t|\mathbf{s}_t,\mathbf{z}_k)\right) = \nabla_\theta\pi_{\theta_h}(\mathbf{z}_k|\mathbf{s}_0)$$
$$\times\prod_{t=0}^{T-1}\pi_{\theta_l}(\mathbf{a}_t|\mathbf{s}_t,\mathbf{z}_k) + \sum_{t=0}^{T-1}\pi_{\theta_h}(\mathbf{z}_k|\mathbf{s}_0)\left(\nabla_\theta\pi_{\theta_l}(\mathbf{a}_t|\mathbf{s}_t,\mathbf{z}_k)\right.$$
$$\left.\times\prod_{t'=0,t'\neq t}^{T-1}\pi_{\theta_l}(\mathbf{a}_{t'}|\mathbf{s}_{t'},\mathbf{z}_k)\right) = \mathcal{O}\left(T\rho^{T-1}\right). \tag{7}$$

By merging (7) in (6), the gradient is then updated by considering the calculation corresponding to the optimal latent variable $\mathbf{z}^*$ as well as the other $K-1$ non-optimal $\mathbf{z}_k$

$$\nabla_\theta \log \pi_\theta(\tau) = \frac{\nabla_\theta\left(\pi_{\theta_h}(\mathbf{z}^*|\mathbf{s}_0)\prod_{t=0}^{T-1}\pi_{\theta_l}(\mathbf{a}_t|\mathbf{s}_t,\mathbf{z}^*)\right)}{\pi_{\theta_h}(\mathbf{z}^*|\mathbf{s}_0)\prod_{t=0}^{T-1}\pi_{\theta_l}(\mathbf{a}_t|\mathbf{s}_t,\mathbf{z}^*)}$$
$$+ (K-1)\mathcal{O}\left(T\rho^{T-1}\right) \tag{8}$$

where the computation $(K-1)\mathcal{O}\left(T\rho^{T-1}\right)$ is the source of instability that must be carefully tackled in learning process.

For the reward setting in HRL, $R_l(\tau) = \{r_l(\mathbf{s}_t,\mathbf{z})\}$ is the reward from environment, and $r_h(\mathbf{s}_0,\mathbf{z}^*) = 1$ and $r_h(\mathbf{s}_0,\mathbf{z}_k) = 0$ at initial state $\mathbf{s}_0$ are defined. As a result, the gradient of low-level policy with non-optimal $\mathbf{z}_k$ can be eliminated by disregarding any trajectory τ with $r_h(\mathbf{s}_0,\mathbf{z}_k) = 0$ stored in high-level replay buffer $\mathcal{D}_h$. In the implementation, high-level policy $\pi_{\theta_h}(\cdot)$ is trained by the policy gradient (PG) and low-level policy is trained by the proximal policy optimization (PPO) [15] with the clipped surrogate objective given by

$$L_{\text{clip}}(\theta_l) = \mathbb{E}_{(\mathbf{s},\mathbf{a},\mathbf{z})\sim\mathcal{D}_l}\left[L\left(\mathbf{s},\mathbf{a},\mathbf{z},\theta_l^{\text{old}},\theta_l\right)\right] \tag{9}$$

using individual buffer $\mathcal{D}_l$ and previous parameter θ_l^{old} in

$$L\left(\mathbf{s},\mathbf{a},\mathbf{z},\theta_l^{\text{old}},\theta_l\right) = \min\left(r(\theta_l)A_{\theta_l^{\text{old}}}(\mathbf{s},\mathbf{a},\mathbf{z}),\right.$$
$$\left.\text{clip}(r(\theta_l),1-\epsilon,1+\epsilon)A_{\theta_l^{\text{old}}}(\mathbf{s},\mathbf{a},\mathbf{z})\right). \tag{10}$$

Here, the ratio $r(\theta_l) = \frac{\pi_{\theta_l}(\mathbf{a}|\mathbf{s},\mathbf{z})}{\pi_{\theta_l^{\text{old}}}(\mathbf{a}|\mathbf{s},\mathbf{z})}$ between current policy π_{θ_l} and old policy $\pi_{\theta_l^{\text{old}}}$ in low-level policy is calculated with the output of high-level policy $\mathbf{z}$, and the advantage function using current policy $\pi_{\theta_l^{\text{old}}}$ is estimated by [47]

$$A_{\theta_l^{\text{old}}}(\mathbf{s}_t,\mathbf{a}_t,\mathbf{z}) = \delta_V(\mathbf{s}_t,\mathbf{z}) + \gamma\lambda A_{\theta_l^{\text{old}}}(\mathbf{s}_{t+1},\mathbf{a}_{t+1},\mathbf{z}) \tag{11}$$

where $\delta_V(\mathbf{s}_t,\mathbf{z}) = r_l(\mathbf{s}_t,\mathbf{z}) + \gamma V_{\phi^{\text{old}}}(\mathbf{s}_{t+1},\mathbf{z}) - V_{\phi^{\text{old}}}(\mathbf{s}_t,\mathbf{z})$ is seen as the temporal difference (TD) error [48] of value function V with the updated and the current critic parameters ϕ and ϕ^- in a learning epoch, respectively. γ is the discount factor, and λ is a factor to adjust the bias-variance dilemma in model construction. The learning objective is set by choosing either the weighted advantage function $r(\theta_l)A_{\theta_l^{\text{old}}}(\mathbf{s},\mathbf{a},\mathbf{z})$ or the function $\text{clip}(r(\theta_l),1-\epsilon,1+\epsilon)A_{\theta_l^{\text{old}}}(\mathbf{s},\mathbf{a},\mathbf{z})$ with a clipping threshold ϵ. This clipped surrogate function $L_{\text{clip}}(\theta_l)$ is *maximized* to estimate the policy parameter θ_l.

In addition, the PPO critic parameter ϕ is updated by minimizing the regression error between the predicted value function $V_\phi(\mathbf{s}_t,\mathbf{z})$ and the target value function $y_t = r_l(\mathbf{s}_t,\mathbf{z}) + \gamma V_{\phi^{\text{old}}}(\mathbf{s}_{t+1},\mathbf{z})$ where the HRL state $(\mathbf{s}_{t+1},\mathbf{z})$ and the reward $r_l(\mathbf{s}_t,\mathbf{z})$ are sampled from the low-level replay buffer $\mathcal{D}_l$. Therefore, the regression loss of PPO critic network is yielded as a TD error of value function

$$\mathcal{L}_{\text{V}}(\phi) = \mathbb{E}_{(\mathbf{s},\mathbf{z},r_l)\sim\mathcal{D}_l}\left[\left(y - V_\phi\left(\mathbf{s},\mathbf{z}\right)\right)^2\right]. \tag{12}$$

In order to boost the training of low-level policy θ_l, the optimal weights of policy network in BCAux $\{\psi_{\text{fea}}^*,\psi_{\text{sa}}^*\}$, as seen in Fig. 4, are used as the pre-trained weights for θ_l. From the empirical investigation, the best BC agent could not be determined by its validation loss during training. Instead, the model selection could be performed according to the task success rate in the policy evaluation stage which accordingly reflects real implementation performance.

