上海科技大学
ShanghaiTech University

Dive into Diffusion Models

Group Meeting

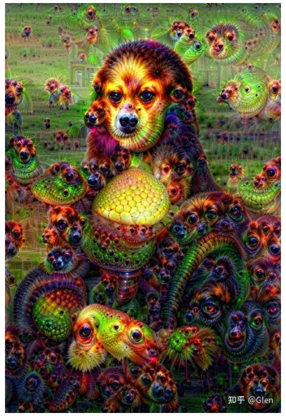
Yang Junjie, Lu Feiyang, ASPIRE Lab

05-10, 2024

立志成才报国裕民

Diffusion Model Intro

2015年Google发布Deep Dream



2016年提出Diffusion Models

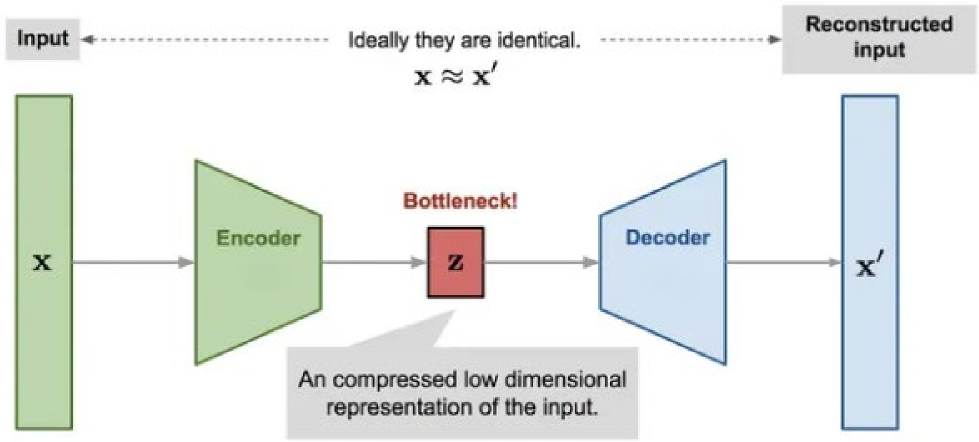
2022 年3月 Midjouney

2022 年 4 月OpenAI 发布DALL-E 2

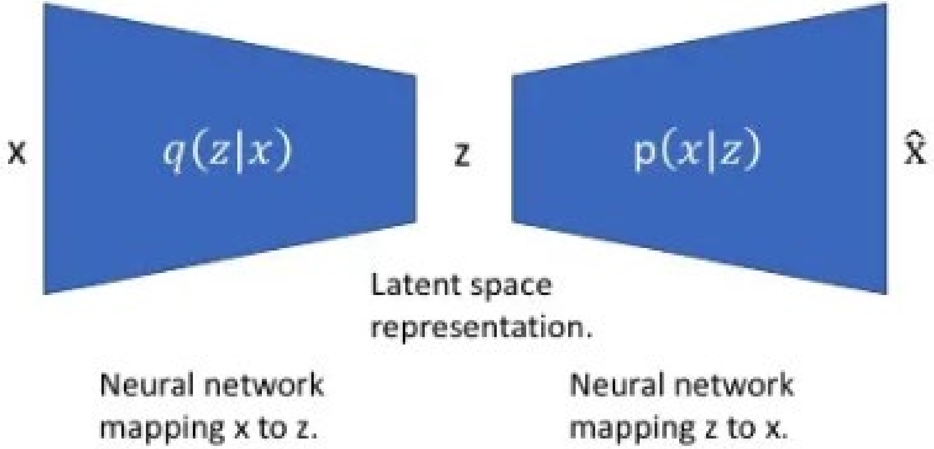
2022年7月 发布Stable diffus

在 AI 艺术生成器的发展历程中，DeepDream 和 DALLE 是两个具有里程碑意义的模型。DeepDream 根据神经网络学到的表征来生成图像。而 DALLE 结合了将图像映射到低维标记的离散变分自编码器（dVAE）和自回归建模文本和图像词元的 Transformer 模型。

1.AutoEncoder



2.VAE



VAE损失函数推导：

$$
\begin{aligned}
&KL(q_\theta(z|x)||p(z|x)) \\
&= \int q_\theta(z|x)\ln\frac{q_\theta(z|x)}{p(z|x)}\mathrm{d}z \\
&= \mathbb{E}_{z\sim q_\theta(z|x)}[\ln\frac{q_\theta(z|x)}{p(z|x)}] \\
&= \mathbb{E}_{z\sim q_\theta(z|x)}[\ln q_\theta(z|x) - \ln p(z|x)] \\
&= \mathbb{E}_{z\sim q_\theta(z|x)}[\ln q_\theta(z|x) - \ln\frac{p(x|z)p(z)}{p(x)}] \\
&= \mathbb{E}_{z\sim q_\theta(z|x)}[\ln q_\theta(z|x) - \ln p(z) - \ln p(x|z)] + \ln p(x) \\
&= KL(q_\theta(z|x)||p(z)) - \mathbb{E}_{z\sim q_\theta(z|x)}[\ln p(x|z)] + \ln p(x)
\end{aligned}
$$

整理后得到：

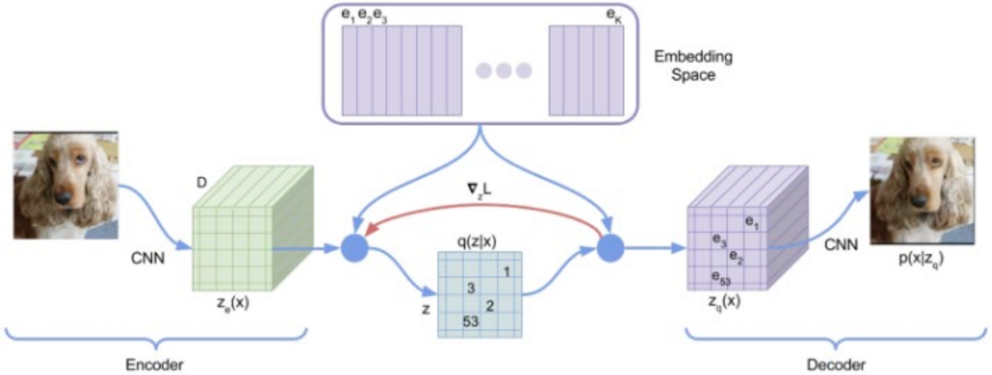
$$\ln p(x) - KL(q_\theta(z|x)||p(z|x)) = \mathbb{E}_{z\sim q_\theta(z|x)}[\ln p(x|z)] - KL(q_\theta(z|x)||p(z))$$

