

25-1 DSL 정규 세션



Generative Model 과제

- ☑ 본 과제는 학회 정규 세션 「Generative Model」의 내용을 다루며, 개념의 적용과 실제 활용 사례에 대한 이해를 돕기 위해 기획되었습니다. 해당 과제는 평가를 위한 것이 아니므로, 주어진 힌트(💡)를 적극 활용하시고 학회원 간 토론 및 Slack 질의응답을 적극 활용하여 해결해주시요. **단, 답안 표절이나 LLM의 남용은 금지합니다.**
- ☑ 서술형 문제는 🗨️, 코딩 문제는 ■으로 표기가 되어 있습니다. 각 문제에서 요구하는 방법에 맞게 해결하며, 서술형 문제들은 따로 작성하시어 .pdf 파일로, 코딩 문제들은 주어진 .ipynb 파일에 답안을 작성하여 제출해 주십시오.
(해당 과제의 경우 학기 중 진행되는 점을 고려, 이론 문제 (문제 1) 풀이는 개인 선택이니 참고바랍니다!)
- ☑ **4/17 (목) 23 시 59 분까지** Github 에 .pdf 파일과 .ipynb 파일을 압축하여 **하나의 .zip 파일로 묶어** 제출해 주십시오.
반드시 **data 나 사용한 모델 관련한 파일들을 제외한** 두 파일 만을 압축해 제출해 주십시오!
Github 에 제출하는 방법을 모른다면 학술부장 혹은 과제 질의응답을 위한 오픈채팅방을 적극적으로 활용해 주십시오.

문제 0 Session Review

0 🗨️: Intro to GM / AE&VAE / GAN / Diffusion 세션에서 배운 내용을 자유롭게 요약 및 정리하세요.

💡 어떠한 형태로든 요약해도 괜찮습니다. 다만 과제의 가장 큰 목적은 세션 내용의 복습이고, 제출하신 과제를 다 살펴볼지 못하더라도 해당 부분은 반드시 확인할 예정이므로 어느 정도 노력을 들여 정리해주시길 바랍니다 :)

DSL Generative 요약

Introduction to GM

1. Introduction

- Discriminative models

입력 데이터 x 에 대해, 알맞은 레이블 y 를 예측하는 결정 경계(Decision Boundary)를 학습

→ 모든 입력 데이터에 대해 label distribution을 주어야 함.

데이터의 실제 구조나 분포를 이해하는 것이 어렵다

- Generative model

데이터 x 의 확률 분포를 학습

- Explicit Density Models: $p_{\theta}(x)$ 를 명시적으로 정의
- Explicit Tractable Density Models: $p_{\theta}(x)$ 를 명시적으로 정의 + 계산
- Explicit Approximate Density Models: $p_{\theta}(x)$ 를 명시적으로 정의하나 직접 계산하지 못하고 근사를 계산
- Implicit Density Models: $p_{\theta}(x)$ 를 명시적으로 정의하지 않으며, 다른 방법을 통해 샘플링 및 학습 진행

2. Autoregressive Models

- $p(x)$ 명시적으로 정의 및 계산 방법: $p(x)$ 에 Chain Rule 적용
- PixelRNN
- PixelCNN

3. Sampling Methods

- $p_{\theta}(x)$ 의 근사를 구하는 방법 → 직접 다루기보다 Latent Variable z 를 반영해 다루는 경우가 많다
- 관측된 데이터를 보고, 가장 적절한 z 를 고른다

- Inverse Transform Sampling
 - 원하는 분포에서 어떻게 Sampling? → 분포의 inverse CDF 이용
- Rejection Sampling
 - 샘플링하기 쉬운 분포를 선택한다
- Importance Sampling

4. MCMC

- 위 Sampling 기법이 제한적임
- MCMC → 고차원에서 이전 Sampling Method 보다 효율적
- Markov Chain
 - Stochastic Process → 확률변수들을 나열해 놓은 것
 - 현재 상태만으로 미래 상태의 확률이 결정된다.
 - MCMC가 되기 위한 Markov Chain의 조건
 1. Irreducibility
 2. Aperiodicity
 3. Time-Reversible
- MCMC: Markov Chain + Monte Carlo
- MCMC Algorithms
 - Metropolis-Hastings
 - Gibbs Sampler

