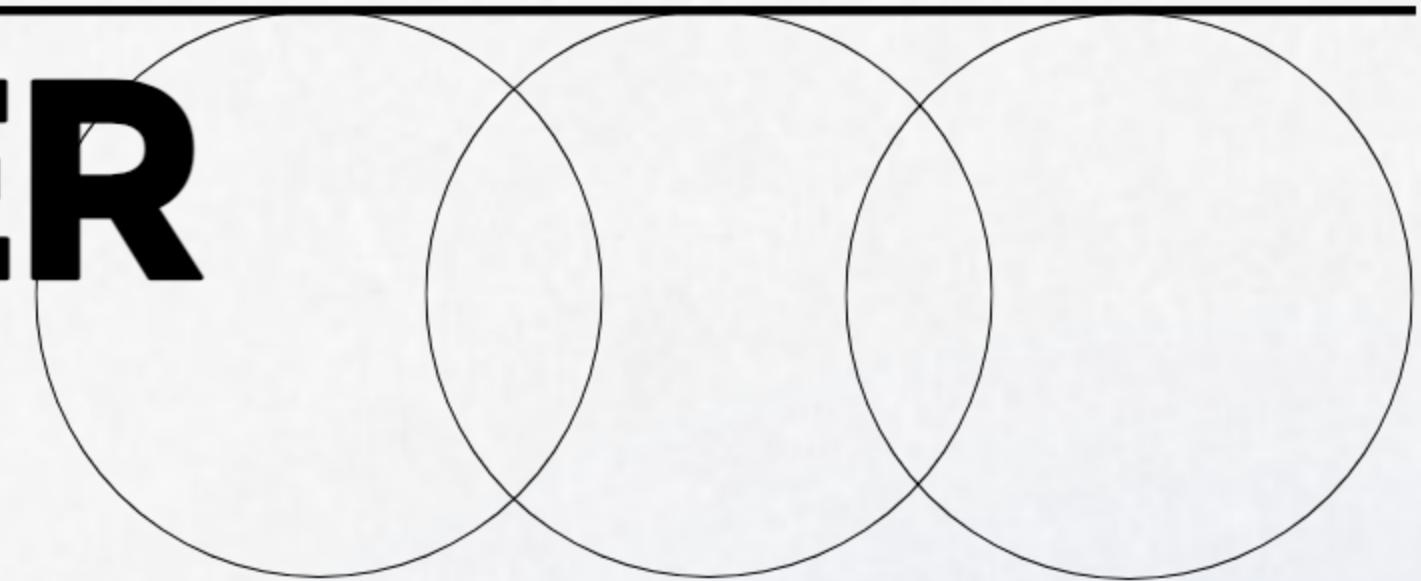
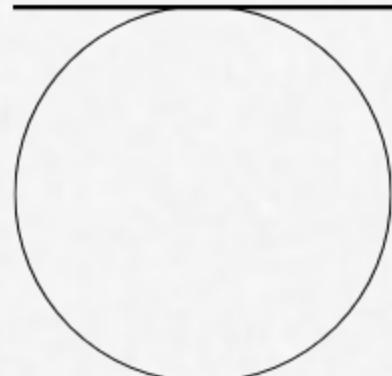


Recommender Syetem:

RECOMMENDER SYSTEM



PAPER REVIEW



10 jobs

-당신에게드리는 10가지 직업 정보

AI빅데이터융합경영학과 김서령

AI빅데이터융합경영학과 김예향

광고학 전공 박지훈

행정학 전공 황건하

Deep Learing

con

AUTOENCODERS: RS

DEEPMF: FOR CTR PREDICTION

BERT: FOR EMBEDDING

HYBRID: FOR PERFORMANCE

PART1

PART2

PART3

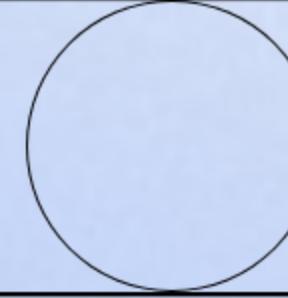
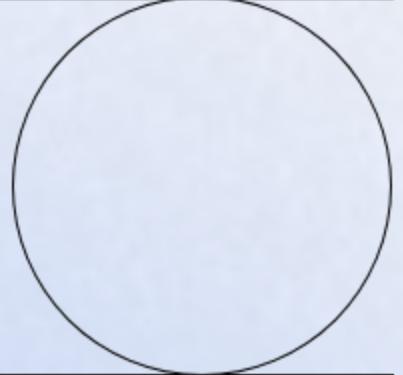
PART4

tents⁺

Recommender Syetem:

PART 1

AUTOENCODERS FOR RECOMMENDER SYSTEMS



10 jobs -당신에게 드리는 10가지 직업 정보

AI빅데이터융합경영학과 김예향



Deep Learning

Intro & Related Work

Previous Problem

-
-
-
-

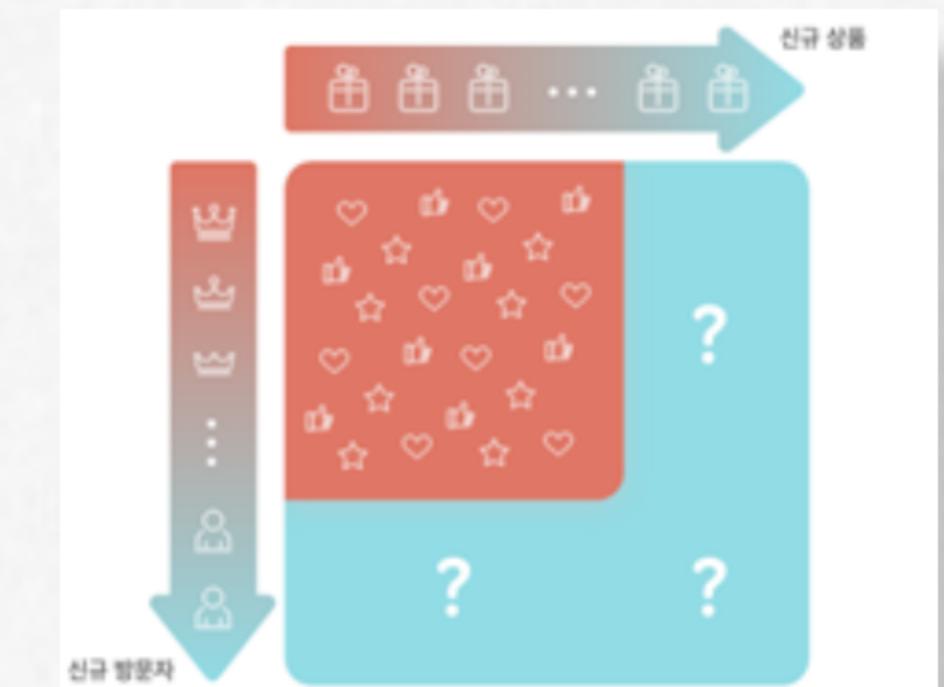
비지도학습 → AutoEncoder

차원 축소 -> 핵심 요소 추출

비선형 차원 축소 방법

오토인코더의 잠재 공간
-> 유저와 아이템의 **잠재적인 요인**을 반영
-> 추천에 활용!

콜드 스타트 문제
-> 새로운 유저 / 새로운 아이템
나타났을 때

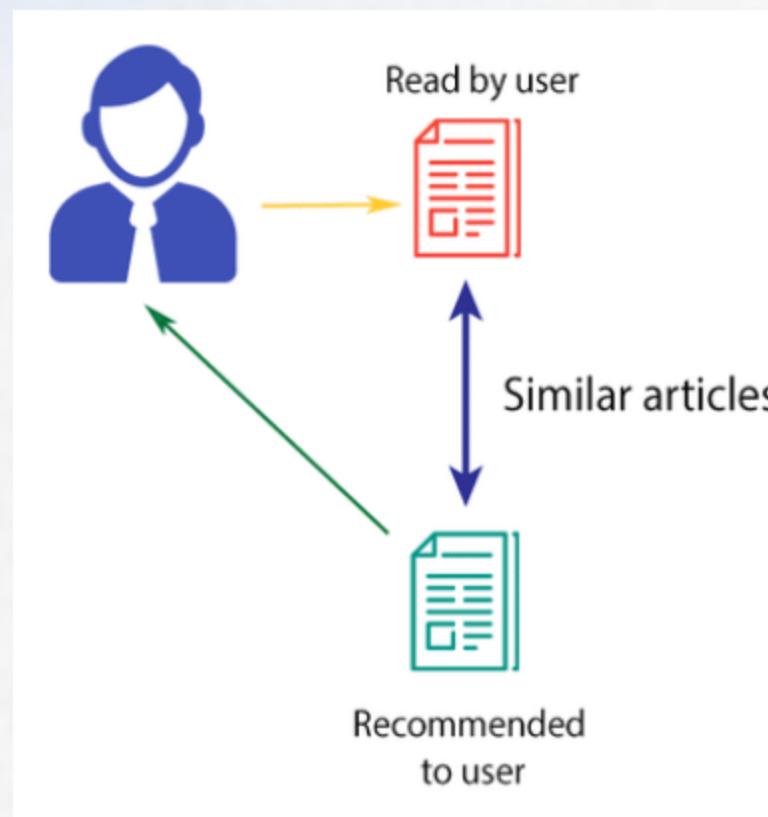


Intro & Related Work

Previous Problem

'오토인코더를 이용한 딥러닝 기반 추천시스템 모형의 비교연구'

• • •



콘텐츠 기반 필터링
(content-based filtering)

+



협업 필터링
(collaborative filtering)

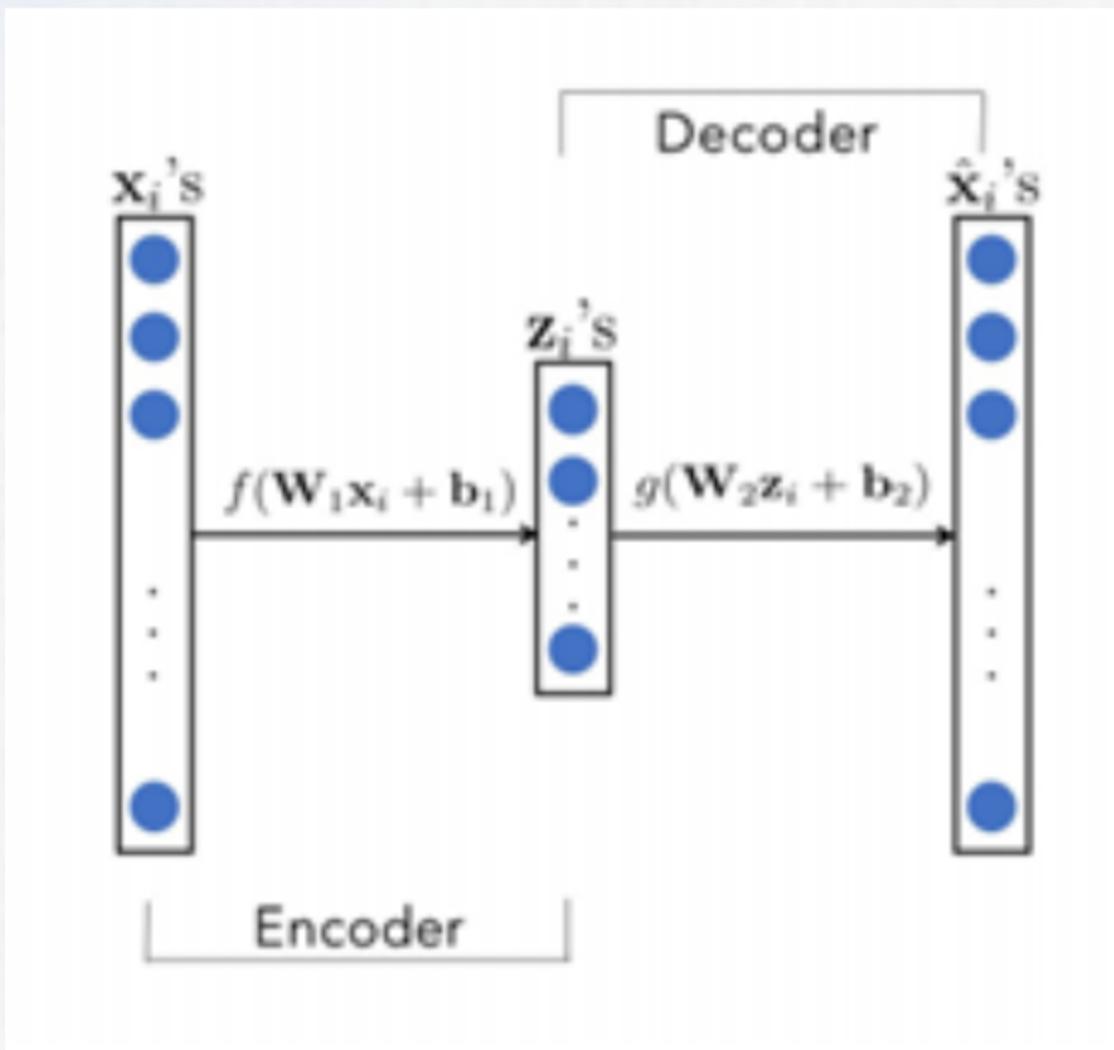
=

하이브리드 방법
(hybrid recommender system)

