

# **Recommendation**

# 정보 홍수의 시대 (Information Overload)

정보의 과잉으로 인해 이용자가 모든 정보를 보고 가치의 유무를 판단할 수 없는 상태

# Information Filtering

여러 가지 항목 중 적당한 항목을 선택하는 기술

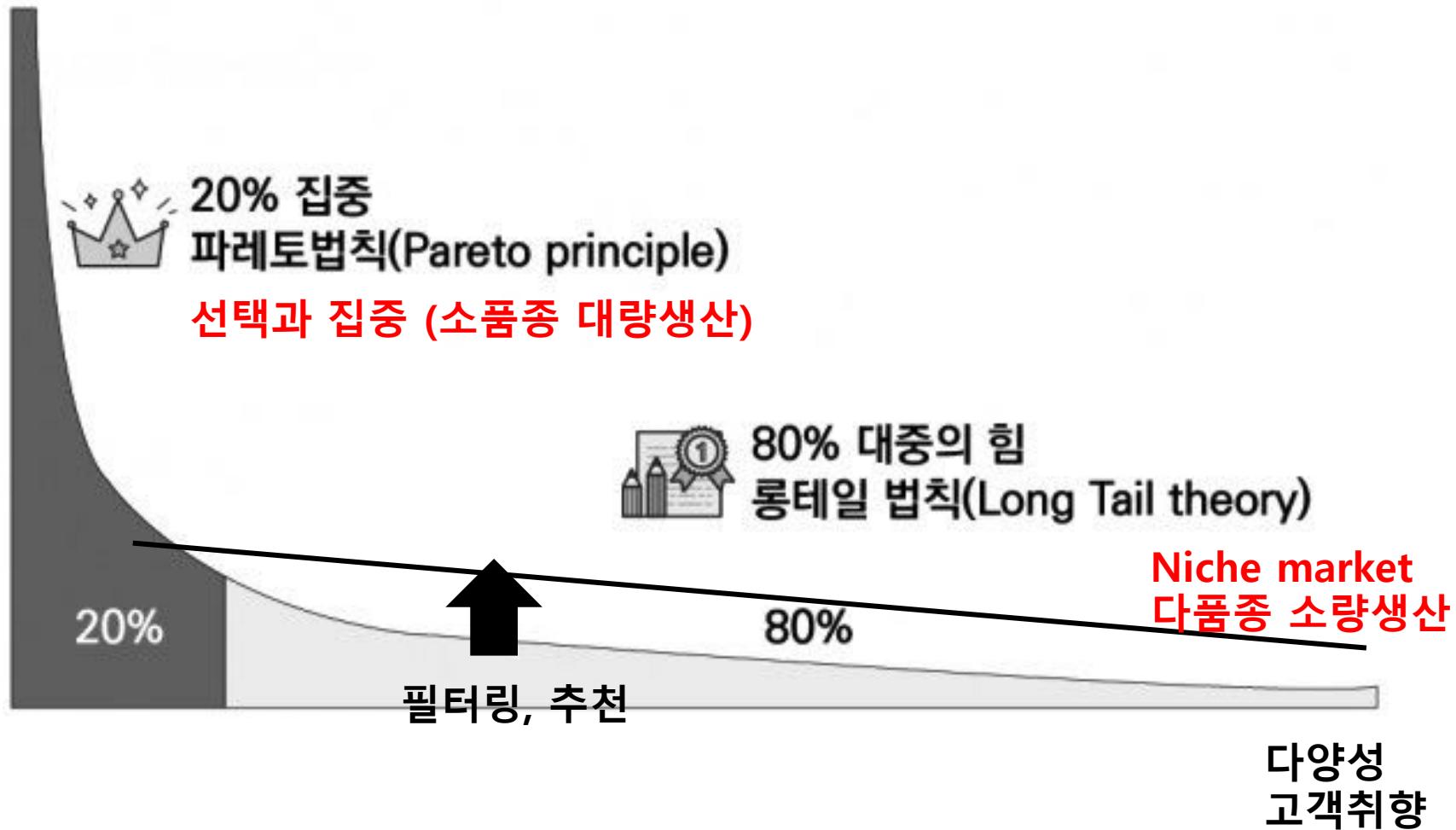
## 검색

이용자가 원하는 정보를 '검색어(Query)'를 통해 특정하여 찾아주는 방식

