



# Recommendation Systems

강필성

고려대학교 산업경영공학부

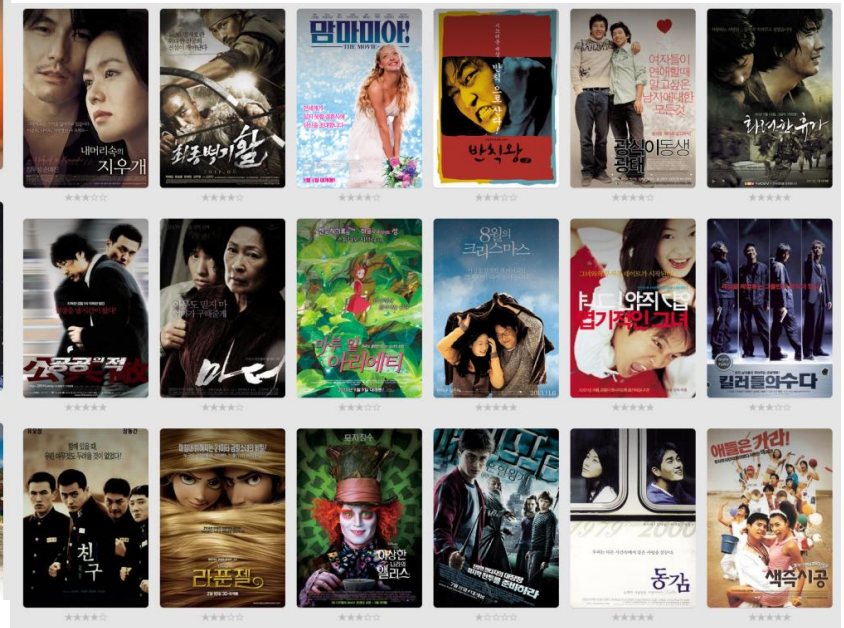
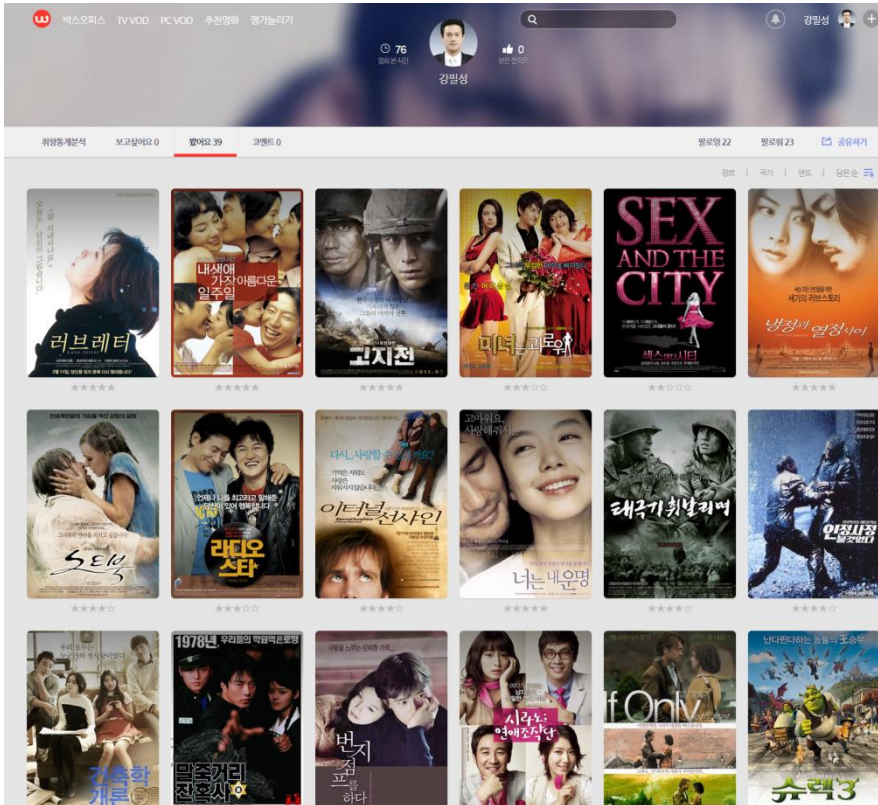
Bflysoft & WIGO AI LAB

# AGENDA

- 01 Recommendation Systems
- 02 Association Rule Mining
- 03 Collaborative Filtering: k-Nearest Neighbor
- 04 Matrix Factorization-based Method

# 추천 시스템

- 어떤 영화를 볼까?



# 추천 시스템

## • 어떤 영화를 볼까?

예상 별점이 가장 높은 영화 × 장르 | 국가 | 연도 | 추천이유

<p>4.2 가장 아름다운 이별</p> <p>다 많은 시간 함께 할 걸 그랬습니다</p> <p>〈내 생애 가장 아름다운 일주일〉과 비슷해요.</p>	<p>4.3 내 나라 내 사랑</p> <p>나를 잊게 하진... 항상 함께</p> <p>예상 별점 4.3개 ★★★★★</p>	<p>4.2 저수지</p> <p>영화가 평생 잊을 사람</p> <p>〈월의 크리스마스〉와 비슷해요.</p>	<p>4.0 조용한 하마</p> <p>가장 아름다운 이별에 가장 따뜻한 이별</p> <p>〈시라노 연애조작단〉과 비슷해요.</p>	<p>4.3 소원</p> <p>가장 아름다운 이별에 가장 따뜻한 이별</p> <p>좋아하는 배우 설경구</p>
<p>4.2 신세계</p> <p>세 남자가 가고 싶었던 세로 나운</p> <p>〈친구〉와 비슷해요.</p>	<p>4.0 내 사랑 내 정애</p> <p>말할 수도 없는 그가 당신을 올립니다</p> <p>〈너는 내 운명〉과 비슷해요.</p>	<p>4.0 한 공 주</p> <p>한 공 주</p> <p>예상 별점 4.0개 ★★★★★</p>	<p>4.3 은교</p> <p>가장 아름다운 이별에 가장 따뜻한 이별</p> <p>예상 별점 4.3개 ★★★★★</p>	<p>4.8 아는 여자</p> <p>가장 아름다운 이별에 가장 따뜻한 이별</p> <p>〈시라노 연애조작단〉과 비슷해요.</p>
<p>4.6 Before Sunset</p> <p>이러날 전사인) 외 2편과 비슷해요.</p>	<p>4.0 만족</p> <p>우리의 행복한 시간과 비슷해요.</p>	<p>4.0 세레니티</p> <p>〈노르북〉 외 1편과 비슷해요.</p>	<p>4.0 RESIDENT EVIL</p> <p>예상 별점 4.0개 ★★★★★</p>	<p>4.0 running on empty</p> <p>예상 별점 4.0개 ★★★★★</p>

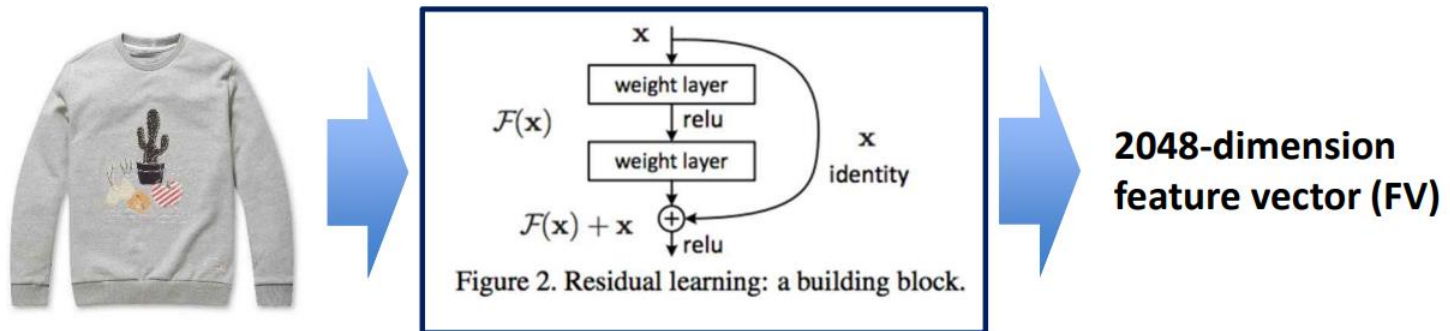


# 추천 시스템

- 어떤 옷을 살까?/입을까?

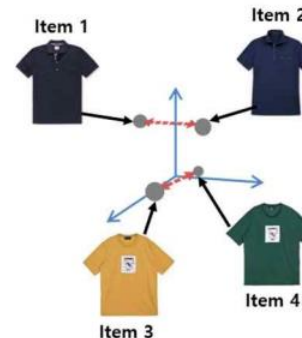
- CNN(Convolutional Neural Networks) 기반 이미지 프로세싱**

- 오픈 소스로 활용되는 *Pre-trained ResNet\** 활용
- 이미지가 갖는 추상적인 정보를 고차원의 벡터로 저장



- 상품 간 유사도 정량화**

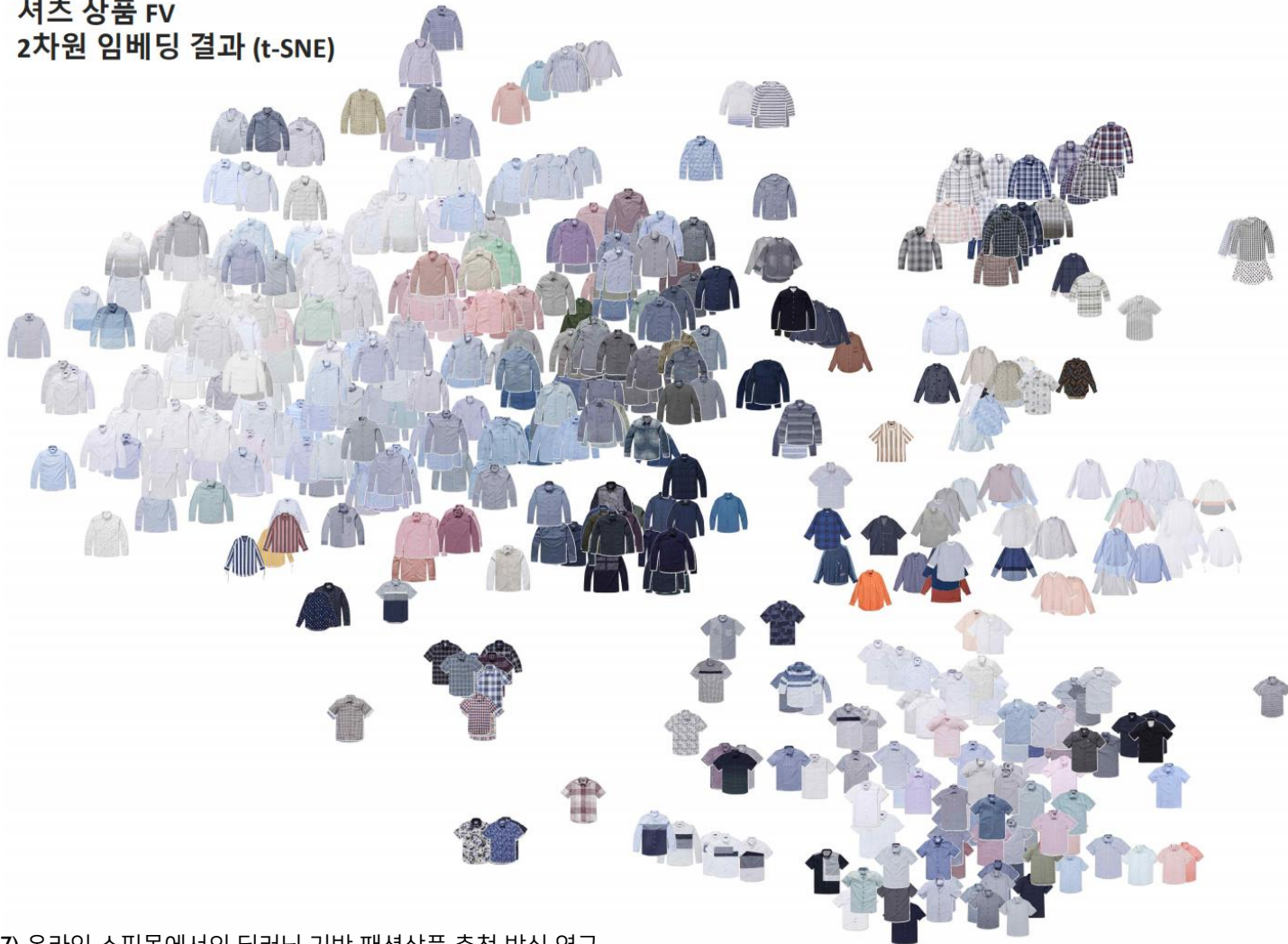
- 각 상품들의 FV 사이 L2-distance를 계산
- 유사한 제품 간 거리는 작고,  
유사하지 않은 제품 간 거리는 큼



# 추천 시스템

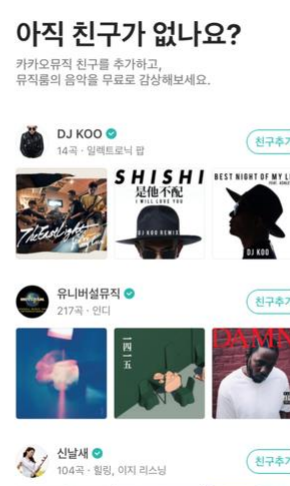
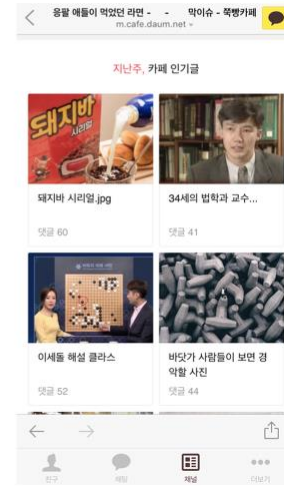
- 어떤 옷을 살까?/입을까?