C. Guidance Learning With Human-in-The-Loop

Human-in-the-loop (HITL) is mainly introduced to provide a guidance to the agent during training due to the fact that agent is prone to be trapped in the confounded states. Instead of using

TABLE I
THREE EXAMPLES OF CONFOUNDED STATES OR REPEATED GOAL IDS WHICH TRAP THE AGENT FOR BEING STUCK. SYSTEM SENTENCES REFLECT THE CORRESPONDING ACTIONS. THE REPETITION OF DIALOG ACTS FROM BOTH USER AND SYSTEM SENTENCES CAN BE A CLEAR INDICATOR FOR AGENT TO ASK HUMAN CORRECTION

Goal Id	User Dialog Act	User Sentence	System Sentence
19	[['Request', 'Attraction', 'Type', '?'], ['Request', 'Attraction', 'Fee', '?']]	What type of attraction is this? What are the entrance fees?	Its entrance fee is ? . You are welcome. Is there anything else I can help you with today?
19	[['Request', 'Attraction', 'Type', '?'], ['Request', 'Attraction', 'Fee', '?']]	What is the attraction type? What are the entrance fees?	Its entrance fee is ? . You are welcome. Is there anything else I can help you with today?
19	[['Request', 'Attraction', 'Type', '?'], ['Request', 'Attraction', 'Fee', '?']]	What is the attraction type? What are the entrance fees?	Its entrance fee is ? . Did you need any further assistance today?
85	[['Request', 'Hotel', 'Area', '?']]	What is the area?	Is there anything else I can help you with?
85	[['Request', 'Hotel', 'Area', '?']]	What area is it in?	Is there anything else I can help you with?
85	[['Request', 'Hotel', 'Area', '?']]	Yeah, what area of town is it in?	Did you need any further assistance today?
102	[['Request', 'Attraction', 'Fee', '?']]	What are the entrance fees?	Is there anything else I can help you with today?
102	[['Request', 'Attraction', 'Fee', '?']]	Can you give me the entrance fee?	Is there anything else I can help you with today?
102	[['Request', 'Attraction', 'Fee', '?']]	Can you give me the entrance fee?	Is there anything else I can help you with today?

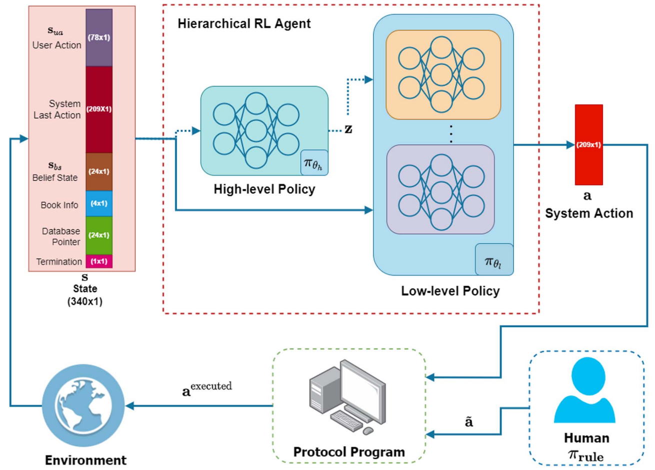


Fig. 5. Hierarchical RL with human-in-the-loop process for dialogue policy improvement. $\widetilde{\mathbf{a}}$ denotes the action from human or π_{rule} as a guidance. Meanwhile, $\mathbf{a}$ denotes the action from π_{θ_l}. $\mathbf{s}$ denotes the state from environment due to $\mathbf{a}^{\text{executed}}$ which is obtained either from $\mathbf{a}$ or $\widetilde{\mathbf{a}}$.

real human, the rule-based agent π_{rule} provided by ConvLab-2 framework, is set to act as a human during the learning process. Rule-based agent is a handcrafted agent designed by human that serves as the upper bound in dialogue policy optimization. The guidance in HITL is employed in the designed protocol program which is then performed in dialogue environment as shown Fig. 5. In general, the guidance or feedback from the human can be done in the form of action pruning, reward shaping or state manipulation. Unfortunately, the last two feedback are hard to be designed in the multi-domain dialogue task due to its complexity and manual tuning requirement. Therefore, in this work, the guidance from human which is governed by a protocol program is employed in two different ways based on the action pruning and evaluation scenario. The first way is to identify a confounded state which reflects three repetitions of state representation consecutively as illustrated in Table I, human must provide a corrective action that is executed into the environment through the protocol program. Next, the protocol program removes any trajectory that involves either confounded state or failed trajectory generated by the agent due to the incorrect action of high-level policy. For the second feedback, human must evaluate low-level policy action in every step by assuming that agent may choose a sub-optimal action in certain time steps. Therefore, instead of only maximizing $L(\mathbf{s}, \mathbf{a}, \mathbf{z}, \theta_l^{\text{old}}, \theta_l)$, the low-level policy in PPO explores the environment by further maximizing the negative cross-entropy for the evaluations of the selected actions between human $\widetilde{\mathbf{a}}$ and system $\mathbf{a}$ stored in low-level replay buffer $\mathcal{D}_l$

$$\begin{aligned} L_{\text{CE}}(\theta_l) &= \mathbb{E}_{(\mathbf{s},\mathbf{a},\widetilde{\mathbf{a}},\mathbf{z})\sim\mathcal{D}_l}\left[\pi_{\text{rule}}(\widetilde{\mathbf{a}}|\mathbf{s})\pi_{\theta_l}(\mathbf{a}|\mathbf{s},\mathbf{z})\right] \\ &= \mathbb{E}_{\mathcal{D}_l}\left[H(\pi_{\text{rule}}(\widetilde{\mathbf{a}}|\mathbf{s}))\right] - D_{\text{KL}}\left(\pi_{\text{rule}}(\widetilde{\mathbf{a}}|\mathbf{s})\|\pi_{\theta_l}(\mathbf{a}|\mathbf{s},\mathbf{z})\right) \end{aligned} \quad (13)$$

which is expressed by the entropy $H(\cdot)$ and the Kullback-Leiblier divergence D_{KL}. By maximizing L_{CE}, the action distribution of low-level policy becomes close to the human action distribution which eventually results in near optimal performance. Maximizing the negative cross-entropy for guidance learning is richer than maximizing the policy entropy in standard RL. It is because simply maximizing policy entropy may not assure the performance of agent since the out-of-domain response is likely produced due to the heterogeneous state and action spaces in multi-domain dialogues. Combining all together, the high-level policy is learned according to the PG $\nabla_{\theta_h} J(\theta_h) = \mathbb{E}_{(\mathbf{s}_0,\mathbf{z})\sim\pi_{\theta_h}}[\log \pi_{\theta_h}(\mathbf{z}|\mathbf{s}_0) r_h(\mathbf{s}_0,\mathbf{z})]$, and the low-level policy is estimated by maximizing the regularized PPO objective $J(\theta_l) = \mathbb{E}_{\mathbf{z}\sim\pi_{\theta_h},(\mathbf{s},\mathbf{a})\sim\pi_{\theta_l^{\text{old}}},\widetilde{\mathbf{a}}\sim\pi_{\text{rule}}}[L_{\text{clip}}(\theta_l) + L_{\text{CE}}(\theta_l)]$. The overall learning procedure of actor-critic for HRL with HITL is shown by Algorithm 1.

IV. EXPERIMENTS

The experiments were done by using ConvLab-2 framework [22] which provided the interaction between simulated user and dialogue agent in an environment based on MultiWOZ 2.1 dataset [21]. MultiWOZ 2.1 was an updated version of MultiWOZ 2.0 [49], known as a multi-domain, multi-intent task-oriented dialog corpus [50] that contained 7 domains which are hotel, attraction, restaurant, train, taxi, police and hospital, 13 user intents, 25 slot types, 10,483 dialog sessions, and 71,544 dialog turns. By using ConvLab-2, the end-to-end system evaluation was performed to reflect real-world scenario convincingly. For reward setting, the dialogue agent received −1 in every conversation it made, +5 if current domain was satisfied, and +40 if the task succeeded.

