DIFFUSION MODELS

1 DEEP UNSUPERVISED LEARNING USING NONEQUILIBRIUM THERMODYNAMICS (SOHL-DICKSTEIN ET AL., 2015)

1.1 IDEA

Create a generative Markov chain that transforms a simple distribution into the target distribution by reversing a diffusion process. Some related work:

1. wake-sleep algorithm: train inference and generative model against each other
2. generative stochastic networks: train Markov kernel to match its equilibrium distribution to data distribution
3. neural autoregressive distribution estimators

1.2 FORWARD PROCESS

Start from data distribution $q(x^{(0)})$. Convert it into simple distribution $\pi(y)$. Forward sequence is given by:

$$q(x^{(t)}|x^{(t-1)}) = T_\pi(x^{(t)}|x^{(t-1)};\beta_t)$$

where the Markov diffusion kernel $T_\pi(y|y';\beta)$ has:

$$\pi(y) = \int dy' T_\pi(y|y';\beta)\pi(y')$$

and β is the diffusion rate. Then the joint of whole forward sequence is:

$$q(x^{(0\ldots T)}) = q(x^{(0)})\Pi_{t=1}^{T} q(x^{(t)}|x^{(t-1)})$$

1.3 REVERSE PROCESS

Reverse distribution $q(x^{(t-1)}|x^{(t)})$ depends on whole process and cannot be easily estimated. We train a generative model $p(x^{(0\ldots T)})$. Start from prior:

$$p(x^{(T)}) = \pi(x^{(T)})$$

and follow same trajectory as forward trajectory in opposite direction:

$$p(x^{(0\ldots T)}) = p(x^{(T)})\Pi_{t=1}^{T} p(x^{(t-1)}|x^{(t)})$$

1.3.1 OPTIMIZATION OBJECTIVE SETUP

Training amounts to maximizing data log likelihood:

$$L = \mathbb{E}_{q(x^{(0)})} \log p(x^{(0)})$$

The probability the generative model assigns to the data:

$$p(x^{(0)}) = \int dx^{(1\ldots T)} p(x^{(0\ldots T)})$$

This is intractable as written above. Convert to an expectation over forward trajectories:

$$p(x^{(0)}) = \int dx^{(1\ldots T)} q(x^{(1\ldots T)}|x^{(0)}) \frac{p(x^{(0\ldots T)})}{q(x^{(1\ldots T)})} = \mathbb{E}_{q(x^{(1\ldots T)}|x^{(0)})} \frac{p(x^{(0\ldots T)})}{q(x^{(1\ldots T)})}$$

The fraction above:

$$\frac{p(x^{(0\ldots T)})}{q(x^{(1\ldots T)})} = p(x^{(T)})\Pi_{t=1}^{T} \frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t)}|x^{(t-1)})}$$

1.3.2 Optimization Objective and ELBO

OK so we wrote

$$L = \mathbb{E}_{q(x^{(0)})} \log p(x^{(0)})$$

where

$$\log p(x^{(0)}) = \log \mathbb{E}_{q(x^{(1\dots T)}|x^{(0)})} p(x^{(T)}) \Pi_{t=1}^{T} \frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t)}|x^{(t-1)})}$$

Putting them together:

$$L = \mathbb{E}_{q(x^{(0)})} \log \left[\mathbb{E}_{q(x^{(1\dots T)}|x^{(0)})} p(x^{(T)}) \Pi_{t=1}^{T} \frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t)}|x^{(t-1)})} \right]$$

From Jensen we exchange the log and expectation above to get ELBO:

$$L \geq \mathbb{E}_q \log \left[p(x^{(T)}) \Pi_{t=1}^{T} \frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t)}|x^{(t-1)})} \right]$$

1.4 Simplifying ELBO

1.4.1 Peel off $p(x^{(T)})$ into cross-entropy term

$$\begin{aligned} K &= \mathbb{E}_q \log \left[p(x^{(T)}) \Pi_{t=1}^{T} \frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t)}|x^{(t-1)})} \right] \\ &= \mathbb{E}_q \log \left[\Pi_{t=1}^{T} \frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t)}|x^{(t-1)})} \right] + \mathbb{E}_{q(x^{(T)})} \log p(x^{(T)}) \\ &= \mathbb{E}_q \log \left[\Pi_{t=1}^{T} \frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t)}|x^{(t-1)})} \right] - H\left(q(x^{(T)}), p(x^{(T)}) \right) \end{aligned}$$

