

Deep Neural Networks for YouTube Recommendations

Paul Covington, Jay Adams, Emre Sargin

Google Mountain View

[RecSys '16: Proceedings of the 10th ACM Conference on Recommender Systems](#) September 2016

[://dl.acm.org/doi/pdf/10.1145/2959100.2959190?utm_campaign=Weekly%20dose%20of%20Machine%20Learning&utm_medium=email&utm_source=Revue%20newsletter](https://dl.acm.org/doi/pdf/10.1145/2959100.2959190?utm_campaign=Weekly%20dose%20of%20Machine%20Learning&utm_medium=email&utm_source=Revue%20newsletter)

ABSTRACT

- 유튜브는 현재 매우 큰 규모와 복잡한 추천 시스템을 보여주고 있는 산업 중 하나
- 이 논문에서는 딥러닝을 사용하여 추천 시스템의 성능을 향상시킨 것에 대해 다루고, 2가지 기본 구조를 따르고 있음
 - 첫번째는 deep candidate generation model이며, 두번째는 separate deep ranking model임
 - 또한, 추천 시스템 설계 및 학습, 서빙 과정에서 얻게 된 현실적인 깨달음과 인사이트에 대해서도 언급

1. introduction

- 유튜브는 계속해서 올라오는 새로운 영상 콘텐츠들로부터 수억명의 사용자들에게 개인화된 콘텐츠를 제공하기 위한 미션을 항상 가지고 있음
- Sclae
 - 수많은 사용자와 영상콘텐츠가 있는 유튜브에 적용할 수 있는 알고리즘이 마땅히 없음
- Freshness
 - 유튜브에는 새로운 영상 콘텐츠들이 새롭게 무수히 많이 올라오고 있음
 - 시스템에서는 실시간 반응 요소들을 적절하게 적용 및 반영할 수 있는 방법을 찾아야 함
- Noise
 - 사용자 만족도에 대한 explicit feedback을 구하기 굉장히 힘들기 때문에 implicit feedback을 사용하게 되며, 메타데이터가 잘 구축되지 않기도 함