Model

1. AutoEncoder (기본 오토인코더)

•
•
•



인코더 : 핵심 요소만 **추출**

은닉층 : 추출된 핵심 요소들의 집합

디코더 : 입력값과 동일한 차원의 **출력값 산출**

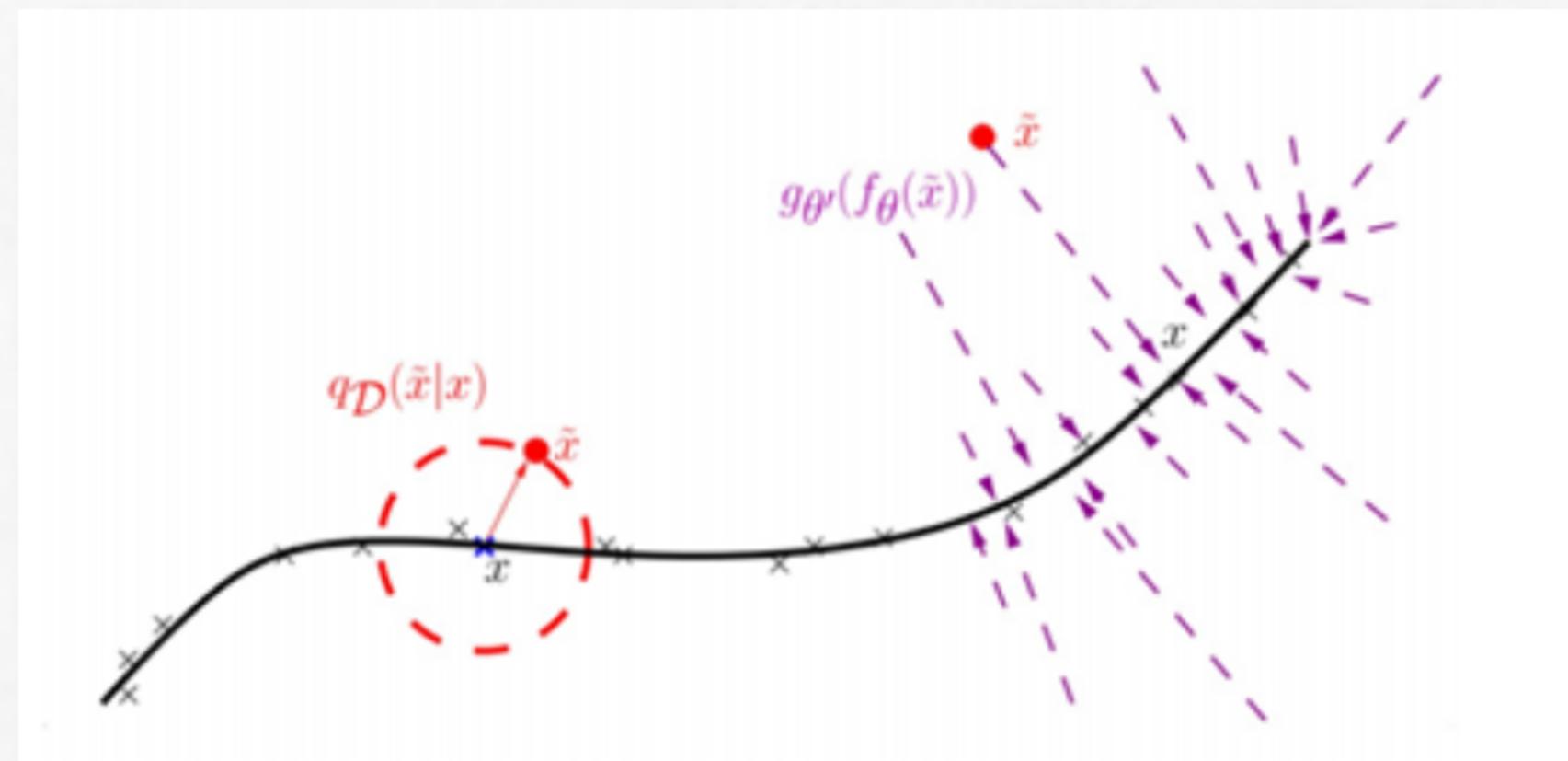
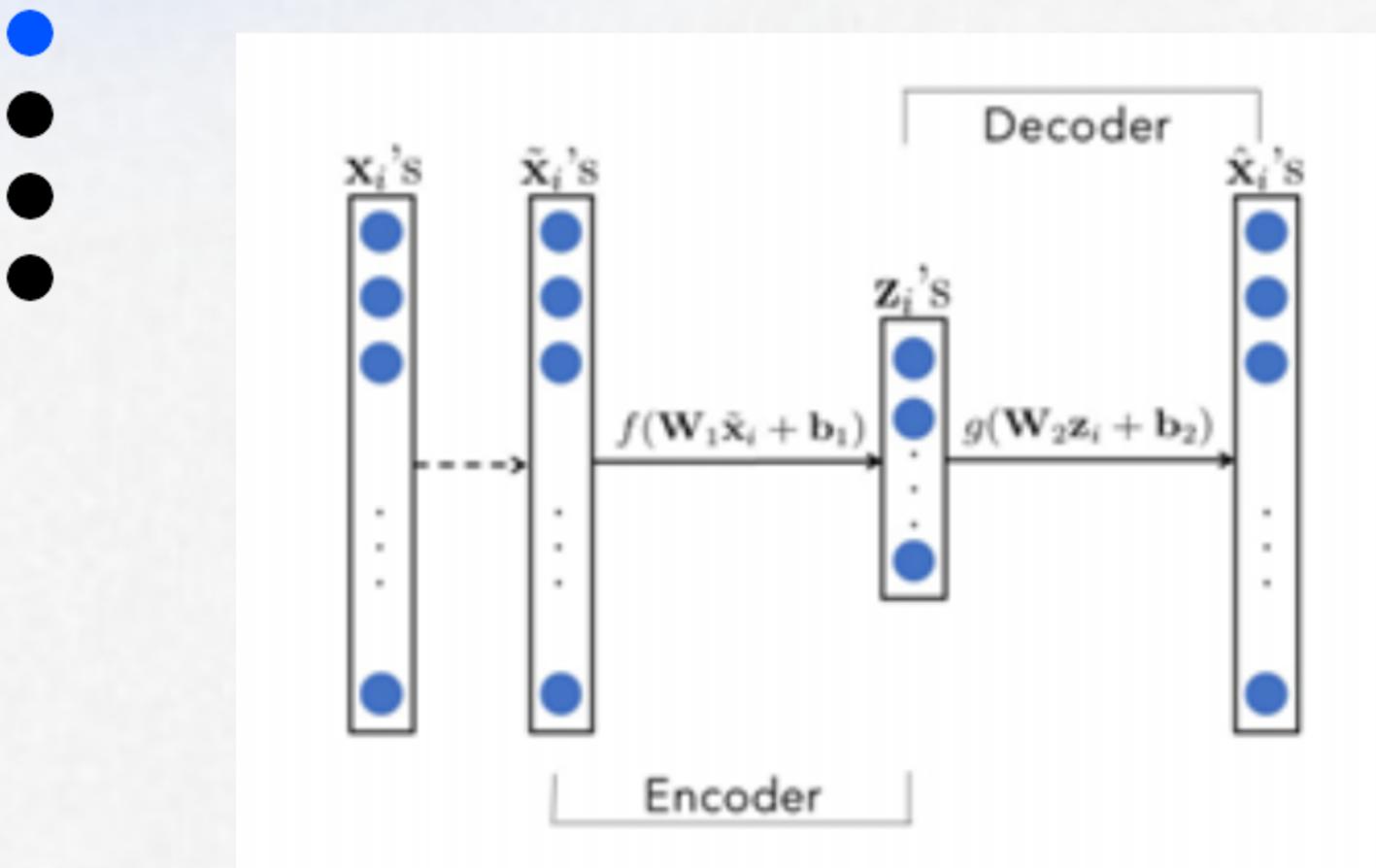
목적함수 :
input, output 사이의 **RMSE** 최소화하는 것

$$\underset{\mathbf{W}_1, \mathbf{W}_2, \mathbf{b}_1, \mathbf{b}_2}{\operatorname{argmin}} L(\mathbf{X}, \hat{\mathbf{X}}) = \underset{\mathbf{W}_1, \mathbf{W}_2, \mathbf{b}_1, \mathbf{b}_2}{\operatorname{argmin}} \sqrt{\sum_{i=1}^N \sum_{j=1}^M \frac{(x_{ij} - \hat{x}_{ij})^2}{NM}}.$$

Model

2. Denoising AutoEncoder (잡음 제거 오토인코더)

매니폴드 학습 관점



잡음이 추가된 입력값을 최대한 잡음이 추가되기 전의
입력값으로 재구성

-> 대표성이 더욱 높은 요소 학습!

Model

3. Variational AutoEncoder (변분 오토인코더)

•

•

•

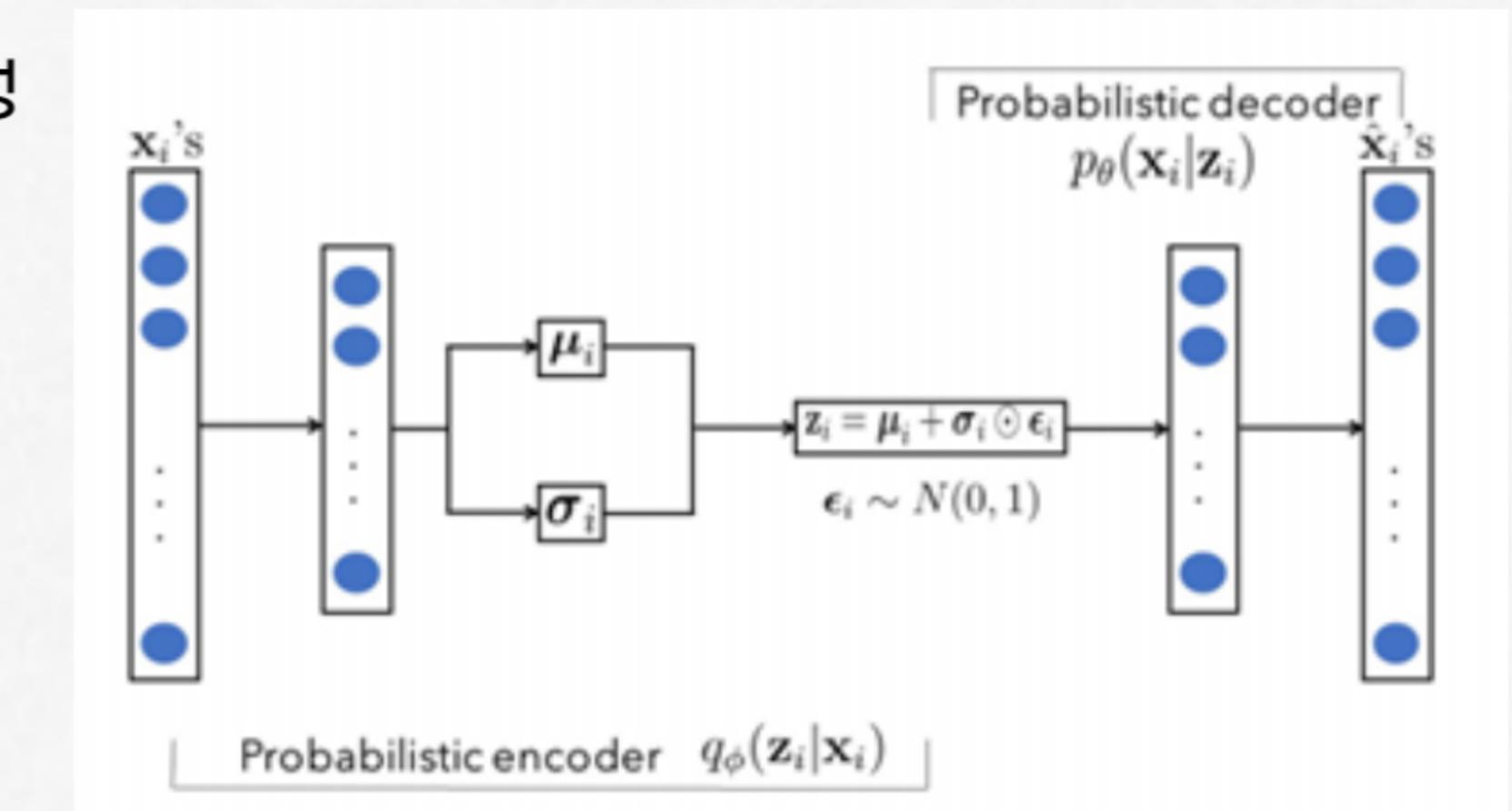
추출될 핵심 요소의 분포 찾기 -> 새로운 자료 생성

q : 가우시안 확률 분포

p : z 로부터 x_i 를 생성해내는 부분

p 와 q 를 **근사시키는** 것이 목적

두 분포의 유사성을 측정하는 척도인
쿨백-라이블러 발산 **최소화**



Model

4. AutoSVD / 5. AutoSVD++

• 잡음 제거 오토인코더

잡음이 추가된 입력값으로부터

잡음이 제거된 출력값을 산출하도록

디코더 부분을 학습

축약 오토인코더

입력값에 변화가 있어도,

원래 입력값을 대표할 수 있는 핵심 요소를 추출하는
인코더 부분 학습

축약 오토인코더 + SVD 방법 / SVD++ 방법

- 아이템의 **보조 정보 (side information)**
- 아이템에 대해 추출된 **비선형 잠재 요소 고려**

$$\hat{x}_{ij} = \mu + b_i + b_j + (\beta \cdot \text{cae}(\mathbf{x}_j) + \epsilon_j)^T \mathbf{p}_i.$$

AutoSVD++

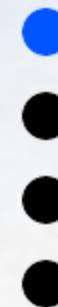
내재적 피드백을 함께 고려한 방법

$$\hat{x}_{ij} = \mu + b_i + b_j + (\beta \cdot \text{cae}(\mathbf{x}_j) + \epsilon_j)^T \left(\mathbf{p}_i + |N(i)|^{-\frac{1}{2}} \sum_{j \in N(i)} \mathbf{y}_j \right).$$

Data Analysis / conclusion

상위의 아이템, 유저에 대한 정보만을 고려
희박성 비율 : 82% 94% 98%

1. MovieLens 1M Data : 영화에 대한 사용자들 평점
2. Amazon Review Data : 제품에 대한 후기, 제품정보
3. Yelp Review Data : 레스토랑, 여러 서비스에 대한 리뷰



		Hyperparameters	AE	DAE	VAE	AutoSVD	AutoSVD++
MovieLens 1M data	Nodes	(350, 245, 171)	(359, 245, 171)	(165, 54, 18, 6)	(165, 54)	(165, 54)	
	Dropout	0.4	0.6	0.6	0.6	0.6	
	Activation function	Tanh, ReLU	SeLU, ReLU	Tanh, ReLU	SeLU, ReLU	SeLU, ReLU	
	λ, γ	-	-	-	0.1, 0.01	0.1, 0.01	
	masked RMSE	0.8898	0.9551	0.9723	0.8600	0.8651	
	MAE	0.7012	0.7814	0.7611	0.6768	0.6811	
Amazon Review data	Nodes	(165, 54, 18)	(165, 54, 18, 6)	(165, 54, 18, 6)	(350, 245)	(350, 245)	
	Dropout	0.4	0.6	0.4	0.4	0.4	
	Activation function	SeLU, ReLU	ReLU, ReLU	Tanh, ReLU	Tanh, ReLU	Tanh, ReLU	
	λ, γ	-	-	-	0.1, 0.001	0.001, 0.0001	
	masked RMSE	0.8493	0.8505	0.8981	0.9355	0.8763	
	MAE	0.6403	0.6665	0.6905	0.7097	0.7242	
Yelp Review data	Nodes	(350, 245, 171)	(165, 54, 18, 6)	(165, 54, 18, 6)	(165, 54)	(165, 54)	
	Dropout	0.4	0.6	0.4	0.4	0.4	
	Activation function	SeLU, ReLU	ReLU, ReLU	Tanh, ReLU	ReLU, ReLU	ReLU, ReLU	
	λ, γ	-	-	-	0.001, 0.001	0.0001, 0.001	
	masked RMSE	0.9307	0.9834	1.0577	0.9512	0.9524	
	MAE	0.7368	0.7676	0.7920	0.7392	0.7406	

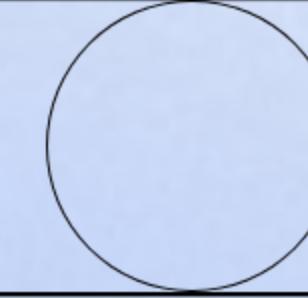
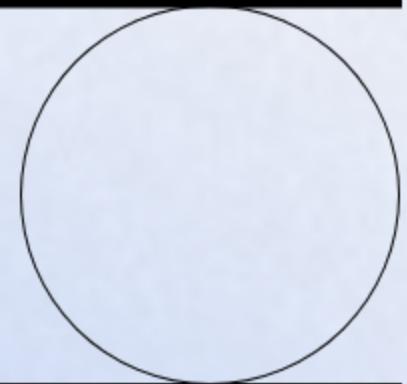
결론

세 데이터 모두 **오토인코더 방법**이
잡음 제거 오토인코더와 변분 오토인코더에 비해 성능이 우수함

희박성 비율이 높은 두 데이터는 AutoSVD, AutoSVD++에 비해
잡음제거 오토인코더와 변분 오토인코더 성능이 더 우수함

PART 2

DEEPMF: FOR CTR PREDICTION



10 jobs -당신에게 드리는 10가지 직업 정보

AI빅데이터융합경영학과 김서령

Deep Learning



Intro & Related Work

what is CTR?



CTR (Click-through rate)

추천된 아이템을 유저가 클릭할 확률

-
-
-

CTR 예측?



Implicit Feature Interaction

- 유저들이 식사 시간에 음식 배달 앱을 다운로드 경향이 있음
→ ‘app category’ 와 ‘time’ 간의 interaction (order-2)
- 10대 남자들은 슈팅 게임과 RPG게임을 좋아하는 경향이 있음
→ ‘app category’ 와 ‘gender’ 와 ‘age’ 간의 interaction (order-3)
- 기저귀를 구매하는 사람은 맥주를 함께 구매하는 경향이 있음
→ ‘beer’ 와 ‘diaper’ 간의 숨겨진 interaction (order-2)

DeepFM

what is DeepFM?

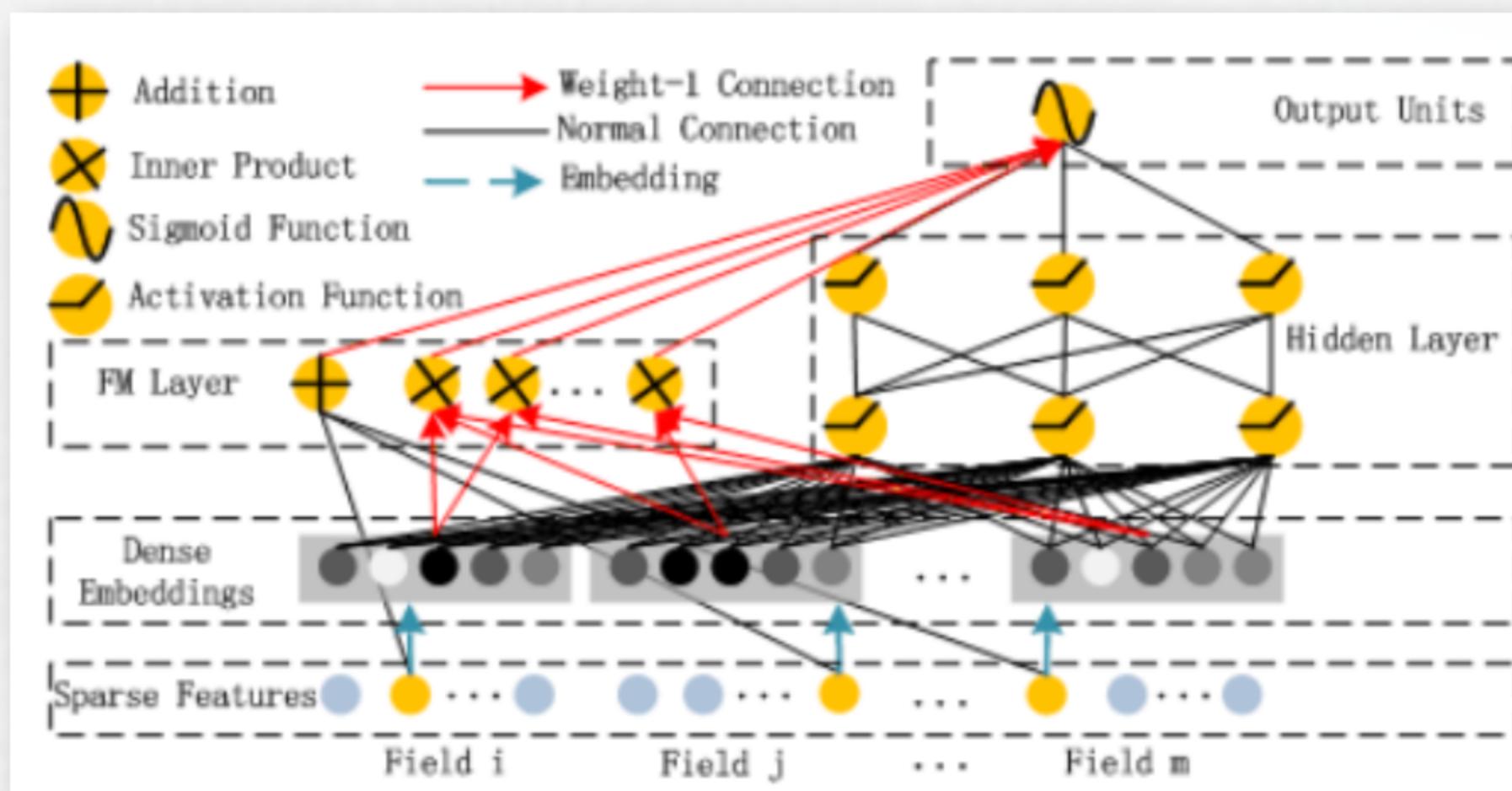
DeepFM = Factorization Machine + Deep Learning



Low-order
feature interaction



High-order
feature interaction



All-orders interaction을
고려할 수 있는 모델인
DeepFM을 제안

Related Work

what is FM?

