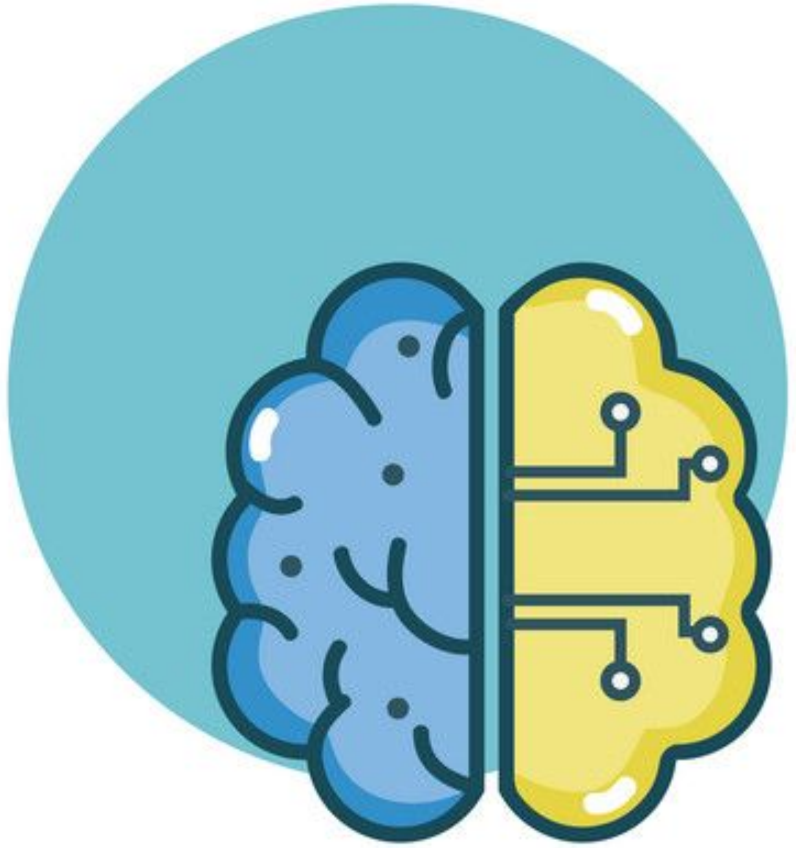
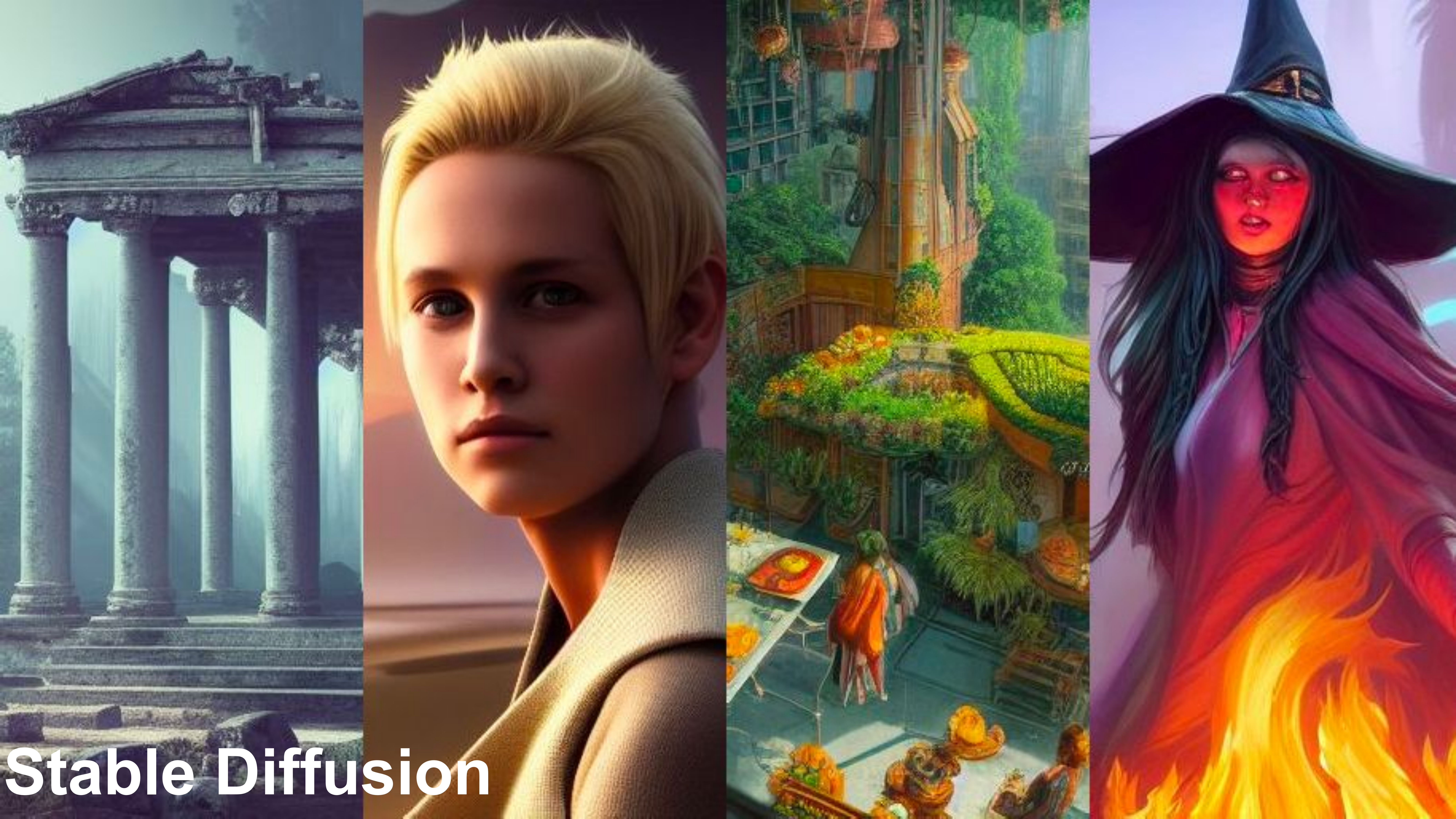
INTRODUCTION TO MACHINE LEARNING

Diffusion Models

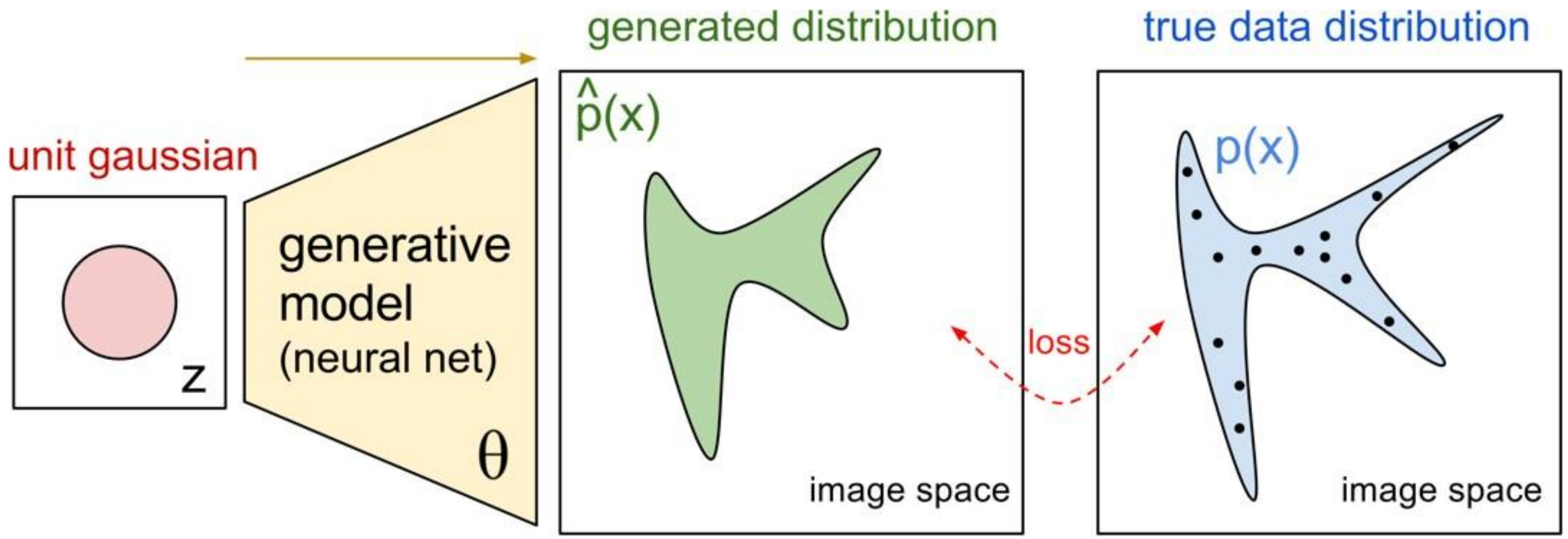
Elisa Ricci




Stable Diffusion

Generative Models

ricordarsi che questi modelli vanno bene per ogni tipo di structured data
non solo immagini

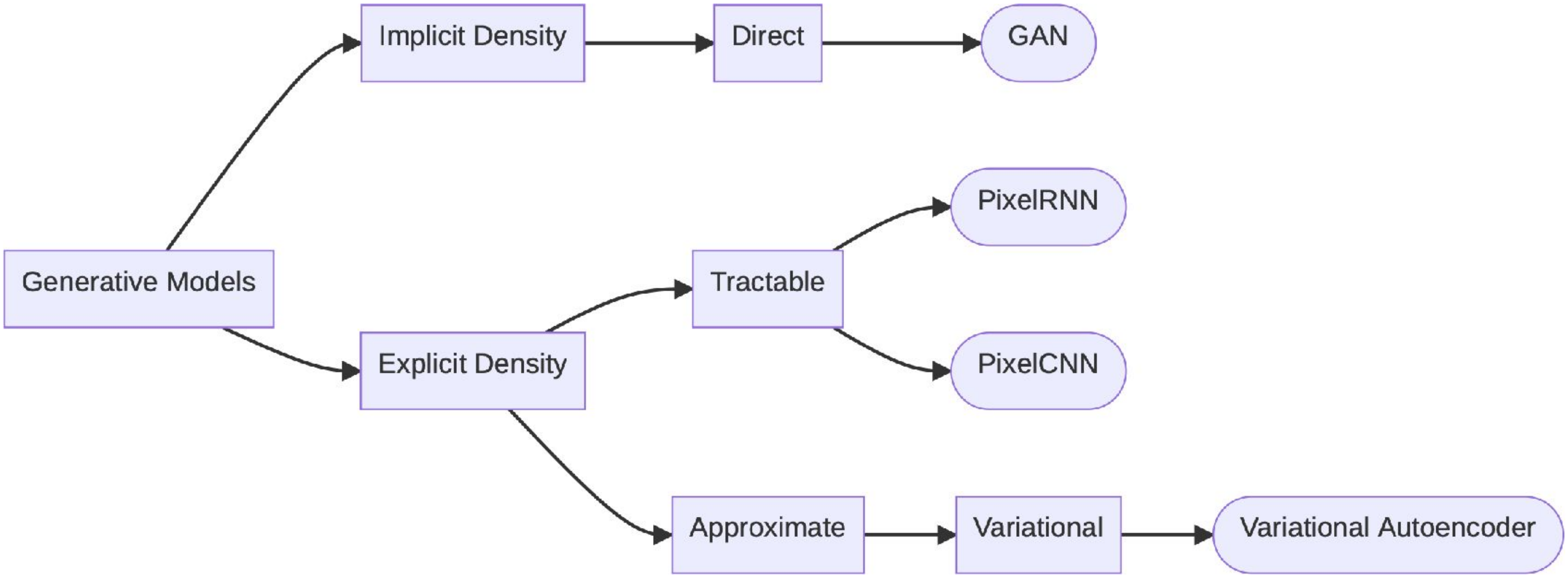


Train a generator G from latent space to data space and approximate the real data distribution

Training is difficult:

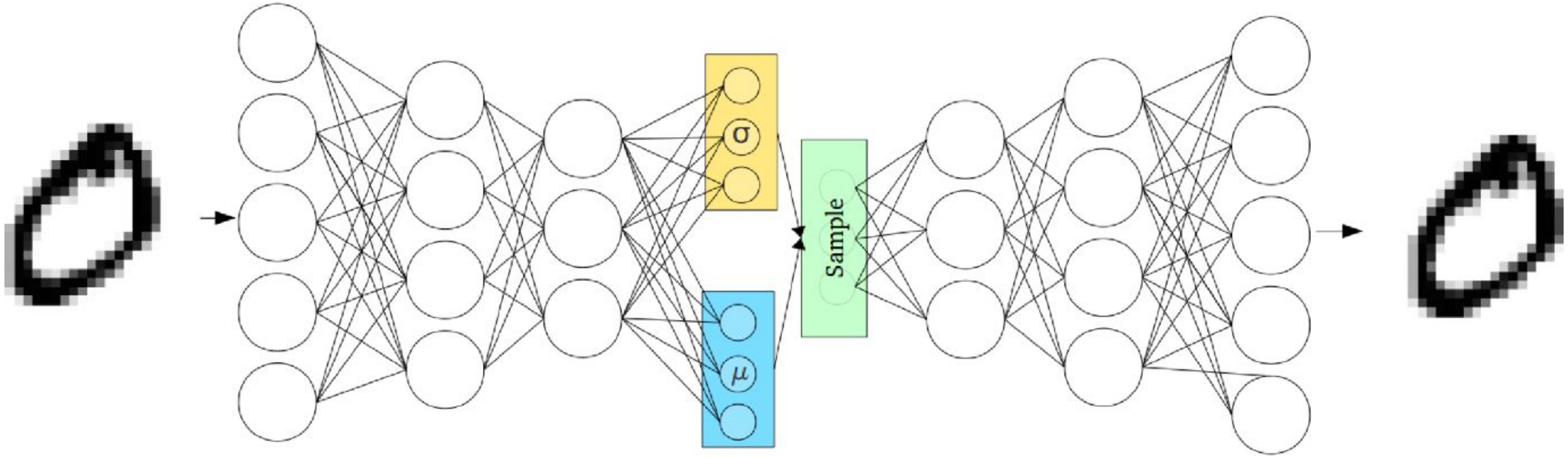
- Hyperparameters choice
- Quantify similarity between sets
- Choice of latent space

Generative Models



Goodfellow, I. (2016). Nips 2016 tutorial: Generative adversarial networks. arXiv preprint arXiv:1701.00160.

