NEURAL INFORMATION PROCESSING SYSTEMS

DPOK: Reinforcement Learning for Fine-tuning Text-to-Image Diffusion Models

Ying Fan*, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, Kimin Lee*
(*Equal technical contribution)

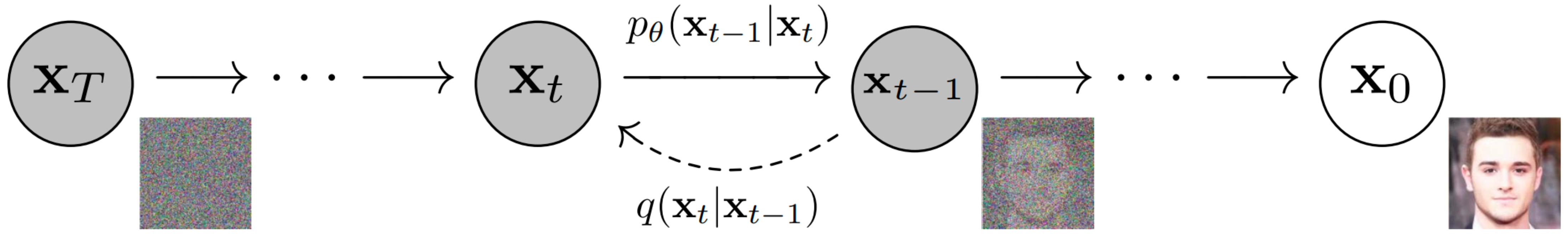
WISCONSIN UNIVERSITY OF WISCONSIN–MADISON Google Research Berkeley UNIVERSITY OF CALIFORNIA amazon KAIST

Denoising diffusion probabilistic models

Forward (diffusion) process: $q(x_{1:T} | x_0) = \prod_{t=1}^{T} q(x_t | x_{t-1}), q(x_t | x_{t-1}) = \mathcal{N}(\sqrt{1-\beta_t} x_{t-1}, \beta_t I)$

$$p(x_T) = \mathcal{N}(0, I), p_\theta(x_{t-1} | x_t) = \mathcal{N}(\mu_\theta(x_t, t), \Sigma_t$$

$$\alpha_t = 1 - \beta_t \; \overline{\alpha}_t = \prod_{s=1}^{t} \alpha_s \; \tilde{\beta}_t = \frac{1-\overline{\alpha}_{t-1}}{1-\overline{\alpha}_t} \beta_t \; \mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1-\overline{\alpha}_t}} \epsilon_\theta(x_t, t) \right)$$

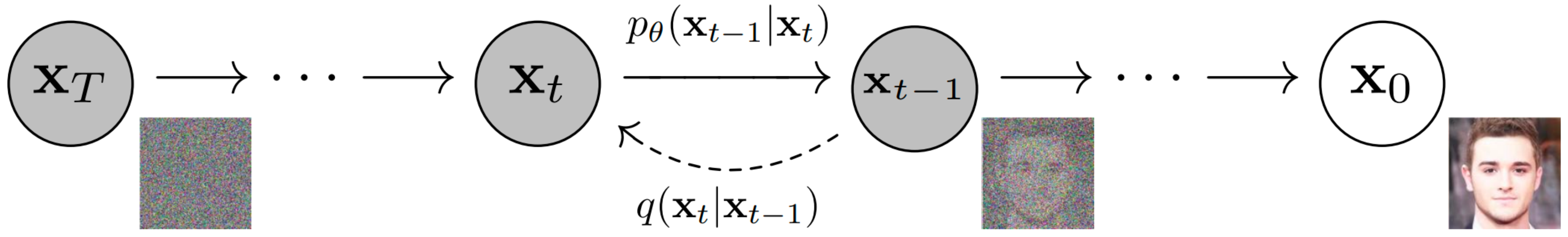


Denoising diffusion probabilistic models

Forward (diffusion) process: $q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}), q(x_t|x_{t-1}) = \mathcal{N}(\sqrt{1-\beta_t}x_{t-1}, \beta_t I)$

Backward process: $p(x_T) = \mathcal{N}(0,I), p_\theta(x_{t-1}|x_t) = \mathcal{N}(\mu_\theta(x_t, t), \Sigma_t$

where $\alpha_t = 1 - \beta_t, \bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s, \tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$, $\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)\right)$



