Thompson Sampling with Diffusion Generative Prior

Yu-Guan Hsieh[1], Shiva Kasiviswanathan[2], Branislav Kveton[2], Patrick Blöbaum[2] ([1]Université Grenoble Alpes [2]AWS AI Labs)

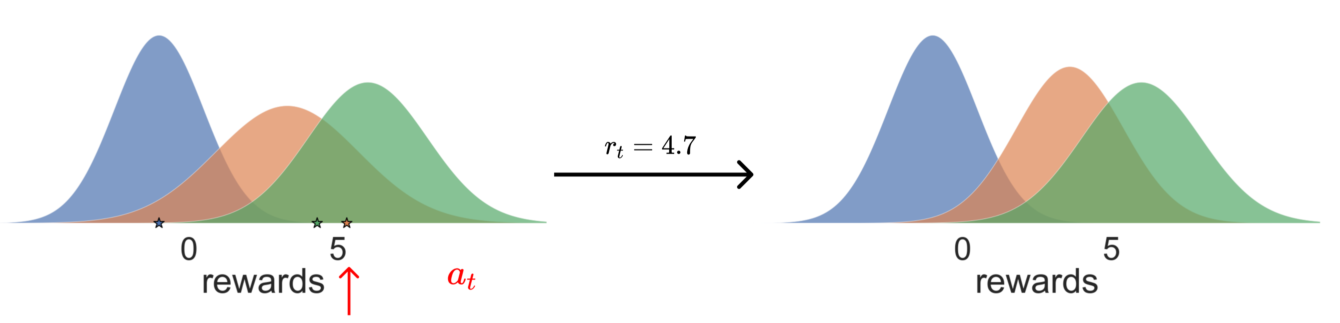
Multi-Armed Bandits

A model for online decision making

- Learner pulls arm $a_t \in \mathcal{A} = \{1, \ldots K\}$ at round t
- Learner receives rewards r_t drawn from the arm's distribution
- The goal is to maximize the cumulative rewards $\sum_t r_t$

Thompson Sampling

- Given a prior $p(\mu)$ over mean reward vector μ and $\mathcal{H}_t = (a_s, r_s)_{s \in \{1,\ldots,t\}}$ is the interaction history
- Maintain posterior distribution $p(\mu \mid \mathcal{H}_t) \propto p(\mathcal{H}_t \mid \mu) p(\mu)$
- Sample $\tilde{\mu}_t$ from the posterior and pull $a_t \in \arg\max_{a \in \mathcal{A}} \tilde{\mu}^a_t$



Meta-Learning For Bandits

Different bandit instances can have similar patterns

- Recommend items to different customers
- Assign price to different items using an online pricing algorithm