셔츠 상품 FV  
2차원 임베딩 결과 (t-SNE)



# 추천 시스템

## • 일상 생활에서의 추천



# 추천 시스템

## • 금융에서의 추천



### Finda Users

핀다는 어떤 사람들이 이용할까요?



### Product

어떤 상품이 인기가 많았을까요?



### Behavior

투자자와 대출자에게 인기 있는 상품은?



### Keywords

리뷰에서 가장 많은 나온 단어는?



1:29

재테크

카드추천
대출협상
보험설계
노후준비

내 카드 ?

현대카드M3
를 바꾸면

연 710,754원이 절약되네요 ✨

이 카드의 예상 연 혜택 : 608,703원 >

할인/적립
항공 마일리지

연회비 이벤트 중인 카드만 보기

1 씨티 리워드 카드

연 혜택
710,754원
연회비 100% 캐시백

상세보기
신청하기

2 신세계 씨티 리워드 카드

연 혜택
680,372원
연회비 100% 캐시백

상세보기
신청하기

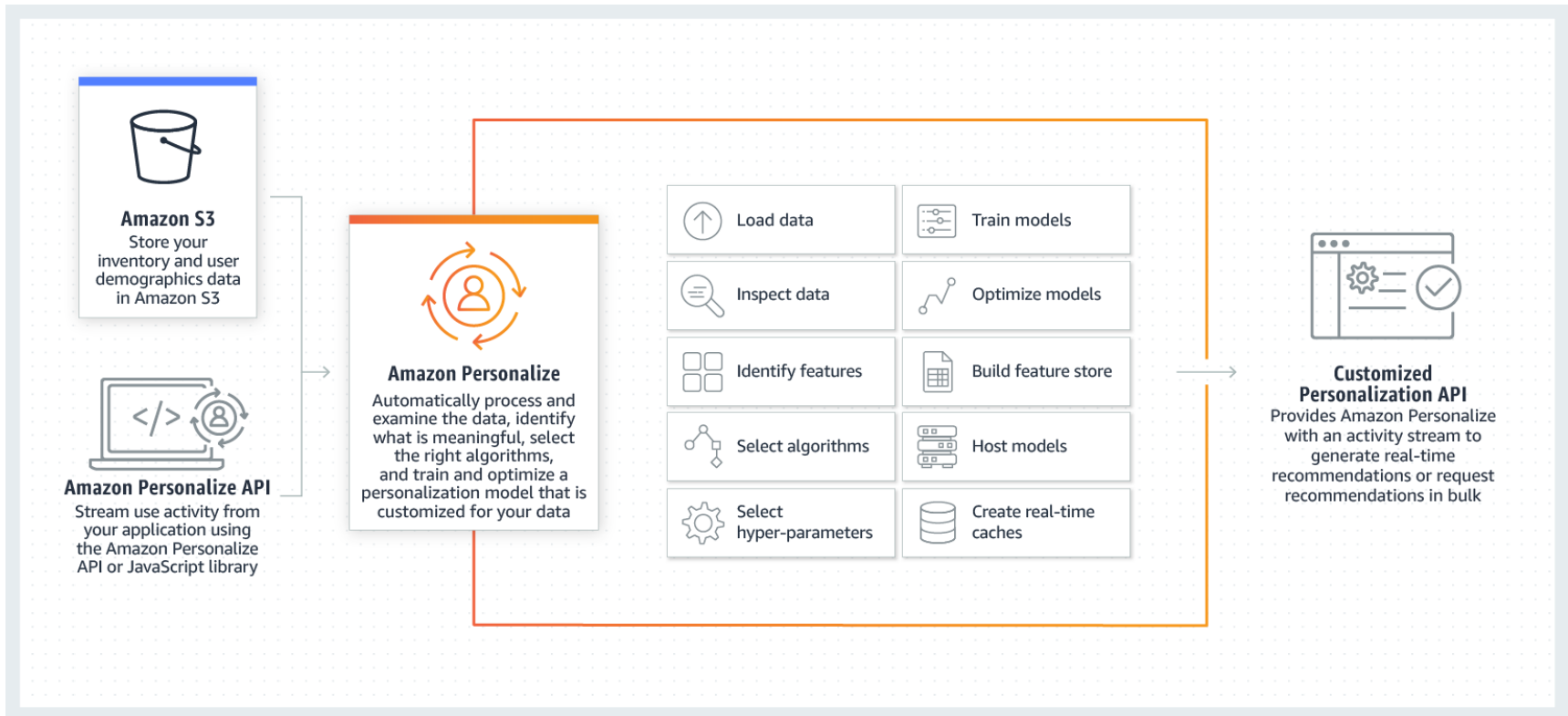
MY금융
가계부
금융비서
재테크
더보기



# 추천시스템

- 비즈니스 관점에서의 추천시스템의 효과

- ✓ Netflix: 소비되는 콘텐츠의 2/3가 추천으로부터 발생
- ✓ Google News: 38% 이상의 조회가 추천에 의해 발생
- ✓ Amazon: 판매의 35%가 추천으로부터 발생



# 추천시스템

- 추천 시스템의 목적

- ✓ 고객의 만족도를 높이고 더 나아가 고객 스스로 기꺼이 유/무형의 비용을 지불할 의사가 생기는 제품/서비스/콘텐츠를 제안하는 것
  - 유형 비용: 해당 서비스를 이용함으로써 직접적으로 지불하는 비용 (예: 쇼핑몰에서 추천한 상품을 추가로 구매)
  - 무형 비용: 고객이 해당 서비스를 오래 이용함으로써 해당 사업자가 이득을 취하는 구조를 만들 수 있는 것 (플랫폼에서 무료 콘텐츠를 지속적으로 사용하여 광고에 노출될 수 있는 시간을 확보하는 것)

- 추천 시스템의 핵심

- ✓ 누구에게 무엇을 언제 추천해줄 것인가?

# 추천시스템

- 추천 시스템의 핵심

Who What When

누구에게 무엇을 언제 추천해주어야 하는가?

# 추천시스템

## • 추천 시스템 유형

통계형 추천



TOP 100

상품 기준 추천



[삼우물티슈] M  
BC불만제로모  
17,900원



++하기스매직  
팬티 플레이수  
53,900원



더블하트 PPSU  
노꼭지 젓병(2개)  
31,800원

사용자 기준 추천



초특가 고급면  
스판체크/NC스  
19,010원



[남성 골프웨어]  
여를 반팔 중년  
19,010원



프로선수들도  
즐겨입는/2014여  
39,800원

검색어 기준 추천



[DUCLY]듀클라  
이 후드티 맨투  
24,800원



[단독파격가]남  
자 겨울 라운드  
14,900원



+正品보증+당  
일발송+ 리바이  
44,000원



# 추천시스템

## • 추천 시스템의 종류

### ✓연관성 기반 추천: Association-based Recommendation

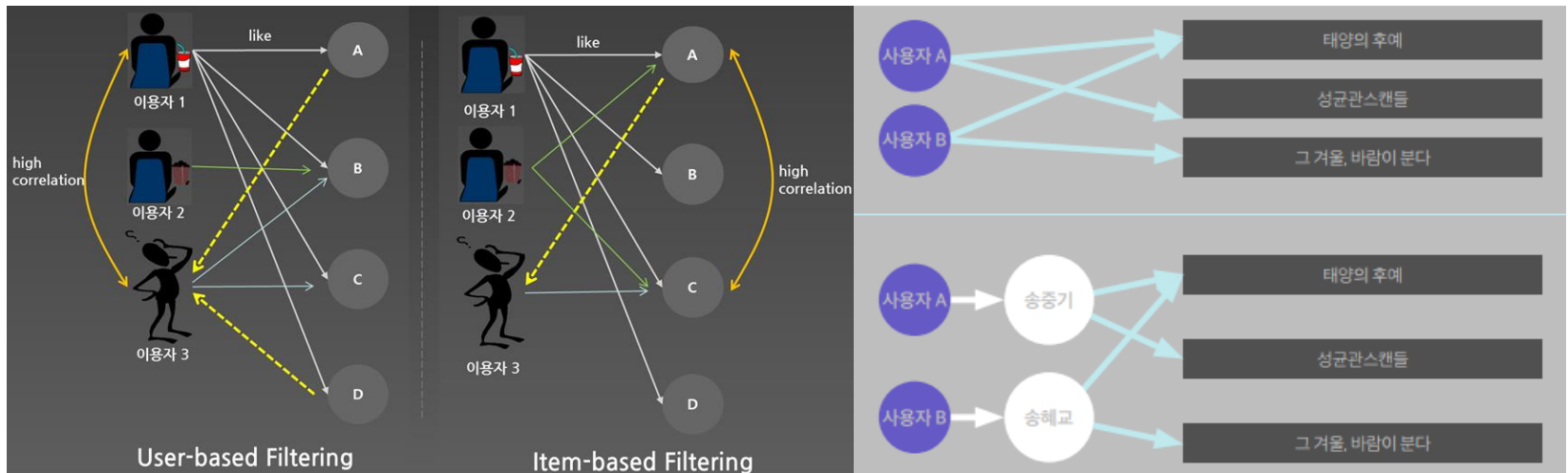
- 사용자의 제품 구매 이력/콘텐츠 이용 패턴을 파악하여 연관성이 높은 제품/콘텐츠 추천

### ✓협업 필터링: Collaborative Filtering

- 대상 고객과 기호(preference)가 유사한 사용자가 높은 평가를 내린 제품/콘텐츠를 추천

### ✓잠재 요인 분석: Latent Factor Model

- 고객들의 평가 속에 숨겨진 잠재적 핵심 요소(latent factor)를 파악하고 이를 기반으로 추천



# AGENDA

- 01 Recommendation Systems
- 02 Association Rule Mining
- 03 Collaborative Filtering: k-Nearest Neighbor
- 04 Matrix Factorization-based Method

# 연관규칙분석

- 연관규칙 분석 a.k.a 장바구니 분석 (MBA)



Wall Mart (USA)



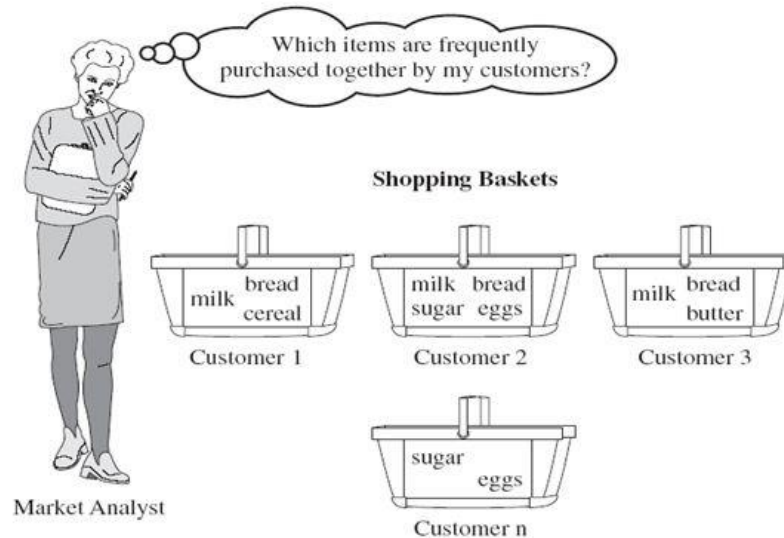
E-Mart (Korea)



# 연관규칙분석

- 목적

- ✓ 어떤 두 아이템 집합이 **빈번히 발생**하는가를 알려주는 **일련의 규칙**들을 생성
  - Produce rules that define “what goes with what”
- ✓ 우리의 데이터에 의하면 “X 아이템을 구매하는 고객들은 Y 아이템 역시 구매할 가능성이 높다”
- ✓ 장바구니 분석 Market Basket Analysis으로도 널리 알려짐





# 연관규칙분석

- 데이터 속성

- ✓ 각 레코드는 트랜잭션의 형태를 가짐

- ✓ 행렬의 형태로 표현하게 되면 대부분의 셀이 0의 값은 희소행렬<sup>sparse matrix</sup>이 됨

[Item list 형태]

























Transaction ID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

[Item matrix 형태]

Transaction ID	Bread	Milk	Diaper	Beer	Eggs	Coke
1	1	1	0	0	0	0
2	1	0	1	1	1	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	1

# 연관규칙분석: 예제

- 동네 작은 가게 매출 데이터

Transaction	Item 1	Item 2	Item 3	Item 4
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

# 연관규칙분석: 용어 및 규칙 생성

- 용어: Terminology

- ✓ 조건절(Antecedent) – “IF” part
- ✓ 결과절(Consequent) – “THEN” part
- ✓ 아이템 집합(Item set) – 조건절 또는 결과절을 구성하는 아이템들의 집합
- ✓ 조건절 아이템 집합과 결과절 아이템 집합은 상호배반 (한 아이템이 조건절과 결과절에 모두 포함될 수 없음)

- 규칙 생성: Generating rules

- ✓ 매우 많은 수의 규칙이 생성 가능 (예시: 첫번째 트랜잭션)
  - 계란을 구매하는 사람들은 라면도 함께 구매한다.
  - 계란과 라면을 구매하는 사람들은 참치도 함께 구매한다.
  - 참치를 구매하는 사람들은 계란도 함께 구매한다.
  - ...

