Future-conditioned Unsupervised Pretraining for Decision Transformer

Zhihui Xie[1] Zichuan Lin[2] Deheng Ye[2] Qiang Fu[2] Wei Yang[2] Shuai Li[1]

[1]Shanghai Jiao Tong University [2]Tencent AI Lab

TL;DR

- We consider the following question:
  How can reward-free offline interaction data be used to enhance downstream decision-making tasks?
- We propose **PDT**, an unsupervised pretraining method for decision making.
- Experimental results show that PDT achieves **superior few-shot generalization performance**.

Offline RL via Sequence Modeling

Recent works (Chen et al., 2021; Lee et al., 2022) pose offline RL as a sequence modeling problem.

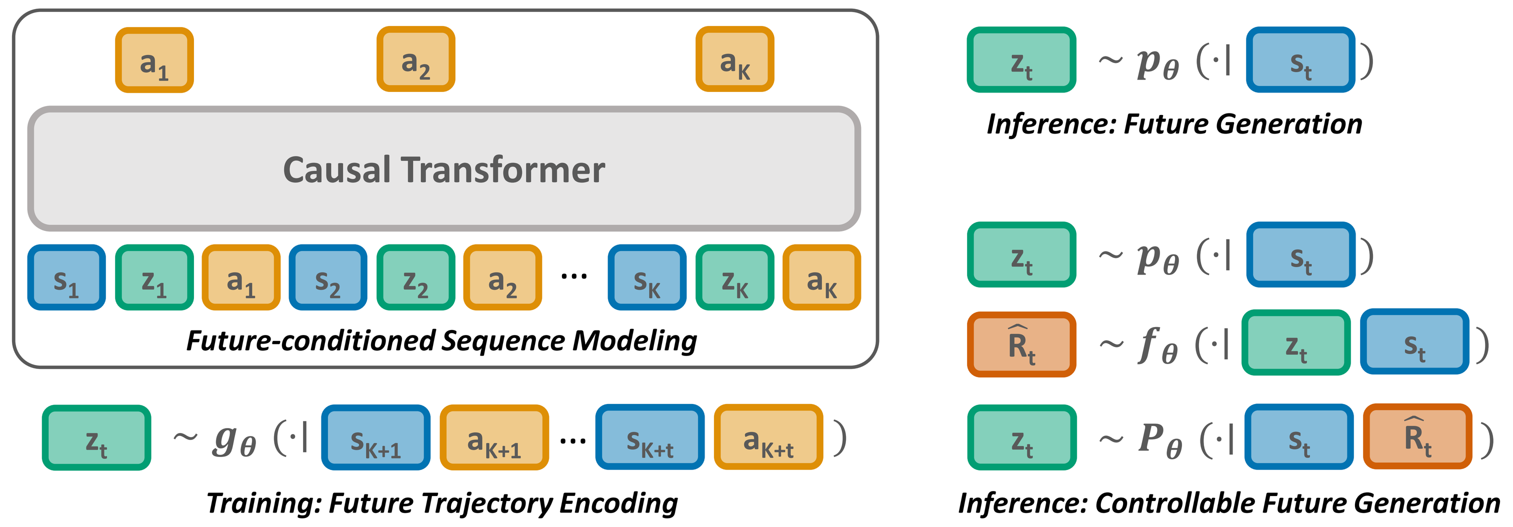
- Trajectory sequences as inputs:
$$\hat{\tau} = (\hat{R}_1, s_1, a_1, \hat{R}_2, s_2, a_2, \dots, \hat{R}_T, s_T, a_T)$$
  where $\hat{R}_t = \sum_{t'=t}^{T} r_{t'}$ is the target return.
- Autoregressive models (e.g., GPT) as policies:
$$\pi_\theta(a_t \mid \hat{\tau}_{1:t-1}, s_t, \hat{R}_t)$$
- Next action prediction as the learning objective:
$$\mathcal{L}_{\text{DT}} = \mathbb{E}_{\hat{\tau}\sim\hat{\mathcal{D}}}\left[\sum_{t=1}^{T} -\log \pi_\theta(a_t \mid \hat{\tau}_{1:t-1}, s_t, \hat{R}_t)\right]$$

While promising, return-conditioned methods have their **shortcomings**:

- They can not handle reward-free data, which is much easier to scale up.
- Conditioning on scalar reward values can lead to inconsistent policies (Paster et al., 2022).

