Generative AI and LLM

Diffusion models
CS5202

Course Instructor : Dr. Nidhi Goyal

Mahindra™ University
Global Thinkers. Engaged Leaders.

Recap

- Variational Autoencoders
- Generative Adversarial Networks
- Flow based generative models
- Normalizing Flows
- Training flow based models

Strict restriction

Every grain of sand must always correspond to exactly one grain from before.
You are never allowed to tear the sand, glue grains together, remove sand, or create new structure from nothing.

Downside of Normalizing Flows

- Every transformation must be:
  - Bijective
  - Differentiable
  - Have tractable Jacobian determinant
- Jacobian Determinant is Expensive
  - To compute exact likelihood:

$$\boxed{\log p(x) = \log p(z) + \log |\det J_f|}$$

  That determinant becomes:
    - Expensive
    - Memory heavy
    - Hard in high dimensions

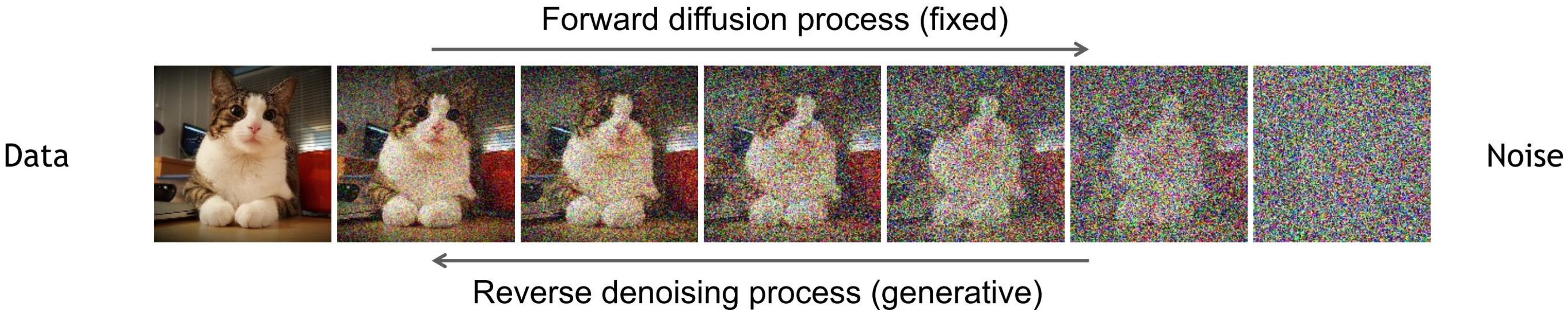
Problem with GANs

- Mode collapse
- Training instability (minimax is hard to balance)
- No likelihood estimation
- Vanishing gradients (discriminator too strong)
- No objective evaluation metric
- No latent encoding (vanilla GANs)
- Hyperparameter sensitive
- Difficult to scale to high-resolution without tricks (progressive growing, etc.)
- Non-convergent loss curves
- Diversity-quality tradeoff

