上海科技大学
ShanghaiTech University

Diffusion for Offline Bandit

Shouchen Zhou, Xinyue Ying

Introduction

Diffusion models[1] are generative models that learn data distributions through step-by-step denoising. While popular in image generation, they also excel at modeling complex, multimodal structures in other domains.

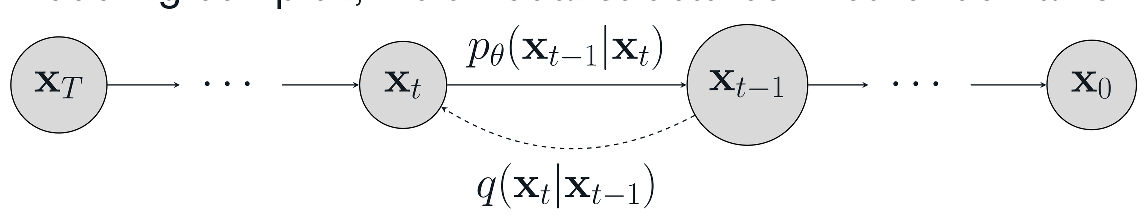


Figure 1: Diffusion model represented as a Markov chain, with a forward process for noise addition during training and a backward process for denoising and generation.

In this work, we explore diffusion in **offline bandit learning** from two perspectives:

- **Stochastic Bandit**: A discrete diffusion model generates additional trajectories from interaction logs, expanding the limited offline dataset to enhance pretraining and reduce exploration costs.
- **Contextual Bandit**: A diffusion model is trained to approximate the conditional reward distribution $P(r \mid c, a)$, enabling action selection via sampling and capturing aleatoric uncertainty.

Stochastic Bandit

- Traditional algorithms: cold start or rely on limited offline logs, which suffer from size limitations, narrow coverage, and distribution bias, constraining performance.
- Our approach: employ a discrete diffusion model to synthesize additional pseudo-trajectories, broadening data diversity and coverage, and apply a policy gradient based bandit algorithm to fully exploit the expanded offline dataset.

Similarly to the policy gradient methods [2] in Reinforcement Learning algorithms, in the bandit settings, the online interaction log of a stochastic multi-armed bandit can be viewed as a trajectory composed of action–reward pairs.

$$\psi = (a_0, r_1, \cdots, , a_{T-1}, r_T)\,. \tag{1}$$

By archiving several past trajectories into an offline dataset, we can pre-train the stochastic bandit and thereby cut down the cost of subsequent online interactions. The probability of encountering a specific trajectory ψ is

$$P_\theta(\psi) = \prod_{t=0}^{T-1} \pi_\theta(a_t \mid a_0, r_1, \ldots, a_{t-1}, r_t). \tag{2}$$

The objective function is given by

$$J(\theta) = \mathbb{E}_{\Psi \sim P_\theta}[R(\Psi)] = \sum_\psi P_\theta(\psi) R(\psi). \tag{3}$$

The policy gradient can be expressed as

$$\nabla_\theta J(\theta) \approx \frac{1}{m} \sum_{i=1}^{m} \sum_{t=0}^{T-1} \left[\left(R_t^i - b_t\right) \nabla_\theta \log \pi_\theta \left(a_t^i \mid a_0^i, r_1^i, \ldots, a_{t-1}^i, r_t^i\right) \right] \tag{4}$$

Thus, given an offline dataset, we improve stochastic bandit performance:

1. **Dataset expansion**: generate extra trajectories with diffusion model.
2. **Pre-training**: train stochastic bandit algorithms on the enlarged dataset.
3. **Online adaptation**: run and refine the pretrained agents online.

