Lecture Title and Date

Deep Generative Models - Part II

3/31/2025

Objectives of the Lecture

By the end of this lecture, students should be able to:

1. Explain the concepts of RBM, VAE, GAN
2. Explain the concept of Wasserstein distance and how it can be applied to construct Wasserstein Autoencoder
3. How can Wasserstein Autoencoder be applied to various biological problems
4. Describe Diffusion Probabilistic Models and its applications

Key Concepts and Definitions

- **Restricted Boltzmann Machine (RBM):** An energy-based model with a visible layer (observed data) and a hidden layer (latent features). It learns by assigning low energy to training data and their associated hidden representations. Parameters are trained using maximum likelihood estimation, often via contrastive divergence.
- **Variational Autoencoder (VAE):** A generative model composed of a stochastic encoder and decoder. It minimizes a combination of reconstruction loss and the KL divergence between the latent posterior and a prior distribution (Gaussian).
- **Generative Adversarial Network (GAN):** Consists of a generator and a discriminator trained in a minimax game. The generator tries to produce data that fools the discriminator, while the discriminator tries to distinguish real from fake data. When successful, the generator captures the data distribution.
- **Wasserstein Distance:** Earth Mover's Distance (EMD), it quantifies the minimum cost to transport one distribution to another.
- **Wasserstein Autoencoder:** Combines autoencoder structure with Wasserstein Distance. Model's goal is to minimize reconstruction error and the divergence between the *aggregated* posterior and the prior, rather than pointwise KL divergence as in VAEs. Thus, instead of forcing each data point to fit the target shape, they just make sure that all the compressed versions combined match the target shape.
- **Diffusion Probabilistic Model**: A generative model that learns how to generate data from a process of denoising a dataset that has had gaussian noise added to it iteratively until it is indistinguishable from pure gaussian noise.
- **Diffusion autoencoder:** the model encodes any image into a two-part latent code that captures both semantics and stochastic variations and allows near-exact reconstruction.
- **Equivariant diffusion model:** the model defines a noising process on both atom coordinates and types, and then learns the generative denoising process using an equivariant neural network

Main Content/Topics

Recap on RBM, VAE and GAN:

- RBMs are energy-based models that learn to represent data using two layers (one input layer and one hidden feature layer). The connectivity is restricted to make inference and learning easier.
- VAEs are a type of autoencoder that not only tries to reconstruct the input data accurately but also ensures that the internal representations (called latent variables) follow a smooth, known distribution (usually Gaussian), by adding a regularization term called KL divergence.
- GANs can be distinguished with other models since they consist of two parts: a generator that tries to create realistic data, and a discriminator that tries to tell real from fake. However, GANs can be tricky to train because they may suffer from issues like instability and vanishing gradients, which make learning difficult.

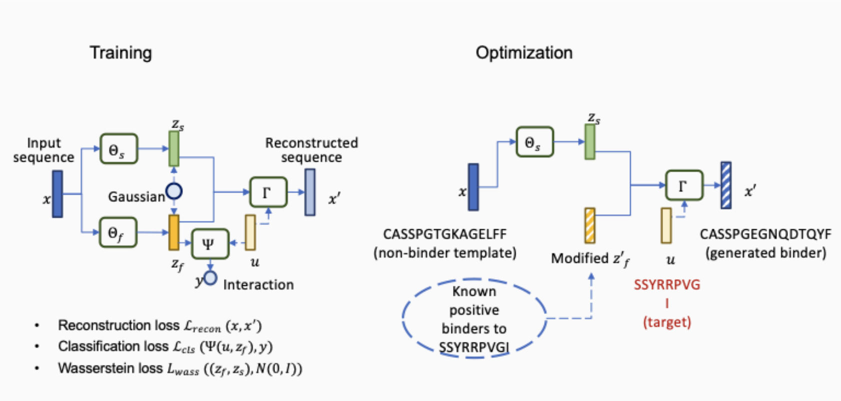
Wasserstein Distance definition: it is a way to measure how different two groups of data are. Imagine you have two piles of dirt shaped like two different data distributions, and you want to turn one pile into the other. WD can be interpreted as the minimum cost of changing a certain probability distribution shape into a certain different probability distribution shape. In this case, the cost is quantified by multiplying the amount of soil moved and the distance traveled. Even when two distributions are located in lower dimensional manifolds without overlaps, Wasserstein distance can still provide a meaningful and smooth representation of the distance in-between.

Comparing Wasserstein Distance with KLD and JSD: As mentioned above briefly, unlike traditional measures such as KL divergence or Jensen-Shannon divergence (JSD), the Wasserstein distance works well even when the data distributions don't overlap, which is common during the early stages of training generative models. KL and JSD often give zero or undefined gradients in such cases, which makes learning very difficult. In contrast, Wasserstein distance always provides a useful, smooth signal that helps the model learn how to improve, making it a more stable and reliable tool for comparing distributions in generative modeling.