# 연관규칙분석: 규칙의 효용성 측정 지표

For the rule  $A \rightarrow B$

- 지지도: Support

$$\text{support}(A \rightarrow B) = P(A) \text{ or } P(A, B)$$

- ✓ 빈발 아이템 집합 frequent item sets을 판별하는데 사용
- ✓ 지지도가 높을수록 해당 규칙을 적용할 기회가 많아짐



# 연관규칙분석: 규칙의 효용성 측정 지표

For the rule  $A \rightarrow B$

- 신뢰도: Confidence

$$\text{confidence}(A \rightarrow B) = \frac{P(A, B)}{P(A)}$$

- ✓ 조건절이 발생했다는 가정 하에서 결과절이 발생할 조건부 확률
- ✓ 아이템 집합 간의 연관성 강도를 측정하는데 사용
- ✓ 규칙의 신뢰도가 지지도보다 낮으면 규칙으로서 효용 가치가 없음

# 연관규칙분석: 규칙의 효용성 측정 지표

For the rule  $A \rightarrow B$

- 향상도: Lift

$$\text{lift}(A \rightarrow B) = \frac{P(A, B)}{P(A) \cdot P(B)}$$

- ✓ 생성된 규칙이 실제 효용가치가 있는지를 판별하는데 사용
- ✓ 지지도 = 1: 조건절과 결과절은 통계적으로 독립 사건임을 의미함 → 규칙에 포함된 아  
이템 집합 사이에는 유의미한 연관성이 없음
- ✓ 지지도 > 1: 조건절과 결과절은 서로 긍정적인 연관관계를 나타냄
- ✓ 지지도 < 1: 조건절과 결과절은 서로 부정적인 연관관계를 나타냄

# 연관규칙분석: 규칙 생성

- 유용한 연관 규칙들을 어떻게 찾아낼 것인가?
  - ✓ 이상적으로는 모든 생성 가능한 규칙을 만든 뒤, 각 규칙의 지지도, 신뢰도, 향상도를 측정하여 유용한 규칙들만을 찾아냄
  - ✓ 아이템 수가 증가할수록 계산에 소요되는 시간이 기하급수적으로 증가함
- Brute-force approach
  - ✓ 가능한 모든 규칙을 나열함
  - ✓ 모든 규칙의 지지도와 신뢰도를 계산함
  - ✓ 최소지지도와 최소신뢰도 조건을 만족하지 못하는 규칙을 제거
  - ✓ **Computationally prohibitive!**

# 연관규칙분석

- A priori algorithm

- ✓ 빈발 집합 frequent item sets만을 고려하여 규칙 생성

- ✓ 지지도 support

- 조건절에 속하는 아이템 집합이 발생할 확률
    - 아이템 집합 {계란, 라면}의 지지도는 40%

- ✓ 최소 지지도 minimum support

- 유용한 규칙으로 인정받기 위해 필요한 최소 지지도

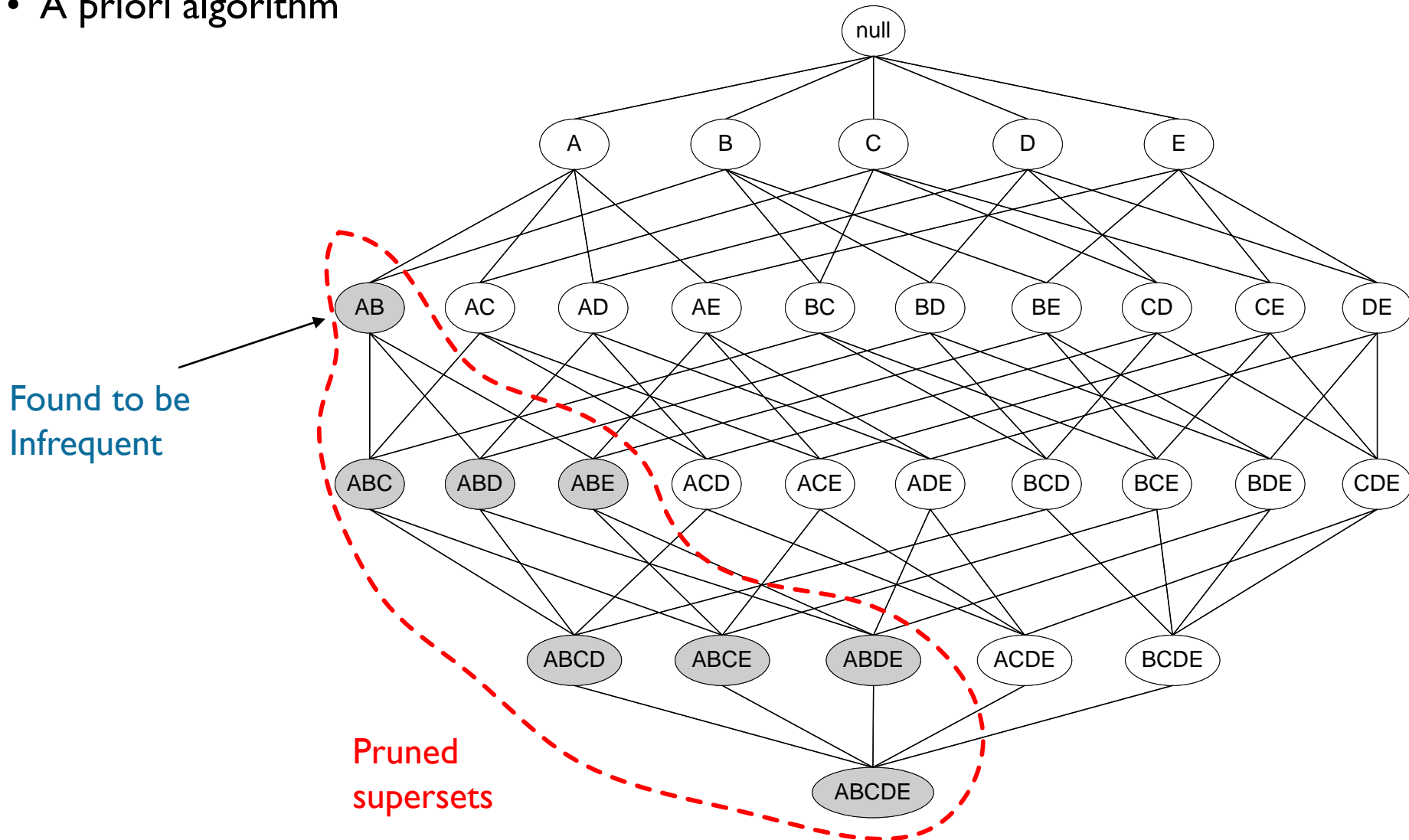
- ✓ 최소 지지도를 만족하지 못하는 아이템 집합의 상위 집합 superset은 항상 최소 지지도를 만족하지 않음

- Support of an item set never exceeds the support of its subsets, which is known as **anti-monotone** property of support.



# 연관규칙분석

























- A priori algorithm



# 연관규칙분석

- 연관규칙분석 Step 1: 최소 지지도 조건 부여

✓ 최소 지지도: 2 transactions or 20%

Transaction	Item 1	Item 2	Item 3	Item 4
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

# 연관규칙분석

- 연관규칙분석 Step 2: 빈발 집합 생성

✓ Step 2-1: 최소 지지도 조건을 만족하는 1개짜리 아이템 집합을 생성



8 (80%)



5 (50%)



5 (50%)



3 (30%)



2 (20%)



1 (10%)

✓ 양파는 최소 지지도 조건을 만족하지 못했으므로 이후 분석에서 제외

# 연관규칙분석

- 연관규칙분석 Step 2: 빈발 집합 생성

- ✓ Step 2-2: 최소 지지도 조건을 만족하는 2개짜리 아이템 집합을 생성

	noodle	egg	cola	rice	tuna
noodle		40%	40%	20%	20%
egg			30%	0%	20%
cola				0%	10%
rice					0%
tuna					

- ✓ 조건을 만족하는 2개짜리 아이템 집합

- {noodle, egg}, {noodle, cola}, {noodle, rice}, {noodle, tuna}, {egg, cola}, {egg, tuna}

- ✓ 조건을 만족하지 못하는 2개짜리 아이템 집합

- {egg, rice}, {cola, rice}, {cola, tuna}, {rice, tuna}

# 연관규칙분석

- 연관규칙분석 Step 2: 빈발 집합 생성

✓ Step 2-N: 더 이상 최소 지지도 조건을 만족하는 아이템 집합이 없을 때까지 아이템 집합의 크기를 1씩 증가시키면서 반복 수행

Set-size	Item 1	Item 2	Item 3	...	Item 6
1	noodle				
1	egg				
1	cola				
1	rice				
1	tuna				
2	noodle	egg			
2	noodle	cola			
2	noodle	rice			
...	...	...			

# 연관규칙분석

- 연관규칙분석 Step 3: 규칙 평가 수행

- ✓ 빈발 아이템 집합들로 생성한 모든 경우의 수에 대해 신뢰도 Confidence와 향상도 lift를 계산
- ✓ 예시: “라면을 사면 계란을 산다”

$$\text{support}(\text{noodle}) = P(\text{noodle}) = \frac{8}{10}, \quad \text{support}(\text{egg}) = P(\text{egg}) = \frac{5}{10}$$

$$\text{confidence}(\text{noodle} \rightarrow \text{egg}) = \frac{P(\text{noodle}, \text{egg})}{P(\text{noodle})} = \frac{4/10}{8/10} = 0.5(50\%)$$

$$\begin{aligned} \text{lift}(\text{noodle} \rightarrow \text{egg}) &= \frac{\text{confidence}(\text{noodle} \rightarrow \text{egg})}{\text{support}(\text{egg})} = \frac{\frac{P(\text{noodle}, \text{egg})}{P(\text{noodle})}}{P(\text{egg})} = \frac{P(\text{noodle}, \text{egg})}{P(\text{noodle}) \times P(\text{egg})} \\ &= \frac{\frac{4}{10}}{\frac{8}{10} \times \frac{5}{10}} = 1 \end{aligned}$$



# 연관규칙분석

- 최종 결과

✓ 기준 지지도: 20%, 기준 신뢰도(optional): 70%

Rule #	Antecedent (a)	Consequent	Support	Confidence	Lift
1	tuna=>	egg, noodle	2	100	2.5
2	tuna=>	egg	2	100	2
3	noodle, tuna=>	egg	2	100	2
4	rice=>	noodle	3	100	1.25
5	egg, tuna=>	noodle	2	100	1.25
6	tuna=>	noodle	2	100	1.25
7	cola=>	noodle	5	80	1
8	egg=>	noodle	5	80	1

# 연관규칙분석

- 연관규칙분석 요약

- ✓ 트랜잭션 데이터베이스에 존재하는 아이템 집합들 간의 연관성을 나타내는 규칙을 생성하는 분석 기법
- ✓ 다양한 분야의 추천 시스템 구축에 널리 사용됨
- ✓ 전체 규칙을 모두 생성하는 것이 비효율적이기 때문에 효율적인 빈발 집합을 찾아내는 A Priori 알고리즘을 사용
- ✓ 규칙의 효용성은 지지도, 신뢰도, 향상도의 세 가지를 이용하여 평가
- ✓ 규칙 1:  $A \rightarrow B$ 와 규칙 2:  $C \rightarrow D$ 에 대해 지지도, 신뢰도, 향상도가 모두 클 경우에만 규칙 1이 규칙 2보다 효과적인 규칙으로 결론지을 수 있음

# AGENDA

- 01 Recommendation Systems
- 02 Association Rule Mining
- 03 Collaborative Filtering: k-Nearest Neighbor
- 04 Matrix Factorization-based Method

