DDPM loss design

We now know q and p are both normal distribution of the probability projection model (diffusion model)

We proved the each step of denoising can be numerically calculated

We want the model's parameters (theta) can be better, by having a better estimation of each time steps' denoising process. (When the model defined by theta can have a closer denoising to the theoretical 'reverse diffuse')

We use the KL divergence to measure the difference of the processes, and we use the maximum likelihood estimation to seek the theta parameters. So we can build a loss that supervising each step t.

⇒ in Diffusion we have ✡ $\begin{cases} q(x_t \mid x_{t-1}) & \overset{t-1}{O} \rightarrow \overset{t}{O} \quad \text{diffuse} \\ q(x_{t-1} \mid x_t) & \overset{t-1}{O} \leftarrow \overset{t}{O} \quad \text{reverse diffuse} \\ p_\theta(x_{t-1} \mid x_t) & \overset{t-1}{O} \leftarrow \overset{t}{O} \quad \text{denoise } \theta \end{cases}$

Previous

and, ① $q(x_{1:T} \mid x_0) = \prod_{t=1}^{T} q(x_t \mid x_{t-1})$ (all the diffuse)

② $p_\theta(x_{0:T}) = p(x_T) \prod^{T} p_\theta(x_{t-1} \mid x_t)$ (all the denoise)

Our aim is to use p_θ to estimate the reverse q.

if we can have all the $x_0, x_1 \cdots x_T$

x_T

$O \rightarrow O \rightarrow O \rightarrow$

$x_0 \leftarrow \leftarrow \leftarrow$

the process supervise Target here is to find a θ

who makes $p_\theta(x_{0:T} \mid x_0)$ is very close to $q(x_{0:T} \mid x_0)$

So we need two distribution (abstractive) things are very close

---

Measure KL Divergence (information theory)

Def:

$$D_{KL}(P \| Q) = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} \, dx \quad \text{(Def)}$$

As we know

1. $D_{KL}(P \| Q) \neq D_{KL}(Q \| P)$ (0)

2. $D_{KL}(P \| Q) \geq 0$; "=" at $P = Q$ (1)

Maximum likelyhood estimation (MLE)

Variance

now we know we can calculate $\mu_\theta(x_t, t)$ and $\Sigma_\theta(x_t, t)$ of p_θ

but how to get the correct θ so the value works?

denoise

MLE: $\mathcal{L}$ is a function about θ; when we have the given output of x, the value of $\mathcal{L}$ is possibility of obtain x under θ:

$$\mathcal{L}(\theta|x) = P(X=x|\theta)$$

we want the chance to be the highest (maximum-ization)

also as

$$\Rightarrow \min -\mathcal{L} \Rightarrow \min -\mathcal{L}(\theta|x) = \min -P(X=x|\theta) \quad P_\theta(X)$$

How to build this $\mathcal{L}$?

$$P_\theta(x_0) := \int P_\theta(x_{0:T})\, dx_{1:T}$$

by right this is the mathmatic Target def.

Search for the minimal negative log Expectation

$$\mathcal{L} = E_q(-\log P_\theta(x_0))$$

最小化负对数似然

$0 \rightarrow 0 \rightarrow 0$

$x_0 \qquad\qquad T$

We have

$$D_{KL}(q(x_{1:T}|x_0)\,||\,P_\theta(x_{1:T}|x_0)) \geq 0 \quad \text{(Def) (1)}$$

add something ≥ 0

$$\Rightarrow -\log P_\theta(x_0) \leq -\log P_\theta(x_0) + D_{KL}(q(x_{1:T}|x_0)\,||\,P_\theta(x_{1:T}|x_0)) \quad \text{Def?}$$

$$E[-\log P_\theta(x_0)] \leq E[-\log P_\theta(x_0)] + E_{x_{1:T}\sim q(x_{1:T}|x_0)}\left[\log \frac{q(x_{1:T}|x_0)}{P_\theta(x_{0:T})/P_\theta(x_0)}\right]$$

remove

$$\Rightarrow -\log P_\theta(x_0) \leq -\log P_\theta(x_0) + E_q\left[\log \frac{q(x_{1:T}|x_0)}{P_\theta(x_{0:T})} + \log P_\theta(x_0)\right]$$

$$\Rightarrow -\log P_\theta(x_0) \leq E_q\left[\log \frac{q(x_{1:T}|x_0)}{P_\theta(x_{0:T})}\right]$$