FM (Factorization Machine)

- Latent space로 각각의 특성들을 매핑하여 벡터의 내적을 통해 Interaction을 계산하는 방법

$$\hat{y}(\mathbf{x}) := w_0 + \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j=i+1}^n \langle \mathbf{v}_i, \mathbf{v}_j \rangle x_i x_j$$

연산 복잡도가 Linear(선형)하게 유지됨 → order -2 ~ order-n Interaction 모델링 가능

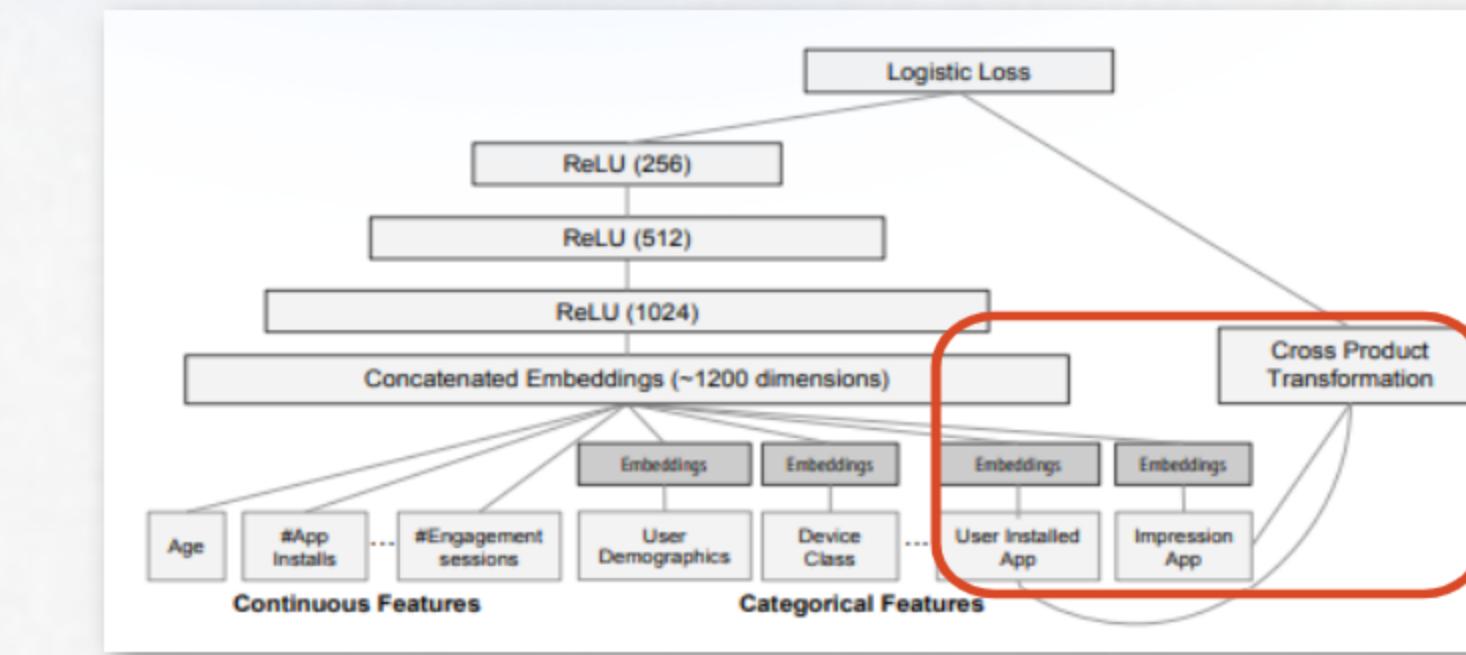
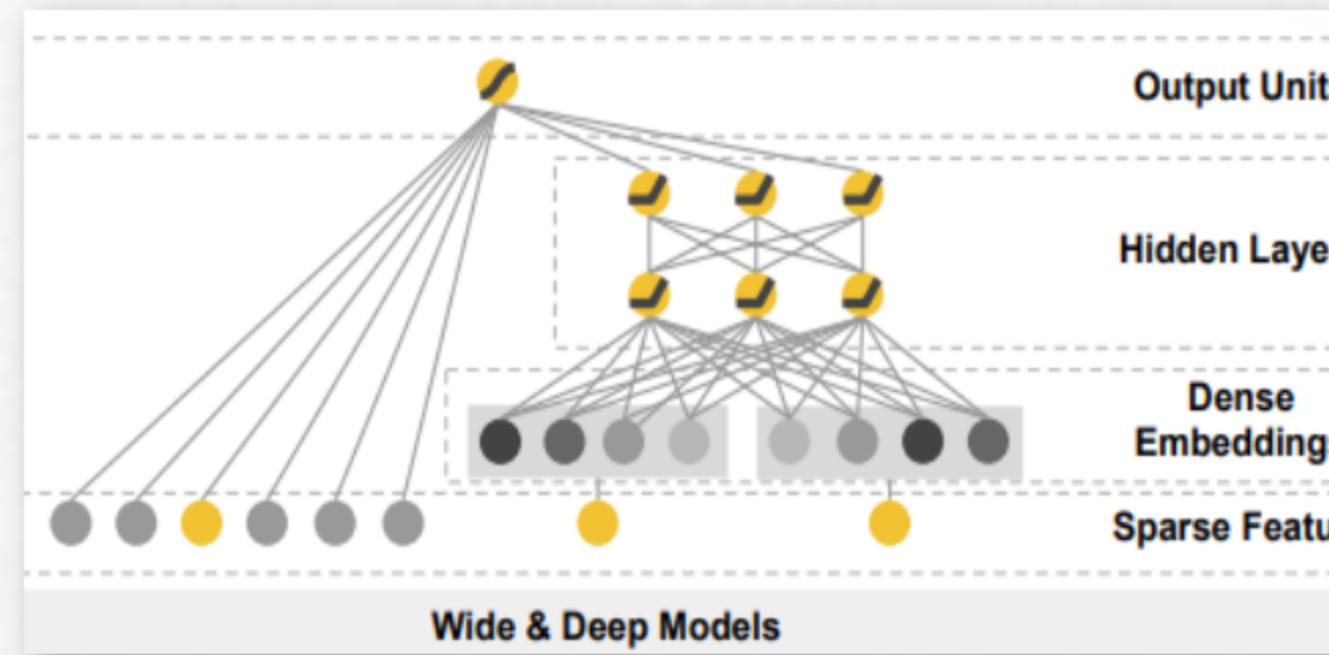
But, 실제로는 높은 복잡도로 인해 High-order이 아닌 order-2 가 주로 사용됨

Related Work

what is Wide & Deep?

Wide & Deep

Linear Model (Wide) + Feed-Forward Neural Network (Deep)



High-order feature interaction을 고려하기 위해 Wide Component에서 Cross Product Transformation하여 새로운 특성으로 추가함

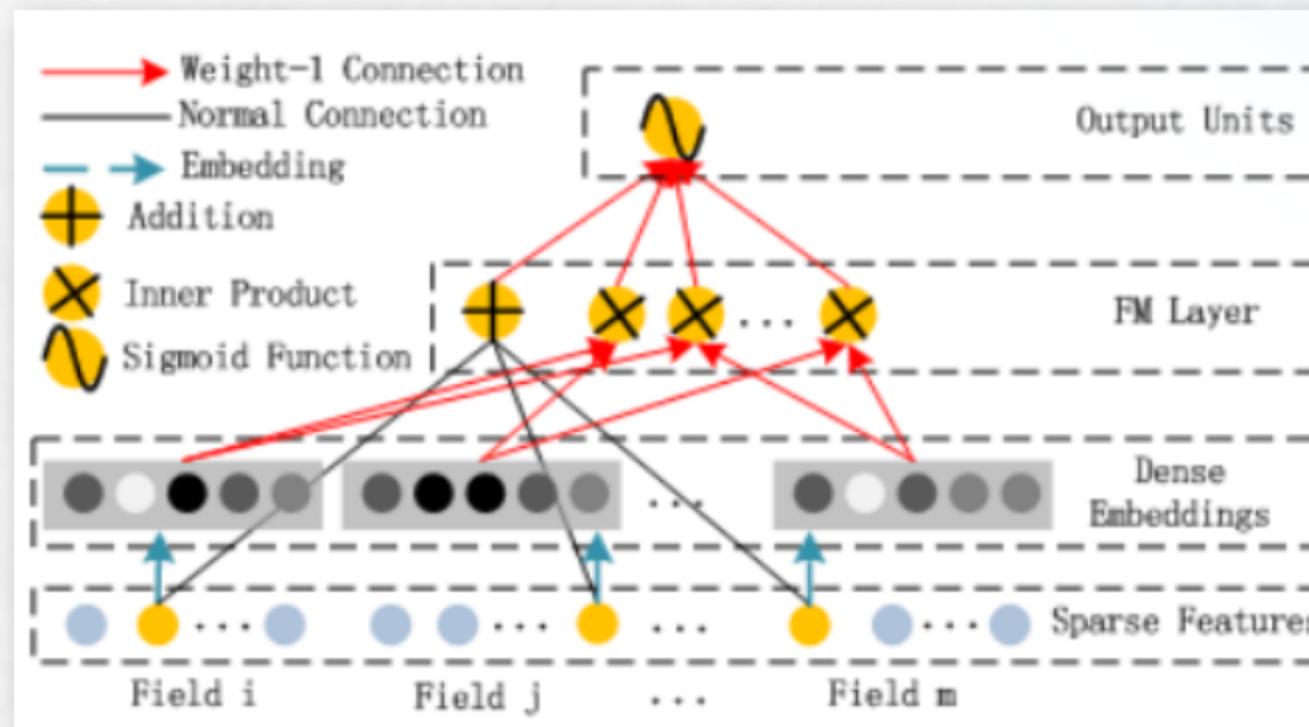


전문적인 Feature Engineering이 필요

Architecture

FM Component & Deep Component

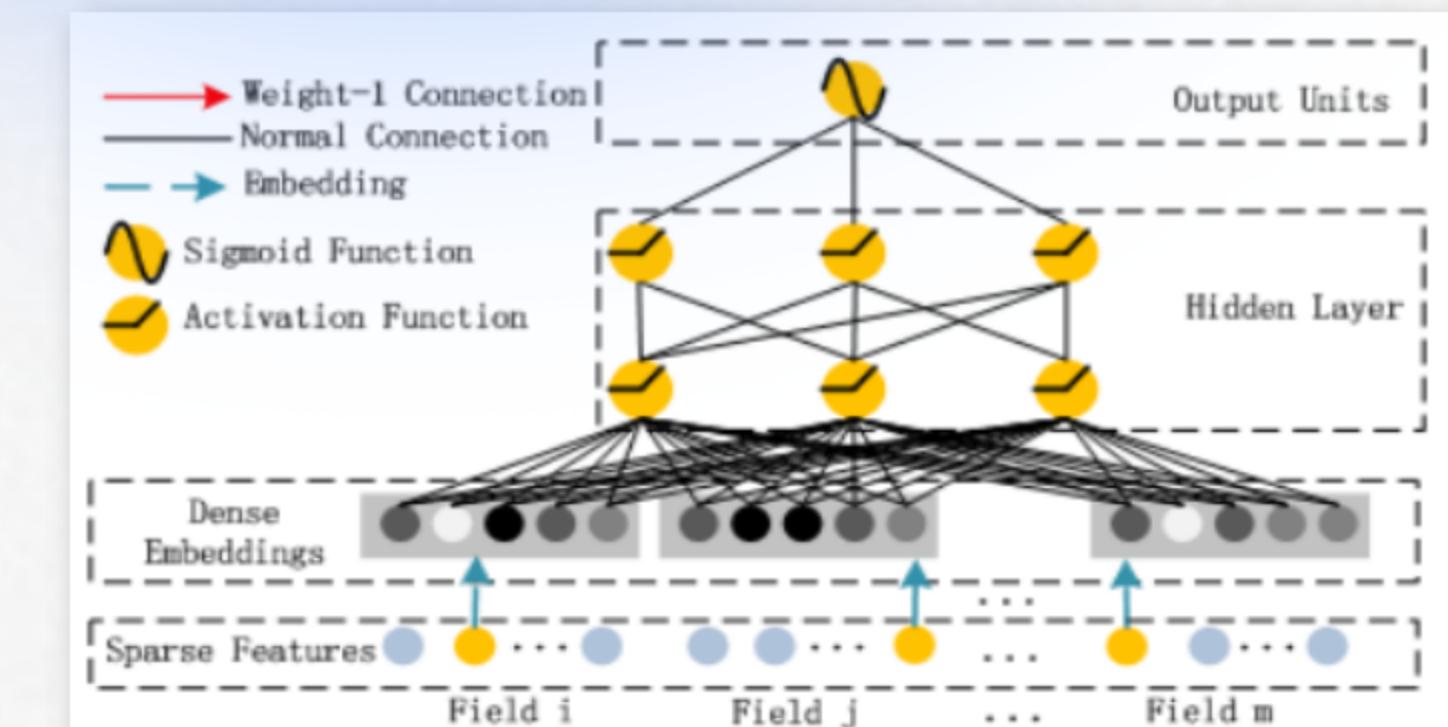
FM Component = Factorization Machine



$$y_{FM} = \langle w, x \rangle + \sum_{j_1=1}^d \sum_{j_2=j_1+1}^d \langle V_i, V_j \rangle x_{j_1} \cdot x_{j_2}$$

- $\langle w, x \rangle$: order-1 feature의 중요도
- $\langle V_i, V_j \rangle$: 특성 i, j 사이의 내적으로, order-2 가중치
(다른 특성과의 interaction)

Deep Component = Feed-Foward Neural Network

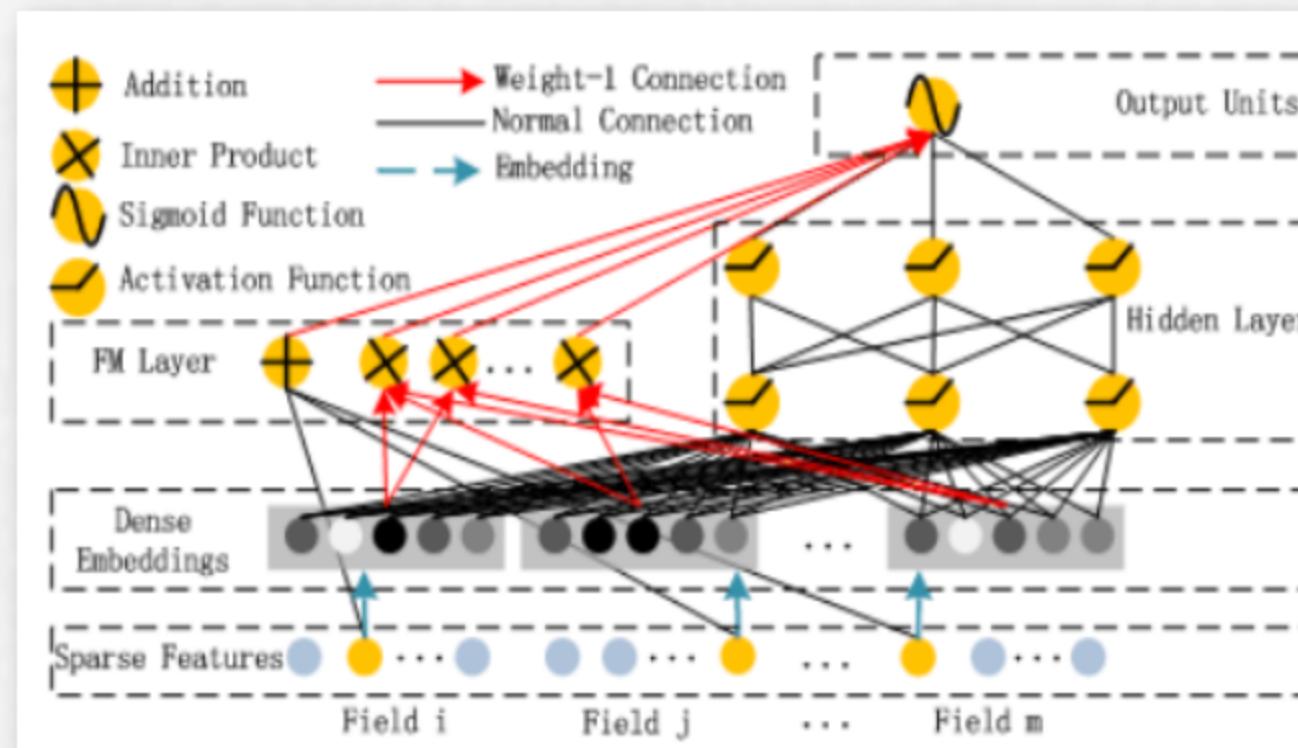
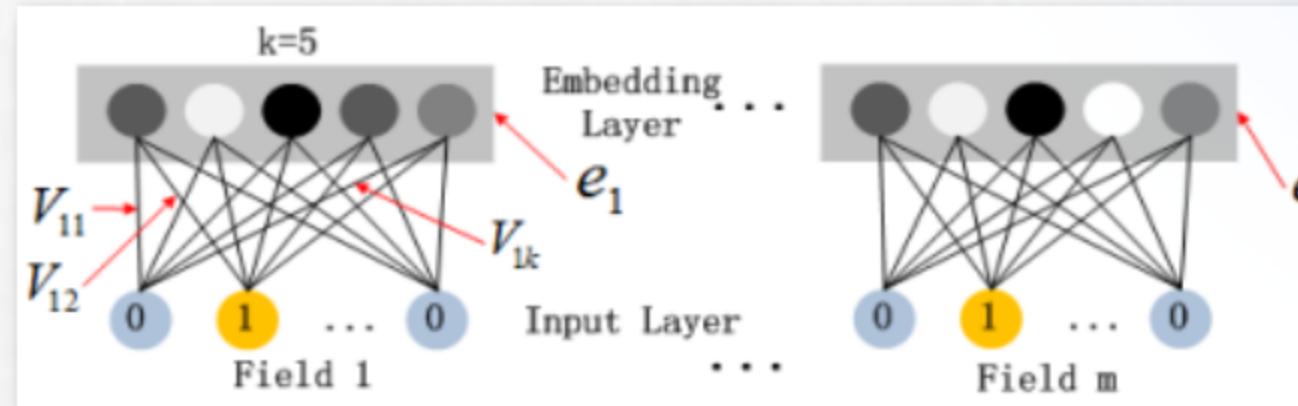