Algorithm 1: Hierarchical RL With Guidance (HRLG) for Multi-Domain Dialogue Management.

```
Require: Dialog data $\mathcal{D}$ with state-target pairs $(\mathbf{s}, \mathbf{y})$,
         belief states $\mathbf{y}_{\text{bs}}$ and user actions $\mathbf{y}_{\text{ua}}$
update BCAux $\psi$ using $\mathcal{D}$ by minimizing $\mathcal{L}$ in Eq. (2)
initialize high level policy $\theta_h$, low-level policy
  $\theta_l \leftarrow \psi^*$ and critic $\phi$
initialize $\mathbf{s}_0$ from user first utterance
while HRL with human-in-the-loop training do
    $\mathbf{z} \leftarrow \pi_{\theta_h}(\mathbf{s}_0)$
    store experience $(\mathbf{s}_0, \mathbf{z}, r_h)$ in $\mathcal{D}_h$
    for $t = 0, \ldots, T$ do
        $\mathbf{a}_t \leftarrow \pi_{\theta_l}(\mathbf{s}_t, \mathbf{z})$
        execute $\mathbf{a}_t$ and observe $\mathbf{s}_{t+1}$, $r_l(\mathbf{s}_t, \mathbf{z})$
        if being trapped then
            $\widetilde{\mathbf{a}}_t \leftarrow \pi_{\text{rule}}(\mathbf{s}_t)$
            execute $\widetilde{\mathbf{a}}_t$ and observe $\widetilde{\mathbf{s}}_{t+1}$, $\widetilde{r}_l(\mathbf{s}_t, \mathbf{z})$
            store $(\mathbf{s}_t, \widetilde{\mathbf{a}}_t, \mathbf{z}, \widetilde{\mathbf{s}}_{t+1}, \widetilde{r}_l(\mathbf{s}_t, \mathbf{z}))$ in $\mathcal{D}_l$
        else
            store $(\mathbf{s}_t, \mathbf{a}_t, \mathbf{z}, \mathbf{s}_{t+1}, r_l(\mathbf{s}_t, \mathbf{z}))$ in $\mathcal{D}_l$
    for $t = T-1, \ldots, 0$ do
        compute $y_t$ using stored trajectories $\mathcal{D}_l$
        compute $A_{\pi_{\theta_l}}(\mathbf{s}_t, \mathbf{a}_t, \mathbf{z})$ by Eq. (11) using $\mathcal{D}_l$
    compute $J(\theta_h)$, $J(\theta_l)$ $(L_{\text{clip}}(\theta_l), L_{\text{CE}}(\theta_l))$, $\mathcal{L}_{\text{V}}(\phi)$
    update $\theta_h$, $\theta_l$ by maximizing $J(\theta_h)$, $J(\theta_l)$
    update $\phi$ by minimizing $\mathcal{L}_{\text{V}}(\phi)$
```

A. Experimental Settings

In the first stage of optimization which involved BCAux training, MultiWOZ 2.1 dataset was split into the training, validation and test data with 8434, 999 and 1000 dialogues, respectively. The policy network consisting of $f_{\psi_{\text{fea}}}$ and $f_{\psi_{\text{sa}}}$ was a feedforward network with two hidden layers while the auxiliary network containing $f_{\psi_{\text{bs}}}$ and $f_{\psi_{\text{ua}}}$ was a feedforward network with one hidden layer. Activation functions in all hidden layers were ReLU, and the output layer was sigmoid. For the hierarchical RL architecture, π_{θ_h} was formed by feedforward network with two hidden layers with ReLU activation and the softmax output layer. Number of domains K was chosen as 5 instead of 7 because taxi, hospital and police domain was merged together as one base domain due to their limited samples. Interpolation parameters for three terms in $\mathcal{L}_{\text{BCAux}}$ were 1, 0.8 and 0.6. Meanwhile, considering the model size, each sub-policy in low-level policy π_{θ_l} and the critic network V_ϕ were identical to the policy network in BCAux. The only difference was the output activation where critic network used linear activation function. During HRL training, the agent collected roughly 2048 dialogue utterances that were divided into 32 batches for updating the parameters. The actor network was optimized by using Adam with initial learning rate 1e-4. The hyperparameters λ, ϵ and γ were set as 0.95, 0.2 and 0.99, respectively. The critic network was optimized by RMSprop [51] with initial learning rate 5e-5.

The experimental results were investigated by using the end-to-end system evaluation, involving the simulated user where the NLU, DST and NLG in a pipeline system were identical to the ConvLab-2 default settings including BERT [52] based NLU, rule-based DST and template NLG, respectively. The same configuration was also applied into system agent which used hierarchical RL as its dialogue policy. With this configuration, a strong policy should be learned to compensate for the imperfect state representation caused by NLU's inability to provide faultless user conversation acts over the whole dialogue flow. The following six main metrics were set for providing the comparative study.

- *success rate:* judges whether user goals of constraints (e.g. hotel location or hotel price) and requests (e.g hotel phone number) have been satisfied by system
- *F1 score:* judges if all requested information like taxi type or taxi phone number has been informed
- *complete rate:* ratio of the completed user constraints
- *booking rate:* calculates the proportion of the successful dialogues for booking hotel, restaurant or train
- *average turn:* calculates the average number of returns to handle user goals for successful and all dialogues
- *computation time:* measures the computation in seconds required to complete 1000 test dialogues

The proposed method was compared with two types of baseline methods. The first type of baselines was the methods which only optimized the dialogue policy as follows

- *maximum likelihood estimation (MLE):* a standard BC that learns to choose an action given a certain state using the supervised learning method
- *policy gradient (PG):* a standard policy based method in RL where the gradient of objective for cumulative reward is calculated to estimate $\pi_\theta(\cdot)$
- *proximal policy optimization (PPO) [15]:* an actor-critic implementation [53] by maximizing the clipped surrogate objective, (10), to train actor and minimizing the regression error, (12), to train critic
- *guided dialogue policy learning (GDPL) [30]:* a method based on adversarial inverse RL [29] which learns the reward by using the expert trajectories and uses it to train the dialogue policy agent sequentially in the same loop

Another type of baselines conducted the optimization in an end-to-end manner. All components from NLU until NLG were optimized jointly.

- *domain aware multi-decode (DAMD) [1]:* a multi-action data augmentation scheme to produce diverse response by using additional state-action pairs
- *minimalist transfer learning (MinTL) [37]:* a transfer learning framework offering plug-and-play approach for task-oriented dialogue system
- *UBAR [38]:* a task-oriented dialogue system in a dialogue session using distilGPT-2 model [54]. The model was fed not only with the user and response sentences, but also with database search result and belief state from the previous steps.

TABLE II
COMPARISON OF DIFFERENT CONFIGURATIONS IN END-TO-END EVALUATION FOR MULTI-DOMAIN DIALOGUE SYSTEM. THE BEST RESULTS ARE SHOWN IN BOLD. UP ARROW SYMBOL (↑) AND DOWN ARROW SYMBOL (↓) INDICATE THE IMPROVEMENT AND DEGRADATION OF THE PROPOSED DIALOGUE POLICY THAT USES BEHAVIOR CLONING WITH AUXILIARY TASKS (BCAUX) AS THE PRE-TRAINED WEIGHTS COMPARED TO THE POLICY WITH ORIGINAL MODEL IMPLEMENTATION INDICATED BY *, RESPECTIVELY. THE DIALOGUE TEST SET CONTAINS 1, 2 AND 3 DOMAINS

Configuration				Success	F1	Complete	Booking	Average
NLU	**DST**	**Policy**	**NLG**	**Rate (%)**	**Score**	**Rate (%)**	**Rate (%)**	**Turn (All)**
Original model implementation								
BERT	Rule	MLE*	Template	47.0	64.5	50.1	28.3	19.4
BERT	Rule	PG*	Template	44.7	60.6	47.1	29.7	20.1
BERT	Rule	GDPL*	Template	47.2	64.6	50.0	26.8	19.3
BERT	Rule	PPO*	Template	61.2	68.2	64.7	62.4	18.2
Model with BCAux								
BERT	Rule	MLE	Template	50 (↑ 3.0)	70 (↑ 5.5)	54.7 (↑ 4.6)	28.7 (↑ 0.4)	18.3 (↑ 1.1)
BERT	Rule	PG	Template	50.7 (↑ 6.0)	74.5 (↑ 13.9)	57.7(↑ 10.6)	27.8 (↓ 1.9)	18.2 (↑ 1.9)
BERT	Rule	GDPL	Template	50.5 (↑ 3.3)	73.6 (↑ 9.0)	54.4 (↑ 4.4)	23(↓ 3.8)	16.6 (↑ 2.7)
BERT	Rule	PPO	Template	**78** (↑ 16.8)	**81.7** (↑ 13.5)	**82.3** (↑ 17.6)	**86.8** (↑ 24.4)	**15.5** (↑ 2.7)

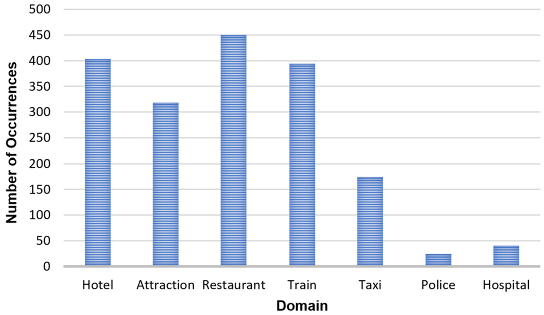


Fig. 6. Domain proportion in an end-to-end system evaluation.

