LLM-Driven Policy Diffusion: Enhancing Generalization in Offline Reinforcement Learning

Hanping Zhang[1], **Yuhong Guo**[1,2]
[1]School of Computer Science, Carleton University, Ottawa, Canada
[2]Canada CIFAR AI Chair, Amii, Canada
jagzhang@cmail.carleton.ca, yuhong.guo@carleton.ca

Abstract

Reinforcement Learning (RL) is known for its strong decision-making capabilities and has been widely applied in various real-world scenarios. However, with the increasing availability of offline datasets and the lack of well-designed online environments from human experts, the challenge of generalization in offline RL has become more prominent. Due to the limitations of offline data, RL agents trained solely on collected experiences often struggle to generalize to new tasks or environments. To address this challenge, we propose LLM-Driven Policy Diffusion (LLMDPD), a novel approach that enhances generalization in offline RL using task-specific prompts. Our method incorporates both text-based task descriptions and trajectory prompts to guide policy learning. We leverage a large language model (LLM) to process text-based prompts, utilizing its natural language understanding and extensive knowledge base to provide rich task-relevant context. Simultaneously, we encode trajectory prompts using a transformer model, capturing structured behavioral patterns within the underlying transition dynamics. These prompts serve as conditional inputs to a context-aware policy-level diffusion model, enabling the RL agent to generalize effectively to unseen tasks. Our experimental results demonstrate that LLMDPD outperforms state-of-the-art offline RL methods on unseen tasks, highlighting its effectiveness in improving generalization and adaptability in diverse settings.

1 Introduction

Reinforcement Learning (RL) has emerged as a powerful paradigm for sequential decision-making and control, achieving remarkable success across a wide range of applications, from robotics (Tang et al., 2024) and autonomous driving (Lee et al., 2024) to finance (Liu et al., 2022) and healthcare (Yu et al., 2021). With the growing application of RL in real-world scenarios, the limitations of real-world RL environments have become increasingly evident—in many cases, access to either data or the environment is restricted. This highlights the importance of generalization in RL, which focuses on training RL agents on limited datasets or environments while ensuring their generalizability to unseen tasks or environments (Kirk et al., 2023), thereby reducing the need for extensive task-specific training and direct access to all possible environments. This not only enhances efficiency but also significantly reduces human effort in designing and collecting training data.

Generalization in RL however poses significant challenges, as it requires maximizing the performance of RL agents on unseen tasks or environments that were not covered in the training data. Prior research has identified several factors contributing to the generalization gap in deep RL agents, such as overfitting and memorization in deep neural networks, which can lead to poor adaptability (Arpit et al., 2017). To address these challenges, various methods have been explored, including data augmentation to enhance generalization to unseen states (Yarats et al., 2021; Raileanu et al., 2021;

2nd Workshop on Aligning Reinforcement Learning Experimentalists and Theorists (ARLET 2025).

Zhang and Guo, 2021), generation of synthetic environments to increase training diversity (Wang et al., 2019, 2020), approaches to reduce discrepancies between different environments (Liu et al., 2020), and optimization strategies that account for environment variations (Raileanu and Fergus, 2021). However, in many real-world applications, direct access to online environments is often impractical. For example, in autonomous driving (Lee et al., 2024), training data is primarily collected from human drivers, resulting in large offline datasets rather than interactive online environments. This makes generalization in offline RL an especially important area of study. The goal is to train RL agents from offline data that can achieve strong performance while generalizing to unseen circumstances. Prior research has shown that generalization in offline RL is more challenging than in online RL (Mazoure et al., 2022; Mediratta et al., 2024), highlighting both its significance and inherent difficulties.

The main challenges of generalization in offline RL fall into two primary categories: (1) the generalization gap inherited from standard deep RL (Arpit et al., 2017), and (2) the lack of sufficient exploration data in offline RL training. Previous works have employed techniques such as data augmentation (Laskin et al., 2020; Sinha et al., 2022; Modhe et al., 2023) and adversarial training (Qiao and Yang, 2024) to address the generalization gap caused by overfitting. To improve training efficiency in offline RL—where arbitrary exploration like in online RL is not possible—researchers have sought to enhance data utilization (He et al., 2023) and mitigate the effects of out-of-distribution data, thereby improving generalization (Ma et al., 2024; Wang et al., 2024a). However, most prior work has primarily focused on reducing reliance on training data to enhance generalization. Few studies have leveraged readily available task-specific information such as text descriptions from offline data or incorporated easily collectible task-related data to enhance generalization.

In this work, we introduce LLM-Driven Policy Diffusion (LLMDPD), a novel approach to improving generalization in offline RL by leveraging task-specific prompts. We introduce two types of prompts: a text prompt, which is a textual description of the task or environment, and a trajectory prompt, which consists of a single trajectory collected from the target task or environment, both of which are easy and cheap to obtain. First, leveraging the capabilities of large language models (LLMs) in natural language processing (Qin et al., 2024) and knowledge distillation (Xu et al., 2024; Yang et al., 2024), we use a pre-trained LLM to process the text prompt, extracting useful insights from the task description while also drawing on the pre-collected knowledge embedded in the LLM. Second, we train a transformer model to process the trajectory prompt, capturing task-specific behavioral patterns from the transition dynamics of the prompt. Both prompts are encoded into latent embeddings, which serve as conditional inputs to support adaptive and context-aware policy training. We adopt policy diffusion as our policy function, which takes the state and the task-specific prompt embeddings as inputs and outputs a task-aware action distribution, enabling generalization to unseen tasks without fine-tuning. We evaluate LLMDPD on the Meta-World (Yu et al., 2020)benchmark, and our experimental results show that LLMDPD outperforms state-of-the-art methods in offline RL generalization, demonstrating the effectiveness of our approach.

