From Diffusion to Flow

Willis Ma

College of Arts and Science / Courant Institute of Mathematical Science

Oct 13th, 2023

Introduction - Generative Models

Given some x observed from underlying distribution, our interest is to find

$$q_\theta(x) \sim p(x)$$

which enables us to

- obtain samples from $q_\theta(x)$.
- compute likelihood of any x.

For high-dimensional, intractable, and multimodal real-life data distribution, this is extremely hard.

Introduction - Generative Models

- Adversarial Learning:
  - Generator - simulating sampling process.
  - Discriminator - classify samples as either real(from domain) or fake(from generator).
- Likelihood-based Learning:
  - Assigning high likelihood $\log p(x)$ to observed samples x by maximizing the Evidence Lower Bound:

$$\log p(x) \geq \mathbb{E}[\log \frac{p(x,z)}{q_\theta(z|x)}] \tag{1}$$

- Energy-based Learning:
  - Parameterize an energy function f_θ that

$$q_\theta(x) = \frac{1}{Z} e^{-f_\theta(x)} \sim p(x) \tag{2}$$

Diffusion Model

- Intersection of both Likelihood-based and Energy-based methods.
- Forward process:
  Progressively destruct an observed signal (data) to Gaussian noise
- Backward process:
  Progressively reconstruct a signal (sample) from Gaussian noise

Diffusion Model - Forward Process

Explicitly maintain the process as a Markov Chain, we have

$$q(x_1, \ldots, x_T | x_0) = \prod_{t=1}^{T} q(x_t | x_{t-1}) \tag{3}$$

Each step in the forward process is defined by

$$q(x_t | x_{t-1}) = (x_t; \sqrt{\alpha_t} x_{t-1}, (1 - \alpha_t)\mathbf{I}) \tag{4}$$

where we assume $x_0 \sim p(x)$, $x_T \sim \mathcal{N}(0, \mathbf{I})$.

Diffusion Model - Backward Process

Given our Markovian forward process, if we have a $p_\theta(x_{t-1}|x_t)$ that is strictly inverting $q(x_t|x_{t-1})$ for $\forall t \in \{1, \dots, T\}$, starting from $\varepsilon \sim \mathcal{N}(0, \mathbf{I})$, we could recursively run p_θ backward in time to reconstruct the signal.
How to obtain p_θ?

Frame Title

By (4), we can show that

$$q(x_t|x_0) = \mathcal{N}\left(\sqrt{\prod_{i=1}^{t}\alpha_i}, (1 - \prod_{i=1}^{t}\alpha_i)\mathbf{I}\right) \tag{5}$$

$$= \mathcal{N}(\sqrt{\bar{\alpha}_t}x_0, (1 - \bar{\alpha}_t)\mathbf{I}) \tag{6}$$

$q(x_{t-1}|x_t, x_0) = \mathcal{N}(\mu_q(x_t, x_0), \Sigma_q(t))$ can thus be derived by Bayes rule. Then we simply optimize $p_\theta \sim \mathcal{N}(\mu_\theta, \Sigma_q(t))$ by

$$\arg\min_\theta \|\mu_\theta(x_t, t) - \mu_q(x_t, x_0)\|^2 \tag{7}$$

Furthermore, with some reparametrization tricks we can see that (7) can be transformed into a simpler objective

$$\arg\min_\theta \omega(t)\|\varepsilon_\theta(x_t, t) - \varepsilon\|^2 \tag{8}$$

for $\varepsilon \sim \mathcal{N}(0, \mathbf{I})$.

Diffusion Model - Backward Process

As in likelihood-based methods, we could also directly optimize over ELBO as given in (1)

$$\arg\max_{\theta} \mathbb{E}[\log \frac{p_\theta(x_0, x_1, \ldots, x_T)}{q(x_1, \ldots, x_T|x_0)}] \tag{9}$$

plug in (3) and

$$p_\theta(x_0, x_1, \ldots, x_T) = p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t) \tag{10}$$

we can show that (9) is equivalent to (7) up to scaling factors. Since $p_\theta(x_{t-1}|x_t)$ does not depending on x_0, we could start from $x_T \sim \mathcal{N}(0, \mathbf{I})$ and obtain the reconstructed signal from noise.

Diffusion Model - Energy Function

From (2), we have

$$\nabla \log p_\theta(x) = \nabla \log(\frac{1}{Z}) - \nabla f_\theta(x) \simeq -\nabla f_\theta(x) \tag{11}$$

By Tweedie's formula, we have

$$\mathbb{E}_{q(x_t|x_0)}[\mu_{x_t}|x_t] = x_t + (1 - \bar{\alpha_t})\nabla \log p(x) \tag{12}$$

$$\rightarrow x_0 = \frac{x_t + (1 - \bar{\alpha_t})\nabla \log p(x)}{\sqrt{\bar{\alpha_t}}} \tag{13}$$

Plug into (7), we see that optimizing over score function is equivalent to optimizing over mean.