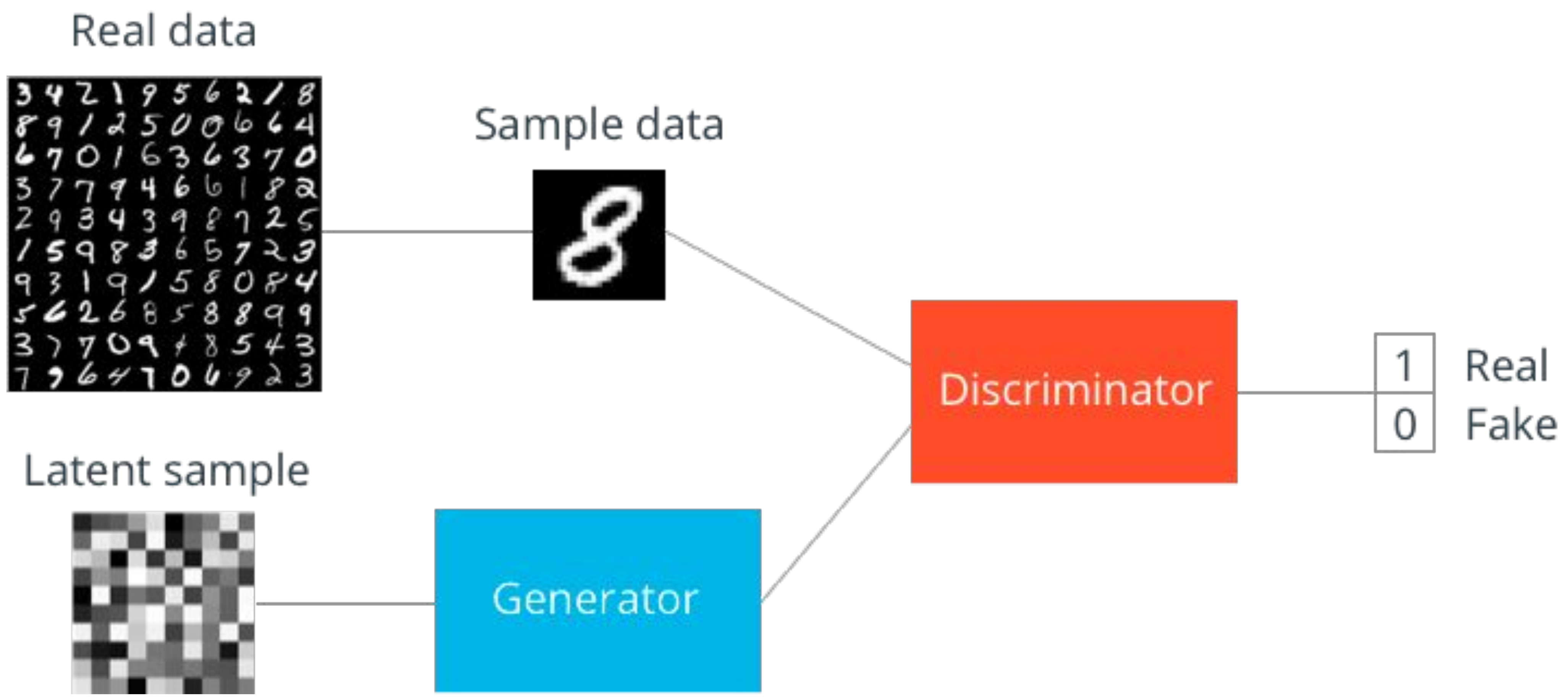
Variational Autoencoders Recap



- Add to autoencoders the ability to generate new samples
  - Probabilistic latent space (assumed to be multivariate Gaussian)
- Minimize at the same time reconstruction error and KL divergence between latent and $N(0,\mathbf{I})$
- Sampling from the latent distribution allows generation and interpolation

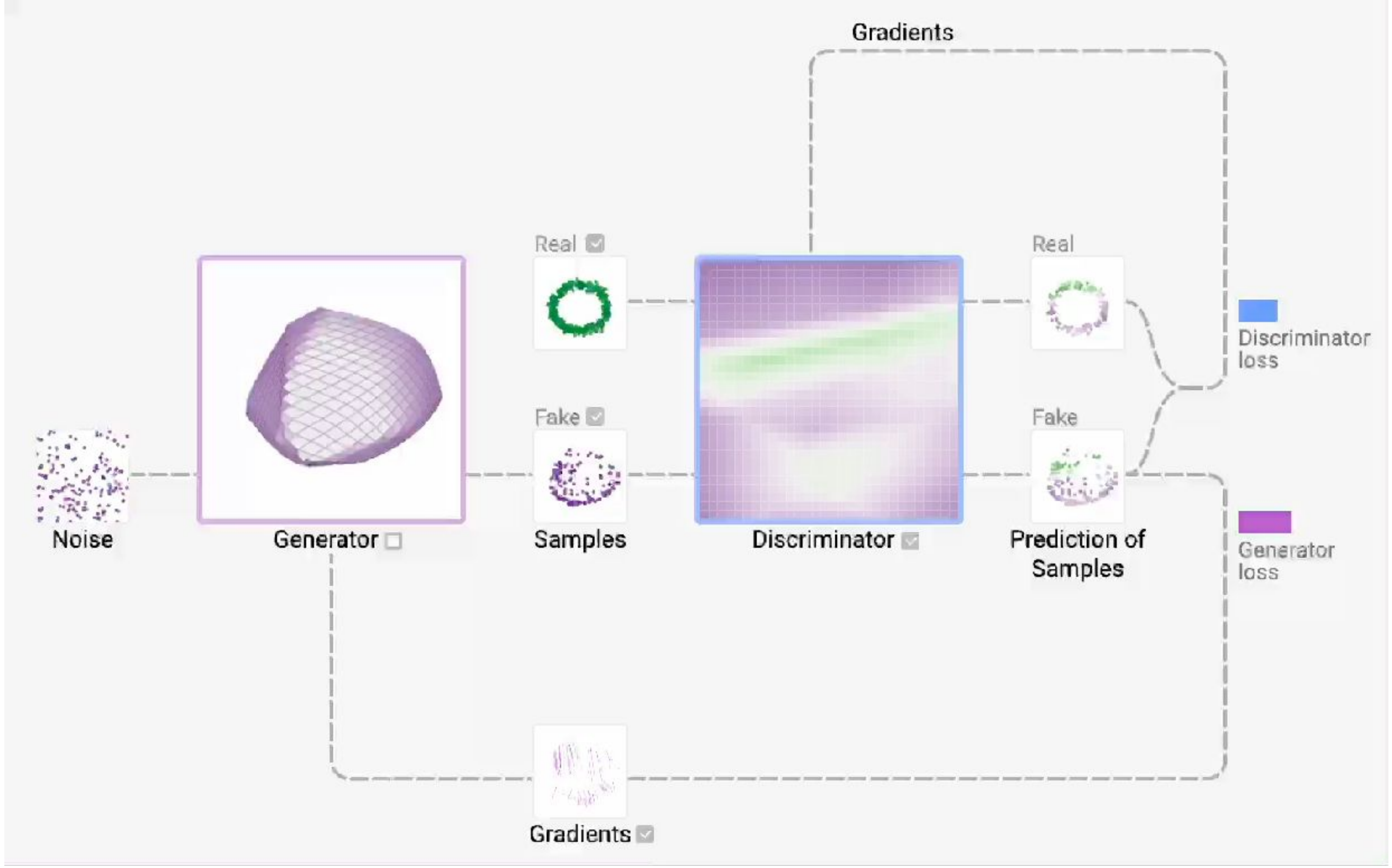
Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114.

Generative Adversarial Networks Recap



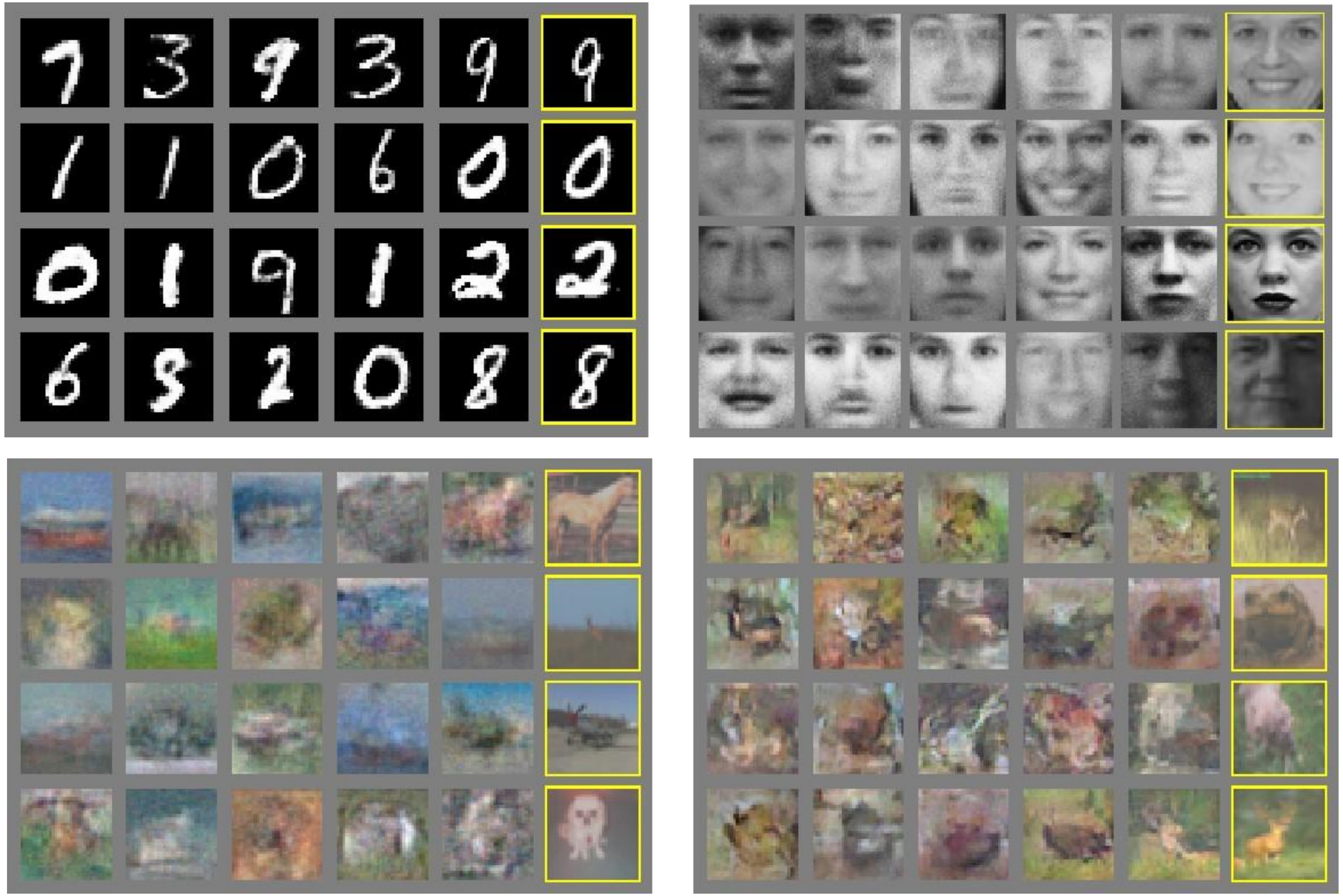
- Generator starts from Gaussian noise and generates a data point in order to fool the discriminator
- Discriminator tries to distinguish between real and fake data
- Training becomes a games between generator and discriminator
  - Solution lies in the Nash equilibrium between the two participants

GAN Visualization



https://poloclub.github.io/ganlab/

Results - 2016



MNIST, Toronto Face Dataset, FC CIFAR-10, Con CIFAR-10 results

Results - 2021

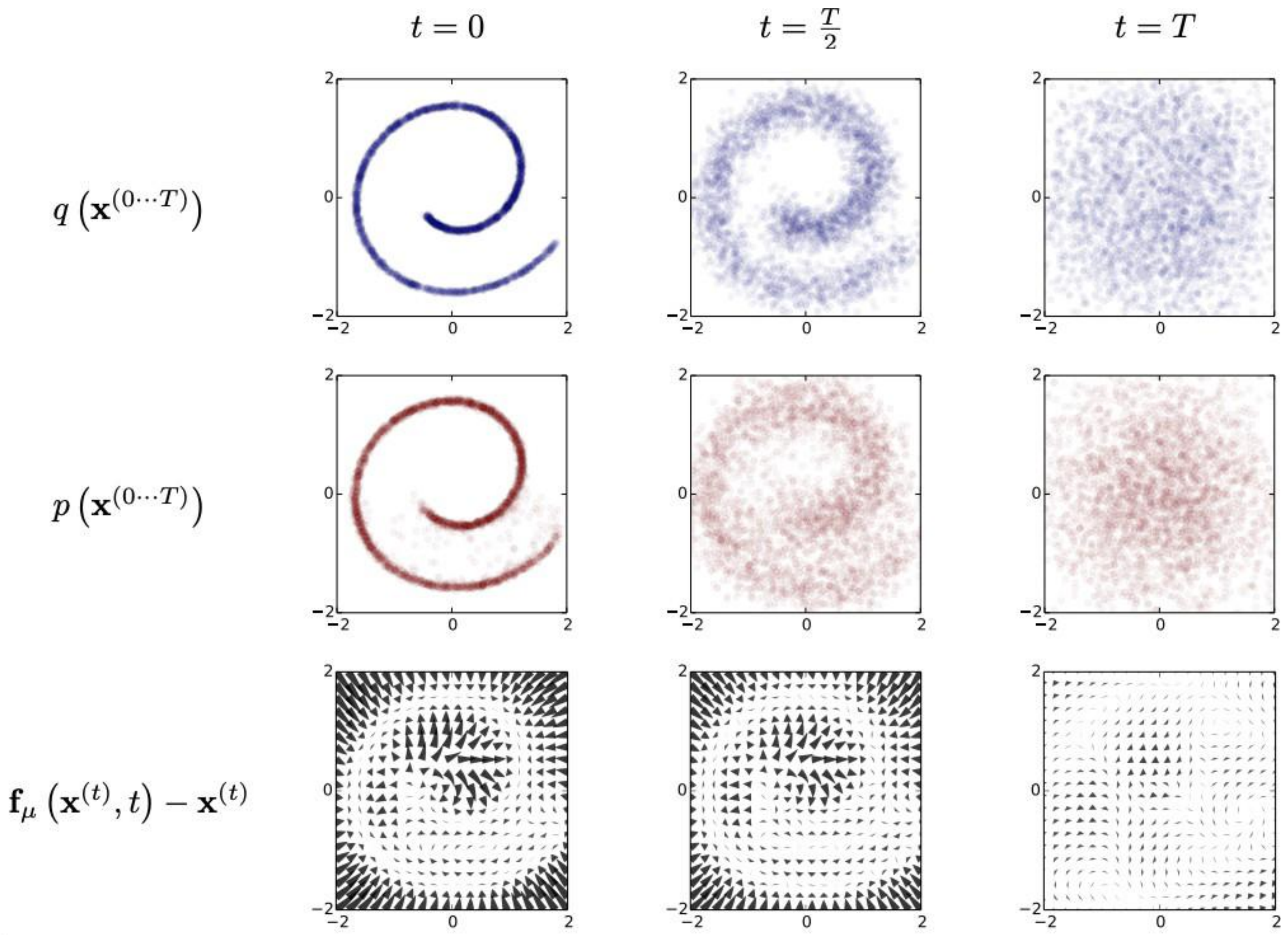
- Huge improvements thanks to several tricks, especially in the StyleGAN family
- Smooth walking in the latent space bears meaningful change in the output