# k-Nearest Neighbor

- 배경

類類相從 近墨者黑

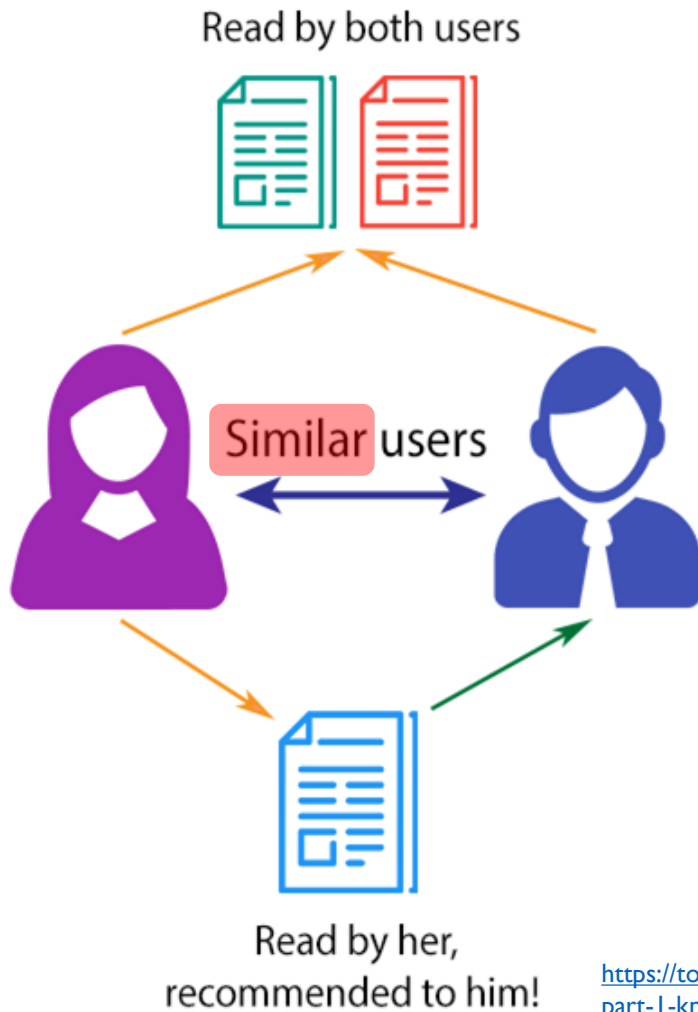
“Birds of a feather flock together”



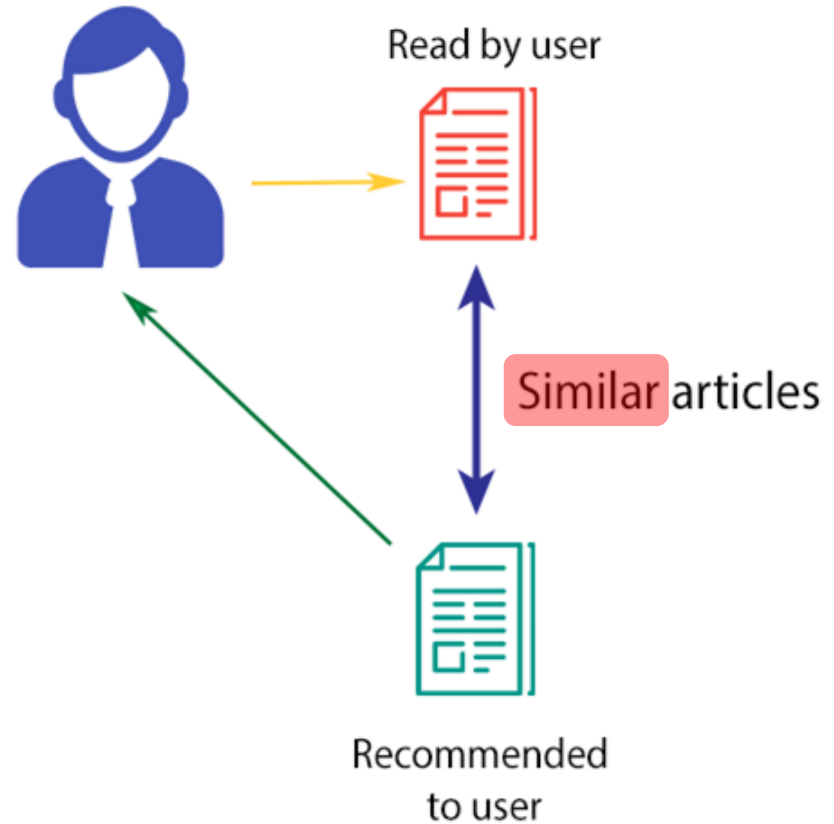
# k-Nearest Neighbor

- 배경

[User-based Collaborative Filtering]



[Item-based Collaborative Filtering]



# k-Nearest Neighbor

- 유사한 사용자/아이템을 찾기 위해서는 이들을 특정 공간 상의 한 점<sup>point</sup>으로 표현해야 함
  - ✓ 고전적 방식: 사용자와 아이템에 대한 속성을 정의하고 해당 속성에 값을 부여하여 특정 차원의 벡터를 생성

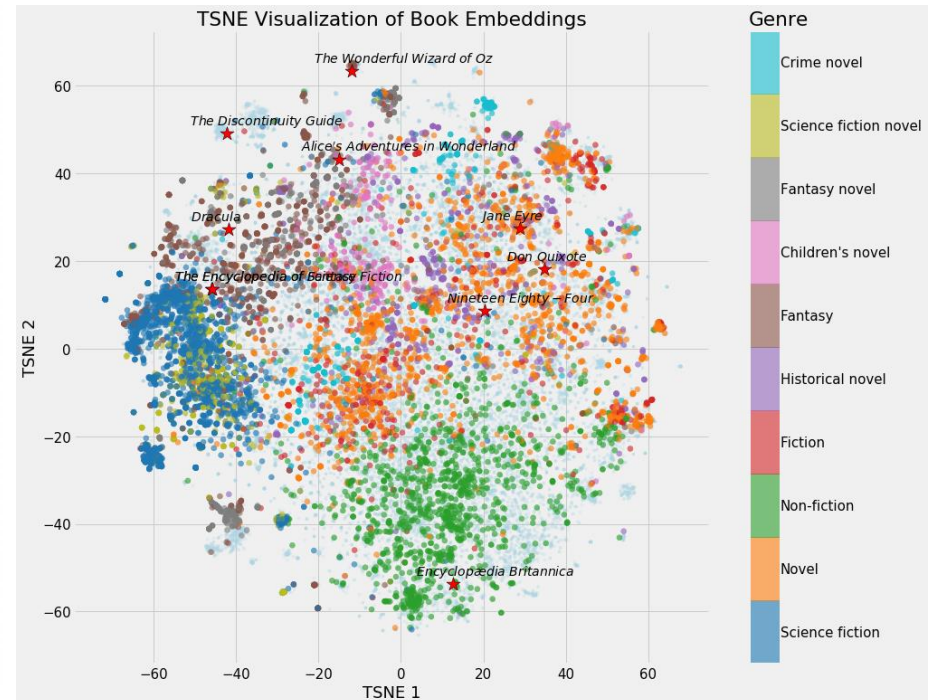
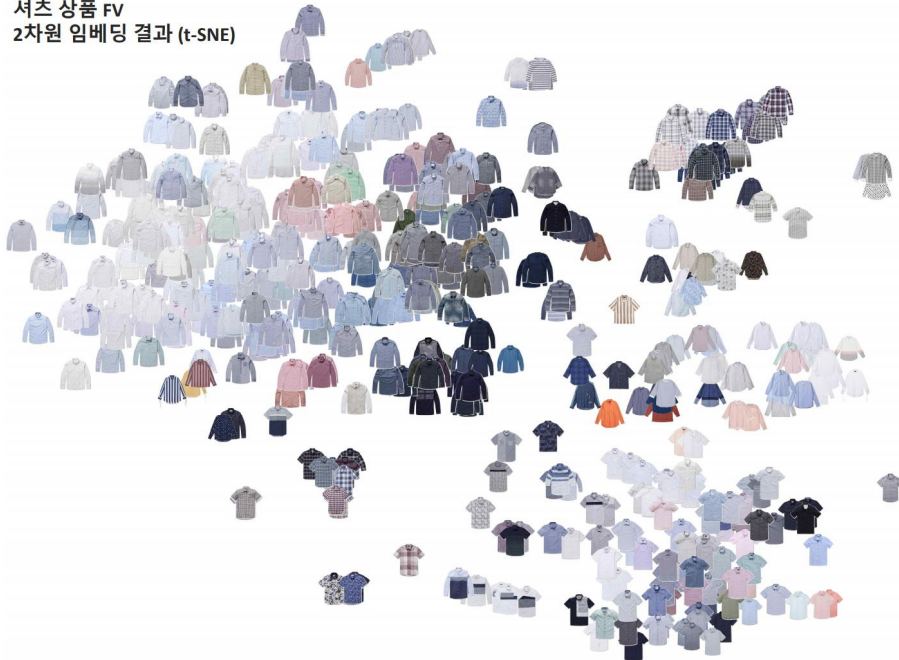




# k-Nearest Neighbor

- 유사한 사용자/아이템을 찾기 위해서는 이들을 특정 공간 상의 한 점<sup>point</sup>으로 표현해야 함
  - ✓ 최근 트렌드: 사용자나 아이템의 의미 관계<sup>semantic relationship</sup>가 보존되는 분산 표상<sup>distributed representation</sup>을 딥러닝<sup>deep learning</sup> 모델로 학습

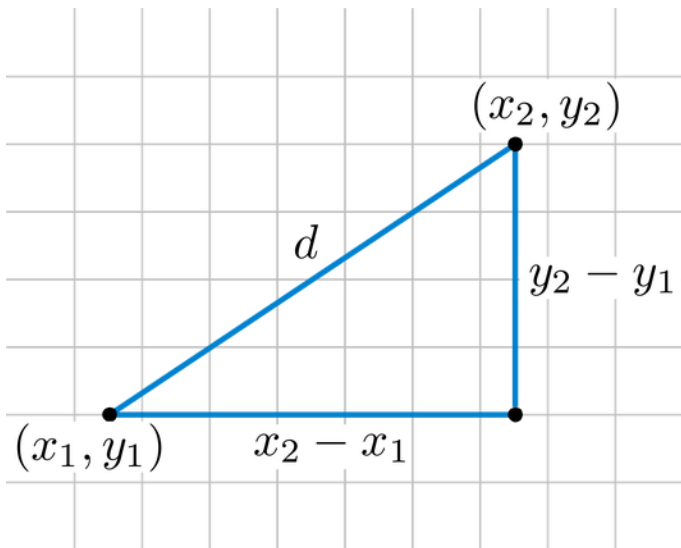
셔츠 상품 FV  
2차원 임베딩 결과 (t-SNE)



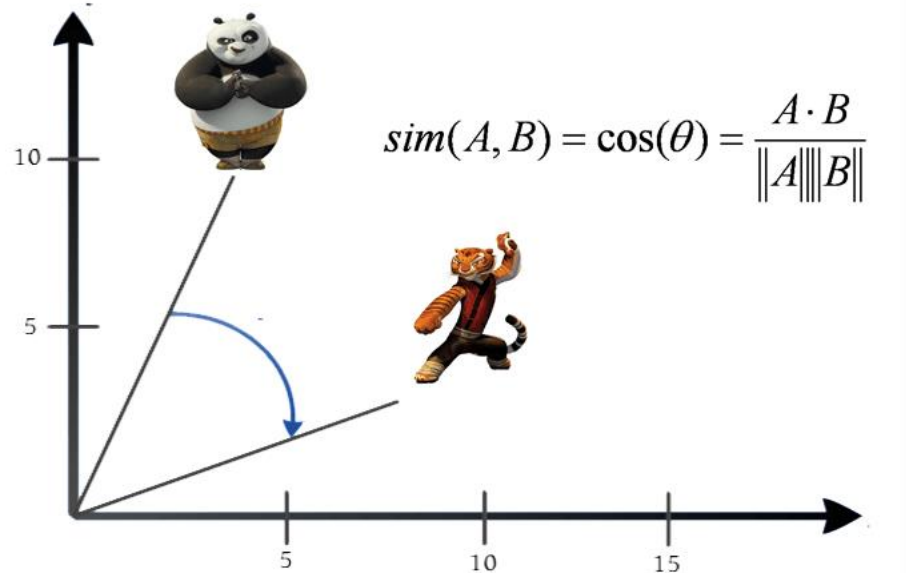
# k-Nearest Neighbor

- 공간 상에서 유사한 객체를 어떻게 찾을 것인가?
  - ✓ 유사도<sup>similarity</sup> 측정의 문제
  - ✓ 유사도 지표를 사용하거나 거리<sup>distance</sup> 지표를 사용한 뒤 거리에 반비례하도록 유사도를 정의

[Euclidean distance]