$$y_{DNN} = \sigma(W^{|H|+1}a^{|H|} + b^{|H|+1})$$

- 모든 특성의 저차원 임베딩을 신경망의 Input으로 활용함

Architecture

DeepFM Architecture



$$\hat{y} = \text{sigmoid}(y_{FM} + y_{DNN})$$

- FM Component에서 order-2 Interaction을 계산하기 위한 가중치로 쓰인 V 가 Deep Component에서는 input 특성들을 임베딩 벡터로 만들기 위한 Embedding layer의 가중치 역할을 함
→ 두 Component가 동일한 Embedding을 공유함
- 두 Component가 서로 같은 Embedding을 공유하는 이점
 - 1) Low-order와 High-order feature interaction을 모두 고려할 수 있음
 - 2) 전문적인 Feature Engineering을 필요로 하지 않음
- FM의 output과 DNN의 output을 합하여 sigmoid 통과함
- 본 연구의 목적은 CTR의 예측이므로, $y \in (0,1)$ 이며 sigmoid를 사용함

Experiments

Dataset & Experiment

Dataset

유저 클릭 기록을 포함한 Criteo Dataset , Company Dataset

Evaluation Metrics

AUC와 Logloss를 평가 지표로 사용

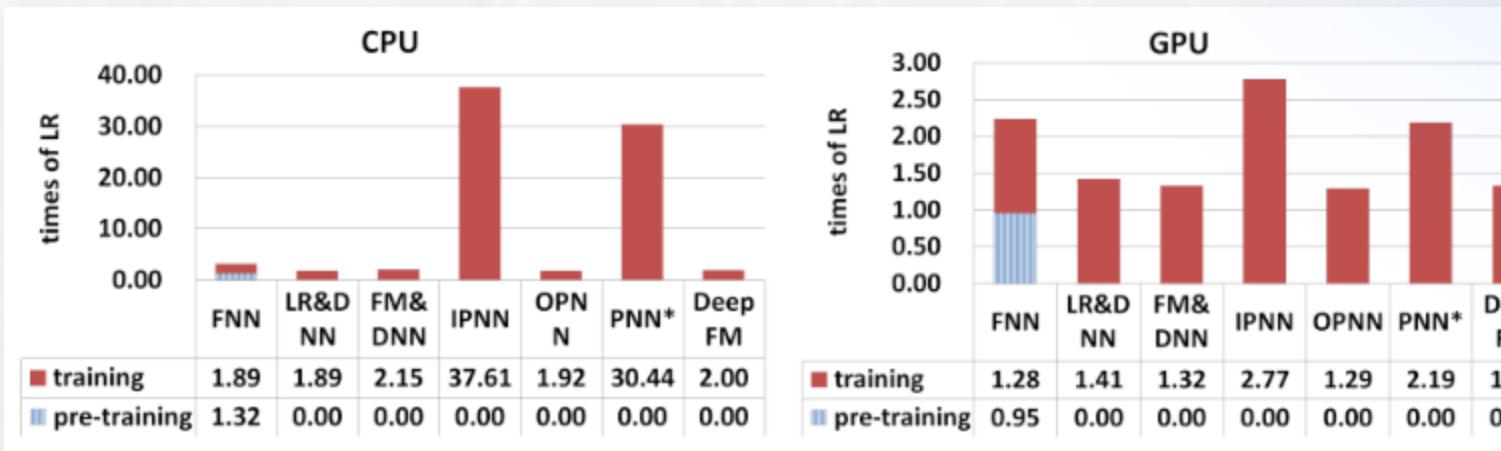
Experiment

LR (feature interaction X), FM, FNN, PNN, Wide & Deep
(Wide 부분을 LR로 사용한 모델과 FM으로 사용한 모델 두 가지), DeepFM

Experimental Results

Performance Evaluation

1. Efficiency 비교



- DeepFM의 학습 시간이 뛰어나게 좋은 것은 아니지만, Pre-training을 필요로 하지 않는다는 장점이 있음

2. Effectiveness 비교

	Company*		Criteo	
	AUC	LogLoss	AUC	LogLoss
LR	0.8640	0.02648	0.7686	0.47762
FM	0.8678	0.02633	0.7892	0.46077
FNN	0.8683	0.02629	0.7963	0.45738
IPNN	0.8664	0.02637	0.7972	0.45323
OPNN	0.8658	0.02641	0.7982	0.45256
PNN*	0.8672	0.02636	0.7987	0.45214
LR & DNN	0.8673	0.02634	0.7981	0.46772
FM & DNN	0.8661	0.02640	0.7850	0.45382
DeepFM	0.8715	0.02618	0.8007	0.45083

- Interaction을 고려하지 않는 LR모델이 Interaction을 고려하는 모델에 비해 성능이 떨어짐
- High-order를 고려하는 모델들이 Low-order Interaction만을 고려하는 모델 (LR, FM) 보다 성능이 높음
- 임베딩을 공유하는 DeepFM의 성능이 가장 높음

Conclusions

“DeepFM은 FM component와 Deep component를 결합하여 학습한다”



Pre-training이 필요하지 않음



Low-order feature interaction과 High-order feature interaction을 모두 고려할 수 있음



FM Component와 Deep Component가 Embedding을 공유하여 Feature Engineering이 필요하지 않음

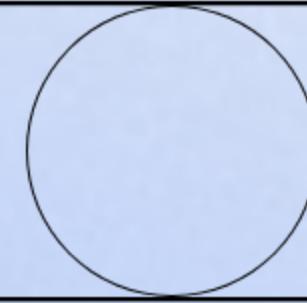
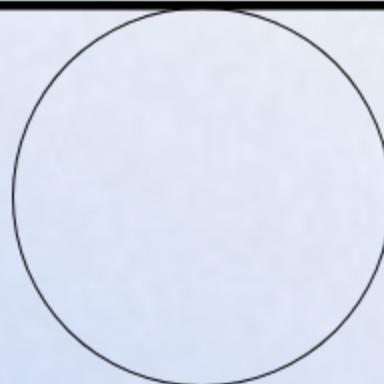
effectiveness 측면이나 efficiency 측면을 다른 SOTA 모델과 비교한 결과,

- 1) DeepFM이 AUC와 Logloss 지표에서 SOTA모델을 앞질렀음
- 2) 제일 efficient한 모델과 비교할 만큼 성능이 좋았음

Recommender System:

PART 3

BERT FOR EMBEDDING



10 jobs -당신에게 드리는 10가지 직업 정보

광고학전공 박지훈

Deep Learning



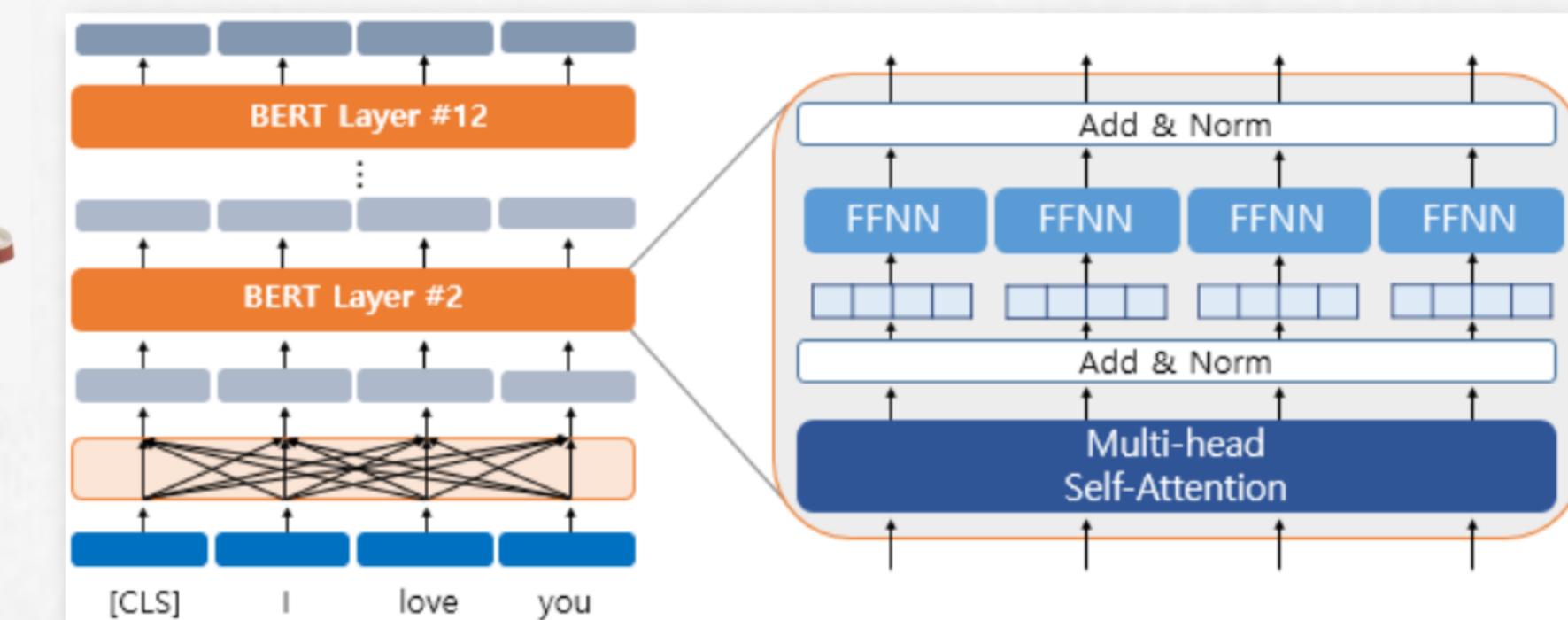
Intro & Related Work

What is BERT?



BERT (Bidirectional Encoder Representations of Transformers)

트랜스포머의 인코더 부분을 가져온
Language Model로 양방향 사전 학습과
셀프 어텐션을 한다는 것이 특징이다



Intro & Related Work

Bidirectional Train



28. Journeys are the midwives of thought. Few places are more conducive to internal conversations than a moving plane, ship, or train. There is an almost peculiar correlation between what is in front of our eyes and the thoughts we are able to have in our heads: large thoughts at times requiring large views, new thoughts new places. Introspective reflections which are liable to stall are helped along by the flow of the landscape. The mind _____ when thinking is all it is supposed to do. The task can be as paralyzing as having to tell a joke or mimic an accent on demand. Thinking improves when parts of the mind are given other tasks, are charged with listening to music or following a line of trees.

- ① may be reluctant to think properly
- ② may focus better on future thoughts
- ③ can become confused by multitasking
- ④ is likely to be paralyzed by fear of new tasks
- ⑤ can be distracted from what is before the eyes

[Bidirection = 빈칸넣기]

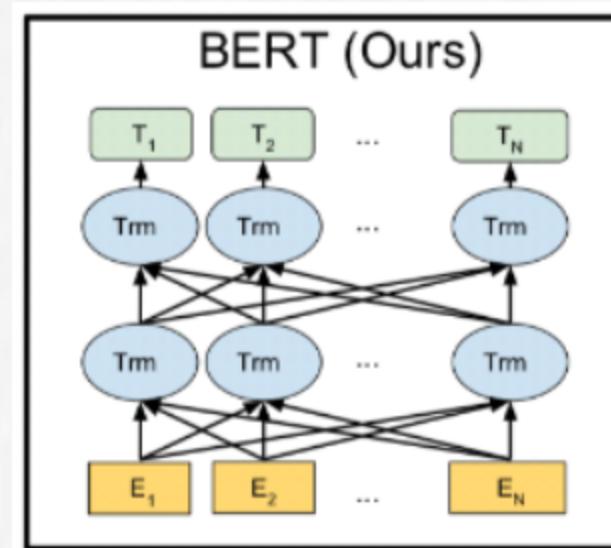
Main Idea

뒤에 나올 내용에 대한 정보를 알고 있다면
더 정확한 답을 추론할 수 있지 않을까?