Denoising Diffusion Models

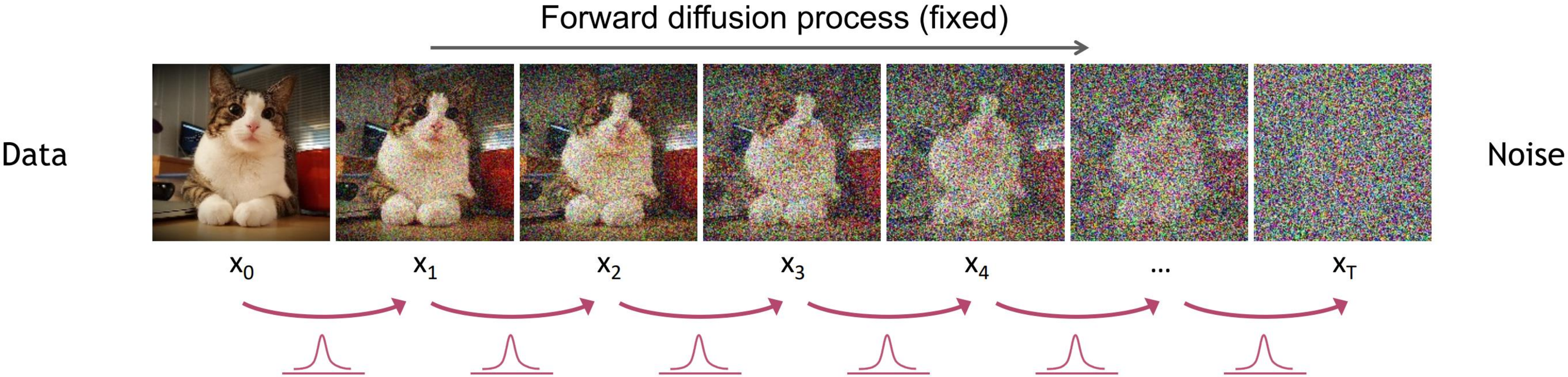
Denoising diffusion models consist of two processes:

- Forward diffusion process that gradually adds noise to input
- Reverse denoising process that learns to generate data by denoising



Forward Diffusion Process

- The formal definition of the forward process in T steps:

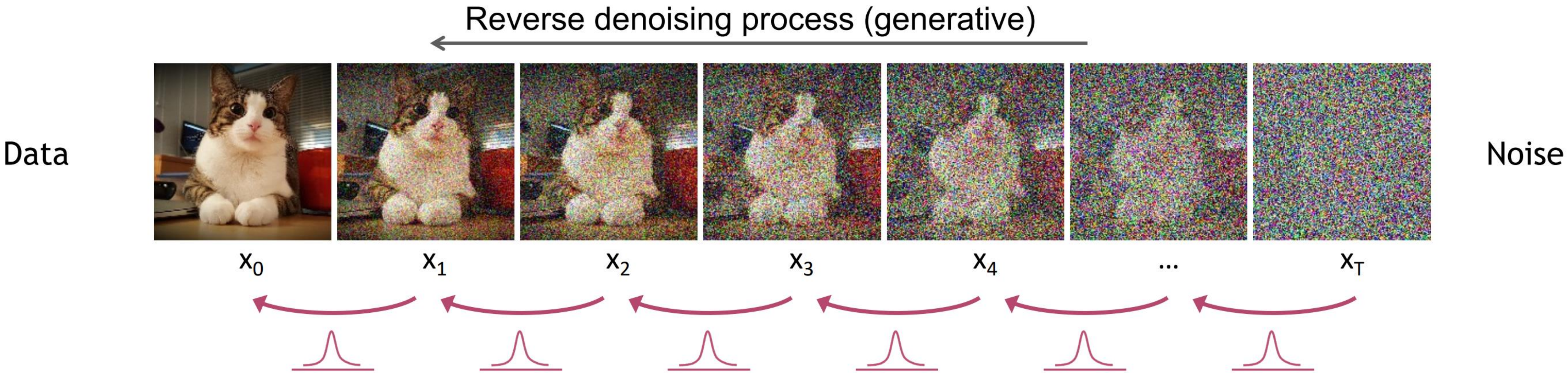


$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x_t}; \sqrt{1-\beta_t}\mathbf{x_{t-1}}, \beta_t\mathbf{I}) \quad \Rightarrow \quad q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})$$

(joint)