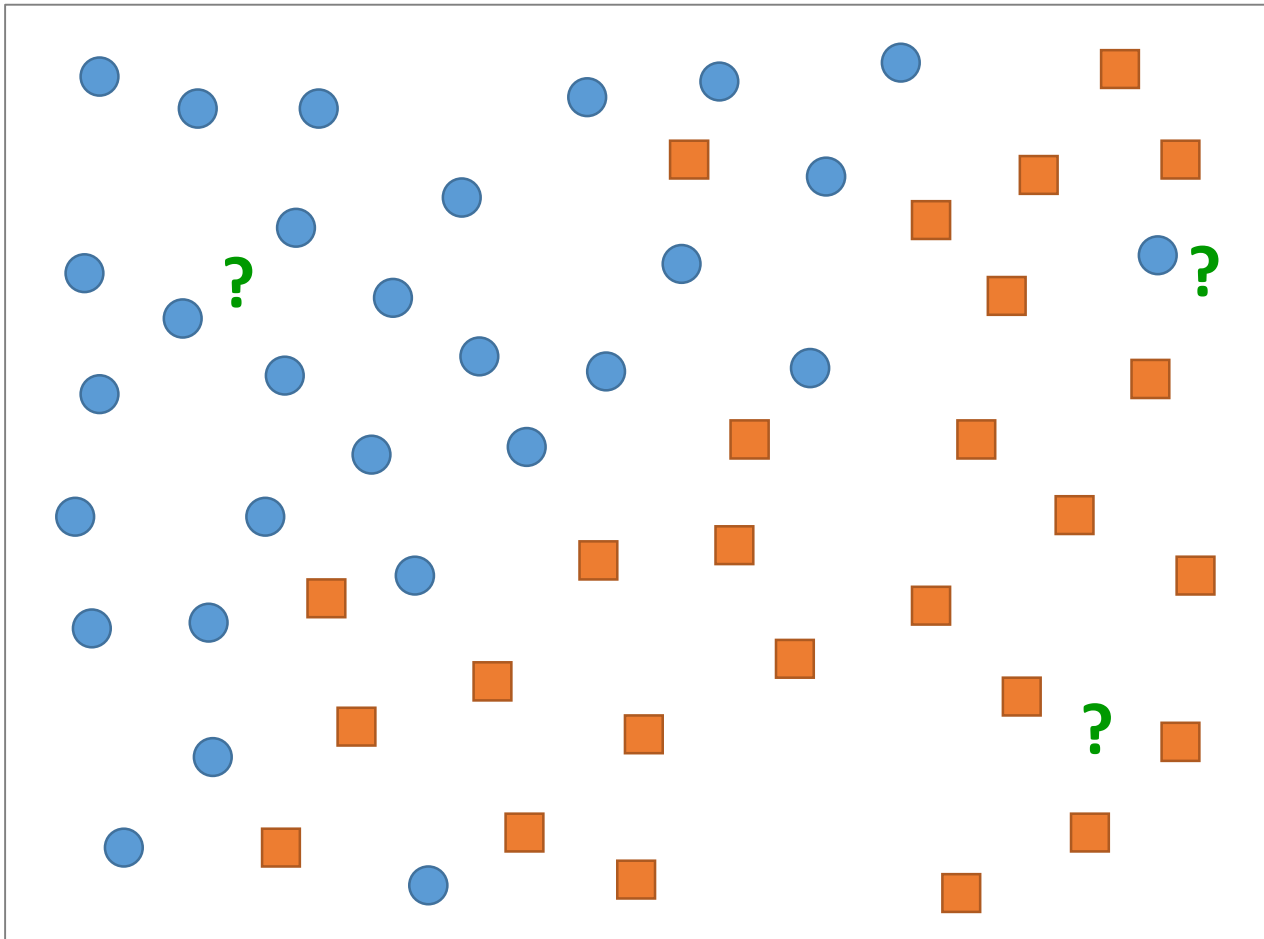


[Cosine similarity]



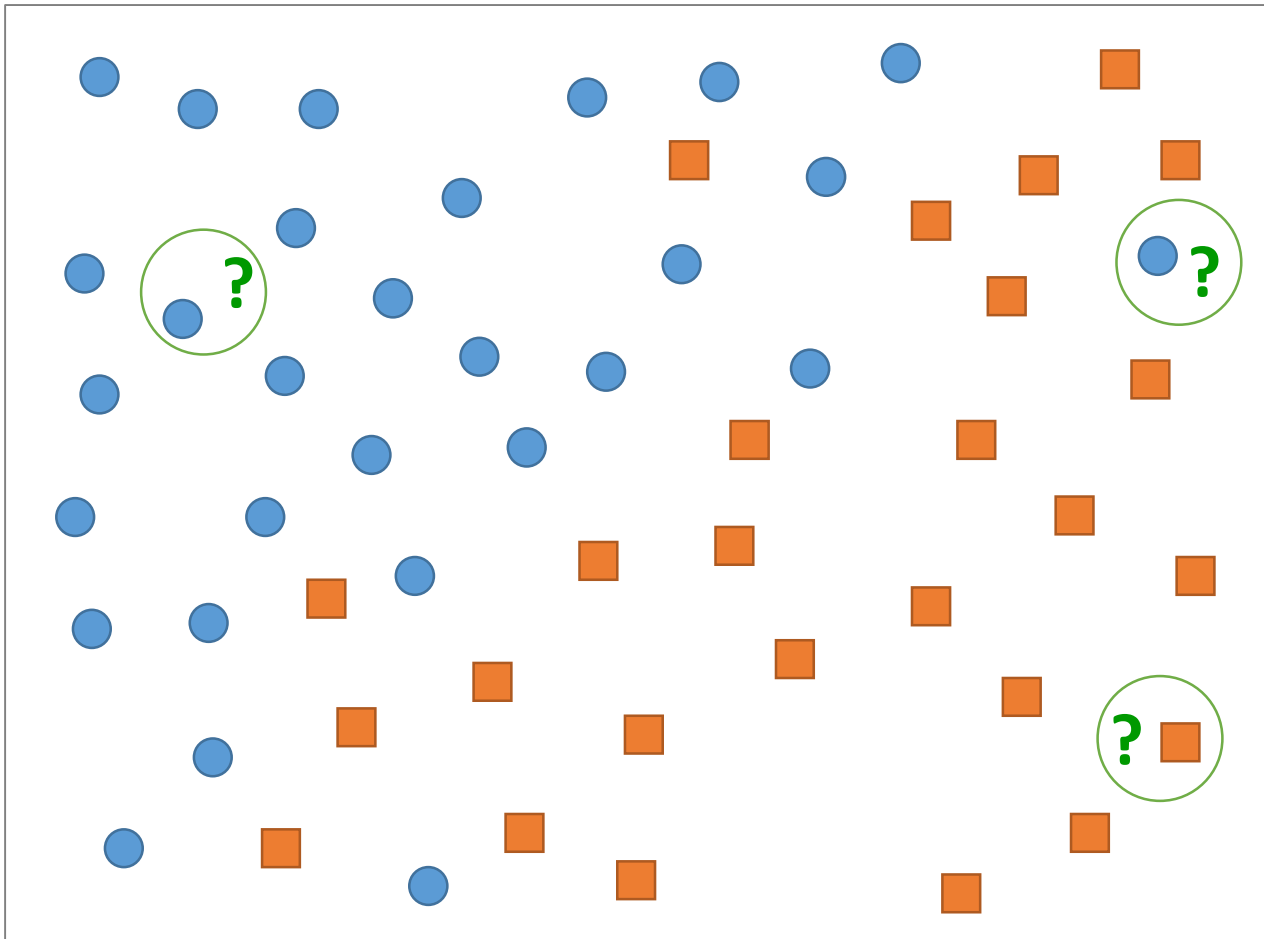
# k-Nearest Neighbor

- 유사한 객체를 몇 개나 참고하여 최종 결론을 도출할 것인가?
  - ✓ 아래의 예시에서 물음표들은 동그라미인가 네모인가?



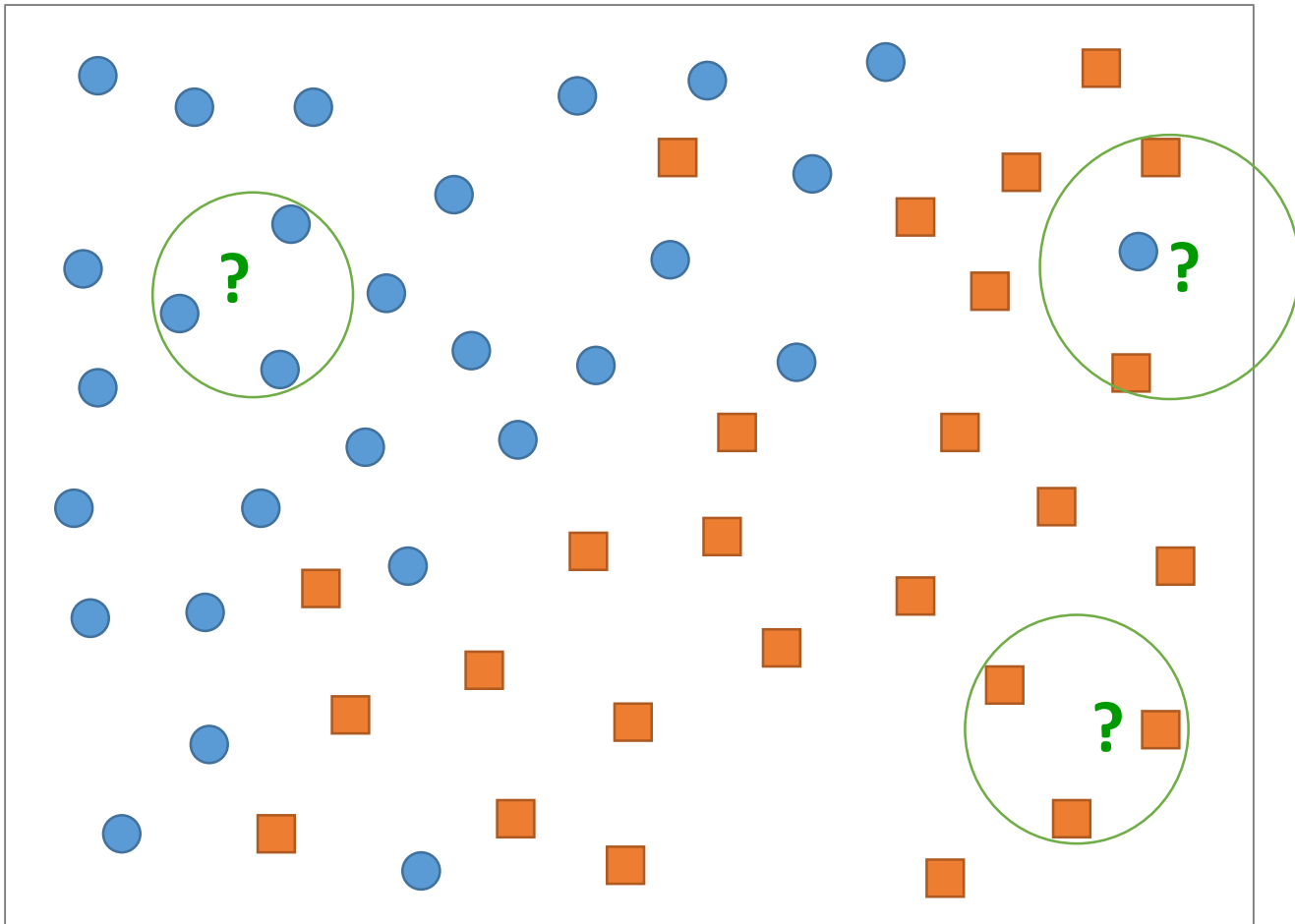
# k-Nearest Neighbor

- 유사한 객체를 몇 개나 참고하여 최종 결론을 도출할 것인가?
  - ✓ 가장 가까운 객체 하나만 참고하면...



# k-Nearest Neighbor

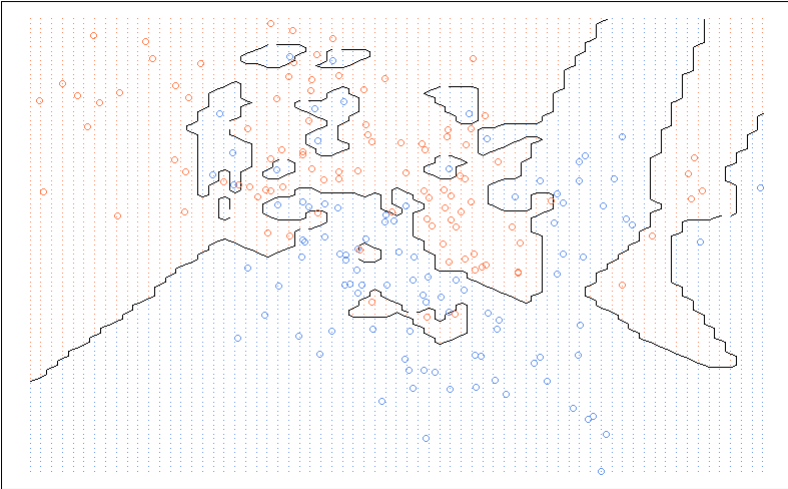
- 유사한 객체를 몇 개나 참고하여 최종 결론을 도출할 것인가?
  - ✓ 가까운 세 개의 객체들을 참고하면...