Our key contributions are summarized as follows:

- We are the first to incorporate text prompts leveraging the advantages of LLMs, along with trajectory prompts, into offline RL. This approach significantly enhances the generalizability of offline RL agents to unseen tasks.
- We introduce Context-aware Conditional Policy Diffusion (CCPD), which incorporates prompt information as a conditioning mechanism within the offline RL framework. This approach enhances policy flexibility while improving generalization to unseen tasks.
- Our experimental results show that LLMDPD significantly outperforms state-of-the-art offline RL methods in generalization, demonstrating the effectiveness of our approach.

2 Related Works

Generalization in Offline RL Generalization in RL focuses on addressing the generalization gap to enhance RL agent's ability to perform well on unseen tasks or environments. Traditional generalization studies in RL primarily examine the generalizability of RL agents in online environments. However, with the increasing availability of large offline datasets and the lack of direct access to online environments in many real applications, present research has shifted toward a more challenging yet practical objective: improving generalization in offline RL. Mazoure et al. (2022) systematically analyzed the differences in generalization between online and offline RL, providing

theoretical evidence that online RL algorithms struggle to generalize in offline settings. Mediratta et al. (2024) conducted experiments evaluating the generalization capabilities of widely used RL methods in both online and offline settings. Their results indicate that standard RL methods generalize more poorly in offline environments, reinforcing that generalization in offline RL is a more difficult problem. Laskin et al. (2020) and Sinha et al. (2022) introduced data augmentation schemes to enhance the generalization ability of offline RL agents. He et al. (2023) proposed the Multi-Task Diffusion Model (MTDIFF), which leverages knowledge from multi-task data to improve generalization in offline RL through shared information. Modhe et al. (2023) proposed an unseen state augmentation method to improve both generalization and value estimation for unseen states. Qiao and Yang (2024) introduced Soft Adversarial Offline Reinforcement Learning (SAORL), which imposes constraints on traditional adversarial examples, formulating a worst-case optimization problem to generate soft adversarial examples. Zhao et al. (2024) proposed Offline Trajectory Generalization through World Transformers for Offline Reinforcement Learning (OTTO), a method designed to learn state dynamics and reward functions, thereby enhancing generalization to unseen states. Ma et al. (2024) developed Representation Distinction (RD), a plugin method that improves offline RL generalization by detecting and preventing out-of-distribution state-action pairs. Similarly, Wang et al. (2024a) introduced Adversarial Data Splitting (ADS) to relax rigid out-of-distribution boundaries, ultimately improving generalization in offline RL. Xu et al. (2022) introduced Prompt-based Decision Transformer (Prompt-DT), which adopts a prompt framework to enable few-shot adaptation in offline RL, thereby facilitating few-shot policy generalization. Li et al. (2024) and Wang et al. (2024b) incorporate contextual information into offline meta-RL, offering insights into how context can be utilized to improve task conditioning or policy adaptation. Li et al. (2024) introduced the UNICORN, which provides a theoretical perspective on the importance of task variables and their latent representations by incorporating contextual information. Wang et al. (2024b) proposed the Meta Decision Transformer (Meta-DT), which learns a context-aware world model to encode task-specific information. While their work does not explicitly focus on generalization, it provides a valuable perspective on leveraging contextual cues within offline RL.

Diffusion-based RL Diffusion models have recently emerged as a powerful generative modeling approach for capturing complex data distributions, and their application to RL has gained traction. Diffuser (Janner et al., 2022) introduces the concept of using diffusion models to model trajectory distributions in offline RL. Several diffusion-based approaches have since extended this idea. Decision Diffuser (Ajay et al., 2022) conditions trajectory generation on high-level task information, such as rewards. PlanDiffuser (Sharan et al., 2024) integrates diffusion models with planning techniques to enhance precision in control tasks. MetaDiffuser (Ni et al., 2023) learns a contextual representation of tasks as conditional input to the diffusion model, enabling the generation of task-oriented trajectories. Similarly, Hierarchical Diffuser (Chen et al., 2024) decomposes long planning horizons into smaller segments, learning subgoals for each to improve planning efficiency. In addition to trajectory-based diffusion models, Wang et al. (2023) introduced Diffusion Policy to offline RL at the action level rather than the trajectory level, providing greater flexibility and more accurate transitions. Chi et al. (2023) further extended Diffusion Policy to broader RL scenarios, enabling effective policy learning for high-dimensional control tasks. Our proposed work is the first that exploits diffusion policy for generalization in offline RL.

Applications of LLM in RL Large Language Models (LLMs) have demonstrated remarkable capabilities in understanding, generating, and reasoning over text, making them powerful tools for a wide range of applications. Recently, many studies have started exploring the integration of LLMs into RL to enhance learning efficiency and decision-making. Sun et al. (2024) surveyed the application of LLMs in multi-agent RL frameworks, highlighting key advancements and future research directions. Cao et al. (2024) reviewed the existing literature on LLM-enhanced RL, summarizing advancements and challenges in the field. Du et al. (2023) introduced Exploring with LLMs (ELLM), a method that guides RL pretraining by generating prompted descriptions of the agent's current state. Wang et al. (2024c) proposed LLM-Empowered State Representation (LESR), which leverages LLMs to generate task-relevant state representations, thereby improving the training efficiency of standard RL methods. More recently, Yan et al. (2025) treated LLMs as prior action distributions, integrating them into RL frameworks by using Bayesian inference methods to enhance the sample efficiency of traditional RL algorithms. Shi et al. (2024) introduced Language Models for Motion Control (LaMo), a framework that leverages pretrained language models to enhance Decision Transformers (Chen et al., 2021) and improve the quality of policies learned from offline RL. Yang and Xu (2024) proposed the Language

```
Task: Meta-World push.
Objective: Push the puck to a goal.
Constraints: Randomize puck and goal positions.
...
```

Figure 1: An example of a text prompt, where the description is structured as a short instruction guiding the LLM on how to refine the prompt.

