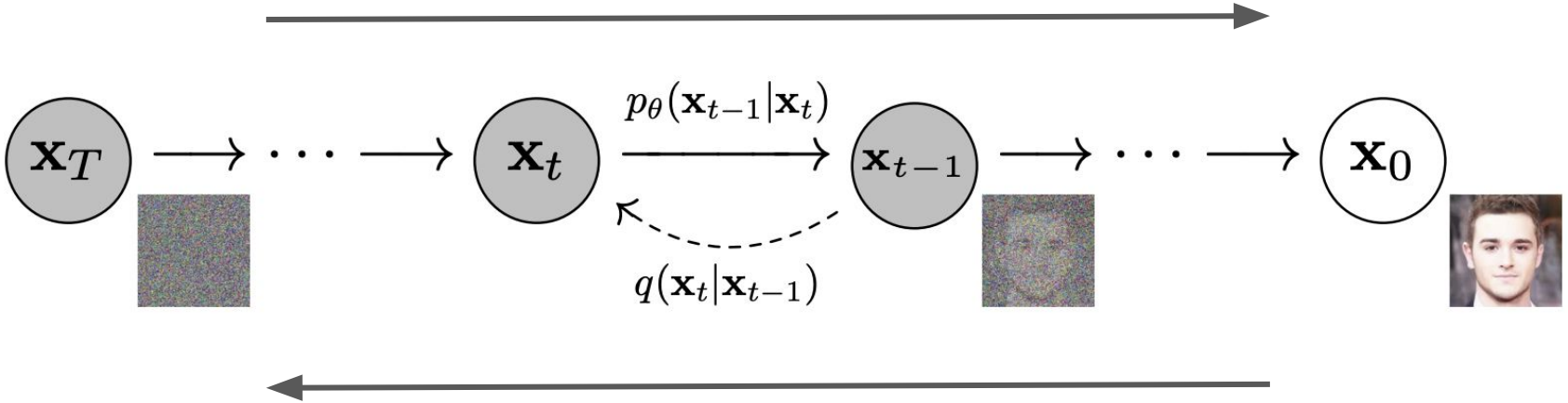
Classifier-Free Diffusion Guidance

Аржанцев Алексей
ПМИ 192

Диффузионная модель

Обратный процесс. Пытаемся убрать шум.



Прямой процесс. Добавляем шум.

Диффузионная модель

Используется модель с непрерывным временем.

Прямой процесс

$$q(\mathbf{z}_\lambda|\mathbf{x}) = \mathcal{N}(\alpha_\lambda \mathbf{x}, \sigma_\lambda^2 \mathbf{I}),\ \text{where}\ \alpha_\lambda^2 = 1/(1+e^{-\lambda}),\ \sigma_\lambda^2 = 1-\alpha_\lambda^2$$

$$q(\mathbf{z}_\lambda|\mathbf{z}_{\lambda'}) = \mathcal{N}((\alpha_\lambda/\alpha_{\lambda'})\mathbf{z}_{\lambda'}, \sigma_{\lambda|\lambda'}^2 \mathbf{I}),\ \text{where}\ \lambda < \lambda',\ \sigma_{\lambda|\lambda'}^2 = (1-e^{\lambda-\lambda'})\sigma_\lambda^2$$

Обратный процесс

$$p_\theta(\mathbf{z}_{\lambda'}|\mathbf{z}_\lambda) = \mathcal{N}(\tilde{\boldsymbol{\mu}}_{\lambda'|\lambda}(\mathbf{z}_\lambda, \mathbf{x}_\theta(\mathbf{z}_\lambda)), (\tilde{\sigma}_{\lambda'|\lambda}^2)^{1-v}(\sigma_{\lambda|\lambda'}^2)^v)$$

Минимизируем

$$\mathbb{E}_{\boldsymbol{\epsilon},\lambda}\left[\|\boldsymbol{\epsilon}_\theta(\mathbf{z}_\lambda) - \boldsymbol{\epsilon}\|_2^2\right]$$