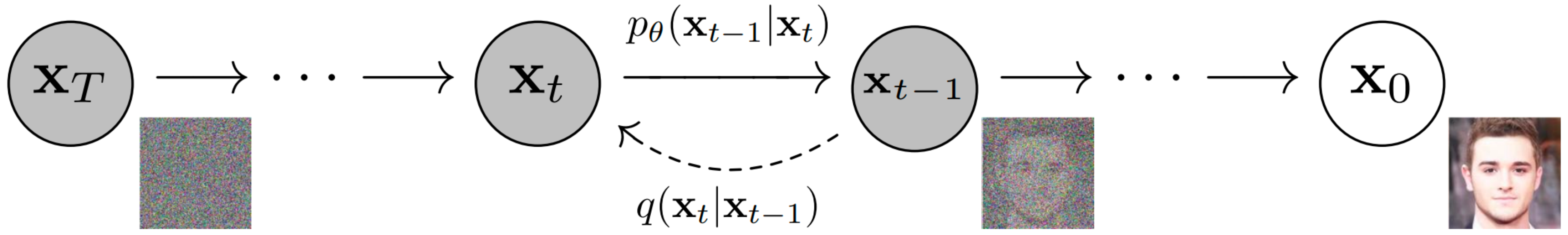
Denoising diffusion probabilistic models

Forward (diffusion) process: $q(x_{1:T} | x_0) = \prod_{t=1}^{T} q(x_t | x_{t-1}), q(x_t | x_{t-1}) = \mathcal{N}(\sqrt{1-\beta_t} x_{t-1}, \beta_t I)$

Backward process: $p(x_T) = \mathcal{N}(0,I), p_\theta(x_{t-1} | x_t) = \mathcal{N}(\mu_\theta(x_t, t), \Sigma_t$

where $\alpha_t = 1 - \beta_t, \bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s, \tilde{\beta}_t = \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} \beta_t$, $\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(x_t, t) \right)$

Training a DDPM [1]: Minimizing $\mathbb{E} \left[\sum_{t=1}^{T} \mathrm{KL}(q(x_{t-1} | x_t, x_0) \parallel p_\theta(x_{t-1} | x_t)) \right]$



[1] Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, *33*.

Text-to-image diffusion models

- Text-to-image diffusion model as conditional generation:

z $\epsilon_\theta(x_t, t)$ $\epsilon_\theta(x_t, t, z)$

$\bar{\epsilon}_\theta = w\epsilon_\theta(x_t, t, z) + (1 - w)\epsilon_\theta(x_t, t)$ w

z $\bar{\epsilon}_\theta$

Text-to-image diffusion models

- Text-to-image diffusion model as conditional generation:
    - Given text z, we learn both unconditional $\epsilon_\theta(x_t, t)$ and conditional $\epsilon_\theta(x_t, t, z)$