Model-initialized Prompt Decision Transformer (LPDT), which builds upon the standard Prompt-DT framework by replacing the learned prompt embeddings with those derived from a pretrained language model. They further fine-tune the model using Low-Rank Adaptation (LoRA) (Hu et al., 2022) to improve performance over Prompt-DT. Although LLMs have been applied across various domains of RL, their potential for improving generalization in RL remains largely unexplored. Our proposed work is the first to exploit the capacity of LLMs to enhance generalization in offline RL.

3 Method

Problem setting In generalization learning for offline RL, we assume an offline dataset $\mathcal{D} = \{(\mathcal{D}^i, z^i_{\text{text}}, z^i_\tau)\}_{i=1}^m$ that contains data for m seen tasks is given, where the training data $\mathcal{D}^i$ for each task i contains a collection of offline trajectory instances, paired with two additional prompts: a text prompt z^i_{text} and a trajectory instance prompt z^i_τ. The text prompt provides a textual description for the corresponding task, while the trajectory prompt consists of a *single* trajectory collected from the given task using a behavior policy. Both prompts are easily collectible for either training or test tasks and can provide task-specific context information for the RL model, thereby supporting generalization to unseen tasks. Our goal is to learn an optimal policy $\pi^\star$ from the offline dataset $\mathcal{D}$ over multiple seen tasks such that it can generalize effectively to unseen tasks without fine-tuning, guided by the text and trajectory prompts from the target unseen tasks.

In this section, we present LLM-Driven Policy Diffusion (LLMDPD), a novel approach to enhancing generalization in offline RL. LLMDPD leverages both text and trajectory prompts for adaptive policy diffusion learning. We utilize a pre-trained large language model (LLM) and a parametric transformer as the embedding module to encode the text prompt and trajectory prompt, respectively. The resulting prompt embeddings capture task specific information and serve as conditional inputs for context-aware policy diffusion. We use diffusion Q-learning to jointly train the prompt embedding module and the policy diffusion module in an end-to-end manner, inducing prompt-based policy functions with enhanced adaptability and generalizability. The approach is elaborated below.

3.1 Prompt Embedding

3.1.1 LLM-Driven Text Prompt Embedding

The text prompt z_{text} for each task consists a natural language description that provides explicit information for the corresponding task. It can be utilized to extract high-level semantic representations of the task, supporting subsequent learning. To facilitate information extraction, we convert the text descriptions of the tasks into a structured format for expressing information of various components, including the task name, objective, constraints, and other specific attributes. An example of the structured text prompt z_{text} is provided in Figure 1.

Next we utilize a pre-trained large language model (LLM), denoted as $\mathcal{M}$, to produce a latent prompt embedding $\boldsymbol{z}_{\text{text}}$ from the structure text prompt z_{text}. By harnessing LLMs' ability to process natural language texts and leveraging knowledge distillation from their embedded prior knowledge, the latent prompt embedding is expected to encode rich task-relevant information. To enhance the efficiency of prompt interpretation, we also include a default brief instruction, e.g., "convert the following task description into a structured policy representation", to guide the LLM in processing the text prompt. Specifically, by using the structured text prompt together with the interpretation instruction as input, we obtain an embedding output by performing mean pooling to the token embeddings produced from the last hidden layer of the LLM, effectively capturing the overall context to ensure a comprehensive

representation of the processed prompt. To enable adaptation to the subsequent policy learning task, we further introduce a multilayer perceptron (MLP) project head h_ψ parameterized by ψ on top of the LLM $\mathcal{M}$, refining the embedding output to a final embedding vector $\boldsymbol{z}_{\text{text}}$. This embedding process can be expressed using the following equation:

$$\boldsymbol{z}_{\text{text}} = h_\psi(\mathcal{M}(z_{\text{text}})). \tag{1}$$

Here the default interpretation instruction is omitted for simplicity. The parametric projection head can be trained end-to-end within the overall policy learning framework.

3.1.2 Transformer-Driven Trajectory Prompt Embedding

The trajectory instance prompt consists of a single trajectory collected from the corresponding task, represented as a sequence of state-action transitions, such as

$$z_\tau = [\boldsymbol{s}_0, \boldsymbol{a}_0, \boldsymbol{s}_1, \boldsymbol{a}_1, \cdots, \boldsymbol{s}_t, \boldsymbol{a}_t, \cdots, \boldsymbol{s}_T, \boldsymbol{a}_T] \tag{2}$$

where $\boldsymbol{s}_t$ and $\boldsymbol{a}_t$ denote the state and action at timestep t respectively, and T denotes the length of the trajectory prompt. Unlike text prompts which explicitly provide task-specific descriptions, a trajectory prompt captures the transition dynamics and behavior patterns of the environment for the corresponding task. To generate informative embeddings from the trajectory prompts, we devise a parametric transformer as the encoder for the trajectory prompts, leveraging Transformer's ability for capturing long-range dependencies in sequential data and supporting effective structural information extraction (Vaswani et al., 2017). Similar to text prompt embedding, we apply mean pooling over the transformer's output and deploy a parametric MLP projection head on top of it. We use a function g_φ, parameterized by φ, to denote the overall trajectory prompt encoder that includes both the transformer and the MLP projection head. The final prompt embedding $\boldsymbol{z}_\tau$ for a trajectory prompt z_τ can be produced from the encoder g_φ as follows:

$$\boldsymbol{z}_\tau^i = g_\varphi(z_\tau). \tag{3}$$

This transformer based encoder g_φ can be trained in an end-to-end manner through back-propagation within the policy learning framework, ensuring that the prompt embedding effectively supports the learning of a context-aware adaptive policy.