$\ln p(x) \geq ELBO$，其中 $ELBO = \mathbb{E}_{z\sim q_\theta(z|x)}[\ln p(x|z)] - KL(q_\theta(z|x)||p(z))$

为了方便我们假设p(z)~Normal

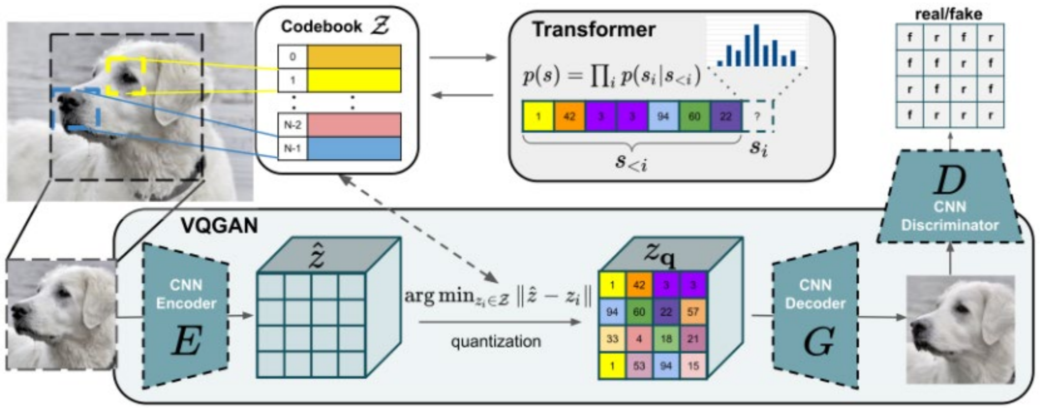
$$KL(q_\theta(z|x)||\mathcal{N}(0,I)) = \frac{1}{2}(-\ln\sigma^2 + \sigma^2 + \mu^2 - 1)$$

3.VQ-VAE



损失函数为 $\mathcal{L}_{\mathrm{VQ}}(E,G,\mathcal{Z}) = \|x-\hat{x}\|^2 + \|\mathrm{sg}[E(x)] - z_{\mathbf{q}}\|_2^2 + \beta\|\mathrm{sg}[z_{\mathbf{q}}] - E(x)\|_2^2$

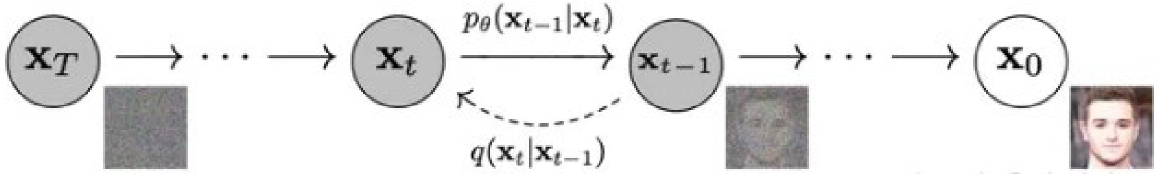
4.VQGAN



损失函数在VQVAE模型的损失函数上加上一部分

$$\mathcal{L}_{\mathrm{GAN}}(\{E,G,\mathcal{Z}\},D) = [\log D(x) + \log(1-D(\hat{x}))]$$

5.DDPM



加噪过程：$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I})$

可推导出 $q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I})$

去噪过程：$q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}(\mathbf{x}_t,\mathbf{x}_0), \tilde{\beta}_t\mathbf{I})$

$$\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\cdot\beta_t \qquad \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t,\mathbf{x}_0) = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0$$

用网络去拟合去噪过程

$$\begin{aligned}\log p_\theta(\mathbf{x}_0) &= \log\int p_\theta(\mathbf{x}_{0:T})d\mathbf{x}_{1:T}\\ &= \log\int \frac{p_\theta(\mathbf{x}_{0:T})q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}d\mathbf{x}_{1:T}\\ &\geq \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}[\log\frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}]\end{aligned}$$

$$L = -L_{\mathrm{VLB}} = \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}[-\log\frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}] = \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}[\log\frac{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{p_\theta(\mathbf{x}_{0:T})}]$$

$$= \underbrace{D_{\mathrm{KL}}(q(\mathbf{x}_T|\mathbf{x}_0) \parallel p_\theta(\mathbf{x}_T))}_{L_T} + \sum_{t=2}^{T} \underbrace{\mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} \left[D_{\mathrm{KL}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \parallel p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)) \right]}_{L_{t-1}} - \underbrace{\mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} \log p_\theta(\mathbf{x}_0|\mathbf{x}_1)}_{L_0}$$

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0, \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} \left[\frac{1}{2\sigma_t^2} \left\| \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t(\mathbf{x}_0, \epsilon) - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon \right) - \boldsymbol{\mu}_\theta(\mathbf{x}_t(\mathbf{x}_0, \epsilon), t) \right\|^2 \right]$$

$$= \mathbb{E}_{\mathbf{x}_0, \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} \left[\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1-\bar{\alpha}_t)} \left\| \epsilon - \epsilon_\theta(\mathbf{x}_t(\mathbf{x}_0, \epsilon), t) \right\|^2 \right]$$

$$= \mathbb{E}_{\mathbf{x}_0, \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} \left[\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1-\bar{\alpha}_t)} \left\| \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t) \right\|^2 \right]$$

$$L_{t-1}^{\text{simple}} = \mathbb{E}_{\mathbf{x}_0, \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} \left[\left\| \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t) \right\|^2 \right]$$