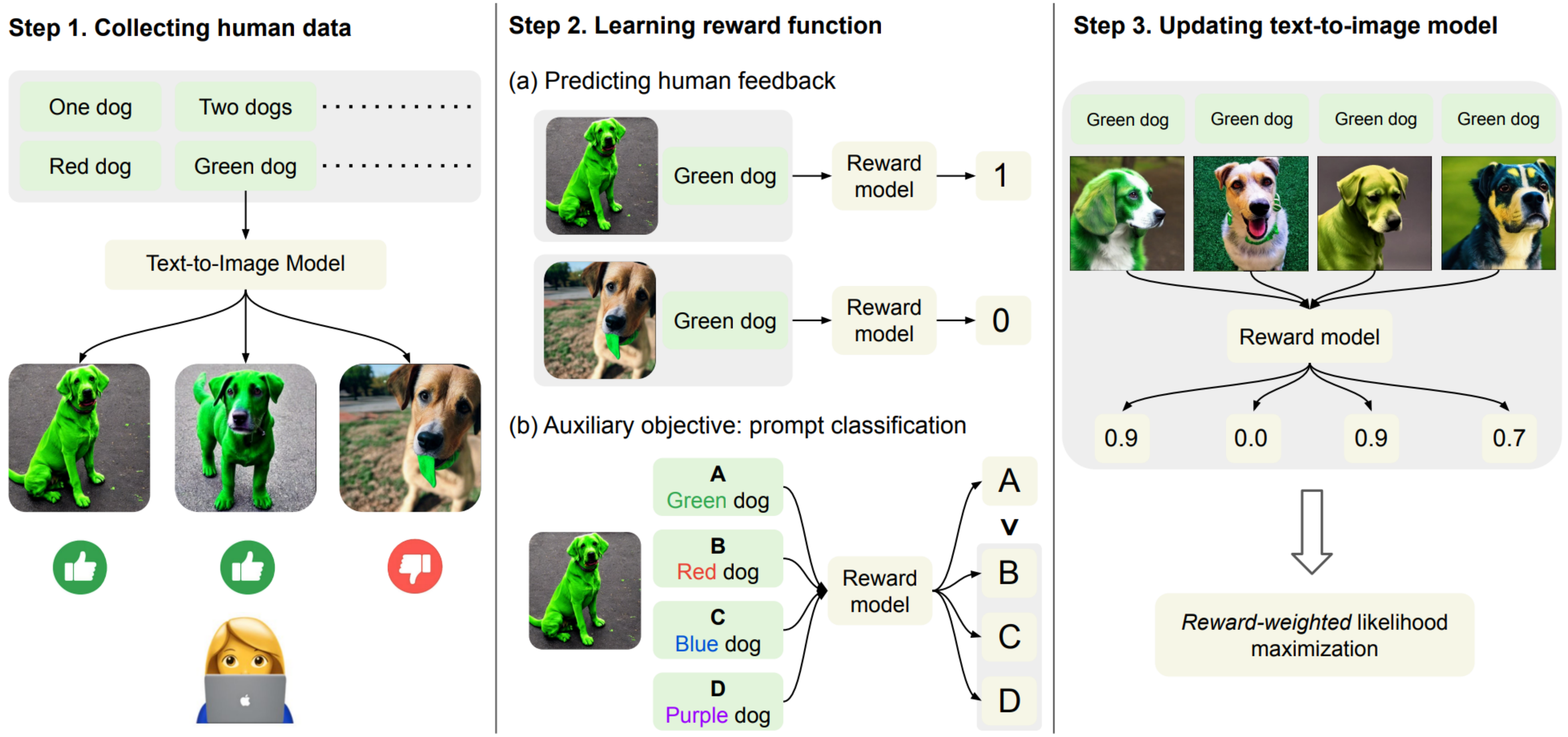
Text-to-image diffusion models

- Text-to-image diffusion model as conditional generation:
  - Given text z, we learn both unconditional $\epsilon_\theta(x_t, t)$ and conditional $\epsilon_\theta(x_t, t, z)$
  - Let $\bar{\epsilon}_\theta = w\epsilon_\theta(x_t, t, z) + (1 - w)\epsilon_\theta(x_t, t)$ where w is the guidance scale

Text-to-image diffusion models

- Text-to-image diffusion model as conditional generation:
  - Given text z, we learn both unconditional $\epsilon_\theta(x_t, t)$ and conditional $\epsilon_\theta(x_t, t, z)$
  - Let $\bar{\epsilon}_\theta = w\epsilon_\theta(x_t, t, z) + (1 - w)\epsilon_\theta(x_t, t)$ where w is the guidance scale
  - At test time, given z, the image is generated with $\bar{\epsilon}_\theta$

Supervised fine-tuning (SFT) for diffusion models



SFT [2] objective: $\mathbb{E}_{p(z)}\mathbb{E}_{p_{\text{pre}}(x_0|z)}[-r(x_0, z)\log p_\theta(x_0 \mid z)]$

[2] Lee, Kimin, Liu, Hao, Ryu, Moonkyung, Watkins, Olivia, Du, Yuqing, Boutilier, Craig, Abbeel, Pieter, Ghavamzadeh, Mohammad, and Gu, Shixiang Shane. Aligning text-to-image models using human feedback. arXiv preprint arXiv:2302.12192, 2023.

Supervised fine-tuning (SFT) for diffusion models

- As shown by Lee et al., although effective in improving the reward, SFT often induces a deterioration in image quality (e.g., over-saturated or non-photorealistic images)

(a) Seen text prompt: `Two green dogs on the table.`

Supervised fine-tuning (SFT) for diffusion models

- As shown by Lee et al., although effective in improving the reward, SFT often induces a deterioration in image quality (e.g., over-saturated or non-photorealistic images)

1. The reward re-weighted distribution is estimated ***using samples coming from the pre-trained model***, which might not be diverse and good enough to learn from

   - What if we do **online training**?

(a) Seen text prompt: `Two green dogs on the table.`

Supervised fine-tuning (SFT) for diffusion models

- As shown by Lee et al., although effective in improving the reward, SFT often induces a deterioration in image quality (e.g., over-saturated or non-photorealistic images)

1. The reward re-weighted distribution is estimated ***using samples coming from the pre-trained model***, which might not be diverse and good enough to learn from

    - What if we do **online training**?

2. Optimization of the SFT objective could leads to images too ***far away from the pre-trained distribution***, resulting in lower image quality

    - What if we add **some regularization**?

(a) Seen text prompt: `Two green dogs on the table.`

Online RL fine-tuning of diffusion models

- MDP formulation:
    - $s_t = (z, x_{T-t}), a_t = x_{T-t-1}, P_0(s_0) = (p(z), \mathcal{N}(0, I)), P(s_{t+1} \mid s_t, a_t) = (\delta_z, \delta_{a_t})$
    - $R(s_t, a_t) = \begin{cases} r(s_{t+1}) = r(x_0, z) & \text{if } t = T - 1, \\ 0 & \text{otherwise}. \end{cases}$
    - $\pi_\theta(a_t \mid s_t) = p_\theta(x_{T-t-1} \mid x_{T-t}, z)$

$$\min_\theta \mathbb{E}_{p(z)} \mathbb{E}_{p_\theta(x_0 \mid z)}[-r(x_0, z)]$$

$$\nabla_\theta \mathbb{E}_{p(z)} \mathbb{E}_{p_\theta(x_0 \mid z)}[-r(x_0, z)] = \mathbb{E}_{p(z)} \mathbb{E}_{p_\theta(x_{0:T} \mid z)} \left[-r(x_0, z) \sum_{t=1}^{T} \nabla_\theta \log p_\theta(x_{t-1} \mid x_t, z) \right]$$