Reverse Diffusion Process

- Formal definition of forward and reverse processes in T steps:



$$p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I})$$

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

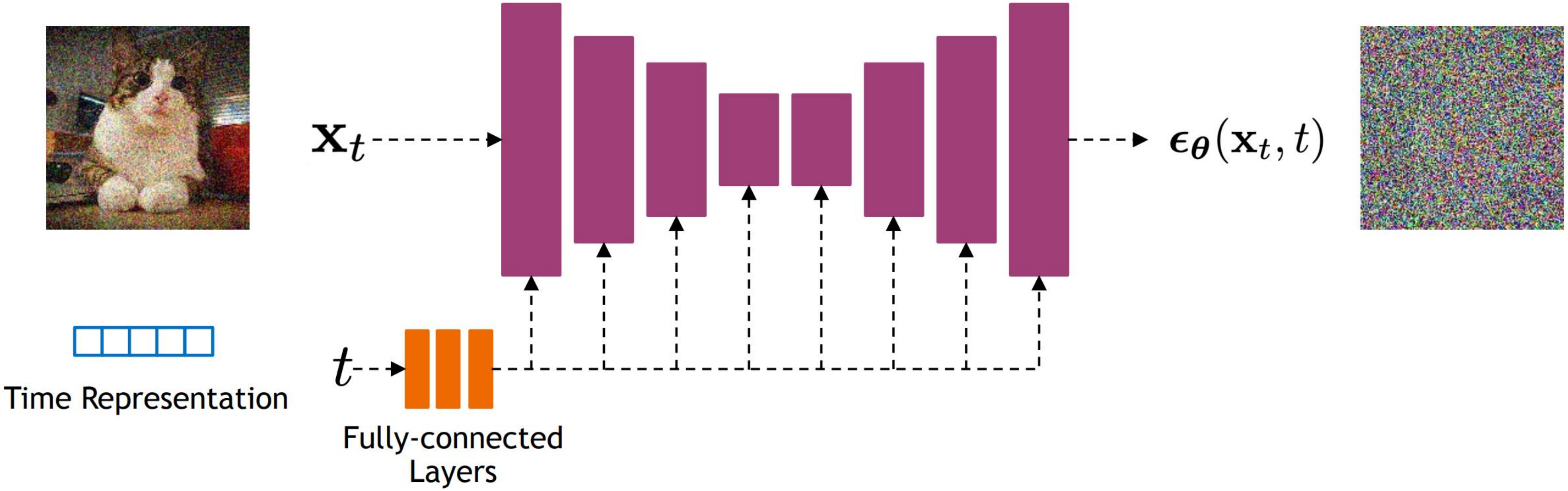
Trainable network
(U-net, Denoising Autoencoder)

$\Rightarrow$ $$p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_T)\prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$$

Similar to the generative model in hierarchical VAEs.

Implementation Architectures

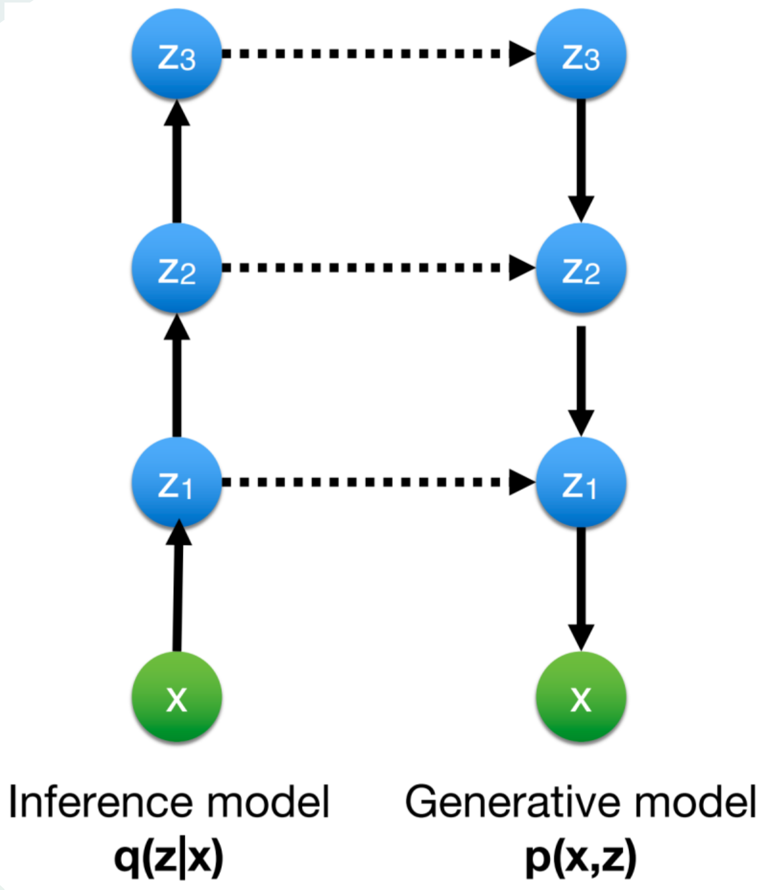
- Diffusion models often use U-Net architectures with ResNet blocks and self-attention layers