6.DDIM

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0, \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t} \cdot \beta_t \mathbf{I})$$

修改后 $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_{t-1}-\tilde{\beta}_t} \cdot \frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0}{\sqrt{1-\bar{\alpha}_t}}, \tilde{\beta}_t \mathbf{I})$

其中 $\tilde{\beta}_t(\eta) = \eta \frac{(1-\bar{\alpha}_{t-1})}{(1-\bar{\alpha}_t)} \cdot \beta_t$ $\eta = 0$，模型是DDIM；$\eta = 1$，模型是DDPM

	S	CIFAR10 (32 × 32) 10	20	50	100	1000	CelebA (64 × 64) 10	20	50	100	1000
η	0.0	**13.36**	**6.84**	**4.67**	**4.16**	4.04	**17.33**	**13.73**	**9.17**	**6.53**	3.51
	0.2	14.04	7.11	4.77	4.25	4.09	17.66	14.11	9.51	6.79	3.64
	0.5	16.66	8.35	5.25	4.46	4.29	19.86	16.06	11.01	8.09	4.28
	1.0	41.07	18.36	8.01	5.78	4.73	33.12	26.03	18.48	13.93	5.98
	$\hat{\sigma}$	367.43	133.37	32.72	9.99	**3.17**	299.71	183.83	71.71	45.20	**3.26**

7.classifier guidance

Algorithm 1 Classifier guided diffusion sampling, given a diffusion model $(\mu_\theta(x_t), \Sigma_\theta(x_t))$, classifier $p_\phi(y|x_t)$, and gradient scale s.

Input: class label y, gradient scale s
$x_T \leftarrow$ sample from $\mathcal{N}(0, \mathbf{I})$
for all t from T to 1 **do**
  $\mu, \Sigma \leftarrow \mu_\theta(x_t), \Sigma_\theta(x_t)$
  $x_{t-1} \leftarrow$ sample from $\mathcal{N}(\mu + s\Sigma \nabla_{x_t} \log p_\phi(y|x_t), \Sigma)$
end for
return x_0

本质核心就是利用一个分类器提供分类梯度，用于指导Diffusion Model合理采样

8.classifier-free guidance

通过梯度更新图像会导致对抗攻击效应，生成图像可能会通过人眼不可察觉的细节欺骗分类器，实际上并没有按条件生成。

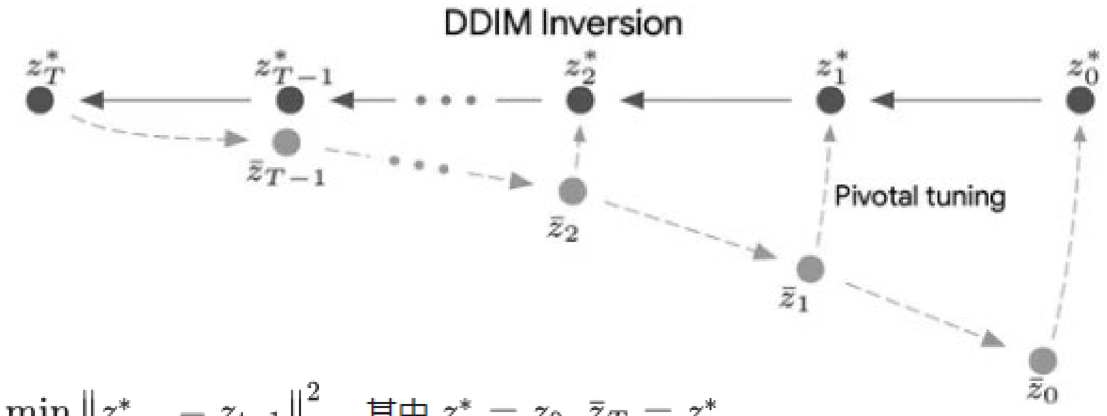
$$\tilde{\varepsilon}_\theta(z_t, t, \mathcal{C}, \varnothing) = w \cdot \varepsilon_\theta(z_t, t, \mathcal{C}) + (1 - w) \cdot \varepsilon_\theta(z_t, t, \varnothing)$$

9.DDIM Inversion

$$x_t = \frac{\sqrt{\overline{\alpha_t}}}{\sqrt{\overline{\alpha_t - 1}}}\left(x_{t-1} - \sqrt{1 - \overline{\alpha_{t-1}}}\,\epsilon_t\right) + \sqrt{1 - \overline{\alpha_t}}\,\epsilon_t$$

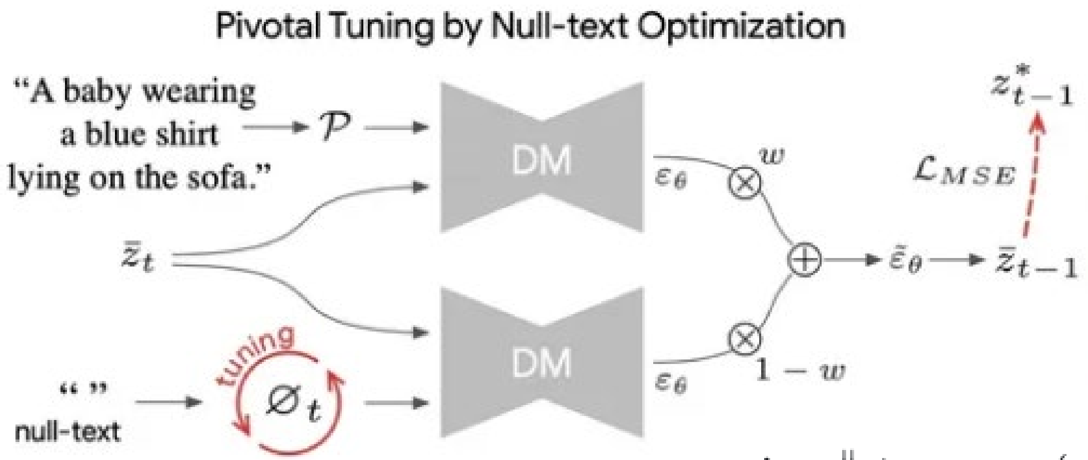
Inversion是为了找到一个latent embedding，使其能够经过生成器得到目标图像。而对于Diffusion来说，Inversion就是找到一个噪音，使得以该噪音为起点，经过采样得到目标图像。