Online RL fine-tuning of diffusion models

- MDP formulation:
  - $s_t = (z, x_{T-t}), a_t = x_{T-t-1}, P_0(s_0) = (p(z), \mathcal{N}(0,I)), P(s_{t+1} \,|\, s_t, a_t) = (\delta_z, \delta_{a_t})$
  - $R(s_t, a_t) = \begin{cases} r(s_{t+1}) = r(x_0, z) & \text{if} t = T - 1, \\ 0 & \text{otherwise} . \end{cases}$
  - $\pi_\theta(a_t \mid s_t) = p_\theta(x_{T-t-1} \mid x_{T-t}, z)$

$$\min_\theta \mathbb{E}_{p(z)} \mathbb{E}_{p_\theta(x_0|z)}[-r(x_0, z)]$$

$$\nabla_\theta \mathbb{E}_{p(z)} \mathbb{E}_{p_\theta(x_0|z)}[-r(x_0, z)] = \mathbb{E}_{p(z)} \mathbb{E}_{p_\theta(x_{0:T}|z)} \left[-r(x_0, z) \sum_{t=1}^{T} \nabla_\theta \log p_\theta(x_{t-1} \mid x_t, z) \right]$$

Online RL fine-tuning of diffusion models

- MDP formulation:
  - $s_t = (z, x_{T-t}), a_t = x_{T-t-1}, P_0(s_0) = (p(z), \mathcal{N}(0,I)), P(s_{t+1} \mid s_t, a_t) = (\delta_z, \delta_{a_t})$
  - $R(s_t, a_t) = \begin{cases} r(s_{t+1}) = r(x_0, z) & \text{if} t = T-1, \\ 0 & \text{otherwise}. \end{cases}$
  - $\pi_\theta(a_t \mid s_t) = p_\theta(x_{T-t-1} \mid x_{T-t}, z)$
- We can show that optimizing this MDP with policy gradient is equivalent to minimizing $\min_\theta \mathbb{E}_{p(z)} \mathbb{E}_{p_\theta(x_0|z)}[-r(x_0, z)]$ (similar to [3]):

$$\nabla_\theta \mathbb{E}_{p(z)} \mathbb{E}_{p_\theta(x_0|z)}[-r(x_0, z)] = \mathbb{E}_{p(z)} \mathbb{E}_{p_\theta(x_{0:T}|z)} \left[-r(x_0, z) \sum_{t=1}^{T} \nabla_\theta \log p_\theta(x_{t-1} \mid x_t, z) \right]$$

[3] Fan, Ying and Lee, Kangwook. Optimizing DDPM sampling with shortcut fine-tuning. *Proceedings of the 40 th International Conference on Machine Learning*.

Adding online KL regularization

- We can add the KL divergence between the fine-tuned and the pre-trained model as a regularizer to avoid overfitting the reward: $\mathrm{KL}(p_\theta(x_0 \mid z) \parallel p_{\mathrm{pre}}(x_0 \mid z))$

Adding online KL regularization

- We can add the KL divergence between the fine-tuned and the pre-trained model as a regularizer to avoid overfitting the reward: $\mathrm{KL}(p_\theta(x_0 \mid z) \parallel p_{\text{pre}}(x_0 \mid z))$

- Unfortunately, $p_\theta(x_0 \mid z)$ is not tractable, so we propose to consider an upper-bound:

$$\mathbb{E}_{p(z)}[\mathrm{KL}(p_\theta(x_0 \mid z)) \parallel p_{\text{pre}}(x_0 \mid z))] \leq \mathbb{E}_{p(z)}\left[\sum_{t=1}^{T} \mathbb{E}_{p_\theta(x_t \mid z)}[\mathrm{KL}(p_\theta(x_{t-1} \mid x_t, z) \parallel p_{\text{pre}}(x_{t-1} \mid x_t, z))] \right]$$

$$\mathbb{E}_{p(z)}\mathbb{E}_{p_\theta(x_{0:T} \mid z)}[-\alpha r(x_0, z) \sum_{t=1}^{T} \nabla_\theta \log p_\theta(x_{t-1} \mid x_t, z) + \beta \sum_{t=1}^{T} \nabla_\theta \mathrm{KL}(p_\theta(x_{t-1} \mid x_t, z) \parallel p_{\text{pre}}(x_{t-1} \mid x_t, z))]$$

Adding online KL regularization

- We can add the KL divergence between the fine-tuned and the pre-trained model as a regularizer to avoid overfitting the reward: $\mathrm{KL}(p_\theta(x_0 \mid z) \parallel p_{\mathrm{pre}}(x_0 \mid z))$