This work: *Can we retrofit the return-conditioned framework for unsupervised pretraining?*

Our Method: PDT



The proposed Pretrained Decision Transformer (PDT) is a two stage **pretrain-then-finetune** method:

- **Offline pretraining**: Learning a **future-conditioned** policy $\pi_\theta(a_t \mid \tau_{1:t-1}, s_t, z)$ that utilizes reward-free future trajectories $\tau_{t+1:T} = (s_{t+1}, a_{t+1}, \dots, s_T, a_T)$:
$$z \sim g_\theta(\cdot \mid \tau_{t+1:T}) \ \texttt{\# training}$$
$$z \sim p_\theta(\cdot \mid s_t) \quad \texttt{\# inference}$$
- **Online finetuning**: Learning to controllably sample high-return futures via **return prediction**:
$$p(z \mid \hat{R}_t, s_t) \propto p(z \mid s_t) \underbrace{p(\hat{R}_t \mid z, s_t)}_{\texttt{learned}}$$

PDT can be seen as an instance of **Successor Features** (SFs, Barreto et al., 2017):

- SFs assume that rewards can be decomposed into task-agnostic dynamics ϕ and task preference $\mathbf{w}$:
$$r(s,a) = \phi(s,a)^\top \mathbf{w}$$
- PDT tames return conditioning in a similar way:
$$\hat{R}_t = \left[\sum_{t'=t}^{T} \phi(s_{t'+1}, a_{t'+1})\right]^\top \mathbf{w}$$
  where the summation can be pretrained as g_θ and $\mathbf{w}$ is learned via return prediction during finetuning.

Experimental Results

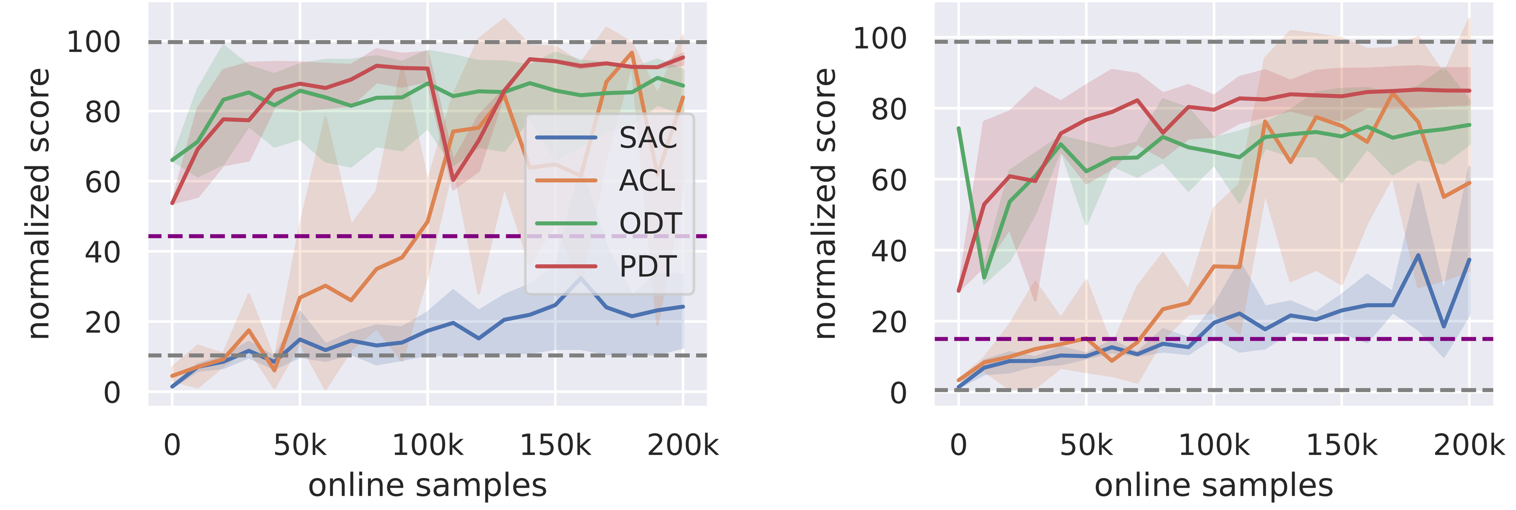


Figure 1. PDT outperforms other unsupervised pretraining methods and performs on par with its supervised pretraining counterpart in few-shot settings.