10.Pivotal inversion



$\min \left\| z^*_{t-1} - z_{t-1} \right\|_2^2$，其中 $z_0^* = z_0, \bar{z}_T = z_T^*$

11.Null-text Optimization



$\min_{\varnothing_t} \left\| z^*_{t-1} - z_{t-1}\left(\bar{z}_t, \varnothing_t, \mathcal{C}\right) \right\|_2^2$

在实践中，DDIM Inversion每一步都会产生误差，对于无条件扩散模型，累积误差可以忽略。但是对基于classifier-free guidance的扩散模型，累积误差会不断增加，DDIM Inversion最终获得的噪声向量可能会偏离高斯分布，再经过DDIM采样，最终生成的图像会严重偏离原图像，并可能产生视觉伪影。因此，作者提出Pivotal Inversion来解决classifier-free guidance扩散模型误差累积的问题。

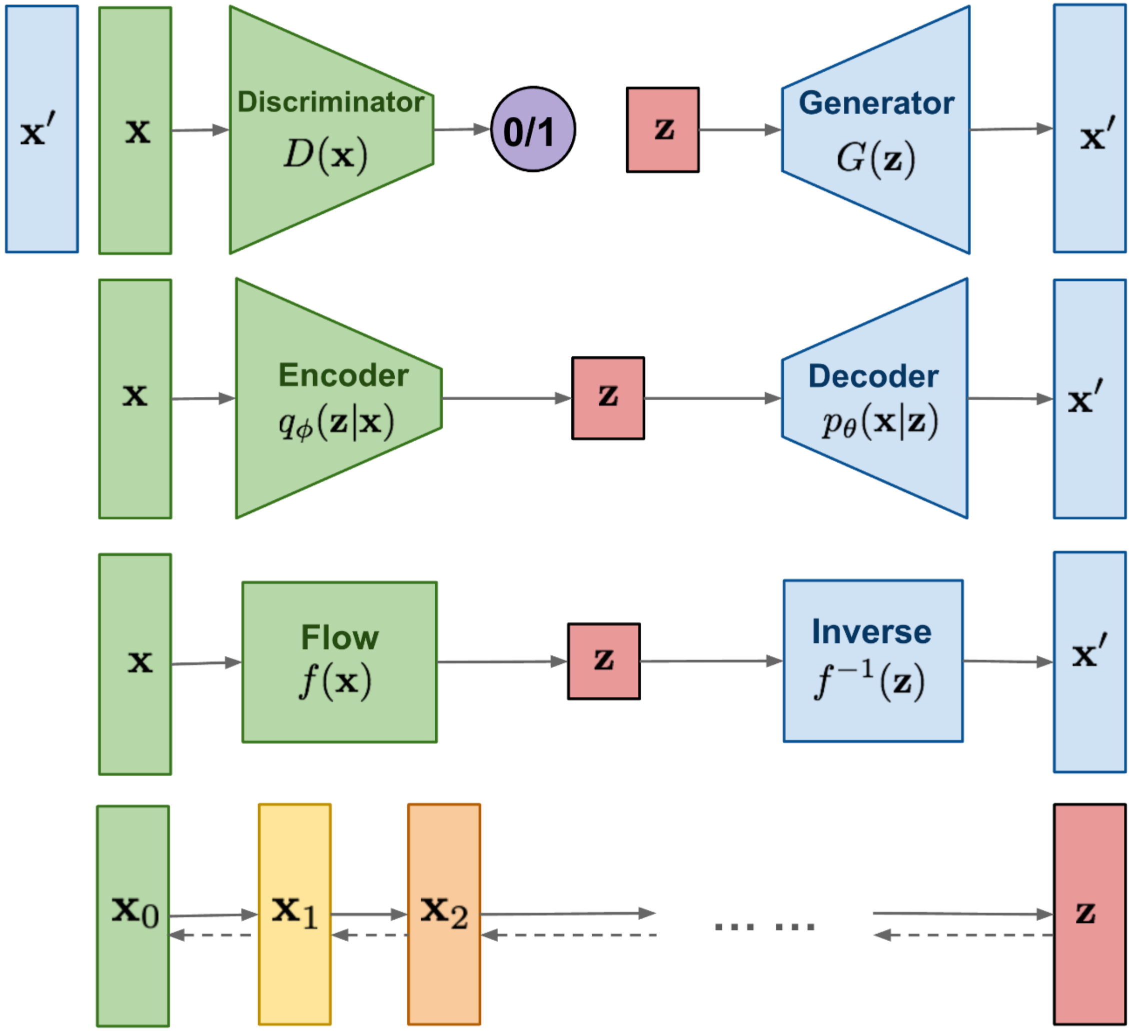
Generative models 区别回顾

GAN: Adversarial training

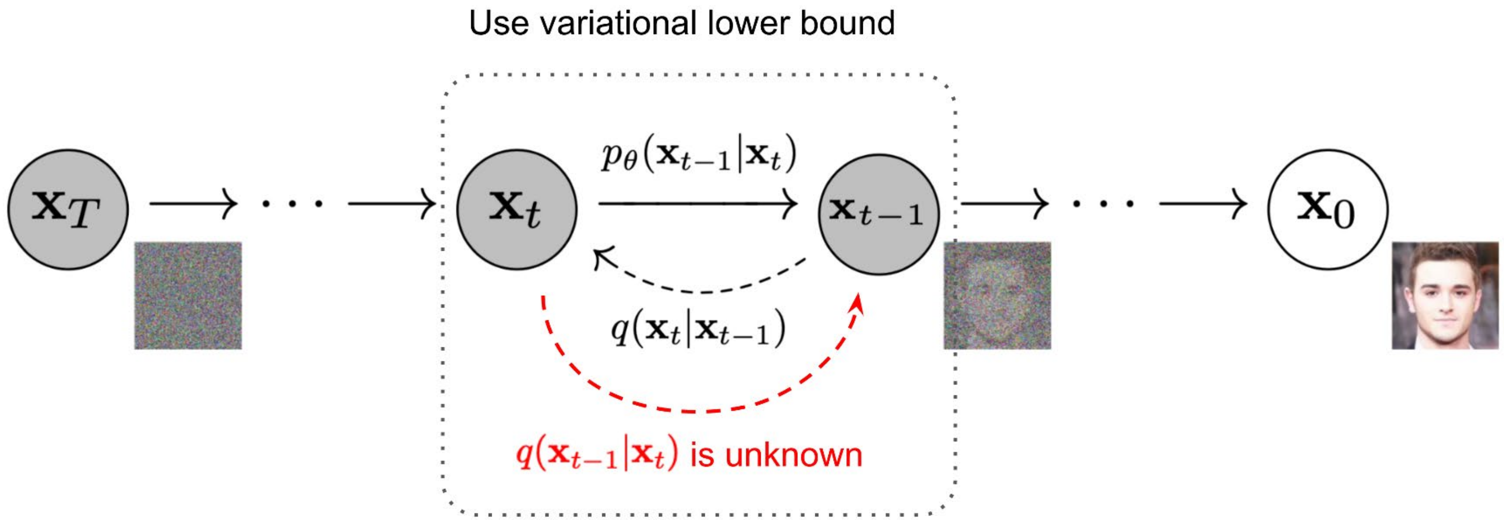