## 추천

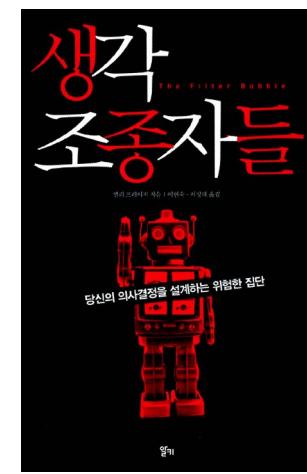
보다 능동적인 방식으로서 '검색어'와 같은 이용자의 명시적인 요청이 없어도 이용자의 평상시 서비스 사용 패턴(과거 데이터)을 기반으로 하여 이용자가 선호할만한 콘텐츠를 제공해주는 구조

수요  
판매량



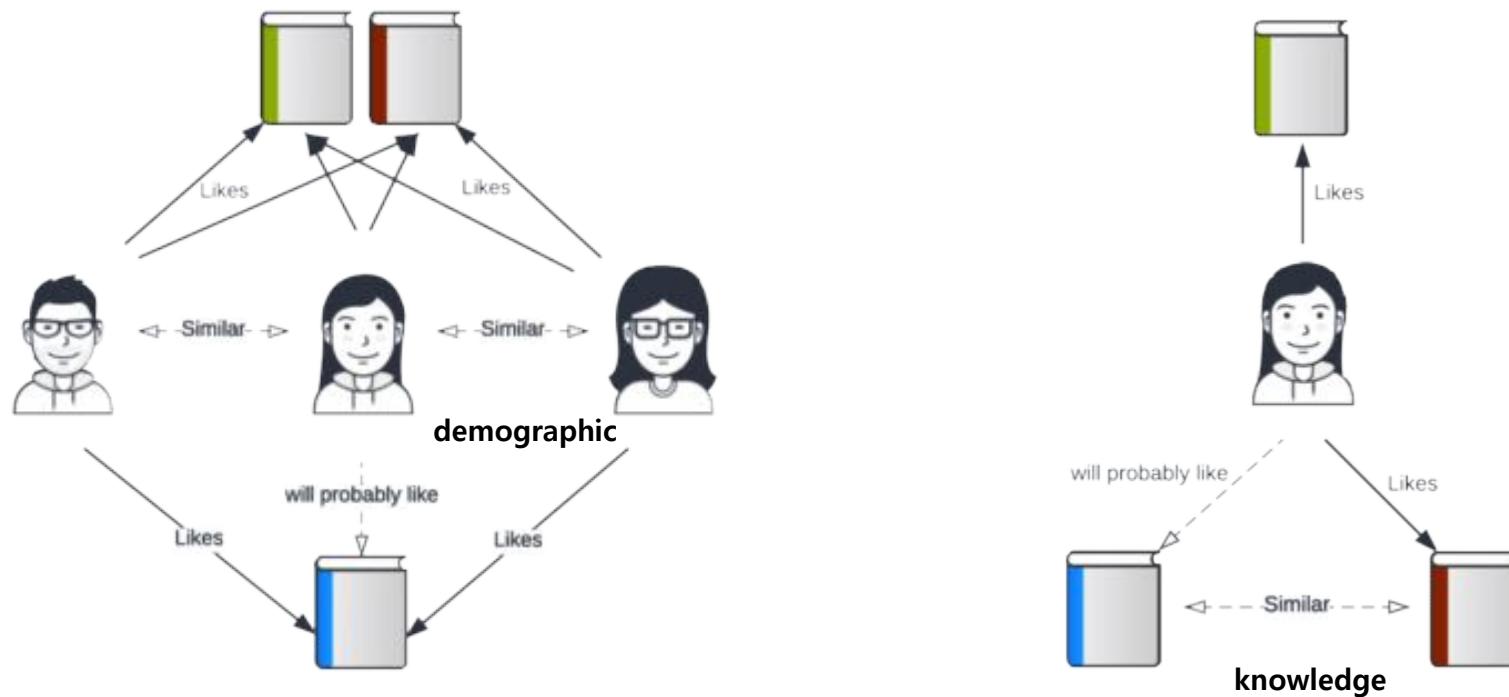
## ■ 필터 버블 (Filter Bubble)

