Neurocomputing 648 (2025) 130639

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Beyond game environments: Evolutionary algorithms with parameter space noise for task-oriented dialogue policy exploration

Qingxin Xiao [a,b], Yangyang Zhao [c], Lingwei Dang [a], Yun Hao [a], Le Che [d], Qingyao Wu [a,b,*]

[a] South China University of Technology, School of Software Engineering, Guangzhou, 510006, Guangdong, PR China
[b] Institute for Super Robotics(Huangpu), Guangzhou, 510700, Guangdong, PR China
[c] Changsha University of Science and Technology, School of Computer and Communication Engineering, Changsha, 410114, Hunan, PR China
[d] South China University of Technology, School of Architecture, Guangzhou, 510006, Guangdong, PR China

ARTICLE INFO

Communicated by D. Macciò

Keywords:
Task-oriented dialogue
Dialogue policy exploration
Reinforcement learning
Evolutionary algorithms

ABSTRACT

Reinforcement learning (RL) has achieved significant success in task-oriented dialogue (TOD) policy learning. Nevertheless, training dialogue policy through RL faces a critical challenge: insufficient exploration, which leads to the policy getting trapped in local optima. Evolutionary algorithms (EAs) enhance exploration breadth by maintaining and selecting diverse individuals, and they often add parameter space noise among different individuals to simulate mutation, thereby increasing exploration depth. This approach has proven to be an effective method for enhancing RL exploration and has shown promising results in game domains. However, previous research has not analyzed its effectiveness in TOD dialogue policy. Given the substantial differences between gaming contexts and TOD dialogue policy, this paper explores and validates the efficacy of EAs in TOD dialogue policy, investigating the effects of different evolutionary cycles and various noise strategies across different dialogue tasks to determine which combination of evolutionary cycle and noise strategy is most suitable for TOD dialogue policy. Additionally, we propose an adaptive noise evolution method that dynamically adjusts noise scales to improve exploration efficiency. Experiments on the MultiWOZ dataset demonstrate significant performance improvements, achieving state-of-the-art results in both on-policy and off-policy settings.

1. Introduction

Task-oriented dialogue (TOD) systems are designed to help users complete specific tasks, such as booking a restaurant or purchasing movie tickets, through natural language conversations [1]. These systems rely on dialogue policies (DP) to determine the optimal response action at each turn, thereby guiding the conversation to successfully fulfill the user's task. Dialogue policies are typically learned through reinforcement learning (RL), where the system receives rewards based on interaction outcomes and incrementally improves its action selection [2].

In recent years, large language models (LLMs) have demonstrated remarkable performance across various natural language processing (NLP) tasks [3], opening up new possibilities for TOD dialogue policy learning. However, directly applying LLMs to TOD policy learning presents significant challenges. TOD systems are inherently user-goal-driven, whereas LLMs are primarily trained for broad, general-purpose tasks, often performing suboptimally in goal-oriented scenarios. Domain-specific fine-tuning can potentially address this issue [4], but such a process requires substantial computational resources. Moreover, LLMs struggle to maintain user-goal memory over extended dialogues, which significantly impairs their reasoning ability in multi-turn interactions. This limitation may even result in generating irrelevant or incoherent responses (known as hallucinations), thereby undermining the system's capability to assist users in achieving their specific goals [5].

Given the aforementioned limitations of LLMs [6], TOD systems still primarily rely on RL to learn dialogue policies. While RL is well-suited for sequential decision-making tasks like TOD [2], RL-based approaches in TOD face a critical challenge of insufficient exploration. This issue arises because TOD systems often use a reward function that provides small penalties per turn to encourage shorter dialogues and a larger reward upon task success. Consequently, the training reward signal tends to be sparse [7], making it difficult for the model to receive sufficient feedback early in training to learn an effective dialogue policy. Additionally, RL-based TOD systems map dialogue information (such as user intent and context) to the state space and the system's

* Corresponding author at: South China University of Technology, School of Software Engineering, Guangzhou, 510006, Guangdong, PR China.
E-mail address: qyw@scut.edu.cn (Q. Wu).

https://doi.org/10.1016/j.neucom.2025.130639
Received 5 December 2024; Received in revised form 15 May 2025; Accepted 28 May 2025
Available online 13 June 2025

responses to the action space, resulting in highly complex and dynamic state–action spaces. This complexity often hinders sufficient exploration by RL methods, causing the system to converge prematurely to local optima [8]. These exploration limitations not only restrict the discovery of globally optimal policies but also reduce the system's adaptability to diverse dialogue scenarios, ultimately impacting overall performance.

To alleviate the issue of insufficient exploration in RL-based dialogue policy in the TOD domain, several approaches have been proposed. These approaches primarily focus on policy dead-end detection and resurrection [9], curiosity-driven exploration [10], and action space noise injection [11], among others. While these methods partially address the exploration challenges, they are often limited to adjustments within a single dimension of exploration, making it difficult to balance the breadth and depth of exploration. When applied to more complex multi-domain dialogue scenarios, their exploration efficiency tends to significantly decrease.

We note that in the gaming domain, evolutionary algorithms (EAs) are commonly used to address the exploration limitations of game agents [12]. EAs simulate natural selection and genetic mechanisms, maintaining and selecting diverse individuals to balance the breadth and depth of exploration, thereby conducting an extensive search of the solution space to find the optimal solution [13]. EAs operate on two timescales: population evolution and individual lifelong learning. These two learning processes complement each other and can provide rich training signals for reinforcement learning. EAs typically inject noise into the individual parameter space, which further enhances exploration efficiency. We find this approach particularly suitable for TOD policy, where the reward signal is less pronounced. However, there are significant differences between game tasks and TOD policies, and the effectiveness of EAs has not been validated in task-oriented dialogue policies. These differences are mainly reflected in the following aspects: in the gaming domain, learning objectives are clear, feedback is immediate, and the environment is controllable, enabling EAs to quickly optimize policies through quantitative metrics; whereas in the TOD domain, the objectives are complex and diverse, and the environment is highly uncertain, leading to sparse rewards and delayed feedback, making it difficult to apply EAs in such complex environments. In summary, given the significant differences between the gaming domain and TOD, this paper mainly explores and validates the effectiveness of evolutionary algorithms in task-oriented dialogue policy learning and designs an evolutionary algorithm framework that effectively addresses the dialogue policy exploration problem, tailored to the characteristics of TOD tasks.

This paper explores the effectiveness of EAs in TOD dialogue policy. Specifically, we focus on three dimensions across different task scenarios:

- The impact of EAs with different evolution cycles on TOD dialogue policies
- The effects of combining EAs with various types of parameter space noise in TOD dialogue policy
- An adaptive noise evolutionary exploration framework is proposed, specifically designed for TOD tasks and characterized by its plug-and-play nature. To the best of our knowledge, this is possibly the first EAs training framework aimed at addressing the exploration limitations in TOD dialogue policy learning

Our work provides a comprehensive analysis of these strategies and offers valuable insights and benchmarks for future research on optimizing TOD systems.

2. Related work

2.1. Exploration challenges in TOD

Insufficient exploration of RL dialogue policies has long been a challenge in TOD systems. To address this issue, several methods have been proposed. For example, [9] introduced a 'dead-end detection and resurrection' strategy, which effectively detects and guides the dialogue system to avoid dead ends during the policy exploration phase, thereby improving dialogue quality. However, this approach has limitations when applied to dialogue systems that cannot access a database or encounter dead ends beyond the scope of the dialogue policy. [10] proposed an alternative curiosity-driven exploration strategy, which aims to promote the learning of task-oriented dialogue policies by balancing exploration and exploitation. However, this method greatly increases the complexity of the exploration strategy and is only effective in the early stages of exploration, limiting its overall effectiveness. Additionally, researchers have attempted to enhance exploration by alleviating sparse rewards. In this context, inverse reinforcement learning (IRL) and reward shaping techniques have been proposed to learn denser rewards and accelerate the learning process [14,15]. However, IRL comes with significant computational overhead, and reward shaping may lead to unintended behaviors [16]. Moreover, some methods design reward signals for each dialogue step, but such reward designs may lack semantic relevance to dialogue goals, limiting learning effectiveness [17,18].

Researchers have also explored behavior cloning methods. Unlike reinforcement learning methods that rely on sparse reward signals, behavior cloning directly learns policies by imitating expert behavior from labeled dialogue data, fundamentally avoiding the sparse reward issue. [19] pre-trained dialogue policies using behavior cloning on labeled dialogue corpora, significantly improving learning efficiency. [20] used expert simulators to provide dialogue behavior, employing behavior cloning to train diverse user models, enhancing the performance and robustness of dialogue policies. Although behavior cloning partially mitigates the sparse reward problem, it still depends on large amounts of labeled data and lacks true autonomous learning capabilities, making it difficult to fully leverage its advantages in complex, dynamic dialogue environments.

2.2. EAs for RL enhancement

The integration of EAs with RL for policy optimization has been widely explored due to EAs' diverse exploration capabilities and global optimization advantages, which make them a valuable tool for addressing various challenges in RL. For example, Moriarty et al. [21] highlighted the potential of EAs in RL by utilizing them to guide policy optimization. Leite et al. [13] introduced the use of EAs to replace the traditional Bellman equation for value function optimization, demonstrating how EAs can enhance exploration within the solution space. Kalashnikov et al. [22] applied EAs to initialize a random action population, combining them with the cross-entropy method and value function guidance to optimize action selection. Additionally, Jaderberg et al. [23] implemented an evolutionary process to train a population of policies with varying hyperparameters, iteratively replacing individuals based on fitness and applying hyperparameter perturbations to further refine the solutions.