With the pretrained weights, the policy executed at each online step is:

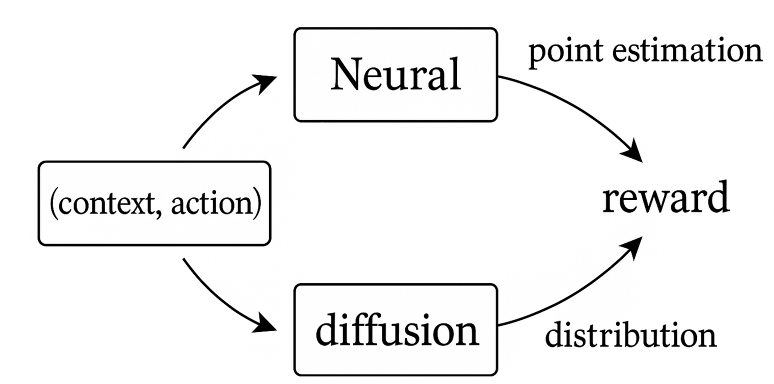
$$\pi_\theta(a) = \frac{e^{\beta_t H_\theta(a)}}{\sum_{a'} e^{\beta_t H_\theta(a')}} \tag{5}$$

Which is similar to the traditional gradient bandit algorithm [3], and has similar online update rules. The non-Bernoulli distribution $X \in \{0, 0.5, 1\}$ with probs $\theta_1, \theta_2, 1-\theta_1-\theta_2$ was also used to test the performance.

Contextual Bandit

- Traditional algorithms: estimating the expected reward and epistemic uncertainty.
- In real-world scenarios, distributional features such as multimodality, skewness, and heavy tails, reflecting aleatoric uncertainty, can provide valuable information for decision-making.

This project investigates a method that makes decisions by directly sampling from the full conditional reward distribution $P(r \mid c, a)$, which is learned through a diffusion model.



Reward Distribution Modeling:
Pretrain: A diffusion model is trained on (c, a, r) data to approximate $P(r \mid c, a)$.
Action Selection (at context c_t):

1. For each action a: sample $\tilde{r}_a \sim P(r \mid c_t, a; W_{\text{diffusion}})$ using the trained diffusion model.
2. Choose action: $a_t = \arg\max_{a \in \mathcal{A}} \tilde{r}_a$.

Stochastic Bandit Results

Trajectory generation policy	UCB	TS	policy gradient
no offline dataset	1691.827	163.483	858.794
offline, no enlarge	1303.405	43.550	82.254
offline+copy	1156.635	26.596	54.048
offline+diffuse pair	1028.613	7.296	42.993
offline+diffusion sequence	923.779	0.071	3.522
offline+diffusion sequence (Transformer)	743.441	0.008	0.002

Table 1: Performance(average accumulated regret) of various algorithms on Bernoulli-reward bandits under different offline-dataset enlargement (trajectory-generation) policies.