TABLE III
PERFORMANCE COMPARISON BETWEEN ORIGINAL MODEL IMPLEMENTATION INDICATED BY * AND THE MODEL WITH BCAUX FOR THE DIALOGUE TEST SET ONLY CONTAINING 2 AND 3 DOMAINS. DIFFERENT DIALOGUE POLICIES ARE EVALUATED

Dialogue	Success Rate (%)		F1 Score		Complete Rate (%)	
Policy	**2 dom**	**3 dom**	**2 dom**	**3 dom**	**2 dom**	**3 dom**
MLE*	39.2	9.3	62.5	43.8	43.6	10.7
MLE	44.2	14.3	66.3	50.9	50.3	13.6
PG*	38.6	6.4	60.0	38.8	42.8	6.4
PG	46.5	4.3	71.8	52.1	57.5	3.6
GDPL*	40.1	5.0	62.7	43.5	44.7	5.7
GDPL	45.5	10.7	71.3	53.6	51.0	12.9
PPO*	56.8	23.6	67.0	50.3	62.7	26.3
PPO	76.1	44.3	80.3	73.5	83.0	47.1

– three offline RL methods including GPT-critic [39], critic regularized regression (CRR) [55] and decision transformer [56]. All of them were trained by using GPT-2.

– *the first winner of DSTC9 track 2 [40]:* performed five critical processes. The first was the domain adaptation using the pre-trained GPT-2 where the datasets including Schema [57], Camrest [58], Taskmaster 2019, Taskmaster 2020 [59] and MSR-e2e were used. Multi-task fine-tuning using MultiWOZ 2.1, data pre-processing and post-processing, fault tolerance mechanism, and rule-based post-processing for refining the agent utterances were the other four processes.

– *the second winner of DSTC9 track 2 [41]:* conducted similar implementation as the first winner with two distinctions. First, there was no post-processing approach in this work. Second, the auxiliary tasks were employed to increase the consistency in sentence generation given the identical system action responses.

B. End-to-End System Evaluation

A number of experiments were carried out in this work to evaluate the performance of the proposed methods for multi-domain dialogue task over 1000 test dialogues where 337, 523 and 140 dialogues containing 1,2 and 3 domains respectively. Fig. 6 depicts the discourse percentage of individual domains during evaluation. There were seven imbalanced domains where the domain structure was complicated due to high-dimensional semantic slots and values for different acts in high-proportion domains such as hotel and restaurant which resulted in a difficult assignment for the agent. All tests were carried out on a PC with a CPU i9-10900 K, 128 GB of RAM, and a GPU NVIDIA RTX 2080Ti.

Firstly, an evaluation to examine the benefit of auxiliary tasks in BC was done as shown in Table II. Original model results were obtained by using the weights provided by ConvLab-2 framework. The result shows that introducing auxiliary tasks into BC optimization resulted in significant performance improvement. In particular, compared to the original BC, all model that utilizes BCAux weights gained 3% absolute improvement in success rate and more than 4.4% in F1 score and complete rate. Furthermore, for PPO optimization, using BCAux as the pre-trained weights dramatically advanced the performance indicated by a significant performance gap compared to the original PPO implementation. The only drawbacks of using BCAux as the pre-trained weights were reported in case of PG and GDPL, where the booking rate metric shows the decreasing trend. In order to show more advantages of the BCAux, Table III reports the comparison between original model implementation and the model with BCAux in the dialogue test set containing two and three domains. More domains imply the increasing challenge and the degraded performance. The NLU, DST and NLG components were set identically to Table II. It is shown that the models use BCAux weights can handle multi-domain dialogue much better compared to the original model marked by *. More convincing improvement is obtained in the dialogue test with three domains which is a very challenging task.

Fig. 7 depicts the 2-D visualization of the samples of $\{\mathbf{z}_k\}_{k=1}^{5}$ from the output of high-level policy $\pi_{\theta_h}(\mathbf{z}|\mathbf{s}_0)$ of the proposed

TABLE IV
COMPARISON OF DIFFERENT CONFIGURATIONS IN END-TO-END SYSTEM EVALUATION FOR MULTI-DOMAIN DIALOGUE TASK. THE BEST RESULTS EXCEPT THE RULE POLICY ARE SHOWN IN BOLD. THE PROPOSED HIERARCHICAL RL WITH GUIDANCE (HRLG), SHOWN IN THE LAST TWO ROWS, OBTAINS THE COMPETING PERFORMANCE WITH LOW COMPUTATION COST COMPARED TO THE END-TO-END OPTIMIZATION APPROACHES. * MEANS THAT THE MODELS WERE TESTED BY USING THE PROVIDED WEIGHTS. ‡ MEANS THAT THE RESULTS WERE TAKEN FROM [39]

Configuration				Success	F1	Complete	Booking	Average	Comp. Time	No. of Param.
NLU	DST	Policy	NLG	Rate (%)	Score	Rate (%)	Rate (%)	Turn (Succ/All)	(seconds)	(policy model)
BERT	Rule	MLE*	Template	47.0	64.5	50.1	28.3	12.6/19.4	681±2	65K
BERT	Rule	PG*	Template	44.7	60.6	47.1	29.7	12.5/20.1	711±2	65K
BERT	Rule	GDPL*	Template	47.2	64.6	50.0	26.8	**11.9**/19.3	671±4	65K
BERT	Rule	PPO*	Template	61.2	68.2	64.7	62.4	13.0/18.1	646±3	65K
BERT	Rule	Rule	Template	82.7	85.2	92.1	91.2	11.3/11.8	305±2	–
Perfect	Rule	Rule	Template	91.8	89.2	97.4	98.5	11.5/11.7	153±1	–
End-to-End (DAMD)*				34.2	56.9	39.6	52.0	15.6/30.2	3866±50	2M
End-to-End (MinTL)‡				68.1	69.0	71.4	65.4	15.7/20.7	–	≥125M
End-to-End (UBAR)‡				74.3	76.0	79.8	80.8	14.2/18.1	–	≥125M
End-to-End (CRR)‡				72.6	76.0	78.2	82.2	13.6/17.9	–	≥125M
End-to-End (Decision Transformer)‡				75.3	77.0	81.3	83.5	14.8/18.0	–	≥125M
End-to-End (GPT-Critic)‡				77.7	79.0	84.3	85.4	16.3/19.4	–	≥125M
End-to-End (1st winner of DSTC9 track 2)*				91.0	**86.4**	**96.9**	95.8	15.0/15.7	4485±16	163M
End-to-End (2nd winner of DSTC9 track 2)*				60.0	70.2	89.3	86.0	12.7/13.9	3247±2	163M
BERT	Rule	HRLG	Template	82.5	84.0	87.4	89.0	12.5/14.0	517± 2	338K
Perfect	Rule	HRLG	Template	**92.8**	85.0	95.0	**97.1**	12.6/**13.1**	**326±3**	338K

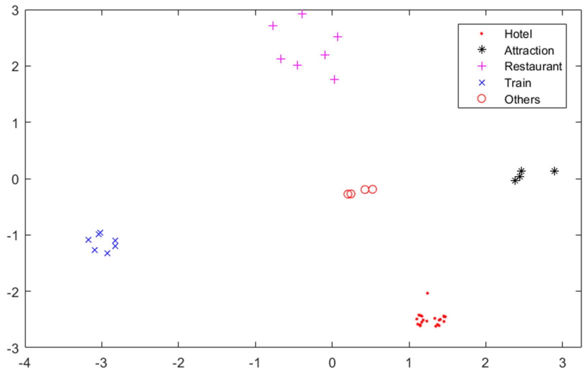


Fig. 7. Visualization of 250 random samples of training dialogues from the outputs of high-level policy $\mathbf{z}$ corresponding to five base domains.

