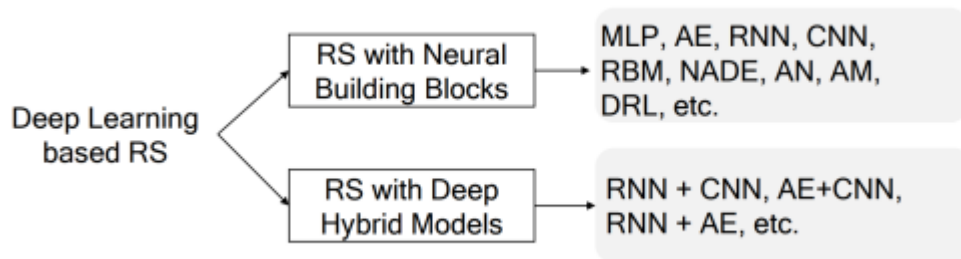


딥러닝을 이용한 추천 모델

Table 1. A lookup table for reviewed publications.

| Categories | Publications |
|-------------------|---|
| MLP | [2, 13, 20, 27, 38, 47, 53, 54, 66, 92, 95, 157, 166, 185], [12, 39, 93, 112, 134, 154, 182, 183] |
| Autoencoder | [34, 88, 89, 114, 116, 125, 136, 137, 140, 159, 177, 187, 207], [4, 10, 32, 94, 150, 151, 158, 170, 171, 188, 196, 208, 209] |
| CNNs | [25, 49, 50, 75, 76, 98, 105, 127, 130, 153, 165, 172, 202, 206], [6, 44, 51, 83, 110, 126, 143, 148, 169, 190, 191] |
| RNNs | [5, 28, 35, 56, 57, 73, 78, 90, 117, 132, 139, 142, 174–176], [24, 29, 33, 55, 68, 91, 108, 113, 133, 141, 149, 173, 179] |
| RBM | [42, 71, 72, 100, 123, 167, 180] |
| NADE | [36, 203, 204] |
| Neural Attention | [14, 44, 70, 90, 99, 101, 127, 145, 169, 189, 194, 205], [62, 146, 193] |
| Adversary Network | [9, 52, 162, 164] |
| DRL | [16, 21, 107, 168, 198–200] |
| Hybrid Models | [17, 38, 41, 82, 84, 87, 118, 135, 160, 192, 193] |



NCF(MLP)

기존 선형적인 모델에서 **MLP**을 통해 비선형성을 추가한 모델

- 유저의 선호도와 상품 특징 간의 양방향성 상호작용을 활용
- 고차원 벡터를 표현

필요 데이터

- 사용자 데이터 - 사용자 아이디, 사용자 선호 카테고리 혹은 키워드, 행동 데이터(클릭, 찜)
- 상품 데이터 - 상품 id, 키워드 혹은 카테고리, 가격 등의 상품 정보들

추천 시스템에서의 **MLP** 모델의 발전

- Locally-Connected wide & deep Learning Model

- Locally-Connected Network를 통해, 실행시간을 10배 감소
(Locally-Connected Network는 Convolutional Neural Network(CNN)에서 아이디어를 차용한 구조.)
- Feature Engineering 감소

- Youtube DNN(사용 사례)

- Candidate Generation과 Candidate Ranking 두 단계로 구성
- Feature Engineering의 많은 노력을 기울임 → 중요성을 확인

⇒ [요약문헌](#)

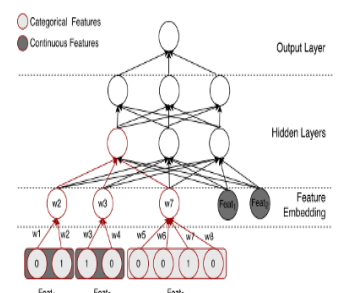


Figure 5: A 5-layer locally connected DNN (LC-DNN).