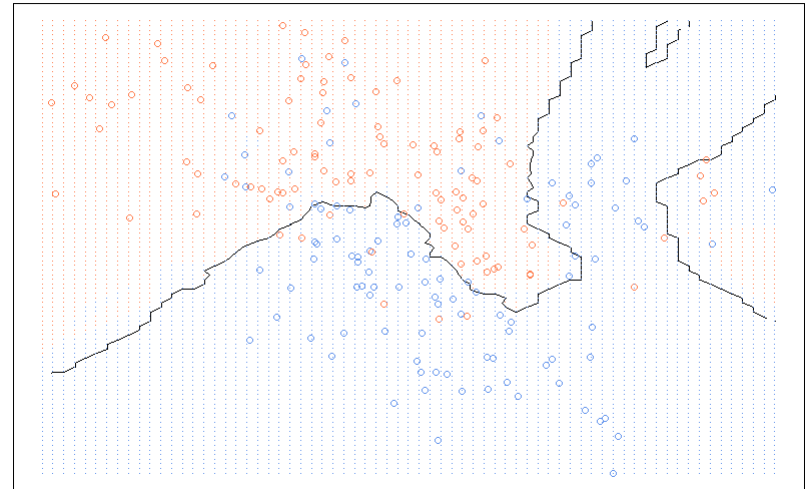
# k-Nearest Neighbor

- 이웃의 수  $k$ 의 효과

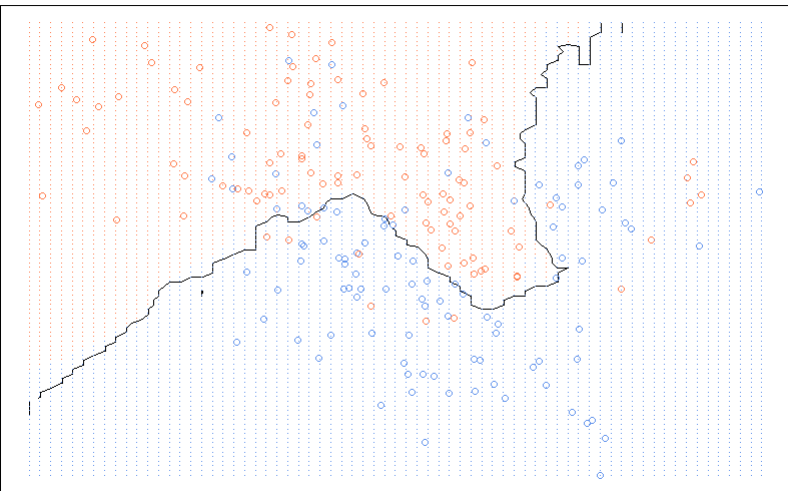
1-nearest neighbour



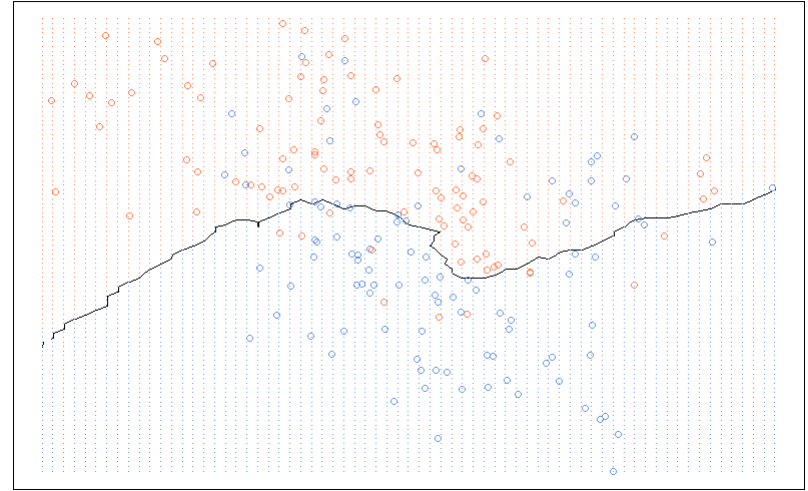
10-nearest neighbour



20-nearest neighbour



50-nearest neighbour





# k-Nearest Neighbor

- 이웃의 수  $k$ 의 효과

- ✓ 만일  $k$ 가 매우 작으면 노이즈에 민감한 과적합의 우려가 있음 (highly locally sensitive, over-fitting)
- ✓ 만일  $k$ 가 매우 크면 지역적 구조를 파악할 수 있는 능력을 잃게 됨(lose the ability to capture the local structure)
- ✓ 적절한  $k$ 를 찾아내는 것이 우수한  $k$ -인접이웃 모델을 만드는 데 필수적인 요소임
- ✓ 검증 데이터에 대한 에러가 가장 낮은  $k$ 값을 선택

# k-Nearest Neighbor

- 이웃의 정보를 어떻게 결합할 것인가?

✓ 예제: 한 사람의 키/몸무게/성별로부터 체지방률을 추정

개체	키	몸무게	성별(F=1)	체지방률
1	187	93	0	15
2	165	51	1	25
3	174	68	0	14
4	156	48	1	29
...	...	...	...	...
N	168	59	0	12

# k-Nearest Neighbor

- 이웃의 정보를 어떻게 결합할 것인가?

✓ 단순 평균 Simple average vs. 가중 평균 Weighted average

For a new data  
**X**

이웃	체지방률	거리	1/거리	가중치
N1	15.4	1	1.00	0.44
N2	17.2	2	0.50	0.22
N3	12.3	3	0.33	0.15
N4	11.5	4	0.25	0.11
N5	10.9	5	0.20	0.08

- 단순 평균 이용

- X의 체지방률 =  $(15.4+17.2+12.3+11.5+10.9)/5 = 13.46$

- 가중 평균 이용

- X의 체지방률 =  $0.44*15.4+0.22*17.2+0.15*12.3+0.11*11.5+0.08*10.9 = 14.54$

# k-Nearest Neighbor

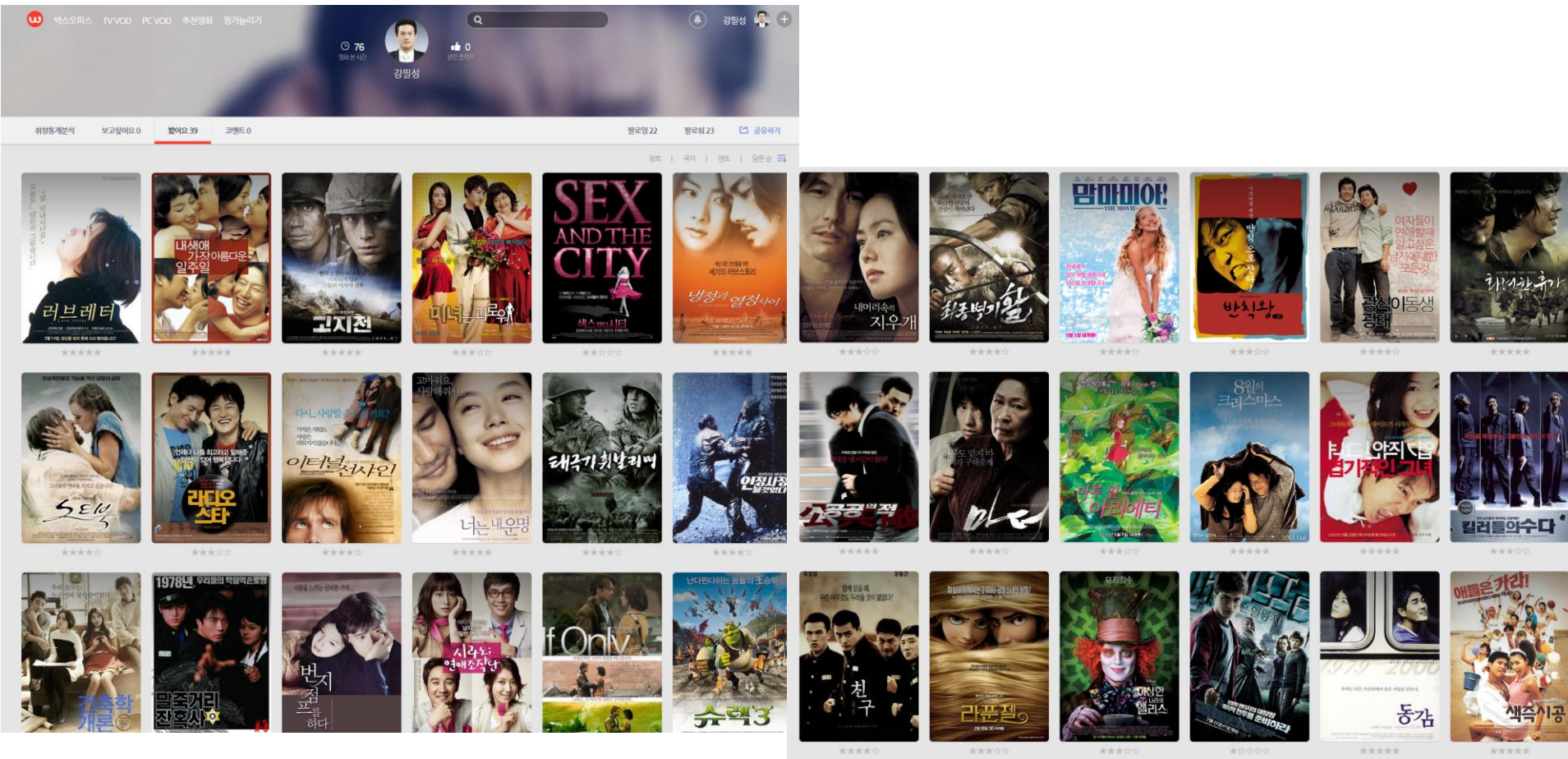
- 영화 추천 시스템: 협업 필터링

	영화 1	영화 2	영화 3	영화 4	영화 5	...	영화 D
강필성	10	9	5	6	9	...	? 9

User	영화 1	영화 2	영화 3	영화 4	영화 5	...	영화 D
1	10	8	4	7	10	...	10
2	8	5	7	9	4	...	5
3	10	9	6	5	8	...	9
4	4	2	10	10	5	...	3
5	7	4	6	8	5	...	3
6	5	2	10	10	10	...	6
7	10	8	6	6	8	...	8
...	...	...	...	...	...	...	...
N	5	7	1	5	4	...	7

# k-Nearest Neighbor

- 예시: 사용자 선호도 기반의 영화 추천 시스템



<http://watcha.net>



# k-Nearest Neighbor

## • 추천 영화 리스트

예상 별점이 가장 높은 영화 × 장르 | 국가 | 연도 | 추천이유

<p>4.2</p> <p>세상에서 가장 아름다운 이별</p> <p>바깥은 지친, 함께 할 걸 그랬습니다</p> <p>예상 별점 4.3개 ★★★★★</p> <p>《내 생애 가장 아름다운 일주일》과 비슷해요.</p>	<p>4.3</p> <p>괴물</p> <p>나를 웃게 하느니, 화를 입는다</p> <p>예상 별점 4.3개 ★★★★★</p> <p>《8월의 크리스마스》와 비슷해요.</p>	<p>4.2</p> <p>괴물</p> <p>연애가 만능이긴 않은 사랑</p> <p>예상 별점 4.0개 ★★★★★</p> <p>《8월의 크리스마스》와 비슷해요.</p>	<p>4.0</p> <p>조수만 더 가까이</p> <p>사랑은 나의 옆에 불구야</p> <p>예상 별점 4.3개 ★★★★★</p> <p>《시리노 연애조각단》과 비슷해요.</p>	<p>4.3</p> <p>소원</p> <p>가장 아름다운 소녀의 비밀, 가장 따뜻한 아버지의</p> <p>예상 별점 4.3개 ★★★★★</p> <p>《시리노 연애조각단》과 비슷해요.</p>
<p>4.2</p> <p>신세계</p> <p>세 남자가 가고 싶었던 세로 다른</p> <p>예상 별점 4.2개 ★★★★★</p> <p>《친구》와 비슷해요.</p>	<p>4.0</p> <p>내 사랑 내 고향</p> <p>말할 수도, 울릴 수도 없는 그가 당신을 울립니다</p> <p>예상 별점 4.0개 ★★★★★</p> <p>《너는 내 운명》과 비슷해요.</p>	<p>4.0</p> <p>한 공주</p> <p>한 공주</p> <p>예상 별점 4.0개 ★★★★★</p> <p>《너는 내 운명》과 비슷해요.</p>	<p>4.3</p> <p>은하수</p> <p>은하수</p> <p>예상 별점 4.3개 ★★★★★</p> <p>《시리노 연애조각단》과 비슷해요.</p>	<p>4.8</p> <p>아는여자</p> <p>아는여자</p> <p>예상 별점 4.8개 ★★★★★</p> <p>《시리노 연애조각단》과 비슷해요.</p>
<p>4.6</p> <p>Before Sunset</p> <p>Before Sunset</p> <p>예상 별점 4.6개 ★★★★★</p> <p>《이터널 선샤인》의 2편과 비슷해요.</p>	<p>4.0</p> <p>만족</p> <p>만족</p> <p>예상 별점 4.0개 ★★★★★</p> <p>《우리의 행복한 시간》과 비슷해요.</p>	<p>4.0</p> <p>세레니티</p> <p>세레니티</p> <p>예상 별점 4.0개 ★★★★★</p> <p>《노트북》의 1편과 비슷해요.</p>	<p>4.0</p> <p>Resident Evil</p> <p>Resident Evil</p> <p>예상 별점 4.0개 ★★★★★</p> <p>《노트북》의 1편과 비슷해요.</p>	<p>4.0</p> <p>running on empty</p> <p>running on empty</p> <p>예상 별점 4.0개 ★★★★★</p> <p>《노트북》의 1편과 비슷해요.</p>