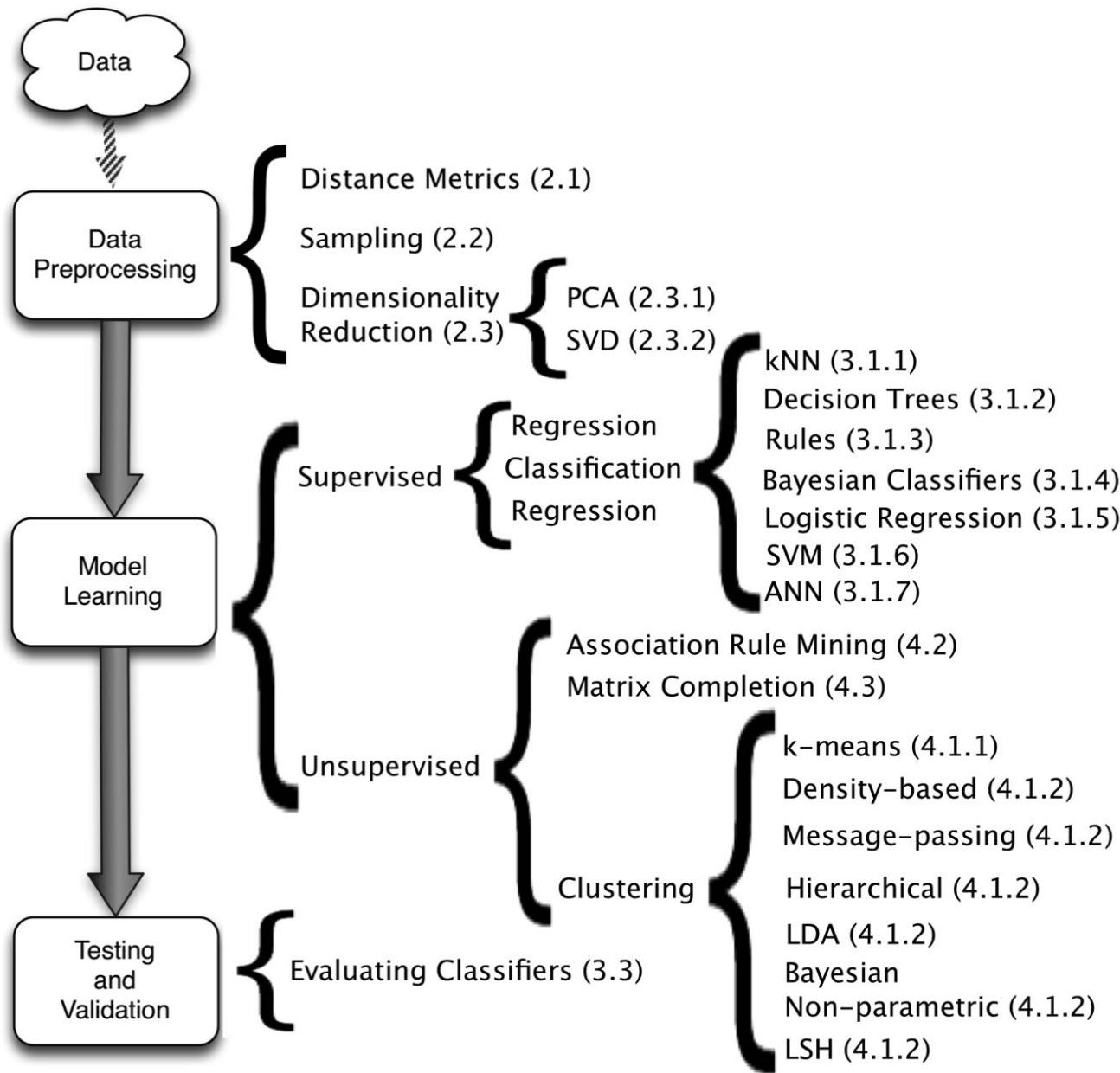
- 필터가 거품처럼 늘어남
- 미국 시민단체 무브온(Move on)의 엘리 프레이저(Eli Pariser)가 자신의 저서 '생각 조종자들(원제: The Filter Bubble)'에서 인터넷 시대의 위험성을 경고하면서 처음 사용한 용어.
- 개인 맞춤형 콘텐츠 추천 시스템은 넘쳐나는 정보 속에서 개인의 취향에 맞는 정보만 '필터링'해서 볼 수 있다는 점에서 편리하지만 필터링된 정보는 개인의 고정관념이나 편견을 강화할 수 있고, 제공된 정보에 의해 생각이 조작될 가능성이 있음
  - 보고싶은 정보만 보고, 보기 불편한 정보는 자동으로 건너뛰는 것이 기술적으로 가능해지면서 야기될 수 있는 정보의 편향적 제공은 극단적인 양극화와 같은 사회적 문제를 가져올 수도 있음
    - 가짜 뉴스, 실시간 검색어
- 2014년에 페이스북 연구진이 PNAS에 발표한 논문(**Experimental evidence of massive-scale emotional contagion through social networks**, Kramer et al. 2014)에 의하면 추천되는 정보에 따라서 사용자의 감정 조정 가능
- 1984 vs 멋진 신세계
- 편향, 스마트한 생각들