Диффузионная модель

Во время генерации семплим лямбды и идем по ним.

$$\lambda_{\min} = \lambda_1 < \lambda_2 < \cdots < \lambda_T = \lambda_{\max}$$

$$\tilde{\mathbf{x}}_t = (\mathbf{z}_t - \sigma_{\lambda_t}\tilde{\boldsymbol{\epsilon}}_t)/\alpha_{\lambda_t}$$
$$\mathbf{z}_{t+1} \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}_{\lambda_{t+1}|\lambda_t}(\mathbf{z}_t, \tilde{\mathbf{x}}_t), (\tilde{\sigma}^2_{\lambda_{t+1}|\lambda_t})^{1-v}(\sigma^2_{\lambda_t|\lambda_{t+1}})^v)$$

Так как наша модель предсказывает шум

$$\boldsymbol{\epsilon}_\theta(\mathbf{z}_\lambda) \approx -\sigma_\lambda \nabla_{\mathbf{z}_\lambda} \log p(\mathbf{z}_\lambda)$$

Что такое guidance и зачем он нужен

Иногда диффузионная модель генерирует не слишком понятные картинки. Мы хотим улучшить “качество” получаемых картинок. При этом вероятно будет утеряно “разнообразие”.

Используемые метрики

- **FID (Fréchet inception distance)**
  Считаем “расстояние” между распределениями на последнем слое классификатора. Меньшее расстояние означает большее “разнообразие”.
- **IS (Inception score)**
  Генерим картинки и считаем матожидание KL-дивергенции между усредненным распределением и ответом на конкретном изображении. Чем больше, тем более понятные картинки.

Трейдофф FID-IS

Улучшая качество мы уменьшаем разнообразие.

Важный момент. Обе наши метрики используют классификаторы!

Classifier-guidance

Используем классификатор для улучшения качества.

При генерации модифицируем оценку диффузионной модели.

$$\tilde{\boldsymbol{\epsilon}}_\theta(\mathbf{z}_\lambda, \mathbf{c}) = \boldsymbol{\epsilon}_\theta(\mathbf{z}_\lambda, \mathbf{c}) - w\sigma_\lambda \nabla_{\mathbf{z}_\lambda} \log p_\theta(\mathbf{c}|\mathbf{z}_\lambda) \approx -\sigma_\lambda \nabla_{\mathbf{z}_\lambda} [\log p(\mathbf{z}_\lambda|\mathbf{c}) + w \log p_\theta(\mathbf{c}|\mathbf{z}_\lambda)],$$

оценка шума моделью

добавляем антиградиент оценки классификатора

вероятность картинки по классу

вероятность класса по картинке

Classifier-guidance

В итоге мы меняем распределение изображений

$$\tilde{p}_\theta(\mathbf{z}_\lambda|\mathbf{c}) \propto p_\theta(\mathbf{z}_\lambda|\mathbf{c})p_\theta(\mathbf{c}|\mathbf{z}_\lambda)^w$$

Модели с большим скором классификатора получают больший вес.
Параметр w позволяет контролировать трейдофф.

Classifier-guidance

Недостатки classifier-guidance:

1. Нужно отдельно обучать классификатор. Взять предобученный не получится, т.к. он должен быть обучен на зашумленных данных.
2. Данный метод использует градиенты классификатора, при том, что метрики тоже используют классификаторы. Таким образом classifier guidance можно считать adversarial атакой на наши метрики.
3. Идея шагать в сторону уверенности классификатора чем-то напоминает обучение GAN-ов. Возможно мы просто превращаем диффузионную модель в GAN.

Classifier-free guidance. Обучение

Хотим добавить модель не знающую про классы. Просто добавляем токен который соответствует отсутствию класса.