Diffusion - what's the caveats?

- Sampling too expensive! $T \sim 1000$
- Increasing exposure bias throughout different denoising steps.
- Unable to calculate the exact likelihood $\log p(t)$.

From Discrete to Continuous

Let's go continuous!

From Discrete to Continuous

We could rewrite (4) in terms of a perturbation kernel, that

$$x_t = \sqrt{\alpha_t} x_{t-1} + \sqrt{(1-\alpha_t)}\varepsilon \tag{14}$$

where $\varepsilon \in \mathcal{N}(0, \mathbf{I})$. Taking the limit of $T \to \infty$, the limit of the Discrete Markov Chain is given by

$$dx_t = \sqrt{\alpha(t)} x dt - \frac{1}{2}\alpha(t) dw \tag{15}$$

where w is the standard Brownian motion, and $t \in [0, 1]$. We see that (15) coincides with an Itô SDE in forward time.

From Discrete to Continuous

By (Reverse-Time Diffusion Equation), (15) has a corresponding SDE in reverse time expressed as

$$dx = [\sqrt{\alpha(t)}x - \frac{1}{4}\alpha(t)^2 \nabla \log p_\theta(x_t, t)]dt - \frac{1}{2}\alpha(t) d\bar{w} \tag{16}$$

where $d\bar{w}$ is the reverse time standard Brownian motion.

From Discrete to Continuous

With a slight abuse of notation, we denote the mean and variance of $p_t(x_t)$ α_t, σ_t. By (Applied Stochastic Differential Equations), we know

$$\frac{d\alpha_t}{dt} = \mathbb{E}\{f(t)x\} = \sqrt{\alpha(t)}\alpha_t \tag{17}$$

$$\begin{aligned}\frac{d\sigma_t}{dt} &= \mathbb{E}\{(f(t)x - \mathbb{E}[f(t)x])(x-\alpha_t)^T\} \\ &\quad + \mathbb{E}\{(x-\alpha_t)(f(t)x - \mathbb{E}[f(t)x])^T\} + \mathbb{E}\{g(t)^2\mathbf{I}\}\end{aligned} \tag{18}$$

Again, by Tweedie's formula and the fact that $x_t = \alpha_t x + \sigma_t \varepsilon$, we have

$$\nabla \log p_t(x_t) = -\sigma_t \varepsilon \tag{19}$$

From Discrete to Continuous

We see that (19) can be optimized using (8)

$$\arg\min_{\theta} \|s_\theta(x_t, t) - \nabla \log p_t(x_t)\|^2 = \arg\min_{\theta} \omega(t)\|\varepsilon_\theta(x_t, t) - \varepsilon\|^2 \tag{20}$$

and that (16) can then be readily solved by numerical methods (Euler-Maruyama) to obtain

$$x(0) \sim p(x)$$

SDE - pitfalls

- Estimated score could be inaccurate in low density areas - derailing the trajectory from the beginning.
- Fluctuating on small time interval - still demanding large number of time steps to reach high precision.
- Still unable to calculate exact likelihood.

From SDE to ODE

We know that marginal density of the forward time SDE is uniquely determined by a Fokker-Planck equation

$$\frac{\partial}{\partial t}p_t(x) = -\sum \frac{\partial}{\partial x_i}(f(t)xp_t(x)) + \frac{1}{2}\sum\sum \frac{\partial^2}{\partial x_i x_j}(g(t)p_t(x)) \tag{21}$$

from which we could derive

$$\tilde{f}(x,t) = f(t)x - \frac{1}{2}g(t)^2\nabla \log p(x) \tag{22}$$

that satisfies the continuity equation

$$\frac{\partial}{\partial t}p_t(x) = -\nabla[\tilde{f}(x,t)p_t(x)] \tag{23}$$

From SDE to ODE

$\tilde{f}(x,t)$ thus shares the marginal density as the SDE in (15). Since the corresponding diffusion term to $\tilde{f}$ is 0, we now have a probability flow ODE

$$dx = \tilde{f}(x,t)dt \tag{24}$$

with $x(0) = x \sim p(x)$

Flow ODE

It's surprising how many fast and stable numerical methods we could use to solve (24); moreover, now the likelihood can be explicitly computed by (23) with change of variable

$$\frac{\partial}{\partial t} p_t(x) = -\mathrm{div}(\tilde{f}(x, t)) \tag{25}$$

yielding another ODE to be solved.

Flow ODE

Yet the inaccuracy of score function in low density area would still deviate our ODE from its optimal trajectory; could we alleviate this issue?