Explicit data vs implicit data

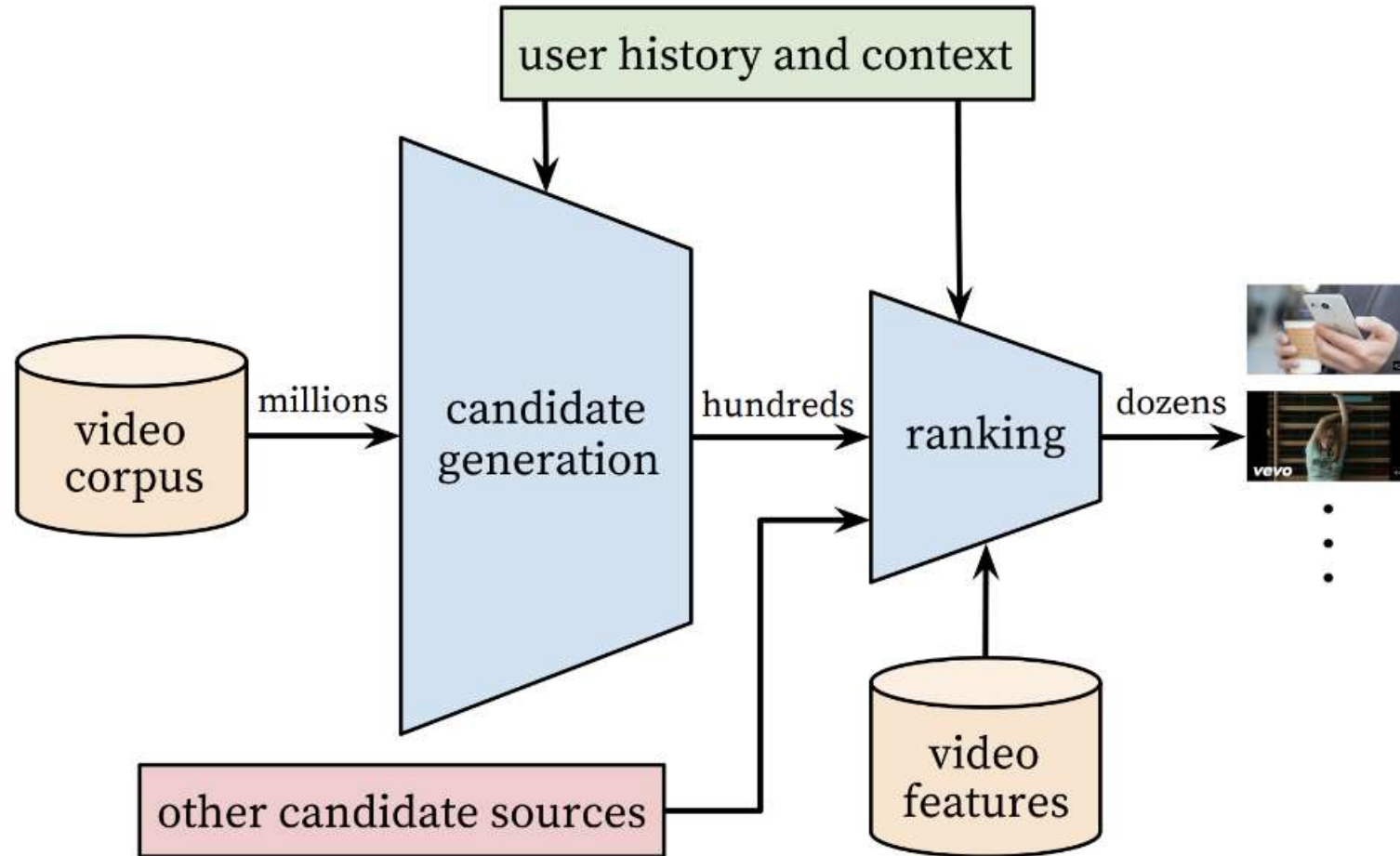
- Explicit data

- 사용자가 자신의 선호도를 직접(Explicit) 표현한 Data
- 좋아요, 별점
- 데이터를 얻기 힘들다는 단점이 있음

- Implicit data

- 사용자가 간접적(Implicit)으로 선호, 취향을 나타내는 데이터를 의미
- 검색기록, 방문페이지
- 데이터양이 많음
- Negative한 feedback이 없고 noise가 많다는 단점이 있음

2. System overview



2. System overview

- 유튜브 추천 시스템은 크게 candidate generation model과 ranking model인 2단계로 구성되어 있음
- Candidate generation
 - 유저의 시청 기록과 demographic 정보 등을 통해서 추천 후보를 millions에서 hundreds로 줄이는 과정
 - 높은 precision을 가지도록 추천 후보 구성하는 것이 목표
 - 보고 싶은 영상을 추천해주지 않는 것(FN)보다 보기 싫은 영상을 추천하지 않는 것(FP)이 더 중요함
- Ranking
 - 필터링이 된 영상들이기 때문에 recall이 중요
 - 유저와 영상에 대한 다양한 특성들이 활용 됨
 - 최종적으로 유저가 좋아할만한 영상을 얻는 단계

Precision, recall

- Precision(정밀도)
 - 분류모델이 positive라고 판단한 것 중 실제 positive인 비율
- Recall(재현율)
 - 실제 positive 샘플 중 분류모델이 positive로 판정한 비율

		Actual	
		Positive	Negative
Predicted	Positive	TP	FP
	Negative	FN	TN

Figure 3. Precision

		Actual	
		Positive	Negative
Predicted	Positive	TP	FP
	Negative	FN	TN

Figure 4. Recall

3. Candidates generation

- candidate generation, 후보 생성 단계에서는 엄청난 규모의 영상 콘텐츠에서 사용자와 관련된 몇 백개의 영상으로 후보군을 만드는 작업이 진행됨
 - 사용자(U)와 Context(C)를 기반으로 특정 시간(t)에서 수백만개의 아이템(V) 중 각 아이템(i)의 시청 class를 예측하는 multiclass classification으로 추천 문제를 정의할 수 있음
 - 사용자의 기록과 context의 역할을 하는 u 벡터를 학습으로 후보 아이템의 시청 확률을 예측하는 softmax classifier임