Algorithm 1 Joint training a diffusion model with classifier-free guidance

Require: p_{uncond}: probability of unconditional training

repeat
  $(\mathbf{x}, \mathbf{c}) \sim p(\mathbf{x}, \mathbf{c})$ ▷ Sample data with conditioning from the dataset
  $\mathbf{c} \leftarrow \varnothing$ with probability p_{uncond} ▷ Randomly discard conditioning to train unconditionally
  $\lambda \sim p(\lambda)$ ▷ Sample log SNR value
  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  $\mathbf{z}_\lambda = \alpha_\lambda \mathbf{x} + \sigma_\lambda \boldsymbol{\epsilon}$ ▷ Corrupt data to the sampled log SNR value
  Take gradient step on $\nabla_\theta \left\| \boldsymbol{\epsilon}_\theta(\mathbf{z}_\lambda, \mathbf{c}) - \boldsymbol{\epsilon} \right\|^2$ ▷ Optimization of denoising model
until converged

Classifier-free guidance. Генерация

В процессе генерации берем ответ модели с классами и вычитаем ответ модели без классов с разными весами.

Algorithm 2 Conditional sampling with classifier-free guidance

Require: w: guidance strength
Require: $\mathbf{c}$: conditioning information for conditional sampling
Require: $\lambda_1, \ldots, \lambda_T$: increasing log SNR sequence with $\lambda_1 = \lambda_{\min}, \lambda_T = \lambda_{\max}$
1: $\mathbf{z}_1 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = 1, \ldots, T$ **do**
▷ Form the classifier-free guided score at log SNR λ_t
3: $\tilde{\boldsymbol{\epsilon}}_t = (1 + w)\boldsymbol{\epsilon}_\theta(\mathbf{z}_t, \mathbf{c}) - w\boldsymbol{\epsilon}_\theta(\mathbf{z}_t)$
▷ Sampling step (could be replaced by another sampler, e.g. DDIM)
4: $\tilde{\mathbf{x}}_t = (\mathbf{z}_t - \sigma_{\lambda_t}\tilde{\boldsymbol{\epsilon}}_t)/\alpha_{\lambda_t}$
5: $\mathbf{z}_{t+1} \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}_{\lambda_{t+1}|\lambda_t}(\mathbf{z}_t, \tilde{\mathbf{x}}_t), (\tilde{\sigma}^2_{\lambda_{t+1}|\lambda_t})^{1-v}(\sigma^2_{\lambda_t|\lambda_{t+1}})^v)$ if $t < T$ else $\mathbf{z}_{t+1} = \tilde{\mathbf{x}}_t$
6: **end for**
7: **return** $\mathbf{z}_{T+1}$

Classifier-free guidance. Почему это работает

Главная идея в аналогии с classifier-guidance

$$\widetilde{\epsilon}_\theta(z_\lambda, c) = (1+w)\epsilon_\theta(z_\lambda, c) - w\epsilon_\theta(z_\lambda) = \epsilon_\theta(z_\lambda, c) + w(\epsilon_\theta(z_\lambda, c) - \epsilon_\theta(z_\lambda))$$

Распишем то что в скобках

$$\epsilon_\theta(z_\lambda, c) - \epsilon_\theta(z_\lambda) \approx -\sigma_\lambda \nabla_{z_\lambda}(\log p(z_\lambda|c) - \log p(z_\lambda))) =$$