## ■ 추천 시스템 (Recommender systems or Recommendation Engines)

- 사용자가 선호할 만한 아이템을 추측함으로써 여러 가지 항목 중 사용자에게 적합한 특정 항목을 선택 (information filtering)하여 제공하는 시스템
- 최근의 기술 발전에 따라 여러 가지 새로운 기술이 사용되고 있지만 기본적인 추천 시스템은
  - 협업 필터링(Collaborative filtering)
  - 콘텐츠(내용) 기반 필터링 (Content-based filtering)





## ■ 협업 필터링

- 기존 사용자 행동 정보를 분석하여 해당 사용자와 비슷한 성향의 사용자들이 기존에 좋아했던 항목을 추천하는 기술
- **비슷한 패턴을 가진 사용자나 항목을 추출하는 기술이 핵심**
  - user
  - item
  - model
  - 행렬분해(Matrix Factorization),
  - k-최근접 이웃 알고리즘 (k-Nearest Neighbor algorithm; kNN) 등의 방법
- 협업 필터링을 위해서는 **기존 자료를 활용**
  - 협업 필터링은 사용자들이 자연스럽게 사이트를 사용하면서 검색을 하고, 항목을 보고, 구매한 내역을 사용
- 신선한 추천(**Serendipity** Recommendation)이 가능
- 다른 이용자들의 정보도 활용하기 때문에 내용 기반의 추천에 비해 이용자의 '행동 데이터'(Behavioral Data)가 상대적으로 많지 않더라도 추천이 가능
  - 이용자의 행동 데이터는 서비스에서 이용자가 하는 모든 행동을 기록한 정보
    - 뉴스 서비스의 경우, 이용자가 어떤 뉴스를 읽었으며, 해당 뉴스를 읽은 시간이 얼마인지, 그리고 이용자가 서비스를 주로 사용하는 시간 등이 중요한 행동 데이터로 취급
- 콘텐츠 메타 데이터를 사용하지 않아 언어 종속적인 텍스트 분석이 필요하지 않고, 복잡한 '특성 추출 및 선택'(Feature Extraction & Selection) 기술이 필요하지 않음

## ■ 문제점

### ○ 콜드 스타트(Cold-Start) 문제

- 기존에 많이 축적된 이용자의 행동 정보에만 의존하여 새로운 콘텐츠보다는 종래의 콘텐츠를 주로 추천하며, 데이터가 부족한 상황에서 추천 품질이 저하되는 '콜드 스타트' 문제를 내재
- 예를 들어 음악 서비스의 경우, 신곡이 발표되면 이를 추천할 수 있는 정보가 쌓일 때까지 추천이 어려워지는 것.