Karras, Tero, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. "Alias-free generative adversarial networks." *Advances in Neural Information Processing Systems* 34 (2021).

StyleGAN2 | StyleGAN3 (Ours)

Random latent walk using directions from StyleCLIP, GANSpace, and SeFa.

Deep Unsupervised Learning using Nonequilibrium Thermodynamics



Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., & Ganguli, S. (2015, June). Deep unsupervised learning using nonequilibrium thermodynamics. In International Conference on Machine Learning (pp. 2256-2265). PMLR.

Denoising Diffusion Probabilistic Model

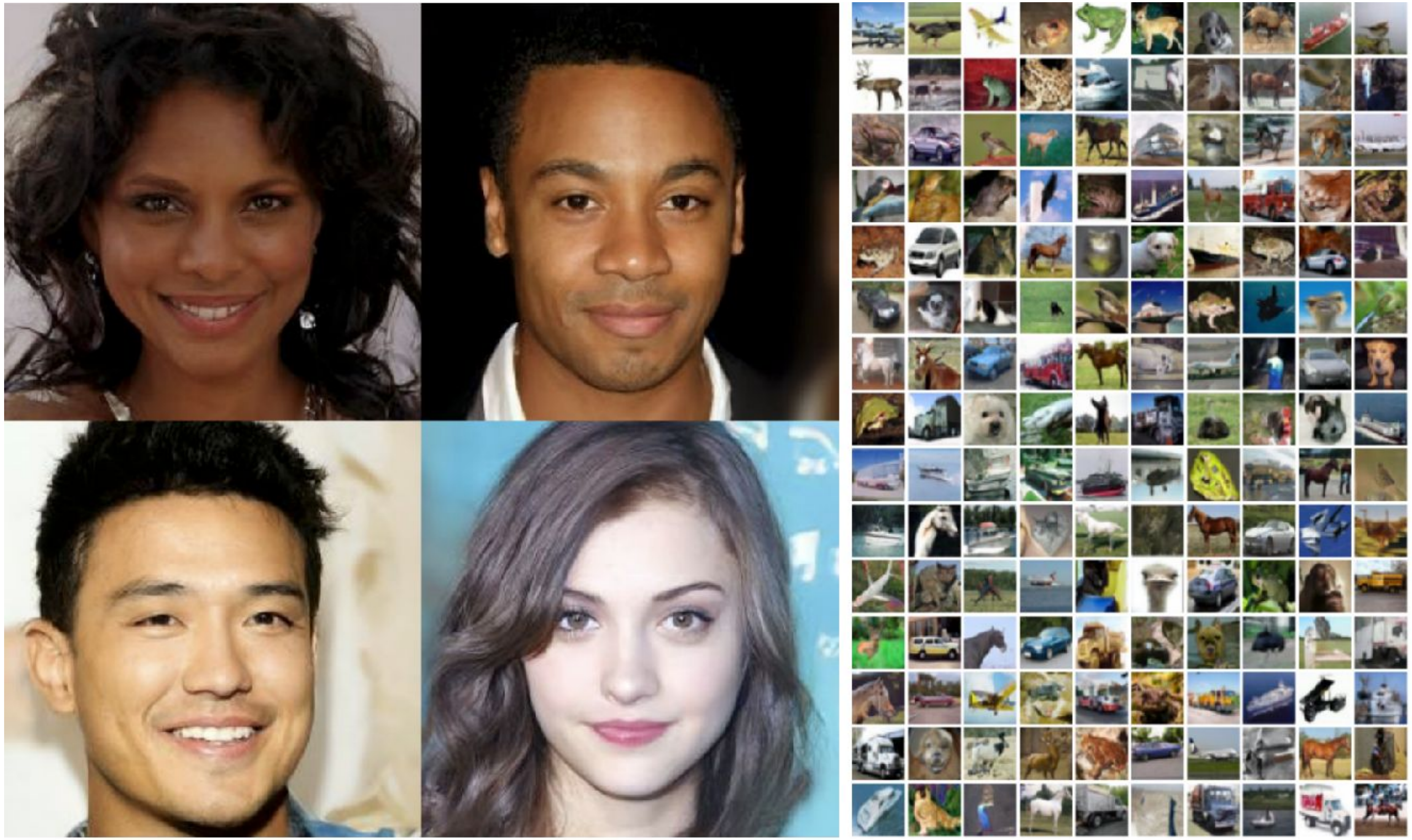


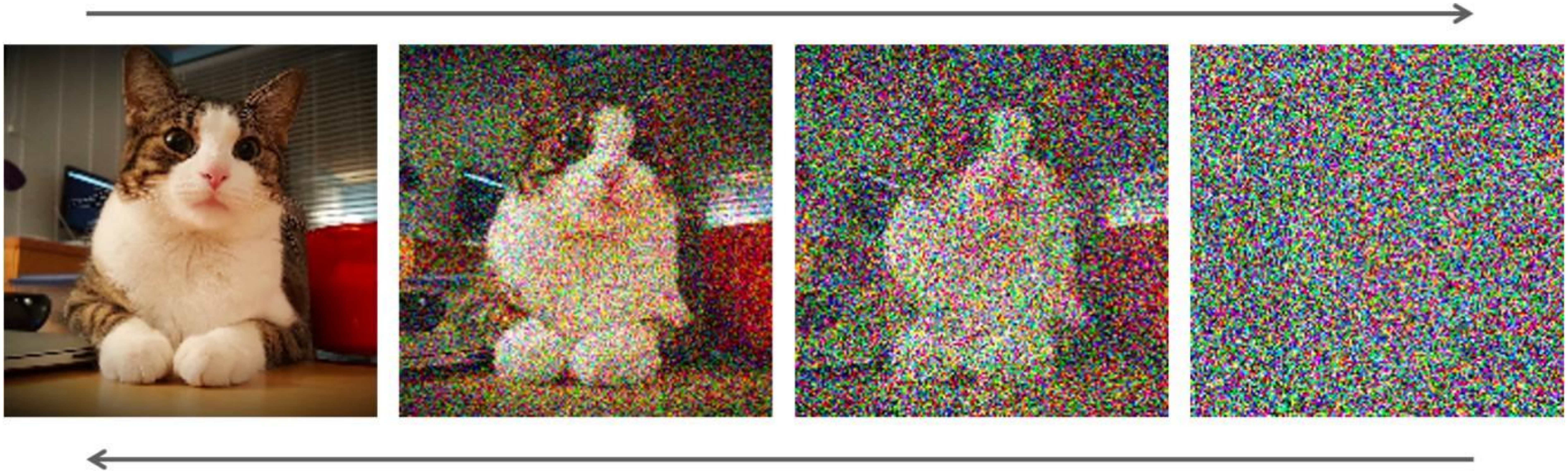
Figure 1: Generated samples on CelebA-HQ 256 × 256 (left) and unconditional CIFAR10 (right)

Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems, 33, 6840-6851.

Theory

È importante che impariamo il flavour e l'algoritmo

Basic formulation

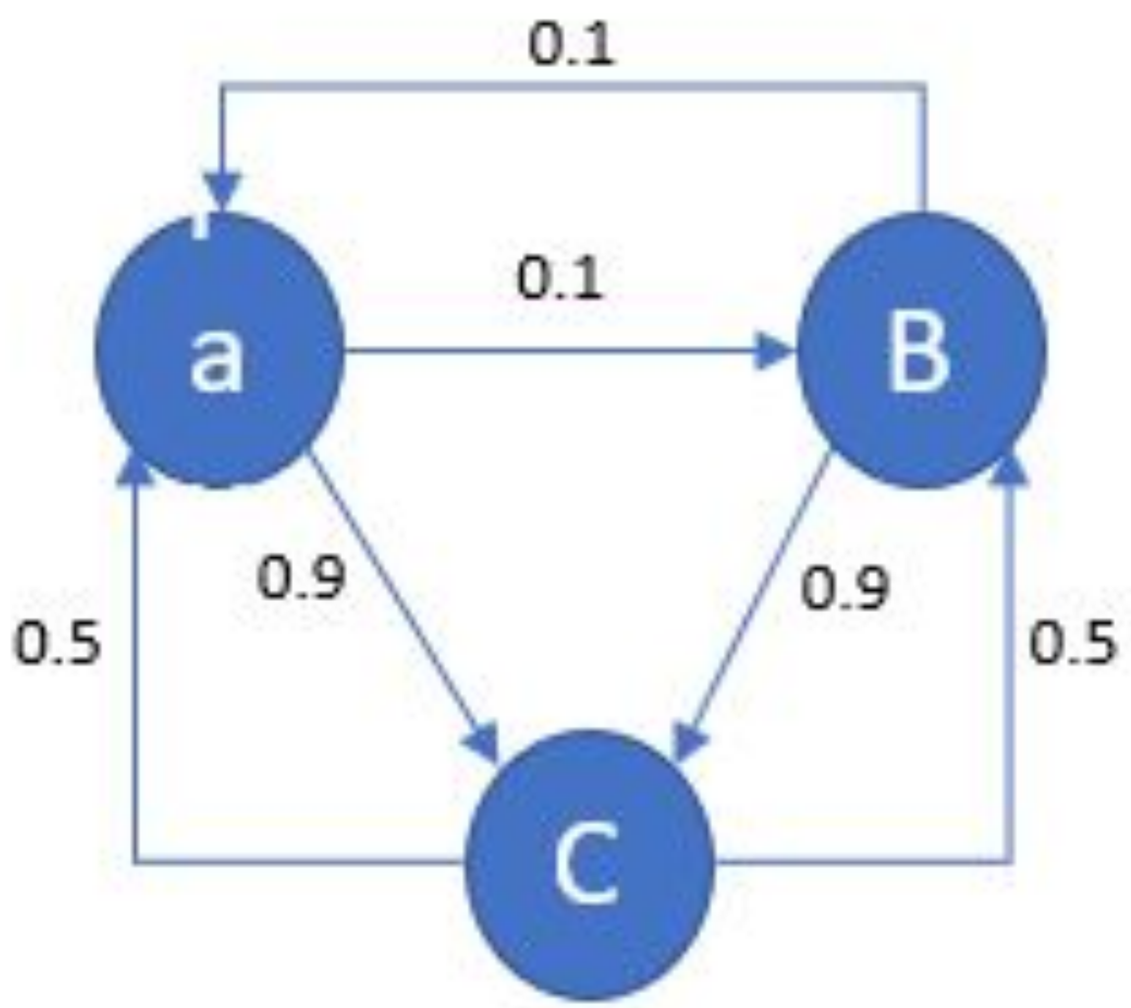


Denoising diffusion models consists of two processes:

- Forward process to add noise
- Reverse process denoises to generate data

Recap: Markov Chain

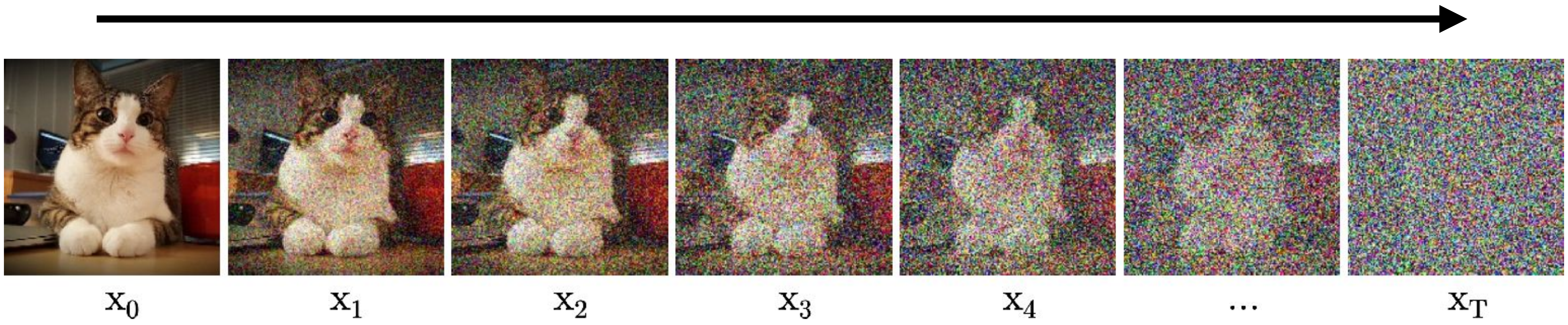
A Markov chain or Markov process is a stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event



$$P = \begin{bmatrix} 0 & 0.1 & 0.9 \\ 0.1 & 0.0 & 0.9 \\ 0.5 & 0.5 & 0.0 \end{bmatrix}$$

gli archi rappresentano la probabilità di muoversi da un evento all'altro

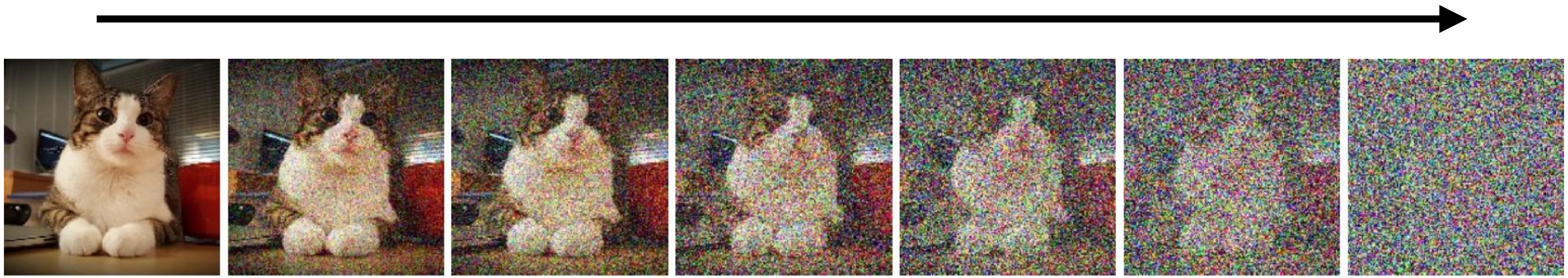
Forward process



Formally we have the following formulation:

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

Forward process

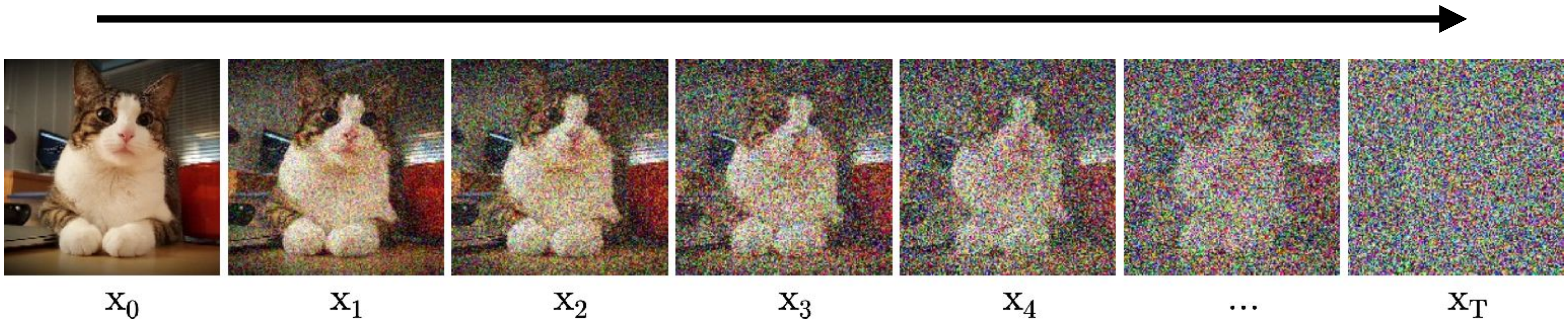


$$q(x_t|x_{t-1}) = \mathcal{N}(\sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I}) \implies q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1})$$

With a change of variables:

Define: $\alpha_t = (1-\beta_t) \implies \bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s) \implies q(x_t|x_0) = \mathcal{N}(\sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)\mathbf{I})$

Forward process



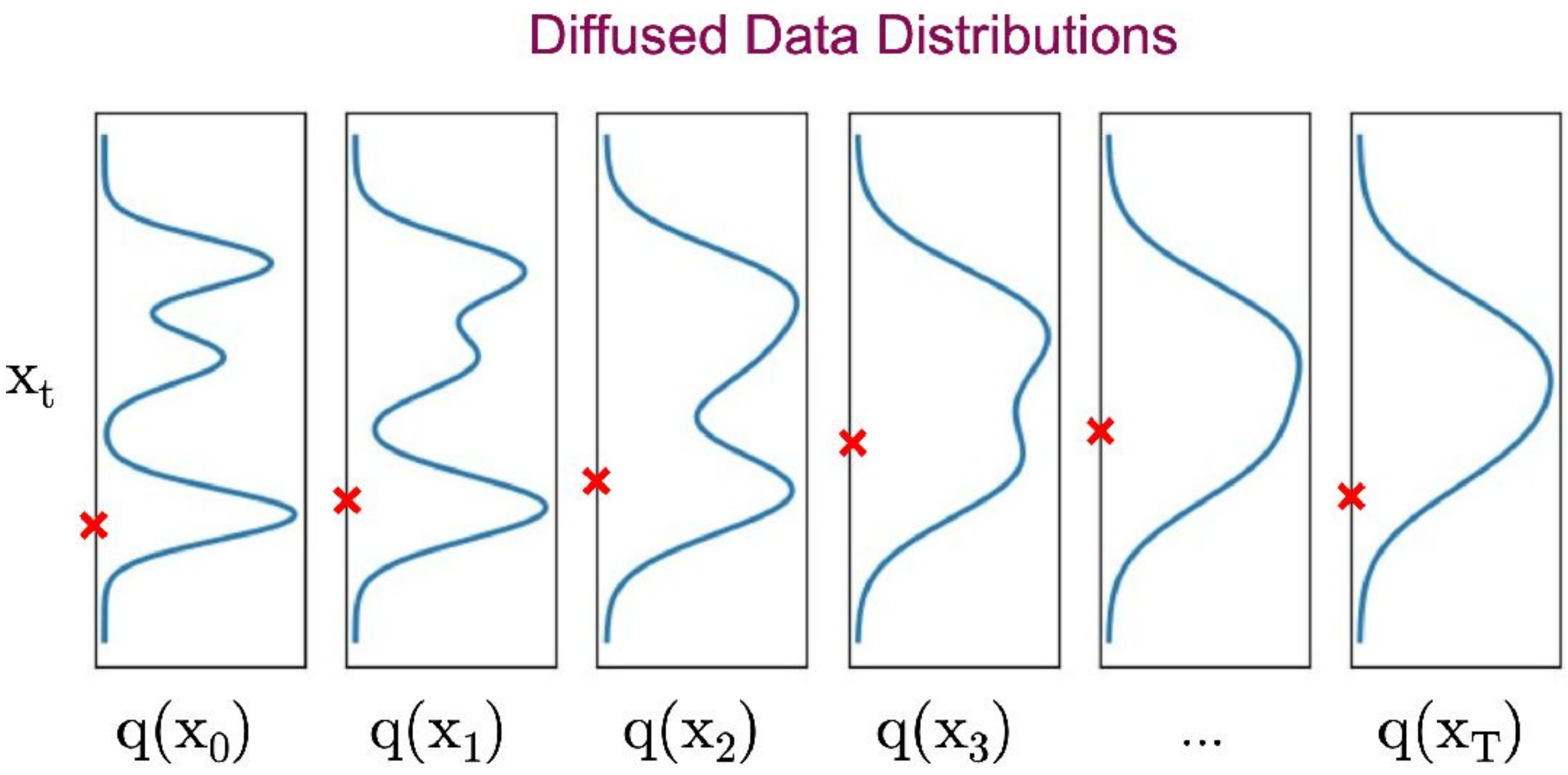
t

We can then sample directly at the desired timestep $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{(1 - \bar{\alpha}_t)}\epsilon, \epsilon \sim \mathcal{N}(0,\mathbf{I})$

β_t is the noise schedule

Forward process

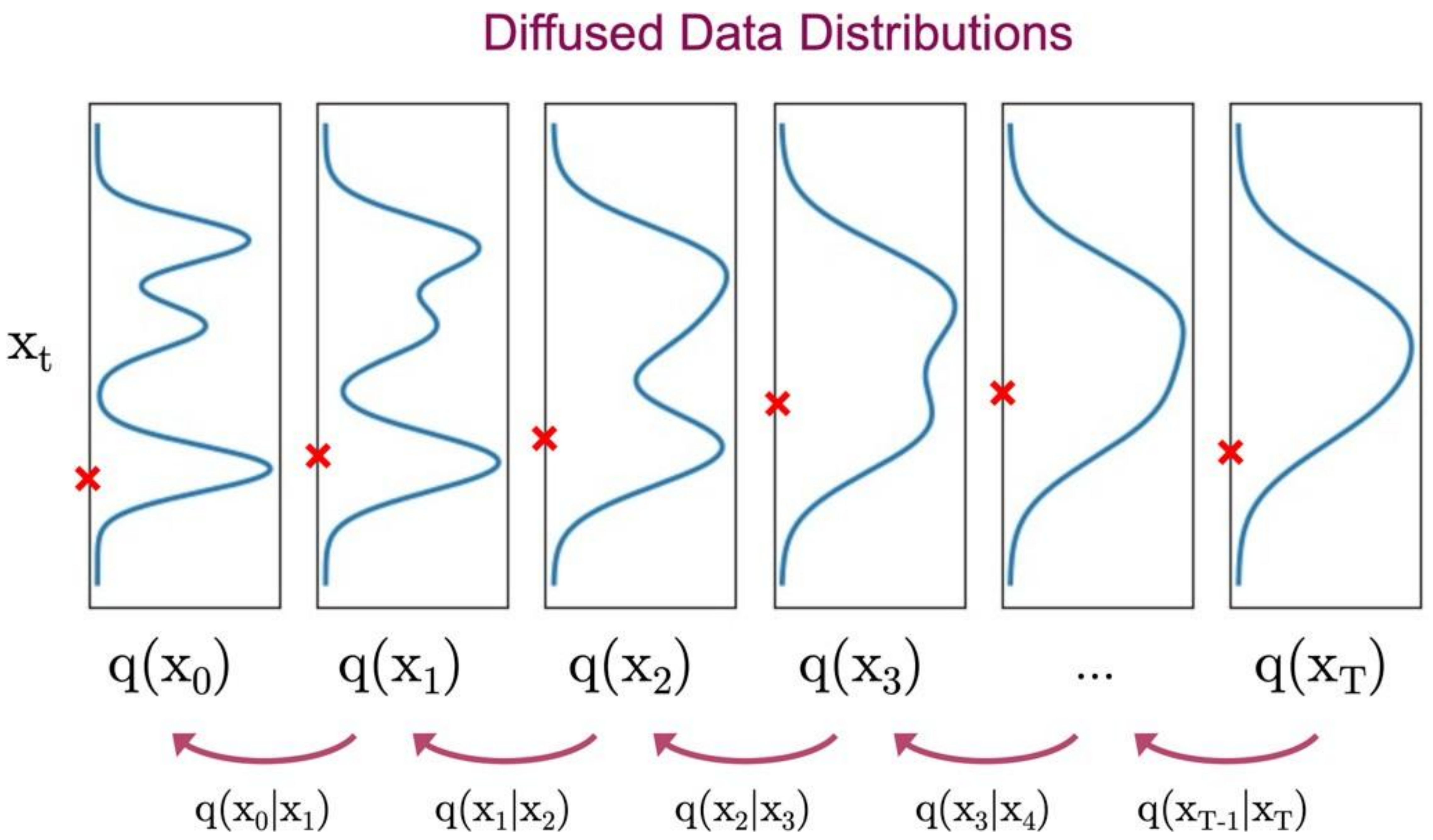
- Formally we are applying a **Gaussian convolution** to the data at each timestep
- Practically we are **smoothening** out the distribution to a Gaussian one



Generation process

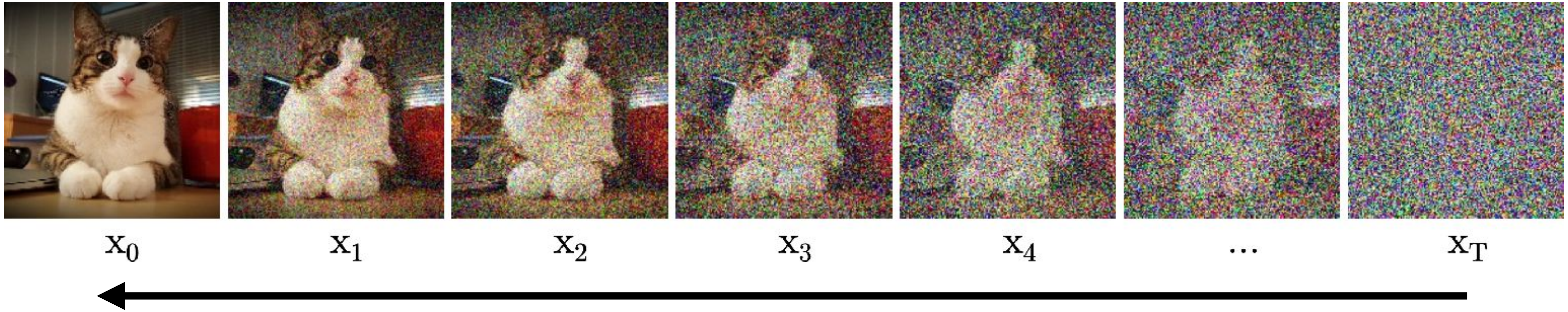
Given that $q(x_T) \approx \mathcal{N}(0,\mathbf{I})$:

- Sample $x_T \sim \mathcal{N}(0,\mathbf{I})$
- Iteratively sample $x_{t-1} = q(x_{t-1} \mid x_t)$



$q(x_{t-1} \mid x_t) \propto q(x_{t-1})q(x_t \mid x_{t-1})$ is generally **intractable,** but we can **approximate** with another Gaussian