However, despite the promising use of EAs in RL, their potential to address specific challenges in TOD systems has not been fully explored. This gap is precisely what our research aims to address.

3. Task-oriented dialogue system

The framework of a TOD system is illustrated in Fig. 1. In a TOD system, information from speech recognition or text input first passes through the natural language understanding (NLU) module [24], which converts it into a corresponding semantic frame. This semantic frame is then passed to the dialogue manager (DM), which includes a state tracker and a policy learner [25]. The DM accumulates the semantic information from each utterance, consistently tracks the dialogue state [26], and generates the system's next action. To demonstrate experimental effectiveness, this study uses Deep Q-networks (DQN) [27]

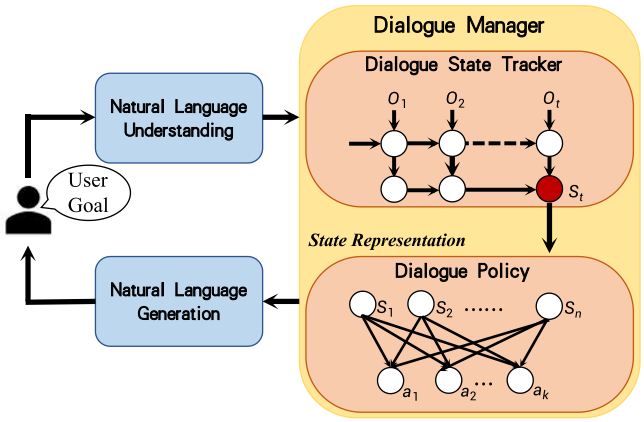


Fig. 1. Illustration of the task-oriented dialogue system.

and Proximal Policy Optimization (PPO) [28], two RL methods known for their fast convergence and strong performance on TOD tasks, as dialogue policies for exploration and validation. This section provides a detailed description of the dialogue policy module and its reinforcement learning paradigm [29].

3.1. Dialogue policy

Dialogue policy (DP) in task-oriented dialogue systems aims to generate the next system response based on the current dialogue state, assisting users in completing their tasks [30]. The state representation typically includes the user's most recent action, the system's latest response, available results retrieved from the database, dialogue turn information, and historical dialogue records [31]. After processing by the dialogue state tracker, this information is transformed into vector form, which serves as the input for DP [32]. In this process, the system generates the next action a through the policy $\pi(s)$ based on the state representation s. The learning of dialogue policies requires continuous real-time interaction with users to adjust the policy [33], maximizing dialogue success rates. Therefore, reinforcement learning methods are predominantly employed for dialogue policy learning [34].

3.2. RL paradigm for DP

In reinforcement learning-based policy learning for task-oriented dialogue systems, the agent learns an optimal action-selection policy π to maximize cumulative rewards. The agent interacts with the environment E, transitioning through the state space S and choosing actions within the action space A to generate responses that meet the task objectives. At each timestep t, given a state s_t, the agent selects an action a_t based on the policy $\pi(a_t|s_t)$ and receives a reward r_t, which quantifies the effectiveness of the action in progressing toward the task objective. The goal of policy learning is to find a policy π that maximizes the expected cumulative reward $R = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r_t\right]$, where T is the terminal timestep of the dialogue, and $\gamma \in [0, 1)$ is a discount factor that places greater emphasis on immediate rewards. Through continuous interaction and adjustment, the agent learns a generalized policy across diverse dialogue scenarios, balancing exploratory actions with those that lead to immediate success.

4. Methods

As research on evolutionary algorithms (EAs) and RL has progressed, more researchers have found that combining the strengths of these two approaches can effectively address challenges in traditional RL, leading to the development of various hybrid methods. In the field of gaming, both RL and EA have achieved significant progress, and their integration has shown potential for synergistic optimization. However, in TOD systems based on RL, similar research remains unexplored. This points to an important future direction: exploring how combining EA and RL policies could better address exploration deficiencies in TOD policies.

In this chapter, we first introduce the basic paradigm of EAs in Section 4.1 and construct a training framework for TOD dialogue policies based on EAs. In Section 4.2, we present the parameter space noise perturbation strategy, which introduces diverse spatial noise strategies to simulate individual mutations in EAs. In Section 4.3, we propose an adaptive noise mechanism and develop an adaptive noise evolution framework tailored for TOD dialogue policies. This framework adaptively adjusts noise scales based on the specific exploration performance of each model, thereby better guiding dialogue policy exploration.

4.1. Evolutionary algorithms

EAs are a class of gradient-free black-box optimization methods that iteratively optimize solutions by simulating Darwinian evolution. By leveraging population diversity and gradient-free random search, EAs achieve robust global search capabilities. Compared to traditional local search algorithms, such as gradient descent, evolutionary algorithms exhibit superior global optimization performance in the solution space. The differences between evolutionary algorithms and the RL methods discussed in this paper are illustrated in Fig. 2.

To facilitate a more convenient and intuitive exploration of the effectiveness of EAs in TOD dialogue policies, this paper integrates the EAs framework with TOD systems, constructing an EAs-based reinforcement learning framework (DERL) to explore the performance of different EAs. The process begins by initializing a population of DQN network agents $P = \{I_1, I_2, \ldots, I_N\}$, which are used to simulate the population. These neural network agents make action decisions and interact with the environment. Throughout their entire lifecycle, the agents are trained, and their fitness scores are periodically evaluated. The individuals with the highest fitness scores are selected as parent individuals, and offspring are generated through mutation. The offspring are then evaluated for fitness, and the elite individuals are selected to form the next generation of the population. This process is repeated iteratively to continuously evolve the population.

4.2. Exploration with parameter-space noise

EAs commonly enhance exploration by adding parameter space noise during the evolutionary selection process. Injecting noise into the parameter space allows EAs to diversify individuals across generations, increasing the coverage of the solution space and helping to avoid local optima. Compared to exploration methods that introduce noise in the action space, parameter-space noise injection generally results in a more stable and convergent evolutionary training process [35]. In this paper, we integrate this exploration approach into the EAs framework discussed in the previous section to evaluate the performance of noise-based exploration methods in TOD systems. Unlike the methods described in the previous section, this approach involves creating multiple copies of each elite individual during the selection step, with each copy being perturbed by adding different types of parameter space noise to simulate mutation. This allows us to explore the effectiveness of different noise strategies in exploration. Specifically, we represent the policy neural network as $\pi(\theta)$, where θ is the parameter vector. To achieve structured exploration, we apply additive noise to the parameter space vector of the elite individual's copies, thereby sampling from a set of perturbed policies: $\theta_e = \theta + \text{Noise}(0, \sigma^2)$. The perturbed policy is sampled at the beginning of the current evolutionary round and remains unchanged throughout the round.

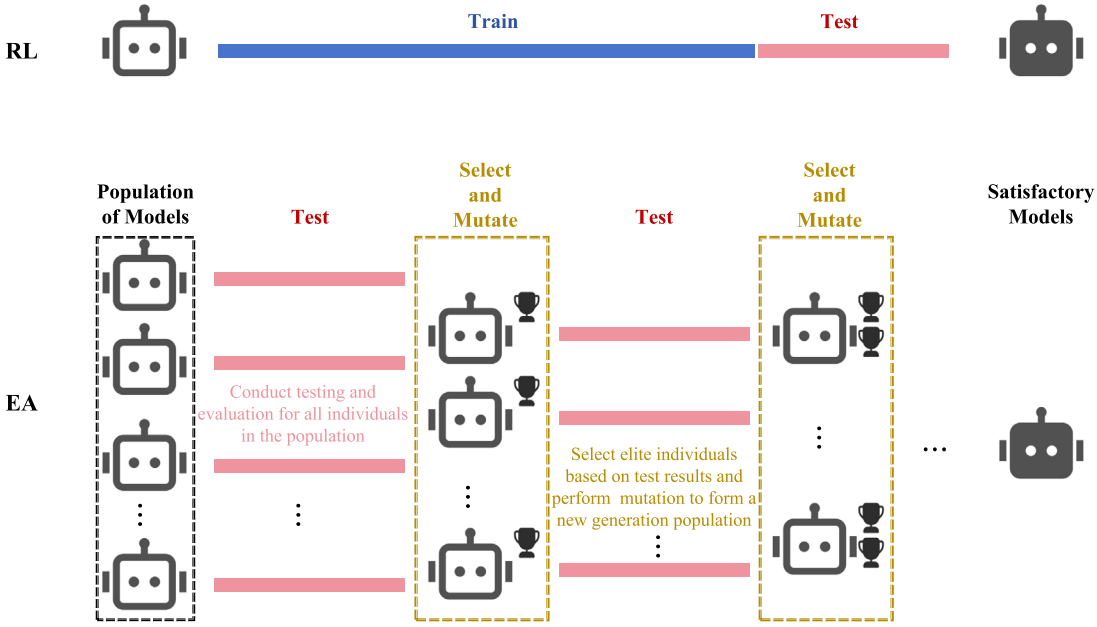


Fig. 2. Illustration of RL and EA algorithms: In RL, training, exploration, and testing of the model occur in a single dimension. In contrast, EA directly selects a high-performing model from the population across multiple generations. This approach optimizes the performance of agents more broadly by iteratively applying selection, crossover, and mutation to individuals within the population over multiple generations.