Variational lower bound (VLB)

Evidence lower bound (ELBO)

VAE

let $L_{VLB} = E_{q(x_{0:T})}\left[\log \frac{q(x_{1:T}|x_0)}{P_\theta(x_{0:T})}\right] \geq -E_{q(x_0)}[\log P_\theta(x_0)]$

Then

$$
\begin{aligned}
L_{\mathrm{VLB}} &= \mathbb{E}_{q(\mathbf{x}_{0:T})}\Big[\log \frac{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{p_\theta(\mathbf{x}_{0:T})}\Big] \\
&= \mathbb{E}_q\Big[\log \frac{\prod_{t=1}^T q(\mathbf{x}_t|\mathbf{x}_{t-1})}{p_\theta(\mathbf{x}_T)\prod_{t=1}^T p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}\Big] \\
&= \mathbb{E}_q\Big[-\log p_\theta(\mathbf{x}_T) + \sum_{t=1}^T \log \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1})}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}\Big] \\
&= \mathbb{E}_q\Big[-\log p_\theta(\mathbf{x}_T) + \sum_{t=2}^T \log \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1})}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} + \log \frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)}\Big] \\
&= \mathbb{E}_q\Big[-\log p_\theta(\mathbf{x}_T) + \sum_{t=2}^T \log \Big(\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}\cdot\frac{q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)}\Big) + \log \frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)}\Big] \\
&= \mathbb{E}_q\Big[-\log p_\theta(\mathbf{x}_T) + \sum_{t=2}^T \log \frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} + \sum_{t=2}^T \log \frac{q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)} + \log \frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)}\Big]
\end{aligned}
$$

$$= \mathbb{E}_q\Big[-\log p_\theta(\mathbf{x}_T) + \sum_{t=2}^{T} \log \frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} + \log \frac{q(\mathbf{x}_T|\mathbf{x}_0)}{q(\mathbf{x}_1|\mathbf{x}_0)} + \log \frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)} \Big]$$

$$= \mathbb{E}_q\Big[\log \frac{q(\mathbf{x}_T|\mathbf{x}_0)}{p_\theta(\mathbf{x}_T)} + \sum_{t=2}^{T} \log \frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} - \log p_\theta(\mathbf{x}_0|\mathbf{x}_1) \Big]$$

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

$\Rightarrow$ $L_{VLB} := L_T + L_{T-1} + \cdots L_0$

$L_T := D_{KL}(q(x_T|x_0) \,||\, p_\theta(x_T))$

$L_{t-1} := D_{KL}(q(x_{t-1}|(x_t, x_0)) \,||\, p_\theta(x_{t-1}|x_t))$, $1 \le t \le T$

$L_0 := -\log p_\theta(x_0|x_1)$

---

for L_T: its based on $x_T \sim N(0,1)$.

So we can ignore it. (first step can go anywhere)

why? q has no parameter. and p_θ cannot be supervised based on $x_T \sim N(0,1)$

---

for L_0: not very helpful.

The prove will be explored later.

---

for L_t: $D_{KL}(q(x_{t-1}|(x_t, x_0)) \,||\, p_\theta(x_{t-1}|x_t))$, $1 \le t \le T-1$

$q(x_{t-1}|(x_t, x_0))$ can be numerically solve

$$\tilde{\mu}_t = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t\right)$$

$$\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t} \cdot \beta_t$$

dim dim dim
mask mesh mesh

$p_\theta(x_{t-1}|x_t)$ is also a gaussian distribution (move + rescale)

given as $p_\theta(x_{t-1}|x_t) = N(x_{t-1}; \mu_\theta(x_t,t), \Sigma_\theta(x_t,t))$

---

the D_{KL} of two gaussian $\underset{(\mu_1,\sigma_1)}{p}$, $\underset{(\mu_2,\sigma_2)}{q}$ can be given

Fomular of $D_{KL}(p,q)$

$$D_{KL}(p,q) = \log\frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 + (\mu_1-\mu_2)^2}{2\sigma_2^2} - \frac{1}{2}$$

---

to optimize the D_{KL} of p_θ and q.