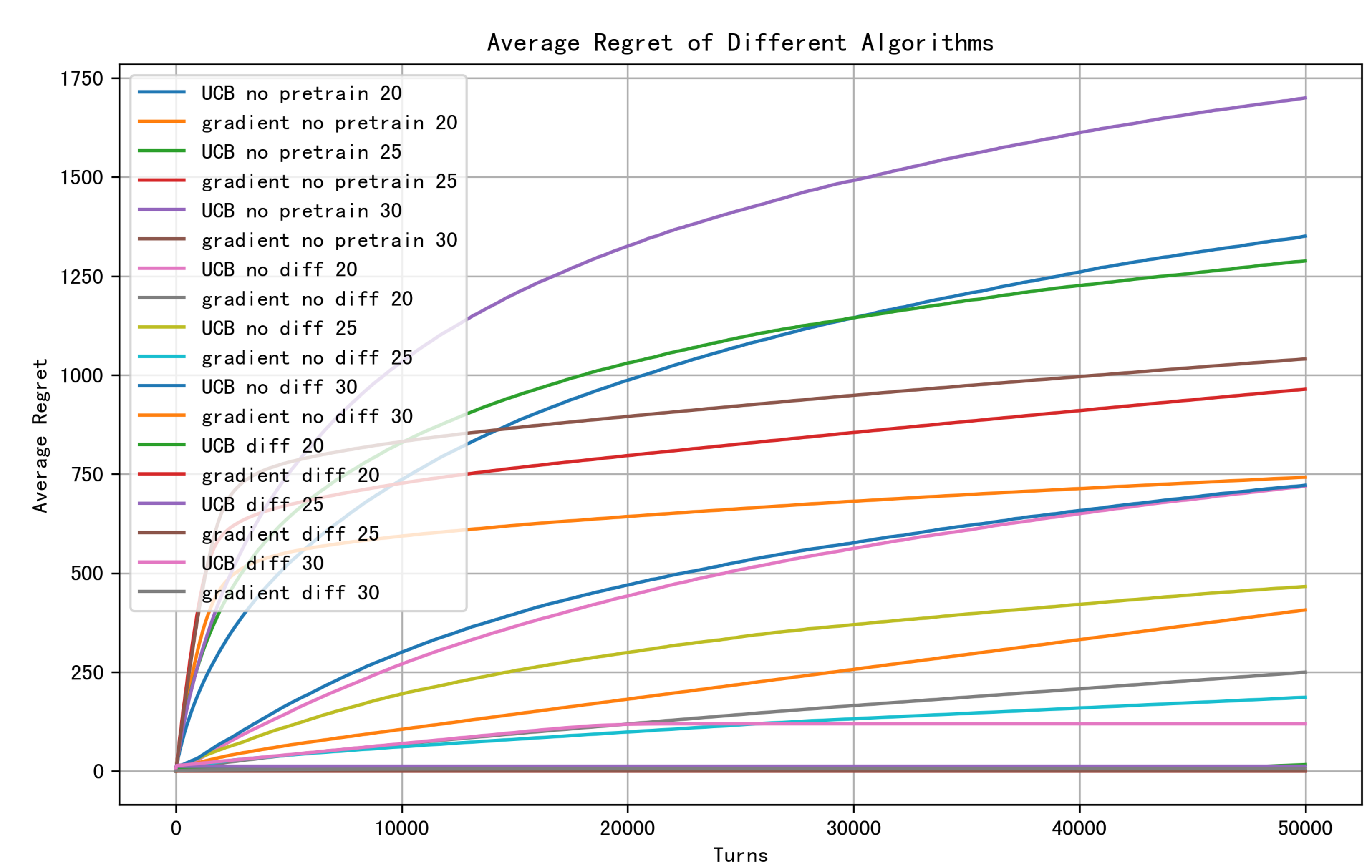


Fig. 1: Performance of each algorithm(UCB, policy gradient) across arm counts(20, 25, 30) with non-Bernoulli rewards, evaluated under three pre-training settings: none, offline (500 trajectories generated by diffusion sequence (Transformer).

Contextual Bandit Results

Diffusion can learn the conditional distribution of 0/1 rewards from MNIST and make decisions accordingly, but underperforms compared to NeuralTS and Neural Epsilon-Greedy.

- The aleatoric uncertainty learned in Diffusion is unnecessary for Bernoulli-type rewards, and instead increases the randomness of its outcomes.
- Its advantage in modeling complex distributions (e.g., multimodal, heavy-tailed) is less relevant for simple binary rewards. Mean estimation suffices.