Deep FM ,wide & deep Learning Model

저차원 뿐만 아니라 고차원 간의 상관 관계

Collaborative Metric Learning(CML)

- User와 Item의 임베딩을 고도화 - User가 싫어하는 Item은 거리를 최대화 & User가 좋아하는 Item은 거리를 최소화
- 유클리디안 거리를 통해 선호도를 학습

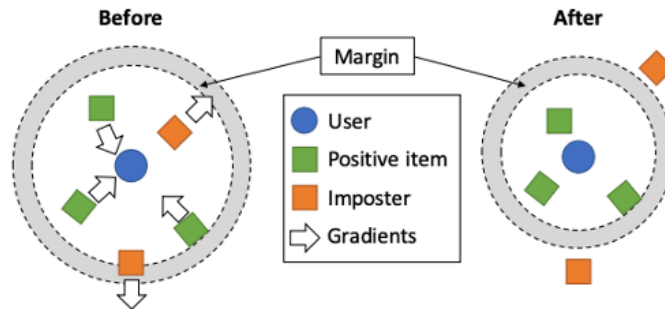
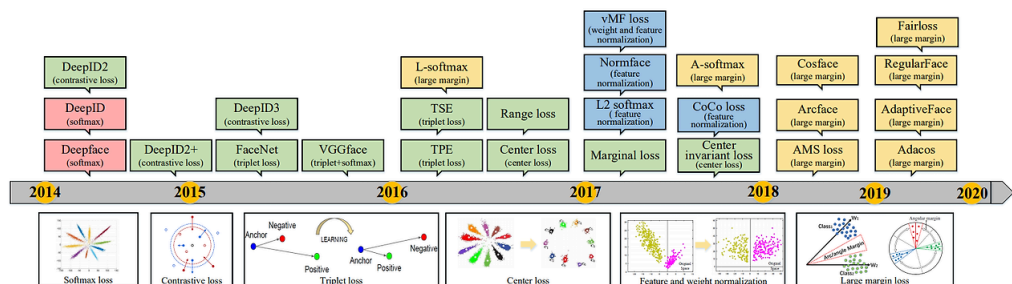


Figure 1: An illustration of collaborative metric learning. The hinge loss defined in Eq. 1 creates a gradient that *pulls* positive items closer to the user and *pushes* the intruding imposter items (i.e., items that the user did not like) away until they are beyond the safety margin.

Metric Learning

- 분류(class) 대상이 매일 바뀌는 경우 (ex. 얼굴인식, 상품 검색)
- 핵심:
 1. 사용하는 거리(유사도)
 2. Loss→ Contrastive Loss, Triplet Loss, Margin Loss 등

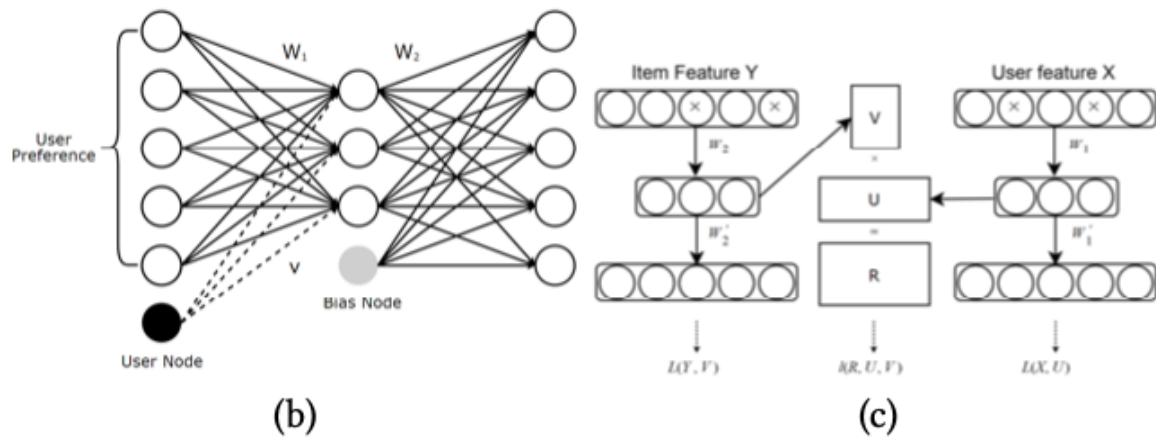


- 장점:
 1. 일반적인 Classification은 학습한 이미지에 대해서만 인식 가능 그러나 ML은 학습하지 않은 이미지도 DB에 구축 시 인식
 2. ML은 비유사한 Feature들을 멀리 떨어지도록 학습
→ Feature 공간을 더 잘 사용 가능