Intro & Related Work

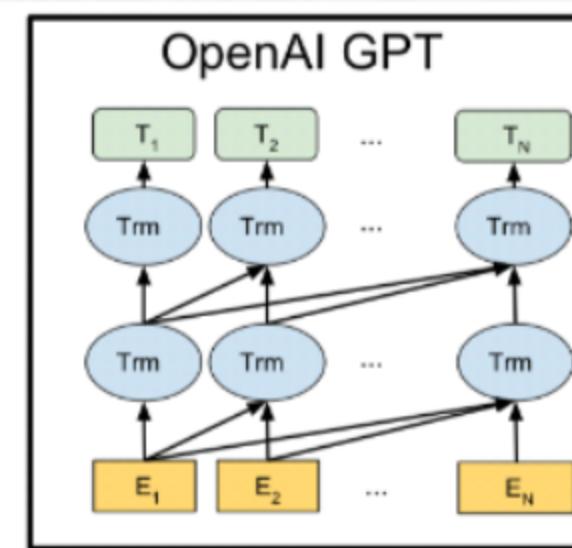
Previous Work

● ● ● ● ●



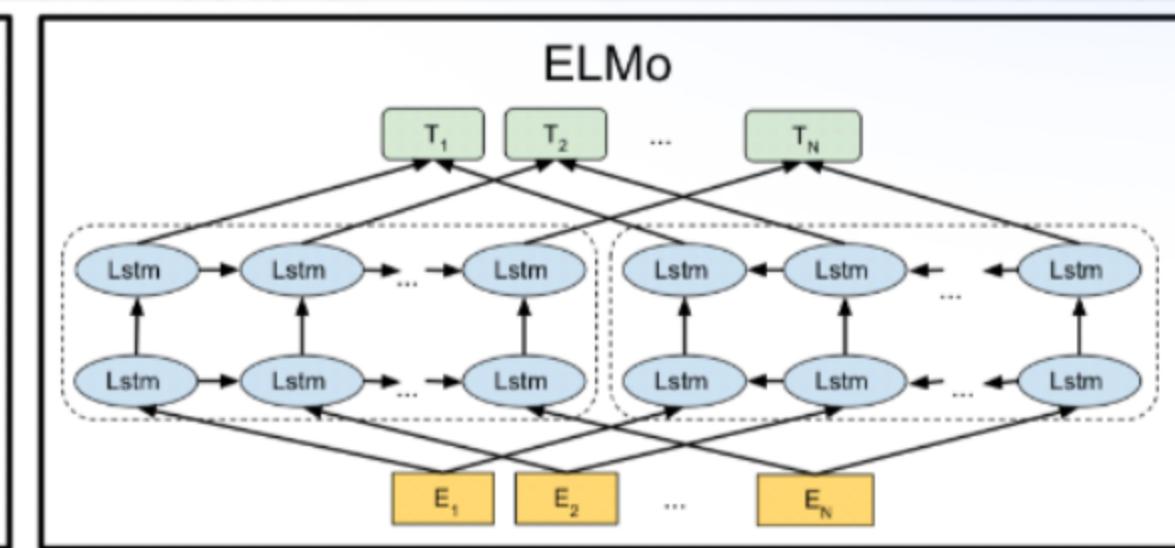
Bidirection

Transformer



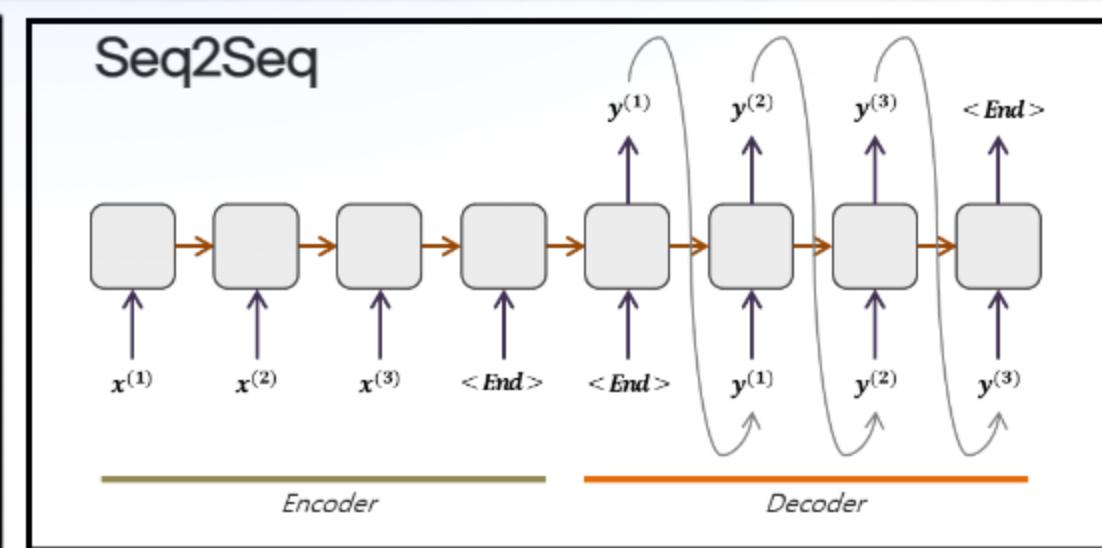
Unidirection

Transformer



Bidirection but independent

based on RNN

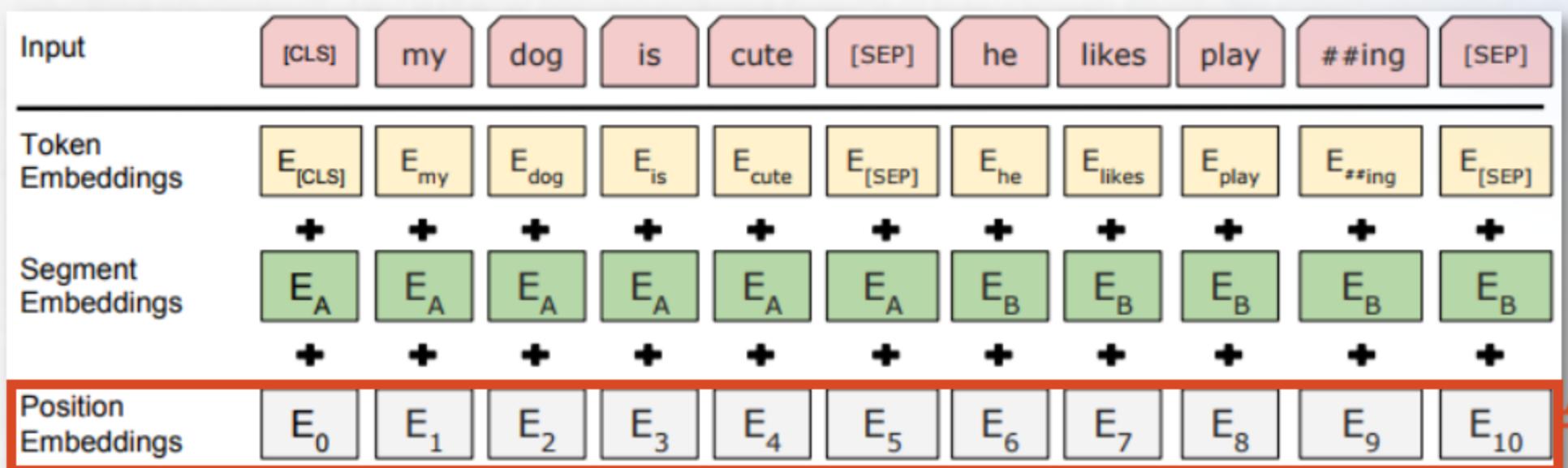


Unidirection

based on RNN

Methodology

Embedding



*Segment Embedding : 두 문장을 구분하기 위한 임베딩
ex. 어떤 토큰이 어떤 문장에 속하는지?

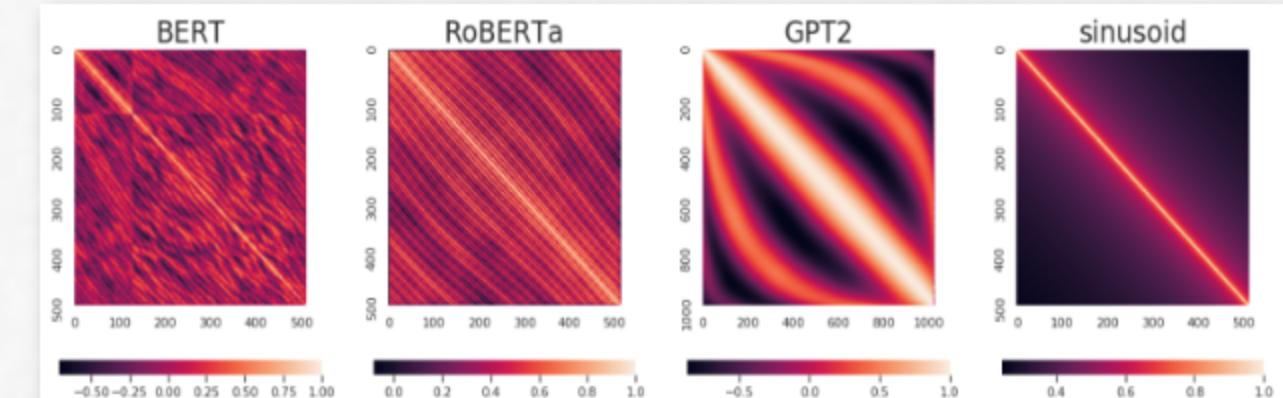
> 각 Token과 Segment, Position 벡터가 모두 더해진 상태로 들어간다

$$x_i = t_i + s_i + p_i$$

<Transformer: Fixed position encoding>

$$PE_{(pos,2i)} = \sin(pos/10000^{2i/d_{\text{model}}})$$

$$PE_{(pos,2i+1)} = \cos(pos/10000^{2i/d_{\text{model}}})$$



<Learnable position embedding>

Fixed position

홍대입구는
북위 37° 33' 25.0" 동경...

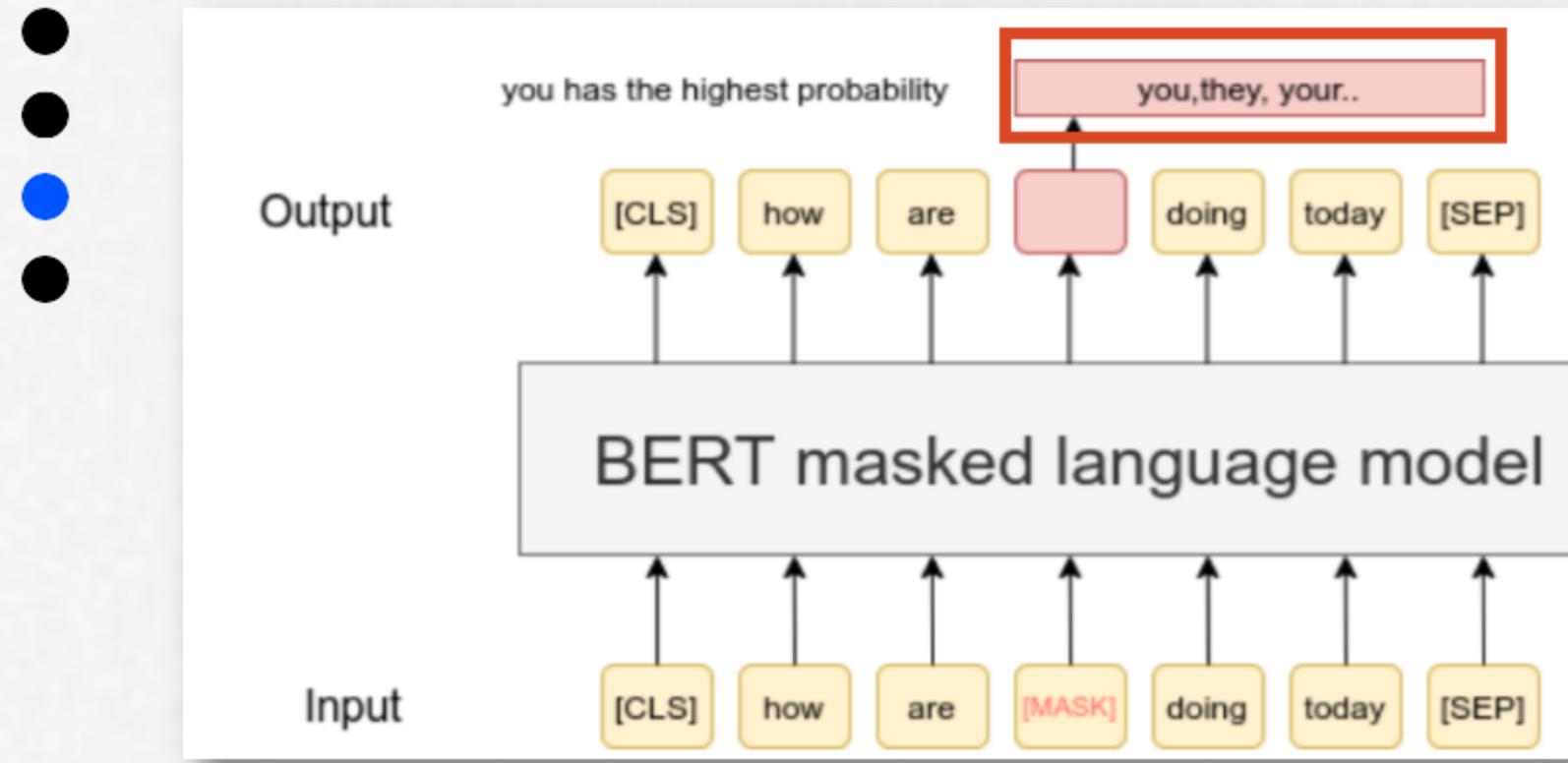
Learnable position

홍대입구는
KFC 앞에 있어요

Methodology

Pretraining

Masked Language Modeling: Cloze Task



Mask Token을 Input으로 넣은 후
실제 단어를 맞추며 주변 단어와의 Context를 학습한다
*이때 Mask Token의 개수는 제한이 없으나
Hyper Parameter로 관리해야 한다

■ MLM training은 Sequential training보다
더 풍부한 학습샘플을 만들 수 있다

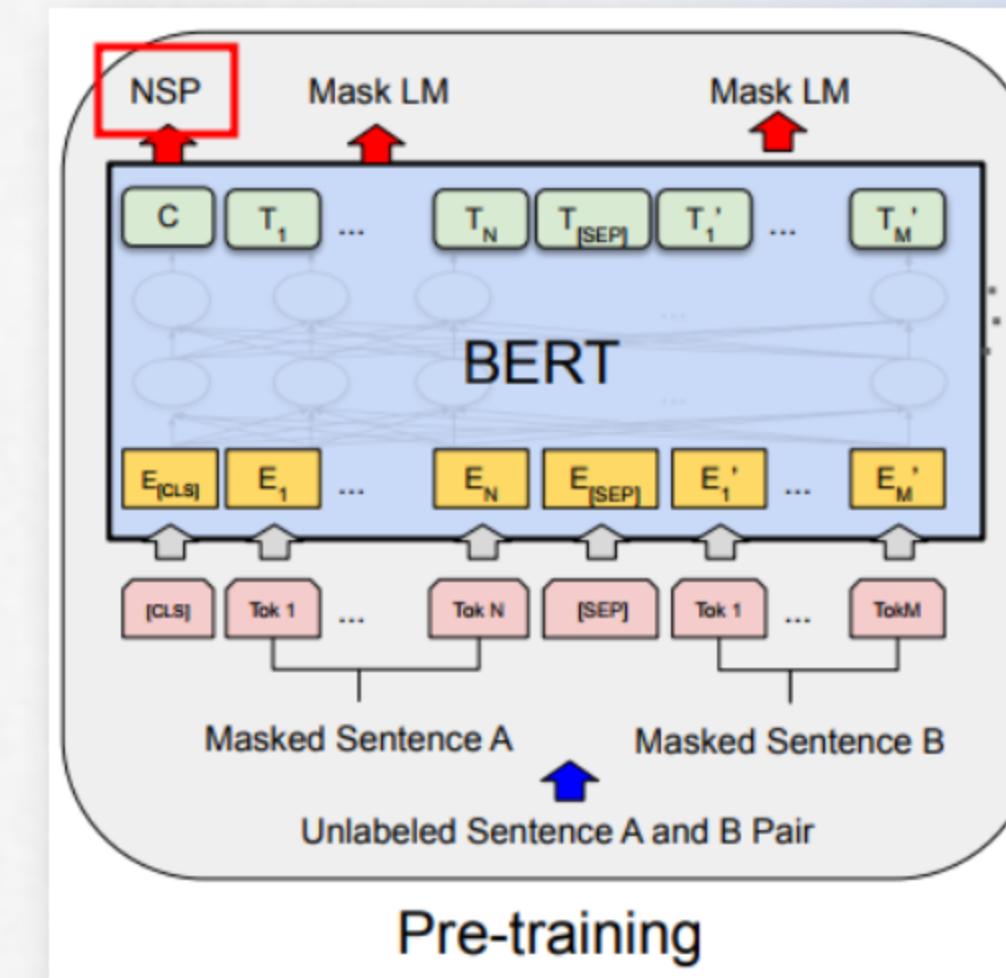
Sequential train : 1 epoch 동안 최대 N개의 샘플 학습

BERT : 1 epoch 동안 n 개 샘플 중 k개 샘플 학습 $\binom{n}{k}$

Methodology

Pretraining

Next-Sentence-Prediction(NSP)

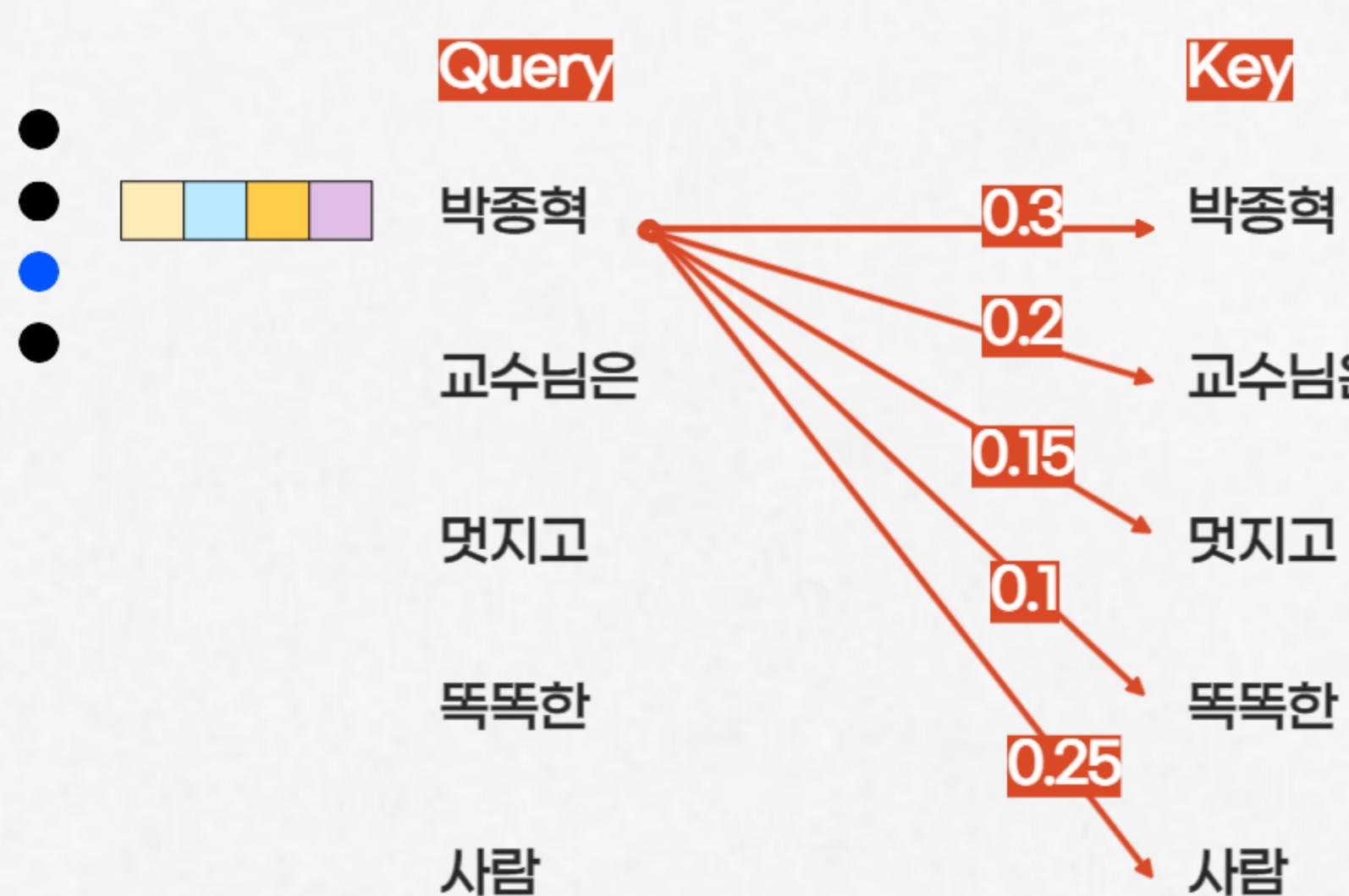


[두 문장이 서로 이어지는 문장인가?]

