BERT & Generative Model

Sep 22th 2022/ 7기 김예진

0. 목차

1. After Transformer

- NLP BERT
- I/O
- Pre-trained
- Fine-tuning

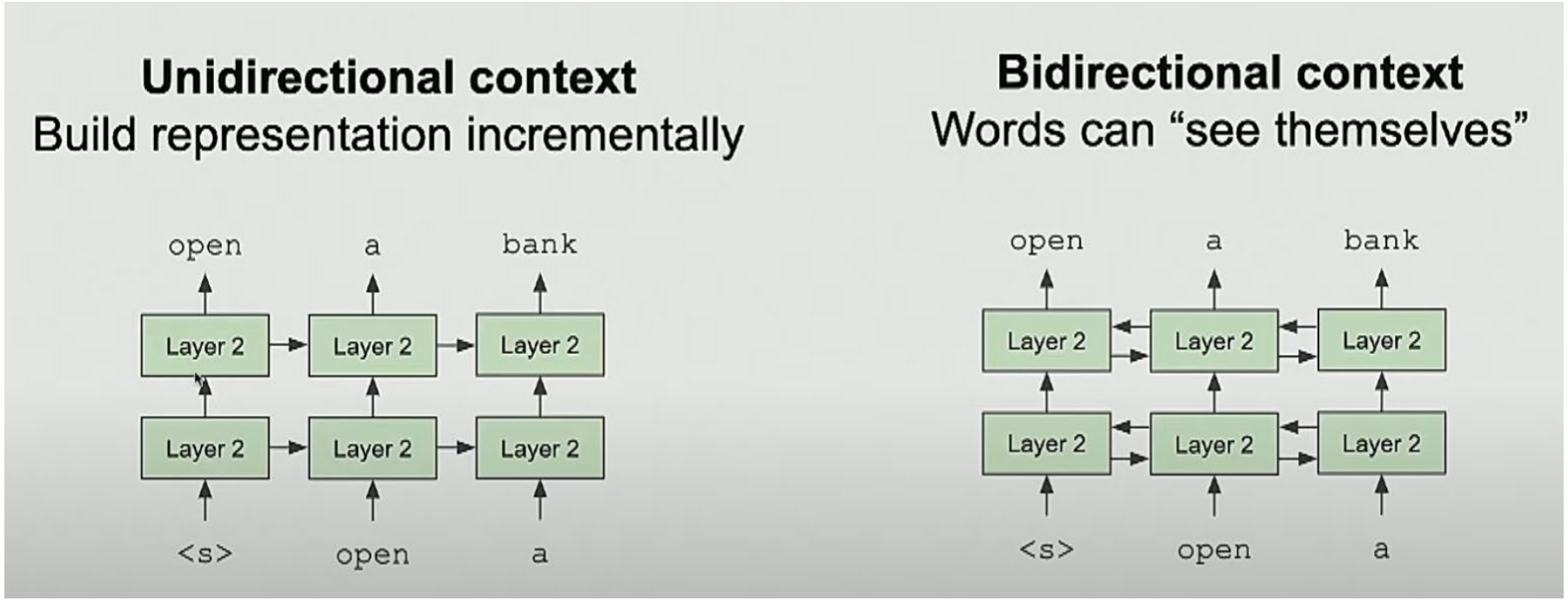
2. Generative Model

- Bayesian
- (Un)Supervised “Learning” - AE
- Generative & Discriminative Model
- Models - VAE

1. After Transformer

NLP - BERT

(sentence level) Bidirectional in RNN



1. After Transformer

NLP - BERT

BERT = Bidirectional Encoder Representation from Transformer!

Key point)

- Bidirectional: 양방향의 정보를 어떻게 학습할까?
- Encoder: 기존 Transformer에서 Encoder

Run)

Task에 무관한 모델 구조를 가진다!

→ Representation도 무관해야 한다!

- pre-trained: Unsupervised
- Fine-tuning: (All param from pre-trained)

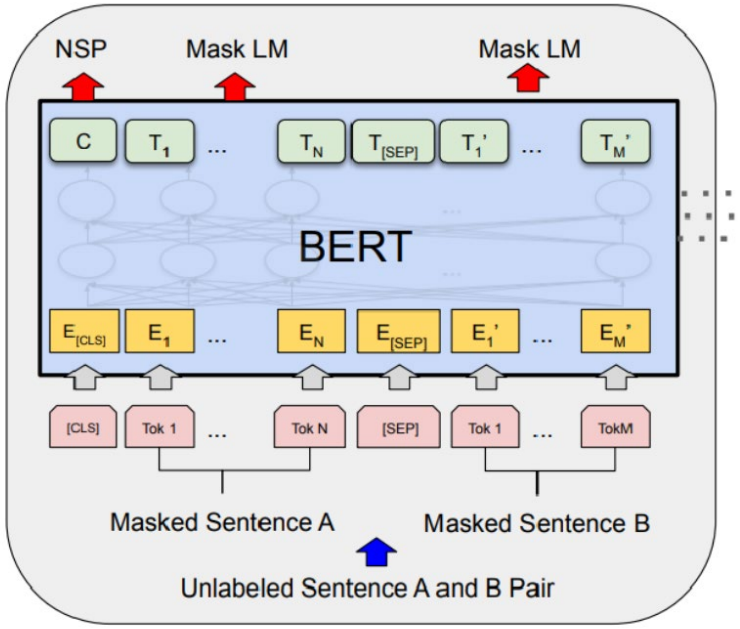
Supervised depending on task

유의사항)

- Sentence: 연속된 단어의 나열, sen
- Sequence: 하나 혹은 두 sentence, seq

1. After Transformer

NLP - BERT



Pre-training

Input = Masked Sen A & Masked Sen B

Output(Task): NSP와 Masked LM

NSP = Next Sentence Prediction

Sentence A와 Sentence B가 연속된 문장인지 분류

→ 맥락을 이해하는 학습 = Encoder?!

Masked LM: Mask된 내용물 예측

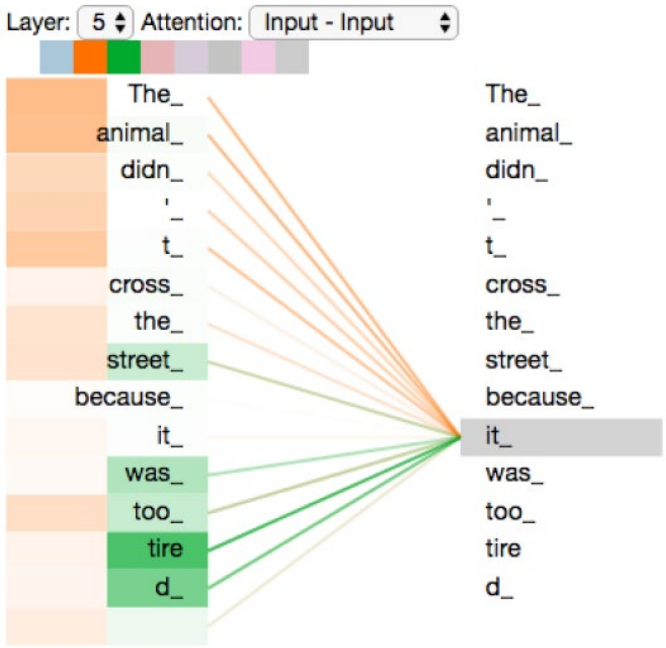
일부 토큰이 뭔지 모를 때 그 토큰이 뭔지 예측

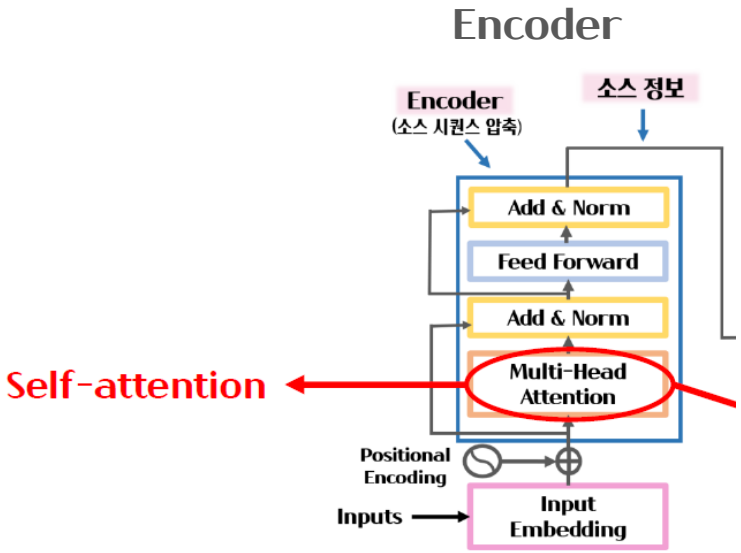
→ 앞뒤의 토큰을 보고 어떤 토큰인지 보기에 Bidirectional!