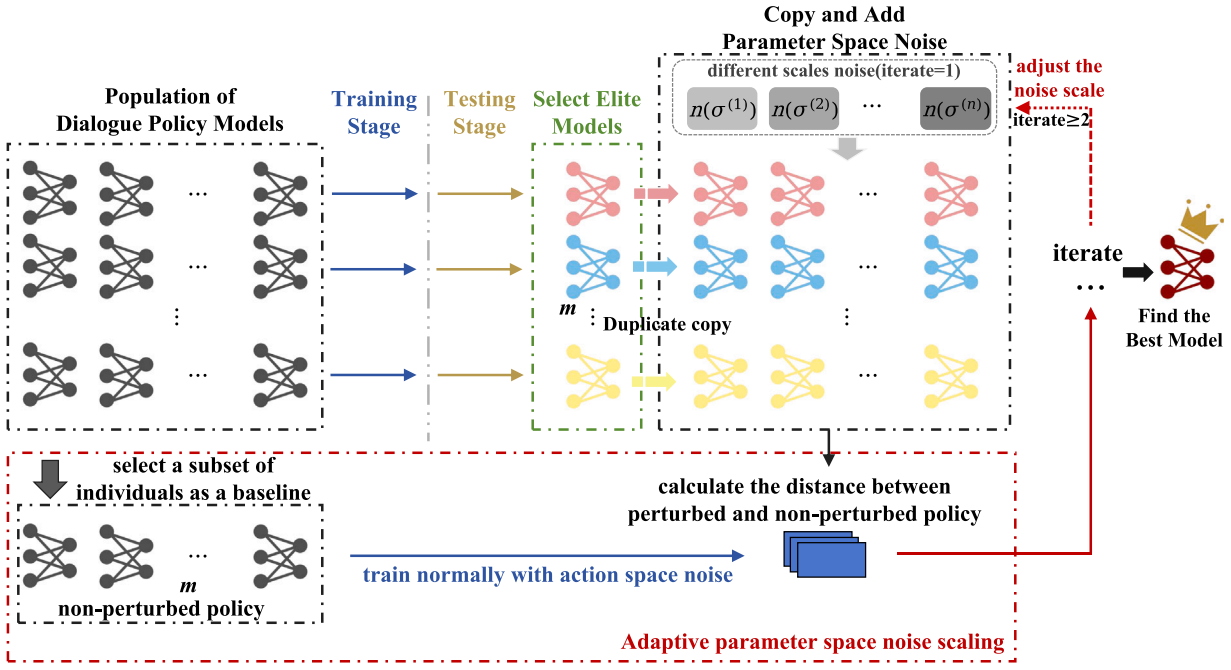


Fig. 3. The framework of Adapter-DERL. We first maintain a population of dialogue policy models. These models are trained simultaneously, and after a specified evolutionary cycle, all models are evaluated to select a certain number of elite individual models. For each elite model, multiple copies are created, with each copy perturbed by parameter-space noise of varying scales. Before training begins, a subset of models is randomly selected to serve as baseline models without parameter-space noise perturbation, using action-space noise strategies instead. After applying parameter-space noise perturbations to the elite models and their copies, we calculate the action-space difference between the perturbed and unperturbed models. This difference is compared to a predefined threshold to dynamically adjust the noise scale for the subsequent training process. This process is iteratively repeated until the optimal model that meets the desired criteria is identified. Notably, the content outside the red box in the diagram also represents the flowchart of DERL proposed in Section 3.1.

4.3. Adaptive parameter space noise scaling

In the action space, the impact of noise on the selection or variation of each action is relatively intuitive, as one can directly observe how noise affects decisions. However, in the parameter space, parameters manifest as weights and biases within the network, making it less straightforward to infer how perturbations in these parameters translate into changes in behavior. Consequently, adjusting and understanding the effects of noise in the parameter space becomes more complex. As the agent continues to learn, its parameters exhibit varying sensitivity to noise; thus, to maintain stable training, the scale of noise in the parameter space must also adapt.

To effectively control the variation in noise scales and ensure stable dialogue policy exploration, we propose a simple yet effective solution.

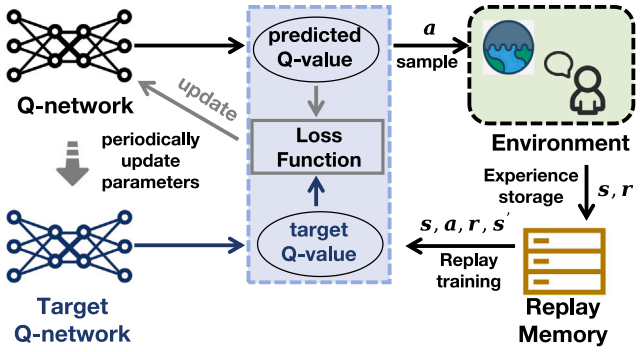


Fig. 4. The diagram illustrates the workflow of DQN in TOD task. It shows how the agent interacts with the environment (e.g., user input), selects actions based on the current policy, and receives rewards. The network is continuously updated using backpropagation through the Bellman equation to optimize the policy and generate the best responses in different dialogue states.

By associating the noise scale with the variance generated in the action space, we achieve adaptive noise scale adjustment. Building on the evolutionary approach and noise strategies introduced in Sections 4.1 and 4.2, we construct Adapter-DERL (A-DERL). Specifically, we select a certain number of individuals from the initial population as non-perturbed policies. These non-perturbed policies are trained using action-space noise through conventional training and do not participate in the evolutionary process. At regular intervals, we measure the divergence in the action space between perturbed policies undergoing evolutionary exploration and non-perturbed policies. Based on whether this divergence is below or above a given threshold, we adaptively increase or decrease the noise applied to the parameter space of the individual networks. The A-DERL process is shown in Fig. 3.

To demonstrate our method, we utilize two classic dialogue policies: DQN and PPO, representing off-policy and on-policy algorithms, respectively. Our noise scale adjustment formula is:

$$\sigma_{k+1} = \begin{cases} \alpha\sigma_k & \text{if } \mathcal{J}(\pi, \tilde{\pi}) \le \delta \\ \frac{1}{\alpha}\sigma_k & \text{otherwise} \end{cases} \tag{1}$$

where $\alpha \in \mathbb{R}^{>0}$ is a scaling factor, and $\delta \in \mathbb{R}^{>0}$ is a threshold. The function $\mathcal{J}(\cdot,\cdot)$ represents a divergence measure between the action distributions of the non-perturbed and perturbed policies.

4.3.1. DQN with adaptive noise scaling

The core idea of the DQN algorithm is to approximate the Q-value function using a deep neural network, thereby optimizing the policy to maximize long-term rewards in dialogue tasks. DQN trains the model by minimizing the difference in the Bellman equation:

$$L(\theta) = \mathbb{E}\left[\left(r + \gamma \max_{a\prime} Q(s\prime, a\prime; \theta^-) - Q(s, a; \theta)\right)^2\right]$$

where s is the current state, a is the action taken, r is the immediate reward, γ is the discount factor, θ represents the current Q-network parameters, and θ^- represents the target Q-network parameters. The terms $s\prime$ and $a\prime$ represent the next state and the next action, respectively, after taking action a in state s. By updating the parameters through backpropagation, DQN gradually optimizes the dialogue policy. In TOD, the DQN model interacts with the environment (such as user inputs) to learn the optimal response policy. By evaluating possible actions across various dialogue states, DQN can form an efficient decision-making model tailored to specific tasks, as shown in Fig. 4.

For the DQN algorithm, the policy is implicitly defined by the Q-value function. Solely using the discrepancy metric between Q and $\widetilde{Q}$ may present some issues. When the perturbed policy only modifies the bias of the final layer, causing every action's Q-value to increase by the same constant value, the simple metric $\|Q - \widetilde{Q}\|_2$ may still be non-zero, while the actual policies π and $\tilde{\pi}$ remain identical. Therefore, this paper adopts a probabilistic representation of the policy, converting the predicted Q-values into a probability distribution using the softmax function, defining the policies as $\pi, \tilde{\pi} : S \times \mathcal{A} \mapsto [0, 1]$, where

$$\pi(s) = \frac{\exp Q_i(s)}{\sum_i \exp Q_i(s)}$$

and $Q_i(\cdot)$ denotes the Q-value for action i, with $\tilde{\pi}$ defined similarly. By calculating the difference between the non-perturbed policy π and the perturbed policy $\tilde{\pi}$ in the action space, we can more accurately reflect the differences between the policies before and after perturbation. Using this probabilistic form of the policy, we measure this divergence in the action space as:

$$\mathcal{J}(\pi, \tilde{\pi}) = D_{\mathrm{KL}}(\pi \parallel \tilde{\pi}) \tag{2}$$

where D_{KL} denotes the Kullback–Leibler (KL) divergence, thereby normalizing the Q-values to mitigate the previously mentioned issues.

4.3.2. PPO with adaptive noise scaling

The PPO algorithm optimizes decision-making through two components: the actor network and the critic network. The actor network is responsible for generating the action policy based on the current state, while the critic network estimates the value of state–action pairs, typically using the advantage function ($\hat{A}_t$) to evaluate the quality of actions. The objective function of PPO can be expressed as:

$$\mathcal{L}(\theta) = \mathbb{E}_t\left[\min\left(r_t(\theta)\hat{A}_t, \mathrm{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right]$$

where $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\mathrm{old}}}(a_t|s_t)}$ is the ratio between the current and old policies, and $\hat{A}_t$ is the advantage function, which measures the advantage of taking a particular action compared to the average. The advantage function is computed as:

$$\hat{A}_t = Q(s_t, a_t) - V(s_t)$$

where $Q(s_t, a_t)$ is the state–action value function, representing the expected long-term reward for taking action a_t in state s_t, and $V(s_t)$ is the value function, representing the expected return of state s_t. By using the advantage function, PPO reduces excessive policy updates, thereby improving training stability. In TOD, the actor network selects the optimal response action based on the current dialogue state (e.g., user input), while the critic network provides feedback by estimating the advantage of each possible action. This approach effectively balances exploration and exploitation, gradually optimizing the dialogue policy and improving the system's performance across various dialogue scenarios, as shown in Fig. 5.