- Unfortunately, $p_\theta(x_0 \mid z)$ is not tractable, so we propose to consider an upper-bound:

$$\mathbb{E}_{p(z)}[\mathrm{KL}(p_\theta(x_0 \mid z)) \parallel p_{\mathrm{pre}}(x_0 \mid z))] \leq \mathbb{E}_{p(z)}\left[\sum_{t=1}^{T} \mathbb{E}_{p_\theta(x_t \mid z)}[\mathrm{KL}(p_\theta(x_{t-1} \mid x_t, z) \parallel p_{\mathrm{pre}}(x_{t-1} \mid x_t, z))]\right]$$

- We use the following gradient to optimize our KL-regularized RL training:

$$\mathbb{E}_{p(z)}\mathbb{E}_{p_\theta(x_{0:T} \mid z)}[-\alpha r(x_0, z)\sum_{t=1}^{T} \nabla_\theta \log p_\theta(x_{t-1} \mid x_t, z) + \beta \sum_{t=1}^{T} \nabla_\theta \mathrm{KL}(p_\theta(x_{t-1} \mid x_t, z) \parallel p_{\mathrm{pre}}(x_{t-1} \mid x_t, z))]$$

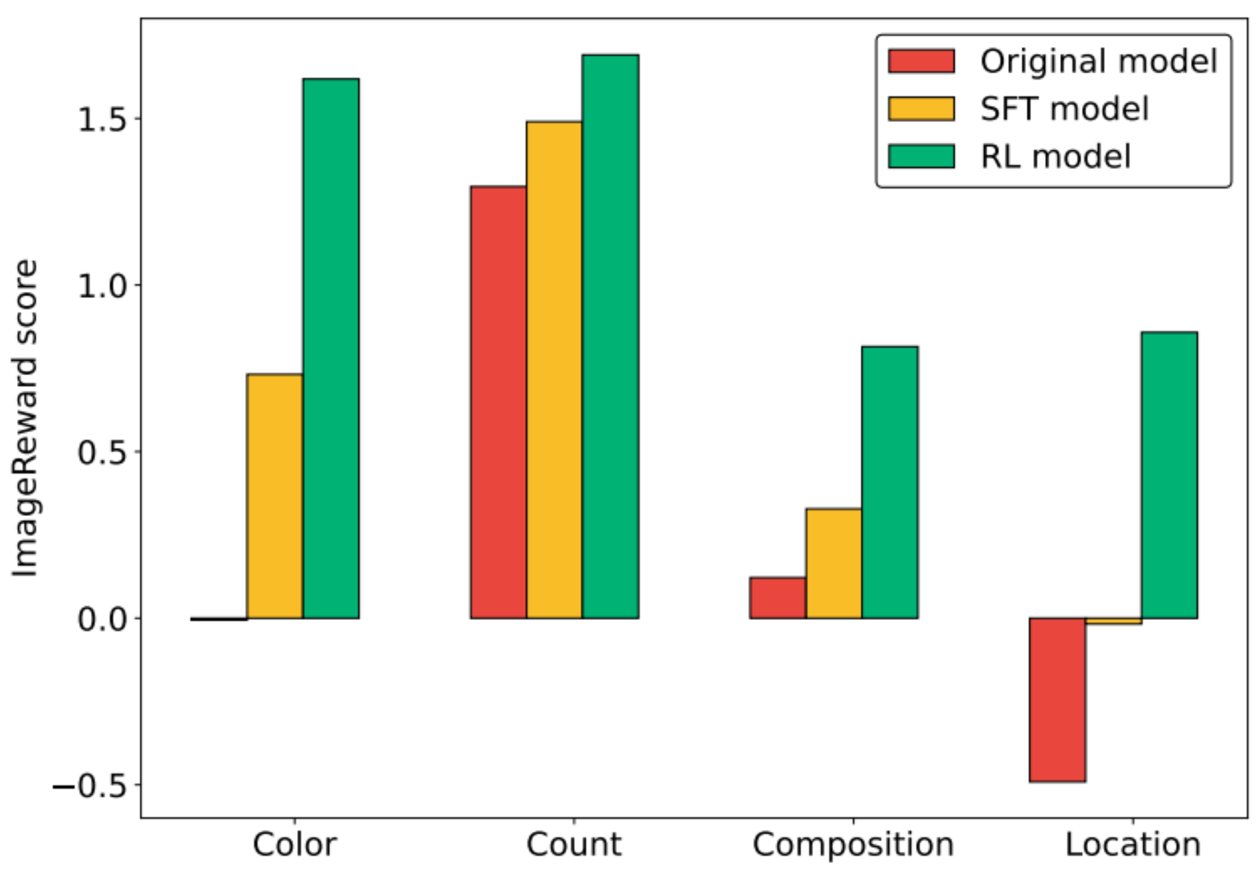
Results: SFT vs RL fine-tuning

- We compare the original model, SFT model (with supervised KL regularization) and RL model (with online KL regularization), using ImageReward [4] as the reward model.
- We focus on capabilities like generating specific color, composition, count and location

