Diffusion Models (Guided Diffusion)

Dr. Alireza Aghamohammadi

Guided Diffusion

- So far, we have examined diffusion models as a method for modeling the unconditional data distribution $P_r(x)$, which is estimated from a set of training samples $\{x_i\}$ drawn independently from $P_r(x)$.
- Once trained, diffusion models allow us to generate new samples from this distribution.
- However, in many applications, we may wish to sample from a conditional distribution $P_r(x \mid c)$, where the conditioning variable c can represent attributes such as a class label or a textual description relevant to the desired sample (e.g., an image's content).
- There are two main approaches for achieving conditional sampling:
  1. **Classifier guidance**
  2. **Classifier-free guidance**

Classifier Guidance

- Assume we have a trained classifier $P_r(c \mid x)$ available.
- This classifier modifies the denoising step from z_t to z_{t-1} to incorporate class information c.
- In the standard denoising update:

$$\mathbf{z}_{t-1} \leftarrow f(\mathbf{z}_t, \phi, t) + \sqrt{\beta_t}\boldsymbol{\epsilon}$$

  we update the latent variable z_{t-1} without any conditional guidance.
- With classifier guidance, however, we use:

$$\mathbf{z}_{t-1} \leftarrow f(\mathbf{z}_t, \phi, t) + \lambda \frac{\partial \log P_r(c \mid \mathbf{z}_t)}{\partial \mathbf{z}_t} + \sqrt{\beta_t}\boldsymbol{\epsilon}$$

  where the second term directs the denoising process towards the direction that maximizes the probability of the target label c under the classifier model.
- The strength of the classifier's influence can be controlled by a hyperparameter λ, known as the **guidance scale**.
- A limitation of this approach is that it requires training a separate classifier that can handle varying levels of noise, whereas standard classifiers are typically trained on clean data.

Classifier-free Guidance

- Classifier-free guidance eliminates the need for a separate classifier $P_r(c \mid z_t)$. Instead, it directly incorporates class information into the main model, represented as $g(z_t, \phi, t, c)$.
- Practically, this is achieved by adding an embedding based on the conditioning variable c into the network's layers.
- The model is trained jointly on both conditional and unconditional objectives by randomly omitting class information c during training.
- This approach enables the model to generate either unconditional or conditional samples at test time, or even a blend of both based on a weighted combination.