VAE: maximize variational lower bound

Flow-based models: Invertible transform of distributions

Diffusion models: Gradually add Gaussian noise and then reverse



DDPM 前后传播回顾



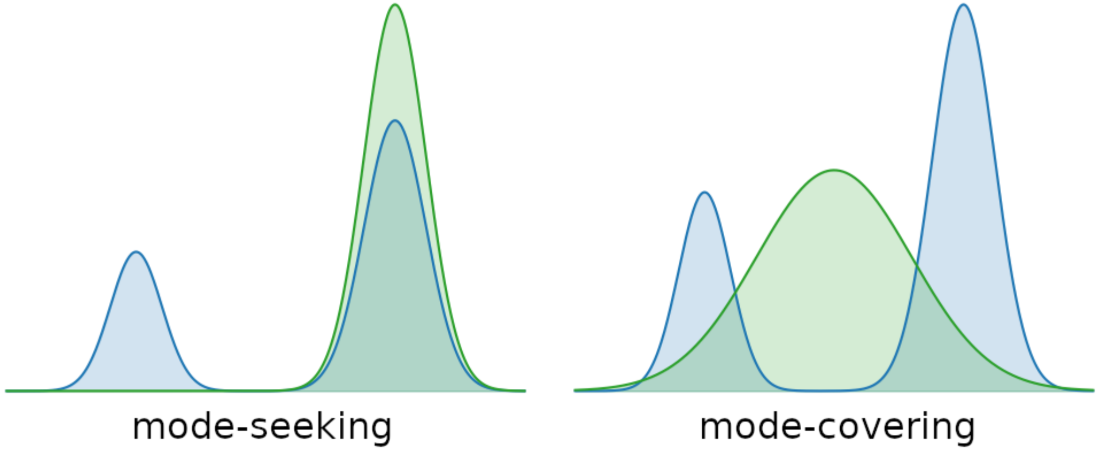
Algorithm 1 Training

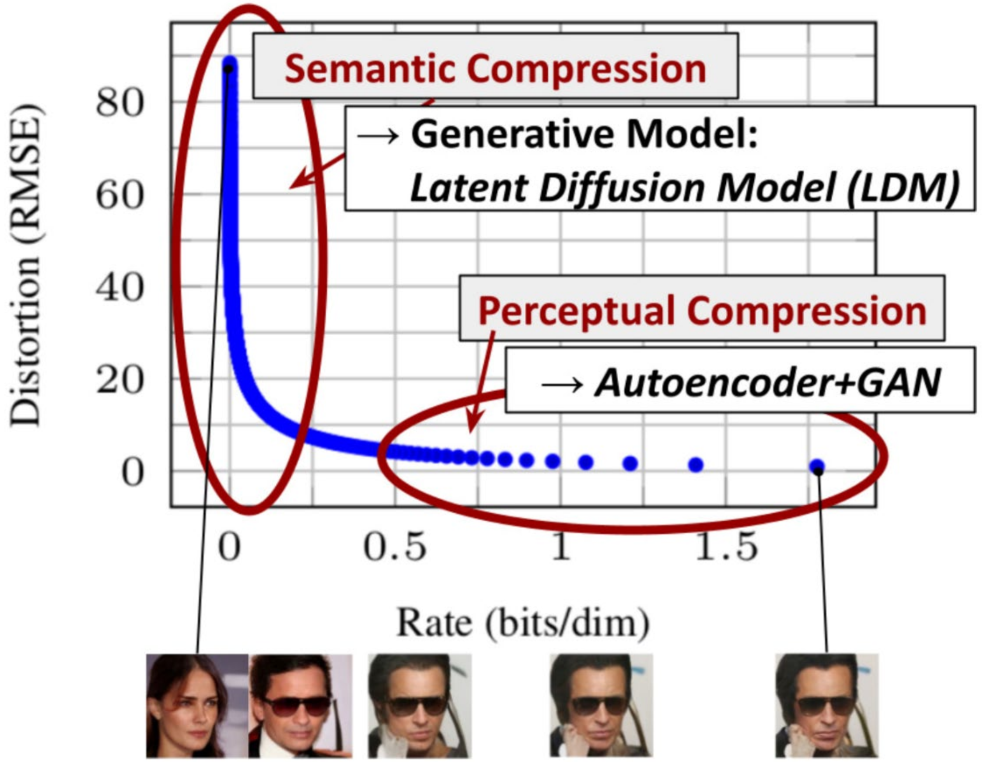
1: **repeat**
2: $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
3: $t \sim \mathrm{Uniform}(\{1, \ldots, T\})$
4: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
5: Take gradient descent step on
$$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$$
6: **until** converged