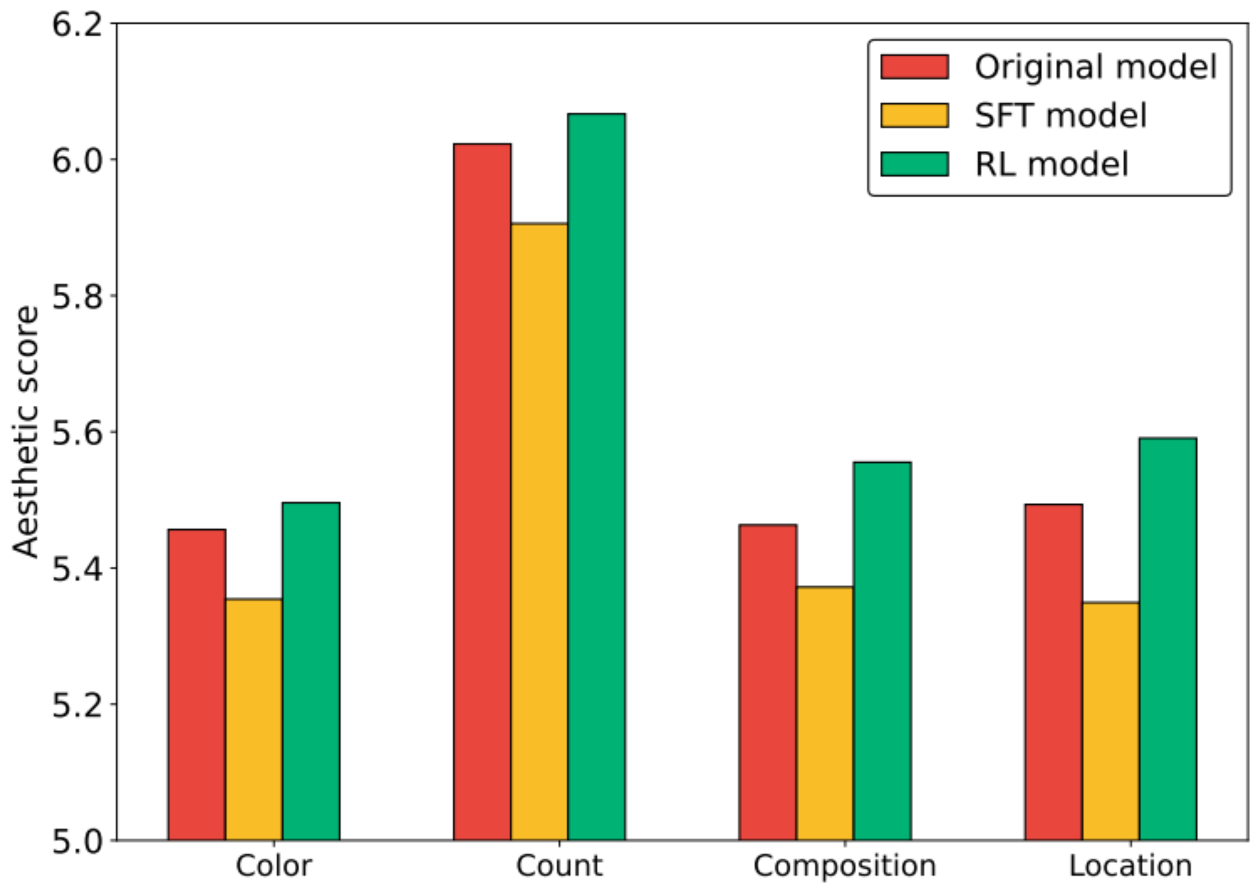
[4] Xu, Jiazheng, Liu, Xiao, Wu, Yuchen, Tong, Yuxuan, Li, Qinkai, Ding, Ming, Tang, Jie, and Dong, Yuxiao. ImageReward: Learning and evaluating human preferences for text-to-image generation. In Advances in Neural Information Processing Systems, 2023