HRLG where t-SNE [60] is used. These latent codes are successfully diverse over five domains. Therefore, the proposed learning strategy may simplify the multi-domain dialogue task to individual base domain task which leads to outperform the other dialogue policy optimization strategies with very significant numbers. Furthermore, due to the task simplification, the feedback from both human and environment by interacting with the simulated user is learned efficiently.

The result of the proposed hierarchical RL with guidance (HRLG) compared to the baseline methods is shown by Table IV. The result shows that all of the dialogue policy optimization methods which are not built based on PPO perform very bad since the corresponding metrics indicated very low score. These methods even obtained the task success and completion rates that are less than 50%. Meanwhile, very low booking rates are revealed with a rate of less than 30%. These empirical results have demonstrated the difficulty of establishing multi-domain task-oriented dialogue policy with good performance due to the huge state and action spaces. On the other hand, the dialogue system configuration with PPO-based policy showed competitive results by exhibiting reasonable performances. Especially, in case of utilizing HRLG, this work attained very convincing result by performing very close to the rule policy which serves as the human to provide a guidance to the agent during training. Moreover, considering all results in dialogue test set, either successful or unsuccessful, the proposed HRLG only required 13.1 turns in average which are 6 turns fewer than the average of those dialogue policy baselines. As a result, HRLG showed faster computation time in completing the entire dialogue test than the dialogue policy baselines. When compared to the majority of the end-to-end optimization methods utilizing GPT-2 model, the suggested learning strategy shows domination in all metrics while taking significantly low computation time to complete the test. Furthermore, when compared to the first winner of DSTC9 track 2 [40], the proposed method shows very competitive results with considerably reduced computation cost about 8 times cheaper with lower average dialogue turns for satisfying user goals. The reduction of computation cost can be explained by comparing the computation time required by each model to finish the test and the average turn conducted by each model during test stage.

There are two main reasons of why end-to-end approaches required high computation cost. The first is the pre- and post-processing steps in every sentence generation turn. The second, since they used one model to represent whole system, then in every turn, they needed to initially predict the user dialogue act and belief state using a large pre-trained model, for example GPT-2 model. Next, the predicted belief state and the dialogue history were fed to the GPT-2 model for generating the response. In other words, the inferences using large model must be done at least twice until producing the system response. This sequential process required very long time to complete. Meanwhile, the large model using BERT was only used once in the pipeline system, that is in the NLU part for predicting the user's dialogue act.

An interesting finding is depicted by the last row of Table IV which shows the performance of the HRLG by assuming it has a perfect NLU. The results show that the performance of HRLG

TABLE V
ABLATION STUDY ON DIFFERENT CONFIGURATIONS FOR DIALOGUE POLICY LEARNING WITH GUIDANCE UNDER DIFFERENT METRICS. THE BEST RESULTS ARE SHOWN IN BOLD

Policy Configuration			**Success**	Inform			**Complete**	**Booking**	**Average**
Hierarchy	**Action Pruning**	L_{CE}	**Rate**	**Precision**	**Recall**	**F1**	**Rate**	**Rate**	**Turn (succ/all)**
No	No	No	78.0	79.0	89.3	81.7	82.3	86.8	12.8/15.5
Yes	No	No	78.5	79.3	90.7	82.6	84.1	86.6	13.3/15.6
No	Yes	No	79.7	80.9	90.4	83.6	85.1	87.2	12.6/14.7
Yes	Yes	No	81.8	80.8	90.8	83.7	86.7	88.3	12.6/14.4
No	Yes	Yes	81.4	**82.0**	**91.4**	**84.6**	86.2	88.4	12.8/14.8
Yes	Yes	Yes	**82.5**	81.3	91.0	84.0	**87.4**	**89.0**	**12.5/14.0**

TABLE VI
ABLATION STUDY UNDER THREE DIALOGUE TEST SCENARIOS INCLUDING THE EVALUATION RESULTS BASED ON (TOP) 337 DIALOGUES CONTAINING ONLY ONE DOMAIN, (MIDDLE) 523 DIALOGUES CONTAINING TWO DOMAINS, AND (BOTTOM) 140 DIALOGUE CONTAINING THREE DOMAINS

Policy Configuration			**Success**	**Inform**			**Complete**	**Booking**	**Average**
Hierarchy	**Action Pruning**	L_{CE}	**Rate**	**Precision**	**Recall**	**F1**	**Rate**	**Rate**	**Turn (succ/all)**
No	No	No	94.7	84.8	96.4	88.6	95.0	94.4	7.5/7.9
Yes	No	No	95.8	84.5	97.2	88.6	96.4	97.0	8.0/8.2
No	Yes	No	94.6	85.0	96.5	88.7	96.7	99.1	7.4/7.8
Yes	Yes	No	94.6	85.0	96.0	88.6	96.1	99.1	6.9/7.3
No	Yes	Yes	94.1	84.5	96.0	88.2	95.5	97.0	7.4/8.1
Yes	Yes	Yes	94.4	85.8	96.2	89.2	96.1	99.1	7.1/7.5

Policy Configuration			**Success**	**Inform**			**Complete**	**Booking**	**Average**
Hierarchy	**Action Pruning**	L_{CE}	**Rate**	**Precision**	**Recall**	**F1**	**Rate**	**Rate**	**Turn (succ/all)**
No	No	No	75.1	77.2	89.4	81.0	82.6	86.0	15.3/16.6
Yes	No	No	74.1	76.7	89.8	81.0	82.6	84.6	15.7/17.1
No	Yes	No	75.3	79.1	89.3	82.0	82.2	84.9	14.8/16.3
Yes	Yes	No	77.8	78.6	90.1	82.2	84.5	87.5	15.2/16.4
No	Yes	Yes	77.2	80.4	90.9	83.6	83.9	86.8	15.1/16.5
Yes	Yes	Yes	77.8	78.8	89.9	82.1	84.5	88.0	14.8/16.0

Policy Configuration			**Success**	**Inform**			**Complete**	**Booking**	**Average**
Hierarchy	**Action Pruning**	L_{CE}	**Rate**	**Precision**	**Recall**	**F1**	**Rate**	**Rate**	**Turn (succ/all)**
No	No	No	43.6	79.4	73.6	73.7	46.4	77.0	25.1/30.5
Yes	No	No	52.8	78.0	79.5	76.6	60.0	82.0	24.1/27.8
No	Yes	No	60.0	78.8	81.5	78.4	67.8	81.6	21.4/25.6
Yes	Yes	No	65.7	79.8	81.8	78.8	72.1	78.6	21.2/23.8
No	Yes	Yes	66.4	82.5	83.4	80.4	72.1	83.9	21.2/24.5
Yes	Yes	Yes	71.4	80.6	83.1	79.6	77.1	80.2	20.5/22.0

TABLE VII
ABLATION STUDY ON DIFFERENT CONFIGURATIONS BY SHOWING SUCCESS RATE (SR) AND F1 SCORE FOR EACH DOMAIN. THE BEST CONFIGURATIONS ARE SHOWN IN BOLD

Policy Configuration			**Hotel**		**Attraction**		**Restaurant**		**Train**		**Taxi**		**Police**		**Hospital**	
Hierarchy	**Action Pruning**	L_{CE}	**SR**	**F1**	**SR**	**F1**	**SR**	**F1**	**SR**	**F1**	**SR**	**F1**	**SR**	**F1**	**SR**	**F1**
No	No	No	76.5	51.4	95.2	93.2	95.9	71.9	97.1	87.6	70.8	74.1	100	94.4	100	100
Yes	No	No	76.1	52.9	94.5	92.7	96.3	70.0	97.1	87.3	79.3	83.8	100	94.4	100	100
No	Yes	No	74.5	51.9	93.0	93.2	**97.5**	**72.7**	98.4	89.2	88.4	91.6	100	94.4	100	100
Yes	Yes	No	76.5	52.7	94.0	92.8	97.0	72.3	98.4	89.0	94.4	97.2	100	94.4	100	100
No	Yes	Yes	75.4	51.7	**96.5**	**97.2**	96.8	**72.7**	**98.7**	**89.7**	92.5	94.6	100	94.4	100	100
Yes	Yes	Yes	**77.5**	**53.2**	94.2	94.1	97.0	72.3	98.4	89.0	**98.6**	**98.8**	100	94.4	100	100
Rule Policy			81.6	52.3	98.1	91.6	97.7	77.8	97.1	86.9	100	100	100	100	100	100

increases significantly as the dialogue act from user sentences can be predicted perfectly. It is because, the information in DST which is transformed as a state for dialogue policy contains true dialogue act from the current turn. It subsequently affects the dialogue policy to choose appropriate decision. This evidence clearly indicates that the solution to improving NLU part is urgently required to further improve the performance of the pipeline dialogue system.