Reverse process



$$p(x_T) \sim \mathcal{N}(0, \mathbf{I})$$

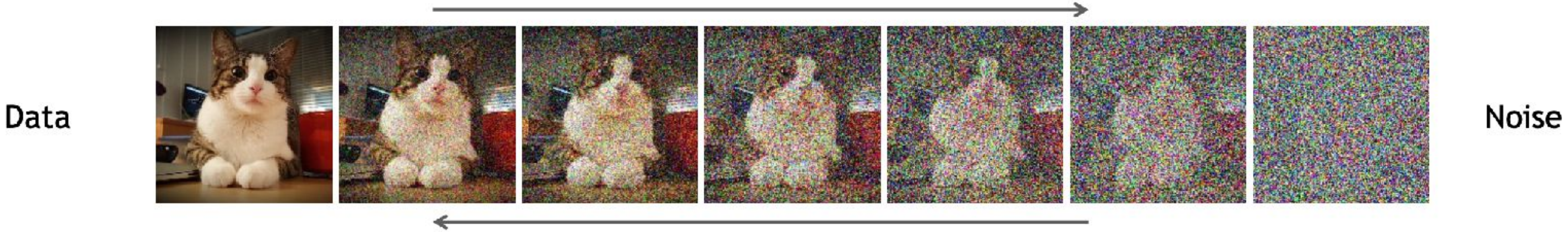
$$p_\theta(x_{t-1} \mid x_t) \sim \mathcal{N}(\boxed{\mu_\theta(x_t, t)}, \sigma_t^2 \mathbf{I})$$

Trainable network

$$\longrightarrow \quad p_\theta(x_{0:T}) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1} \mid x_t)$$

Noising Schedule

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x_t}; \sqrt{1-\beta_t}\mathbf{x_{t-1}}, \beta_t\mathbf{I})$$



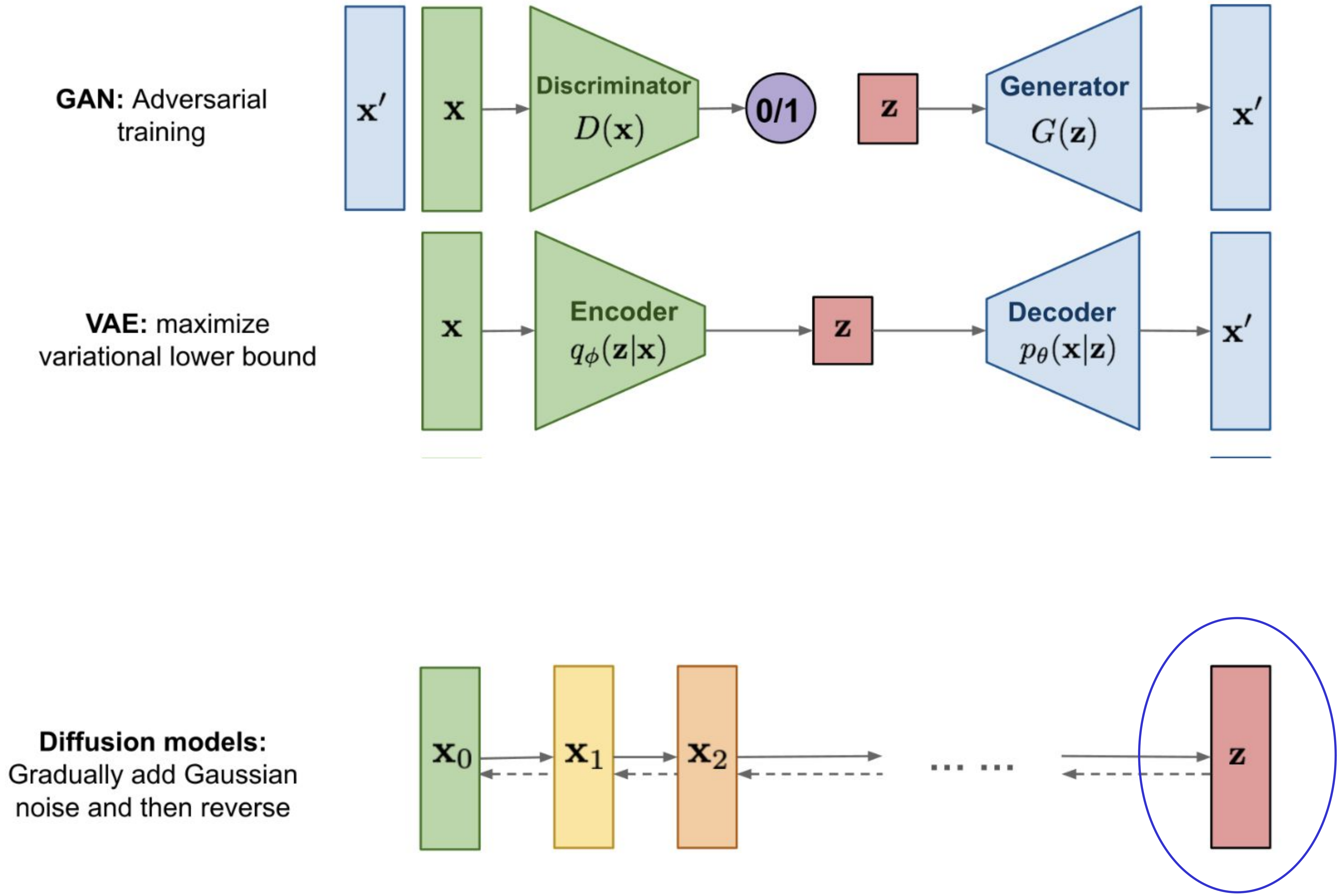
$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \sigma_t^2\mathbf{I})$$

We can control the variance of the forward diffusion and reverse denoising processes respectively. Often a linear schedule is used for β_t, and σ_t^2 is set equal to β_t .

Kingma et al. [NeurIPS 2022] introduce a new parameterization of diffusion models using signal-to-noise ratio (SNR), and show how to learn the noise schedule by minimizing the variance of the training objective.
Other improvements: (Improved DPM by Nichol and Dhariwal [ICML 2021], Analytic-DPM by Bao et al. [ICLR 2022]).

Connection to VAE, GANs

- Latent variables have the **same dimensionality** of data
- The **same model** is applied across different timesteps
- The model is trained by **reweighing** the variational bound



Training parametrisation

We can train the model in a similar fashion as VAE, with a Variational Upper Bound

$$L = \mathbb{E}_{q(x_0)}\left[-\log p_\theta(x_0)\right] \leq \mathbb{E}_{q(x_0)q(x_{1:T}|x_0)}\left[-\log \frac{p_\theta(x_{0:T})}{q(x_{1:T}|x_0)}\right]$$

These can be divided into three terms

$$L = \mathbb{E}_q\left[\boxed{D_{KL}(q(x_T|x_0)\,||\,p(x_T))} + \sum_{t>1} D_{KL}(q(x_{t-1}|x_t, x_0)\,||\,p_\theta(x_{t-1}|x_t)) - \log p_\theta(x_0|x_1)\right]$$

This is the original loss. Can this be simplified?

Training parametrisation

$$L = \mathbb{E}_q \left[D_{KL}(q(x_T | x_0) || p(x_T)) + \sum_{t>1} D_{KL}(q(x_{t-1} | x_t, x_0) || p_\theta(x_{t-1} | x_t)) - \log p_\theta(x_0 | x_1) \right]$$

KL between Gaussians has a nice closed form, but Ho (with some math) proves the training can be simplified to a noise prediction problem

Skipping some math steps, we obtain a new loss

$$L_{simple} = \mathbb{E}_{x_0 \sim q(x_0), \epsilon \sim \mathcal{N}(0,\mathbf{I}), t \sim \mathcal{U}(1,T)} \left[|| \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t} x_0 + \sqrt{(1 - \bar{\alpha}_t)} \epsilon, t) ||^2 \right]$$

Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems, 33, 6840-6851.

Training and Sampling

In questo caso il training è più semplice dell'inference phase

Algorithm 1 Training

1: **repeat**

2: $\quad \mathbf{x}_0 \sim q(\mathbf{x}_0)$

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

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

5: $\quad$ Take gradient descent step on

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

6: **until** converged

Algorithm 2 Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$

2: **for** $t = T, \ldots, 1$ **do**

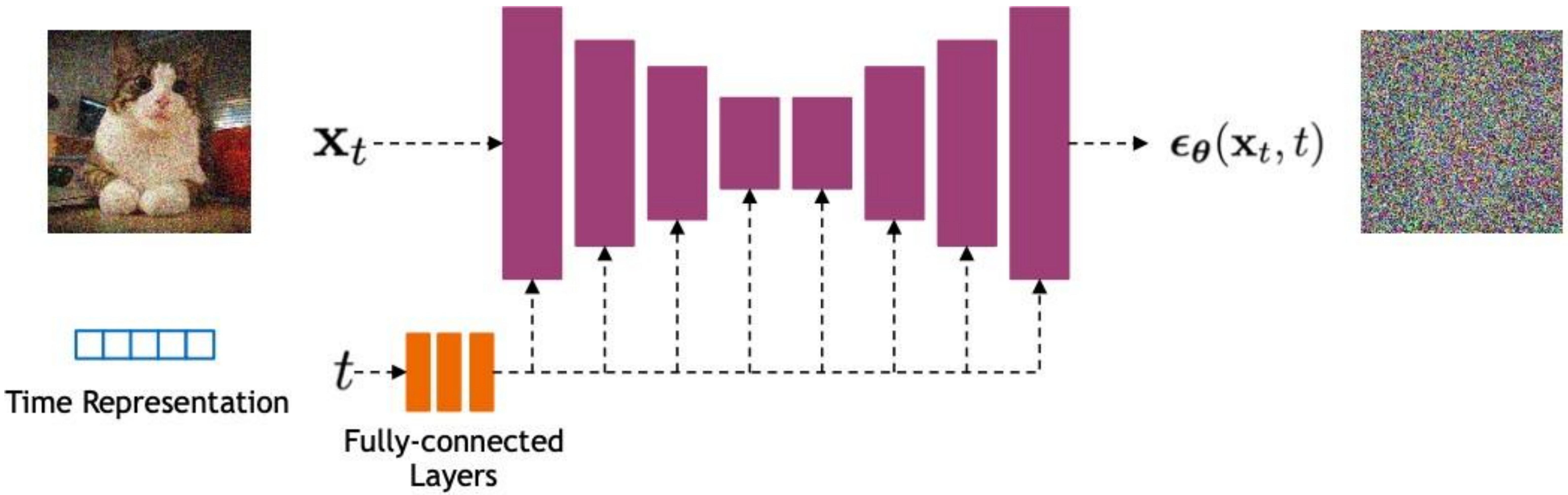
3: $\quad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$

4: $\quad \mathbf{x}_{t-1} = \boxed{\frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right)} + \sigma_t \mathbf{z}$

5: **end for**

6: **return** $\mathbf{x}_0$

Network

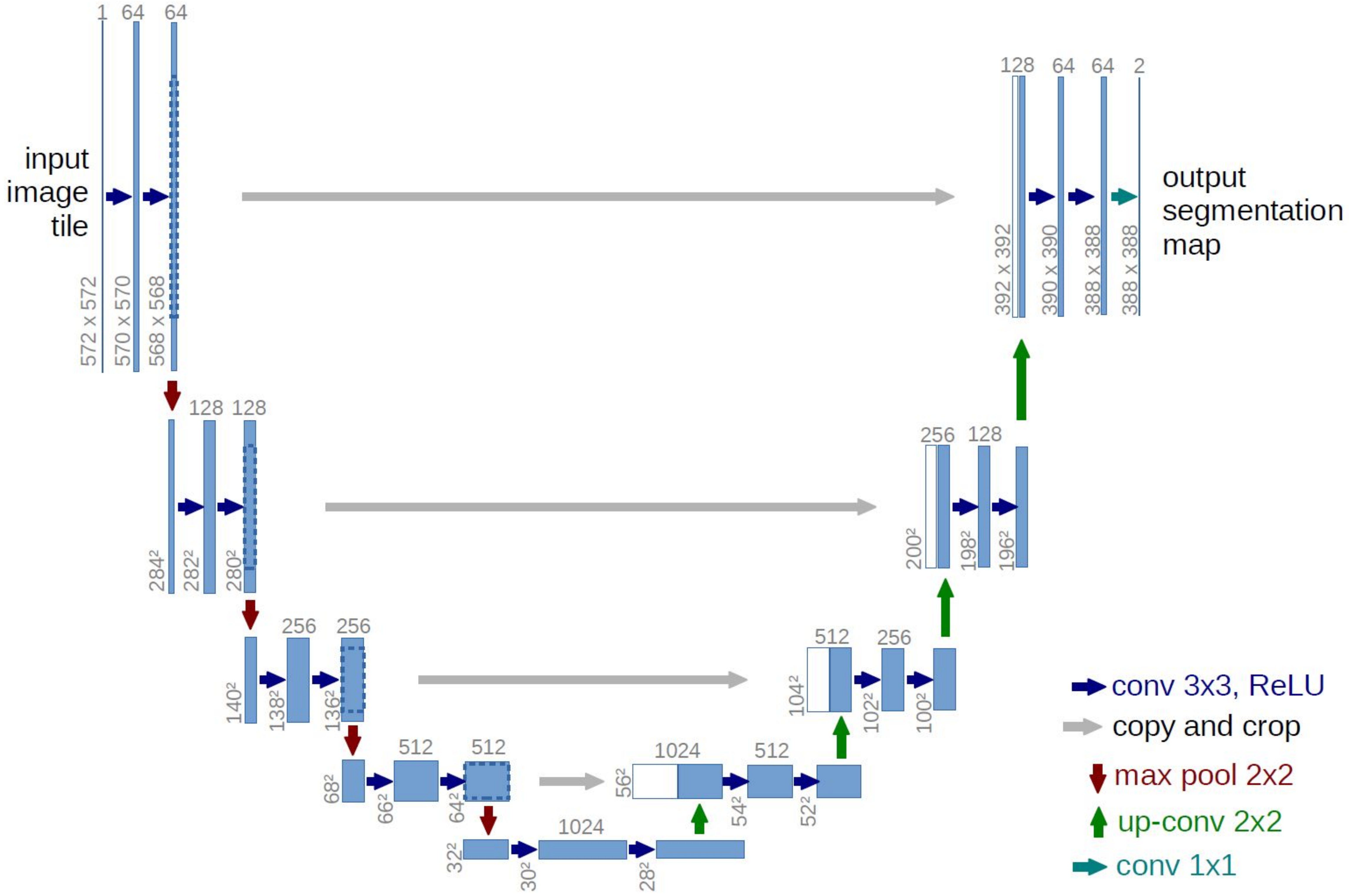


Pretty much free choice on the network architecture, no theoretical constraints on this
For images use **U-Net** (with attention)