# AGENDA

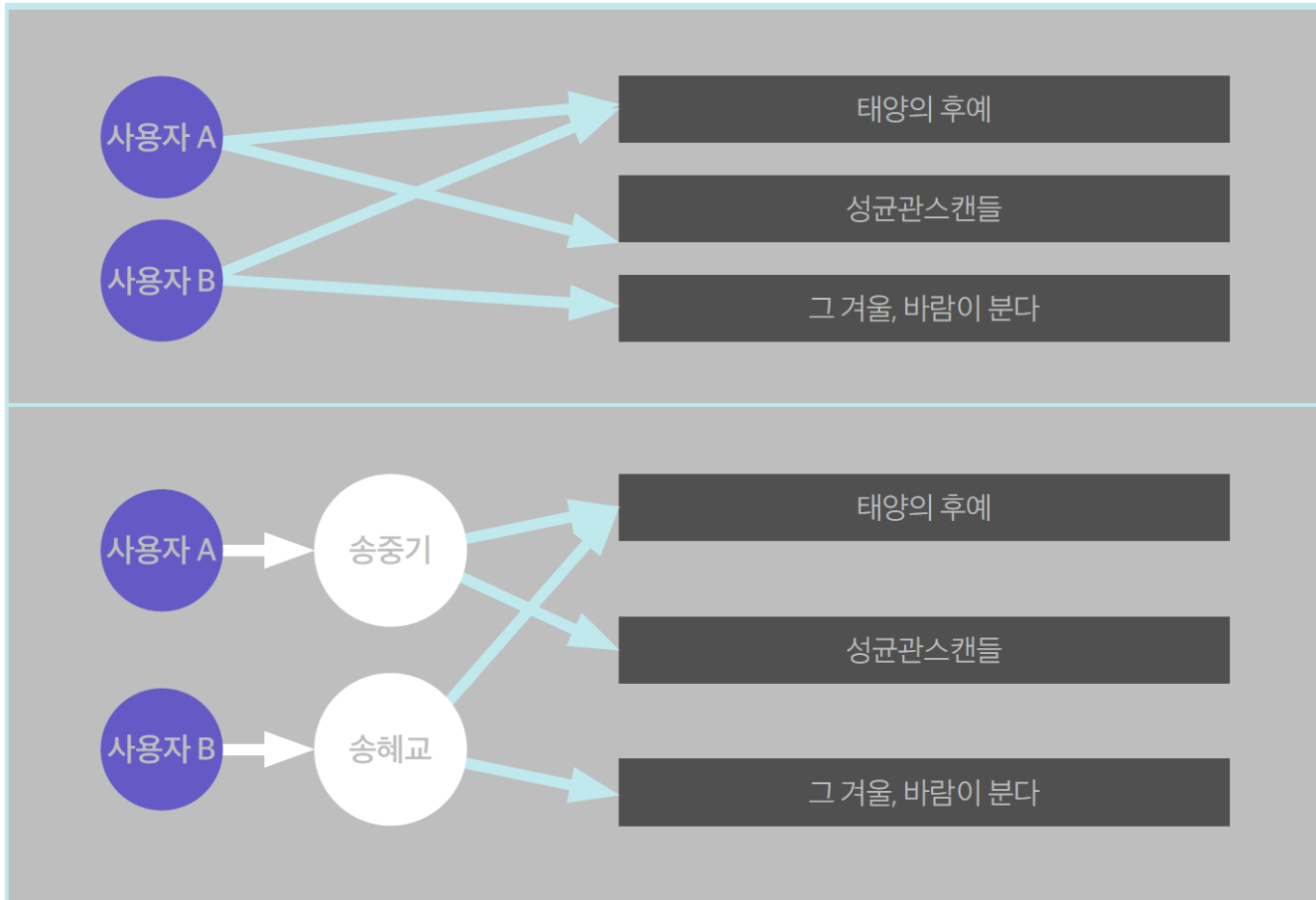
- 01 Recommendation Systems
- 02 Association Rule Mining
- 03 Collaborative Filtering: k-Nearest Neighbor
- 04 **Matrix Factorization-based Method**



# Matrix Factorization-based Method

- 잠재 요인 분석(Latent Factor Analysis)의 목적

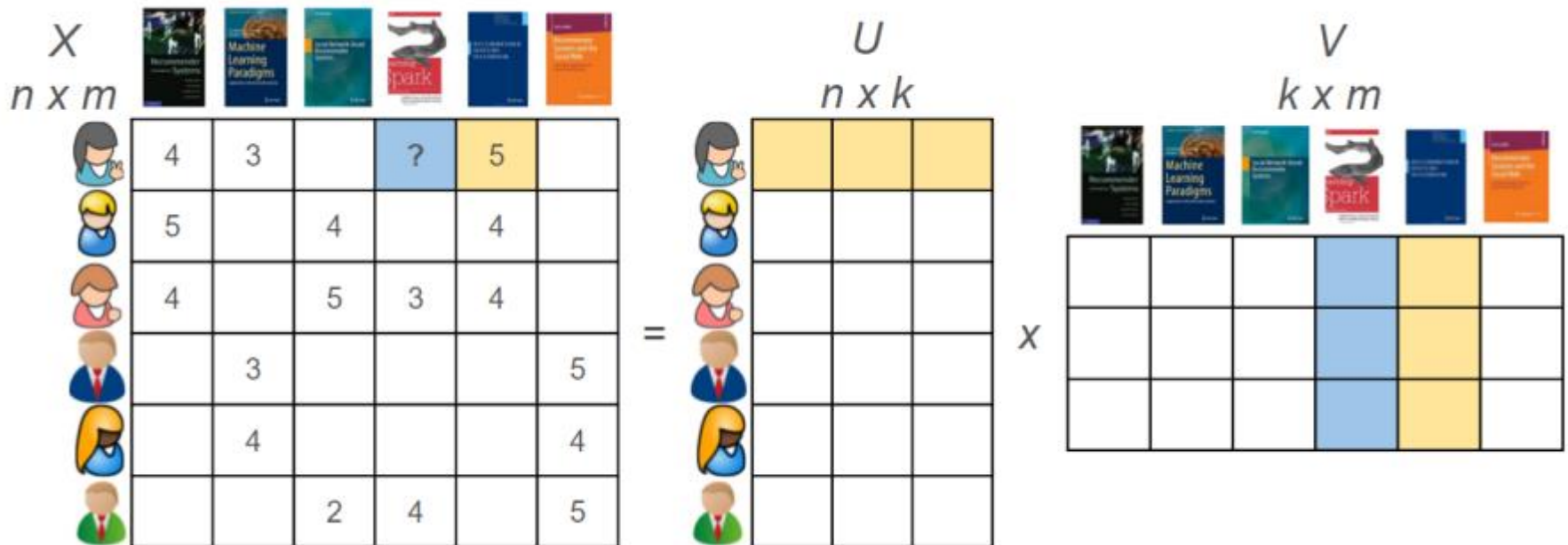
✓ 사람들의 선호도 속에 숨겨진 진짜 요인(factor)를 찾자!



# Matrix Factorization-based Method

- 수리적 배경

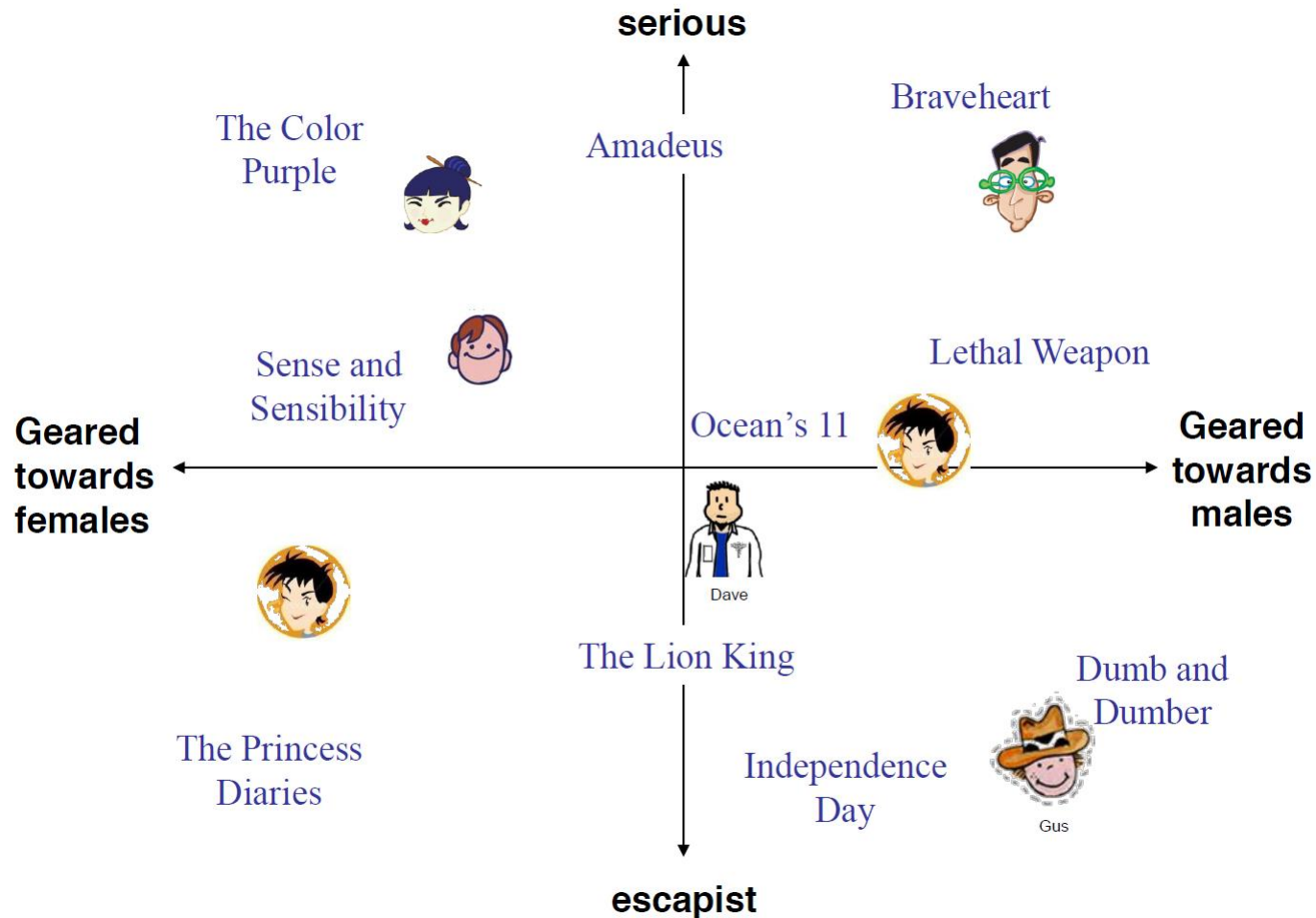
- ✓ 밀도가 낮은 큰 행렬(sparse matrix)은 보다 작은 크기의 밀도가 높은 행렬의 결합으로 표현될 수 있음