Comparing Wasserstein GAN with GAN: WGAN improves upon the original GAN by using the Wasserstein distance instead of the Jensen-Shannon divergence. This change results in more stable training and better guidance for the generator, especially when the generated data is still far from real data. WGAN avoids common GAN problems like vanishing gradients and mode collapse (when the generator produces very limited types of outputs)

Wasserstein Autoencoder and its advantages: WAE is a generative model that combines the structure of an autoencoder with the principles of Wasserstein distance. Like other autoencoders, it learns to compress and reconstruct data, but unlike VAEs, it matches the overall distribution of the latent variables to a target shape instead of forcing each individual point to match it. This leads to clearer, higher-quality outputs, avoids issues like blurry reconstructions, and allows for more flexible and meaningful representations of data.

- **Properties of WAE:** (1) disentangled representations, meaning each part of the compressed data can capture a different, interpretable feature (like motion, color, or structure). (2) they tend to produce sharper and more realistic outputs compared to VAEs, which often struggle when generating data from sampling overlapping latent spaces. (3) WAEs naturally promote a strong connection between the input and its representation, which helps preserve important details and improve reconstruction quality.
- **How to train WAE:** Training a WAE involves two main goals: reconstructing the input data as accurately as possible, and making sure that the overall shape of the encoded data (the latent space) matches a desired prior distribution, like a normal distribution. (1) a GAN-like approach (WAE-GAN), where a separate network tries to distinguish encoded data from random samples (2) statistical method called Maximum Mean Discrepancy (WAE-MMD), which compares the average behavior of the two distributions. Both approaches help align the encoded data with the prior while maintaining high reconstruction quality.
- **Example application of WAE:** Deconstructing T-cell receptor sequence into functional (binding) and structural sequences. By separating these regions into separate latent representations (z_s and z_f), it enables the user to manipulate these sequences independently. For example, inputting a novel target sequence while holding z_s constant to generate a new T-cell receptor sequence



  - This same principle can be applied to small molecules by applying equivariant graph neural networks that function as encoders and decoders. It can still be used to deconstruct the molecule into structure and function

Diffusion Probabilistic Model The forward diffusion process involves adding noise to the original data in repeated steps until the data is indistinguishable from a gaussian. Each addition of noise takes the form of a closed-form gaussian, therefore the marginal distribution and joint distributions are also gaussians, so there is no learning in the encoding. This gives these models a deterministic quality that allows the reverse process (the decoder) to recover the original data. The model is trained by teaching it how to undo each step when gaussian noise was added to the image. This process also allows backpropagation during training.

- Classifier can be added by adding delta ln(p(y|x)) to the update function

Diffusion autoencoder was designed to empower the diffusion model with semantic representations and therefore tackle the difficulty of controlling latent representations of the diffusion model. The implementation is having 2 parallel encoders: (1) Stochastic encoder: similar to a typical diffusion model, adding noise to the original image recursively to get X^T; (2) Semantic encoder: similar to a typical autoencoder, squeezing the original image into a lower-dimensional latent space (and optionally training classifiers to create modified semantic embeddings). Then, the decoder reconstructs the image from both X^T and semantic embeddings (Figure 1). The semantic encoding process enables the diffusion autoencoder to compositionally manipulate multiple attributes in image generation. For example–a portrait–we can modify the attributes such as how they smile, how young they look, how wavy their hair to change the portrait while keeping them the same person. Based on diffusion autoencoder, the latent diffusion model was developed to enable text-conditioned image-to-video generation. Specifically, the model learns the relationship between texts and optical flows/facial expressions, and then given a new static image and various texts, the model can make the image move like the text describes.