Time features are usually **sinusoidal** of random **Fourier** features. How to embed them in the network is another free choice (e.g. spatial concatenation, AdaIN, etc)

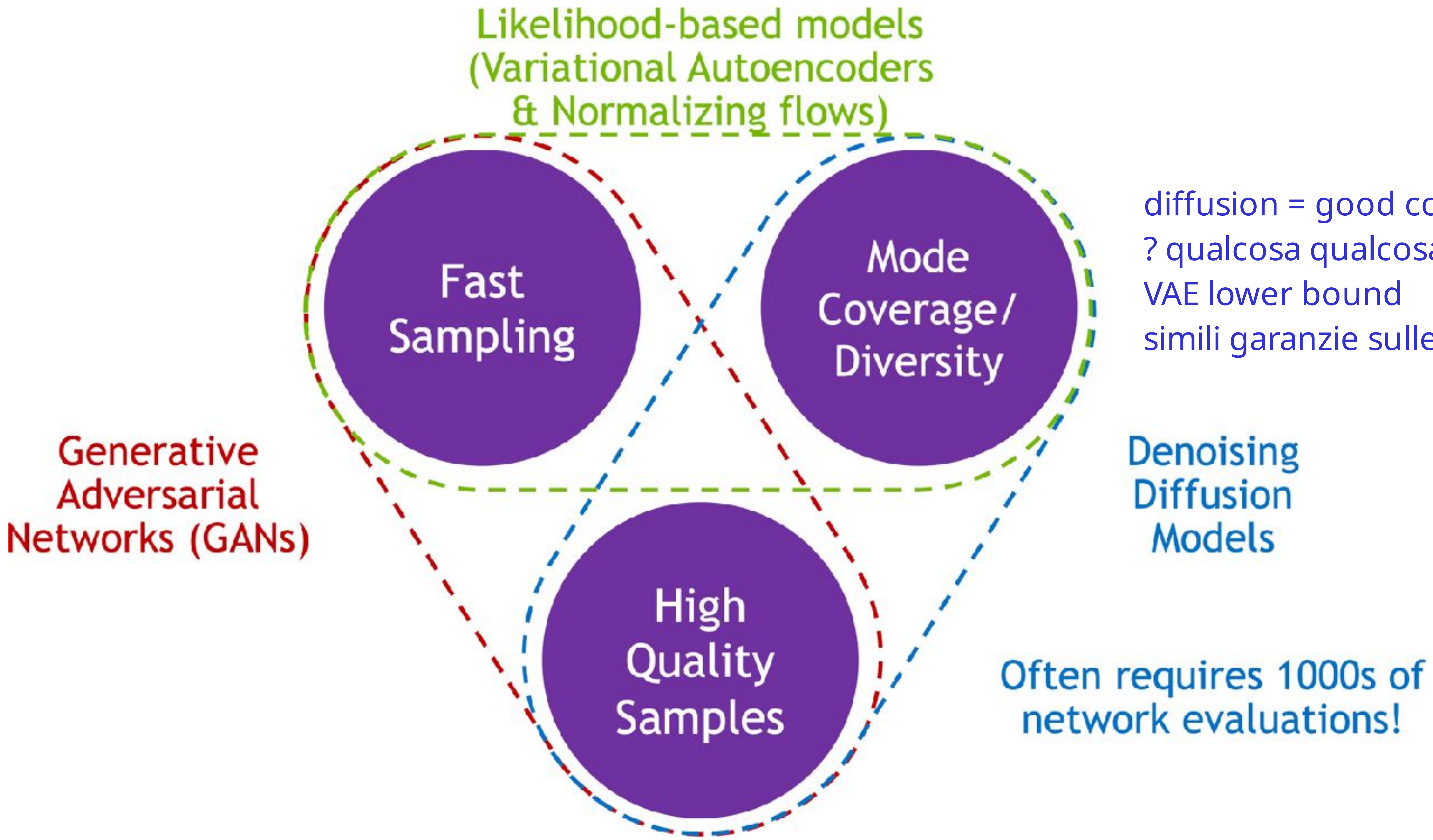
U-Net

- The U-NET architecture contains two paths.
- First path is the contraction path (also called as the encoder) which is used to capture the context in the image. The encoder is just a traditional stack of convolutional and max pooling layers.
- The second path is the symmetric expanding path (also called as the decoder) which is used to enable precise localization using transposed convolutions.
- It is an end-to-end fully convolutional network (FCN), i.e. it only contains Convolutional layers and does not contain any Dense layer because of which it can accept image of any size.



Olaf Ronneberger, Philipp Fischer, Thomas Brox, U-Net: Convolutional Networks for Biomedical Image Segmentation, Medical Image Computing and Computer-Assisted Intervention (MICCAI), Springer, LNCS, Vol.9351: 234--241, 2015,

Generative Trilemma



diffusion = good coverage
? qualcosa qualcosa
VAE lower bound
simili garanzie sulle space

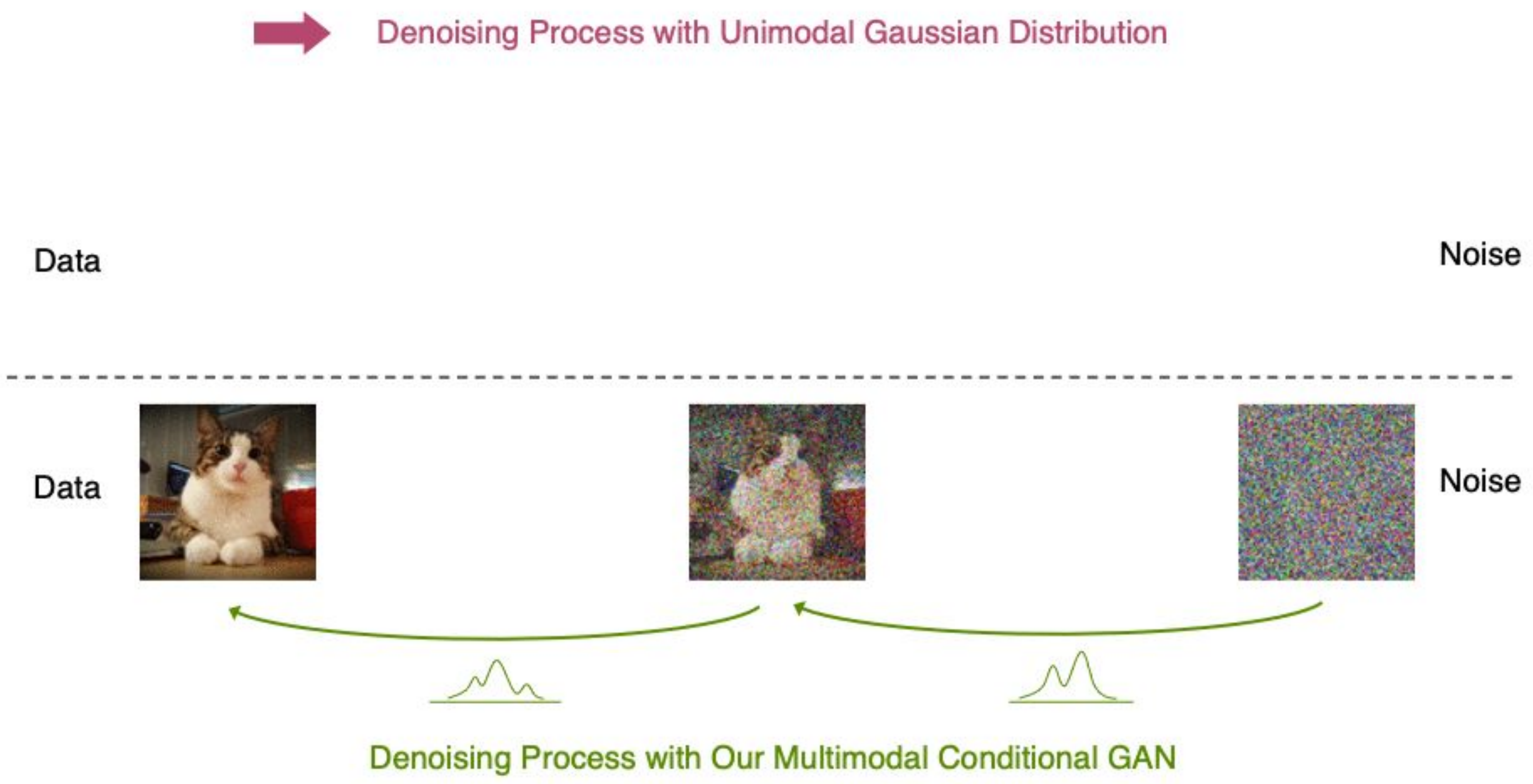
Xiao, Z., Kreis, K., & Vahdat, A. (2021). Tackling the generative learning trilemma with denoising diffusion gans. ICLR 2022.

Advanced Tricks

Il resto è extra

Diffusion GANs

Generative denoising diffusion models typically assume that the denoising distribution can be modeled by a Gaussian distribution. This assumption holds only for small denoising steps, which in practice translates to thousands of denoising steps in the synthesis process. In diffusion GANs, the denoising model is represented using multimodal and complex conditional GANs, enabling to efficiently generate data in a few steps.

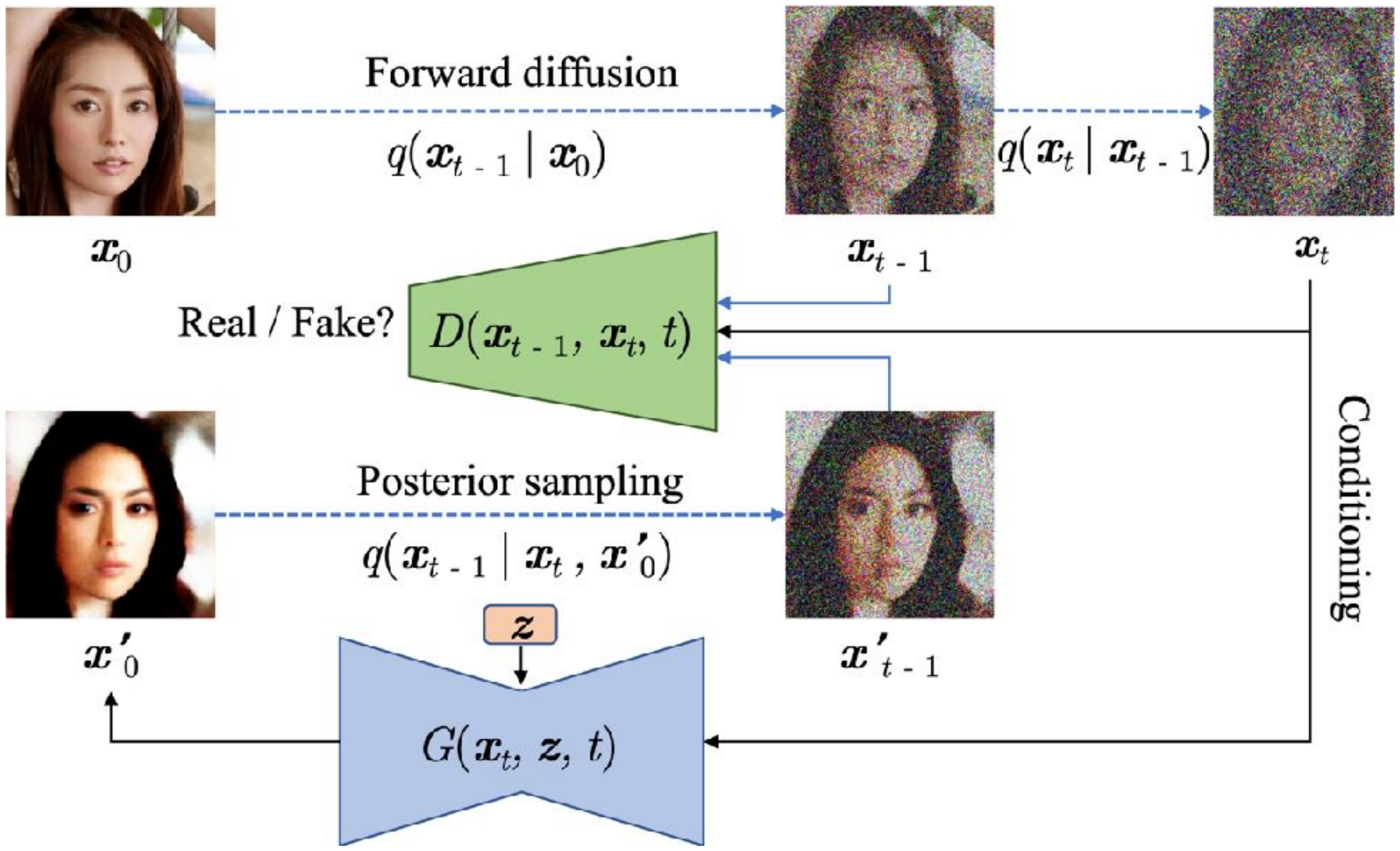


Xiao, Z., Kreis, K., & Vahdat, A. (2021). Tackling the generative learning trilemma with denoising diffusion gans. ICLR 2022.