$$P(w_t = i | U, C) = \frac{e^{v_i u}}{\sum_{j \in V} e^{v_j u}}$$

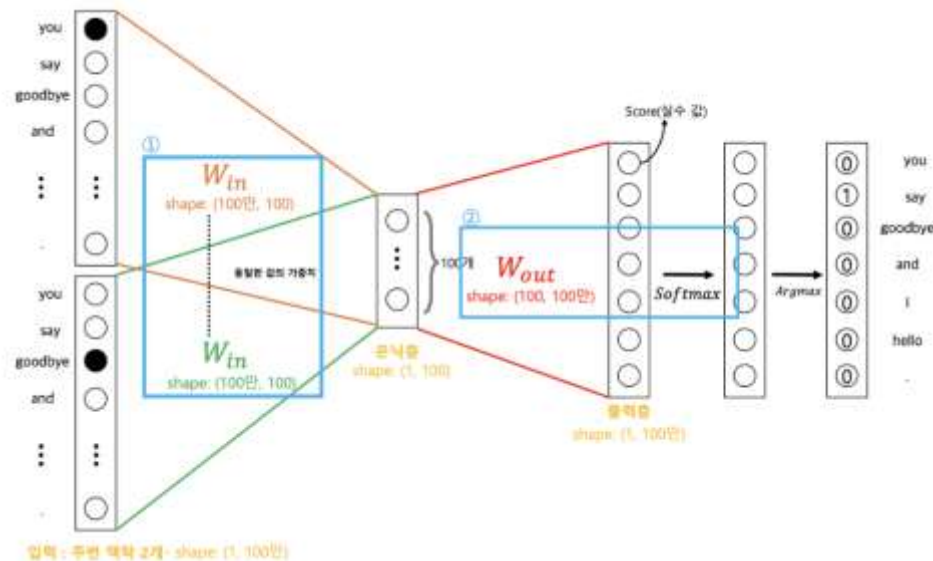
- w_t 에서의 softmax를 구하는 과정에서 영상이 M개라면 $O(M)$ 의 시간 복잡도가 발생하는데 M이 매우 큰 유튜브 scale에서는 연산량이 매우 커짐
- 따라서 softmax를 구하는 과정을 최적화해야 하는데 논문에선 negative sampling을 이용해 이진 분류 문제로 바꿔서 해결했다.