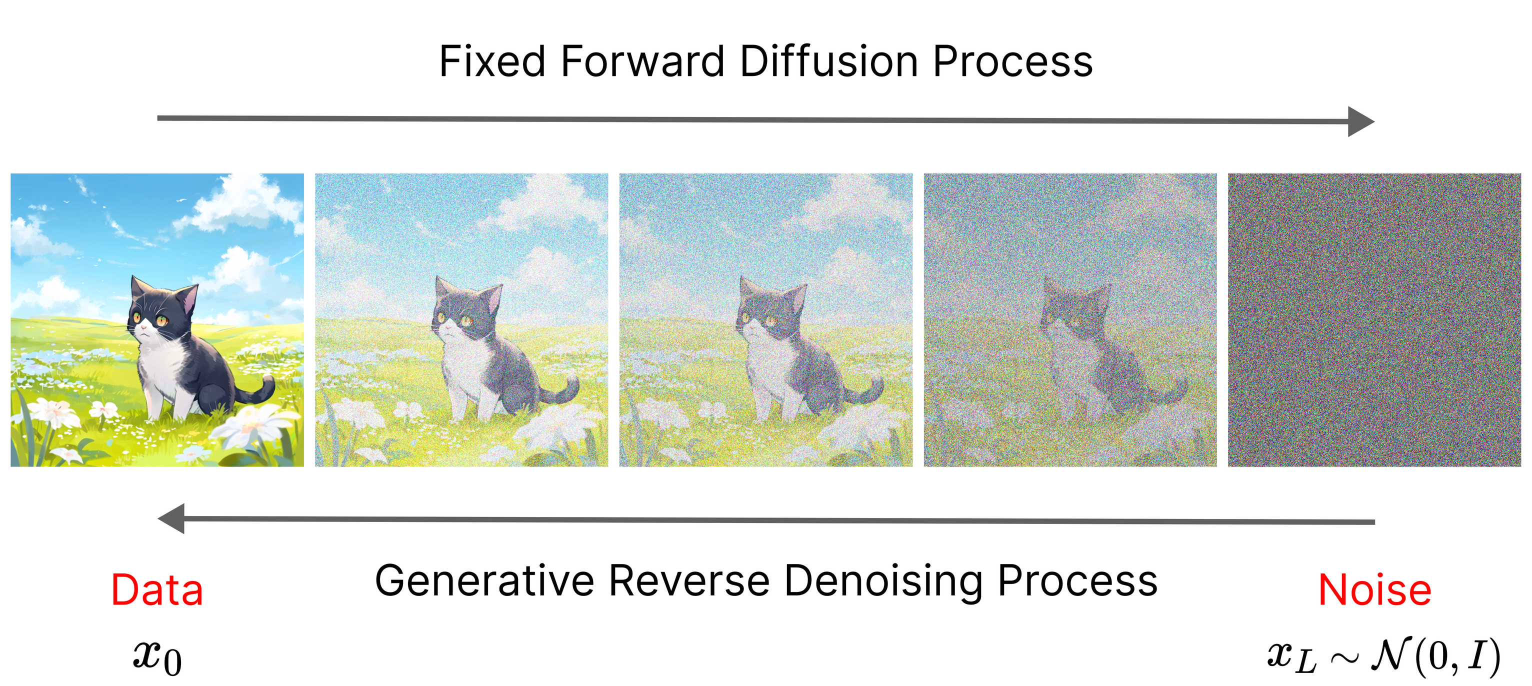
Diffusion Models

- Noise is gradually added in the forward diffusion process that goes from x_0 to x_L so that $q(X_{\ell+1} \mid x_\ell)$ is gaussian
- The model learns a reverse process

$$p_\theta(X_\ell \mid x_{\ell+1}) = q(X_\ell \mid x_{\ell+1}, X_0 = h_\theta(x_{\ell+1}, \ell + 1))$$

where h_θ is the trained denoiser that predicts x_0

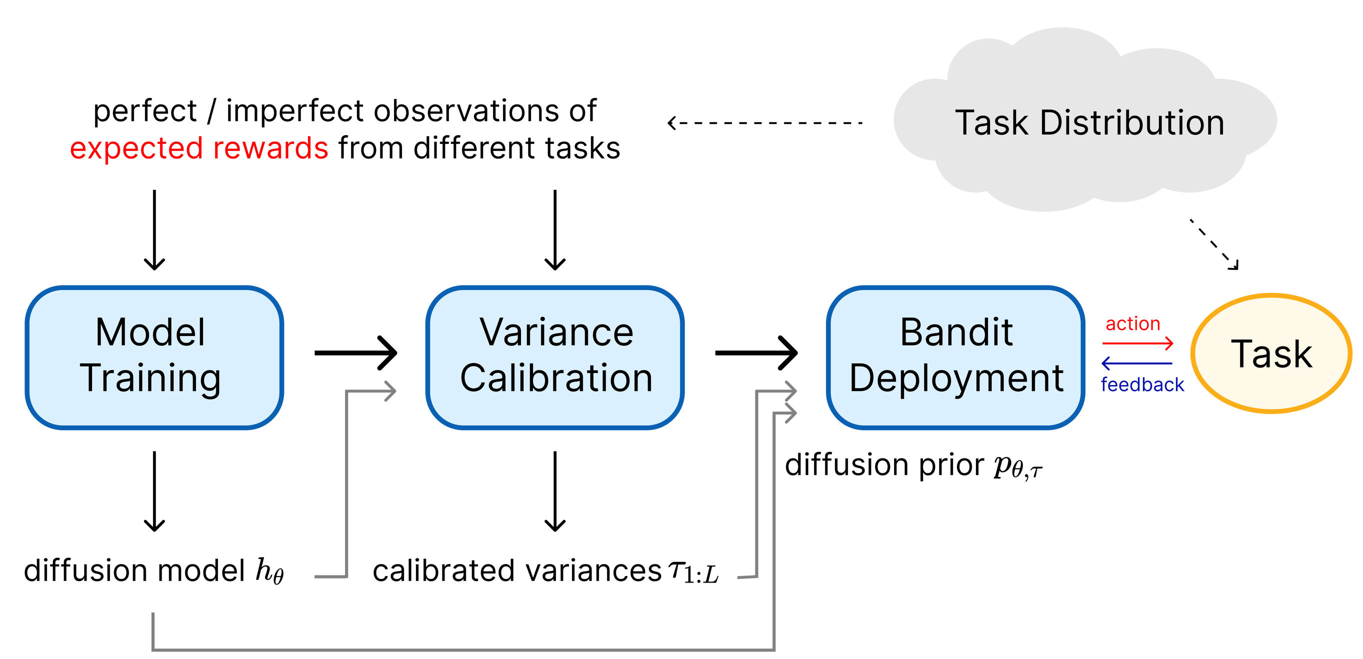
- The iterative process allows easy manipulation of the learned distribution for downstream tasks