문장 사이의 이어지는 종속성을 학습하기 위해
CLS 토큰을 통해 Input으로 들어간 두 개의 문장이
서로 이어지는 문장인지 Binary Classification을 수행함

Methodology

Self-Attention



X Value

$$\text{softmax} \left(\frac{Q \times K^T}{\sqrt{d_k}} \right) V = Z$$

<self-attention formula>



자기 자신을 포함한 다른 단어의 정보를 포함한 벡터로 표현한다
= 문맥을 가진 Contextual Vector이다

Experiments

Downstream task

[General Language Understanding Evaluation]

System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	Average
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
BERT _{BASE}	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
BERT _{LARGE}	86.7/85.9	72.1	92.7	94.9	60.5	86.5	89.3	70.1	82.1

Top Leaderboard Systems (Dec 10th, 2018)				
System	Dev	Test	EM	F1
Human				
#1 Ensemble - nlnet	-	-	82.3	91.2
#2 Ensemble - QANet	-	-	86.0	91.7
Published				
BiDAF+ELMo (Single)	-	85.6	-	85.8
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
BERT _{BASE} (Single)	80.8	88.5	-	-
BERT _{LARGE} (Single)	84.1	90.9	-	-
BERT _{LARGE} (Ensemble)	85.8	91.8	-	-
BERT _{LARGE} (Sgl.+TriviaQA)	84.2	91.1	85.1	91.8
BERT _{LARGE} (Ens.+TriviaQA)	86.2	92.2	87.4	93.2

System	Dev	Test
ESIM+GloVe	51.9	52.7
ESIM+ELMo	59.1	59.2
OpenAI GPT	-	78.0
BERT _{BASE}	81.6	-
BERT _{LARGE}	86.6	86.3
Human (expert) [†]	-	85.0
Human (5 annotations) [†]	-	88.0

[SQuAD v1.1 / v2.0]

[SWAG]

GLUE, SQuAD, SWAG 등 자연어 처리 Downstream task에서 GPT, ELMO 등을 앞지르며 SOTA 달성 특히 SQuAD v1.1의 질문-응답쌍의 독해 테스트에서는 인간의 능력을 크게 뛰어넘음

Top Leaderboard Systems (Dec 10th, 2018)				
System	Dev	Test	EM	F1
Human				
#1 Single - MIR-MRC (F-Net)	-	-	74.8	78.0
#2 Single - nlnet	-	-	74.2	77.1
Published				
unet (Ensemble)	-	-	71.4	74.9
SLQA+ (Single)	-	-	71.4	74.4
Ours				
BERT _{LARGE} (Single)	78.7	81.9	80.0	83.1

[Parameter of BERT]

	Layer_Num	Hidden state	Attention head_Num	Total Parameter
BERT _{BASE}	12	768	12	110M
BERT _{LARGE}	24	1024	16	340M

Conclusion & Limitation

[Conclusion]

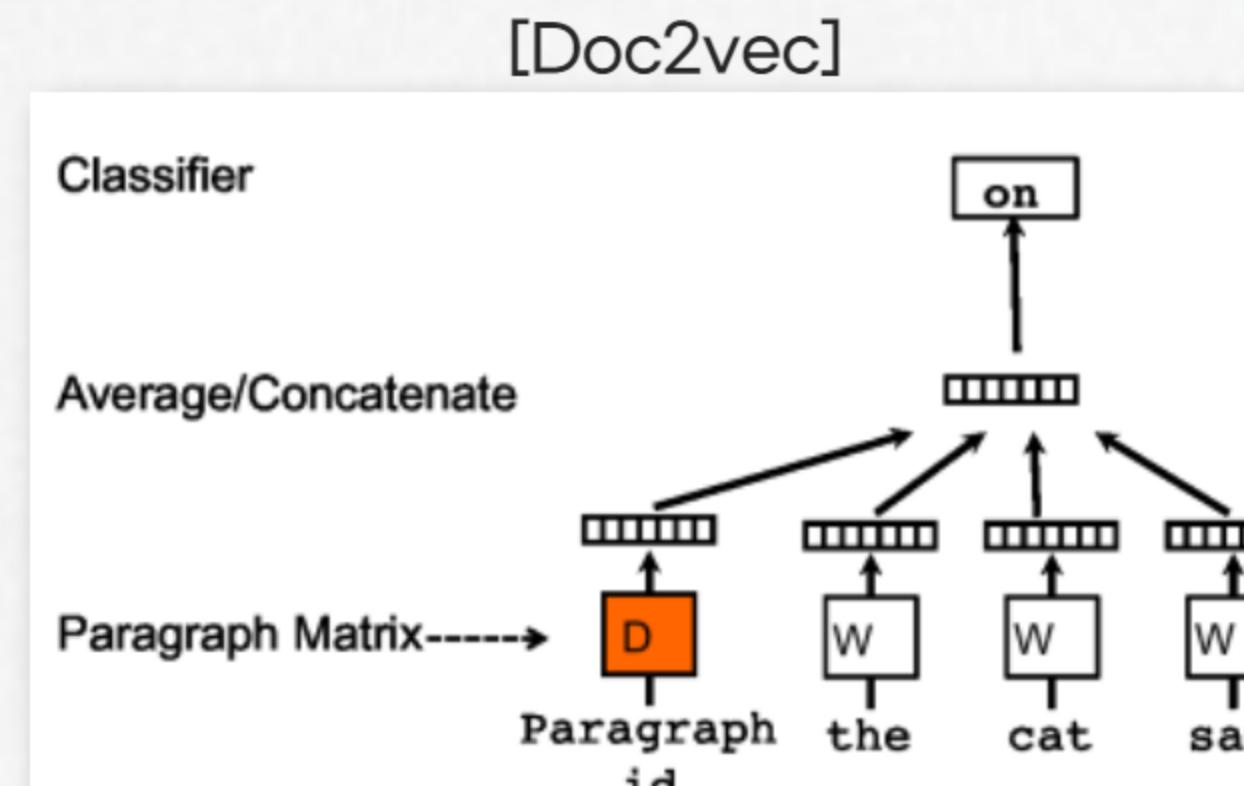
- End to End 모델보다 거대한 Corpus로 사전학습한 거대한 모델이 성능 향상에 더 유의미하다
● -> 데이터와 자본이 많은 기업이 대부분의 연구 성과를 가져갈 것이다
- 문장 전체를 contextualize할 수 있는 양방향 학습 방법인 MLM(Masked Language Model)이 유의미했다
●

[Limitation]

- 업데이트 할 파라미터가 굉장히 많아 모델이 커져 학습 속도가 굉장히 느리다는 단점이 있다
*BASE, LARGE 모델 모두 4일 소요

Application

Content-based filtering



각 기업의 채용공고(문서)를 임베딩할 수 있다

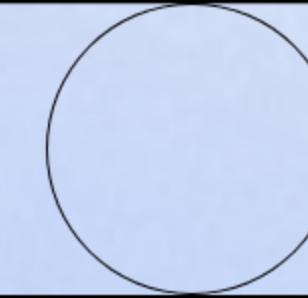
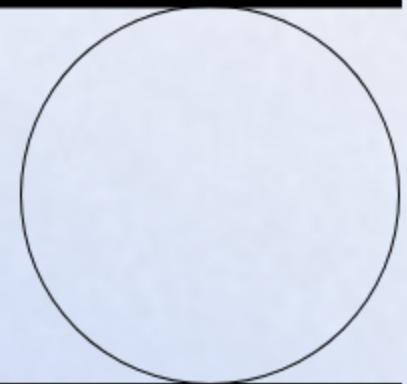
유사한 아이템을 추천해주는 것이 주요 Task인 Content-based filtering에서 가지고 있는 아이템인 채용공고 문서를 하나의 벡터 공간에 Embedding을 시켜야 한다

Doc2vec과 같이 문서에 ID token을 부착하여 BERT기반으로 학습한다면 더 정교한 Embedding이 가능할 것이다

Recommender Syetem:

PART 4

HYBRID RECOMMENDER FOR PERFORMANCE



10 jobs -당신에게 드리는 10가지 직업 정보

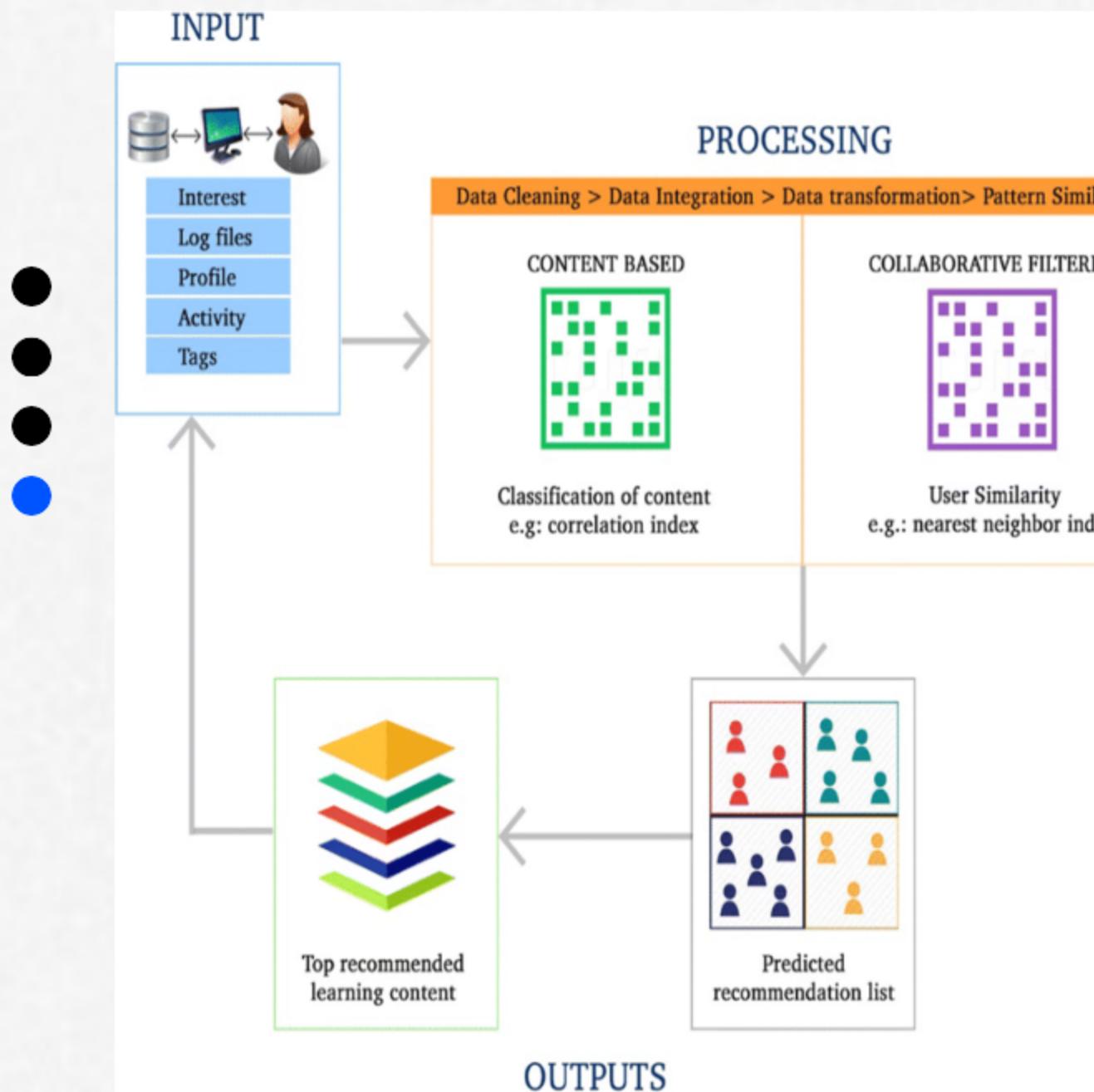
행정학전공 황건하

Deep Learning



Intro & Related Work

What is Hybrid Recommender Systems?



Hybrid RC (Hybrid Recommender Systems)

2개 이상의 다양한 종류의 추천시스템
알고리즘을 서로 다른 방식으로 결합하여
상호 보완적인 이점을 활용하는 것이 특징

Hybrid Recommender Systems - A Systematic Literature Review (2019)

2009년부터 2019년까지의 약 10년 간의
하이브리드 추천 시스템의 최신 기술을 제시

Hybrid RS를 다루는 학술지 및 컨퍼런스에서
76개의 주요 연구를 분석

Intro & Related Work

Types of Hybrid Recommender Systems

1. CF (Collaborative Filtering)

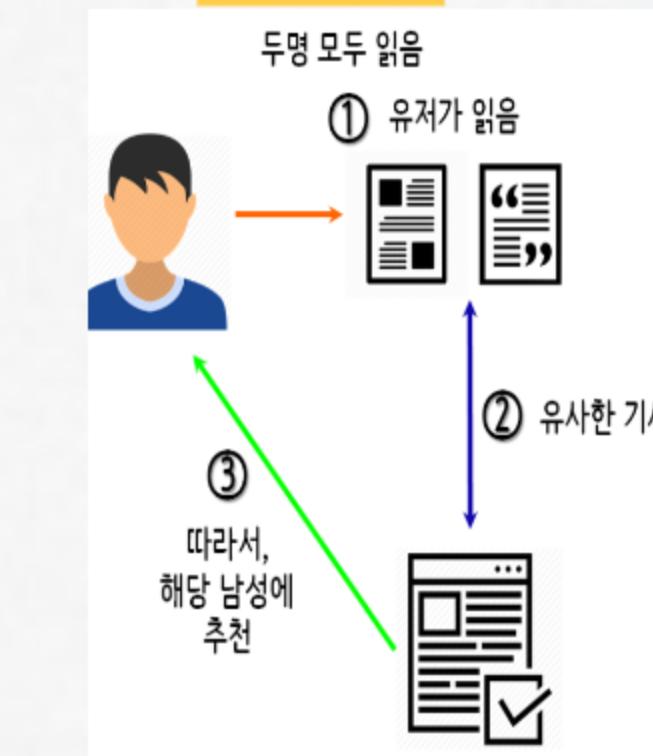


- 과거에 **비슷한 취향**을 가졌던 사람들이 미래에도 **비슷한 취향**을 가질 것

- User-Item에 초점

<https://bitnine.tistory.com/380>

2. CBF (Content-Based Filtering)



- 과거에 **특정 속성을 가진** 아이템을 좋아했던 사람들이 미래에도 **같은 종류의** 아이템을 좋아할 것

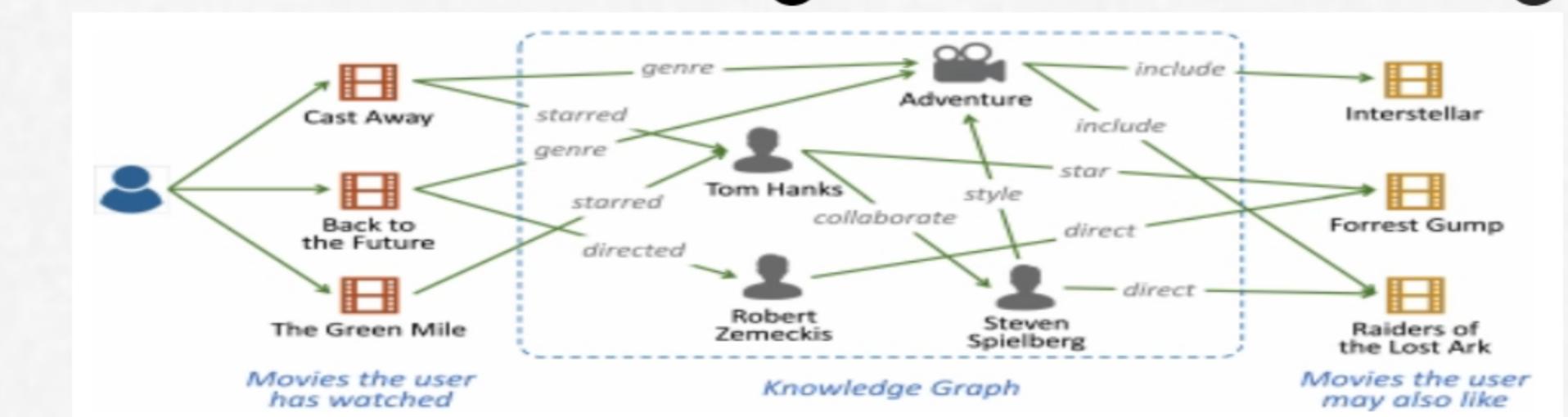
- Item-Attribute에 초점

<https://bitnine.tistory.com/380>

3. DF (Demographic Filtering)



4. KBF (Knowledge-Based Filtering)



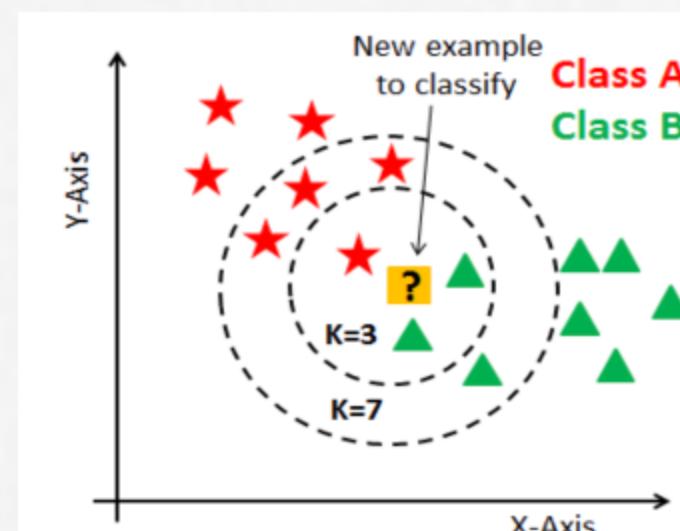
https://zzeabok.github.io/knowledge_graph/recommender_system/KG_recommend/