Negative sampling

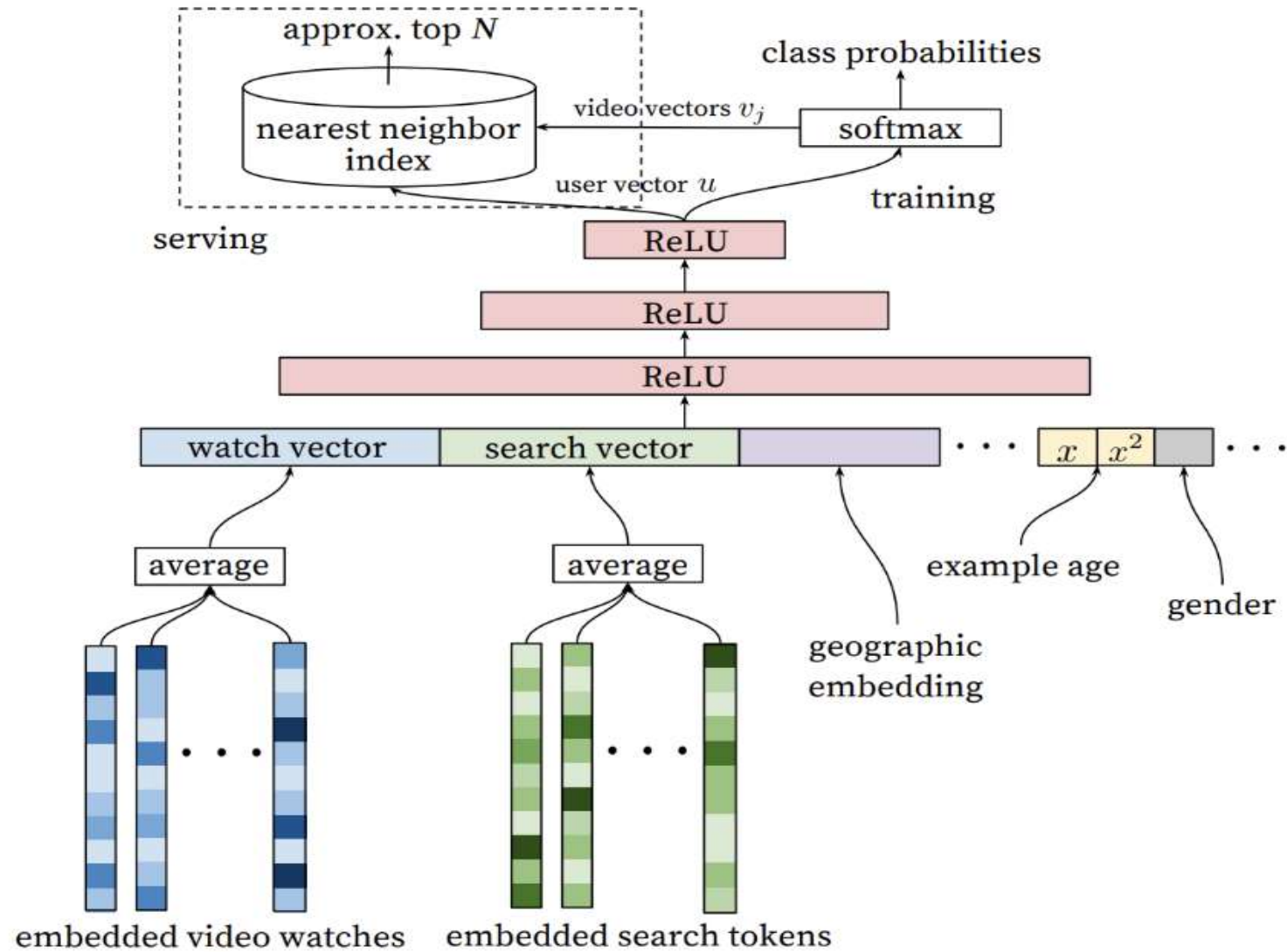
- Negative sampling의 예시로 word2vec의 학습을 예로 들 수 있음

- 1. 인덱스를 잘 매핑시켜 행렬연산을 안해도 됨
- 2. 활성화함수를 softmax에서 sigmoid로 바꿈

다중분류에서 이진 분류로 바꿈



4. Model architecture



4. Model architecture

- 유저가 시청한 영상(ID) 임베딩을 average하여 고정된 차원의 watch vector 구성
- 유저가 검색한 tokens들의 임베딩을 average하여 고정된 차원의 search vector 구성
- 새로운 유저에게 추천을 하기 위해서 Demographic 정보 활용
 - geographic region과 devices 정보는 임베딩 추출
 - 유저의 성별, 로그인 정보, 나이 같은 정보는 $[0,1][0,1]$ 로 정규화하여 활용
- 유저는 자신의 취향과 비슷한 영상일 경우 최신 영상을 선호하기 때문에 example age를 도입해 최신 영상에 가중치 부여
 - 시청 기록에 있어 아이템 벡터의 평균을 적용하기 때문에 시간적 요소에 대한 sequence가 반영되지 않는데, 이를 보정하기 위해서 아이템의 나이, 즉 영상 콘텐츠의 Age에 해당하는 feature를 추가하게 됨
 - 실제로 example age를 도입할 경우 경험적으로 얻은 분포와 매우 비슷해짐.