⇒ [요약문헌](#)

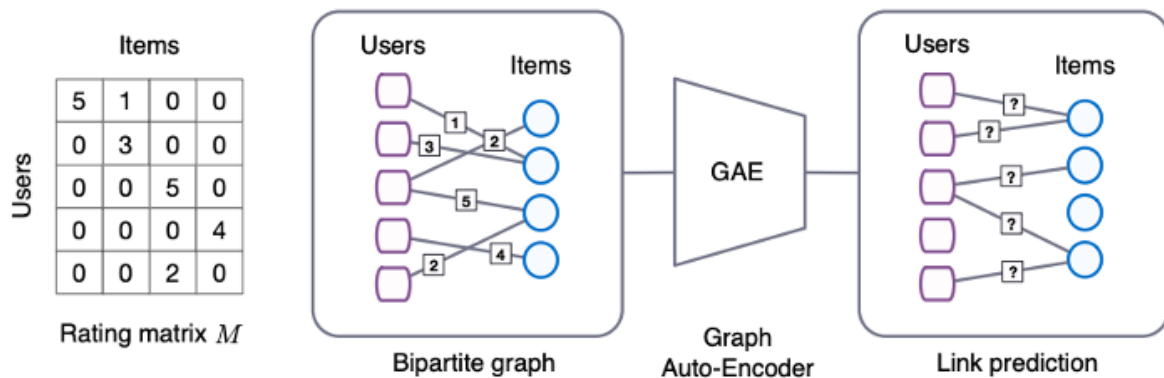
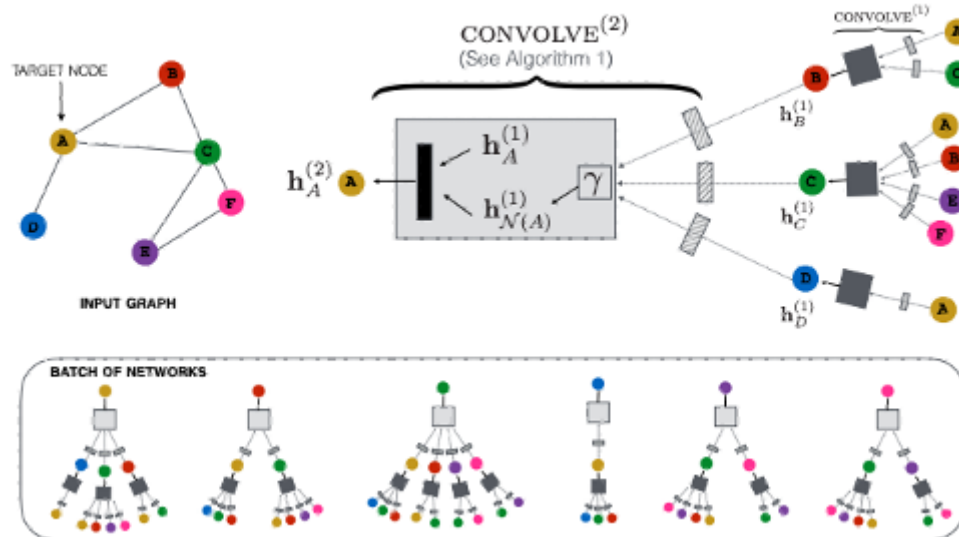
AUTOENCODER