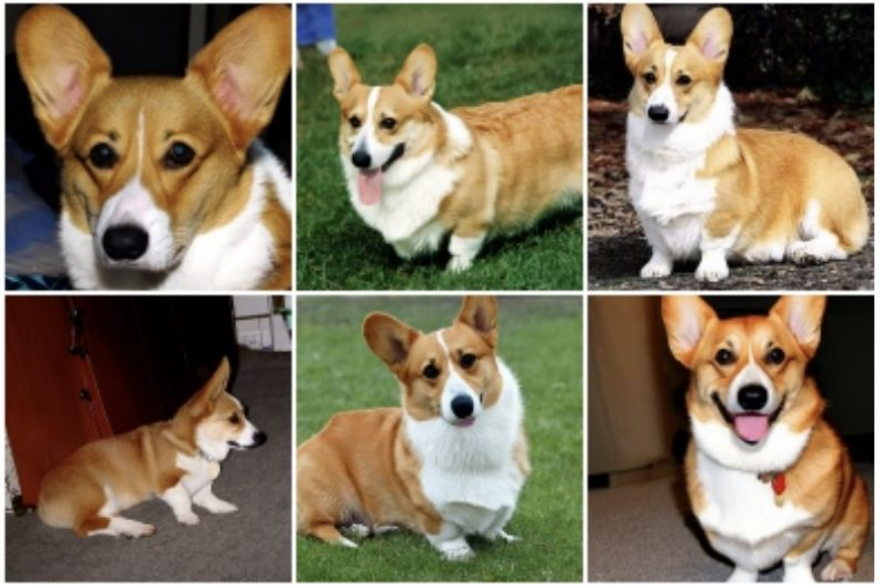
$$= -\sigma_\lambda \nabla_{z_\lambda}(\log p(c|z_\lambda) - \log p(c))) = -\sigma_\lambda \nabla_{z_\lambda} \log p(c|z_\lambda)$$

Примеры

Примеры работы с w = 0 и w = 3.

Примеры

Примеры работы с w = 0 и w = 3.