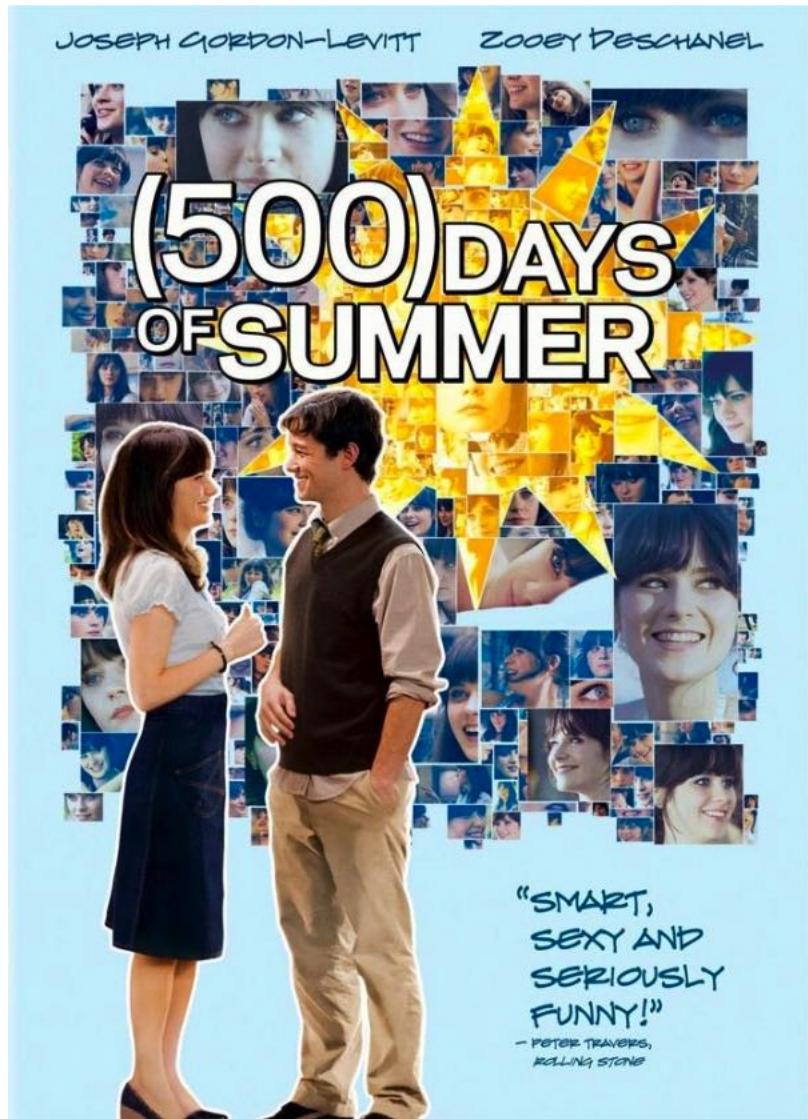
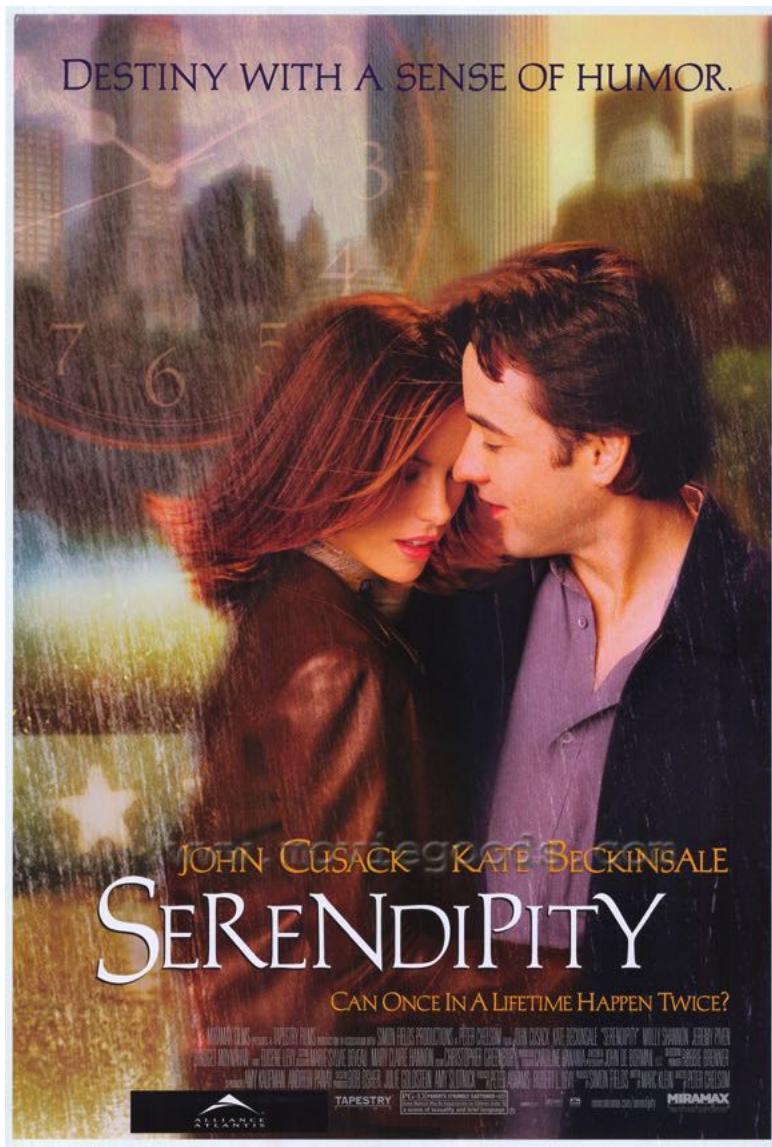
### ○ 롱테일(Long tail) 문제