Diffusion GANs

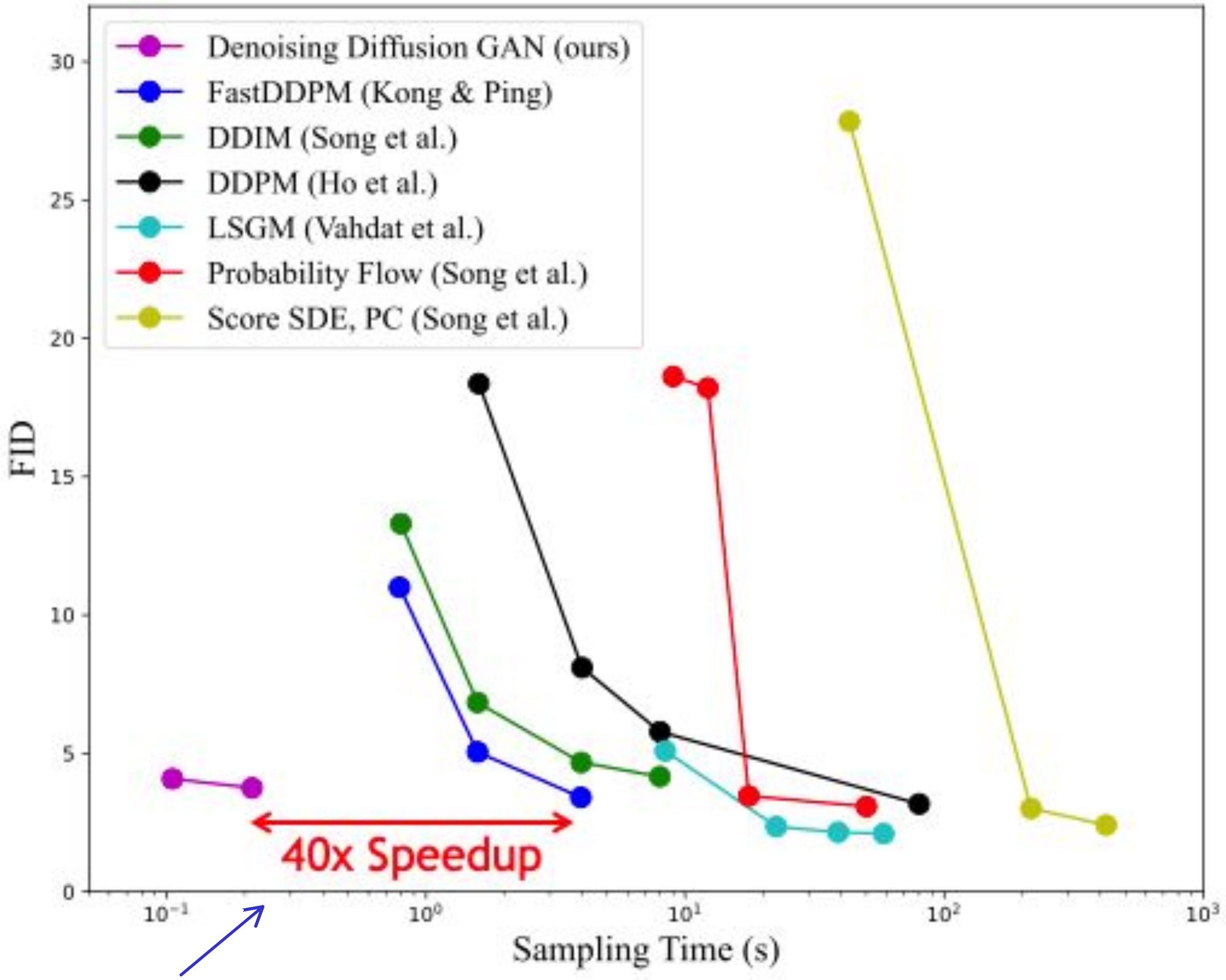
Compared to a one-shot GAN generator:

- Both generator and discriminator are solving a much **simpler** problem.
- Stronger **mode coverage**
- Better training **stability**



Xiao, Z., Kreis, K., & Vahdat, A. (2021). Tackling the generative learning trilemma with denoising diffusion gans. ICLR 2022.

Diffusion GANs



riducendo il numero di steps

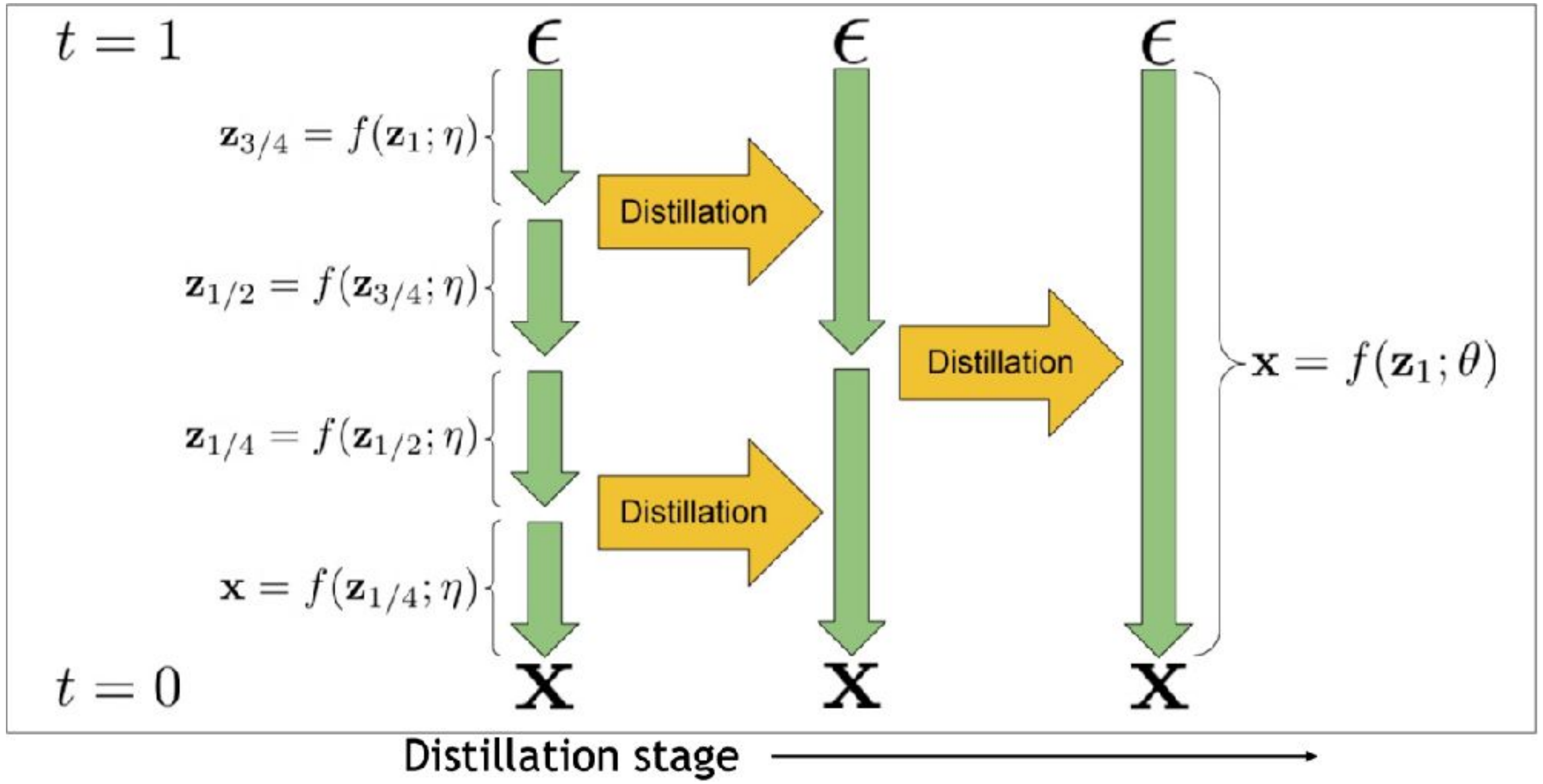
Sample Quality vs. Sampling Time

CIFAR-10 Samples

Xiao, Z., Kreis, K., & Vahdat, A. (2021). Tackling the generative learning trilemma with denoising diffusion gans. ICLR 2022.

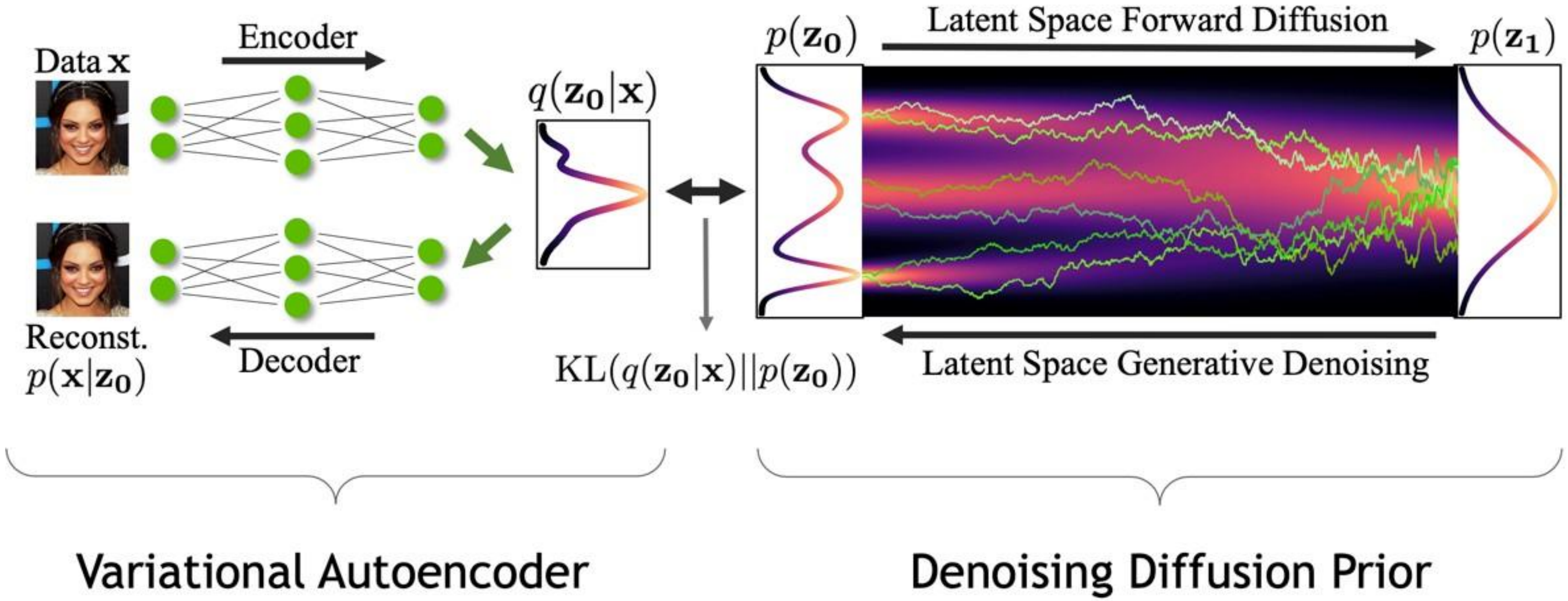
Distillation

- **Distill** a deterministic DDIM sampler to the same model architecture.
- At each stage, a “student” model is learned to **distill two adjacent sampling steps** of the “teacher” model to one sampling step.
- At next stage, the “student” model from previous stage will serve as the **new “teacher”** model.



Salimans, T., & Ho, J. (2022). Progressive distillation for fast sampling of diffusion models. ICLR 2022.

Latent-space diffusion models



- The distribution of **latent embeddings** is close to **Normal** distribution Simpler denoising and faster synthesis
- **Augmented** latent space
- **Tailored** autoencoders (graphs, text, 3D data, etc.)

Vahdat, A., Kreis, K., & Kautz, J. (2021). Score-based generative modeling in latent space. Advances in Neural Information Processing Systems, 34, 11287-11302.

Text-to-image

"a hedgehog using a calculator"

"a corgi wearing a red bowtie and a purple party hat"

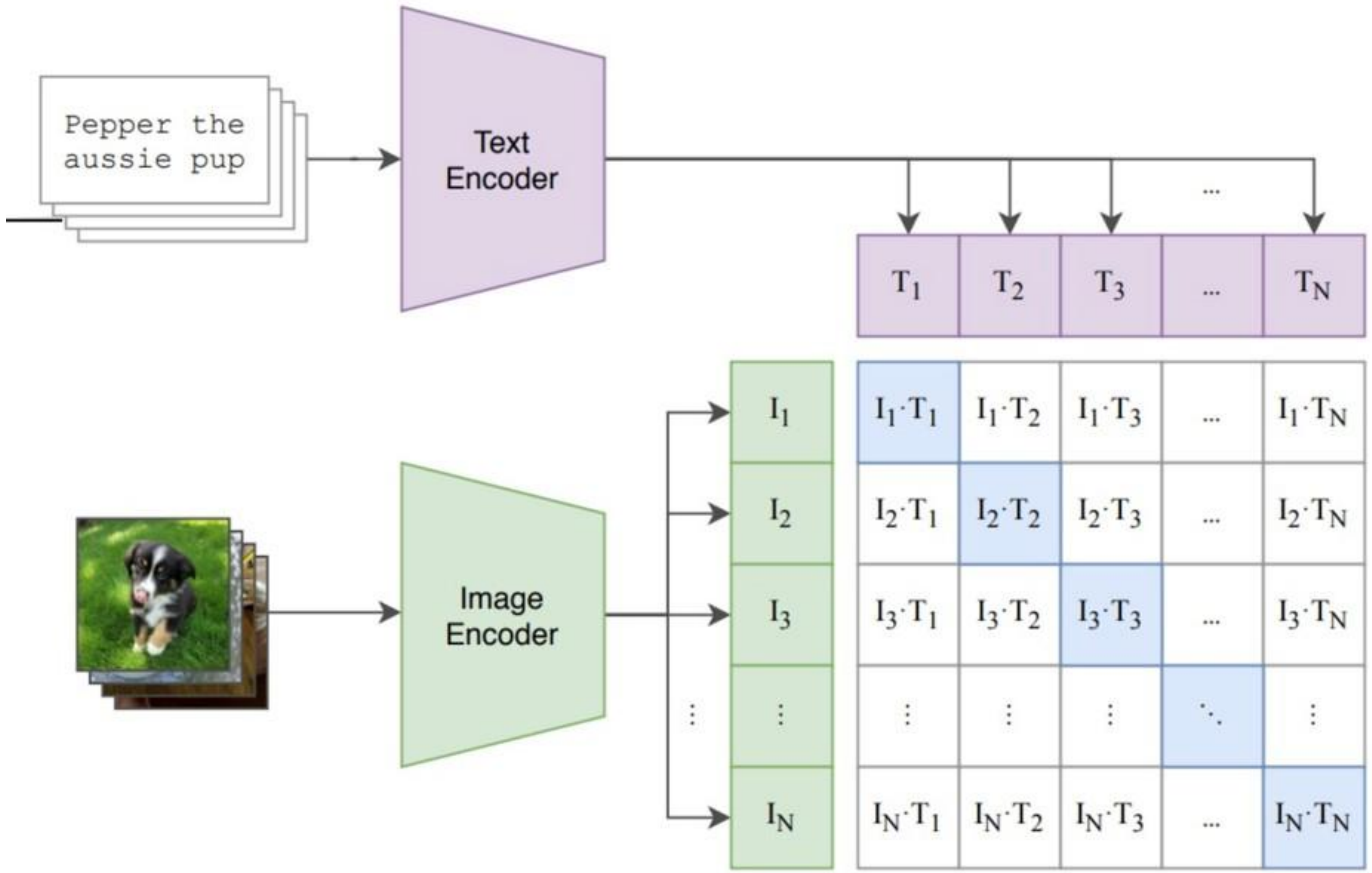
"robots meditating in a vipassana retreat"

"a fall landscape with a small cottage next to a lake"

Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., ... & Chen, M. (2021). Glide: Towards photorealistic image generation and editing with text-guided diffusion models. arXiv preprint arXiv:2112.10741.