# Matrix Factorization-based Method

- 장점

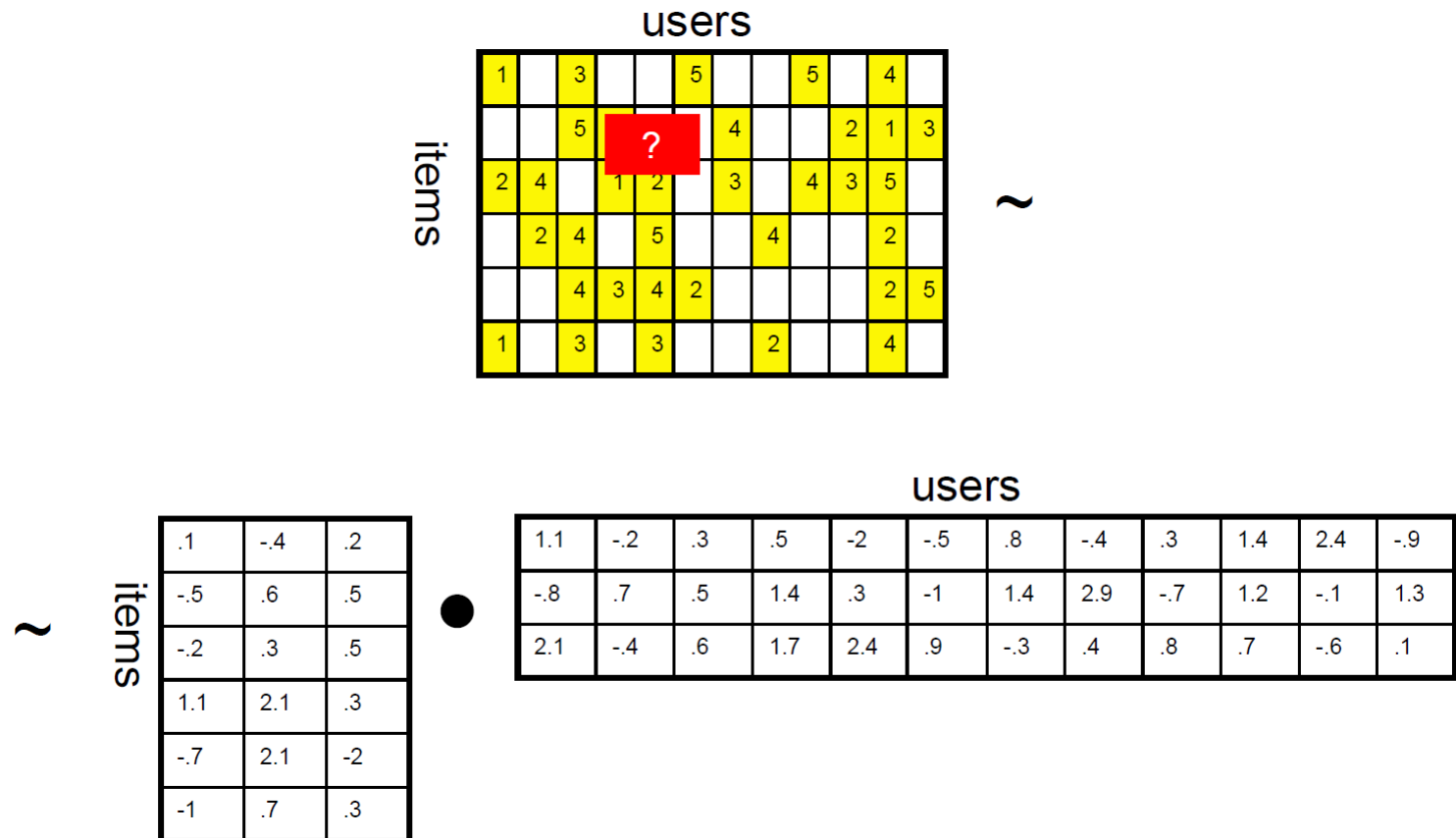
- ✓ 사용자와 아이템을 한 공간에 함께 표현할 수 있음



# Matrix Factorization-based Method

- 장점

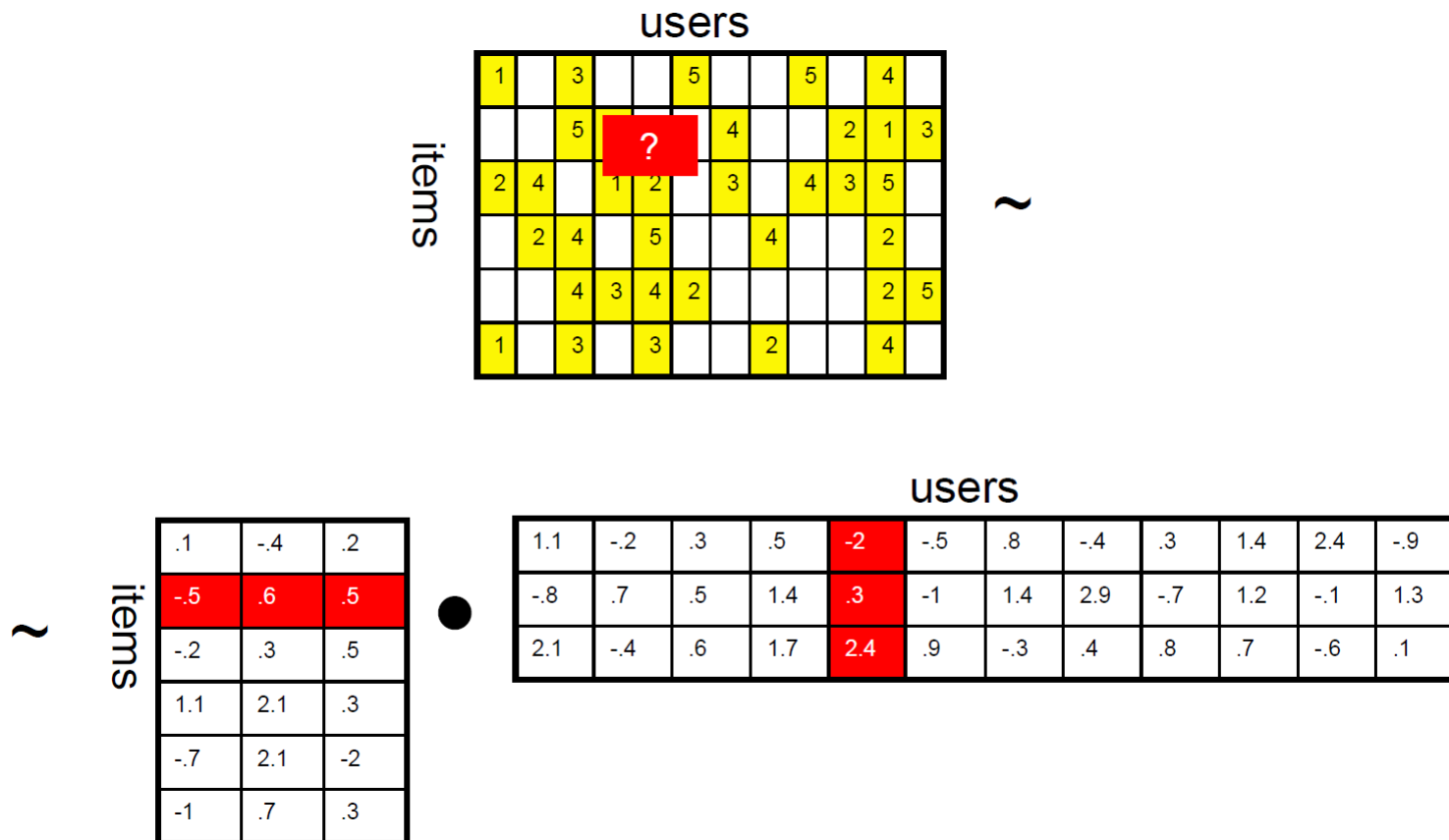
✓ 원래 행렬에서 값이 없었던 셀을 추정할 수 있음



# Matrix Factorization-based Method

- 장점

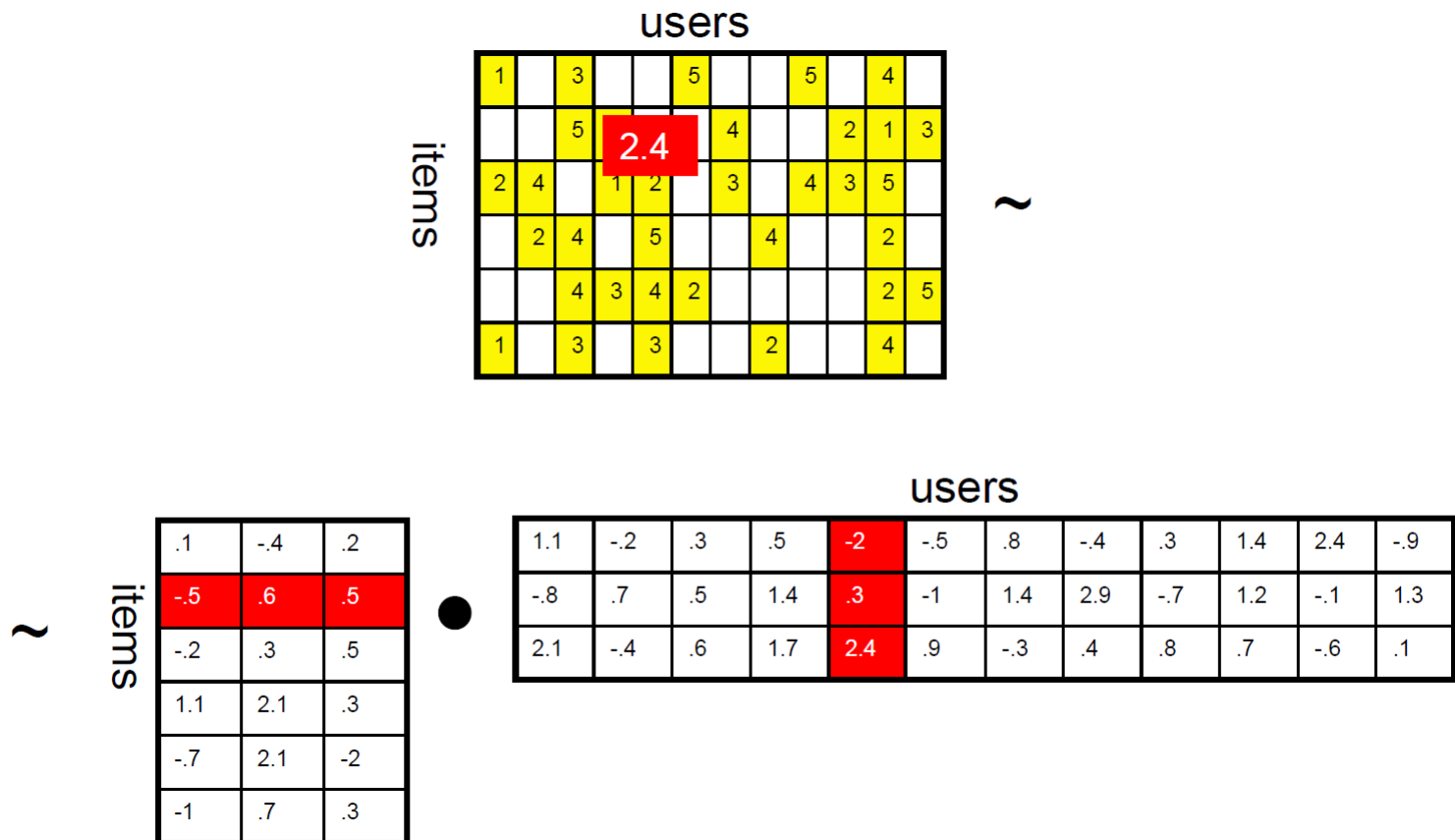
✓ 원래 행렬에서 값이 없었던 셀을 추정할 수 있음



# Matrix Factorization-based Method

- 장점

✓ 원래 행렬에서 값이 없었던 셀을 추정할 수 있음



# Matrix Factorization-based Method

- User-Item rating 행렬에서 어떻게 잠재 사용자 행렬<sup>Latent user matrix</sup>과 잠재 아이템 행렬<sup>Latent item matrix</sup>를 찾아내는가?

$$\hat{r}_{ui} = \mu + b_u + b_i + q_i^T p_u$$

User u가  
Item i에  
매기는 평점

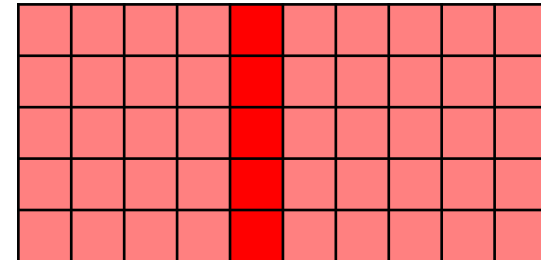
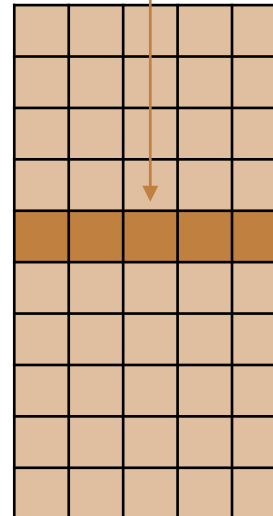
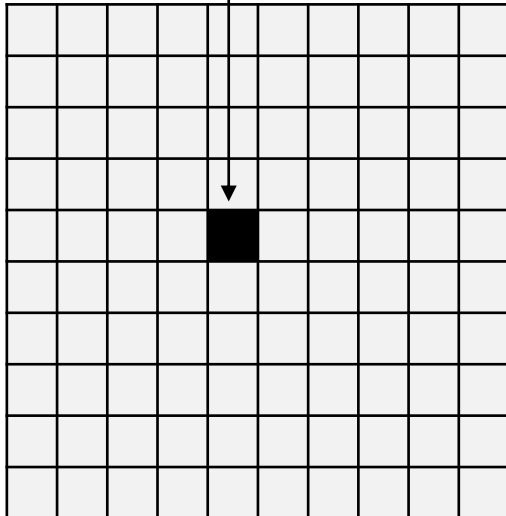
평점의  
전체 평균

User u의  
Bias

Item i의  
Bias

Item i의  
잠재 벡터

User u의  
잠재 벡터



# Matrix Factorization-based Method

- User-Item rating 행렬에서 어떻게 잠재 사용자 행렬<sup>Latent user matrix</sup>과 잠재 아이템 행렬<sup>Latent item matrix</sup>를 찾아내는가?

✓ Rating이 존재하는 셀들에 대해 아래 식을 목적함수로 하는 최적해를 구함

$$\min \sum_{r_{ui} \in R_{train}} \left( r_{ui} - \hat{r}_{ui} \right)^2 + \lambda \left( b_i^2 + b_u^2 + ||q_i||^2 + ||p_u||^2 \right)$$

최대한 실제 Rating과 추정된 Rating이 같도록 하라

과적합<sup>Overfitting</sup> 방지를 위한 Regularization 장치



# Matrix Factorization-based Method

$$\min \sum_{r_{ui} \in R_{train}} \underbrace{(r_{ui} - \hat{r}_{ui})^2}_{\substack{\text{최대한 실제 Rating과} \\ \text{추정된 Rating이 같도록} \\ \text{하라}}} + \underbrace{\lambda(b_i^2 + b_u^2 + \|q_i\|^2 + \|p_u\|^2)}_{\substack{\text{과적합(Overfitting) 방지를} \\ \text{위한 Regularization 장치}}}$$

- Stochastic gradient descent 방식을 사용하여 최적해를 구할 수 있음

$$b_u \leftarrow b_u + \gamma(e_{ui} - \lambda b_u)$$

$$b_i \leftarrow b_i + \gamma(e_{ui} - \lambda b_i)$$

$$p_u \leftarrow p_u + \gamma(e_{ui} \cdot q_i - \lambda p_u)$$

$$q_i \leftarrow q_i + \gamma(e_{ui} \cdot p_u - \lambda q_i)$$

$$e_{ui} = r_{ui} - \hat{r}_{ui}$$