1. After Transformer

NLP - BERT

Bidirectionality & Transformer.Encoder





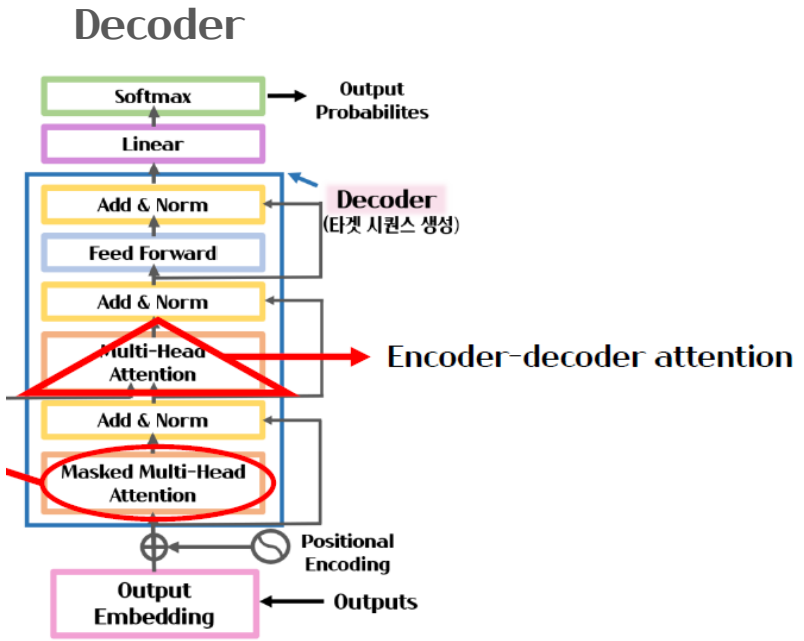
Self-attention 또한 일종의 Transformer에서 Bidirectionality을 녹여낸 부분이긴 함

1. After Transformer

NLP - BERT

Bidirectionality & Transformer.Encoder

	Query	Key
	〈s〉	어제
	I	카페
masking	went	갔었어
masking	to	거기
masking	the	사람
masking	café	많더라
masking	…	

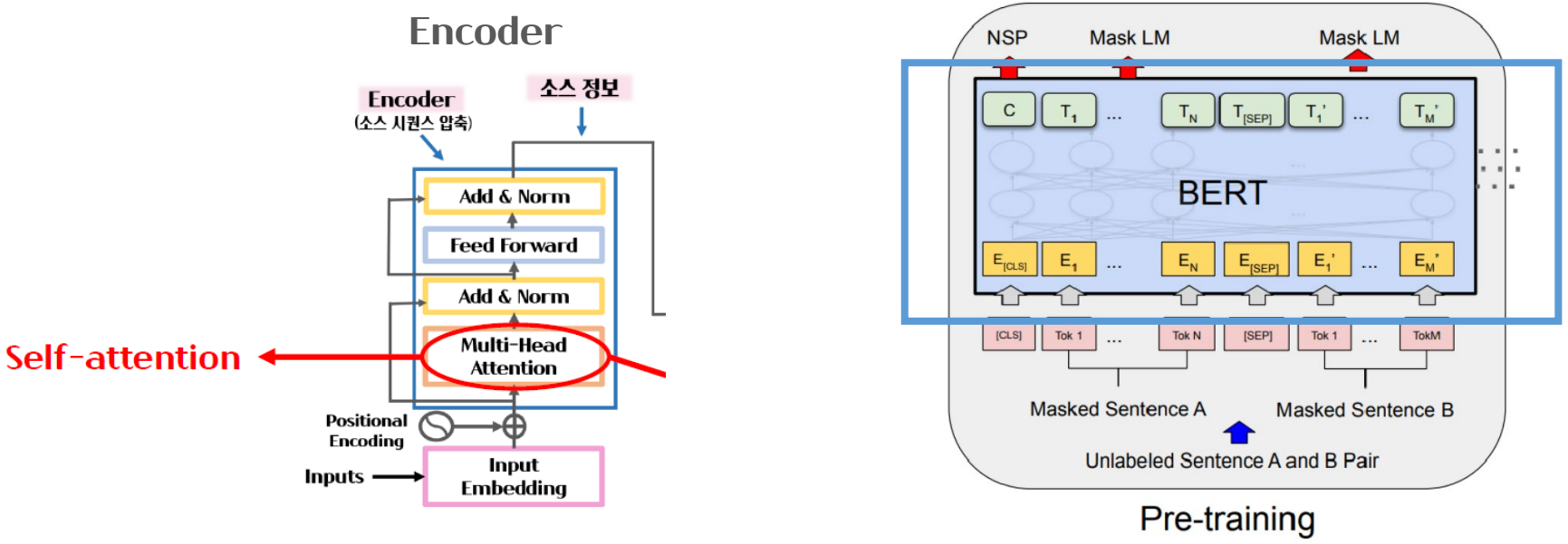


다만 Decoder에서 하는 Task는 결국 순서대로 들어오는 방식으로 진행…!

1. After Transformer

NLP - BERT

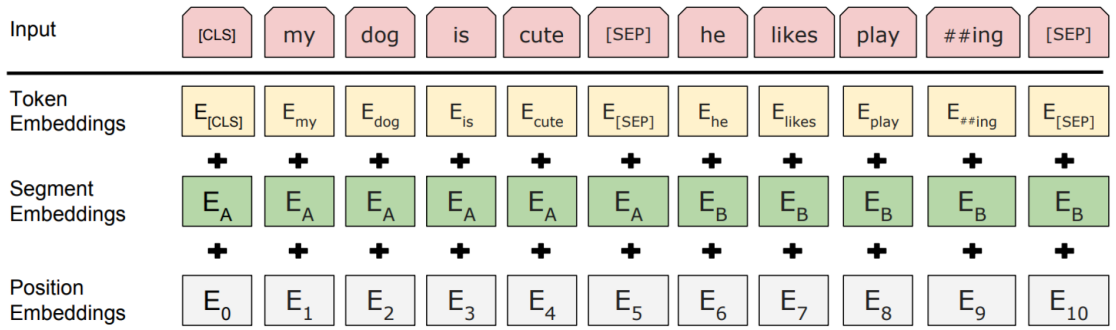
Bidirectionality & Transformer.Encoder



그래서 Transformer의 Encoder 구조만을 이용해서 Task(NSP, Masked LM) 수행하고 학습

1. After Transformer

I/O



Input = Token + Segment + Position

Token

- Tokenizer : Word Piece
- CLS: seq의 시작에 위치

  Aggregate sequence representation

  For classification
- SEP: 두 sentence 구분

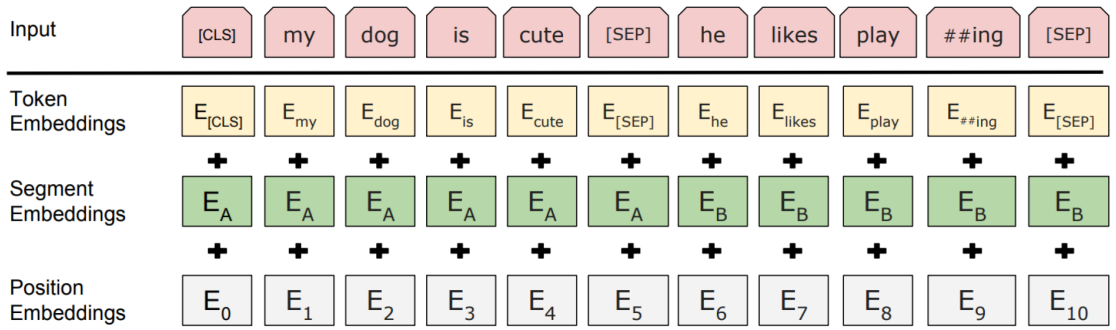
Cf) Word piece tokenizer:

Merge bigram which most increases the likelihood of the data

→ 자주 등장하는 연속된 token 쌍은 하나의 token으로 사용

1. After Transformer

I/O



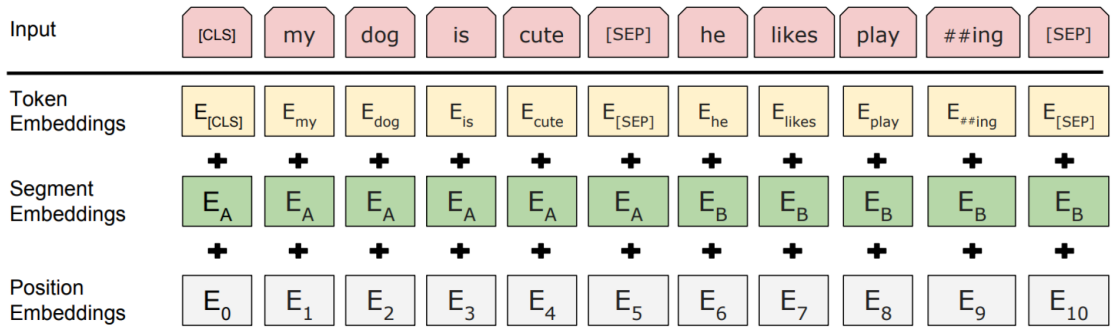
Input = Token + Segment + Position

Segment

- 각 token들이 sen A인지 sen B인지
- CLS와 SEP는 sen A에 포함

1. After Transformer

I/O



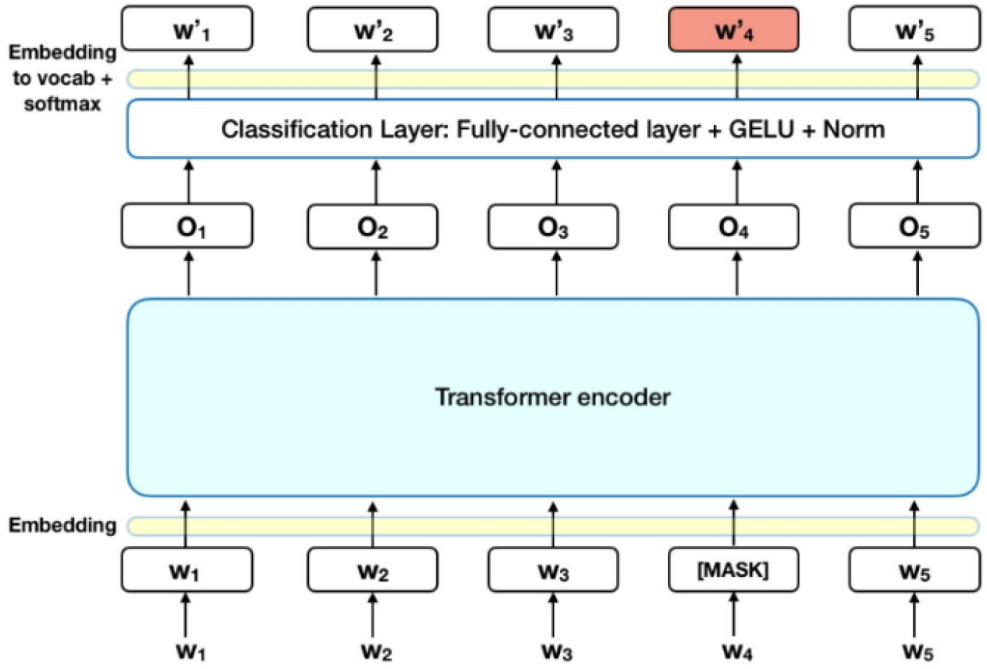
Input = Token + Segment + **Position**

Position

- Transformer에서 사용하던 positional embedding

1. After Transformer

Pre-trained



Output(Task): **Masked LM**와 NSP

Pretrain 과정에서 진행!

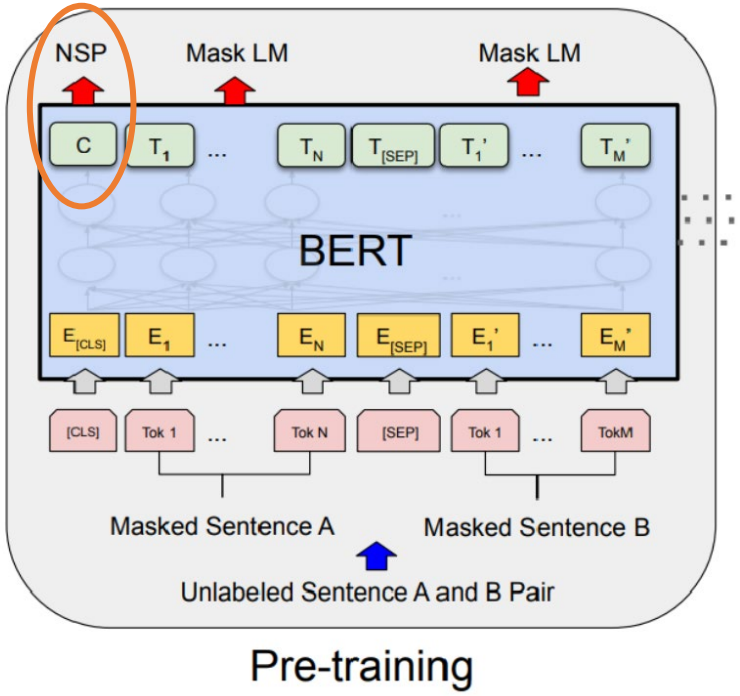
약 15% token"Masking" 그리고 예측

→ 앞뒤의 단어를 보고 학습하기에 **bidirectional**!

그 외에 몇가지 디테일은 Mask LM을 더 찾아보기로

1. After Transformer

Pre-trained



Pre-training

Output(Task): **Masked LM**와 **NSP**

Pretrain 과정에서 진행!

[SEP]로 구분된 두 sen이 연속된 유무 예측

50%는 연속된 것, 50%는 비연속 문장들

→ 맥락을 이해하도록 학습

→ 일부 task들에 대해선 성능 개선!

1. After Transformer

NLP - BERT

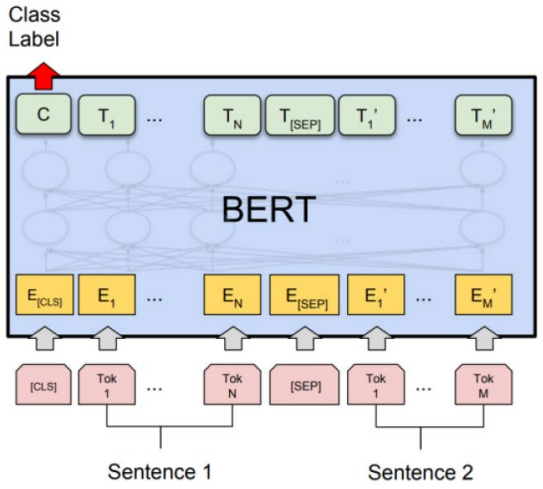
Pre-training

연속된 문장?!

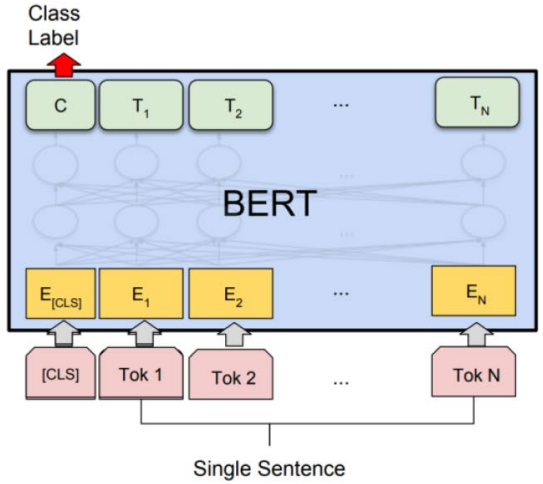
➔ 대본...?!

1. After Transformer

Fine-tuning



(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG



(b) Single Sentence Classification Tasks: SST-2, CoLA

Fine-tuning: 실제로 풀 수 있는 문제들!

→ Classification, QnA, Tagging

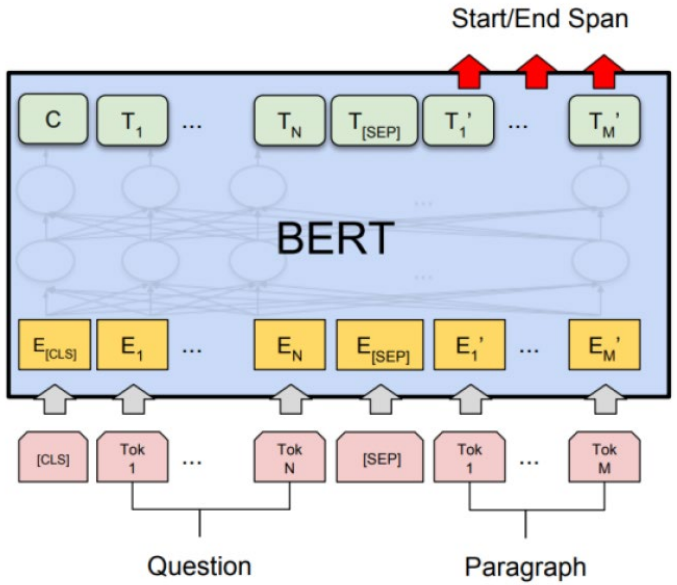
sequence-level

Input이 2문장인 경우: (예) 유사한지 분류

Input이 1문장인 경우: (예) 감성 분류!