Flow ODE

Yes! In fact, we could define

$$I(x_0, x_1, t) = \alpha_t x_0 + \sigma_t x_1 \tag{26}$$

for $x_0 \in p(x)$, $x_1 \in q(x)$, $\alpha_t, \sigma_t \in [0, 1]$ and that $\alpha_0 = \sigma_1 = 1$, $\alpha_1 = \sigma_0 = 0$.
Furthermore, define $v_t(I(x_0, x_1, t)) = \partial_t I(x_0, x_1, t)$. For p_t that satisfies (23) with v_t, it can be shown $p_1 \sim q$, $p_0 \sim p$. To approximate v_t, we simply optimize over the objective

$$\arg\min_{\theta} \| v(I(x_0, x_1, t)) - (\dot{\alpha}_t x_0 + \dot{\sigma}_t x_1) \|^2 \tag{27}$$

Flow ODE

- Fast sampling speed.
- Exact likelihood.
- When $x_1 \sim \mathcal{N}(0, \mathbf{I})$, $I(x_0, x_1, t)$ corresponds to perturbation kernel of score-based model with exact same α_t and σ_t in (17) and (18). Yet, the dynamics of I would not vanish near 0 and 1, preventing inaccuracy from initial time steps when sampling.

Experiments

We will be conducting experiments using both Diffusion model, Score-based Model, and Flow-based Model, and examining their performance on conditional image generation task.

Density Path

We followed Yang Song's Score-Based Generative Model paper, using

$$\alpha_t = \exp[-\frac{1}{4}t^2(\beta_{\max} - \beta_{\min}) - \frac{1}{2}t\beta_{\min}] \tag{28}$$

$$\sigma_t = \sqrt{1 - 1\exp[-\frac{1}{2}t^2(\beta_{\max} - \beta_{\min}) - t\beta_{\min}]} \tag{29}$$

where we take $\beta_{\max} = 20$, $\beta_{\min} = 0.1$.

Backbone - DiT

To estimate ε_θ (8), s_θ (13), v_θ (27), we used Scalable Diffusion Transformer (DiT) as our backbone. The structure is as follows:

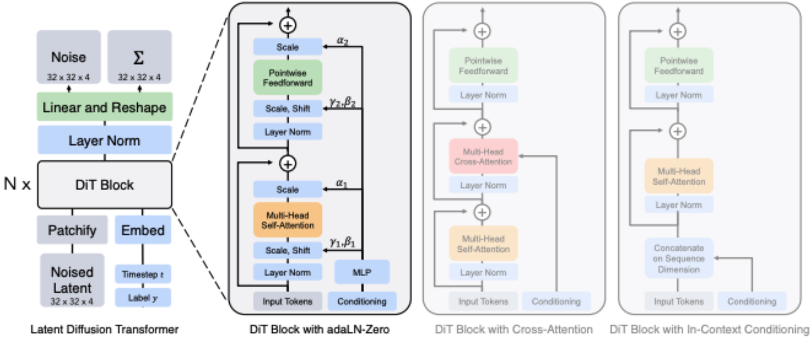


Figure 1: DiT structure.

Backbone - DiT

Different configurations of DiT are provided

Model	Layers N	Hidden size d	Heads	Gflops $(I{=}32,\ p{=}4)$
DiT-S	12	384	6	1.4
DiT-B	12	768	12	5.6
DiT-L	24	1024	16	19.7
DiT-XL	28	1152	16	29.1

Figure 2: DiT configurations.

We will be using DiT-B for all of our experiments.

Dataset - ImageNet

We conducted all of our experiments on ImageNet, a large scale dataset with $\sim$ 1.2 million images splitted into 1000 different classes.
We train all of our three models on downsampled space $\mathcal{Z}$ of $256x256x3$ resolution images from ImageNet, where $\mathcal{Z} \subset \mathbb{R}^{32x32x4}$, with class labels inputs as extra conditionings.

Downsampling - Variational Autoencoder (VAE)

We use an off-the-shelf pre-trained Variational Autoencoder model to downsample original images. It contains an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$, that

$$\begin{aligned}\mathcal{E}(x) &\sim p(z|x)\\ \mathcal{D}(z) &\sim q(x|z)\end{aligned} \tag{30}$$

so that $\mathcal{D}(\mathcal{E}(x)) \sim x$

Metric - Fréchet inception distance

We use Fréchet Inception Distance (FID) as our evaluation metric, which is defined as

$$d_k(\mathcal{N}(\mu_k, \Sigma_k), \mathcal{N}(\mu', \Sigma')) = \|\mu_k - \mu'\|^2 + \mathrm{tr}(\Sigma_k + \Sigma' - 2(\Sigma_k^{\frac{1}{2}} \Sigma' \Sigma_k^{\frac{1}{2}})^{\frac{1}{2}}) \tag{31}$$

where we obtain μ', Σ' from ImageNet training data, and μ_k, Σ_k from k generated samples of our models. We evaluate FID-k for $k \in \{10000, 50000\}$.