[참고 문헌](#) → 협업인데 고객 리뷰 별점을 이요한 듯, 여기서도 리뷰 별로 없다고 희박하다 함



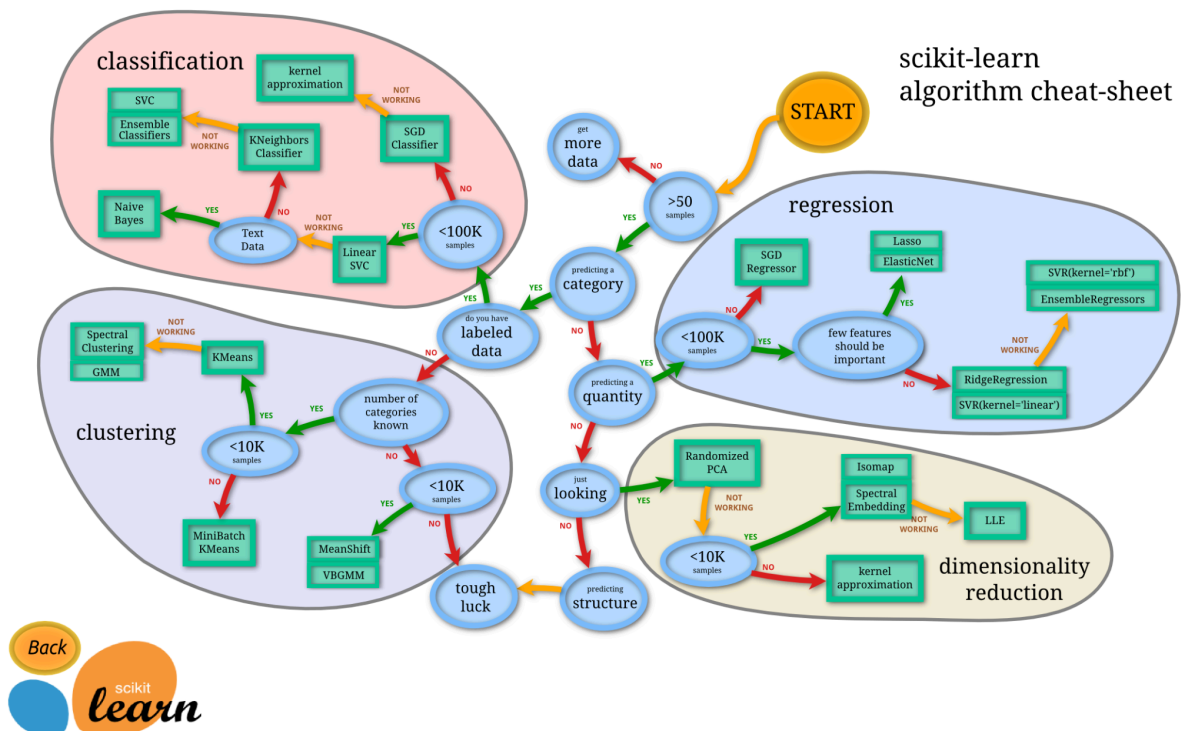
GRAPH CNNs

- Graph CNN은 소셜 네트워크, Knowledge Graph 등과 같은 관계성 데이터에 적합
- 추천 분야는 이분 그래프(Bipartite Graph)로 표현할 수 있음



- 핀터레스트에서 사용 중인 추천 모델

SKlearn 이용한 추천 모델



Naive Bayes

- 베이즈라는 예측 모델
- 모든 상품이 독립적이라는 가정이 들어감
- 모델 속도가 매우 빠름
- 간단함
- 협업 기반이긴 하지만 참고하면 좋을 것 같은 논문 → [문헌](#)

K neighbors classifier (+클러스터링)

- 유클리디언 거리 계산법 사용
- 저희 수업시간에 배웠죠? 그거임

SVD(MF)

The diagram shows the SVD decomposition of a matrix M into three matrices: U , Σ , and V^* . The dimensions are given as $m \times n$ for M , $m \times m$ for U , $m \times n$ for Σ , and $n \times n$ for V^* .

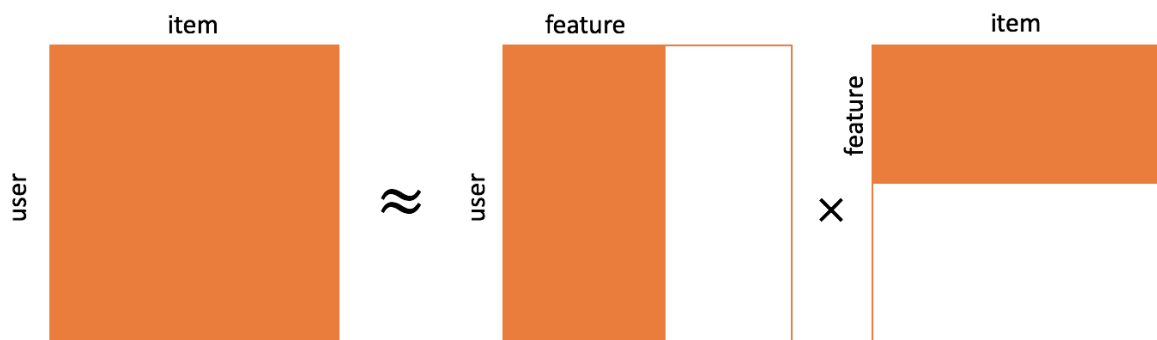
Below the main equation, two smaller equations are shown:

$$U U^* = I_m$$

$$V V^* = I_n$$

- Rating Matrix R에 대해 유저와 아이템의 잠재 요인을 포함할 수 있는 행렬로 분해한다.
- SVD는 선형대수학 기반의 행렬 분해 방법으로, 추천 시스템에서 MF와 유사한 역할을 하지만, 명확한 수학적 정의를 기반으로 동작합니다.
- 장점
 - 수학적 최적화: 분해 과정이 명확히 정의되어 있어 안정적인 결과를 보장.
 - 데이터 압축: 중요한 정보만 남기고 차원을 축소하므로, 계산량 감소.
- 단점
 - 희소 행렬 한계: 사용자-아이템 행렬에 공백이 많으면 직접 사용하기 어려움.
 - 콜드 스타트 문제: 새로운 사용자나 아이템 처리에 한계.

MF



- SVD의 개념과 유사하지만, MF는 관측된 선호도(평점)만 모델링에 활용하고 관측되지 않은 선호도를 예측하는 일반적인 모델을 만든다.
- SVD와는 다르게 실제 관측된 데이터만을 사용하여 모델을 학습
- 장점
 - 잠재적 관계 학습: 사용자와 아이템 간의 숨겨진 패턴(예: 사용자가 특정 장르를 선호함)을 학습.

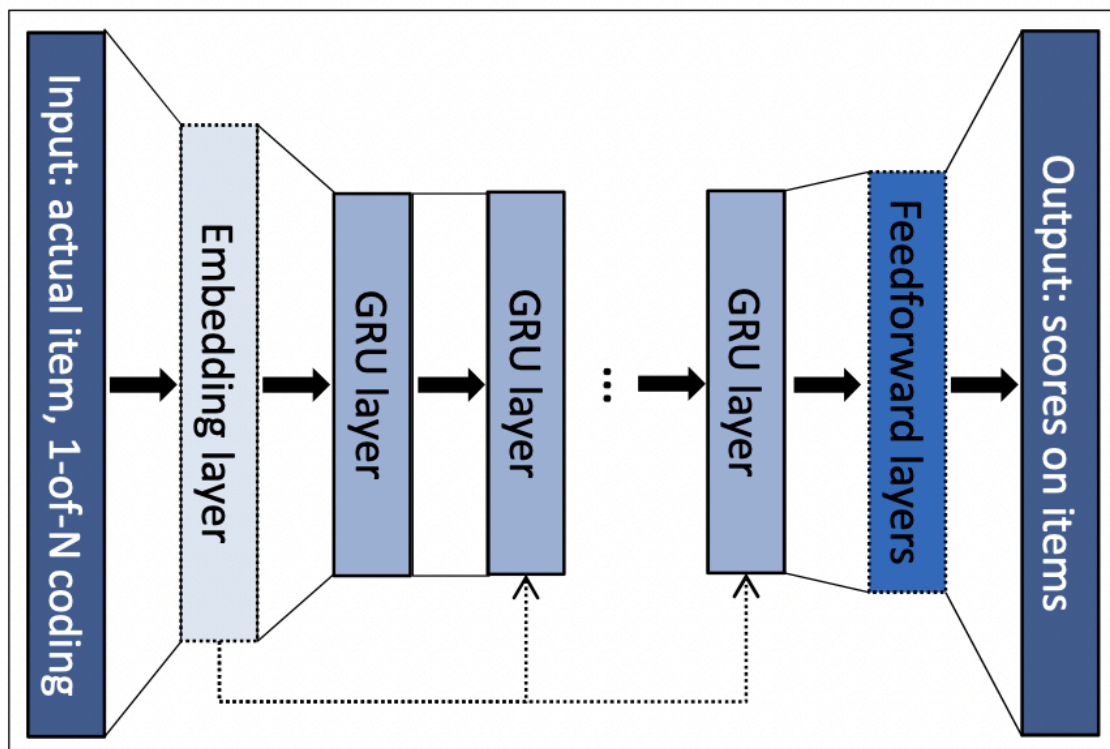
- 효율적: 행렬이 희소할 때에도 적은 데이터로 잘 학습 가능.
- 단점
- 콜드 스타트 문제: 새로운 사용자나 아이템에 대해 학습된 정보가 없어 추천이 어려움.
- 데이터 의존: 충분히 크고 다양한 사용자-아이템 데이터가 필요.

