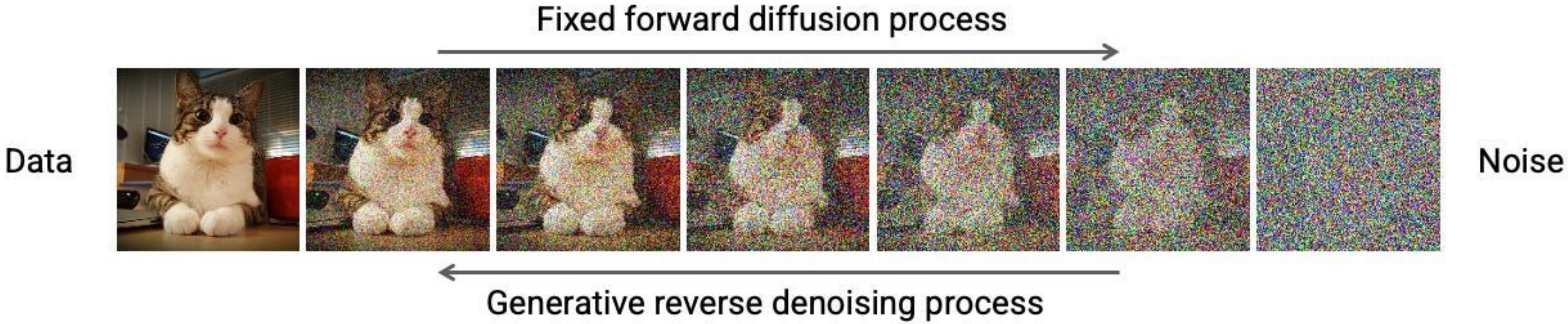
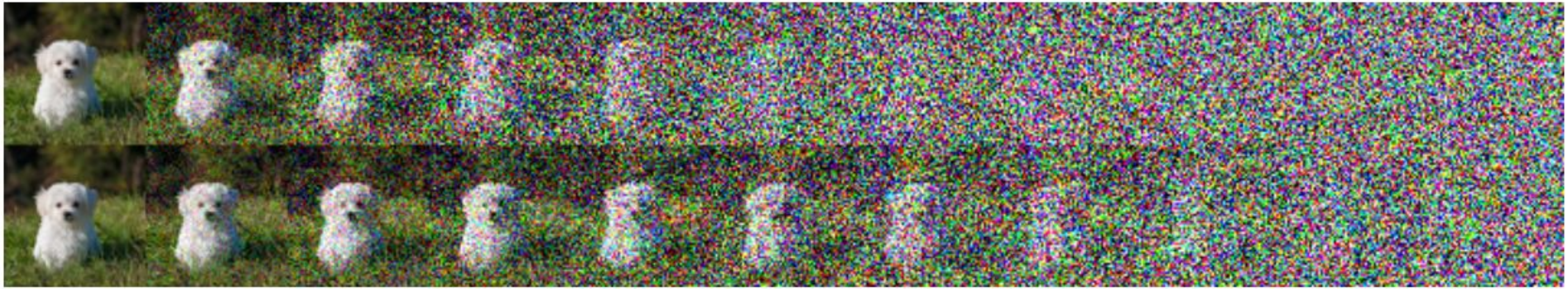
# Diffusion Models

The essential idea, inspired by non-equilibrium statistical physics, is to systematically and slowly destroy structure in a data distribution through an iterative **forward diffusion process**. We then learn a **reverse diffusion process** that restores structure in data, yielding a highly flexible and tractable generative model of the data.