For the PPO algorithm, the dialogue policy is directly defined by a parameterized probability distribution, represented as π_θ. During the policy evolution process, the difference between the original policy π_θ and the perturbed policy $\pi_{\theta\prime}$ is assessed by calculating the KL divergence. Specifically, this difference is quantified as:

$$\mathcal{J}(\pi_\theta, \pi_{\theta\prime}) = \mathbb{E}_{s\sim\rho_\pi}\left[D_{\mathrm{KL}}(\pi_\theta(\cdot|s) \parallel \pi_{\theta\prime}(\cdot|s))\right], \tag{3}$$

where ρ_π represents the state distribution generated by the policy π_θ.

In the context of dialogue policy tasks, we identify the critical characteristic of PPO as its trust region constraint, which ensures smooth policy updates by limiting the KL divergence. This constraint is formalized as:

$$\mathbb{E}_{s\sim\rho_\pi}\left[D_{\mathrm{KL}}(\pi_\theta(\cdot|s) \parallel \pi_{\theta\prime}(\cdot|s))\right] \le \delta, \tag{4}$$

where δ is a predefined threshold that controls the maximum allowable KL divergence, preventing overly large policy updates that could jeopardize learning stability.

Furthermore, to enhance exploration, PPO can dynamically adjust the noise scale based on the KL divergence: reducing noise when the divergence approaches the threshold to ensure policy stability, and increasing noise when the divergence is low to promote exploration.

Table 1
The dataset information of the experiments. In total, there are 47 (domain, slot) pairs from the selected six domains.

Domain	Hotel	Train	Attraction	Restaurant	Taxi	Movie
	price					number of people
	parking	day		price		distance constraints
	type	departure		area	leave by	theater chain, date
	stars	arrive by	area	time	destination	task complete, city
Slot	name	leave at	name	food	departure	format, price
	internet	destination	type	name	arrive by	start time, video
	area	people		day		movie name, state
	people					ticket, theater, zip
	stay, day					greeting, closing

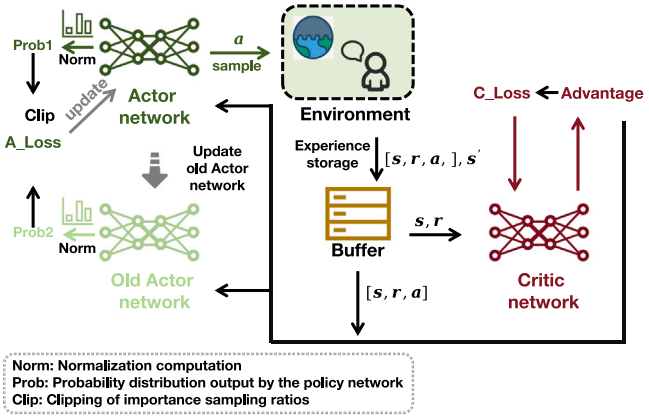


Fig. 5. The diagram illustrates the workflow of PPO in TOD task. It shows how the actor network selects the optimal response action based on the current dialogue state (e.g., user input) and how the critic network estimates the advantage of each action. Through the advantage function, the PPO algorithm balances exploration and exploitation, continuously optimizing the dialogue policy to improve the system's performance across various dialogue scenarios.

5. Experiments

To evaluate the performance of EAs, different noise strategies, and the proposed adaptive noise EAs in task-oriented dialogue scenarios, this paper focuses on conducting experiments at the dialogue policy learning level.

5.1. Dataset

We evaluated the performance in both single-domain tasks, specifically the movie-booking task, and multi-domain tasks, which include five domains such as restaurant reservation and taxi ordering. In the movie-booking tasks [36], the original dialogue data were collected via Amazon Mechanical Turk and annotated by domain experts. The annotated data include 11 types of dialogue acts and 29 slots. The dataset consists of a total of 280 annotated dialogues, with an average dialogue length of approximately 11 turns.

For the multi-domain tasks evaluation, we utilized the Multi-Domain Wizard-of-Oz 2.0 (MultiWoz) dataset [37], which is the largest human dialogue corpus covering seven domains, containing 8438 multi-turn dialogues, with an average of 13.68 turns per dialogue. Due to the limited amount of data in the hospital and police domains, and the fact that only training data were available, our experiments focused on data from the restaurant, hotel, attraction, taxi, and train domains [38]. The slots and corresponding data volumes for each domain in all datasets are shown in Table 1.

5.2. Metrics

The experiment primarily focuses on three metrics: task success rate, average dialogue turns, and average reward. We measure dialogue turns to reflect the dialogue cost, with each user utterance followed by a system response considered as one turn. The success rate is the proportion of dialogues successfully completed, which is achieved when all user requests are satisfied, and the booked entities meet the user's constraints. The average reward is calculated as the average reward obtained by the agent during the conversation. For the experiments based on EAs, all metrics are obtained by calculating the average value of all individuals in the population during the current evolutionary cycle.

5.3. Setup

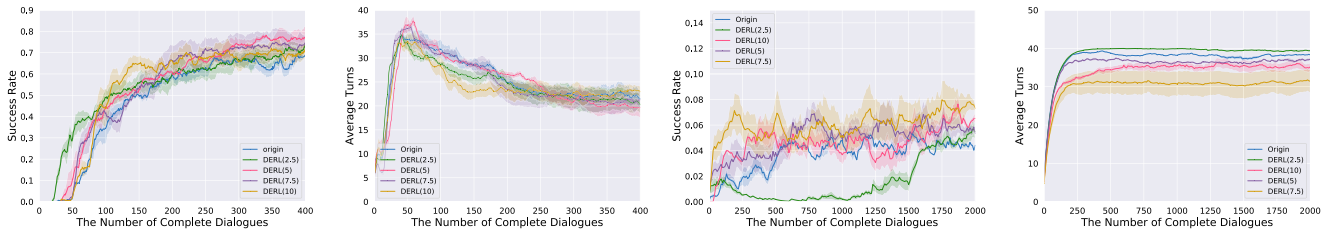
In the experiments, for the single-domain tasks, we employed the Microsoft Dialogue Challenge platform, which offers a unified experimental environment, standardized datasets, and publicly available rule-based user simulators, thereby promoting collaboration and benchmarking within the dialogue research community. For the multi-domain tasks, we leveraged the ConvLab-2 platform [39], which similarly provides standardized datasets and publicly accessible agenda-based simulators. Furthermore, to enhance the evaluation of our approach, we conducted a human evaluation involving real participants, as detailed in Section 5.9. The reinforcement learning agents are modeled as two-layer perceptrons, with a hidden layer size set to 80. The optimizer used is RMSprop, and the activation function is tanh. The batch size is 16, and the learning rate is set to 0.001. The buffer size for the experience replay pool is 10,000. DQN's target network employs a soft update mechanism with exponential moving average, updating parameters with $\tau = 0.01$. In the ϵ-greedy strategy, the initial value of ϵ is set to 0.2 and decays each episode, while the discount factor γ is set to 0.95. The maximum length of simulated dialogues is 40 turns, with dialogues exceeding this limit considered failures. When a dialogue is successful, the agent receives a reward of 80, while a failure incurs a penalty of −40. To encourage the policy to reach the goal more efficiently, the agent incurs a penalty of −1 for each turn, meaning that the more turns the agent takes, the lower the reward. For all experiments, an experience replay buffer containing 100 dialogues was pre-filled for all agents to warm start before training. In Sections 5.4–5.6, we conduct experiments with DQN as the dialogue agent, and in Section 5.7, we utilize PPO as the dialogue agent for experimentation.

In all the experimental plots, the line plot represents the average results from 10 different runs with distinct random seeds [40,41]. The average μ is computed as:

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$$

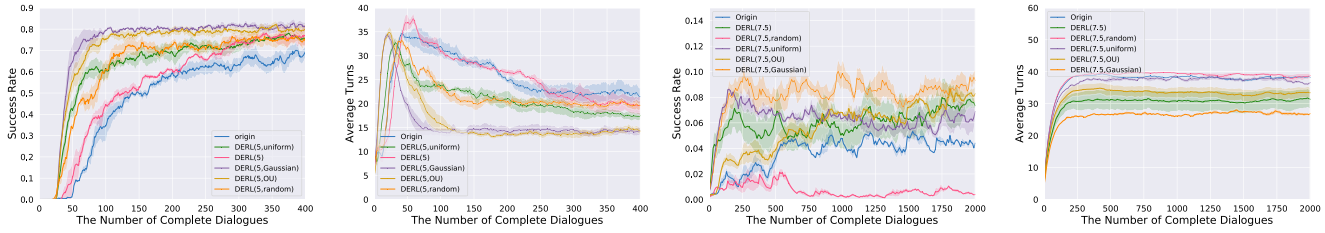
where x_i is the result of the ith experiment, and N is the total number of experiments (set to 10 in this paper). The colored shaded area surrounding the line plot represents the range of 0.5 times the standard deviation of all samples across the 10 experiments [42–44]. The standard deviation σ is calculated as:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$$