Algorithm 2 Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T, \ldots, 1$ **do**
3: $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
4: $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) + \sigma_t \mathbf{z}$
5: **end for**
6: **return** $\mathbf{x}_0$

在 Latent space 推理的可能性





创造具体可行的 LDM 框架

$$L_{DM} = \mathbb{E}_{x,\epsilon\mathcal{N}(0,1),t}[\| \epsilon - \epsilon_\theta(x_t, t)\|_2^2] \longrightarrow L_{LDM} = \mathbb{E}_{\mathrm{E}(x),\epsilon\ \mathcal{N}(0,1),t}[\| \epsilon - \epsilon_\theta(z_t, t)\|_2^2]$$

$$\mathrm{Attention}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\right)\cdot\mathbf{V}$$

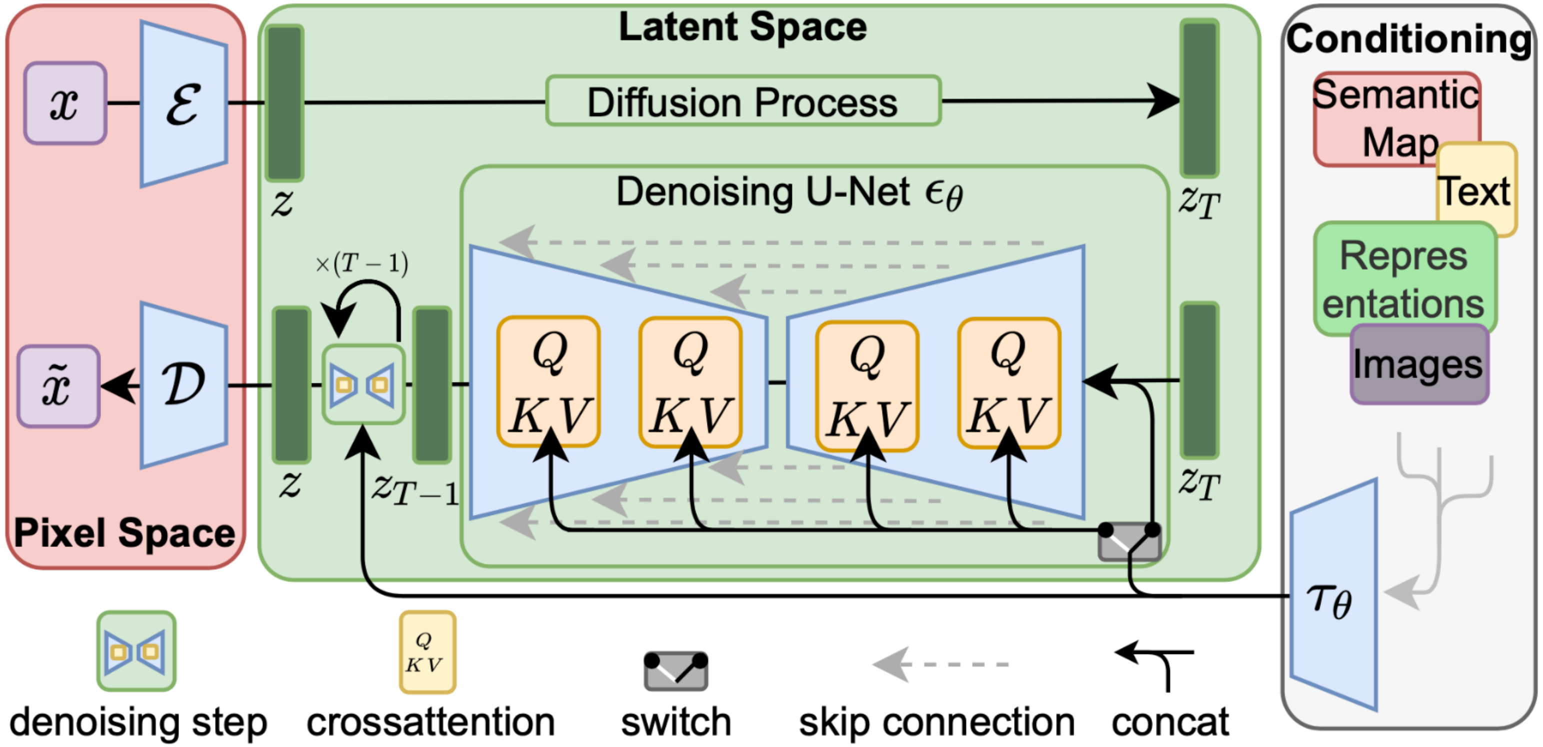
$$\text{where } \mathbf{Q} = \mathbf{W}_Q^{(i)}\cdot\varphi_i(\mathbf{z}_i),$$