In addition, the last column of Table IV shows the number of parameters of the policy models in various methods. It is found that the baseline models had smaller number of parameters than the proposed method, but required more computation time. This is because that the baseline models needed larger number of turns than the proposed method to complete each dialogue in test set. The model size of pipeline method is significantly reduced relative to that of end-to-end approaches.

TABLE VIII

Examples of Success and Failure of a Multi-Domain Dialogue Containing Three Goals Where Taxi Domain Occurred At the Last Turn. The Green Colored Text is a Correct System Response, Indicated by Successfully Providing Taxi Phone Number. Otherwise, the Red Colored Text is an Incorrect System Response. The Proposed HRLG is Compared to the PPO With Action Pruning and L_{CE} [42] as It Shows the Second Best Performance. User Sentence, System Sentence and System Dialogue Act (DA) Are Shown. The Slot ‘reqmore’ Means ’request More’

Goals :	{’restaurant’: {’info’: {’area’: ’centre’, ’food’: ’Indian’, ’price range’: ’expensive’}, ’request’: {’postcode’: ’?’}, ’book’: {’time’: ’15:30’, ’people’: ’7’, ’day’: ’Friday’}, ’booked’: ’?’}, ’hotel’: {’info’: {’name’: ’el shaddai’}, ’request’: {’address’: ’?’}}, ’taxi’: {’info’: {’arrive by’: ’13:45’}, ’request’: {’car type’: ’?’, ’phone’: ’?’}}}
Proposed Method (success dialogue)	
User	I need a restaurant to dine at in Cambridge on my upcoming trip. Is that located in the centre? I would prefer Indian food please. I am looking for a expensive restaurant.
System	It is in the centre area. There are 6 restaurants in that area that fit that criteria. That is an Indian restaurant. They are expensive. How does the Curry Garden sound?
System DA:	[[’Inform’, ’Restaurant’, ’Area’, ’centre’], [’Inform’, ’Restaurant’, ’Choice’, ’6’], [’Inform’, ’Restaurant’, ’Food’, ’Indian’], [’Inform’, ’Restaurant’, ’Price’, ’expensive’], [’Recommend’, ’Restaurant’, ’Name’, ’Curry Garden’]
User	I would like to know the postcode of the restaurant, if possible?
System	Their address is 106 Regent Street City Centre. The restaurant postcode is cb21dp. Would you like to try Curry Garden?
System DA:	[[’Inform’, ’Restaurant’, ’Addr’, ’106 Regent Street City Centre’], [’Inform’, ’Restaurant’, ’Post’, ’cb21dp’], [’Inform’, ’Restaurant’, ’Name’, ’Curry Garden’]
User	I do need a booking for Friday. There are 7 people. Can you please try 15:30?
System	All set. Your reference number is 00000002. Is there anything else I can help you with?
System DA:	[[’Book’, ’Booking’, ’Ref’, ’00000002’], [’reqmore’, ’general’, ’none’, ’none’]]
User	I’m looking for a hotel. I am looking for a hotel by the name of el shaddai. Could I get their address?
System	The hotel address is 41 Warkworth Street. Is there anything else I can help you with?
System DA:	[[’Inform’, ’Hotel’, ’Addr’, ’41 Warkworth Street’], [’reqmore’, ’general’, ’none’, ’none’]]
User	One more thing, I would like to book a taxi to commute between the two places. I want to arrive by 13:45. Do you have their phone number? It would be nice if you would tell me the car type.
System	It will be a toyota. The contact number is 96857634153. Is there anything else I can help you with today?
System DA:	[[’Inform’, ’Taxi’, ’Car’, ’toyota’], [’Inform’, ’Taxi’, ’Phone’, ’96857634153’], [’reqmore’, ’general’, ’none’, ’none’]]
User	I am all set. Have a nice day. Bye.
PPO with Action Pruning and L_{CE}* [42] *(failed dialogue)	
User	I need a restaurant to dine at in Cambridge on my upcoming trip. Is that located in the centre? I would prefer Indian food please. I am looking for a expensive restaurant.
System	It is in the centre area. There are 6 restaurants in that area that fit that criteria. They are Indian restaurant. They are expensive. Is there anything else I can help you with?
System DA:	[[’Inform’, ’Restaurant’, ’Area’, ’centre’], [’Inform’, ’Restaurant’, ’Choice’, ’6’], [’Inform’, ’Restaurant’, ’Food’, ’Indian’], [’Inform’, ’Restaurant’, ’Price’, ’expensive’], [’reqmore’, ’general’, ’none’, ’none’]]
User	Please pick one for me.
System	Okay, may I suggest Indian food? It is at 106 Regent Street City Centre. They serve Indian. They are expensive.
System DA:	[[’Inform’, ’Restaurant’, ’Food’, ’Indian’], [’Inform’, ’Restaurant’, ’Addr’, ’106 Regent Street City Centre’], [’Inform’, ’Restaurant’, ’Food’, ’Indian’], [’Inform’, ’Restaurant’, ’Price’, ’expensive’]
User	What is their postcode?
System	The post code is cb21dp. Is there anything else I can help you with?
System DA:	[[’Inform’, ’Restaurant’, ’Post’, ’cb21dp’], [’reqmore’, ’general’, ’none’, ’none’]]
User	I would like to book a table for Friday. I need a reservation for 7 people. I would like a table for 15:30.
System	Reference number is : 00000002. Did you need any further assistance today?
System DA:	[[’Book’, ’Booking’, ’Ref’, ’00000002’], [’reqmore’, ’general’, ’none’, ’none’]]
User	I need a hotel as well. I am also looking for the el shaddai. May I have the address for that hotel?
System	The hotel address is 41 Warkworth Street. Is there anything else I can help you with?
System DA:	[[’reqmore’, ’general’, ’none’, ’none’], [’Inform’, ’Hotel’, ’Addr’, ’41 Warkworth Street’]]
User	One more thing, I would like to book a taxi to commute between the two places. I want to arrive by 13:45. Do you have their phone number? It would be nice if you would tell me the car type.
System	It will be a toyota. You are welcome. Is there anything else I can help you with today?
System DA:	[[’Inform’, ’Taxi’, ’Car’, ’toyota’], [’reqmore’, ’general’, ’none’, ’none’]]
User	Please provide the phone number of the taxi.
System	Is there anything else I can help you with today?
System DA:	[’reqmore’, ’general’, ’none’, ’none’]

C. Ablation Study on Efficient Learning and Guidance

Ablation study is conducted by evaluating individual components of the proposed method in order to show their impact on learning efficiency. All of the learned dialogue policies are trained by using PPO due to its dominant performance compared to the other policy optimization methods. There are six configurations which are built by investigating three components in HRLG including the hierarchy in HRL, the action pruning for confounded state and the additional objective L_{CE} for low-level policy. The last two components involve in HITL as a guidance for agent during training. First, the importance of hierarchical strategy is evaluated without any guidance which is shown by the first two rows of Table V. It can be seen that this strategy benefits the agent in the training which improves the performance in the majority of metrics. This result indicates the hierarchical method successfully simplifies the multi-domain dialogue task into several dialogue tasks based on the base domain occurrence. An efficient learning can be achieved. Next, various combinations of hierarchical strategy *with guidance learning* are assessed. There are two points in this analysis and comparison. The first point is to examine if the guidance is efficiently learned by the hierarchy in RL. The advantage can be observed in the third and fourth rows of Table V. The guidance that solely comprised of action pruning is learned effectively by employing a hierarchical method, as evidenced by the improved metric scores. By only receiving action pruning as a guidance from human during training, the dialogue policy with hierarchy shows competitive performance compared to the previous work [42] which applied non-hierarchical dialogue policy. The second point examines the joint advantage of action pruning and objective L_{CE}. The results are shown by the last two rows. The efficient learning from guidance was successfully achieved with the combination of three components indicated by the dominant improvement among the other configurations, especially in the success rate, complete rate and booking rate which are important metrics to indicate model capability in handling user goal. Even though this setting resulted in lower precision, recall and F1 score than the setting which only applied two guidances, the differences are not really significant.