Methodology

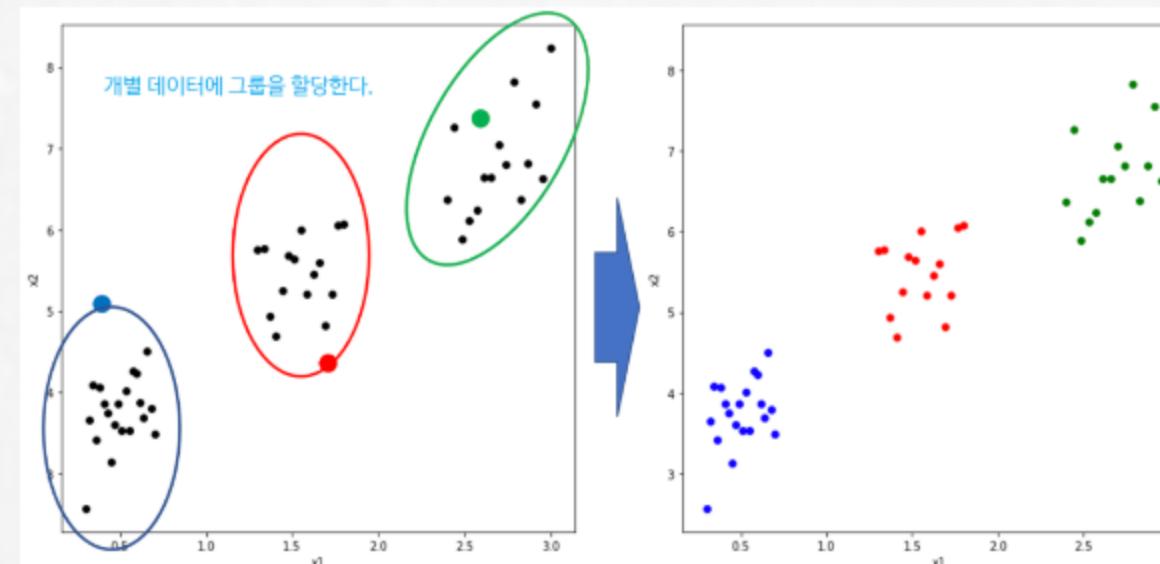
Data mining and machine learning techniques

어떤 데이터 마이닝 및 머신러닝 기술이 사용될까?

1. K-NN



2. Clustering

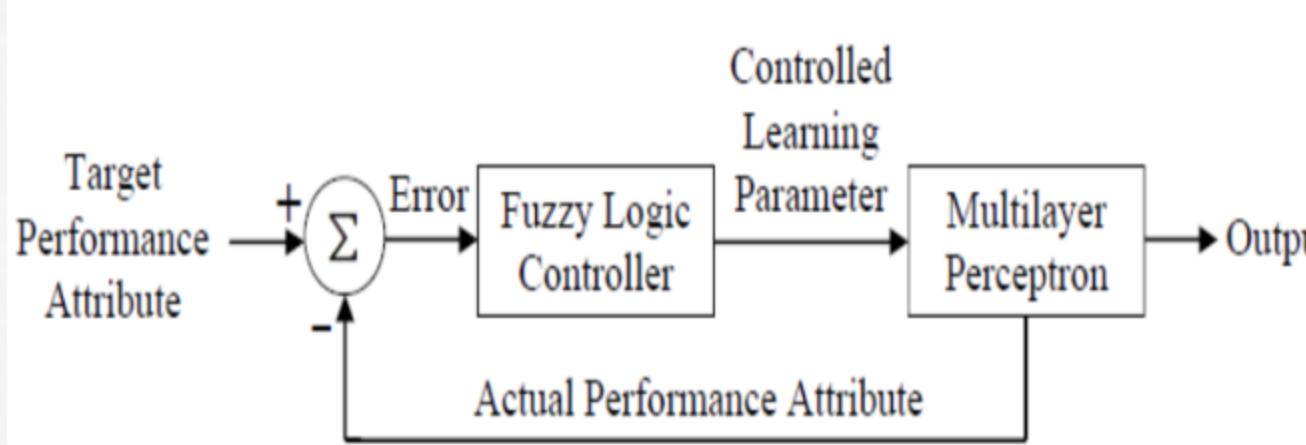


3. Association Rules

$$\text{Support} = \frac{\text{frq}(X, Y)}{N}$$

$$\text{Rule: } X \Rightarrow Y \quad \text{Confidence} = \frac{\text{frq}(X, Y)}{\text{frq}(X)}$$

$$\text{Lift} = \frac{\text{Support}}{\text{Supp}(X) \times \text{Supp}(Y)}$$



4. Fuzzy logic

DM/ML technique

K-NN

59

Clustering

34

Association rules

17

Fuzzy logic

14

Matrix manipulation

9

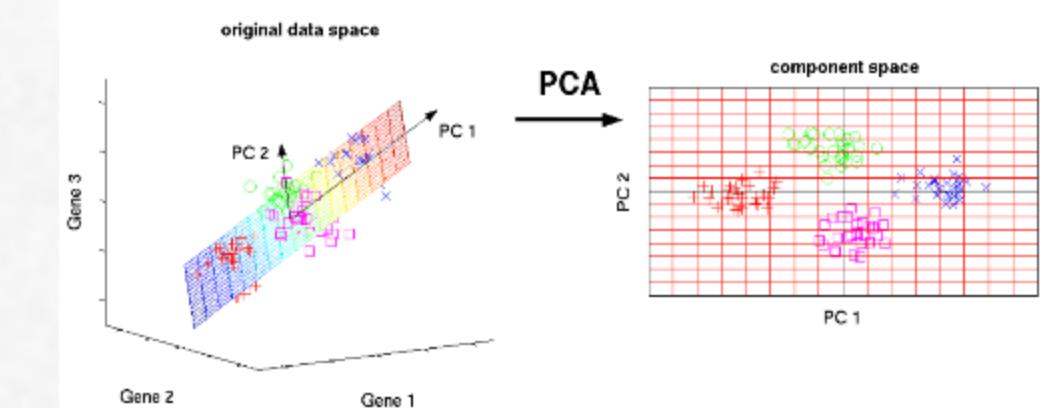
Other

19

Studies

5. Matrix manipulation

■ 특이값 분해 (SVD), 주성분 분석 (PCA), LDA 등



Methodology

Recommendation technique combinations

어떤 추천 기법이 결합되며 어떤 문제를 해결할까?

1. CF-X

- 여기서 X란 KBF, DF, DM/ML 기법
- CF + Clustering
- CF + DF
- CF + KDF

2. CF-CBF

- 가장 인기 있는 Hybrid RS
- CF와 CBF의 추천에 가중치를 부여
- 베이지안 기반 CF-CBF 제안
 - > 각 추천 전략의 가중치가 자동으로 선택되어 문제의 특정 조건에 맞게 모델 조정 가능
 - > 다양한 도메인에 적용 가능

3. CF-CBF-X

- 소셜 네트워크(광고)에서 특히 유용
- 데이터가 부족한 상황에서 좋은 추천을 제공하는 것
- CF + CBF + Clustering
- CF + CBF + DF

4. IICF-UUCF

- Item-Item CF와 User-User CF 두 가지 형태의 CF
- 먼저, 리뷰에서 감성 분석 수행
 - > 사용자 또는 아이템 유사성 계산
- 두번째, IICF or UUCF을 사용
 - > 평점 예측

5. CBF-X

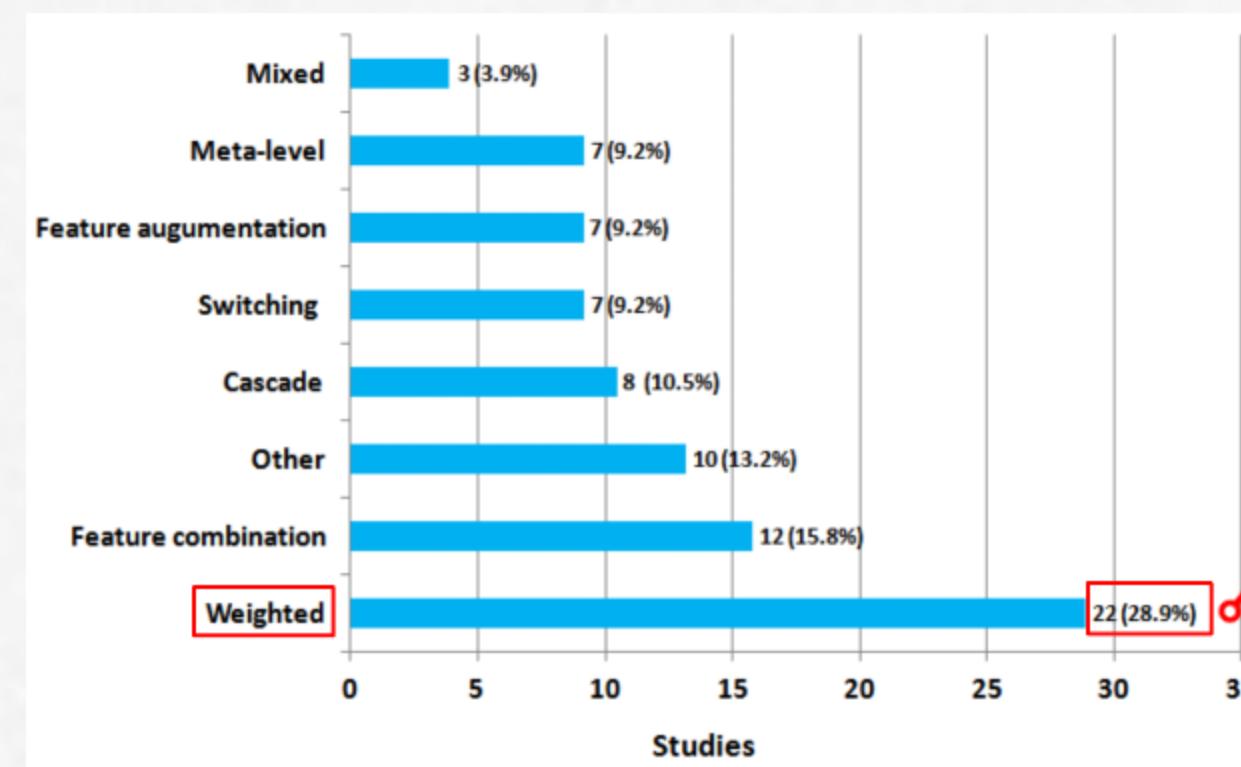
- CBF + Clustering
- CBF + DF
- CBF + KDF

Problem	CF-X	CF-CBF	Hybrid recommenders and studies			
			CF-CBF-X	IICF-UUCF	CBF-X	Other
Cold-start	2	3	2	1	1	5
Data Sparsity	0	5	3	3	4	6
Accuracy	2	3	0	2	2	4
Scalability	0	2	2	0	2	2
Diversity	2	0	0	0	0	1
Other	0	2	1	1	1	2
Total	6	15	8	7	10	20

Methodology

Classes of hybridization

Burke의 분류법에 따라 어떤 하이브리드화 클래스가 사용될까?



1. Weighted

■ 가중치 선형 함수를 사용하여 각 추전기법의 출력점수를 합산하여 추천 항목의 점수를 계산

Classifier 1

A : 0.7
B : 0.3

Classifier 2

A : 0.1
B : 0.9

Classifier 3

A : 0.6
B : 0.4

각 모델의 성능에 따라 **가중치**를 부여

$$A = \frac{0.7 \times 2 + 0.1 \times 1 + 0.6 \times 1}{4} = 0.53$$

$$B = \frac{0.3 \times 2 + 0.9 \times 1 + 0.4 \times 1}{3} = 0.47$$

2. Feature combination

- 보유하고 있는 데이터로부터 얻을 수 있는 다양한 **feature**를 모두 조합하여 추천하는 방식

3. Cascade

- 단계적 추천 프로세스
- 한 가지 기법 사용하여 후보 항목의 **대략적인 순위**를 생성
- 두 번째 기법 사용하여 예비 후보 세트에서 **목록 구체화**

4. Switching

- 플랫폼 내 특정 상황을 고려하여 여러 추천 알고리즘의 추천 결과를 **선택적으로** 보여주는 방식

5. Feature Augmentation

- 결합된 추천 기술 중 하나를 사용하여 항목 예측 또는 분류를 생성한 뒤 **다른 추천 기술의 작동에 구성**

6. Meta Level

- 첫번째 모델이 **다음 모델의 Input**이 되면서 **서로가 서로의 정보를 학습해** 추천하는 방식

7. Mixed

- 추천 알고리즘의 추천 결과를 모두 보여주는 방식
- 많이 사용X

Methodology

Characteristics and Metrics

어떤 추천시스템 특성을 평가하고 어떤 지표를 사용할까?

Recommendation characteristic	Studies
Accuracy	62
User satisfaction	10
Diversity	7
Computational complexity	6
Novelty-Serendipity	4

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

■ 가장 자주 사용되는 지표는
Precision, Recall, MAE

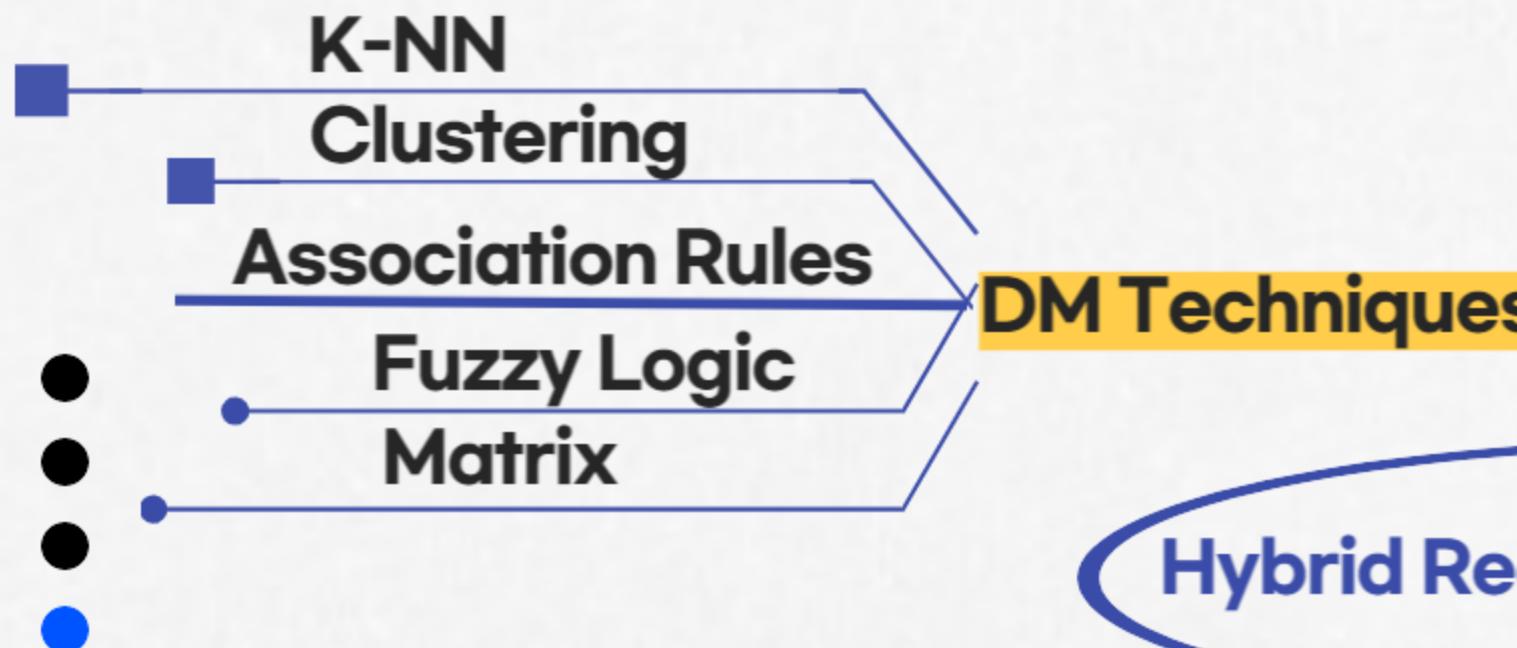
$$MAE = \frac{1}{n} \sum |\hat{y} - y|$$