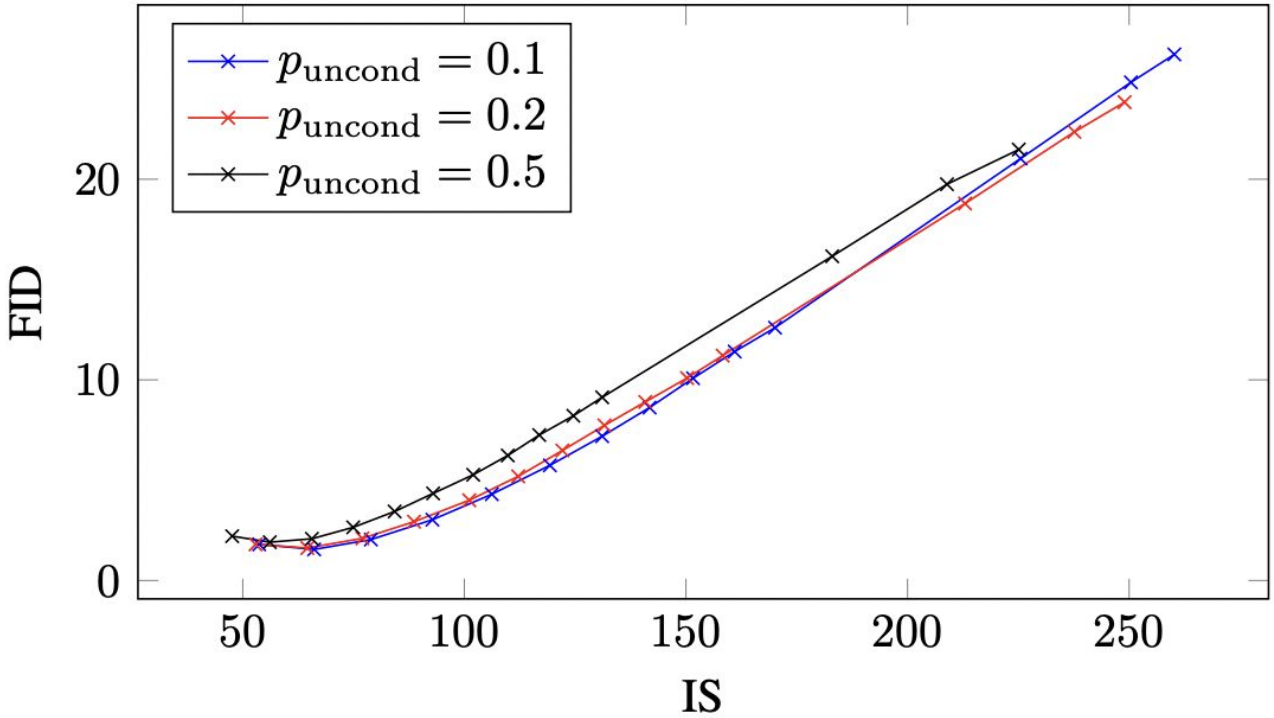
Примеры

Примеры

Эксперименты

Model	FID (↓)	IS (↑)
ADM (Dhariwal & Nichol, 2021)	2.07	-
CDM (Ho et al., 2021)	**1.48**	67.95
Ours	$p_{\text{uncond}} = 0.1/0.2/0.5$	
$w = 0.0$	1.8 / 1.8 / 2.21	53.71 / 52.9 / 47.61
$w = 0.1$	1.55 / 1.62 / 1.91	66.11 / 64.58 / 56.1
$w = 0.2$	2.04 / 2.1 / 2.08	78.91 / 76.99 / 65.6
$w = 0.3$	3.03 / 2.93 / 2.65	92.8 / 88.64 / 74.92
$w = 0.4$	4.3 / 4 / 3.44	106.2 / 101.11 / 84.27
$w = 0.5$	5.74 / 5.19 / 4.34	119.3 / 112.15 / 92.95
$w = 0.6$	7.19 / 6.48 / 5.27	131.1 / 122.13 / 102
$w = 0.7$	8.62 / 7.73 / 6.23	141.8 / 131.6 / 109.8
$w = 0.8$	10.08 / 8.9 / 7.25	151.6 / 140.82 / 116.9
$w = 0.9$	11.41 / 10.09 / 8.21	161 / 150.26 / 124.6
$w = 1.0$	12.6 / 11.21 / 9.13	170.1 / 158.29 / 131.1
$w = 2.0$	21.03 / 18.79 / 16.16	225.5 / 212.98 / 183
$w = 3.0$	24.83 / 22.36 / 19.75	250.4 / 237.65 / 208.9
$w = 4.0$	26.22 / 23.84 / 21.48	**260.2** / 248.97 / 225.1

Эксперименты



Эксперименты

Model	FID (↓)	IS (↑)
BigGAN-deep, max IS (Brock et al., 2019)	25	253
BigGAN-deep (Brock et al., 2019)	5.7	124.5
CDM (Ho et al., 2021)	3.52	128.8
LOGAN (Wu et al., 2019)	3.36	148.2
ADM-G (Dhariwal & Nichol, 2021)	2.97	-
Ours	$T = 128/256/1024$	
$w = 0.0$	8.11 / 7.27 / 7.22	81.46 / 82.45 / 81.54
$w = 0.1$	5.31 / 4.53 / 4.5	105.01 / 106.12 / 104.67
$w = 0.2$	3.7 / 3.03 / 3	130.79 / 132.54 / 130.09
$w = 0.3$	3.04 / **2.43** / **2.43**	156.09 / 158.47 / 156
$w = 0.4$	3.02 / 2.49 / 2.48	183.01 / 183.41 / 180.88
$w = 0.5$	3.43 / 2.98 / 2.96	206.94 / 207.98 / 204.31
$w = 0.6$	4.09 / 3.76 / 3.73	227.72 / 228.83 / 226.76
$w = 0.7$	4.96 / 4.67 / 4.69	247.92 / 249.25 / 247.89
$w = 0.8$	5.93 / 5.74 / 5.71	265.54 / 267.99 / 265.52
$w = 0.9$	6.89 / 6.8 / 6.81	280.19 / 283.41 / 281.14
$w = 1.0$	7.88 / 7.86 / 7.8	295.29 / 297.98 / 294.56
$w = 2.0$	15.9 / 15.93 / 15.75	378.56 / 377.37 / 373.18
$w = 3.0$	19.77 / 19.77 / 19.56	409.16 / 407.44 / 405.68
$w = 4.0$	21.55 / 21.53 / 21.45	**422.29** / 421.03 / 419.06

Статья

- https://arxiv.org/abs/2207.12598