Results: SFT vs RL fine-tuning

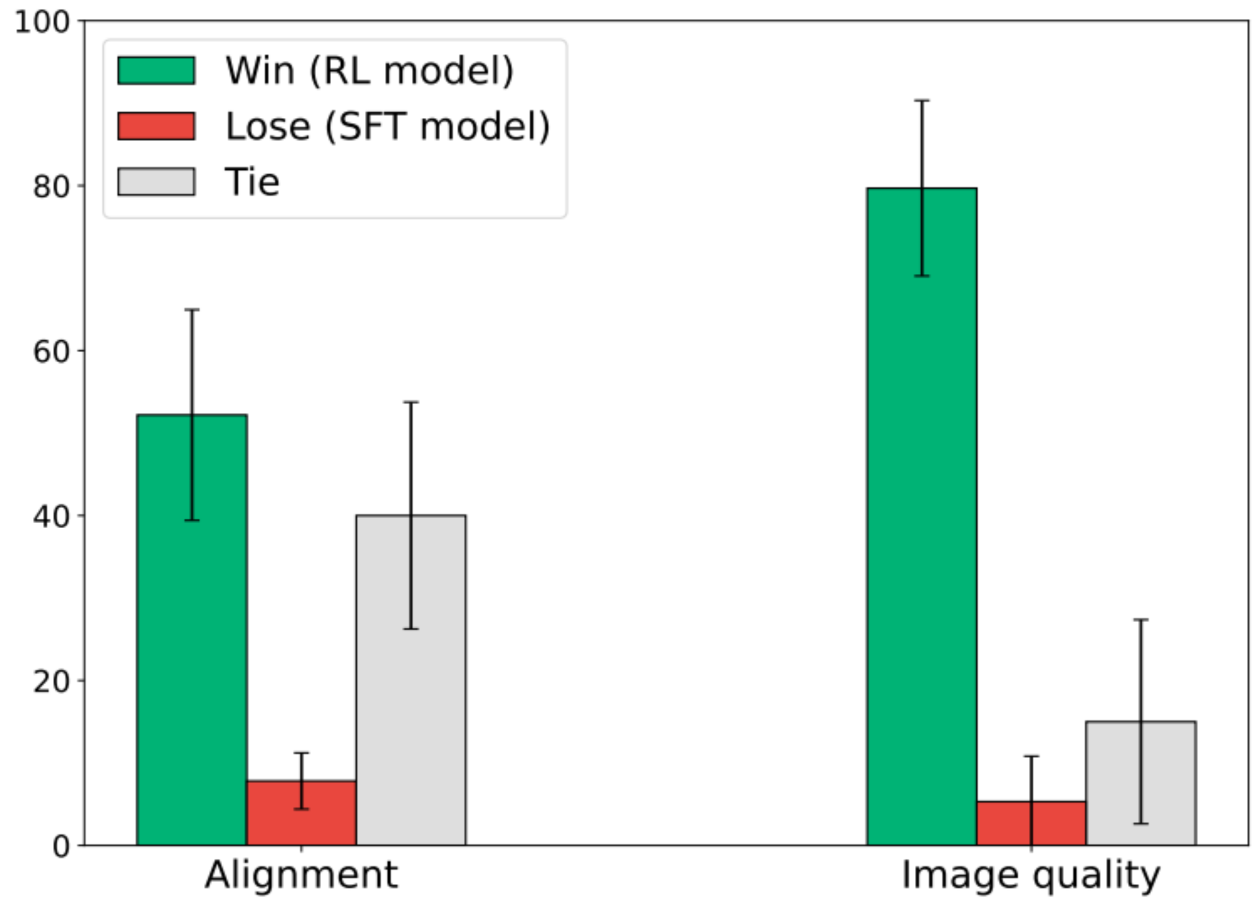
- For image quality, we adopt aesthetic score [5] as a proxy of visual quality
- Besides ImageReward and aesthetic score, we also conduct **human evaluation** on the trained models.
- We observe that compared to SFT, online fine-tuning with KL regularizaiton is more effective in **improving text-to-image alignment while maintaining high image quality**



(a) ImageReward score



(b) Aesthetic score



(c) Human evalution

[5] Schuhmann, Christoph, Beaumont, Romain, Vencu, Richard, Gordon, Cade, Wightman, Ross, Cherti, Mehdi, Coombes, Theo, Katta, Aarush, Mullis, Clayton, Wortsman, Mitchell, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. arXiv preprint arXiv:2210.08402, 2022.

Results: Fine-tuning on multiple prompts

- The proposed method is effective in optimizing rewards given a larger set of prompts
  - We also utilize an extra value function for variance reduction which shows improvement in the multi-prompt training

	MS-CoCo		Drawbench	
	Original model	RL model	Original model	RL model
ImageReward score	0.22	0.55	0.13	0.58
Aesthetic score	5.39	5.43	5.31	5.35

Table 1: ImageReward scores and Aesthetic scores from the original model, and RL fine-tuned model on multiple prompts from MS-CoCo (104 prompts) and Drawbench (183 prompts). We report the average ImageReward and Aesthetic scores across 3120 and 5490 images on MS-CoCo and Drawbench, respectively (30 images per each prompt).