TL;DR

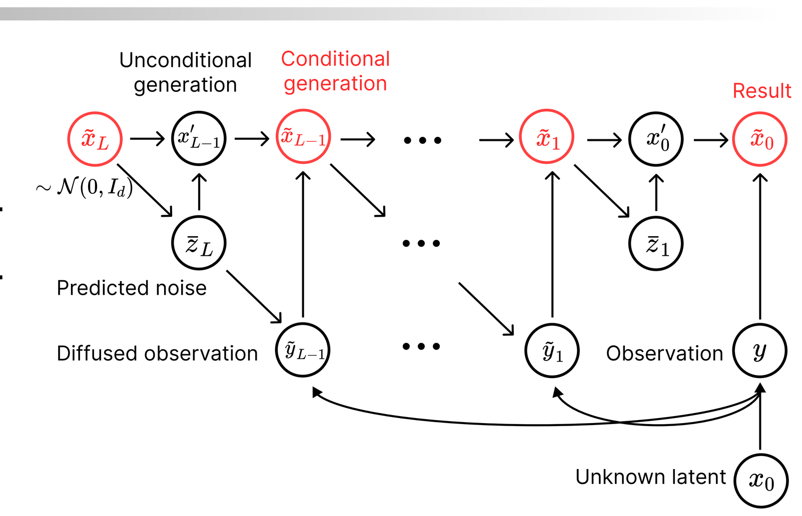
We (i) propose Thompson sampling with a diffusion prior, (ii) show how to estimate the prior from imperfect historical data, and (iii) validate our approach experimentally.

Algorithms



Thompson Sampling with Diffusion Prior

Goal: Sample $\tilde{\mu}_t$ from $X_0 \mid \mathcal{H}_{t-1}$



- Summarize $\mathcal{H}_{t-1}$ with the empirical mean $\hat{\mu}^a_{t-1}$ and the standard error vector σ^a_{t-1}
- Initialize: Sample $\hat{x}_L \sim \mathcal{N}(0, I)$
- Repeat: sample $x'_\ell \sim p_{\theta,\tau}(X_\ell \mid x_{\ell+1})$ with the diffusion model

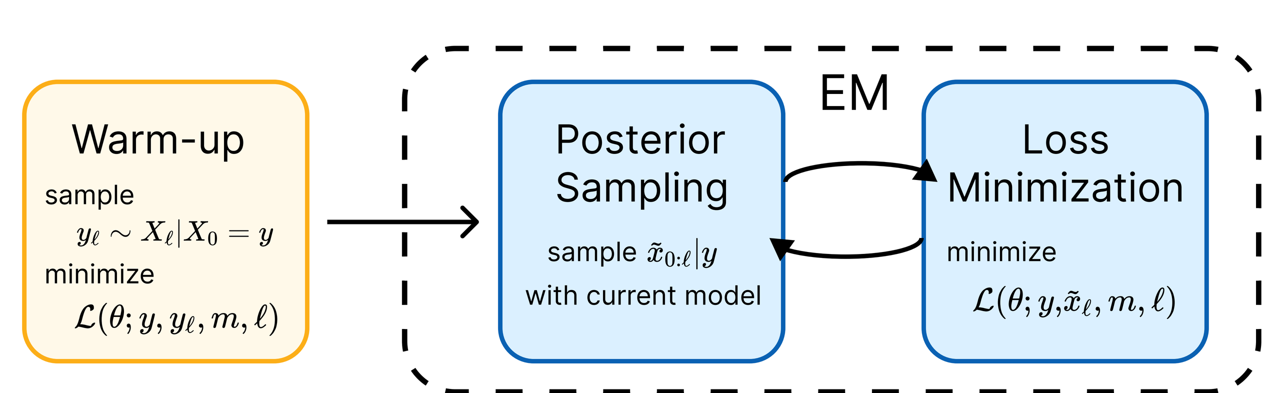
  If a has been pulled, compute $\tilde{y}^a_\ell$ from $y^a = \hat{\mu}^a_{t-1}$ through forward diffusion with noise predicted at $x_{\ell+1}$, and mix x'^a_ℓ and $\tilde{y}^a_\ell$

Diffusion Model Training from Imperfect Data

Data are incomplete and noisy $y_0 = m \odot (x_0 + z)$, where m is a binary mask and z is noise. We use an EM-like procedure and minimize

$$\mathcal{L}(\theta; y_0, \tilde{x}_\ell, m, \ell) = \|m \odot y_0 - m \odot h_\theta(\tilde{x}_\ell, \ell)\|^2 \quad \text{(ignore masked value)}$$