$0 \to 0$

the $\tilde{\beta}_t$ of q and $\Sigma_\theta(x_t,t)$ of p are all constant witch are invariance to θ optimazation. (remove) in L

$\Rightarrow$ So L_t only consider about $\tilde{\mu}_t$ and $\mu_\theta(x_t,t)$

$$L_t = E_q\left[\|\tilde{\mu}_t(x_t,x_0) - \mu_\theta(x_t,t)\|^2\right]$$

$$= E_{x_0,\varepsilon}\left[\left\|\frac{1}{\sqrt{\alpha_t}}\left(x_t(x_0,\varepsilon) - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon\right) - \mu_\theta(x_t(x_0,\varepsilon),t)\right\|^2\right]$$

both of the mean value is about x_t under x_0 and ε $\varepsilon \sim N(0,1)$

assume p_θ and q (reverse) has the same mean value

$$\Rightarrow \quad \mu_\theta(x_t(x_0,\varepsilon),t) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(x_t,t)\right)$$

unknown

put it into the above L_t, to get the lowest D_{KL} is now get two means that are close to each other

$$L_t = E_{x_0,\varepsilon}\left[\left\|\frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon\right) - \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(x_t,t)\right)\right\|^2\right], \varepsilon \sim N(0,1)$$

$\Downarrow$ mean value estimation now become noise estimation

$$L_t \propto E_{x_0,\epsilon}\left[\|\epsilon - \underbrace{\epsilon_\theta(x_t,t)}\|^2\right], \quad \epsilon \sim N(0,1)$$ remove all constant

ϵ_θ is a noise given by x_t and time step t.

$$= E_{x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}\,x_0 + \sqrt{1-\bar{\alpha}_t}\,\epsilon, t)\|^2\right], \quad \epsilon \sim N(0,1)$$

$L_t \Rightarrow l_2\text{-loss}$ → $\epsilon_\theta(\sqrt{\bar{\alpha}_t}\,x_0 + \sqrt{1-\bar{\alpha}_t}\,\epsilon, t)$

→ ϵ, $\epsilon \sim N(0,1)$

---

loss ↑ to is define — the l_2-loss at the timestep $t \in [0, T]$

$$loss_{simple}(\theta) := E_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(\underbrace{\sqrt{\bar{\alpha}_t}\,x_0 + \sqrt{1-\bar{\alpha}_t}\,\epsilon}_{\text{dimer image } + \, \epsilon \, \cdot \, \text{something}}, t)\|^2\right]$$

$$\Rightarrow L_{simple} = E_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(x_t,t)\|^2\right]$$

estimating a noise ϵ,
who was used to generate a dimer & noised "Image" at t.
based on it

$$loss = l_2(\epsilon, I(\epsilon, t, \beta))$$

time step, all settings

At training and inference time, we know the $\boldsymbol{\beta}$'s, $\boldsymbol{\alpha}$'s, and $\mathbf{x_t}$. So our **model only needs to predict the noise at each timestep.** The simplified (after ignoring some weighting terms) loss function used in the *Denoising Diffusion Probabilistic Models* is as follows:

$$L_{\mathrm{simple}}(\theta) := \mathbb{E}_{t,\mathbf{x}_0,\boldsymbol{\epsilon}}\Big[\big\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)\big\|^2\Big]$$

Comparing just the noise.

Which is basically:

$$L_{\text{simple}} = E_{t,x_0,\epsilon}\left[||\epsilon - \epsilon_\theta(x_t, t)||^2\right]$$

This is the final loss function we use to train DDPMs, which is just a "Mean Squared Error" between the noise added in the forward process and the noise predicted by the model. This is the most impactful contribution of the paper Denoising Diffusion Probabilistic Models.

It's awesome because, beginning from those scary-looking ELBO terms, we ended up with the simplest loss function in the entire machine learning domain.