5. Variational Inference

- Variational Inference: Optimization을 통해 $p(z|x)$ 의 근사를 계산
 - Idea: Intractable 한 $p(z|x)$ 에 대해 Tractable한 $q(z|x)$ 을 계산하고, 두 분포의 거리 (KL divergence)를 최소화하는 방향으로 최적화 진행

- ELBO: Evidence of Lower Bound

AE&VAE

1. Latent Variable Models

- 단점
 1. 순차 생성으로 속도가 느림
 2. Unsupervised way로 feature들을 학습하지 못함
- Goal
 - 생성, 밀도 추정, 표현 학습
- Shallow Latent Variable Model
- Deep Latent Variable Model

2. AutoEncoder

- Structure: Representation learning + 신경망
- Utilization
 1. Feature 추출
 2. Denoising Autoencoder
 3. Outlier Detection

3. Variational AutoEncoder

- ELBO
 - 기존의 MLE를 사용하면 수치적으로 해석이 어려워지므로 ELBO 사용
- Good ELBO를 찾으면 → θ 를 optimize 할 수 있다.

4. Learning

- Reinforce gradient trick
- Reparameterization trick
- Inference Amortization

GAN

1. GAN

- Likelihood-free Learning

→ 실제 데이터의 분포를 직접 모델링하지 않음

Generated Data의 Quality를 기준으로 생성 모델의 알고리즘을 학습

- 데이터를 생성하는 단계에 집중하여 학습을 진행하는 likelihood free learning에 기반

- Generative Adversarial Networks

Generator가 Adversary와의 경쟁으로 실제 데이터의 분포를 학습하여 새로운 데이터를 생성

→ Adversary: Input data가 실제 데이터인지, 생성된 데이터인지를 판별하는 discriminative model

- GAN's Global Optimality

1. GAN 모델이 설계한 알고리즘에 의해 Global Optimum으로 $p_G = p_{data}$ 를 갖는가?

2. GAN 모델이 설계한 학습 알고리즘이 해당 최적화 문제를 수렴시키는가?

- Convergence of Algorithm

- GAN의 목적함수가 Generator가 생성하는 분포에 대해 Convex 하여 수렴 가능

→ G가 MLP로 정의되면 여러 개의 Critical Point가 존재 가능하여 수렴 보장

- Challenges of GAN

1. Unstable Optimization

2. Mode Collapse

3. Vanishing Gradients

- GAN Evaluation

Inception Scores

1. Sharpness

2. Diversit

Frechet Inception Distance(FID)

Kernel Inception Distance(KID)

- DCGAN

- Vanilla GAN 구조에서 fc Layer을 convolutional Layer로 바꿈

- Unconditional GAN; WGAN

- 기존의 Discriminator 대신 새로운 판별자 Critic 도입

- StyleGAN

Diffusion

- Diffusion

Noise에서 Original Image에 가까워지는 과정도 학습 가능?

- CFG

Classifier Guidance

- Diffusion에서는 적절한 Condition을 주는 것이 어렵다 → 별도의 Classifier을 학습하여 class 정보를 diffusion 모델에 적절히 condition으로 넣어줄 수 있도록

→ Sampling 할 때 Classifier의 gradient 만으로도 Conditional Sampling이 가능하다

- Latent Diffusion Model

이미지를 두 단계로 학습

1. Perceptual Compression

2. Semantic Compression

문제 1 ELBO for Conditional VAE (Optional)

<Introduction to Generative Model>과 <AE & VAE> 세션에서 우리는 VAE 가 사용하는 중요한 원리 중 하나인 Evidence of Lower Bound, 줄여서 ELBO 라고 부르는 형태를 살펴보았습니다. 우리가 원하는 데이터의 분포를 구하는 행위, 즉 데이터의 likelihood 를 최대화하기 위한 행위는 ELBO 를 최대화하는 방식으로 근사할 수 있다는 것입니다. 이러한 ELBO 는 VAE 에 라벨 정보를 추가로 활용하는 CVAE 에도 동일하게 적용되는데요. 여기서는 그 내용과 관련된 두 가지 사실을 이론적으로 증명해보고자 합니다.