1.4.2 Write log of product as sum of logs and keep first term apart

$$K = \sum_{t=2}^{T} \mathbb{E}_q \log \left[\frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t)}|x^{(t-1)})} \right] + \mathbb{E}_q \log \left[\frac{p(x^{(0)}|x^{(1)})}{q(x^{(1)}|x^{(0)})} \right] - H\left(q(x^{(T)}), p(x^{(T)}) \right)$$

1.4.3 Bayes to simplify each term in the sum from $t = 2$ to T

$$\frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t)}|x^{(t-1)})} = \frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t)}|x^{(t-1)}, x^{(0)})} = \frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t-1)}|x^{(t)}, x^{(0)})} \frac{q(x^{(t-1)}|x^{(0)})}{q(x^{(t)}|x^{(0)})} \tag{1}$$

where we used that q is Markovian to introduce the conditioning on $x^{(0)}$ because this will lead to working with known qs that we can sample from and evaluate KL divergences with.

1.4.4 Disregard Edge Effect

For the $t = 1$ term, we set the final step of the reverse trajectory to be identical to the corresponding forward diffusion step:

$$p(x^{(0)}|x^{(1)}) = q(x^{(1)}|x^{(0)}) \frac{\pi(x^{(0)})}{\pi(x^{(1)})}$$

and now $\mathbb{E}_q \log \left[\frac{p(x^{(0)}|x^{(1)})}{q(x^{(1)}|x^{(0)})} \right]$ is zero.

1.4.5 Putting everything back together

Going back to K it now looks like:

$$\begin{aligned} K &= \sum_{t=2}^{T} \mathbb{E}_q \log \left[\frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t-1)}|x^{(t)}, x^{(0)})} \frac{q(x^{(t-1)}|x^{(0)})}{q(x^{(t)}|x^{(0)})} \right] - H\left(q(x^{(T)}), p(x^{(T)}) \right) \\ &= \mathbb{E}_q \left\{ \sum_{t=2}^{T} \log \left[\frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t-1)}|x^{(t)}, x^{(0)})} \right] + \sum_{t=2}^{T} \log \left[\frac{q(x^{(t-1)}|x^{(0)})}{q(x^{(t)}|x^{(0)})} \right] \right\} - H\left(q(x^{(T)}), p(x^{(T)}) \right) \end{aligned}$$

1.4.6 IDENTIFY MORE ENTROPY TERMS

Terms in second sum above:

$$\begin{aligned}\mathbb{E}_q \log\left[\frac{q(x^{(t-1)}|x^{(0)})}{q(x^{(t)}|x^{(0)})}\right] &= \mathbb{E}_q \log q(x^{(t-1)}|x^{(0)}) - \mathbb{E}_q \log q(x^{(t)}|x^{(0)}) \\ &= -H_q(x^{(t-1)}|x^{(0)}) + H_q(x^{(t)}|x^{(0)})\end{aligned}$$

So the second sum above telescopes and K looks like:

$$K = \sum_{t=2}^{T} \mathbb{E}_q \log\left[\frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t-1)}|x^{(t)}, x^{(0)})}\right] + H_q(x^{(T)}|x^{(0)}) - H_q(x^{(1)}|x^{(0)}) - H\left(q(x^{(T)}), p(x^{(T)})\right)$$

1.4.7 IDENTIFY KL DIVERGENCE TERMS

Terms in remaining sum above:

$$\mathbb{E}_q \log\left[\frac{p(x^{(t-1)}|x^{(t)})}{q(x^{(t-1)}|x^{(t)}, x^{(0)})}\right] = -\mathbb{E}_q \mathrm{D}_{\mathrm{KL}}\left(q(x^{(t-1)}|x^{(t)}, x^{(0)})|p(x^{(t-1)}|x^{(t)})\right)$$

1.4.8 FINAL ELBO OBJECTIVE

$$\begin{aligned}K = &-\sum_{t=2}^{T} \mathbb{E}_q \mathrm{D}_{\mathrm{KL}}\left(q(x^{(t-1)}|x^{(t)}, x^{(0)})|p(x^{(t-1)}|x^{(t)})\right) \\ &+ H_q(x^{(T)}|x^{(0)}) - H_q(x^{(1)}|x^{(0)}) - H\left(q(x^{(T)}), p(x^{(T)})\right)\end{aligned} \tag{2}$$

Note that the entropies can be analytically computed, and the KL divergence can be analytically computed given $x^{(0)}$ and $x^{(t)}$.

1.5 MODEL

As aforementioned we learn the generative reverse process $p(x^{(t-1)}|x^{(t)})$. Kolmogorov forward and backward equations show that for many forward diffusion processes, the reverse diffusion processes can be described using the same functional form.

