CUB Deep Learning, Fall 2024

Notes for Inverse Autoregressive Flow

Suppose we have trained some autoregressive generation model (teacher model):

$$p_T(x_1, \dots, x_T) = \prod_{t=1}^{T} p(x_t | x_{<t}).$$

Example generation models include GPT for texts, PixelCNN for images, WaveNet for audio, etc.

In this model it is fast to compute probability $p_T(x_1, \dots, x_T)$ if all x_t are known. But the inference is performed slow because of autoregressive sequential generation of next element x_t conditioned on all previous elements $x_1, \dots, x_{t-1}$.

Let's consider the following flow model for generating $x_1, \dots, x_T$ from noise $z_1, \dots, z_T$:

1. Initialize $z_1, \dots, z_T \sim p(z) = \mathcal{N}(z|0, 1)$.
2. $x_t^0 := z_t \ \forall t$.
3. For $k = 1, 2, \dots, K$: $x_t^k = \sigma_t^k(x_{<t}^{k-1}, \theta) x_t^{k-1} + \mu_t^k(x_{<t}^{k-1}, \theta) \ \forall t$.
4. Output $x_t = x_t^K \ \forall t$.

For every step k here we have a linear transformation of the form $x^k = \sigma^k(x^{k-1}, \theta) \odot x^{k-1} + \mu^k(x^{k-1}, \theta)$, where vector functions μ^k and σ^k have the required masked property. Superposition of several linear transformation is still a linear transformation, so

$$x = x^K = \sigma(z, \theta) \odot z + \mu(z, \theta).$$

Here $\sigma(z, \theta) = \sigma^1(x^0, \theta) \odot \sigma^2(x^1, \theta) \odot \dots \odot \sigma^K(x^{K-1}, \theta)$ and $\mu(z, \theta)$ is computed correspondingly from all σ^k and μ^k.

It is possible to compute the probability for the introduced flow (student model):

$$\log p_S(x_1, \dots, x_T | \theta) = \log p(z_1, \dots, z_T) - \sum_{t=1}^{T} \log \sigma(z_{<t}, \theta) = \sum_{t=1}^{T} \log \mathcal{N}(z_t|0, 1) - \sum_{t=1}^{T} \log \sigma(z_{<t}, \theta).$$

In the introduced flow it is fast to make inference – just sample noise z and make forward propagation $x = \sigma(z, \theta) \odot z + \mu(z, \theta)$. But computing probability $\log p_S(x)$ for arbitrary x is slow because of sequential finding of elements z_t given all the previous ones $z_1, \dots, z_{t-1}$ by the following formula:

$$z_t = \frac{x_t - \mu_t(z_{<t}, \theta)}{\sigma_t(z_{<t}, \theta)}.$$

Now our goal is to train parameters θ in a way to align distributions from teacher and student models. For this we are going to minimize KL divergence between two distributions:

$$L(\theta) = \mathrm{KL}(p_S(x|\theta) \,\|\, p_T(x)) \to \min_\theta.$$

Hopefully, this minimization can be done without necessity to compute slow inverted flow from x to z. Let's expand the introduced function:

$$L(\theta) = \mathbb{E}_{p_S(x|\theta)} \log p_S(x|\theta) - \mathbb{E}_{p_S(x|\theta)} \log p_T(x) = \mathbb{E}_{p_S(x|\theta)} \left[\sum_{t=1}^{T} \log \mathcal{N}(z_t|0, 1) - \sum_{t=1}^{T} \log \sigma(z_{<t}, \theta) \right] -$$

$$- \mathbb{E}_{p_S(x|\theta)} \sum_{t=1}^{T} \log p_T(x_t | x_{<t}) = \underbrace{\sum_{t=1}^{T} \mathbb{E}_{p(z_t)} \log \mathcal{N}(z_t|0, 1)}_{\text{const}} - \sum_{t=1}^{T} \mathbb{E}_{p(z_t)} \log \sigma(z_{<t}, \theta) -$$

$$- \sum_{t=1}^{T} \mathbb{E}_{p(z)} \log p_T(\underbrace{\sigma(z_{<t}, \theta) \odot z_t + \mu(z_{<t}, \theta)}_{x_t} \,|\, \underbrace{\{\sigma(z_{<s}, \theta) \odot z_s + \mu(z_{<s}, \theta)\}_{s=1}^{t-1}}_{x_1, \dots, x_{t-1}}).$$

Finally we can minimize $L(\theta)$ by sampling mini-batches of sequences z and computing gradient $\nabla_\theta L(\theta)$ by replacing expectations $\mathbb{E}_{p(z)}$ with averages over samples.