1. After Transformer

Fine-tuning



(c) Question Answering Tasks:
SQuAD v1.1

Fine-tuning: 실제로 풀 수 있는 문제들!

→ **Classification**, QnA, Tagging

token-level

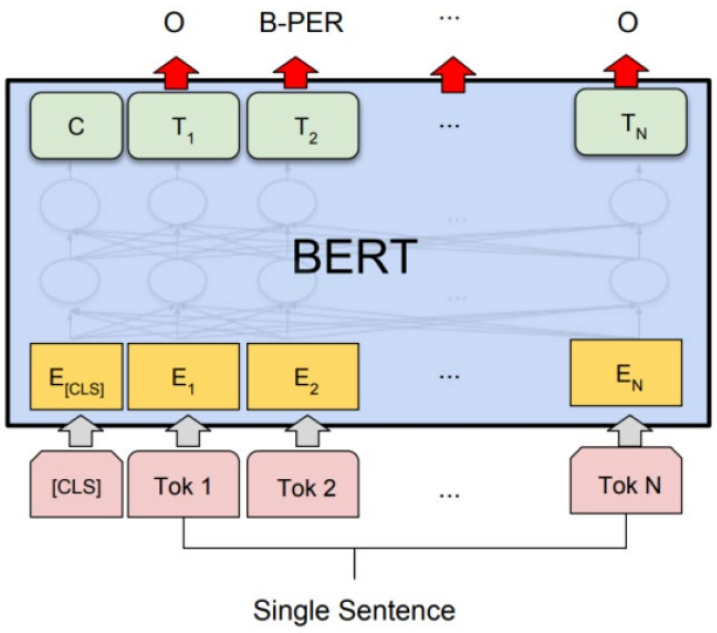
두 sen이 들어오는 경우:

Question과 Paragraph 쌍을 입력했을 때

답안 작성!

1. After Transformer

NLP - BERT



(d) Single Sentence Tagging Tasks:
CoNLL-2003 NER

Fine-tuning: 실제로 풀 수 있는 문제들!

→ Classification, QnA, Tagging

Token-level

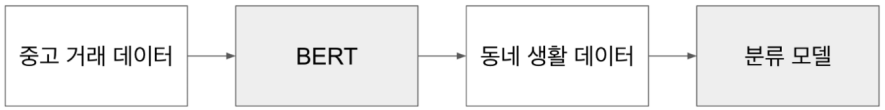
한 sen이 들어오는 경우:

각각의 token마다 응답(ex. 형태소)를 예측

1. After Transformer

NLP - BERT

<BERT 활용 모델>



동네 생활 데이터 통계 (학습 데이터에 사용)

Aa Name	# 데이터 수	분류
동네 관련 이야기	7290	동네생활
중고거래 관련 신고	143	딴 이야기
중고거래 게시글	495	딴 이야기
기타(너무 짧은 글, 업체 홍보 글, 기능 문의 글, 제안, 등등)	395	딴 이야기

당근마켓에서 게시물들에 대한 분류 문제를 해결!

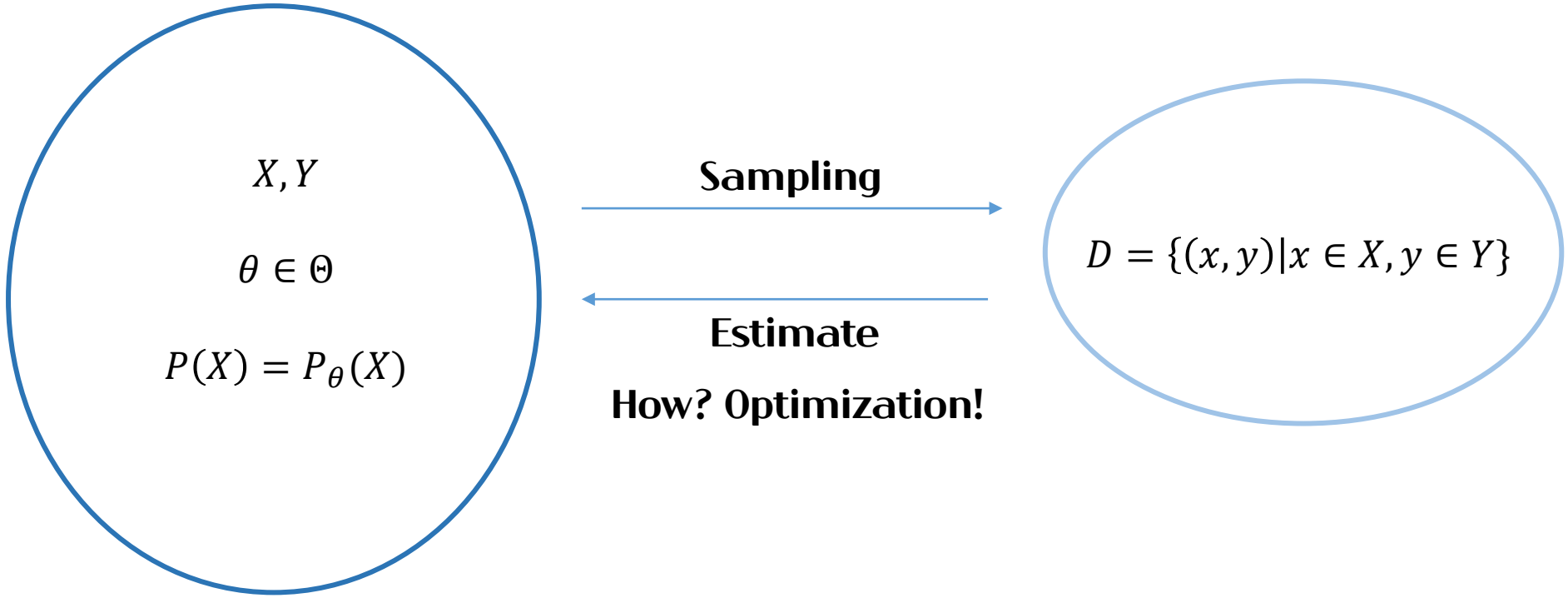
데이터 전처리부터 데이터셋 구축, 모델 구현 등

좋은 구현 예시

https://medium.com/daangn/딥러닝으로-동네생활-게시글-필터링하기-263cfe4bc58d

2. Generative Model

통계학이란

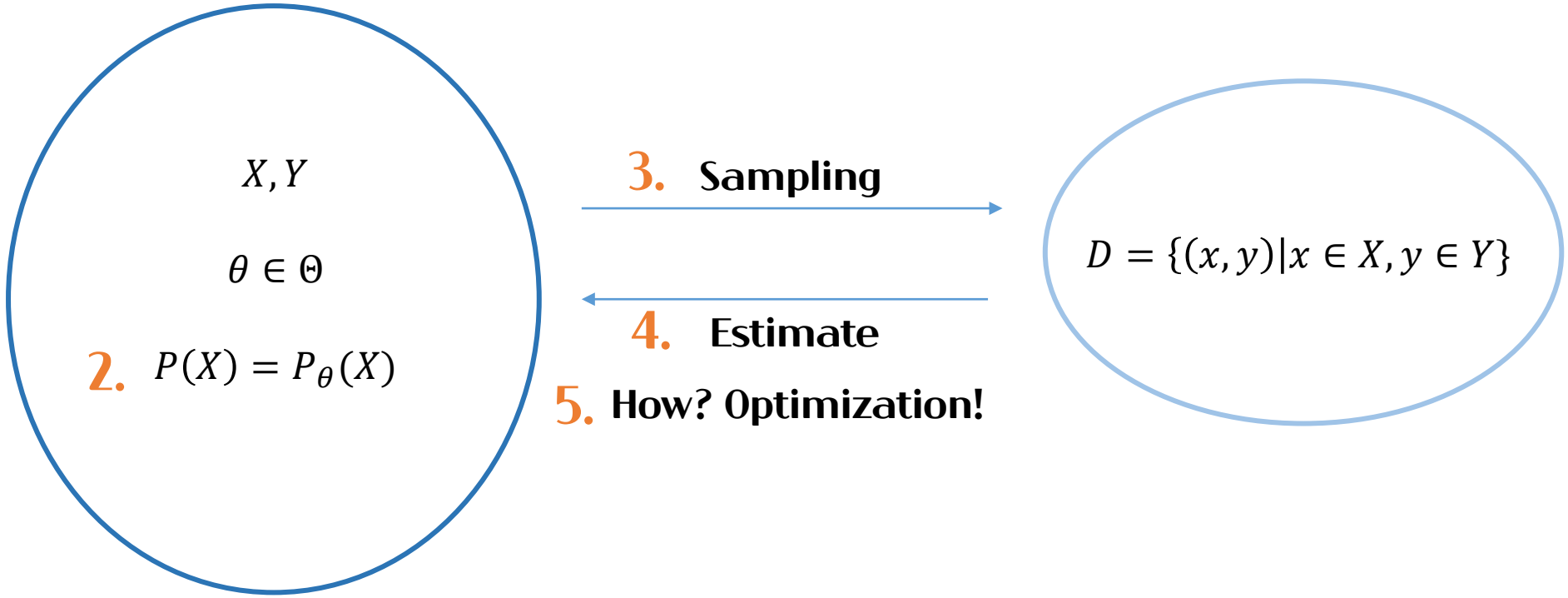


모집단과 관련된 것

표본

2. Generative Model

통계학이란



1. 모집단과 관련된 것

표본

2. Generative Model

Bayesian

Parameter is not a constant value but a "Random Variable"!