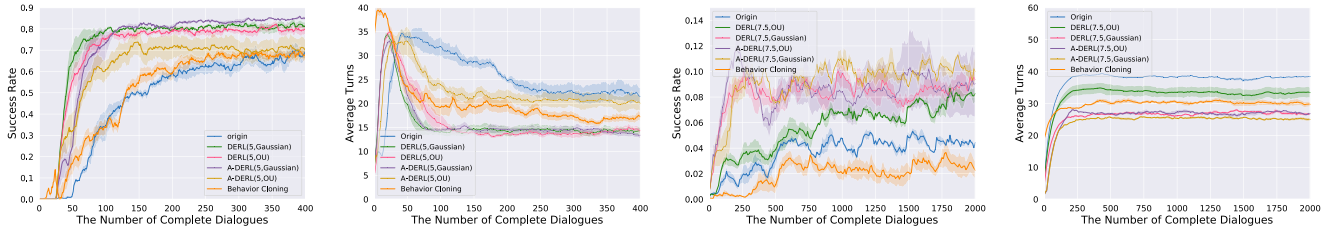
(a) movie-ticket booking (b) movie-ticket booking (c) multi-domain tasks (d) multi-domain tasks

Fig. 6. Effectiveness of different evolutionary cycle strategies in the movie-ticket booking domain and multi-domain tasks. (a) and (c) show the **success rate** for different methods, while (b) and (d) show the **average dialogue turns** for different methods.



(a) movie-ticket booking (b) movie-ticket booking (c) multi-domain tasks (d) multi-domain tasks

Fig. 7. Effectiveness of different space noise strategies in the movie-ticket booking domain and multi-domain tasks. (a) and (c) show the **success rate** for different methods, while (b) and (d) show the **average dialogue turns** for different methods.



(a) movie-ticket booking (b) movie-ticket booking (c) multi-domain tasks (d) multi-domain tasks

Fig. 8. Effectiveness of A-DERL in the movie-ticket booking Domain and multi-domain tasks. (a) and (c) show the **success rate** for different methods, while (b) and (d) show the **average dialogue turns** for different methods.

where μ is the mean, x_i is the result of the ith experiment, and N is the number of experiments.

In experiments involving evolutionary algorithms, we maintained a population of $P = 40$ agent networks for training. After each evolutionary cycle, the performance of individual networks in the population was evaluated. At the end of each generation, the top 10 elite individuals were retained, and three replicas were created for each elite individual. Based on the experimental setup, each replica was perturbed with parameter-space noise of the same type, with progressively increasing scales. The standard deviation of the noise ranged from 0.08 to 0.32 units in the parameter space of the neural networks. This process provided a new population of 40 network individuals for the next generation of evolution. For the validation experiments of A-DERL, we randomly selected 10 individuals from the initial population as unperturbed policies, trained using ϵ-greedy action-space noise. The noise scaling factor α was set to 1.11. For the movie-ticket booking single-domain task, the entire training cycle is set to 400 epochs, with each epoch representing a complete dialogue interaction centered around a user goal, which also corresponds to one reinforcement learning episode. In the multi-domain task, due to memory limitations of the experimental equipment, the training cycle is set to 2000 epochs. Notably, to ensure a fair comparison with the single-domain task, we modified the training mechanism of the ConvLab platform to align with the Microsoft Dialogue Challenge platform, ensuring that each epoch similarly corresponds to a complete dialogue interaction.

5.4. Comparing different evolution strategies

Fig. 6 presents the learning curves of four EAs in the movie-ticket booking task and the multi-domain tasks. (a) and (c) respectively plot the success rates over simulation cycles for the two tasks, while (b) and (d) respectively plot the average dialogue turns for the two tasks.

Previous studies have validated the effectiveness of EAs in improving reinforcement learning performance on the inverted pendulum task [13]. However, these studies only evaluated a fixed evolution cycle and introduced additional exploration techniques. Therefore, this section conducts extensive experiments on TOD tasks with multiple evolution cycle parameters. To balance exploration frequency and computational cost, and to analyze the impact of different evolution cycles on policy optimization, we selected four evolution cycle parameters based on the total number of training iterations: 2.5%, 5%, 7.5%, and 10%. This means that population evolution and selection occur after training reaches 2.5%, 5%, 7.5%, and 10% of the total training cycle, respectively, corresponding to DERL(2.5), DERL(5), DERL(7.5), and DERL(10) in Fig. 6.

In this experiment, the use of EAs improved the performance of reinforcement learning in both the movie-ticket booking task and the multi-domain tasks to varying degrees. For the simpler single-domain tasks of movie-ticket booking, DERL(5) achieved the best performance, indicating that more frequent evolution cycles (5%) were beneficial, likely due to the need for quicker adaptation. In contrast, for the more complex multi-domain tasks, DERL(7.5) yielded better results, as a

Table 2
Results of Different Methods on the MultiWoz Dataset (The Number of Complete Dialogues = 1000, 1500, 2000).

Agent		The Number of Complete Dialogues = 1000			The Number of Complete Dialogues = 1500			The Number of Complete Dialogues = 2000		
		Success	Reward	Turns	Success	Reward	Turns	Success	Reward	Turns
DQN	Origin	0.046	−51.8	38.2	0.040	−50.4	37.4	0.066	−52.2	38.9
	DERL(7.5)	0.056	−51.2	31.3	0.067	−48.4	30.6	0.078	−48.3	31.8
	DERL(7.5, Gaussian)	0.096	−34.8	26.8	0.081	−33.2	27.2	0.098	−30.4	26.6
	A-DERL(7.5, Gaussian)	**0.106**	−35.2	26.4	**0.092**	−34.7	25.6	**0.102**	−28.9	25.2
PPO	Origin	0.042	−42.1	11.2	0.065	−31.2	9.7	0.081	−30.1	6.9
	DERL(7.5)	0.074	−43.6	13.4	0.078	−29.6	10.8	0.089	−27.9	8.3
	DERL(7.5, Gaussian)	0.078	−32.8	10.2	0.082	−28.6	9.3	0.093	−27.5	7.2
	A-DERL(7.5, Gaussian)	**0.085**	−33.7	12.8	**0.102**	−29.2	10.4	**0.108**	−25.4	6.5

slightly less frequent evolution cycle (7.5%) allowed for sufficient exploration of the diverse task space before undergoing selection, leading to better overall performance. We also observed that overly infrequent evolution cycles do not necessarily result in optimal performance, as the population may fail to adjust strategies in a timely manner, reducing its responsiveness to environmental changes and limiting overall effectiveness. Therefore, the evolution frequency must strike a balance between adaptation and exploration depending on the specific task.

5.5. Comparing different parameter space noise to action space noise

To encompass mutation methods with different distribution characteristics and dynamic trends, this section evaluates the exploration effectiveness of four distinct parameter-space noise strategies in DQN, based on EAs: Gaussian noise, Ornstein–Uhlenbeck (OU) noise, uniform noise, and random noise. Gaussian and OU noise simulate stable and time-correlated perturbations, respectively, while uniform and random noise provide uniform and completely unstructured exploration. The baseline model adopts action-space noise with ϵ-greedy exploration. This setup aims to investigate how parameter-space and action-space noise influence the learning process of DQN in task-oriented dialogue systems. Evolution cycles of 5% and 7.5% are selected for the movie-ticket booking task and the multi-domain tasks, respectively, as these parameters showed optimal performance in the previous experiments. The results are shown in Fig. 7.

The experimental results indicate that in the movie-ticket booking task, all noise strategies significantly outperform ϵ-greedy in the early stages. This suggests that the added randomness helps the agent explore a more diverse set of actions, potentially allowing it to discover better action sequences more quickly than ϵ-greedy, which is more conservative and gradually reduces exploration. Both Gaussian noise and OU noise [45] exhibit the most consistent and superior performance, particularly in the later stages, surpassing ϵ-greedy in achieving higher success rates. This can be attributed to the way Gaussian and OU noise operate in the parameter space, introducing more structured exploration by perturbing the agent's internal parameters in a controlled manner, thereby helping the model more effectively avoid local optima. The smooth, continuous perturbation of Gaussian noise and the temporal correlation of OU noise may provide a more stable exploration pattern, resulting in better generalization in the simpler, more structured movie-ticket domain. In contrast, although average noise and random noise initially improved upon ϵ-greedy, they ultimately lagged behind. The reduced randomness introduced by average noise may have limited the diversity of exploration, leading to suboptimal long-term strategies. Random noise, lacking structure and correlation, may have caused the agent to explore too chaotically, diminishing its ability to optimize effective policies. This may explain why random noise failed to maintain its early advantages and performed worse than ϵ-greedy in later stages.

In the more complex multi-domain tasks, Gaussian noise remained the top performer, indicating its ability to balance exploration and exploitation through smooth perturbations of agent parameters, allowing it to more effectively handle the diverse and dynamic task space. However, random noise had almost no success, as its chaotic exploration strategy was particularly ill-suited for the multi-domain environment, where the agent needed to learn more nuanced policies to cope with different objectives and contexts. This further underscores the importance of effectively exploring in the parameter space, where maintaining a balance between exploration diversity and controlled learning is crucial for more complex environments. Interestingly, while OU noise initially did not surpass ϵ-greedy in the multi-domain tasks, it eventually exceeded it in the later stages. This delayed improvement can be explained by the temporal correlation of OU noise, which may allow the agent to explore more effectively as it adapts to complex and changing dialogue environments. Initially, temporally correlated noise may not perform well for rapid adaptation across a wide range of tasks, but as the agent accumulates more experience, the structured exploration enables it to identify better long-term policies than ϵ-greedy. In contrast, although average noise initially performed better than ϵ-greedy, its performance dropped sharply in the later stages. This indicates that over time, average noise may have failed to provide sufficient variability to explore the larger task space, leading to a stagnation in policy improvement.