Contrastive Language-Image Pertaining

- Jointly train a **text** encoder and an **image** encoder
- Train by maximising the **similarity** between embeddings of (text, image) pairs
- The resulting space has **semantics** for both images and text



Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., ... & Sutskever, I. (2021, July). Learning transferable visual models from natural language supervision. In International Conference on Machine Learning (pp. 8748-8763). PMLR.

Glide: IDEA

You already understand the Diffusion.

$$p_\theta(x_{t-1}|\boxed{x_t}) = \mathcal{N}(\mu_\theta(x_t), \Sigma_\theta(x_t))$$

But you need something More Controlled.

Label = “Goldfinch”

Label = “robots meditating in a vipassana retreat”

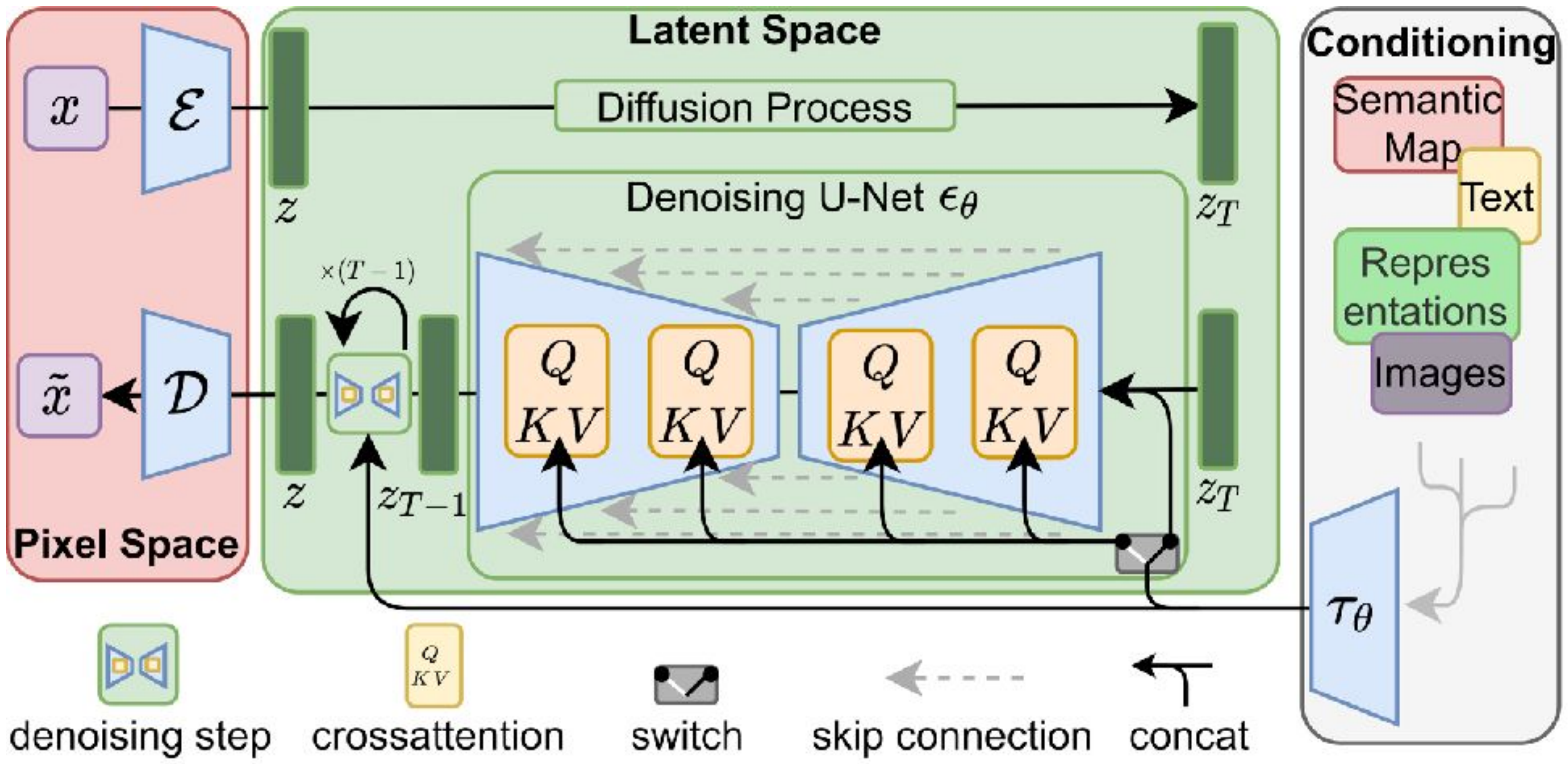
$$p_\theta(x_{t-1}|\boxed{x_t, y}) \longrightarrow \text{Text-Conditioned Diffusion}$$

USE CLIP

Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., ... & Chen, M. (2021). Glide: Towards photorealistic image generation and editing with text-guided diffusion models. arXiv preprint arXiv:2112.10741.

Text-to-image (Stable Diffusion)

- Use CLIP embeddings as conditioning on **latent diffusion** via cross-attention
- Trained on a subset of LAION-5B (original dataset has ***5 billion*** text-image pairs)
- **Fast sampling** thanks to diffusion in latent space



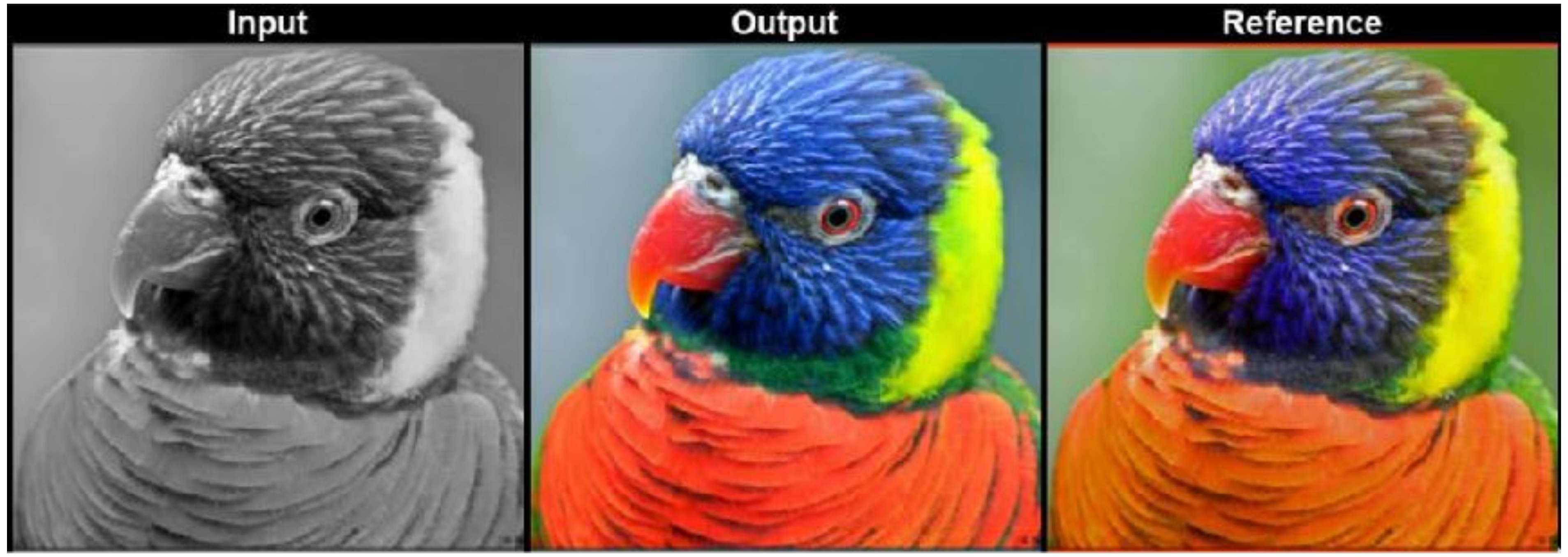
Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 10684-10695).

Diffusion Zoo

Super Resolution

Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D. J., & Norouzi, M. (2022). Image super-resolution via iterative refinement. IEEE Transactions on Pattern Analysis and Machine Intelligence.

Image-to-Image



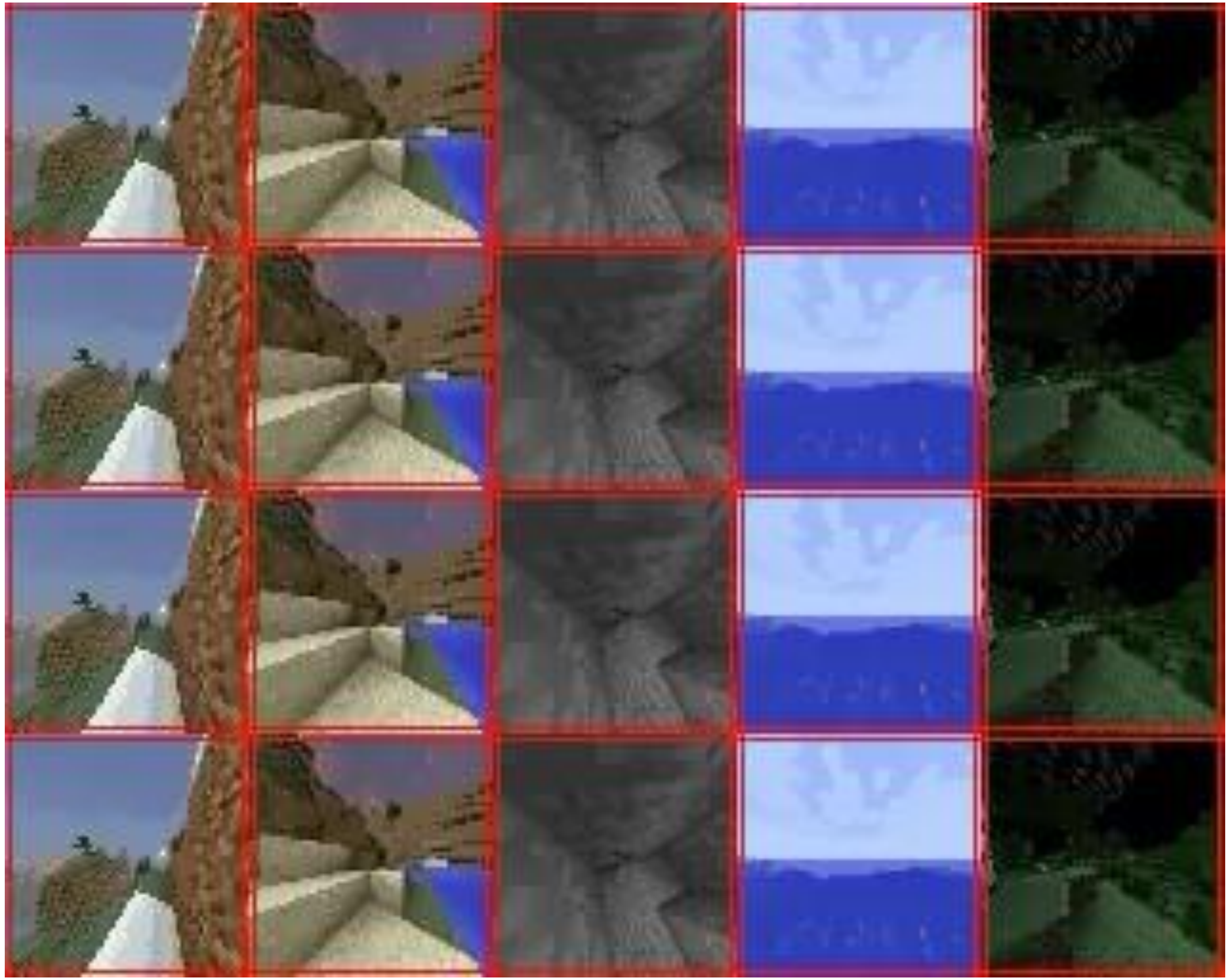
Saharia, C., Chan, W., Chang, H., Lee, C., Ho, J., Salimans, T., ... & Norouzi, M. (2022, July). Palette: Image-to-image diffusion models. In ACM SIGGRAPH 2022 Conference Proceedings (pp. 1-10).

Video Generation

Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., & Fleet, D. J. (2022). Video diffusion models. arXiv preprint arXiv:2204.03458.

https://video-diffusion.github.io/

Video Generation



Harvey, W., Naderiparizi, S., Masrani, V., Weilbach, C., & Wood, F. (2022). Flexible Diffusion Modeling of Long Videos. arXiv preprint arXiv:2205.11495

https://fdmolv.github.io/

QUESTIONS?