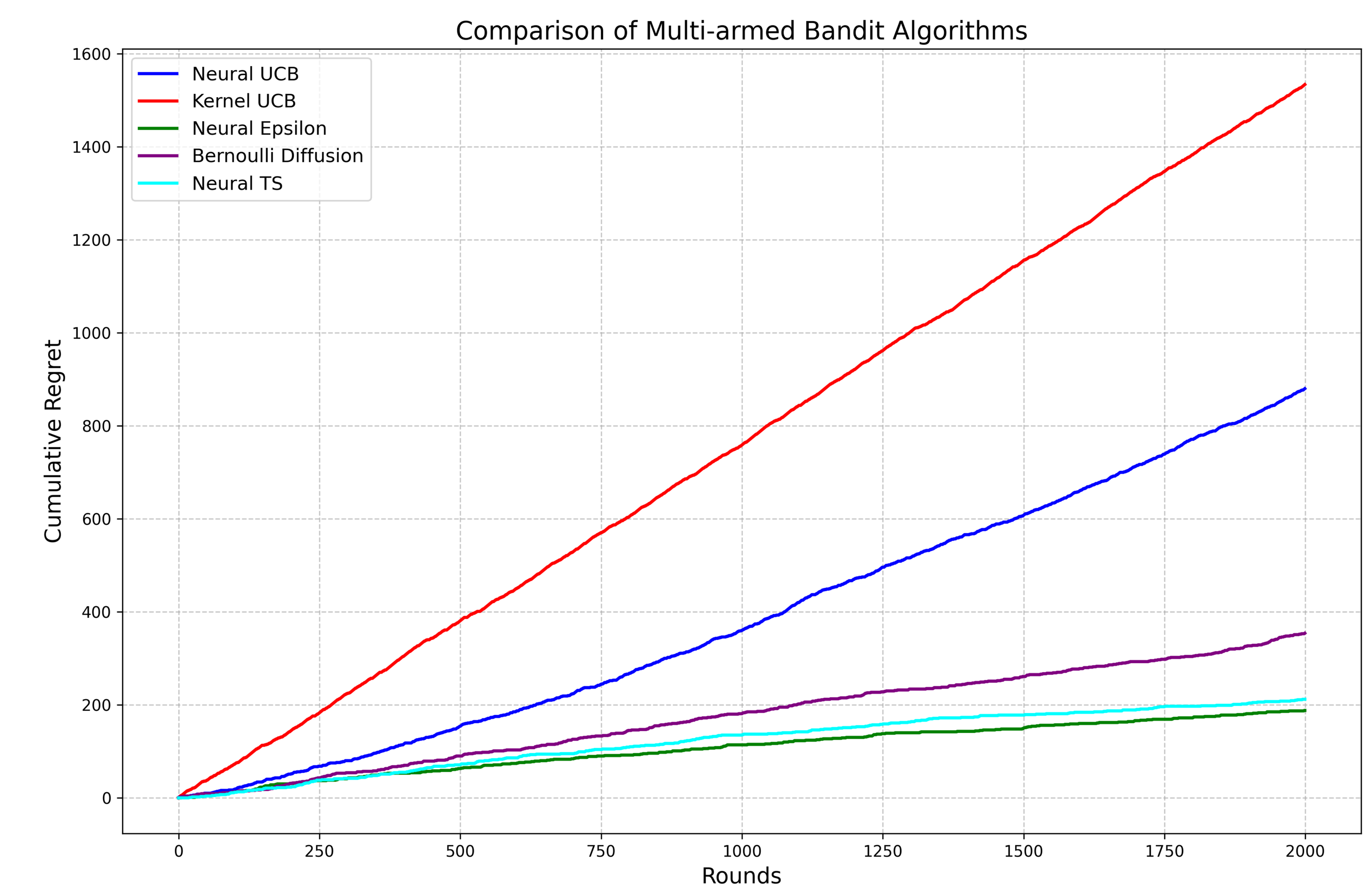


Fig. 2: Comparison of cumulative regret across different algorithms after 25 rounds of pertaining using MNIST Dataset.

Future Work

Stochastic Bandit:

- Replace the linear preference function $H_\theta(a)$ with kernel methods or neural networks to enhance model expressiveness and policy flexibility.
- Combine the discrete diffusion process with Transformer-style autoregressive generation to further improve trajectory quality.
- Extend the discrete diffusion model to dynamic settings such as Restless Bandits, evaluating its effectiveness in more complex decision scenarios.

Contextual Bandit:

- Integrating Bayesian principles (such as Bayesian diffusion models or model ensembling) into Diffusion to more effectively quantify and leverage epistemic uncertainty (exploration).
- Additionally, exploring more sophisticated risk-sensitive decision-making rules based on the full return distribution learned by Diffusion will also be a promising direction.

References

[1] Andrew Campbell et al. "A continuous time framework for discrete denoising models". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 28266–28279.

[2] Richard S Sutton et al. "Policy gradient methods for reinforcement learning with function approximation". In: *Advances in neural information processing systems* 12 (1999).

[3] Ronald J Williams. "Simple statistical gradient-following algorithms for connectionist reinforcement learning". In: *Machine learning* 8 (1992), pp. 229–256.