3.2 Context-Aware Conditional Policy Diffusion

To effectively leverage prompt embeddings that encode rich task-specific information to support generalizable policy learning from offline data, we propose to learn a context-aware conditional policy diffusion (CCPD) module as our policy function: $\pi_\theta(\boldsymbol{a}|\boldsymbol{s}, \boldsymbol{z}_{\text{text}}, \boldsymbol{z}_\tau)$, where θ denotes the parameters of the module. This function conditions the policy generation on task-specific contexts encoded by the text prompt embedding $\boldsymbol{z}_{\text{text}}$ and trajectory prompt embedding $\boldsymbol{z}_\tau$.

The diffusion process of the CCPD module consists of two Markov Chain processes: a forward process and a reverse process. The forward process incrementally adds noise to an action $\boldsymbol{a}^0$ sampled from offline data, transforming it into a Gaussian prior over K diffusion steps. In the reverse process, starting from a Gaussian noise prior $\boldsymbol{a}^K \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$, the model progressively denoises the action at each timestep k, conditioned on the given state $\boldsymbol{s}$ and the corresponding prompt embeddings $\boldsymbol{z}_{\text{text}}$ and $\boldsymbol{z}_\tau$. Specifically, at timestep k, the next action $\boldsymbol{a}^{k-1}$ in the sequential denoising process is generated from the following Gaussian distribution:

$$p_\theta(\boldsymbol{a}^{k-1}|\boldsymbol{a}^k, \boldsymbol{s}, \boldsymbol{z}_{\text{text}}, \boldsymbol{z}_\tau) = \mathcal{N}(\boldsymbol{a}^{k-1}; \mu_\theta(\boldsymbol{a}^k, \boldsymbol{s}, \boldsymbol{z}_{\text{text}}, \boldsymbol{z}_\tau, k), \sigma_k^2 \mathbf{I}) \tag{4}$$

$$\text{with } \mu_\theta(\boldsymbol{a}^k, \boldsymbol{s}, \boldsymbol{z}_{\text{text}}, \boldsymbol{z}_\tau, k) = \frac{1}{\sqrt{\alpha_k}} \left(\boldsymbol{a}^k - \frac{1-\alpha_k}{\sqrt{1-\bar{\alpha}_k}} \epsilon_\theta(\boldsymbol{a}^k, \boldsymbol{s}, \boldsymbol{z}_{\text{text}}, \boldsymbol{z}_\tau, k) \right)$$

Here, $\bar{\alpha}_k = \prod_{i=1}^k \alpha_i$, and α_k follows a predefined variance schedule (Ho et al., 2020); ϵ_θ denotes a learned noise prediction network which estimates the added noise at each diffusion step, allowing the model to recover the clean action after K timesteps. This CCPD module is trained together with the prompt embedding module on the offline data $\mathcal{D}$ by minimizing a diffusion loss $\mathcal{L}_d(\psi, \varphi, \theta)$, defined as the mean squared error (MSE) between the true noise $\boldsymbol{\epsilon}$ and the predicted noise:

$$\mathcal{L}_d(\psi, \varphi, \theta) = \mathbb{E}_{\mathcal{C}} \left[\left\| \boldsymbol{\epsilon} - \epsilon_\theta \left(\sqrt{\bar{\alpha}_k} \boldsymbol{a} + \sqrt{1-\bar{\alpha}_k} \boldsymbol{\epsilon}, \boldsymbol{s}, h_\psi(\mathcal{M}(z_{\text{text}}^i)), g_\varphi(z_\tau^i), k \right) \right\|^2 \right] \tag{5}$$

$$\text{where } \mathcal{C} = \{i \sim [1:m], (\boldsymbol{s}, \boldsymbol{a}) \sim \mathcal{D}^i, k \sim [1:K], \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})\}.$$

Given a trained policy diffusion module, the policy function π_θ is obtained by progressively denoising from a Gaussian prior, following the reverse diffusion process indicated by Eq.(4).

Incorporating Reward Maximization via Actor-Critic Policy Diffusion Minimizing only the diffusion loss $\mathcal{L}_d(\psi, \varphi, \theta)$ results in a behavior-cloned policy, which mimics the offline dataset $\mathcal{D}$ without optimizing for rewards. To address this problem, we introduce a Q-function $Q_\phi(\boldsymbol{s}, \boldsymbol{a})$ to estimate the expected cumulative reward. Specifically, we deploy the double Q-Learning strategy (Hasselt, 2010) that uses two Q-networks, Q_{ϕ_1} and Q_{ϕ_2}, to prevent Q-value overestimation, which are trained by minimizing the following Q-losses (Wang et al., 2023):

$$\mathcal{L}_q(\phi_\ell) = \mathbb{E}_{\mathcal{C}_q}\left[\left\|\left(r_t + \gamma \min_{\ell'=1,2} Q_{\bar{\phi}_{\ell'}}(\boldsymbol{s}_{t+1}, \boldsymbol{a}^0_{t+1})\right) - Q_{\phi_\ell}(\boldsymbol{s}_t, \boldsymbol{a}_t)\right\|^2\right], \quad \text{for } \ell \in \{1, 2\} \tag{6}$$

$$\text{where } \mathcal{C}_q = \{i \sim [1:m], (\boldsymbol{s}_t, \boldsymbol{a}_t, r_t, \boldsymbol{s}_{t+1}) \sim \mathcal{D}^i, \boldsymbol{a}^0_{t+1} \sim \pi_\theta(\cdot|\boldsymbol{s}_{t+1}, \boldsymbol{z}^i_{\text{text}}, \boldsymbol{z}^i_\tau)\}$$

where $\bar{\phi}_{\ell'}$ indicates the stopping-gradient network used for Q-value estimation, and r_t denotes the reward observed in the trajectories of the offline dataset. The two Q-networks will converge to the same solution ϕ in the limit. We utilize the Q-values estimated by either one of the Q-networks ($\phi = \phi_1$ or $\phi = \phi_2$) to guide the reverse policy diffusion process, generating actions that maximize the expected cumulative reward:

$$\mathcal{L}_r(\psi, \varphi, \theta) = \mathbb{E}_{i\sim[1:m], \boldsymbol{s}\sim\mathcal{D}^i, \boldsymbol{a}^0\sim\pi_\theta(\cdot|\boldsymbol{s}, h_\psi(\mathcal{M}(z^i_{\text{text}})), g_\varphi(z^i_\tau))}\left[Q_\phi(\boldsymbol{s}, \boldsymbol{a}^0)\right] \tag{7}$$