Consider gaussian diffusion for continuous data and binomial diffusion for discrete data. Below we review setup for each following the table in appendix.

Prior π for generative model:

1. gaussian diffusion: $\pi = \mathcal{N}(0, I)$
2. binomial case: $\pi = \mathcal{B}(0.5)$

Diffusion kernels (forward markov transitions):

1. gaussian diffusion: $q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I)$
2. binomial case: $q(x_t|x_{t-1}) = \mathcal{B}(x_t; x_{t-1}(1-\beta_t) + 0.5\beta_t)$

Reverse diffusion kernel:

1. gaussian diffusion: $p(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; f_\mu(x_t, t), f_\Sigma(x_t, t))$
2. binomial case: $p(x_{t-1}|x_t) = \mathcal{B}(x_{t-1}; f_b(x_t, t))$

Training targets:

1. gaussian diffusion: noise schedule β_t, reverse diffusion parameters $f_\mu(x_t, t), f_\Sigma(x_t, t))$
2. binomial case: reverse diffusion parameters $f_b(x_t, t)$

2 DENOISING DIFFUSION PROBABILISTIC MODELS (HO ET AL., 2020)

2.1 OVERVIEW

This paper was the first to show diffusion models actually are capable of generating high quality samples. They only consider gaussian diffusion and introduce a weighted variational bound designed according to a novel connection between diffusion probabilistic models and denoising score matching with Langevin dynamics. Models naturally admit a progressive lossy decompression scheme that can be interpreted as a generalization of autoregressive decoding. The downside is majority of models' lossless codelengths describe imperceptible image details and they do not have competitive log likelihoods compared to other likelihood-based models.

2.2 NICE PROPERTIES

They explain some of the nice things we can do because we assumed the forward Markov Chain has Gaussian transitions. Forward transitions are:

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

Introduce notation:

$$\alpha_t = 1 - \beta_t$$

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

2.2.1 SAMPLE FROM FORWARD PROCESS

Forward transitions can also be written as:

$$x_t = \sqrt{1-\beta_t}x_{t-1} + \sqrt{\beta_t}z_{t-1}$$

with $z_{t-1} \propto \mathcal{N}(0, 1)$. Using notation above:

$$\begin{aligned}
x_t &= \sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}z_{t-1} \\
&= \sqrt{\alpha_t}\left[\sqrt{\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_{t-1}}z_{t-2}\right] + \sqrt{1-\alpha_t}z_{t-1} \\
&= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_t}z_{t-1} + \sqrt{\alpha_t - \alpha_t\alpha_{t-1}}z_{t-2} \\
&= \sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\bar{z}_{t-2} \\
&= \cdots = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}z
\end{aligned}$$

We have merged all the additive Gaussian noise throughout the forward process and can write the distribution of x_t as:

$$q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I)$$

2.2.2 TRACTABLE FORWARD PROCESS POSTERIORS (CONDITIONED ON x_0)

Using Bayes we see forward process posteriors are tractable when conditioned on x_0:

$$q(x_{t-1}|x_t, x_0) = \mathcal{N}(x_{t-1}; \tilde{\mu}_t(x_t, x_0), \tilde{\beta}_t I)$$

where

$$\tilde{\mu}_t(x_t, x_0) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}x_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}x_t \tag{3}$$

and

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

This is why in Eq. 1 we introduced the conditioning on x_0.

2.3 LOSS FUNCTION

They also use the ELBO in Eq. 2 but do not drop the edge effect term and work with:

$$K = -\sum_{t=2}^{T} \mathbb{E}_q \mathrm{D}_{\mathrm{KL}} \left(q(x^{(t-1)}|x^{(t)}, x^{(0)}) | p(x^{(t-1)}|x^{(t)}) \right) + H_q(x^{(T)}|x^{(0)}) - H_q(x^{(1)}|x^{(0)}) - H\left(q(x^{(T)}), p(x^{(T)})\right) + \mathbb{E}_q \log \left[\frac{p(x^{(0)}|x^{(1)})}{q(x^{(1)}|x^{(0)})} \right]$$

or

$$K = -\sum_{t=2}^{T} \mathbb{E}_q \mathrm{D}_{\mathrm{KL}} \left(q(x^{(t-1)}|x^{(t)}, x^{(0)}) | p(x^{(t-1)}|x^{(t)}) \right) + \mathbb{E}_q \log p(x^{(0)}|x^{(1)}) + H_q(x^{(T)}|x^{(0)}) - H_q(x^{(1)}|x^{(0)}) - H\left(q(x^{(T)}), p(x^{(T)})\right) - \mathbb{E}_q \log q(x^{(1)}|x^{(0)})$$