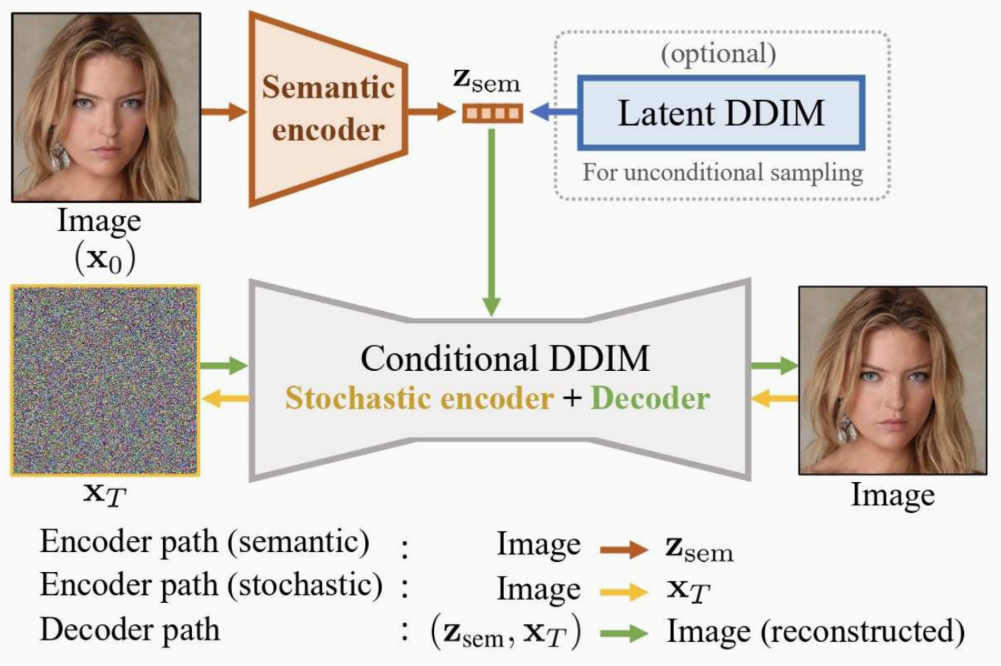


Image
$(\mathbf{x}_0)$
Semantic encoder
$\mathbf{z}_{\text{sem}}$
(optional)
Latent DDIM
For unconditional sampling
Conditional DDIM
Stochastic encoder + Decoder
$\mathbf{x}_T$
Image
Encoder path (semantic) : Image → $\mathbf{z}_{\text{sem}}$
Encoder path (stochastic) : Image → $\mathbf{x}_T$
Decoder path : $(\mathbf{z}_{\text{sem}}, \mathbf{x}_T)$ → Image (reconstructed)

Figure 1 Diffusion Autoencoder Architecture (Preechackul et al., *CVPR*, 2022)

Equivariant diffusion model incorporates equivariant constraints into the denoising process. In other words, while the Wasserstein autoencoder generates molecules by an equivariant graph neural network as the encoder and decoder, the equivariant diffusion model utilizes such networks to learn the Gasussian noise added to the 3D molecules (atom coordinates and types). As exemplified by atom types, the model can be applied to discrete data, where simply adding Gaussian noise wouldn't be effective. One of the solutions is to represent the discrete data as one-hot vectors. After adding Gaussian noise to these vectors, we normalize them and get the most likely categories. Then, in the application of equivariant diffusion models, we can have a semantic encoder to extract the attributes of the molecules and interpolate/manipulate them to generate molecules that have never been seen in experiments or optimize the molecules with new properties (like synthetic accessibility score, dopamine binding score) yet maintaining the same structures.

The fact of limited data for training these generative models presses for a robust experimental design for protein/drug optimization. Given a prior distribution p(x) and applying any model learnt in this lecture, we would want to sample from p(x | y >= desirable value), but this can be challenging because samples satisfying that condition can be rare in the dataset. To tackle this problem, **iterative importance sampling**, based on a reweighting idea, fine tunes the generative model and then samples from it. The weight is the product of 2 components: one is that we require the new distribution not to deviate too much from the prior (the existing data), and the other is that we encourage the new distribution to include desired but rare events. After training the generative model with the existing data, we compute such weights for each sample, use these reweighted samples to fine tune the trained model, and then repeat this process many times so that the model will gradually move towards producing the desired samples.

Discussion/Comments

For diffusion autoencoders, how the semantic encoder is enforced to extract semantic attributes is underexplored. The size of latent space might influence its interpretability, because, theoretically, each dimension would represent an attribute.

For equivariant diffusion models, some technical details are under debate. For example, the current method normalizes the atom coordinates and types by subtracting the center of gravity, but in the future, this might be an unnecessary step.

List all suggested reading here and please answer:

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., & Ganguli, S. (2015, June). **Deep unsupervised learning using nonequilibrium thermodynamics**. In *International conference on machine learning* (pp. 2256-2265). Pmlr. https://proceedings.mlr.press/v37/sohl-dickstein15.pdf
This paper introduces probability diffusion models

Ho, J., Jain, A., & Abbeel, P. (2020). **Denoising diffusion probabilistic models**. *Advances in neural information processing systems*, *33*, 6840-6851.
https://proceedings.neurips.cc/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf
This paper presents a novel method for denoising probabilistic diffusion models

Preechakul et al., Diffusion Autoencoders: Toward a Meaningful and Decodable Representation. CVPR 2022.
This paper introduced the concept and algorithm of the diffusion autoencoders.

Hoogeboom, Emiel, et al., Equivariant diffusion for molecule generation in 3D. PMLR 2022.
Hoogeboom, Emiel, et al., Argmax flows and multinomial diffusion: Learning categorical distributions. NeurIPS 2021.
These 2 papers demonstrate how the equivariant diffusion model can be applied to discrete data.

Brookes, Park, Listgarten et al., Conditioning by adaptive sampling for robust design. PMLR 2019.
This paper introduced the algorithm of interactive importance sampling.

References ISL/ESL (if any)

ISL does not reach this level of depth in its chapter on Deep Learning

ESL:

- 17.4.4 Restricted Boltzmann Machines: pg 643

Other Suggest references for many of the key concepts

Weng, L. (2017, August 20). From GAN to WGAN.
https://lilianweng.github.io/posts/2017-08-20-gan/
Highly recommended for deeper understanding of Wasserstein distance applied to GAN
Rombach, Blattmann, et al. Align your latents: High-Resolution Video Synthesis with Latent Diffusion Models. CVPR 2023.
A complementary for more information about the image-to-video generation with the latent diffusion models, which wasn’t elaborated during the lecture.