$$\mathbf{K} = \mathbf{W}_K^{(i)}\cdot\tau_\theta(y),$$

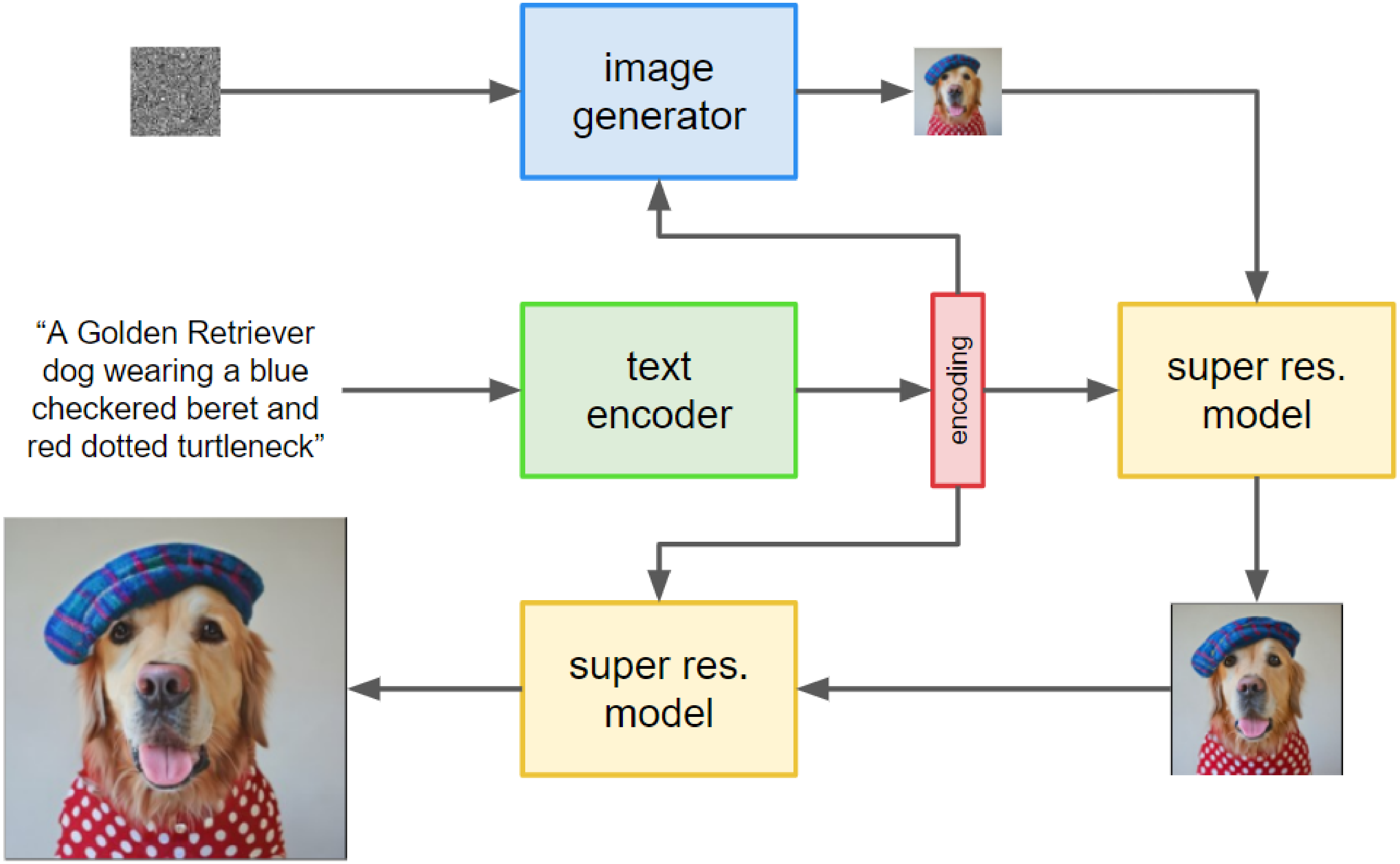
$$\mathbf{V} = \mathbf{W}_V^{(i)}\cdot\tau_\theta(y)$$

$$\text{and } \mathbf{W}_Q^{(i)} \in \mathbb{R}^{d\times d_\epsilon^i}, \mathbf{W}_K^{(i)}, \mathbf{W}_V^{(i)} \in \mathbb{R}^{d\times d_\tau},$$

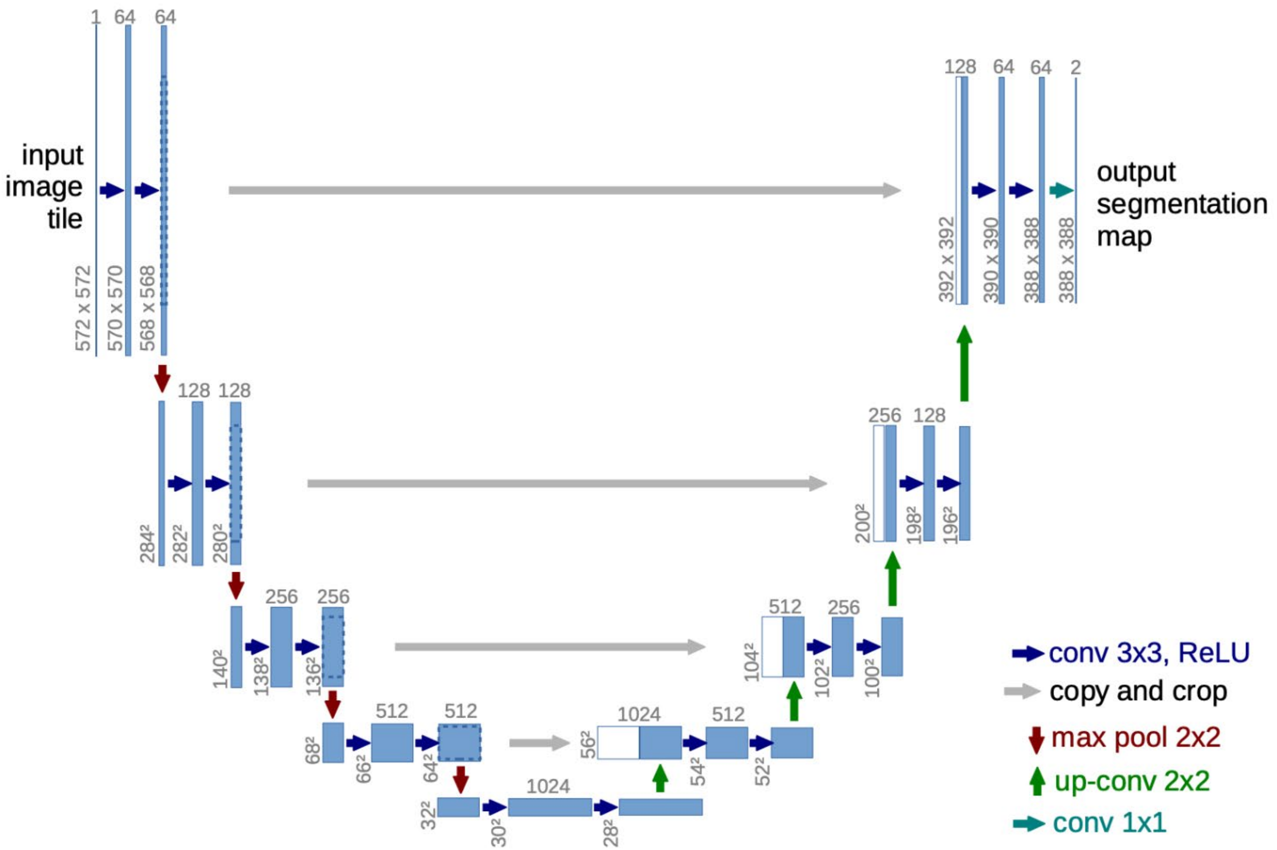
$$\varphi_i(\mathbf{z}_i) \in \mathbb{R}^{N\times d_\epsilon^i}, \tau_\theta(y) \in \mathbb{R}^{M\times d_\tau}$$