Quantitative results - FID-10K

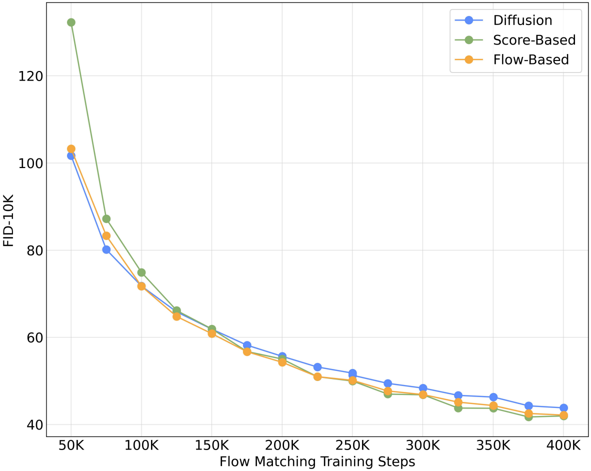


Figure 3: FID-10K results.

Quantitative results - FID-50K

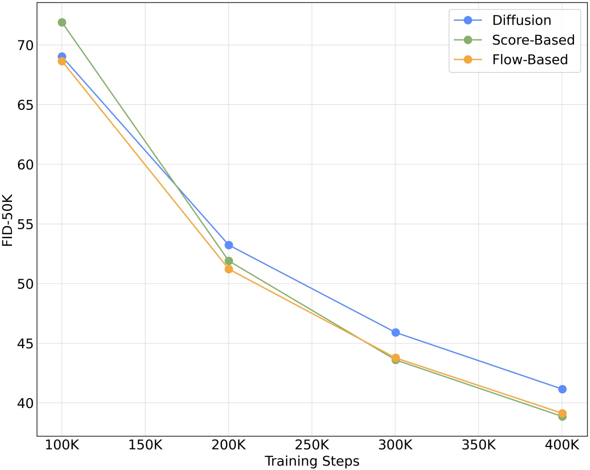


Figure 4: FID-50K results.

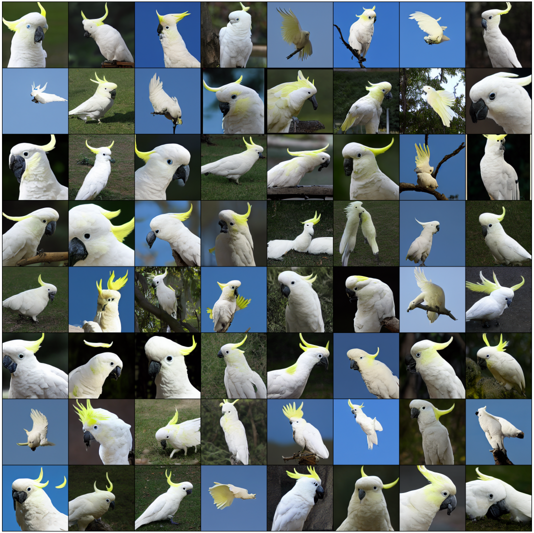
Quantitative results

Model	FID-10K	FID-50K
Diffusion	43.819	41.153
Score	**41.734**	**38.858**
Flow	42.163	39.125

Table 1: FID scores.

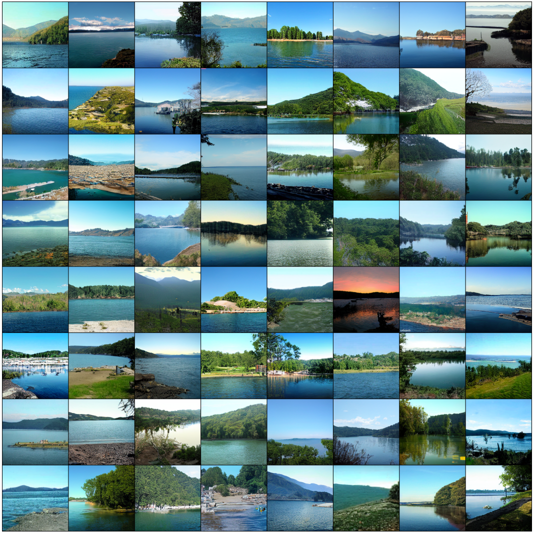
Qualitative results

Qualitative results



Qualitative results

Qualitative results



Conclusion

We examined the performance of Diffusion, Score-Based and Flow-Based models on large scale conditional image generation tasks, demonstrated their capabilities in generating high-quality images, and showed the discrepancy in FID scores under different objective. We plan to explore further and see

- what contribute to the gap in FID score?
- will the performance change with different density path?

Thank you!