Since: *i)* $H_q(x^{(1)}|x^{(0)}) = -\mathbb{E}_q \log q(x^{(1)}|x^{(0)})$ and *ii)* $H_q(x^{(T)}|x^{(0)}) - H\left(q(x^{(T)}), p(x^{(T)})\right) = -\mathrm{D}_{\mathrm{KL}}\left(q(x^{(T)}|x^{(0)}) | p(x^{(T)})\right)$ the above simplifies to:

$$K = -\mathbb{E}_q \sum_{t=2}^{T} \underbrace{\mathrm{D}_{\mathrm{KL}} \left(q(x^{(t-1)}|x^{(t)}, x^{(0)}) | p(x^{(t-1)}|x^{(t)}) \right)}_{L_{t-1}} + \mathbb{E}_q \underbrace{\log p(x^{(0)}|x^{(1)})}_{L_0} - \mathbb{E}_q \underbrace{\mathrm{D}_{\mathrm{KL}} \left(q(x^{(T)}|x^{(0)}) | p(x^{(T)}) \right)}_{L_T}$$

2.4 MODEL SETUP

Reverse process has same functional form and we parameterize it as

$$p(x_{t-1}|x_t) = \mathcal{N}(x_t; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{4}$$

1. fixed noise schedule: linearly increasing constants from $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$
2. fixed reverse process covariance matrices $\Sigma_\theta(x_t, t) = \sigma_t^2 \mathbb{I}$

2.5 REPARAMETERIZATION

Look at $L_{t-1} = \mathrm{D}_{\mathrm{KL}}\left(q(x^{(t-1)}|x^{(t)}, x^{(0)}) | p(x^{(t-1)}|x^{(t)})\right)$:

$$L_{t-1} = -\mathrm{D}_{\mathrm{KL}} \left(q(x^{(t-1)}|x^{(t)}, x^{(0)}) | p(x^{(t-1)}|x^{(t)}) \right) = \mathbb{E}_q \left[\frac{1}{2\sigma_t^2} ||\tilde{\mu}_t(x_t, x_0) - \mu_\theta(x_t, t)|| \right] + \mathcal{C}$$

The most straightforward parameterization of μ_θ is a model that predicts $\tilde{\mu}_t$, the forward process posterior mean. We've seen in section 2.2.1

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon \implies x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}} \left(x_t - \sqrt{1 - \bar{\alpha}_t}\epsilon \right)$$

with $\epsilon \propto \mathcal{N}(0, \mathbb{I})$. Using this expression for x_0 and writing out $\tilde{\mu}$ from Eq. 3 we have

$$\mu_\theta(x_t, t) = \tilde{\mu}_t \left(x_t, \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1 - \bar{\alpha}_t}\epsilon_\theta(x_t)) \right) = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(x_t, t) \right) \tag{5}$$

It is more advantageous in practice to parameterize the noise $\epsilon_\theta(x_t, t)$ with a deepnet. The loss term:

$$\begin{aligned} L_{t-1} &= \mathbb{E}_{x_0,\epsilon}\left[\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}||\epsilon - \epsilon_\theta(x_t,t)||^2\right] \\ &= \mathbb{E}_{x_0,\epsilon}\left[\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}||\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)||^2\right] \end{aligned}$$

2.6 Edge Term

All the images x_i for $i \in [1, \ldots, T]$ have pixel values in $[-1, 1]$. Set the last term of the reverse process to an independent discrete decoder.

2.7 Simplified Training Objective

$$L = \mathbb{E}_{x_0,\epsilon,t}\left[||\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)||^2\right]$$

For $t > 1$ this re-weights terms from above, and for $t = 1$ this approximates L_0 by ignoring edge effects and σ_1.

2.8 Experimental details

- $T = 1000$
- forward process variances are constants increasing linearly from $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$
- architecture is U-net with parameters shared across time and self-attention.
- time is specified with sinusoidal position embedding

2.9 Experimental Results

Loss function ablation:

- training on the true variational bound yields better codelengths
- simplified objective yields the best sample quality

Parameterizarion: predicting ϵ performs approximately as well as predicting $\tilde{\mu}$ when trained on the variational bound with fixed variances, but much better when trained with simplified objective.

Learning reverse process variances leads to unstable training and poorer sample quality.

REFERENCES

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. URL https://arxiv.org/abs/2006.11239.

Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *CoRR*, abs/1503.03585, 2015. URL http://arxiv.org/abs/1503.03585.