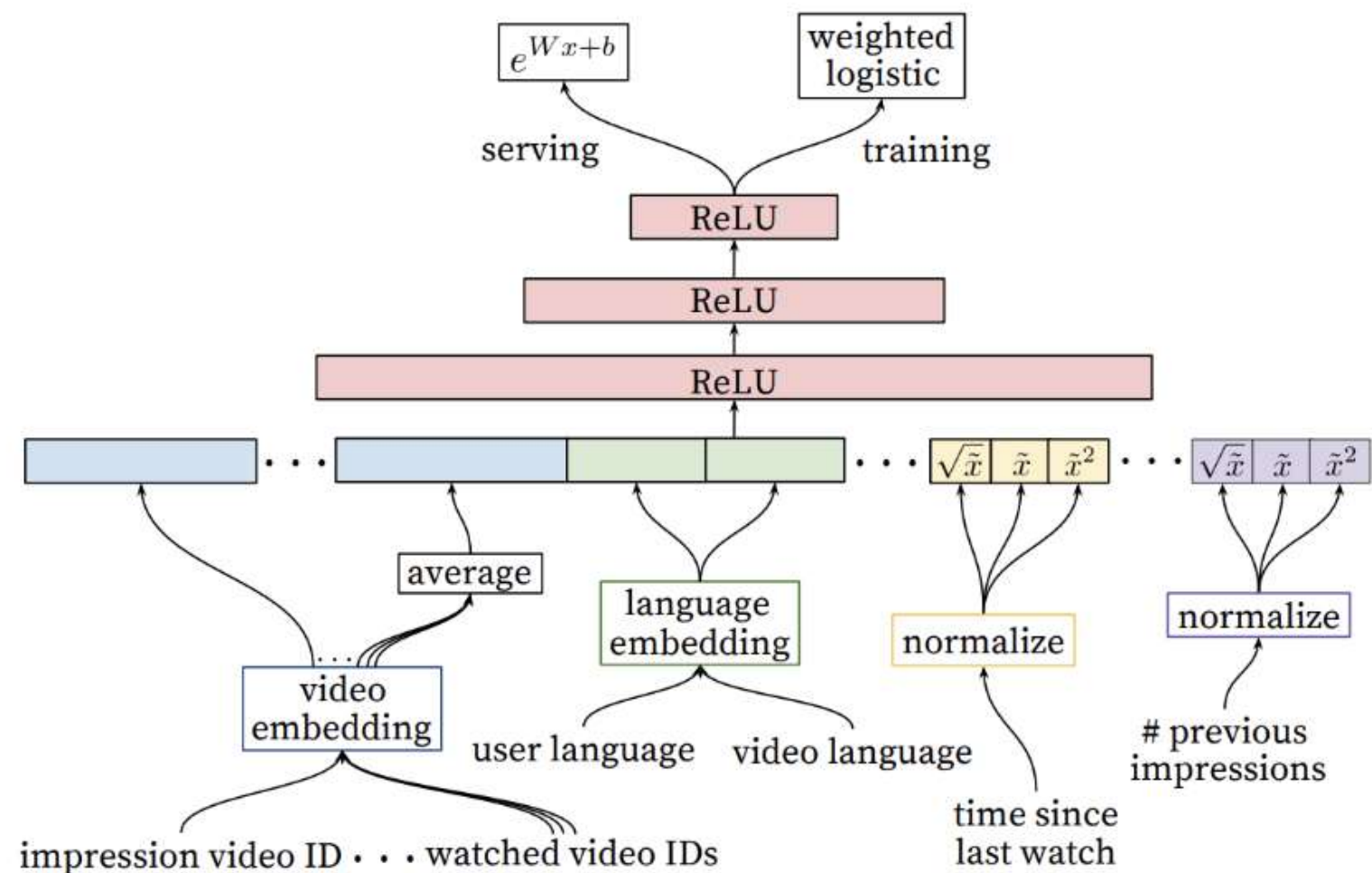
4-1. Cold start problem

- 시스템이 해당 사용자나 아이템에 대해 충분한 정보를 가지고 있지 않아 정확한 추천을 제공하기 어렵다는 것을 의미
 - 추천시스템이 해결하고자 하는 cold start 문제는 demographic 정보를 통해 해결을 한다고 함
 - 지리적 특성 및 기기 정보가 임베딩 되어있으며 그 두개는 이어져있음.
 - 또한 성별, 로그인 상태, 나이같은 경우는 0과 1 사이로 normalized 되어 input 으로 들어간다고함.