$X|\theta \sim f(x|\theta)$: **conditional pdf** of X given $\Theta = \theta$

$\Theta \sim h(\theta)$: **Prior pdf** of Θ ← **표본이 없어서** 과거의 경험 혹은 주관에 따른 "사전 정보"의 역할

X': Random sample from $f(x|\theta)$

x': observation of X'

2. Generative Model

Bayesian

Parameter is not a constant value but a "Random Variable"!

$g_1(x,\theta) = f(x|\theta)h(\theta)$: **Joint pdf of X and Θ**

$g_2(x) = \int g_1(x,\theta)\, d\theta$: **Marginal pdf of X, "Evidence"**

Conditional pdf of Θ given $X' = x'$

"Posterior": $k(\theta|x) = \frac{g_1(x,\theta)}{g_2(x)} = \frac{f(x|\theta)h(\theta)}{g_2(x)}$ ← **표본이 있으니 생각해본 파라미터의 분포?!**

그럼 파라미터는 어떻게 "추정"? Bayes estimate, loss function, risk function…

2. Generative Model

Bayesian

Prior & Posterior

결국 우리가 파라미터에 대해 모르거나, 경험적으로 알고 있는 정보가 있을 때(prior)

데이터를 보고 파라미터의 ~~값~~ 분포를 업데이트 해본다(posterior)!

Conjugate family of distribution

이론적으로, Prior 분포에 따라서 posterior의 분포가 어느정도 결정된다!

	Distribution		
$f(x\|\theta)$	Poisson	Binomial	Normal
$h(\theta)$	Gamma	Beta	Normal
$k(\theta\|x)$	Gamma	Beta	Normal

2. Generative Model

Supervised vs Unsupervised "Learning"

Supervised: Classification, Regression, Object Detection, …

Data: $D = \{(x, y) | x \in X, y \in Y\}$

Model: $y = \mathcal{M}_\theta(x)$

→Mapping!

올바르게 mapping하도록 학습

Whale!

In

Out

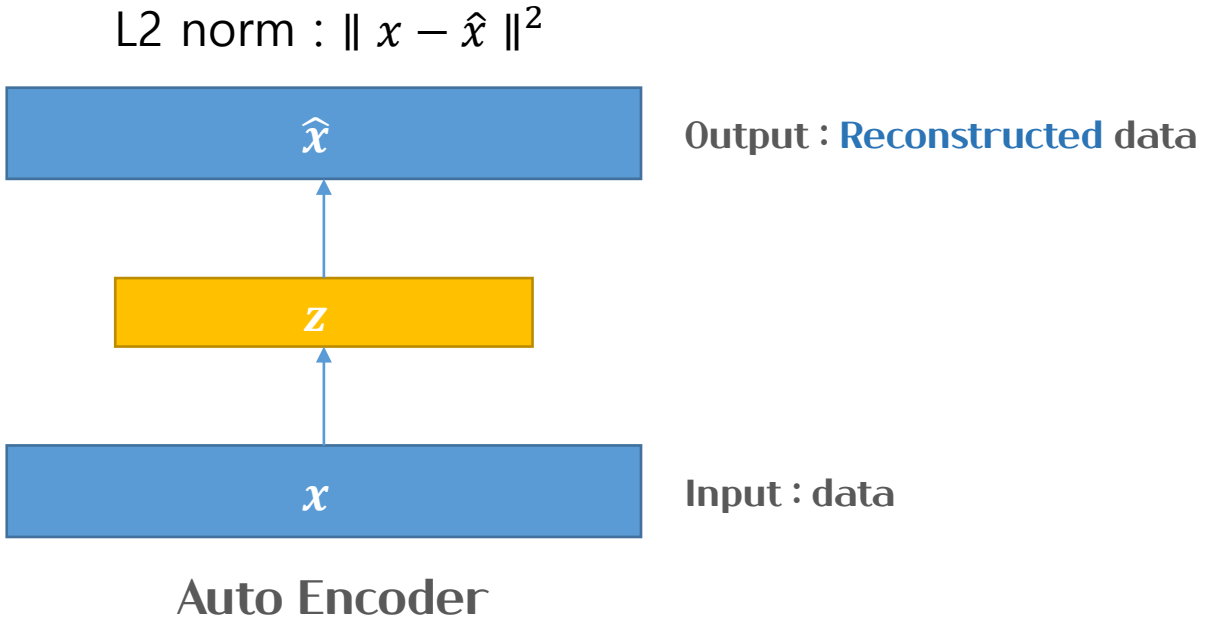
2. Generative Model

Supervised vs Unsupervised "Learning"

Unsupervised: Clustering, Feature learning, Density Estimation …

Data: $D = \{x \mid x \in X\}$

Model: Data structure!



2. Generative Model

Cf) Auto Encoder(AE)

Encoder: 더 찾은 차원의 벡터로 표현하는 방법을 학습!

→ Feature Learning, Dimensionality Reduction

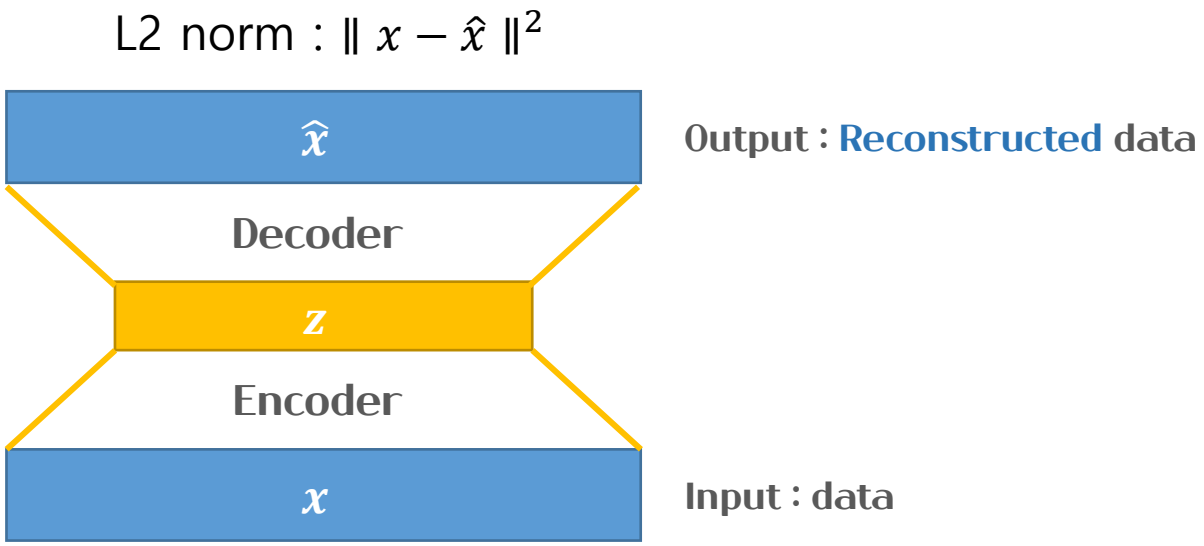
→ How? CNN, FC, …

Decoder: 압축된 정보를 원본으로 복원 생성!

→ ~~Generate…?~~ NO!

→ How? TransposeCNN, FC, …

관련 Task: Semantic segmentation → U-net



Encoder as a feature extractor!

Cf) Auto Encoder(AE)

Encoder can "represent" raw data containing data structure information!

Encoder: 더 작은 차원의 벡터로 표현하는 방법을 학습!

Ex) Feature Extractor or initial value!