Overall, the comparison of parameter-space noise (such as Gaussian and OU noise) with action-space noise (like ϵ-greedy) highlights the trend that parameter-space noise strategies provide more stable and effective exploration in complex environments. The results suggest that balancing exploration with controlled perturbations is vital, as parameter-space noise not only encourages diverse action selection but also helps optimize long-term policies, particularly in dynamic tasks where adaptability is crucial.

5.6. Is adaptive spatial noise effective in evolutionary strategies?

In this section, we validate the adaptive noise evolution method proposed in Section 4.3, selecting the previously best-performing Gaussian noise and OU noise for further in-depth comparison. The experimental results shown in Fig. 8 indicate that, in the movie-ticket booking task, adaptive Gaussian noise initially performs worse than both Gaussian and OU noise, but eventually surpasses them, achieving the best results in the later stages. In contrast, adaptive OU noise underperforms compared to the fixed OU noise. In the multi-domain tasks, adaptive Gaussian noise still proves to be the most effective, while adaptive OU noise performs similarly to fixed Gaussian noise, and OU noise ranks the lowest.

This suggests that Gaussian noise, with its smooth and continuous perturbations, strikes a better balance between exploration and policy convergence. The adaptive mechanism further enhances this balance by dynamically adjusting noise intensity, especially in the later stages, helping the model escape local optima more effectively. This is particularly evident in the movie-ticket booking task, where adaptive noise gradually reduces the noise range, enabling more precise

policy optimization, ultimately leading to superior performance compared to fixed noise strategies. On the other hand, OU noise, which already incorporates time correlation, inherently possesses some degree of adaptiveness. The additional adaptive adjustment may introduce instability in the noise dynamics, disrupting policy convergence and explaining why adaptive OU noise performs worse than the original OU noise.

In the multi-domain tasks, the superior performance of adaptive Gaussian noise underscores the importance of dynamically adjusting noise scales to handle diverse task spaces. multi-domain tasks involve a higher degree of complexity, and fixed noise strategies struggle to adapt effectively across different domains. Adaptive Gaussian noise, by flexibly adjusting its intensity based on feedback from different domains, offers a more responsive exploration mechanism, helping the agent find an optimal balance across varied task scenarios. In contrast, the time-correlated nature of OU noise may not be flexible enough for multi-domain environments. While adaptive OU noise performs similarly to Gaussian noise, it still fails to fully leverage its potential in exploration.

We referenced [20,46], where a reverse dialogue model was constructed by first learning the dialogue pattern through a random policy, and then improving the dialogue policy through imitation of expert demonstration data. A behavior cloning policy was built on the ConvLab-2 platform and compared with our method in two tasks. The experimental results, shown in Fig. 8, indicate that in the single-domain tasks, the behavior cloning method outperforms DQN in the early stages, suggesting that it can quickly adapt to the task and generate effective dialogue policies through imitation learning. However, as training progresses, the success rate of behavior cloning gradually aligns with DQN, and although the average dialogue turns are highest in the early stages, they eventually fall below DQN and A-DERL (5, OU). In the multi-domain tasks, behavior cloning performs the worst, with a success rate only about half of DQN's and the dialogue turns at a moderate level, lower than both DQN and DERL (7.5, OU). This may be due to behavior cloning's over-reliance on expert demonstration data and the lack of sufficient exploration mechanisms to handle the varying scenarios in multi-domain tasks, preventing it from effectively addressing the differences between tasks. Thus, its performance is inferior to other reinforcement learning methods in multi-domain tasks, while it can better leverage the advantages of imitation learning in single-domain tasks.

5.7. Effectiveness of EAs in on-policy algorithms

We previously validated the effectiveness of EA algorithms for TOD tasks using DQN as the dialogue policy. However, since DQN is an off-policy method, its results may not fully capture the performance of EA across different reinforcement learning paradigms. To further verify the effectiveness of EAs, we conducted experiments using the on-policy PPO algorithm as the dialogue policy on the more challenging MultiWoz dataset. We selected the three most representative frameworks from prior experiments: DERL (7.5), DERL (7.5, Gaussian), and A-DERL (7.5, Gaussian).

As shown in Table 2, the different EA variants demonstrated significant performance differences under the PPO framework. Firstly, DERL (7.5), with a selection rate of 7.5% per training cycle, surpassed the PPO baseline in optimizing dialogue policies, illustrating the potential of EA in steadily enhancing policy performance. However, DERL (7.5, Gaussian), which introduces Gaussian parameter-space noise to simulate mutation, further improved performance over the basic DERL (7.5). This result suggests that incorporating moderate random mutations can effectively widen exploration, leading to better policy optimization. Most notably, the adaptive noise framework A-DERL (7.5, Gaussian) achieved the best performance among all models. By dynamically adjusting noise scales based on exploration outcomes, this framework encourages broad exploration in the early stages and gradually narrows focus to enhance stability in later stages, demonstrating its advantage in balancing exploration and convergence. This outcome aligns closely with our DQN-based experiment results, indicating that EAs combined with adaptive noise is robust and adaptable across different reinforcement learning paradigms, effectively enhancing exploration capacity and robustness in complex TOD tasks.

5.8. Comparison of training signals

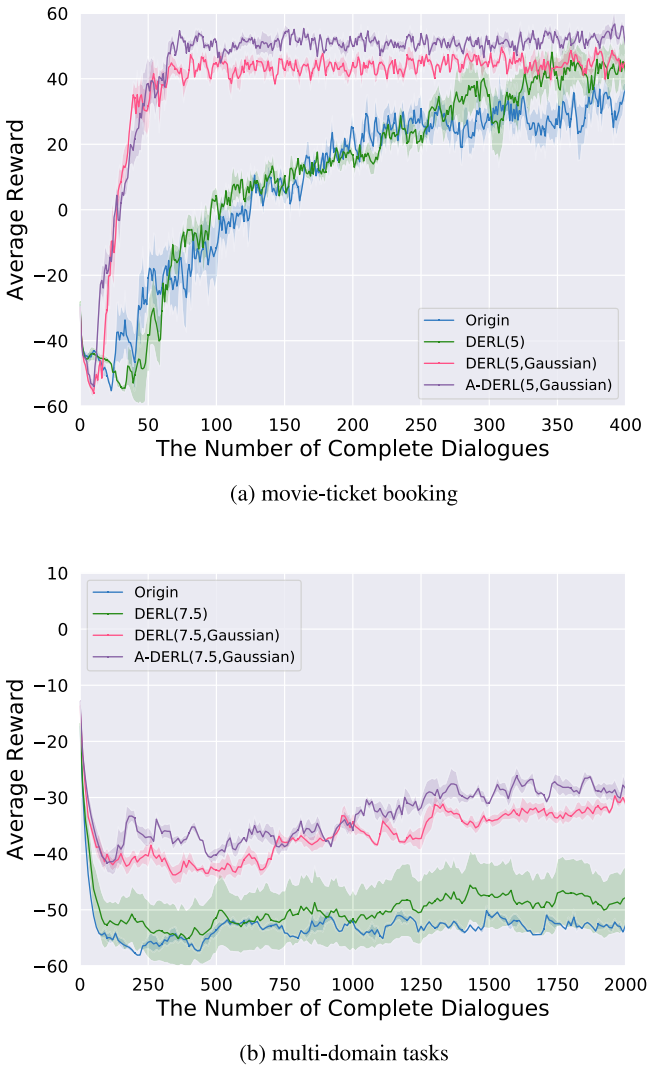
We also selected four representative methods and plotted their reward curves, as shown in Fig. 9. The experimental results indicate that in both the movie-ticket booking task and the multi-domain tasks, adaptive Gaussian noise performed best in terms of reward curves, followed by Gaussian noise, with the DQN baseline showing the worst performance. This result aligns with the previous success rate trends, demonstrating the superiority of the adaptive noise mechanism in enhancing model performance.

From the perspective of reward signals, the significant advantage of A-DERL(5, Gaussian) and DERL(5, Gaussian) can be attributed to the ability of noise-based exploration to effectively mitigate the sparse reward problem. The introduction of noise enriches the exploration signals for the reinforcement learning model, preventing it from getting stuck in suboptimal policies within the sparse reward space. Adaptive Gaussian noise further enhances this by dynamically adjusting the noise intensity, allowing the model to flexibly balance exploration and exploitation at different stages. This is particularly crucial during the later convergence phases, where adaptive noise enables more fine-tuned policy adjustments, explaining why A-DERL(5, Gaussian) ultimately outperforms DERL(5, Gaussian).

Moreover, the EAs itself provides an indirect form of supervision by periodically selecting and evolving the population, allowing the model to continuously optimize its policies across different evolutionary cycles. Although DERL(5) improves the model's adaptability through this evolutionary mechanism, exploration using only action-space noise still has limitations in long-term policy optimization. In contrast, parameter-space noise, such as Gaussian noise, offers a more structured exploration pathway. This explains why DERL(5, Gaussian) significantly outperforms the basic DERL(5) after introducing parameter-space noise. In the movie-ticket booking task, the reward curves of A-DERL(5, Gaussian) and DERL(5, Gaussian) show an entanglement in the early stages, indicating that both models engage in thorough exploration during the initial phases. This may be due to the noise strategies effectively supplementing exploration in the sparse reward environment, providing richer supervision signals. However, as training progresses, the adaptive mechanism of A-DERL(5, Gaussian) begins to take effect. By dynamically adjusting the noise scale, A-DERL(5, Gaussian) becomes more flexible in adapting to environmental changes and efficiently escapes local optima. This adjustment mechanism ensures a balance between exploration and exploitation, allowing A-DERL(5, Gaussian) to demonstrate a clear advantage in reward curves during the later stages.