5. Ranking

- ranking 모델의 주 역할은 각 사용자의 feature를 사용해서 후보 아이템을 특성화하고 조정하는 것임
- ranking 모델 단계에서는 겨우 몇 백개의 아이템에 불과하기 때문에 아이템에 대한 사용자의 action 및 feedback이 훨씬 dense함
- candidate generation 모델과 구조 자체는 유사하며, 각 아이템에 score(점수)를 할당하게 함으로써 정렬하여 사용자에게 반영함

5. Ranking



5. Ranking

■ Feature representation

- features는 데이터 형태에 따라 categorical(범주형)과 continuous(연속형)로 구분할 수 있음
- categorical feature의 경우에는 binary 데이터, 사용자의 마지막 검색 기록, 점수를 부여할 아이템 ID, 최근 본 N개의 아이템 ID가 있음

■ Feature Engineering

- 가장 중요한 과제는 사용자 행동에 대한 sequence를 반영하는 것과 어떻게 사용자 행동을 아이템 점수화와 연관시킬 것인지임
 - 해당 채널에서 시청한 영상의 개수, 해당 이슈와 관련된 영상을 시청한 마지막 시점 등과 같은 연속적인 feature는 아이템과 연관된 사용자의 과거 행동이기 때문에 매우 효과적임

5. Ranking

■ Embedding Categorical Features

- 영상 콘텐츠, 즉 아이템의 ID와 검색 기록을 임베딩하여 인공신경망에 input함
 - categorical 데이터가 지나치게 많을 경우, click의 빈도수를 기반으로 top N을 선정함
 - 반대로 부족한 경우에는 zero 임베딩을 함(padding 개념으로 이해함, 데이터가 부족한 경우)
 - candidate generation 과정처럼 multivalent feature(최근 본 N개의 아이템)의 경우에는 평균을 적용함

■ Normalizing Continuous Features

- continuous feature의 경우에는 0~1로 scaling해주며 super/sub-linear한 특징을 배우기 위해 input으로 넣어줌

■ Modeling Expected Watch Time

- 주어진 학습 데이터에서 사용자가 클릭한 아이템(positive)과 클릭하지 않은 아이템(negative)의 기대 watch time(시청 시간)을 예측하는 것이 목표
- positive 아이템의 경우 사용자가 시청한 시간에 대한 기록이 남겨져 있으며, weighted logistic regression을 사용해서 예측할 수 있음

5 . Conclusion

- Depth 0: A linear layer simply transforms the concatenation layer to match the softmax dimension of 256
- Depth 1: 256 ReLU
- Depth 2: 512 ReLU \rightarrow 256 ReLU
- Depth 3: 1024 ReLU \rightarrow 512 ReLU \rightarrow 256 ReLU
- Depth 4: 2048 ReLU \rightarrow 1024 ReLU \rightarrow 512 ReLU \rightarrow 256 ReLU

Hidden layers	weighted, per-user loss
None	41.6%
256 ReLU	36.9%
512 ReLU	36.7%
1024 ReLU	35.8%
512 ReLU \rightarrow 256 ReLU	35.2%
1024 ReLU \rightarrow 512 ReLU	34.7%
1024 ReLU \rightarrow 512 ReLU \rightarrow 256 ReLU	34.6%

5 . Conclusion

- Youtube의 영상 콘텐츠를 추천하는 딥러닝 모델인 candidate generation과 ranking 모델에 대해 소개
- Candidates generation 과 ranking model로 deep layer를 만들어 그 상호작용을 모델링하여, 이전의 YouTube에서 사용된 행렬 인수분해 접근법을 능가했음
 - 추천 시스템은 특히 사용자의 과거 행동이 특징된 feature로부터 좋은 효과를 얻게 됨
 - 인공신경망은 임베딩되거나 정규화되어 있는 categorical, continuous한 데이터가 필요함
- 사용자가 본 아이템(positive)으로부터 watch time(시청 시간)을 학습한 weighted logistic regression은 CTR을 직접적으로 예측하는 metrics보다 좋은 성능을 보여줌

DeepFM: A Factorization-Machine based Neural Network for CTR Prediction

Huifeng Guo , Ruiming Tang

Shenzhen Graduate School, Harbin Institute of Technology

Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17) mar 2017

<https://arxiv.org/abs/1703.04247>