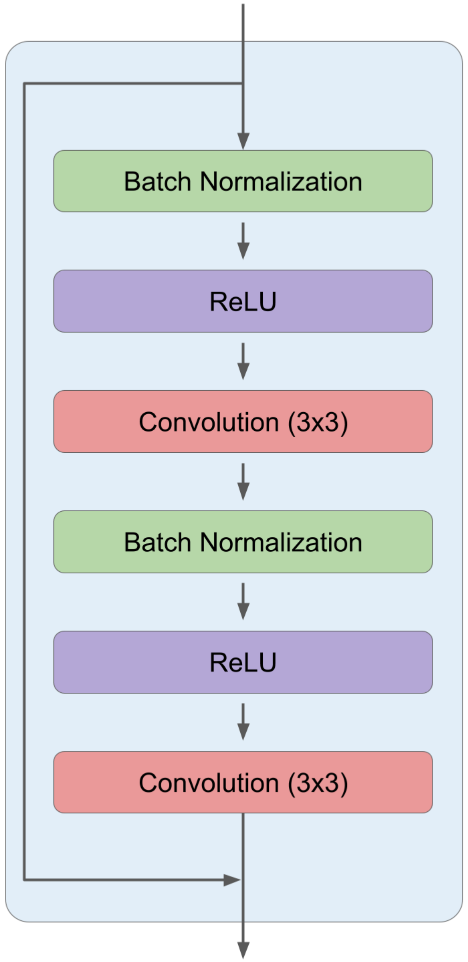


让推导更可控的Conditioning



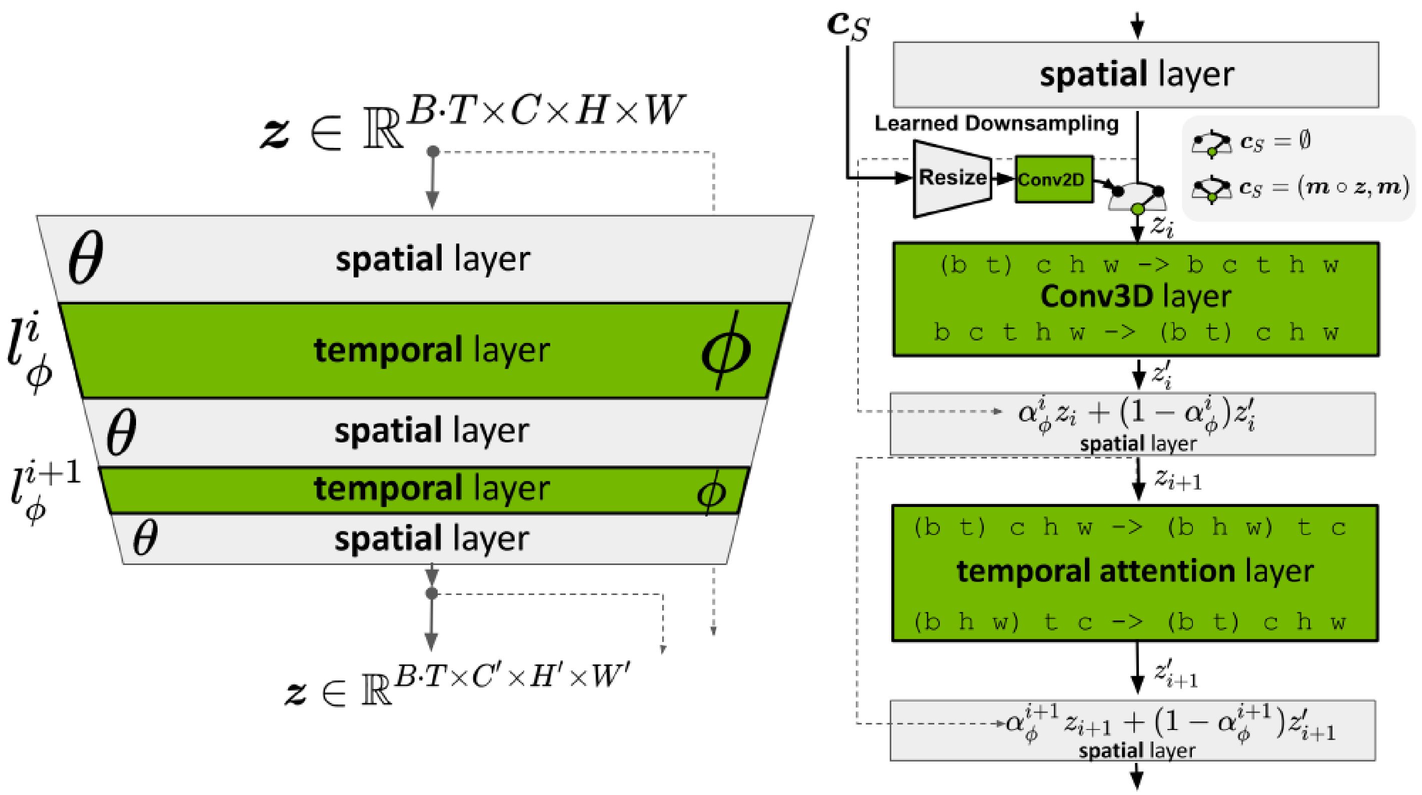
生成的临门一脚，U-net





Align Latents，进入 Video

上海科技大学
ShanghaiTech University



立志成才报国裕民

THANK YOU!