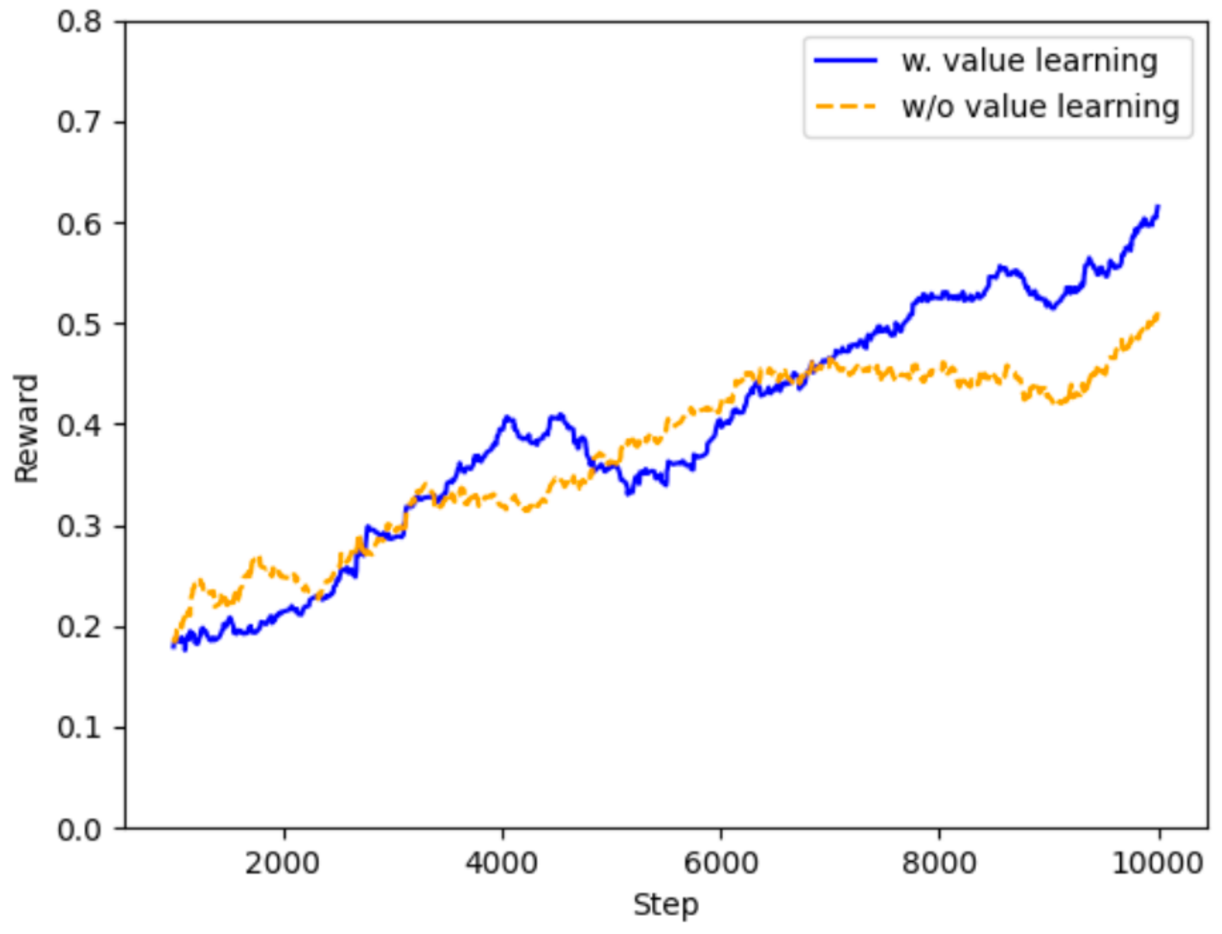


Figure 7: Learning curves with and without value learning, trained on the Drawbench prompt set: Adding value learning could result in higher reward using less time.

References

[1] Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, *33*.
[2] Lee, Kimin, Liu, Hao, Ryu, Moonkyung, Watkins, Olivia, Du, Yuqing, Boutilier, Craig, Abbeel, Pieter, Ghavamzadeh, Mohammad, and Gu, Shixiang Shane. Aligning text-to-image models using human feedback. arXiv preprint arXiv:2302.12192, 2023.
[3] Fan, Ying and Lee, Kangwook. Optimizing ddpm sampling with shortcut fine-tuning. *Proceedings of the 40 th International Conference on Machine Learning*.
[4] Xu, Jiazheng, Liu, Xiao, Wu, Yuchen, Tong, Yuxuan, Li, Qinkai, Ding, Ming, Tang, Jie, and Dong, Yuxiao. Imagereward: Learning and evaluating human preferences for text-to-image generation. In Advances in Neural Information Processing Systems, 2023.
[5] Schuhmann, Christoph, Beaumont, Romain, Vencu, Richard, Gordon, Cade, Wightman, Ross, Cherti, Mehdi, Coombes, Theo, Katta, Aarush, Mullis, Clayton, Wortsman, Mitchell, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. arXiv preprint arXiv:2210.08402, 2022.

Thank you!