To further demonstrate different learning strategies and dialogue properties in HRLG, Table VI reports the test results containing one, two and three domains as shown in top, middle and bottom, respectively, in presence of different number of dialogues. In the first test which involves 337 dialogues with one domain from the test set, very convincing performance is exhibited by all of the configurations which successfully achieve the success rate more than 94%, the complete rate more than 95%, very high number in booking rate and F1 score with very low turn. Different configurations perform comparably. The benefit of the guidance provided by human becomes clear in the test with dialogues containing two domains where there is a clear gap especially in term of success rate and booking rate, more than 2% in absolute improvement from the first two to the last three rows in the configurations. Unfortunately, the PPO implementation in HRLG could not take the advantage from action pruning in the third configuration as the obtained score in all metrics are just the same as the first two configuration. There are 523 dialogues in this test which take the highest proportion. Meanwhile, in the last test which involved 140 dialogues from test set containing 3 domains, which is the most challenging task, the advantage of implementing hierarchical approach is shown convincingly, especially in term of the success rate and complete rate where the absolute improvement reaches more than 9%. The benefit of guidance learning using action pruning and L_{CE} is obvious. In this test, one weakness is that the booking rate tends to be reduced due to the hierarchical strategy, even though all other scores are very convincing.

Table VII illustrates the assessment of success rate and F1 score for different configurations in each domain of the dialogue test set. The most impacted domain owing to the suggested hierarchical setting and guidance learning may be identified by examining this outcome. All of the dialogue policy configurations in the police and hospital domains earn a perfect success rate and a nearly perfect F1 score. In addition, the proposed method significantly benefits the performance for those 140 dialogues with three domains where taxi domain occurs in the last order of domains. Even without human assistance, introducing the hierarchical strategy improves the success rate and F1 score of taxi domain by near 10%. The improvement is increased if the human guidance using action pruning and objective L_{CE} is merged during dialogue policy optimization as the results of both success rate and F1 reach nearly perfect score. This evidence indicates that the proposed learning strategy could properly identify the state information from DST, conditioned by the previous dialogue turns, resulting in the appropriate response. An empirical example of dialogue is shown by Table VIII where the proposed system successfully generates response in taxi domain after addressing the previous turns involving two different domains. The previous method [42] could not provide a correct information due to inability to extract the information from the complicated state built from three different domains. Meanwhile, all of the configurations including those which utilize the guidance could not perform well in the hotel domain, which is the most challenging domain. It has 47 and 21 different possibilities of system and user dialogue act, respectively. Even the rule policy could only achieve the success rate around 81%. In general, the proposed method performs well in most of domains for multi-domain dialogue management. Source codes and model parameters are provided and can be accessed on https://github.com/NYCU-MLLab/.

V. Conclusion

A novel strategy to efficiently learn from the guidance from both of the human and the environment to establish high performance dialogue policy with low computational cost has been proposed in this work. The strategy was initially started by imitation learning with auxiliary tasks from the dialogue dataset to provide a good pre-trained weights for the subsequent stage which applied hierarchical reinforcement learning with human-in-the-loop (HITL). The end-to-end system evaluation findings indicated that the suggested learning technique outperformed most of the previous approaches and performed nearly identically to the rule-based agent that served as a human in HITL. When compared to state-of-the-art approaches that employed end-to-end optimization with large-sized language model GPT-2 as the core model, the proposed method needed much reduced computation cost with competitive performance. Furthermore, based on the ablation study, the hierarchical strategy enabled the dialogue policy agent to learn feedback from human and environment effectively, as the problem formulation in multi-domain dialogue task was simplified due to the hierarchy property. Future research will be undertaken to add optimization in the NLU component in order to appropriately provide acceptable state representation to dialogue policy, allowing the dialogue pipeline system to pick appropriate action to satisfy user goals.

References

[1] Y. Zhang, Z. Ou, and Z. Yu, “Task-oriented dialog systems that consider multiple appropriate responses under the same context,” in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 5, pp. 9604–9611.

[2] W. Chen, J. Chen, P. Qin, X. Yan, and W. Y. Wang, “Semantically conditioned dialog response generation via hierarchical disentangled self-attention,” in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 3696–3709.

[3] T. Zhao, K. Xie, and M. Eskenazi, “Rethinking action spaces for reinforcement learning in end-to-end dialog agents with latent variable models,” in *Proc. Conf. North Amer. Chapter Assoc. Computat. Linguistics*, 2019, pp. 1208–1218.

[4] C.-T. Chu, M. Rohmatillah, C.-H. Lee, and J.-T. Chien, “Augmentation strategy optimization for language understanding,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2022, pp. 7952–7956.

[5] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language models are unsupervised multitask learners,” *OpenAI Blog*, vol. 1, no. 8, pp. 1–24, 2019.

[6] K. Crowston, “Amazon mechanical turk: A research tool for organizations and information systems scholars,” in *Shaping the Future of ICT Research. Methods and Approaches*. Berlin, Germany: Springer, 2012, pp. 210–221.

[7] R. Takanobu, Q. Zhu, J. Li, B. Peng, J. Gao, and M. Huang, “Is your goal-oriented dialog model performing really well? empirical analysis of system-wise evaluation,” in *Proc. Annu. Meeting Special Int. Group Discourse Dialogue*, 2020, pp. 297–310.

[8] J. Li et al., “Multi-domain task completion dialog challenge 2 at DSTC9,” in *Proc. AAAI Workshop Dialog Syst. Technol. Challenge*, 2021.

[9] N. Lubis, et al., “LAVA: Latent action spaces via variational auto-encoding for dialogue policy optimization,” in *Proc. Int. Conf. Comput. Linguistics*, 2020, pp. 465–479.

[10] Z. Chen, L. Chen, X. Liu, and K. Yu, “Distributed structured actor-critic reinforcement learning for universal dialogue management,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 2400–2411, 2020.

[11] G. Weisz, P. Budzianowski, P.-H. Su, and M. Gašić, “Sample efficient deep reinforcement learning for dialogue systems with large action spaces,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 11, pp. 2083–2097, Nov. 2018.

[12] M. Rohmatillah and J.-T. Chien, “Causal confusion reduction for robust multi-domain dialogue policy,” in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2021, pp. 3221–3225.

[13] F. Codevilla, E. Santana, A. Lopez, and A. Gaidon, “Exploring the limitations of behavior cloning for autonomous driving,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9329–9338.

[14] C. Wen, J. Lin, T. Darrell, D. Jayaraman, and Y. Gao, “Fighting copycat agents in behavioral cloning from observation histories,” in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 2564–2575.

[15] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” 2017, *arXiv:1707.06347*.

[16] P. Shah, D. Hakkani-Tür, B. Liu, and G. Tür, “Bootstrapping a neural conversational agent with dialogue self-play, crowdsourcing and on-line reinforcement learning,” in *Proc. Conf. North Amer. Chapter, Assoc. Comput. Linguistics*, 2018, pp. 41–51.

[17] S. Seneff and J. Polifroni, “Dialogue management in the Mercury flight reservation system,” in *Proc. ANLP-NAACL Workshop: Conversational Syst.*, 2000, pp. 11–16.