1-1 📌: Conditional VAE 의 variational lower bound 를 유도하세요. 다시 말해,

$$\begin{aligned} \log p_{\theta}(\mathbf{x}|\mathbf{y}) &\geq \mathcal{L}(\theta, \phi; \mathbf{x}, \mathbf{y}) \\ &= \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x}, \mathbf{y})}[\log p_{\theta}(\mathbf{x}|\mathbf{z}, \mathbf{y})] - D_{KL}(q_{\phi}(\mathbf{z}|\mathbf{x}, \mathbf{y})||p_{\theta}(\mathbf{z}|\mathbf{y})) \end{aligned}$$

where \mathbf{x} is a vector of dimension d , \mathbf{y} is a one-hot vector of dimension c defining a class, \mathbf{z} is a vector of dimension m sampled from the posterior distribution $q_{\phi}(\mathbf{z}|\mathbf{x}, \mathbf{y})$. The posterior distribution is modeled by a neural network of parameters ϕ . The generative distribution $p_{\theta}(\mathbf{x}|\mathbf{y})$ is modeled by another neural network of parameters θ . Similar to the vanilla VAE that we learned in the class, we assume the conditional independence on the components of \mathbf{z} : *i.e.*, $q_{\phi}(\mathbf{z}|\mathbf{x}, \mathbf{y}) = \prod_{j=1}^m q_{\phi}(z_j|\mathbf{x}, \mathbf{y})$, and $p_{\theta}(\mathbf{z}|\mathbf{y}) = \prod_{j=1}^m p_{\theta}(z_j|\mathbf{y})$.

- 💡 \mathbf{y} 라는 label 변수가 추가된 것뿐이지, 어쩌면 vanilla VAE 와 증명 과정이 거의 비슷할 지 모릅니다!
- 💡 VAE 의 증명 과정은 관련 세션들의 자료를 참고하세요.

1-2 🗡️: 두 분포 p 와 q 를 Gaussian Distribution 이라 가정했을 때, 두 분포의 KL Divergence 과 다음과 같음을 보이세요. (단, $p_{\theta}(\mathbf{z}|\mathbf{y}) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$)

$$D_{KL}(q_{\phi}(\mathbf{z}|\mathbf{x}, \mathbf{y}) \| p_{\theta}(\mathbf{z}|\mathbf{y})) = -\frac{1}{2} \sum_{j=1}^m (1 + \log(\sigma_j^2) - \mu_j^2 - \sigma_j^2),$$

where μ_j and σ_j are the outputs of the neural network that estimates the parameters of the posterior distribution $q_{\phi}(\mathbf{z}|\mathbf{x}, \mathbf{y})$.

💡 해당 등식을 증명하는 과정에서 다음과 같은 trace trick 이 필요하며, 기댓값 연산과 trace 연산은 교환 가능합니다.

$$\mathbf{x}^T \mathbf{A} \mathbf{x} = \text{tr}[\mathbf{x}^T \mathbf{A} \mathbf{x}] = \text{tr}[\mathbf{x} \mathbf{x}^T \mathbf{A}]$$