To balance behavior cloning and reward maximization, the total training loss for the policy diffusion module is formulated as a weighted combination of $\mathcal{L}_d$ and the negation of the reward objective $\mathcal{L}_r$:

$$\mathcal{L}(\psi, \varphi, \theta) = \mathcal{L}_d(\psi, \varphi, \theta) - \lambda \mathcal{L}_r(\psi, \varphi, \theta) \tag{8}$$

where λ is a hyperparameter controlling the trade-off between action denoising and reward-driven optimization. The policy diffusion module can be viewed as an actor and the Q-networks can be treated as critics. They can be simultaneously learned using the actor-critic learning strategy. The overall actor-critic diffusion training algorithm is illustrated in Algorithm 1. By combining policy diffusion with Q-learning, our LLMDPD model learns a generalizable and reward-maximizing policy, capable of adapting to unseen tasks under the guidance of task-aware prompts.

Algorithm 1 LLMDPD Training

Input: offline dataset $\mathcal{D} = \{\mathcal{D}^i\}_{i=1}^m$, initialized embedding module (ψ, φ) with pre-trained LLM $\mathcal{M}$, initialized policy diffusion module θ, initialized Q-networks Q_{ϕ_1} and Q_{ϕ_2}
Output: Trained model parameters ψ, φ, θ, ϕ_1, ϕ_2.

```
for each epoch do
    Sample a task i ~ [1 : m].
    Extract text prompt z^i_text and trajectory prompt z^i_τ for task i.
    Sample a batch B = {(s_t, a_t, r_t, s_{t+1}), ···} from seen offline data D^i.
    for each transition (s_t, a_t, r_t, s_{t+1}) in batch B do
        Compute prompt embeddings: z^i_text = h_ψ(M(z^i_text)) and z^i_τ = g_φ(z^i_τ).
        Sample action: a^0_{t+1} ~ π_θ(·|s_{t+1}, z^i_text, z^i_τ).
        Update the Q-networks φ_1, φ_2 by minimizing the Q-loss in Eq.(6)
        Randomly select φ_1 or φ_2 as the critic φ
        Sample action: a^0_t ~ π_θ(·|s_t, z^i_text, z^i_τ).
        Sample diffusion timestep k ~ [1 : K] and noise ε ~ N(0, I).
        Update parameters ψ, φ, θ by minimizing the total loss in Eq.(8).
    end for
end for
```

4 Experiments

To thoroughly evaluate the generalization performance of our LLMDPD method, we conducted experiments on the Meta-World dataset (Yu et al., 2020), which serves as a benchmark for evaluation an RL agent's generalizability to unseen tasks.

4.1 Experimental Settings

Environment Meta-World (Yu et al., 2020) is a widely used benchmark designed for multi-task and meta-RL. It is implemented using the MuJoCo physics engine (Todorov et al., 2012), which provides

Table 1: This table presents the average success rates for various comparison methods on Meta-World-V2 tasks, evaluated over 500 episodes per task. Results are averaged over three runs.

Type	Task	SAC	S4RL	RAD	MTDIFF	LLMDPD
Unseen	box-close	23.46 ± 7.11	73.13 ± 3.51	71.20 ± 4.84	65.73 ± 8.36	$\mathbf{75.24 \pm 4.90}$
	hand-insert	30.60 ± 9.77	60.20 ± 1.57	43.79 ± 3.44	70.87 ± 3.59	$\mathbf{78.71 \pm 5.33}$
	bin-picking	42.13 ± 14.33	72.20 ± 4.17	43.27 ± 4.38	55.73 ± 7.63	$\mathbf{74.65 \pm 6.28}$
Seen	sweep-into	91.80 ± 1.14	90.53 ± 3.52	88.06 ± 9.86	$\mathbf{92.87 \pm 1.11}$	91.84 ± 2.42
	coffee-push	28.60 ± 14.55	28.73 ± 8.44	33.19 ± 2.86	74.67 ± 6.79	$\mathbf{76.56 \pm 2.89}$
	disassemble	60.20 ± 16.29	52.20 ± 5.68	60.93 ± 20.80	69.00 ± 4.72	$\mathbf{72.30 \pm 6.48}$

a diverse set of near-realistic robotic manipulation tasks, such as picking, pushing, and reaching. A notable subvariant, Multi-Task 50 (MT50), consists of 50 robotic manipulation tasks well-suited for offline data collection, with each task accompanied by a detailed description. Among these 50 tasks, 45 are designated as training tasks, while the remaining 5 serve as unseen test tasks to evaluate an RL agent's generalizability to novel tasks. With its pre-collected offline data and comprehensive task descriptions, Meta-World serves as an ideal testbed for our LLMDPD method.

Comparison Methods We compare our LLMDPD method against four baselines on the Meta-World benchmark: SAC, S4RL, RAD, and MTDIFF. SAC (Haarnoja et al., 2018) is an off-policy actor-critic RL algorithm used to collect the offline dataset in Meta-World, serving as a fundamental baseline. RAD (Laskin et al., 2020) applies data augmentation in the state space, enhancing generalization, particularly for image-based observations. S4RL (Sinha et al., 2022) extends this idea by integrating advanced state-space augmentation techniques to improve generalization in offline RL. MTDIFF (He et al., 2023) is a diffusion-based multi-task RL method that facilitates implicit knowledge sharing across tasks, enabling better adaptation to unseen tasks.