- 시스템 항목이 많다 하더라도 사용자들은 소수의 인기 있는 항목에만 관심을 보이기 마련. 따라서 사용자들의 관심이 적은 다수의 항목은 추천을 위한 충분한 정보를 제공하지 못하는 경우가 많음. 이러한 비대칭적 쓸림 현상이 일반적이라는 사실은 크리스 앤더슨(Chris Anderson)이나 클레이 셔키(Clay Shirky) 등이 일찍이 밝힌 바 있음.
- 계산량이 비교적 많은 알고리즘이므로 사용자 수가 많은 경우 효율적으로 추천할 수 없음. 추천 시스템이 관리하는 항목이 많은 경우, 협업 필터링은 한계가 있을 수 있음
- 독특한 취향의 고객(gray sheep / black sheep) 문제

## ■ 콘텐츠 기반 필터링

- 협업 필터링이 사용자들의 행동 기록을 이용하는 반면, 항목 자체를 분석하여 추천 (특정 개인의 행동 정보)
  - 1) 각 콘텐츠의 메타 정보 (Meta data)에서 특성(Features)을 추출하여 콘텐츠의 특성 프로파일(Item profile)을 생성하고, 2) 이용자가 선호했던 콘텐츠들에서 자주 나타나는 특성들을 추출하여 이용자의 선호도 프로파일 (user profile) 을 만든 뒤, 3) 새로 유입되는 콘텐츠들 중 생성된 프로파일과 유사한 콘텐츠들만을 추출하여 이용자에게 제공
  - 음악 사이트인 판도라(Pandora)의 경우, 신곡이 출시되면 음악을 분석하여 장르, 비트, 음색 등 약 400여 항목의 특성을 추출한다. 그리고 사용자로부터는 'like'를 받은 음악의 특색을 바탕으로 해당 사용자의 프로파일을 준비한다. 이러한 음악의 특성과 사용자 프로파일을 비교함으로써 사용자가 선호할 만한 음악을 제공하게 된다.
- 콘텐츠의 내용을 분석해야 하므로 아이템(콘텐츠) 분석 알고리즘이 핵심
  - 군집분석(Clustering analysis),
  - 인공신경망(Artificial neural network),
  - tf-idf(term frequency inverse document frequency) 등의 기술이 사용
- 콘텐츠 기반 필터링은 내용 자체를 분석하므로 협업 필터링에서 발생하는 콜드 스타트 문제를 자연스럽게 해결. 하지만 다양한 형식의 항목을 추천하기 어려운 단점
- 콘텐츠 기반 필터링은 새로운 콘텐츠를 추천할 때 특히 유용
  - 새로운 콘텐츠들에 대한 다른 이용자들의 행동 데이터 유무에 상관없이 과거에 이용자가 선호했던 특성들을 많이 가지고 있으면 이용자 대상의 추천이 가능