| 특징 | MF | SVD |
|--------|---------------------------|------------------------|
| 방식 | 데이터 기반 최적화 (주로 경사하강법 사용). | 선형대수학 기반 분해. |
| 입력 데이터 | 희소 행렬에서 직접 학습 가능. | 완전한 행렬이 필요 (공백 처리 필요). |
| 사용 사례 | 대규모 데이터에서 효율적. | 작거나 완전한 데이터에 적합. |

홈쇼핑 모아에서 고려하고 있는 추천 모델

RNN

- [GRU4Rec](#)
- GRU 기반 세션 추천: **Gated Recurrent Unit(GRU)**을 활용하여 세션 내 사용자 행동 시퀀스를 모델링.
- 시간 순서 학습: 사용자 행동의 순차적 패턴을 학습하여 다음 행동을 예측.
- 세션 기반 특화: 짧은 시간 동안의 사용자 선호 변화를 반영한 실시간 추천에 강점.
- 장점
 1. 효율적인 시퀀스 모델링: **GRU**로 기울기 소실 문제를 완화하고 긴 시퀀스도 처리 가능.
 2. 실시간 추천 가능: 세션 내 행동을 실시간 분석하여 바로 추천 제공.
 3. 간결한 구조: **LSTM**보다 단순해 계산 효율이 높음.
- 단점
 1. 긴 시퀀스 처리 한계: 매우 긴 시퀀스에서는 성능 저하 가능.
 2. 하이퍼파라미터 튜닝 필요: 모델 성능 최적화를 위해 세심한 설정이 요구됨.
 3. 콜드 스타트 문제: 새로운 사용자나 아이템에 대한 데이터 부족 시 성능 한계.



1. 콘텐츠 기반 필터링

- 소비한 콘텐츠를 기준으로 유사한 특성을 가진 콘텐츠 추천

- 키워드로 TF-IDF 계산(TF-IDF: 키워드 빈도를 구해 가중치 적용)

- 유사도

* 코사인 유사도 => 한 상품을 클릭했을 때, 코사인 유사도가 높은 키워드를 가지고있는 상품 추천

* 유클리드 거리 => 유클리드 거리의 값이 작을수록 키워드간 비슷한 의미를 가짐(코사인만큼 유용하지는 못함)

* 자카드 유사도 => 두 집합의 합집합에서 교집합의 비율

* 피어슨 유사도 => 아이템의 벡터간 선형 상관관계

- 추론기반(Word2Vec)

* CBOW: 주변 단어(맥락)를 통해서 중심 단어를 채우는 방법

* Skip-Gram: 중심 단어를 통해서 주변 단어를 채우는 방법

=> 1. 특정 상품 제목의 벡터와 가장 유사한 벡터를 가진 다른 상품 추천(제목 전처리 필요)

2. 사용자가 입력한 키워드와 유사한 키워드를 포함하는 상품 추천

3. 키워드 벡터를 클러스터링하여 비슷한 상품 그룹화

- 사용자가 관심 카테고리로 선택한 카테고리 and 유사도를 비교하여 상품 추천

- 참고 논문

* 코사인 유사도 기법을 이용한 뉴스 추천

시스템(<https://scienceon.kisti.re.kr/srch/selectPORSrchArticle.do?cn=NPAP12013299>)

item2vec

https://github.com/comsa33/Recommndation_System

Word2Vec의 SGNS에서 영감을 받아 추천시스템에 적용한 형태