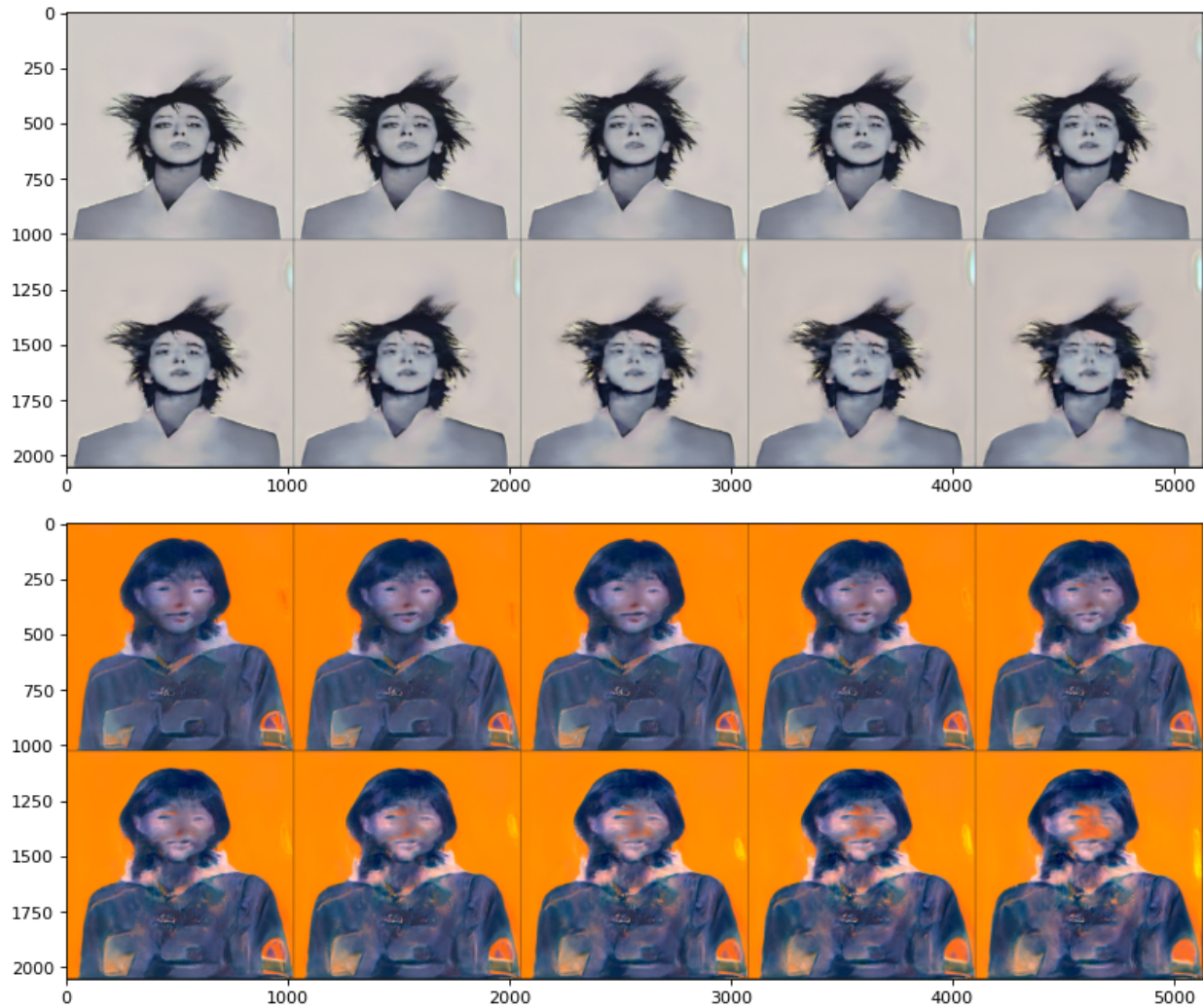
문제 2 | GAN and its variants

GAN 세션은 모두 재밌으셨나요? 이번 문제는 모두 구현 문제로, 이미 코드가 구현이 완료된 두 파일을 실행만 해보시면 되는 간단한!! 문제입니다. 직접 두 파일을 시행하며, GAN의 핵심적인 구조와 StyleGAN을 활용해 얼굴 이미지를 생성하는 과정을 살펴보도록 하죠 :D

2-1 📌, 📌: GAN.ipynb 파일을 직접 실행해 보세요. 해당 결과와 관련 내용을 토대로, GAN에 비해 DCGAN의 성능이 상대적으로 향상된 이유를 간단히 설명하세요.

각각 DCGAN은 Vanilla GAN의 Ac Layer를
Convolution Layer로 바꾼 것이다. (공변, 특징)
이 때문에 DCGAN은 이미지 생성의 변형 (특성, 양상 등)을
더 잘 이해할 수 있기 때문에 DCGAN의 성능이
상대적으로 향상된다.

2-2 ■: StyleGAN.ipynb 파일을 실행하고 나온 결과 (웃는 얼굴, 무표정 얼굴 png 파일)를 여기에 첨부해주세요.



Reference

- 24-2 GAN 과제 (11 기 김현진)
- Statistical Machine Learning, Prof. Lee, Spring 2022, Yonsei Univ.

Data Science Lab

담당자: 12 기 김민규

kimmin01@yonsei.ac.kr

25-1 Generative Model 과제