In comparison, the DQN baseline relies on ϵ-greedy exploration, which becomes limited as the exploration rate decays over time. As a result, the DQN baseline struggles to escape local optima in the later stages, leading to a lagging reward curve. This highlights that the EAs, by offering more structured exploration and periodic policy updates, effectively addresses the challenges of sparse rewards and policy optimization faced by traditional DQN in dialogue tasks.

According to the PPO experiment results in Table 2, when used as a dialogue policy, PPO demonstrates stronger exploration capabilities compared to DQN, particularly showing higher sample efficiency in the optimization process. However, due to the on-policy nature of PPO, which requires frequent updates to the policy distribution, its optimization process relies more heavily on immediate reward feedback, potentially limiting long-term convergence in sparse reward environments. In this context, various EA variants can inject additional reward signals, providing a smoother and more sustained exploration mechanism. This not only enhances PPO's depth in exploring the policy space but also guides the policy toward a global optimum more rapidly, significantly improving overall performance in dialogue policies.



(a) movie-ticket booking

(b) multi-domain tasks

Fig. 9. Reward curves of the baseline method, DERL(5), DERL(5, Gaussian), and A-DERL(5, Gaussian). (a) Average dialogue reward curves of the four methods in the single-domain tasks of movie-ticket booking; (b) Average dialogue reward curves of the four methods in the multi-domain tasks.

5.9. Human evaluations

Since the experiments used a user simulator to model user interactions, there may be differences compared to real users. To address this, we invited 55 human participants for further evaluation. The evaluation was conducted using widely adopted metrics that are consistent with the dataset used in this study: success rate and rating, to assess the naturalness and task completion ability of the dialogue systems. During the human evaluation process, participants interacted with different dialogue systems without knowing which algorithm was being used at the time. If any evaluator found the dialogue to be meaningless or unnatural, they were free to terminate the interaction. Due to the memory limitations of the experimental equipment, we chose to evaluate the dialogue agents trained for 2000 episodes on the MultiWOZ dataset through human evaluation. Additionally, we selected dialogue agents trained for 400 episodes on the movie-ticket booking dataset for evaluation, all of which had achieved optimal performance. The human evaluation results in Table 3 indicate that the proposed method outperforms traditional reinforcement learning methods on both datasets, consistent with the simulation results from previous experiments.

Table 3
Human evaluation of different agents in single-domain and multi-domain tasks.

Agent		Success	Rating
MultiWOZ	Origin	0.051	1.47
	DERL(7.5)	0.042	1.22
	DERL(7.5, Gaussian)	0.075	1.59
	A-DERL(7.5, Gaussian)	**0.091**	**1.83**
movie-ticket booking	Origin	0.58	2.72
	DERL(5)	0.69	3.19
	DERL(5, Gaussian)	0.65	2.28
	A-DERL(5, Gaussian)	**0.74**	**3.66**

6. Conclusion

This paper investigates the performance of EAs in TOD policies and proposes suitable methods based on task scenarios. Additionally, an effective adaptive noise evolution method for TOD dialogue policies is introduced. Among the various EAs, evolution cycles of 5% and 7.5% achieved the best results in movie-ticket booking single-domain tasks and multi-domain tasks scenarios, respectively. In terms of noise exploration strategies, adaptive Gaussian noise yielded the best performance, while adaptive Ornstein–Uhlenbeck noise showed varying results across different task scenarios. The findings of this study provide guidance for applying EAs-based reinforcement learning to dialogue policy learning in the future. We also discovered that exploration methods effective in single-domain tasks did not significantly improve dialogue policy learning when applied to multi-domain tasks settings. Although improving reinforcement learning algorithms to adapt to specific tasks is challenging, our research reveals patterns for selecting EAs and noise strategies based on different task scenarios. Moreover, the proposed adaptive noise evolution method has demonstrated excellent performance across various dialogue tasks, highlighting the necessity of an appropriate noise scale for dialogue policy exploration in EAs. These insights and the methods presented offer valuable inspiration and guidance for future researchers.

In the future, we will further expand the population sample size to investigate the relationship between population size and exploration efficiency. Additionally, we plan to validate more reinforcement learning methods and noise strategies within the evolutionary framework, as well as further optimize the adaptive noise scaling method, to gain a deeper understanding of the exploration logic of EAs.

CRediT authorship contribution statement

Qingxin Xiao: Writing – original draft, Visualization, Validation, Project administration, Methodology, Conceptualization. **Yangyang Zhao:** Writing – review & editing, Resources. **Lingwei Dang:** Supervision. **Yun Hao:** Supervision. **Le Che:** Writing – review & editing, Resources, Formal analysis, Data curation. **Qingyao Wu:** Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by National Natural Science Foundation of China (NSFC) 62272172, Zhuhai Science and Technology Plan Project (2320004002758).

Data availability

Data will be made available on request.

References

[1] T. Young, F. Xing, V. Pandelea, J. Ni, E. Cambria, Fusing task-oriented and open-domain dialogues in conversational agents, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, (10) 2022, pp. 11622–11629.

[2] S. Young, M. Gašić, B. Thomson, J.D. Williams, Pomdp-based statistical spoken dialog systems: A review, Proc. IEEE 101 (5) (2013) 1160–1179.

[3] Z. Hu, Y. Feng, Y. Deng, Z. Li, S.-K. Ng, A.T. Luu, B. Hooi, Enhancing large language model induced task-oriented dialogue systems through look-forward motivated goals, 2023, arXiv preprint arXiv:2309.08949.

[4] A. Madotto, Z. Liu, Z. Lin, P. Fung, Language models as few-shot learner for task-oriented dialogue systems, 2020, arXiv preprint arXiv:2008.06239.

[5] S. Lappin, Assessing the strengths and weaknesses of large language models, J. Log. Lang. Inf. 33 (1) (2024) 9–20.

[6] V. Hudeček, O. Dusek, Are large language models all you need for task-oriented dialogue? in: S. Stoyanchev, S. Joty, D. Schlangen, O. Dusek, C. Kennington, M. Alikhani (Eds.), Proceedings of the 24th Annual Meeting of the Special Interest Group on Discourse and Dialogue, Association for Computational Linguistics, Prague, Czechia, 2023, pp. 216–228.

[7] M. Pecháč, M. Chovanec, I. Farkaš, Self-supervised network distillation: An effective approach to exploration in sparse reward environments, Neurocomputing (2024) 128033.

[8] H. Du, S. Li, M. Wu, X. Feng, Y.-F. Li, H. Wang, Rewarding what matters: Step-by-step reinforcement learning for task-oriented dialogue, in: Y. Al-Onaizan, M. Bansal, Y.-N. Chen (Eds.), Findings of the Association for Computational Linguistics:, EMNLP 2024, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 8030–8046.

[9] Y. Zhao, M. Dastani, J. Long, Z. Wang, S. Wang, Rescue conversations from dead-ends: Efficient exploration for task-oriented dialogue policy optimization, Trans. Assoc. Comput. Linguist. 12 (2024) 1578–1596.

[10] X. Niu, A. Ito, T. Nose, A replaceable curiosity-driven candidate agent exploration approach for task-oriented dialog policy learning, IEEE Access 12 (2024) 142640–142650.

[11] L. Wei, X. Wang, R. Zhang, Y. Cui, J. Mao, R. Jin, Exploitation and exploration in a performance based contextual advertising system: KDD'10, in: Proceedings of the 16th SIGKDD International Conference on Knowledge Discovery and Data Mining, 2010, pp. 133–138.

[12] Q. Zhu, X. Wu, Q. Lin, L. Ma, J. Li, Z. Ming, J. Chen, A survey on evolutionary reinforcement learning algorithms, Neurocomputing 556 (2023) 126628.

[13] A. Leite, M. Candadai, E.J. Izquierdo, Reinforcement learning beyond the bellman equation: Exploring critic objectives using evolution, in: Artificial Life Conference Proceedings 32, MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA, 2020, pp. 441–449, journals-info ....

[14] Z. Li, J. Kiseleva, M. de Rijke, Rethinking supervised learning and reinforcement learning in task-oriented dialogue systems, in: T. Cohn, Y. He, Y. Liu (Eds.), Findings of the Association for Computational Linguistics:, EMNLP 2020, Association for Computational Linguistics, 2020, pp. 3537–3546, Online.

[15] R. Takanobu, H. Zhu, M. Huang, Guided dialog policy learning: Reward estimation for multi-domain task-oriented dialog, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP, Association for Computational Linguistics, Hong Kong, China, 2019, pp. 100–110.

[16] S. Arora, P. Doshi, A survey of inverse reinforcement learning: Challenges, methods and progress, Artificial Intelligence 297 (2021) 103500.

[17] X. Yu, Q. Wu, K. Qian, Z. Yu, KRLS: Improving end-to-end response generation in task oriented dialog with reinforced keywords learning, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 12338–12358.

[18] D. Gupta, Y. Chandak, S. Jordan, P.S. Thomas, B. C da Silva, Behavior alignment via reward function optimization, Adv. Neural Inf. Process. Syst. 36 (2023) 52759–52791.