→ Feature Learning, Dimensionality Reduction

→ How? CNN, FC, …

CNN + Classifier(Softmax Classifier, SVD etc)

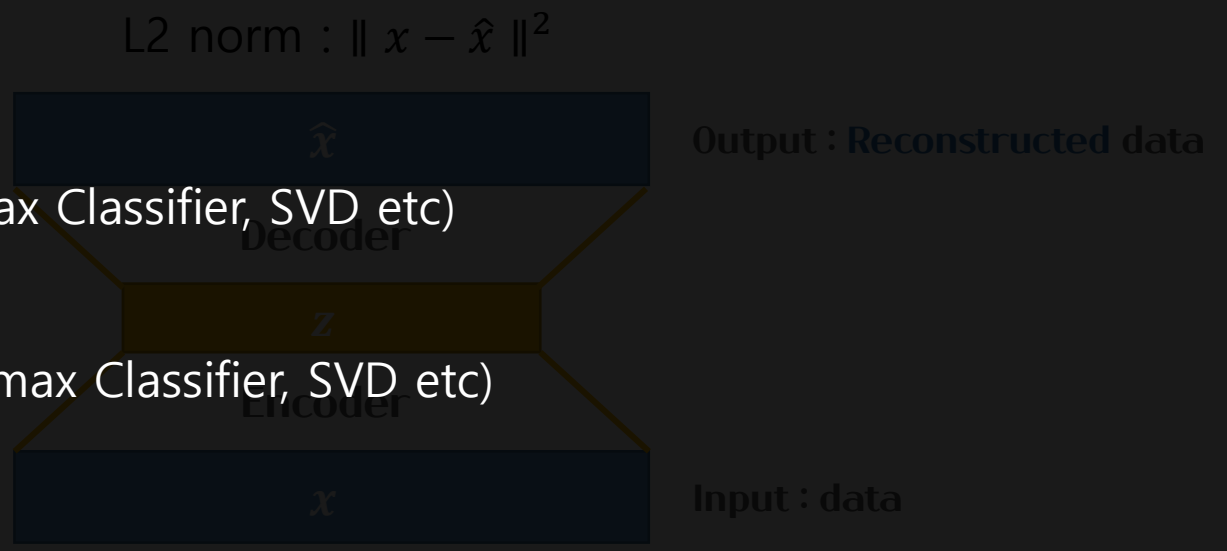
Decoder: 압축된 정보를 원본으로 복원 생성!

Feature Learning, Dimensionality Reduction

→ How? TransposeCNN, FC, …

Encoder + Classifier(Softmax Classifier, SVD etc)

BUT, cannot **sample** "new" data.



WHY?

관련 Task: Semantic segmentation → U-net

2. Generative Model

Discriminative vs Generative "Model"

Discriminative: Learn $P(y|x)$

레이블들 간의 확률을 계산하는 모델이며

사진에 대한 확률이 아니다!

그렇기에 이상한 이미지가 들어온다면,

분류하지 못할 수 있다.

Given

Whale!
Dog..?
Bird?

prob

2. Generative Model

Discriminative vs Generative "Model"

Generative: Learn $P(x)$

Discriminative Model

$$P(x|y) = \frac{P(y|x)}{P(y)} P(x)$$

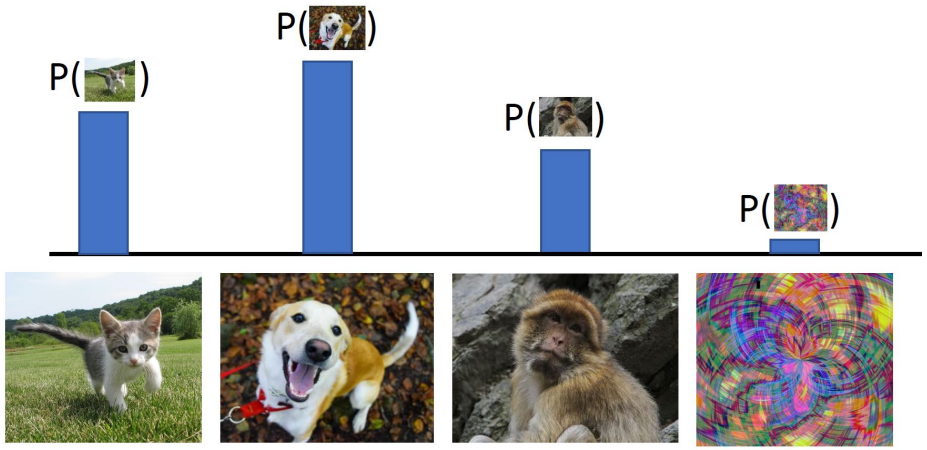
(conditional) Generative Model

Prior over label

(Unconditional) Generative Model

등장하지 않을 사진은 낮은 확률을 가지는 **분포를 계산**하는 모델!

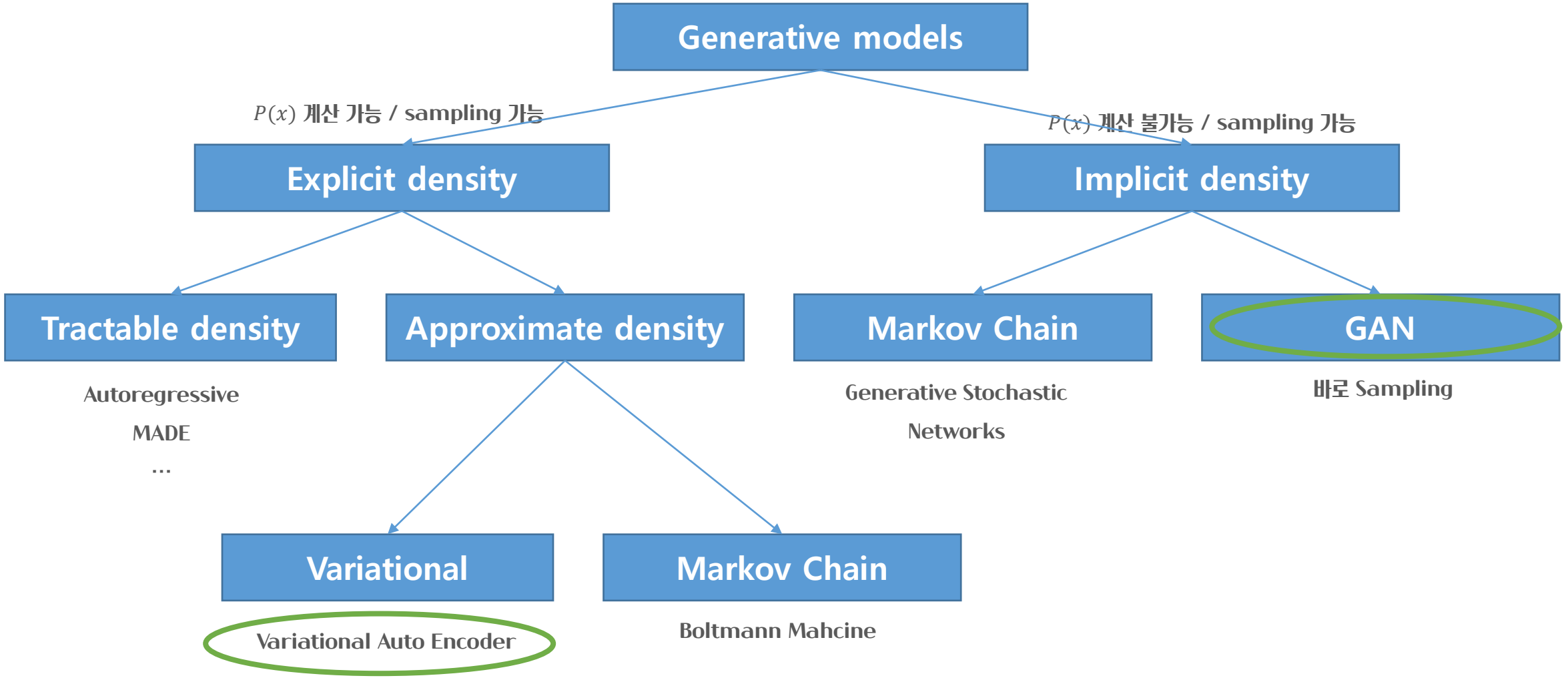
게다가, 분포를 안다면 X의 **sampling**도 가능하지 않을까?



→ 확률분포로 표현해야 샘플링이 가능!

2. Generative Model

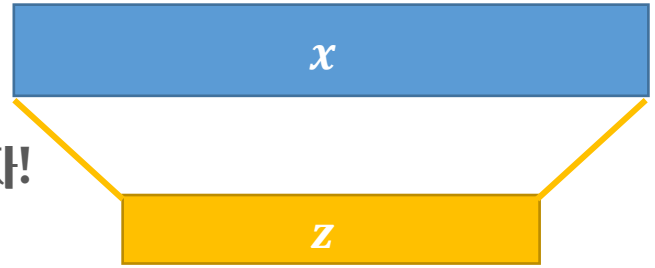
Models



2. Generative Model

Models-VAE

1. Latent feature를 잘 표현해본다
2. Given Latent feature, X가 등장할 확률을 구해보자!



$X \sim p_\theta(x|z)$

$Z \sim p_\theta(z)$

Prior=사전정보 없는데…?