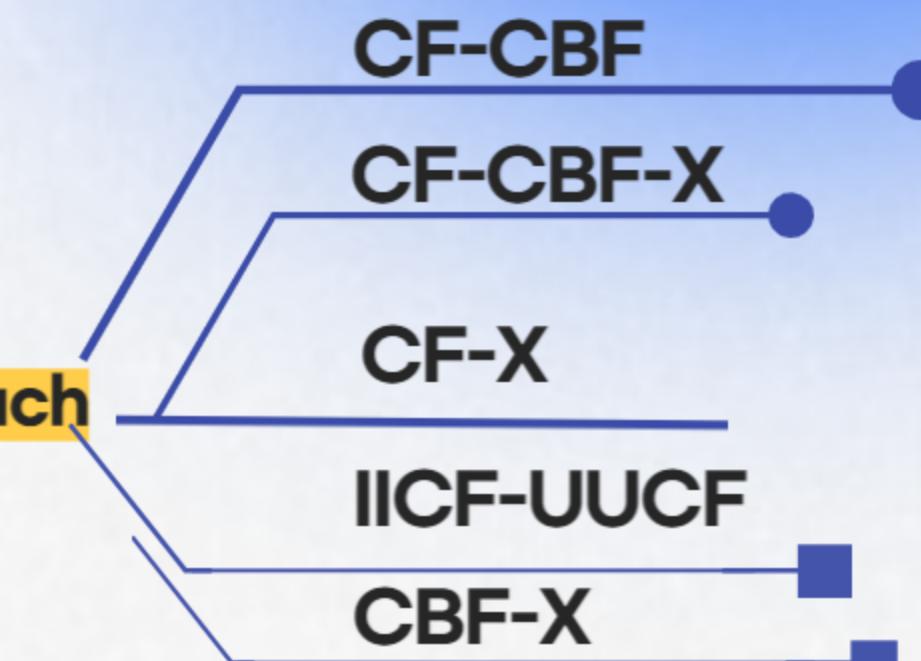
Characteristic	Metrics	Studies
Accuracy	Precision	31
	MAE	27
	Recall	23
	F1	14
	RMSE	6
	Other	15
User satisfaction	Qualitative Subjective Assessment	10
Diversity	Coverage	4
	Ranking distances	3
Complexity	Execution time	6
Novelty-Serendipity	Surprisal	2
	Coverage in Long-Tail	1
	Expected Popularity Complement	1

Results

Organize

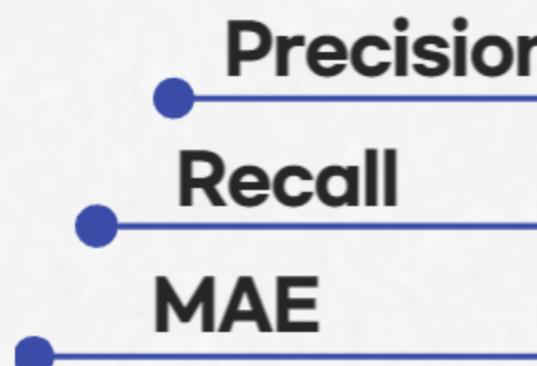


Recommendation Approach



Evaluated Characteristics

Accuracy



Hybrid Class

Weighted

Switching

Feature Combination

Cascade

Feature Augmentation

Meta-Level

Mixed

Conclusion & Contribution & Limitation

[Conclusion]

- 데이터 마이닝 및 머신러닝 기술, 추천 전략, 하이브리드화 클래스, 평가 프로세스 등에 대해서 분석
 - > 데이터 마이닝 기법 중에서는 **K-NN**, **Clustering**이 가장 많이 사용
 - > 추천 단일 기법 중에서는 **CF**, 복합 기법 중에서는 **CF-CBF** 가장 많이 사용
 - > 하이브리드화 클래스 중 **가중치** 하이브리드가 가장 많이 사용
 - > 가장 많이 사용되는 지표는 **Precision**, **Recall**, **MAE**

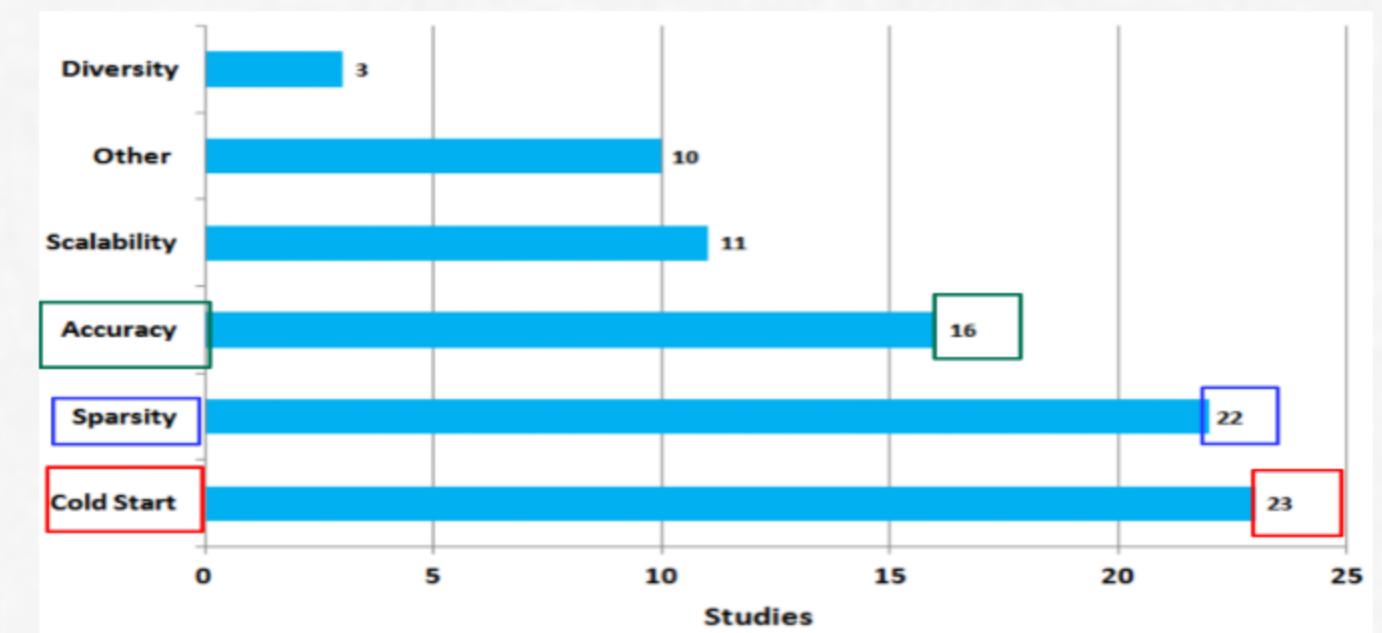
-
-
-
-

[Contribution]

- 빅데이터 시대에 Hybrid 병렬 알고리즘을 통해 더 큰 데이터셋을 처리 가능 (확장성 문제 완화)
- Cross Domain Recommenders 구축 or 기존 기술의 계산 복잡성 개선

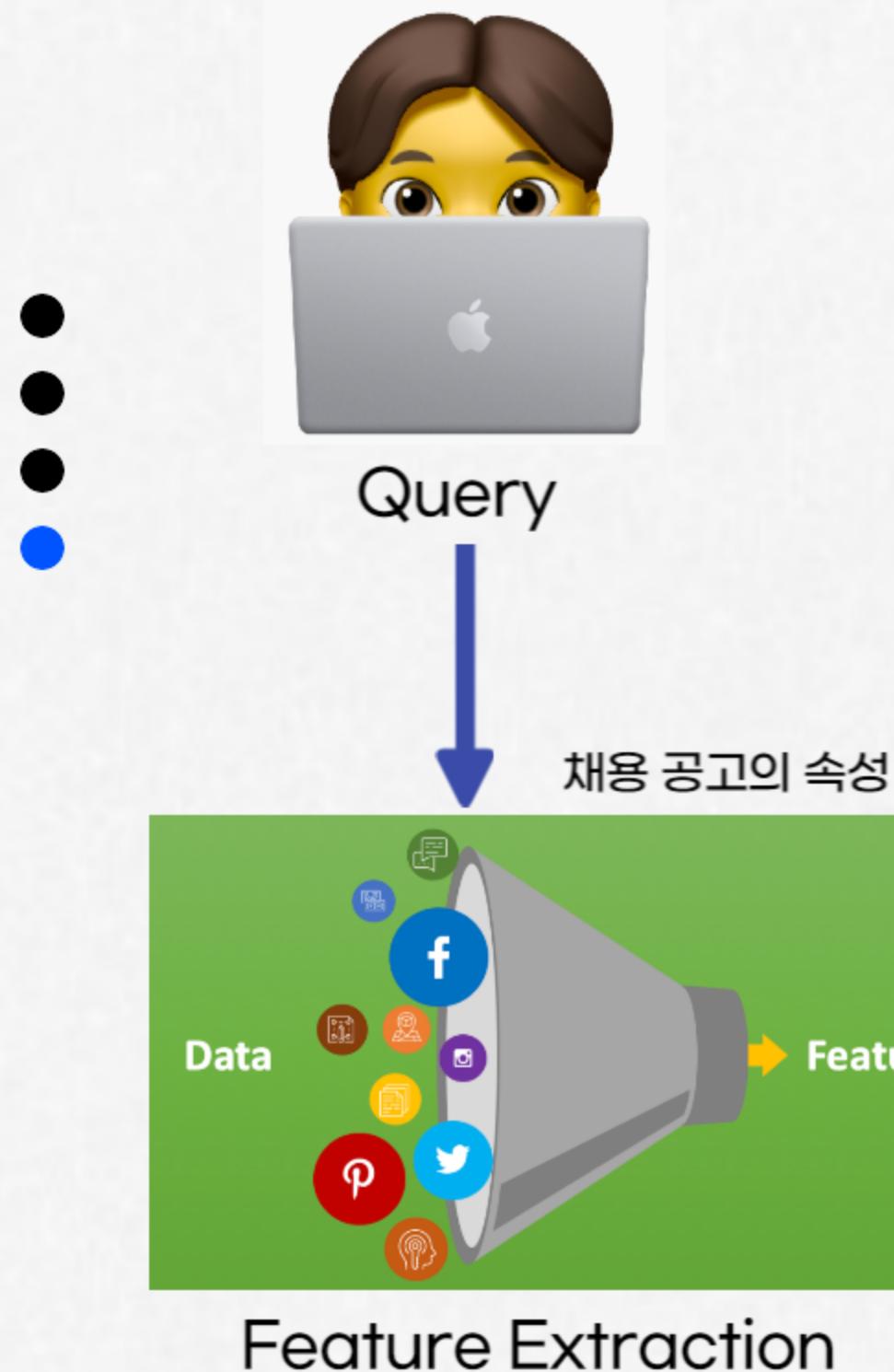
[Limitation]

- **Cold-start**(신규 사용자), **데이터 희소성**, **정확성**이 가장 큰 문제점

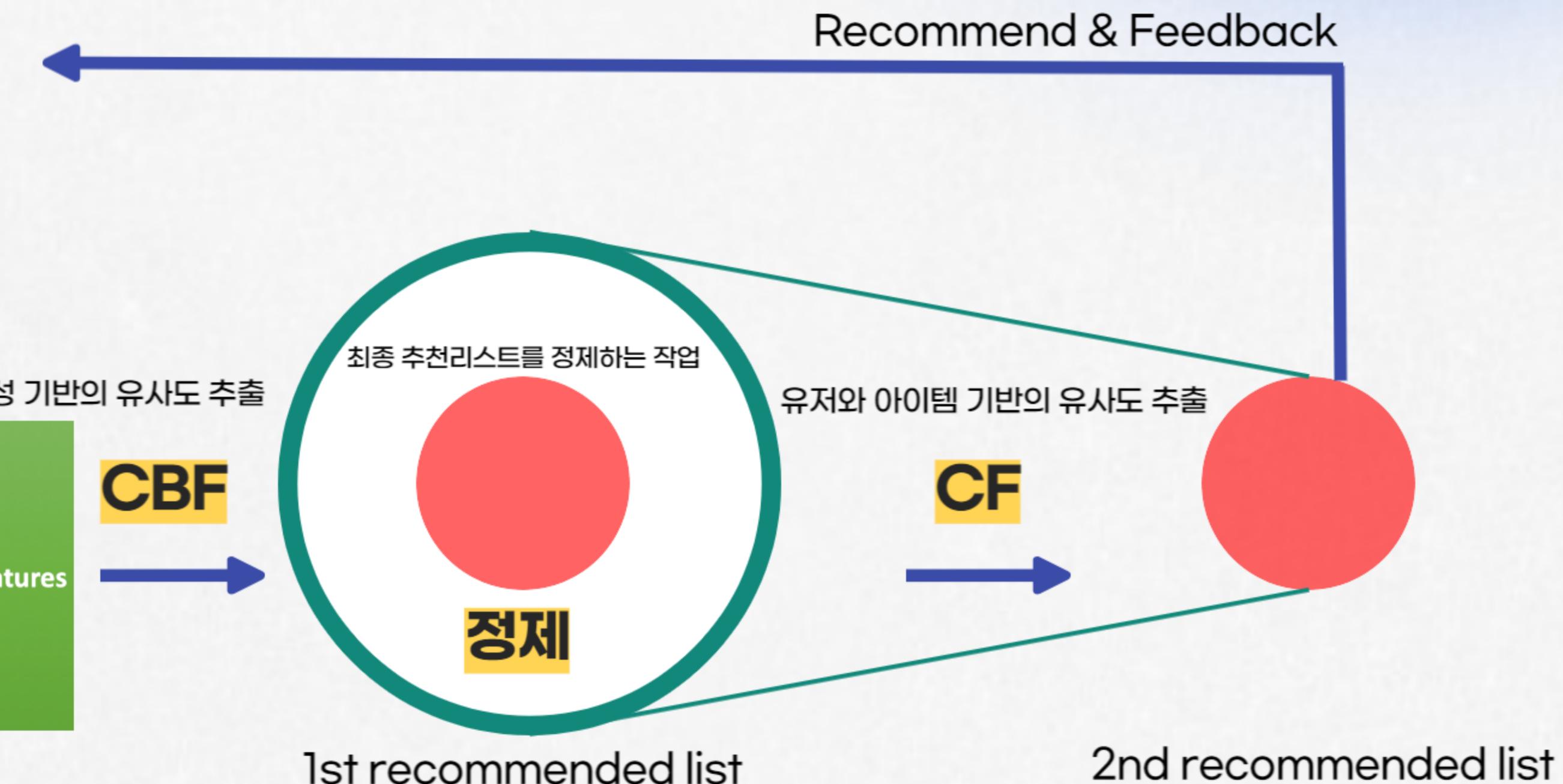


Application

Hybrid Recommender Systems



CF-CBF → 유저의 개인화를 위해 CF 모델 이용
→ CBF 모델 사용하여 Cold-start에서 자유로울 수 있음



References

PART1

- [1] Hyo Jin Lee, Yoonsuh Jung " Comparison of deep learning-based autoencoders for recommender systems" The Korean Journal of Applied Statistics No. 3, 329-345 (2021)
- [2] Dawen Liang, Matthew D.Hoffman, Rahul G.Krishnan, Tony Jebara "Variational Autoencoders for Collaborative Filtering" arXiv:1802.05814v1 [stat.ML] 16 Feb 2018.

PART2

- [1] Heng-Tze Cheng, Levent Koc, et al. "Wide & Deep Learning for Recommender Systems" arXiv preprint arXiv:1606.07792 (2016).
- [2] Steffen Rendle. "Factorization Machines" , Department of Reasoning for Intelligence, The Institute of Scientific and Industrial Research (2010).
- [3] Huifeng Guo, Ruiming Tang, et al. "DeepFM: A Factorization-Machine based Neural Network for CTR Prediction" arXiv preprint arXiv:1703.04247(2017).

PART3

- [1] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).
- [2] Sun, Fei, et al. "BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer." Proceedings of the 28th ACM international conference on information and knowledge management. 2019.
- [3] Warstadt, Alex, Amanpreet Singh, and Samuel R. Bowman. "Neural network acceptability judgments." Transactions of the Association for Computational Linguistics 7 (2019): 625-641. GLUE
- [4] Le, Quoc, and Tomas Mikolov. "Distributed representations of sentences and documents." International conference on machine learning. PMLR, 2014.

PART4

- [1] Erion Çano1, Maurizio Morisio et al. "Hybrid Recommender Systems-A Systematic Literature Review." (2019).
- [2] Software carpentry, <https://statkcleee.github.io/parallel-r/recommendation-sys.html>
- [3] https://zzaebok.github.io/knowledge_graph/recommender_system/KG_recommend/
- [4] 차루 아가르왈(2022), '기초부터 실무까지 머신러닝 추천 시스템 교과서', 에이콘 출판사, pp.185-216.