Fixed forward diffusion process
Data
Noise
Generative reverse denoising process

# Linear vs Cosine Schedule



# What does architecture look like

Noise → Mode (U-NET) →

1. Mean of Noise

   OR

2. Original Image

   OR

3. Noise in the image

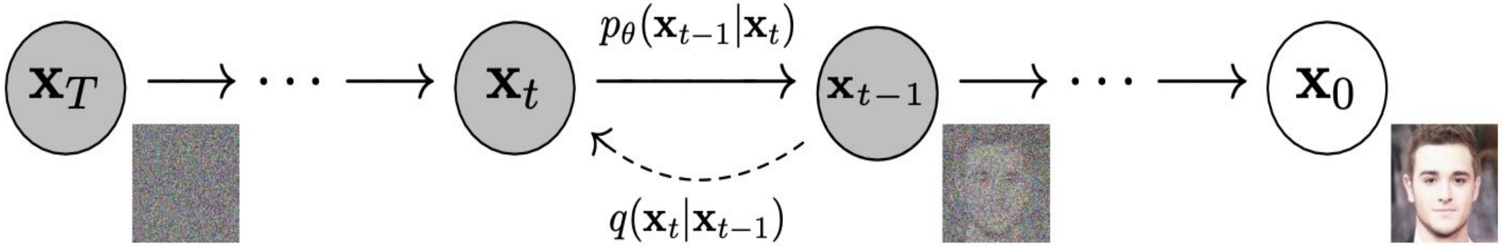



Figure 2: The directed graphical model considered in this work.

## Forward diffusion process

Given a data point sampled from a real data distribution $\mathbf{x}_0 \sim q(\mathbf{x})$, let us define a *forward diffusion process* in which we add small amount of Gaussian noise to the sample in $T$ steps, producing a sequence of noisy samples $\mathbf{x}_1, \dots, \mathbf{x}_T$. The step sizes are controlled by a variance schedule $\{\beta_t \in (0,1)\}_{t=1}^T$.

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \quad q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})$$

The data sample $\mathbf{x}_0$ gradually loses its distinguishable features as the step $t$ becomes larger. Eventually when $T \to \infty$, $\mathbf{x}_T$ is equivalent to an isotropic Gaussian distribution.

$$q(x_t \mid x_{t-1}) = \mathcal{N}(x_t, \sqrt{1-\beta_t}\, x_{t-1}, \beta_t I)$$

$$= \sqrt{1-\beta_t}\, x_{t-1} + \sqrt{\beta_t}\, \epsilon$$

---- reparameterization

$$= \sqrt{\alpha_t}\, x_{t-1} + \sqrt{1-\alpha_t}\, \epsilon$$

---- $\alpha_t = 1 - \beta_t$

$$= \sqrt{\alpha_t \alpha_{t-1}}\, x_{t-2} + \sqrt{1-\alpha_t \alpha_{t-1}}\, \epsilon$$

$$= \cdots$$

$$= \sqrt{\alpha_t \alpha_{t-1} \cdots \alpha_1 \alpha_0}\, x_0 + \sqrt{1-\alpha_t \alpha_{t-1} \cdots \alpha_1 \alpha_0}\, \epsilon$$

$$= \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\, \epsilon$$

--- $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$

reparameterization

$$\mathcal{N}(\mu, \sigma^2) = \mu + \sigma\epsilon$$

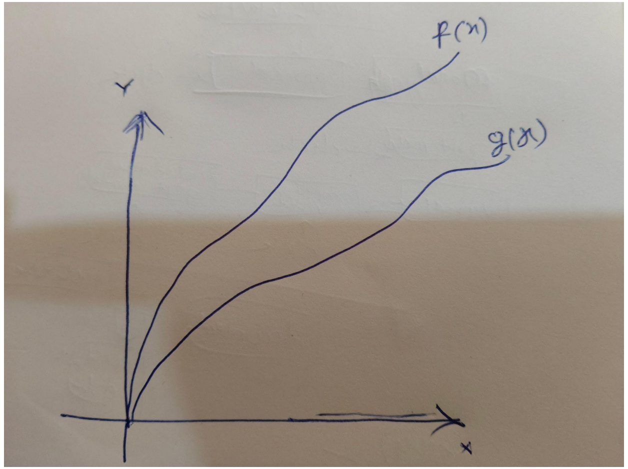
Reverse Diffusion Process:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t))$$

To calculate noise:

- $\log(P_{\theta}(x_0))$

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



**Algorithm 1** Training

**repeat**

$\mathbf{x}_0 \sim q(\mathbf{x}_0)$

$t \sim \mathrm{Uniform}(\{1, \ldots, T\})$

$\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$

Take gradient descent step on

$$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$$

**until** converged

**Algorithm 2** Sampling

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

# Results

| Model | FID | Year |
| --- | --- | --- |
| StyleGAN | 3.26 | 2022 |
| BigRoc-XL | 3.63 | 2021 |
| OpenAI Paper | 4.59/3.94 | 2021 |
| DDPM | 12.3 | 2020 |