[19] R. Takanobu, R. Liang, M. Huang, Multi-agent task-oriented dialog policy learning with role-aware reward decomposition, in: D. Jurafsky, J. Chai, N. Schluter, J. Tetreault (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, 2020, pp. 625–638, Online.

[20] Z. Tang, H. Kulkarni, G.H. Yang, High-quality dialogue diversification by intermittent short extension ensembles, in: C. Zong, F. Xia, W. Li, R. Navigli (Eds.), Findings of the Association for Computational Linguistics:, ACL-IJCNLP 2021, Association for Computational Linguistics, 2021, pp. 1861–1872, Online.

[21] D.E. Moriarty, A.C. Schultz, J.J. Grefenstette, Evolutionary algorithms for reinforcement learning, J. Artificial Intelligence Res. 11 (1999) 241–276.

[22] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, et al., Scalable deep reinforcement learning for vision-based robotic manipulation, in: Conference on Robot Learning, PMLR, 2018, pp. 651–673.

[23] M. Jaderberg, V. Dalibard, S. Osindero, W.M. Czarnecki, J. Donahue, A. Razavi, O. Vinyals, T. Green, I. Dunning, K. Simonyan, et al., Population based training of neural networks, 2017, arXiv preprint arXiv:1711.09846.

[24] D. Hakkani-Tür, G. Tür, A. Celikyilmaz, Y.-N. Chen, J. Gao, L. Deng, Y.-Y. Wang, Multi-domain joint semantic frame parsing using bi-directional rnn-lstm., in: Interspeech, 2016, pp. 715–719.

[25] K. Mishra, M. Firdaus, A. Ekbal, Please be polite: Towards building a politeness adaptive dialogue system for goal-oriented conversations, Neurocomputing 494 (2022) 242–254.

[26] N. Mrkšić, D. Ó Séaghdha, T.-H. Wen, B. Thomson, S. Young, Neural belief tracker: Data-driven dialogue state tracking, in: R. Barzilay, M.-Y. Kan (Eds.), Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 1777–1788.

[27] V. Mnih, K. Kavukcuoglu, D. Silver, A.A. Rusu, J. Veness, M.G. Bellemare, A. Graves, M. Riedmiller, A.K. Fidjeland, G. Ostrovski, et al., Human-level control through deep reinforcement learning, Nature 518 (7540) (2015) 529–533.

[28] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, Proximal policy optimization algorithms, 2017, arXiv preprint arXiv:1707.06347.

[29] L. Matějů, D. Griol, Z. Callejas, J.M. Molina, A. Sanchis, An empirical assessment of deep learning approaches to task-oriented dialog management, Neurocomputing 439 (2021) 327–339.

[30] R. Zhang, Z. Wang, M. Zheng, Y. Zhao, Z. Huang, Emotion-sensitive deep dyna-q learning for task-completion dialogue policy learning, Neurocomputing 459 (2021) 122–130.

[31] H. Zhu, X. Wang, Z. Wang, K. Xv, An emotion-sensitive dialogue policy for task-oriented dialogue system, Sci. Rep. 14 (1) (2024) 19759.

[32] H. Wang, Y. Zhang, Y. Yang, Y. Zheng, K.-F. Wong, Acquiring new knowledge without losing old ones for effective continual dialogue policy learning, IEEE Trans. Knowl. Data Eng. (2023).

[33] Z. Zhou, Z. Liu, Z. Dong, Y. Liu, Model discrepancy policy optimization for task-oriented dialogue, Comput. Speech Lang. 87 (2024) 101636.

[34] W.-C. Kwan, H.-R. Wang, H.-M. Wang, K.-F. Wong, A survey on recent advances and challenges in reinforcement learning methods for task-oriented dialogue policy learning, Mach. Intell. Res. 20 (3) (2023) 318–334.

[35] M. Plappert, R. Houthooft, P. Dhariwal, S. Sidor, R.Y. Chen, X. Chen, T. Asfour, P. Abbeel, M. Andrychowicz, Parameter space noise for exploration, 2017, arXiv preprint arXiv:1706.01905.

[36] Y. Zhao, Z. Wang, K. Yin, R. Zhang, Z. Huang, P. Wang, Dynamic reward-based dueling deep dyna-q: Robust policy learning in noisy environments, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, (05) 2020, pp. 9676–9684.

[37] P. Budzianowski, T.-H. Wen, B.-H. Tseng, I. Casanueva, S. Ultes, O. Ramadan, M. Gašić, MultiWOZ - a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling, in: E. Riloff, D. Chiang, J. Hockenmaier, J. Tsujii (Eds.), Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 5016–5026.

[38] Z. Ding, Z. Yang, Y. Qiao, H. Lin, KMc-ToD: Structure knowledge enhanced multi-copy network for task-oriented dialogue system, Knowl.-Based Syst. 293 (2024) 111662.

[39] Q. Zhu, Z. Zhang, Y. Fang, X. Li, R. Takanobu, J. Li, B. Peng, J. Gao, X. Zhu, M. Huang, ConvLab-2: An open-source toolkit for building, evaluating, and diagnosing dialogue systems, in: A. Celikyilmaz, T.-H. Wen (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Association for Computational Linguistics, 2020, pp. 142–149, Online.

[40] G. Weisz, P. Budzianowski, P.-H. Su, M. Gašić, Sample efficient deep reinforcement learning for dialogue systems with large action spaces, IEEE/ ACM Trans. Audio Speech Lang. Process. 26 (11) (2018) 2083–2097.

[41] T. Cordier, T. Urvoy, F. Lefèvre, L.M. Rojas Barahona, Graph neural network policies and imitation learning for multi-domain task-oriented dialogues, in: O. Lemon, D. Hakkani-Tur, J.J. Li, A. Ashrafzadeh, D.H. Garcia, M. Alikhani, D. Vandyke, O.r. Du˘ sek (Eds.), Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue, Association for Computational Linguistics, Edinburgh, UK, 2022, pp. 91–100.

[42] I. Casanueva, P. Budzianowski, P.-H. Su, N. Mrkšić, T.-H. Wen, S. Ultes, L. Rojas-Barahona, S. Young, M. Gašić, A benchmarking environment for reinforcement learning based task oriented dialogue management, 2017, arXiv preprint arXiv:1711.11023.

[43] P.-H. Su, P. Budzianowski, S. Ultes, M. Gašić, S. Young, Sample-efficient actor-critic reinforcement learning with supervised data for dialogue management, in: K. Jokinen, M. Stede, D. DeVault, A. Louis (Eds.), Proceedings of the 18th Annual SIGDial Meeting on Discourse and Dialogue, Association for Computational Linguistics, Saarbrücken, Germany, 2017, pp. 147–157.

[44] T. Cordier, T. Urvoy, F. Lefèvre, L.M. Rojas Barahona, Few-shot structured policy learning for multi-domain and multi-task dialogues, in: A. Vlachos, I. Augenstein (Eds.), Findings of the Association for Computational Linguistics:, EACL 2023, Association for Computational Linguistics, Dubrovnik, Croatia, 2023, pp. 432–441.
[45] G.E. Uhlenbeck, L.S. Ornstein, On the theory of the Brownian motion, Phys. Rev. 36 (5) (1930) 823.
[46] F. Torabi, G. Warnell, P. Stone, Behavioral cloning from observation, in: Proceedings of the 27th International Joint Conference on Artificial Intelligence, IJCAI '18, AAAI Press, 2018, pp. 4950–4957.

Qingxin Xiao received the B.S. degree in Computer Science and Technology from Northeast Forestry University in 2021. Subsequently, he obtained the M.S. degree in Computer Science and Technology from Northeast Forestry University in 2024. He is currently a Ph.D. student in the Department of Electronic Information at South China University of Technology, Guangzhou. His research interests include task-oriented dialogue, Large Language Model and agent security.

Yangyang Zhao received a B.S. degree from Guangzhou University of Chinese Medicine, Guangzhou, China, in 2017, and a Ph.D. degree from South China University of Technology, Guangzhou, China in 2022. She is currently an Lecture at the Department of Computer and Communication Engineering, Changsha University of Science and Technology, China. Her research interests include Large Language Model, Deep Reinforcement Learning, Dialogue Systems, and Dialogue Policy Learning.

Lingwei Dang received the B.S. degree in the School of Computer Engineering and Science from Shanghai University in 2020. Subsequently, he obtained the M.S. degree in the School of Computer Science and Engineering from South China University of Technology. Currently, he is a Ph.D. student in the School of Software Engineering at South China University of Technology, Guangzhou. His research interests cover embodied AI, reinforcement learning, and deep learning.

Yun Hao is currently pursuing a Ph.D. degree in Software Engineering from the School of Software Engineering at South China University of Technology, China. His current research interests are deep learning, computer vision and 3D recognition.

Le Che received the B.S. degree in architecture and urban planning from Tongji University, China, in 2002, the M.S. degree in architecture and urban planning from Tongji University, China, in 2006, and the Ph.D. degree in architecture and urban planning from Tongji University, China, in 2012. She is currently an Associate Professor with the Department of Urban Planning, South China University of Technology. Her current research interests include sustainable urban-rural development, smart cities, healthy communities, and development control.

Qingyao Wu received the B.S. degree in software engineering from the South China University of Technology, China, in 2007, and the Ph.D. degree in computer science from the Harbin Institute of Technology, China, in 2013. He is currently a Professor with the School of Software Engineering, South China University of Technology. His current research interests include computer vision and data mining.