Implementation Details For offline data collection in Meta-World tasks, we follow the same criteria as (He et al., 2023), using SAC (Haarnoja et al., 2018) to pre-collect 40M timesteps of offline data. Our model adopts the same transformer architecture as MTDIFF (He et al., 2023) and learns policy diffusion based on the Diffusion-QL framework (Wang et al., 2023). We use the formal text descriptions from Meta-World (Yu et al., 2020) as text prompts and employ the same trained SAC agent to generate trajectory prompts. We primarily use Llama3-7B (Dubey et al., 2024) as our base LLM, with 3-layer MLP projection heads applied to both the LLM and transformer outputs. The RL agent is trained on three seen tasks, including sweep-into, coffee-push, and disassemble, and its generalization performance is evaluated on three unseen tasks, including box-close, hand-insert, and bin-picking. All experiments were conducted on 4 nodes, each equipped with 4 Intel Xeon CPUs, 32 GB of RAM, and NVIDIA RTX A5000 GPUs with 24 GB of VRAM.

Experimental Results The experimental results for Meta-World tasks are presented in Table 1. We evaluate the generalization ability of our LLMDPD method on three unseen tasks while using its performance on seen tasks to evaluate its overall sample efficiency during training. The results show that LLMDPD significantly outperforms all other methods on the three unseen tasks. Notably, on the bin-picking task, LLMDPD achieves an 18.92 improvement in average success rate compared to the previous best method, MTDIFF. Similarly, it shows an improvement of 7.84 on box-close and 2.45 on hand-insert, demonstrating strong generalization capabilities, even when compared to state-of-the-art data augmentation methods.

On seen tasks, our LLMDPD method achieves strong results, outperforming all other methods on coffee-push and disassemble tasks. This demonstrates that LLMDPD not only exhibits strong generalization on unseen tasks but also efficiently learns from training on seen tasks. However, on the sweep-into task, our method falls slightly behind MTDIFF. This may be because text descriptions primarily enhance performance on complicated tasks, while offering limited gains on simpler tasks.

4.2 Ablation Study

We conducted an ablation study on six variants of our model across three unseen tasks: (1) 'LLMDPD', our full model, which includes all components; (2) 'LLMDPD-OLMo-1B', which replaces the base

Table 2: This table presents the average success rates for all ablation variants on Meta-World-V2 tasks, evaluated over 500 episodes per task. Results are averaged over three runs.

Task	LLMDPD	LLMDPD-OLMo-1B	w/o-prompt	w/o-z_{text}	w/o-z_τ	w/o-DP
box-close	$\mathbf{75.24 \pm 4.90}$	72.18 ± 5.66	67.29 ± 6.89	70.14 ± 8.57	72.01 ± 8.37	70.53 ± 6.44
hand-insert	$\mathbf{78.71 \pm 5.33}$	76.61 ± 4.21	69.31 ± 5.38	72.16 ± 7.93	75.43 ± 6.70	73.19 ± 6.31
bin-picking	$\mathbf{74.65 \pm 6.28}$	71.76 ± 6.39	61.48 ± 8.85	64.55 ± 10.73	70.13 ± 8.64	66.86 ± 9.59

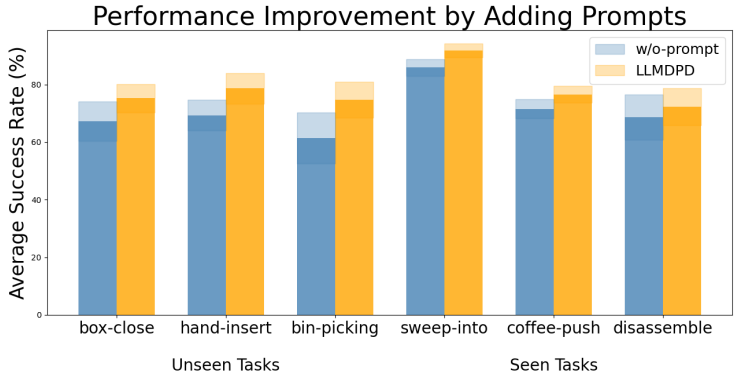

Figure 2: The figure illustrates the performance improvement achieved by incorporating prompts into our LLMDPD method. The blue column represents the average success rate of the 'w/o-prompt' ablation variant, while the orange column represents the full LLMDPD model. The shaded area indicates the standard deviation.

LLM with OLMo-1B (Groeneveld et al., 2024) for processing text prompt embeddings; (3) 'w/o-prompt', which removes both text and trajectory prompts; (4) 'w/o-z_{text}', which excludes only the text prompt; (5) 'w/o-z_τ', which excludes only the trajectory prompt; and (6) 'w/o-DP', which replaces the policy's diffusion model with a standard trajectory-based diffusion model. This study systematically evaluates each component's contribution to overall performance.

The results of the ablation study are presented in Table 2. The full model, 'LLMDPD', achieves the highest performance across all three tasks. Removing any component results in a performance drop, demonstrating the effectiveness of each part of our method. Replacing the base LLM with a smaller model in 'LLMDPD-OLMo-1B' leads to a decline in performance on all three tasks, indicating that larger models provide more detailed guidance to the policy by leveraging natural language processing and pre-collected knowledge. However, this variant still maintains strong generalization performance. Excluding both the text and trajectory prompts ('w/o-prompt') causes a significant performance drop, highlighting the importance of prompting in our method. The 'w/o-prompt' variant, which removes only the text prompt, results in an even more severe decline, particularly on hand-insert and bin-picking tasks. This suggests that without explicit textual guidance, the RL agent struggles to generalize to complex unseen tasks. Similarly, the 'w/o-z_τ' variant, which removes the trajectory prompt, exhibits degraded performance, demonstrating its role in helping the agent capture transition behaviors in unseen tasks. Additionally, 'w/o-DP', which replaces the policy diffusion model with a standard trajectory-based diffusion model, also experiences a significant performance drop. This demonstrates the importance of policy diffusion in reducing overfitting to seen offline data. Overall, the ablation results highlight the contribution of each component to LLMDPD's performance, with prompts playing a particularly crucial role in achieving optimal generalization.