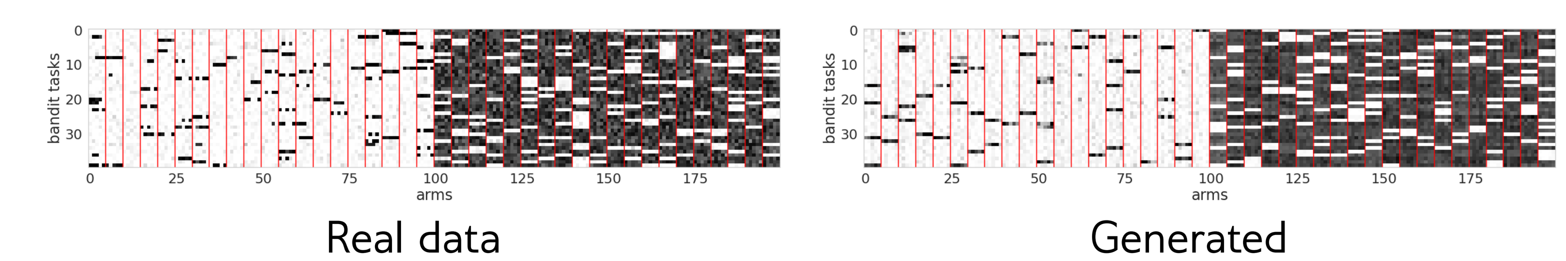
$$+ 2\lambda \sqrt{\bar{\alpha}_\ell}\sigma^2 \, \mathbb{E}_{b \sim \mathcal{N}(0,I)} b^\top \left(\frac{h_\theta(\tilde{x}_\ell + \varepsilon b, \ell) - h_\theta(\tilde{x}_\ell, \ell)}{\varepsilon} \right) \quad \text{(SURE)}$$



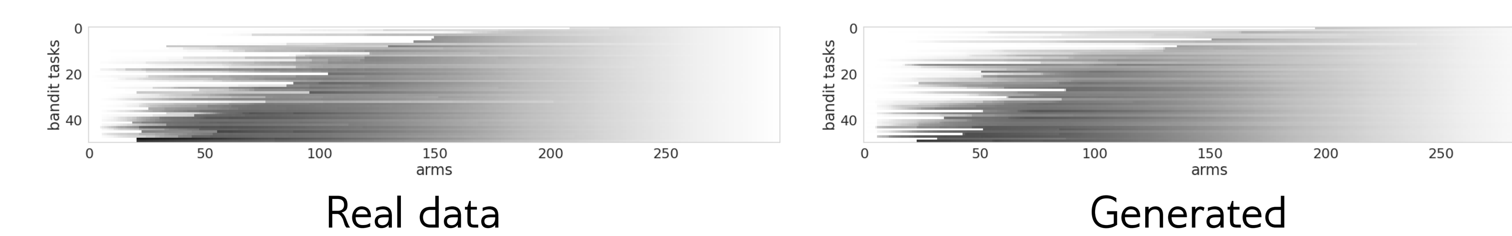
Numerical Experiments

Problem Construction

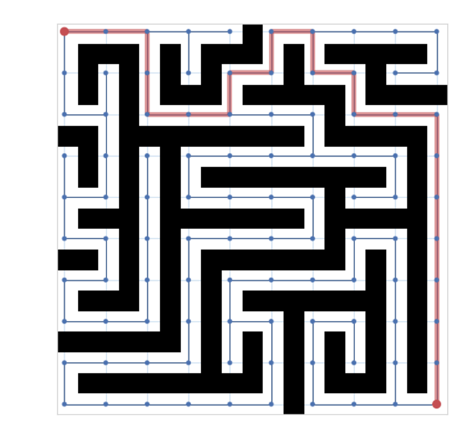
- `Popular and Niche:` 200 arms are separated into 40 groups
  - ▶ 20 groups represent the popular items (right half)
  - ▶ 20 groups represent the niche items (left half)



- `iPinYou Bidding:` Setting the bid price $b \in \{0, \ldots, 299\}$ in auctions. The reward is either $300 - b$ if the learner wins the auction or 0 otherwise.



- `Maze:` Online shortest path routing on grid graphs as reward maximization semi-bandit. The edges' mean rewards are derived from a 2D maze.



Results

- Regret is the difference of cumulative rewards between an algorithm and the one that consistently chooses the best action
- Training from clean data (top): training and validation set size of $5000/1200/5000$ and $1000/100/1000$
- Training from imperfect data (bottom): 50% feature dropping rate and 0.1 noise standard deviation in data