그렇다면 Gaussian 분포를 따른다고 해보자!

$$P(x|z) = \frac{P(z|x)}{P(z)} P(x)$$

(conditional) Generative Model

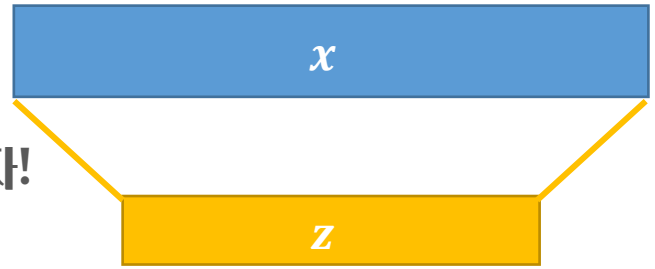
Prior over latent feature

(Unconditional) Generative Model

2. Generative Model

Models-VAE

1. Latent feature를 잘 표현해본다
2. Given Latent feature, X가 등장할 확률을 구해보자!



$X \sim p_\theta(x|z)$

$Z \sim p_\theta(z)$

Conjugate family

Prior를 Gaussian가정 해봤으니,

Posterior도 Gaussian을 따른다고 해보자!

Mean : $\mu_{x|z}$

Covariance: $\Sigma_{x|z}$

(conditional)
Generative Model

$$P(x) = \frac{P(x|z)P(z)}{P(z|x)}$$

Prior over latent feature

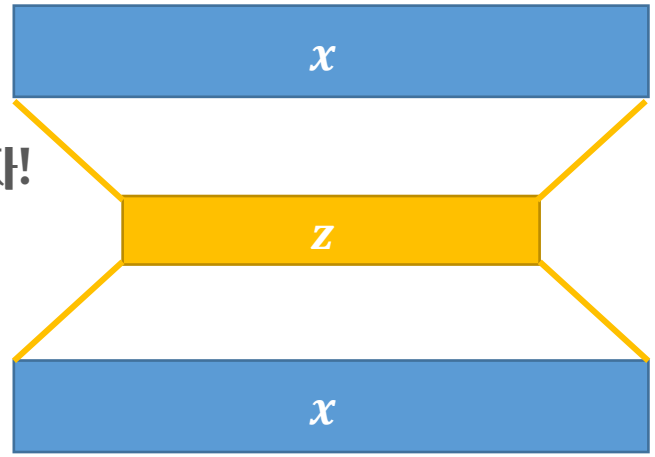
(Unconditional)
Generative Model

Discriminative Model?!

2. Generative Model

Models-VAE

1. Latent feature를 잘 표현해본다
2. Given Latent feature, X가 등장할 확률을 구해보자!



$X \sim p_\theta(x|z)$

$Z \sim p_\theta(z)$

$Z \sim q_\phi(z|x)$

(conditional) Generative Model = Decoder

Discriminative model = Encoder(proxy)

(conditional)
Generative Model

$$P(x) = \frac{P(x|z)P(z)}{q_\phi(z|x)}$$

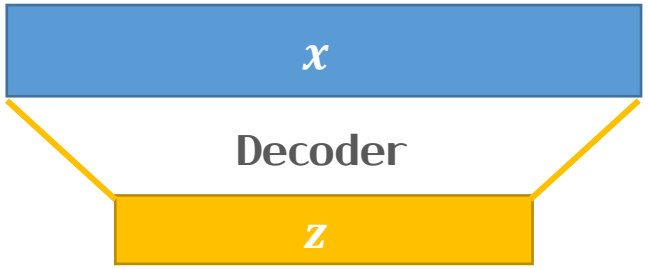
Prior over latent feature

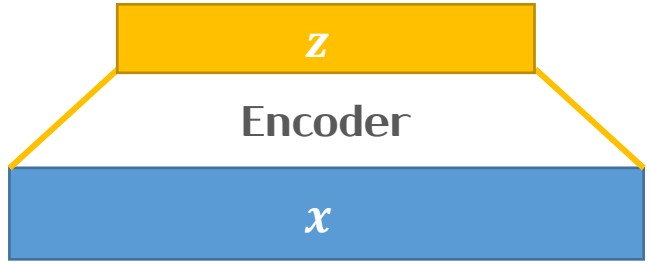
(Unconditional)
Generative Model

Encoder

2. Generative Model

Models-VAE





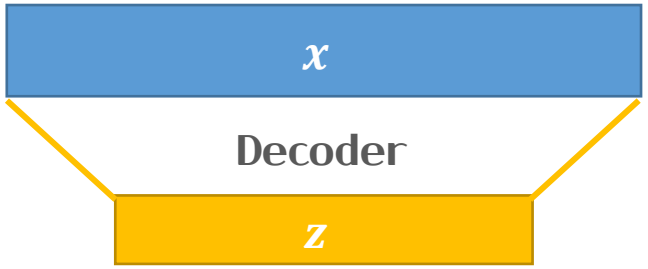
$$p_\theta(x|z) = N\left(\mu_{x|z}, \Sigma_{x|z}\right)$$

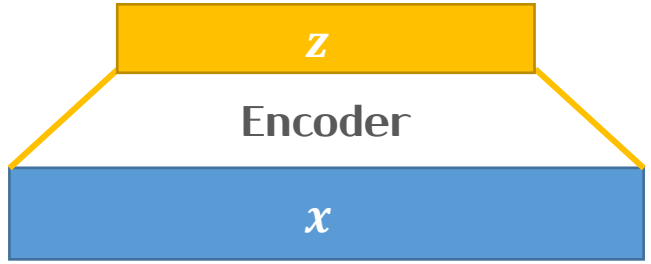
$$q_\phi(z|x) = N\left(\mu_{z|x}, \Sigma_{z|x}\right)$$

→ **Approximate** $p_\theta(x) \cong \frac{p_\theta(x|Z)p(z)}{q_\phi(Z|x)}$

2. Generative Model

Models-VAE





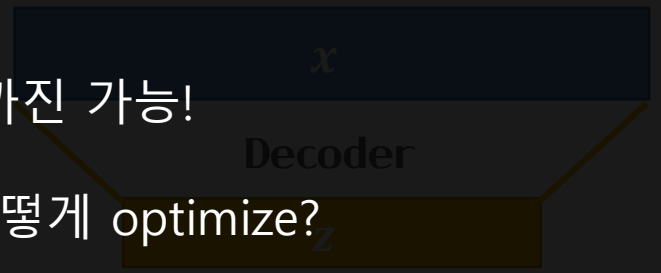
$$p_\theta(x|z) = N\left(\mu_{x|z}, \Sigma_{x|z}\right)$$

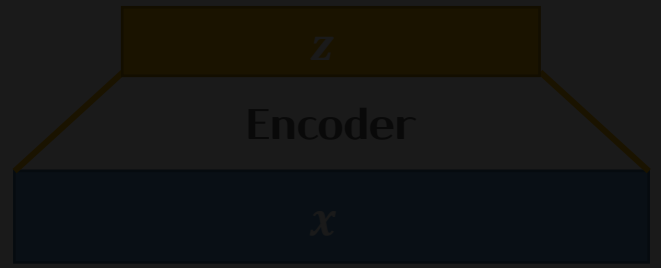
$$q_\phi(z|x) = N\left(\mu_{z|x}, \Sigma_{z|x}\right)$$

이것만 계산해서 알아내면 된다!

2. Generative Model

Models-VAE





Estimate까진 가능!

그런데 어떻게 optimize?

이어서 자세한 식을 드리지만, 이번 시간에서는 그 개념과 통계적인 해석만을 가져봅시다.

$$p_\theta(x|z) = N(\mu_{x|z}, \Sigma_{x|z})$$

$$q_\phi(z|x) = N(\mu_{z|x}, \Sigma_{z|x})$$

$$\rightarrow \textbf{Approximate } p_\theta(x) \cong \frac{p_\theta(x|z)p(z)}{p_\theta(x|z)}$$

2. Generative Model

$$\log p_\theta(x) = \log \frac{p_\theta(x \mid z)p(z)}{p_\theta(z \mid x)} = \log \frac{p_\theta(x|z)p(z)q_\phi(z|x)}{p_\theta(z|x)q_\phi(z|x)}$$

$$= \log p_\theta(x|z) - \log \frac{q_\phi(z|x)}{p(z)} + \log \frac{q_\phi(z|x)}{p_\theta(z|x)}$$

$$= E_z[\log p_\theta(x|z)] - E_z\left[\log \frac{q_\phi(z|x)}{p(z)}\right] + E_z\left[\log \frac{q_\phi(z|x)}{p_\theta(z|x)}\right]$$

2. Generative Model

$$= E_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}\left(q_\phi(z|x), p(z)\right) + D_{KL}(q_\phi(z|x), p_\theta(z|x))$$