We provide a visualization of the performance improvement achieved by incorporating both text and trajectory prompts into our model, as shown in Figure 2. This figure illustrates the impact of prompts on the model's performance across six tasks, with three unseen tasks (left) and three seen tasks (right). Across all tasks, the inclusion of prompts consistently enhances the average success rate, with the orange bars (LLMDPD) outperforming the blue bars ('w/o-prompt'). Notably, incorporating prompts leads to a greater improvement in the three unseen tasks compared to the seen tasks, demonstrating better generalization performance beyond training on the existing offline dataset. Additionally, the inclusion of prompts slightly reduces the standard deviation, indicating more stable training on the offline dataset and improved consistency in performance. These findings suggest that prompt-driven learning not only boosts success rates but also contributes to more reliable and

robust decision-making. Overall, these results highlight the crucial role of prompt-driven guidance in enhancing task understanding, execution, and generalization in offline RL.

5 Conclusion

In this work, we propose LLM-Driven Policy Diffusion (LLMDPD), a novel approach to enhancing generalization in offline RL through task-specific prompts. LLMDPD utilizes both easily collectible text-based task descriptions and single trajectory instances as prompts to guide policy learning. To provide rich task-relevant context, LLMDPD leverages LLMs to encode text-based prompts while using a transformer model to encode trajectory prompts. These prompts serve as conditional inputs to a context-aware policy diffusion module, enabling the RL agent to generalize effectively to unseen tasks. By integrating policy diffusion with Q-learning, LLMDPD employs an actor-critic diffusion algorithm to learn a generalizable, reward-maximizing policy. Experimental results on benchmark tasks show that LLMDPD outperforms state-of-the-art offline RL methods in terms of generalization, demonstrating its effectiveness in improving generalizability and adaptability.

References

C. Tang, B. Abbatematteo, J. Hu, R. Chandra, R. Martín-Martín, and P. Stone, "Deep reinforcement learning for robotics: A survey of real-world successes," *Annual Review of Control, Robotics, and Autonomous Systems*, 2024.

D. Lee, C. Eom, and M. Kwon, "Ad4rl: Autonomous driving benchmarks for offline reinforcement learning with value-based dataset," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

X.-Y. Liu, Z. Xia, J. Rui, J. Gao, H. Yang, M. Zhu, C. D. Wang, Z. Wang, and J. Guo, "Finrl-meta: Market environments and benchmarks for data-driven financial reinforcement learning," *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

C. Yu, J. Liu, S. Nemati, and G. Yin, "Reinforcement learning in healthcare: A survey," *ACM Computing Surveys (CSUR)*, 2021.

R. Kirk, A. Zhang, E. Grefenstette, and T. Rocktäschel, "A survey of zero-shot generalisation in deep reinforcement learning," *Journal of Artificial Intelligence Research*, 2023.

D. Arpit, S. Jastrzebski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio *et al.*, "A closer look at memorization in deep networks," in *International Conference on Machine Learning (ICML)*, 2017.

D. Yarats, I. Kostrikov, and R. Fergus, "Image augmentation is all you need: Regularizing deep reinforcement learning from pixels," in *International Conference on Learning Representations (ICLR)*, 2021.

R. Raileanu, M. Goldstein, D. Yarats, I. Kostrikov, and R. Fergus, "Automatic data augmentation for generalization in reinforcement learning," *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

H. Zhang and Y. Guo, "Generalization of reinforcement learning with policy-aware adversarial data augmentation," *arXiv preprint arXiv:2106.15587*, 2021.

R. Wang, J. Lehman, J. Clune, and K. O. Stanley, "Paired open-ended trailblazer (poet): Endlessly generating increasingly complex and diverse learning environments and their solutions," *arXiv preprint arXiv:1901.01753*, 2019.

R. Wang, J. Lehman, A. Rawal, J. Zhi, Y. Li, J. Clune, and K. Stanley, "Enhanced poet: Open-ended reinforcement learning through unbounded invention of learning challenges and their solutions," in *International Conference on Machine Learning (ICML)*, 2020.

Y. Liu, J. Ding, and X. Liu, "Ipo: Interior-point policy optimization under constraints," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

R. Raileanu and R. Fergus, "Decoupling value and policy for generalization in reinforcement learning," in *International Conference on Machine Learning (ICML)*, 2021.

B. Mazoure, I. Kostrikov, O. Nachum, and J. J. Tompson, "Improving zero-shot generalization in offline reinforcement learning using generalized similarity functions," *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

I. Mediratta, Q. You, M. Jiang, and R. Raileanu, "The generalization gap in offline reinforcement learning," in *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

M. Laskin, K. Lee, A. Stooke, L. Pinto, P. Abbeel, and A. Srinivas, "Reinforcement learning with augmented data," *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

S. Sinha, A. Mandlekar, and A. Garg, "S4rl: Surprisingly simple self-supervision for offline reinforcement learning," in *Conference on Robot Learning (CoRL)*, 2022.

N. Modhe, Q. Gao, A. Kalyan, D. Batra, G. Thattai, and G. Sukhatme, "Exploiting generalization in offline reinforcement learning via unseen state augmentations," *arXiv preprint arXiv:2308.03882*, 2023.

W. Qiao and R. Yang, "Soft adversarial offline reinforcement learning via reducing the attack strength for generalization," in *Proceedings of the 2024 16th International Conference on Machine Learning and Computing (ICMLC)*, 2024.