Connection to VAEs

Diffusion models can be considered as a special form of hierarchical VAEs. However, in diffusion models:

- The inference model is fixed: easier to optimize
- The latent variables have the same dimension as the data.
- The ELBO is decomposed to each time step: fast to train
- Can be made extremely deep (even infinitely deep)
- The model is trained with some reweighting of the ELBO.



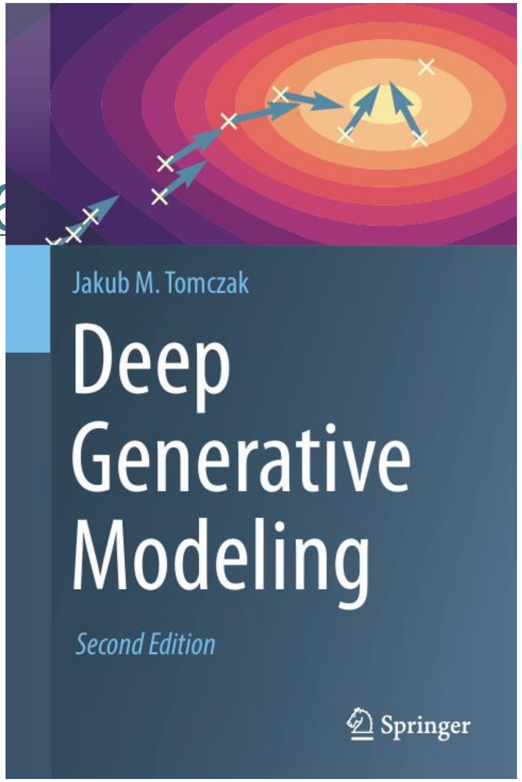
Comparison of DDGM, VAE, and Flows

Table 5.1 A comparison among DDGMs, VAEs, and flows.

Model	Training	Likelihood	Reconstruction	Invertible	Bottleneck (latents)
DDGMs	Stable	Approximate	Difficult	No	No
VAEs	Stable	Approximate	Easy	No	Possible
Flows	Stable	Exact	Easy	Yes	No

Books and lecture notes

Deep Generative Modeling



GitHub

Generative AI & LLMs · Fall 2026

References

Denoising Diffusion Probabilistic Models, https://arxiv.org/abs/2006.11239

Diffusion Models: A Comprehensive Survey of Methods and Applications, https://arxiv.org/abs/2209.00796

Understanding Diffusion Models: A Unified Perspective, https://arxiv.org/abs/2208.11970

GLOW, https://proceedings.neurips.cc/paper/2018/file/d139db6a236200b21cc7f752979132d0-Paper.pdf

Variational inference with normalizing flows, https://arxiv.org/abs/1505.05770

Normalizing Flows: An Introduction and Review of Current Methods, https://arxiv.org/abs/1908.09257

Didrik Nielsen's lecture, https://www.youtube.com/watch?v=2tVHbcUP9b8

Hans van Gorp's lecture, https://www.youtube.com/watch?v=yxVcnuRrKqQ&t=17s

Tim Salimans' lecture, https://www.youtube.com/watch?v=pea3sH6orMc