Data Reconstruction

Divergence btw samples from Encoder & samples from prior

Divergence btw Encoder & counterpart

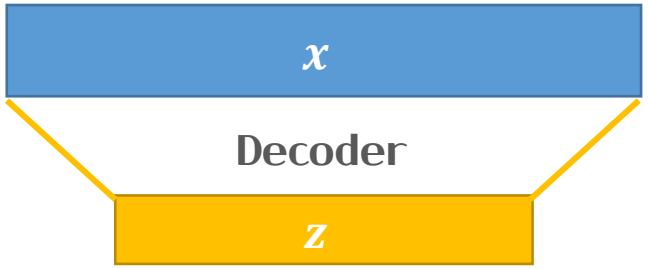
$$\log p_\theta(x) \geq E_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}\left(q_\phi(z|x), p(z)\right)$$

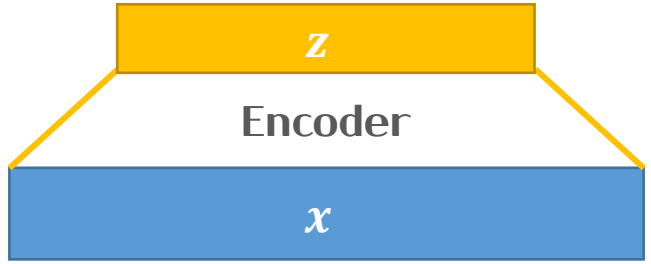
Gap btw ELBo & Evidence

Evidence Lower Bound : ELBo

2. Generative Model

Models-VAE





$$p_\theta(x|z) = N\left(\mu_{x|z}, \Sigma_{x|z}\right)$$

$$q_\phi(z|x) = N\left(\mu_{z|x}, \Sigma_{z|x}\right)$$

Decoder는 z를 주면 $p_\theta(x|z)$를 계산해줌!

2. Generative Model

$$= E_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}\left(q_\phi(z|x), p(z)\right) + D_{KL}(q_\phi(z|x), p_\theta(z|x))$$

Data Reconstruction

Divergence btw samples from Encoder & samples from prior

Divergence btw Encoder & counterpart

$$\log p_\theta(x) \geq E_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}\left(q_\phi(z|x), p(z)\right)$$

Encoder로 sample z를 생성 후 평균값 취하기!

2. Generative Model

$$= E_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}\left(q_\phi(z|x), p(z)\right) + D_{KL}(q_\phi(z|x), p_\theta(z|x))$$

Data Reconstruction

Divergence btw
samples from Encoder
& samples from prior

Divergence btw
Encoder & counterpart

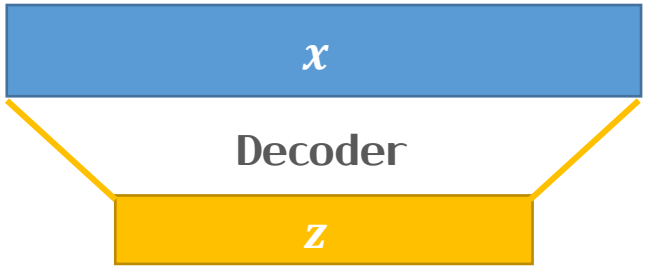
$$\log p_\theta(x) \geq E_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}\left(q_\phi(z|x), p(z)\right)$$

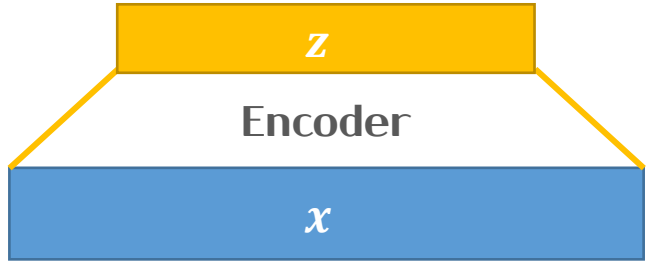
$q_\phi(z|x)$가 diagonal covariance

$p(z)$가 std Normal 분포라면 closed form

2. Generative Model

Models-VAE





$$p_\theta(x|z) = N\left(\mu_{x|z}, \Sigma_{x|z}\right)$$

$$q_\phi(z|x) = N\left(\mu_{z|x}, \Sigma_{z|x}\right)$$

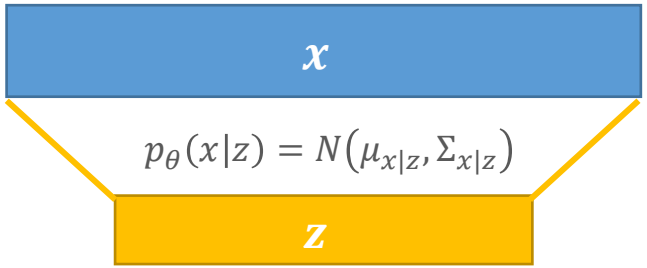
→ **Approximate** $p_\theta(x) \cong \frac{p_\theta(x|Z)p(z)}{q_\phi(Z|x)}$

2. Generative Model

Models-VAE

Sampling(생성) 법

1. Prior of z(Gaussian)으로 표본 생성 : $p_\theta(z)$
2. Decoder로 posterior의 Parameter 계산: $\mu_{x|z}, \Sigma_{x|z}$
3. Posterior 분포로 샘플링: $p_\theta(x|z)$



그렇지만,,, 좋은 퀄리티의 sample이 아니었다!(blurred…)

게다가 Evidence 자체가 아닌, "lower bound"를 maximize를 한 것

2. Generative Model

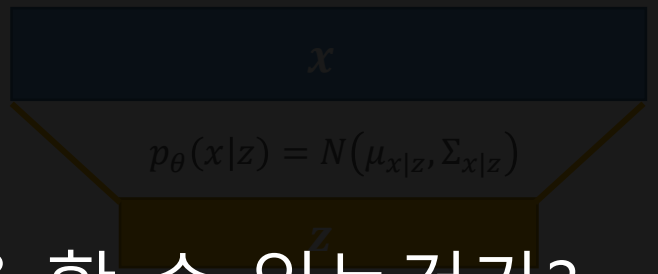
Models-VAE

Sampling(생성) 법

1. Prior of z(Gaussian)으로 표본 생성 : $p_\theta(z)$
2. Decoder로 posterior의 Parameter 계산: $\mu_{x|z}, \Sigma_{x|z}$
3. Posterior 분포로 샘플링 : $p_\theta(x|z)$

그런데...

꼭 확률을 계산해야 sampling을 할 수 있는건가?



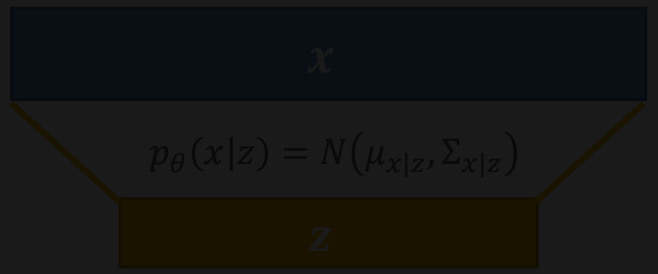
2. Generative Model

Models-VAE

Sampling(생성) 법

1. Prior of z(Gaussian)으로 표본 생성 : $p_\theta(z)$
2. Decoder로 posterior의 Parameter 계산: $p_\theta(x|z)$
3. Posterior 분포로 샘플링: $p_\theta(x|z)$

다음주에!



Reference

고려대학교 산업경영공학부 DSBA 연구실 https://www.youtube.com/watch?v=IwtexRHoWG0

https://github.com/pilsung-kang/Text-Analytics

https://www.youtube.com/watch?v=MN__lSncZBs

https://medium.com/daangn/딥러닝으로-동네생활-게시글-필터링하기-263cfe4bc58d

Stanford CS224N: https://www.youtube.com/watch?v=knTc-NQSjKA

이기복 교수님 "통계적 머신러닝" 강의안

강승호 교수님 "이론통계학(1)" 강의안

임종호 교수님 "수리통계학(2) 강의

5기 한영웅님 AutoEncoder 세션 자료

Appendix

왜 $q_\phi(z|x)$가 diagonal covariance이고 $p(z)$가 std Normal 분포라면 closed form인지

$$-D_{KL}\left(q_\phi(z|x), p(z)\right) = \int_Z q_\phi(z|x) \log \frac{p(z)}{q_\phi(z|x)} dz$$

$$= \int_Z N\left(z; \mu_{z|x}, \Sigma_{z|x}\right) \log \frac{N(z; 0, I)}{N\left(z; \mu_{z|x}, \Sigma_{z|x}\right)} dz$$

$$= \frac{1}{2} \sum_{j=1}^{J} \left(1 + \log\left(\left(\Sigma_{z|x}\right)_j^2\right) - \left(\mu_{z|x}\right)_j^2 - \left(\Sigma_{z|x}\right)_j^2\right)$$