H. He, C. Bai, K. Xu, Z. Yang, W. Zhang, D. Wang, B. Zhao, and X. Li, "Diffusion model is an effective planner and data synthesizer for multi-task reinforcement learning," *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Y. Ma, H. Tang, D. Li, and Z. Meng, "Reining generalization in offline reinforcement learning via representation distinction," *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

D. Wang, L. Li, W. Wei, Q. Yu, H. Jianye, and J. Liang, "Improving generalization in offline reinforcement learning via adversarial data splitting," in *Forty-first International Conference on Machine Learning (ICML)*, 2024.

L. Qin, Q. Chen, X. Feng, Y. Wu, Y. Zhang, Y. Li, M. Li, W. Che, and P. S. Yu, "Large language models meet nlp: A survey," *arXiv preprint arXiv:2405.12819*, 2024.

X. Xu, M. Li, C. Tao, T. Shen, R. Cheng, J. Li, C. Xu, D. Tao, and T. Zhou, "A survey on knowledge distillation of large language models," *arXiv preprint arXiv:2402.13116*, 2024.

C. Yang, Y. Zhu, W. Lu, Y. Wang, Q. Chen, C. Gao, B. Yan, and Y. Chen, "Survey on knowledge distillation for large language models: Methods, evaluation, and application," *ACM Transactions on Intelligent Systems and Technology*, 2024.

T. Yu, D. Quillen, Z. He, R. Julian, K. Hausman, C. Finn, and S. Levine, "Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning," in *Conference on robot learning*. PMLR, 2020, pp. 1094–1100.

Z. Zhao, Z. Ren, L. Yang, F. Yuan, P. Ren, Z. Chen, X. Xin *et al.*, "Offline trajectory generalization for offline reinforcement learning," *arXiv preprint arXiv:2404.10393*, 2024.

M. Xu, Y. Shen, S. Zhang, Y. Lu, D. Zhao, J. Tenenbaum, and C. Gan, "Prompting decision transformer for few-shot policy generalization," in *International Conference on Machine Learning (ICML)*, 2022.

L. Li, H. Zhang, X. Zhang, S. Zhu, Y. YU, J. Zhao, and P.-A. Heng, "Towards an information theoretic framework of context-based offline meta-reinforcement learning," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

Z. Wang, L. Zhang, W. Wu, Y. Zhu, D. Zhao, and C. Chen, "Meta-DT: Offline meta-RL as conditional sequence modeling with world model disentanglement," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine, "Planning with diffusion for flexible behavior synthesis," *arXiv preprint arXiv:2205.09991*, 2022.

A. Ajay, Y. Du, A. Gupta, J. Tenenbaum, T. Jaakkola, and P. Agrawal, "Is conditional generative modeling all you need for decision-making?" *arXiv preprint arXiv:2211.15657*, 2022.

S. Sharan, R. Zhao, Z. Wang, S. P. Chinchali *et al.*, "Plan diffuser: Grounding llm planners with diffusion models for robotic manipulation," in *Bridging the Gap between Cognitive Science and Robot Learning in the Real World: Progresses and New Directions*, 2024.

F. Ni, J. Hao, Y. Mu, Y. Yuan, Y. Zheng, B. Wang, and Z. Liang, "Metadiffuser: Diffusion model as conditional planner for offline meta-rl," in *International Conference on Machine Learning (ICML)*, 2023.

C. Chen, F. Deng, K. Kawaguchi, C. Gulcehre, and S. Ahn, "Simple hierarchical planning with diffusion," in *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

Z. Wang, J. J. Hunt, and M. Zhou, "Diffusion policies as an expressive policy class for offline reinforcement learning," in *The Eleventh International Conference on Learning Representations*, 2023.

C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *The International Journal of Robotics Research*, p. 02783649241273668, 2023.

C. Sun, S. Huang, and D. Pompili, "Llm-based multi-agent reinforcement learning: Current and future directions," *arXiv preprint arXiv:2405.11106*, 2024.

Y. Cao, H. Zhao, Y. Cheng, T. Shu, Y. Chen, G. Liu, G. Liang, J. Zhao, J. Yan, and Y. Li, "Survey on large language model-enhanced reinforcement learning: Concept, taxonomy, and methods," *IEEE Transactions on Neural Networks and Learning Systems*, 2024.

Y. Du, O. Watkins, Z. Wang, C. Colas, T. Darrell, P. Abbeel, A. Gupta, and J. Andreas, "Guiding pretraining in reinforcement learning with large language models," in *International Conference on Machine Learning*, 2023.

B. Wang, Y. Qu, Y. Jiang, J. Shao, C. Liu, W. Yang, and X. Ji, "LLM-empowered state representation for reinforcement learning," in *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

X. Yan, Y. Song, X. Feng, M. Yang, H. Zhang, H. B. Ammar, and J. Wang, "Efficient reinforcement learning with large language model priors," in *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025.

R. Shi, Y. Liu, Y. Ze, S. S. Du, and H. Xu, "Unleashing the power of pre-trained language models for offline reinforcement learning," in *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, "Decision transformer: Reinforcement learning via sequence modeling," *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Y. Yang and P. Xu, "Pre-trained language models improve the few-shot prompt ability of decision transformer," *arXiv preprint arXiv:2408.01402*, 2024.

E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen *et al.*, "Lora: Low-rank adaptation of large language models." *ICLR*, vol. 1, no. 2, p. 3, 2022.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

H. Hasselt, "Double q-learning," *Advances in Neural Information Processing Systems*, 2010.

E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.

T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International Conference on Machine Learning (ICML)*, 2018.

A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan *et al.*, "The llama 3 herd of models," *arXiv preprint arXiv:2407.21783*, 2024.

D. Groeneveld, I. Beltagy, P. Walsh, A. Bhagia, R. Kinney, O. Tafjord, A. H. Jha, H. Ivison, I. Magnusson, Y. Wang *et al.*, "Olmo: Accelerating the science of language models," *arXiv preprint arXiv:2402.00838*, 2024.