[18] A. Bordes, Y.-L. Boureau, and J. Weston, “Learning end-to-end goal-oriented dialog,” 2016, *arXiv:1605.07683*.

[19] X. Jin et al., “Explicit state tracking with semi-supervision for neural dialogue generation,” in *Proc. Int. Conf. Inf. Knowl. Manage.*, 2018, pp. 1403–1412.

[20] S. Ultes et al., “PyDial: A multi-domain statistical dialogue system toolkit,” in *Proc. Annu. Meeting Assoc. Comput. Linguistics: Syst. Demonstrations*, 2017, pp. 73–78.

[21] M. Eric et al., “MultiWOZ 2.1: Multi-domain dialogue state corrections and state tracking baselines,” 2019, *arXiv:1907.01669*.

[22] Q. Zhu et al., “ConvLab-2: An open-source toolkit for building, evaluating, and diagnosing dialogue systems,” in *Proc. Annu. Meeting Assoc. Comput. Linguistics: Syst. Demonstrations*, 2020, pp. 142–149.

[23] T.-C. Luo and J.-T. Chien, “Variational dialogue generation with normalizing flows,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2021, pp. 7778–7782.

[24] P. Budzianowski and I. Vulić, “Hello, it’s GPT-2 - how can I help you? towards the use of pretrained language models for task-oriented dialogue systems,” in *Proc. Workshop Neural Gener. Transl.*, 2019, pp. 15–22.

[25] T.-H. Wen et al., “A network-based end-to-end trainable task-oriented dialogue system,” in *Proc. Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2017, pp. 438–449.

[26] W. Lei, X. Jin, M.-Y. Kan, Z. Ren, X. He, and D. Yin, “Sequicity: Simplifying task-oriented dialogue systems with single sequence-to-sequence architectures,” in *Proc. Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 1437–1447.

[27] J.-T. Chien, W.-L. Liao, and I. El Naqa, “Exploring state transition uncertainty in variational reinforcement learning,” in *Proc. Eur. Signal Process. Conf.*, 2020, pp. 1527–1531.

[28] J.-T. Chien and P.-C. Hsu, “Stochastic curiosity exploration for dialogue systems,” in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, 2020, pp. 3885–3889.

[29] J. Fu, K. Luo, and S. Levine, “Learning robust rewards with adversarial inverse reinforcement learning,” 2017, *arXiv:1710.11248*.

[30] R. Takanobu, H. Zhu, and M. Huang, “Guided dialog policy learning: Reward estimation for multi-domain task-oriented dialog,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2019, pp. 100–110.

[31] Z. Li et al., “Guided dialogue policy learning without adversarial learning in the loop,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2020, pp. 2308–2317.

[32] C.-E. Hsu, M. Rohmatillah, and J.-T. Chien, “Multitask generative adversarial imitation learning for multi-domain dialogue system,” in *Proc. IEEE Autom. Speech Recognit. Understanding Workshop*, 2021, pp. 954–961.

[33] I. J. Goodfellow et al., “Generative adversarial nets,” in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[34] B. Peng, X. Li, J. Gao, J. Liu, and K.-F. Wong, “Deep Dyna-Q: Integrating planning for task-completion dialogue policy learning,” in *Proc. Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 2182–2192.

[35] Y. Wu, X. Li, J. Liu, J. Gao, and Y. Yang, “Switch-based active deep Dyna-Q: Efficient adaptive planning for task-completion dialogue policy learning,” in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 7289–7296.

[36] A. Vaswani et al., “Attention is all you need,” in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.

[37] Z. Lin, A. Madotto, G. I. Winata, and P. Fung, “MinTL: Minimalist transfer learning for task-oriented dialogue systems,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2020, pp. 3391–3405.

[38] Y. Yang, Y. Li, and X. Quan, “UBAR: Towards fully end-to-end task-oriented dialog system with GPT-2,” in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 16, pp. 14230–14238.

[39] Y. Jang, J. Lee, and K.-E. Kim, “Offline reinforcement learning for end-to-end task-oriented dialogue system,” in *Proc. Int. Conf. Learn. Representations*, 2022.

[40] B. Zhang et al., “A hybrid task-oriented dialog system with domain and task adaptive pretraining,” 2021, *arXiv:2102.04506*.

[41] J. Kulhánek, V. Hudeček, T. Nekvinda, and O. Dušek, “AuGPT: Dialogue with pre-trained language models and data augmentation,” 2021, *arXiv:2102.05126*.

[42] M. Rohmatillah and J.-T. Chien, “Corrective guidance and learning for dialogue management,” in *Proc. ACM Int. Conf. Inf. Knowl. Manage.*, 2021, pp. 1548–1557.

[43] J.-W. Jang, M. Rohmatillah, and J.-T. Chien, “AVAST: Attentive variational state tracker in a reinforced navigator,” in *Proc. Int. Joint Conf. Natural Lang. Process.*, 2022, pp. 424–433.

[44] X. Zhou et al., “EAST: An efficient and accurate scene text detector,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2642–2651.

[45] O. Nachum, S. Gu, H. Lee, and S. Levine, “Data-efficient hierarchical reinforcement learning,” in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 3307–3317.

[46] R. J. Williams, “Simple statistical gradient-following algorithms for connectionist reinforcement learning,” *Mach. Learn.*, vol. 8, no. 3-4, pp. 229–256, 1992.

[47] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, “High-dimensional continuous control using generalized advantage estimation,” in *Proc. Int. Conf. Learn. Representations*, 2016.

[48] J.-T. Chien and Y.-C. Chiu, “Bayesian multi-temporal-difference learning,” *APSIPA Trans. Signal Inf. Process.*, vol. 11, no. 1, pp. 1–31, 2022.

[49] P. Budzianowski et al., “MultiWOZ - A large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 5016–5026.

[50] R. Takanobu, R. Liang, and M. Huang, “Multi-agent task-oriented dialog policy learning with role-aware reward decomposition,” in *Proc. Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 625–638.

[51] T. Tieleman and G. Hinton, “Divide the gradient by a running average of its recent magnitude,” *COURSERA: Neural Netw. Mach. Learn.*, vol. 4, no. 2, pp. 26–31, 2012.

[52] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2019, pp. 4171–4186.

[53] J.-T. Chien and S.-H. Yang, “Model-based soft actor-critic,” in *Proc. Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2021, pp. 2028–2035.

[54] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter,” 2019, *arXiv:1910.01108*.

[55] Z. Wang et al., “Critic regularized regression,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 7768–7778.

[56] L. Chen et al., “Decision transformer: Reinforcement learning via sequence modeling,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 15084–15097.

[57] A. Rastogi, X. Zang, S. Sunkara, R. Gupta, and P. Khaitan, “Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset,” in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 5, pp. 8689–8696.

[58] T.-H. Wen et al., “Conditional generation and snapshot learning in neural dialogue systems,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 2153–2162.

[59] B. Byrne et al., “Taskmaster-1: Toward a realistic and diverse dialog dataset,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2019, pp. 4516–4525.

[60] L. Van der Maaten and G. Hinton, “Visualizing data using t-SNE,” *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.

Mahdin Rohmatillah (Graduate Student Member, IEEE) received the B.Eng. degree majoring in electrical engineering from Brawijaya University, Malang, Indonesia, in 2016, and the M.S. degree majoring in electrical engineering from the National Sun Yat-sen Univesity, Kaohsiung, Taiwan, in 2018. He is currently working toward the Ph.D. degree with the National Yang Ming Chiao Tung University, Hsinchu, Taiwan. His research interests include machine learning, deep reinforcement learning, and dialogue system.

Jen-Tzung Chien (Senior Member, IEEE) is currently the Lifetime Chair Professor of electrical and computer engineering, and computer science with the National Yang Ming Chiao Tung University, Hsinchu, Taiwan. He was the tutorial speaker of AAAI, IJCAI, ACL, KDD, MM, ICASSP, ICME, CIKM, IJCNN, COLING and INTERSPEECH. He has authored or coauthored extensively three books and more than 250 peer-reviewed articles His research interests include machine learning, deep learning, Bayesian learning with applications on natural language processing, and computer vision.