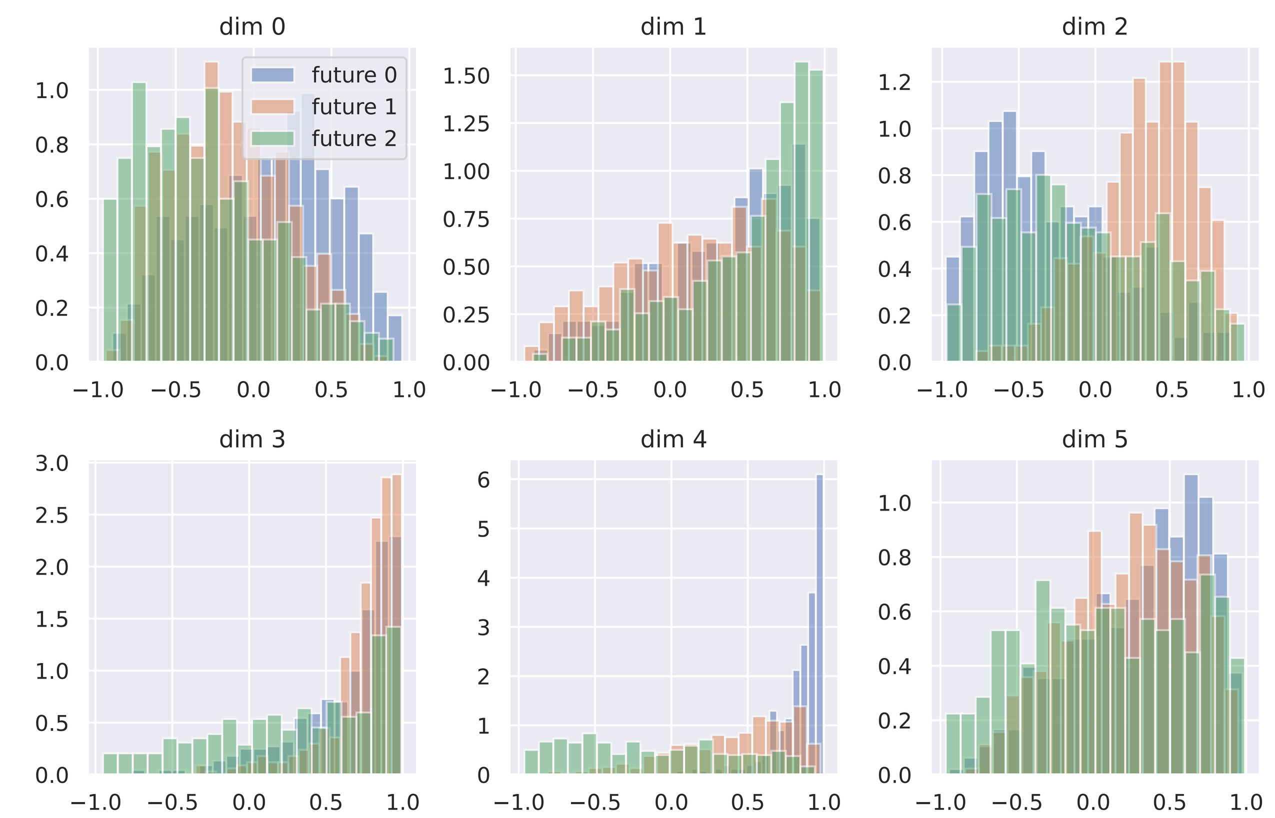


Figure 2. PDT can generate diverse behaviors conditioning on different futures.

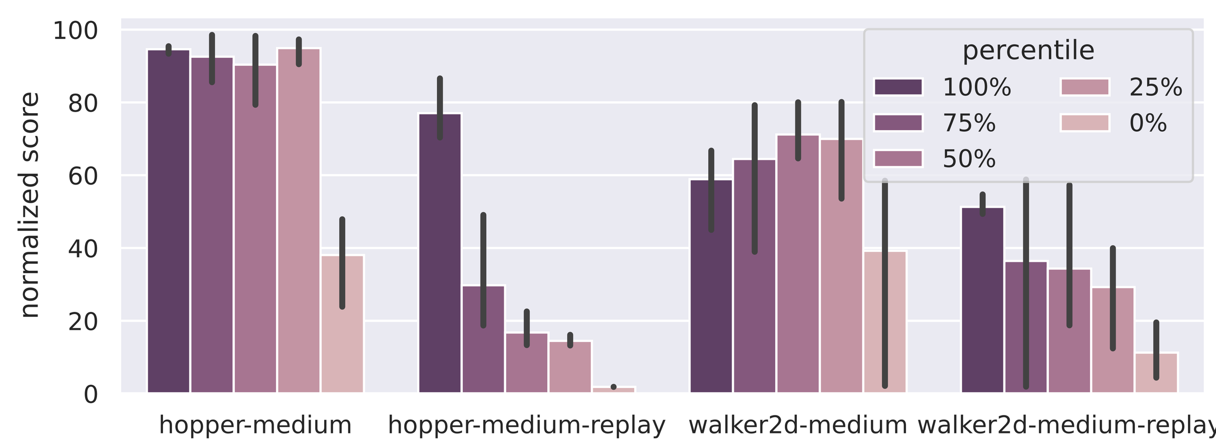


Figure 3. PDT can controllably generate high-return behaviors via online finetuning.

Please refer to our paper and code for more details!