Content Based (CB)	Collaborative Filtering (CF)
<p>Content-based recommender system relies on keywords that describe the items and a user profile to indicate the type of item the user likes. In other words, these algorithms try to recommend items that are similar to those that a user liked in the past (or is examining in the present). In particular, various candidate items are compared with items previously rated by the user and the best-matching items are recommended.</p>	<p>Collaborative filtering methods are based on collecting and analyzing a large amount of information on users' behaviors, activities or preferences and predicting what users will like based on their similarity to other users.</p>
Advantages	
<ul style="list-style-type: none"> <li>Easy implementation: only needs user profile and item features for recommendation.</li> <li>No cold start: New items can be recommended without substantial feedbacks from users.</li> </ul>	<ul style="list-style-type: none"> <li>Pure CF methods utilize only ratings and do not require any additional information about users or items (machine-recognizable features).</li> <li>CF systems can produce personalized recommendations because they consider and recommend based on other people experience.</li> <li>CF can suggest serendipitous items by observing similar-minded people's behavior.</li> </ul>
Disadvantages	
<ul style="list-style-type: none"> <li><b>Serendipity:</b> No inherent method for finding something unexpected.</li> <li>Not suitable if items does not contain enough information to discriminate or items cannot be encoded in meaningful features.</li> <li><b>New user:</b> When there's not enough information to build solid profile for a user.</li> </ul>	<ul style="list-style-type: none"> <li><b>Sparse data:</b> CF systems cannot produce recommendations if there are no ratings available. They demonstrate poor accuracy when there is little data about users' ratings.</li> <li><b>Cold start:</b> New items can be recommend without substantial feedbacks from users</li> <li>CF systems are not content aware meaning that information about the items is not considered while producing recommendations.</li> <li>CF lacks heterophilous diffusion, where individuals seek recommendations from more advanced peers unlike them.</li> </ul>



## ■ 콜드 스타트(Cold Start) 문제

- '새로 시작할 때 곤란함'을 의미. 협업 필터링 외에 위키 같은 협업 시스템에서 초기 정보 부족의 문제점을 일컫기 위해 사용
- 이용자의 행동 데이터가 충분히 모이지 않아 추천이 어려운 상황을 지칭
  - '아이템 콜드 스타트' (Item cold-start)
    - 새로운 콘텐츠가 제공됐을 때 충분한 수의 이용자가 새로운 콘텐츠를 소비하기 전에는 해당 콘텐츠가 추천되지 않음. 새 콘텐츠가 추천 시스템에 신규 인입됐을 때 발생
  - '이용자 콜드 스타트' (User cold-start)
    - 새로운 이용자의 행동 패턴을 분석 할 수만큼 충분한 양의 행동 데이터가 모이기 전에 해당 이용자에게 추천을 제공할 수 없음
  - '시스템 콜드 스타트'(System cold-start)
    - 추천 서비스 자체가 출시된 지 오래되지 않아 이용자 행동 데이터가 전반적으로 부족해서 추천 품질이 저하

## ■ 콜드 스타트 문제 해결

### ○ 아이템 콜드 스타트

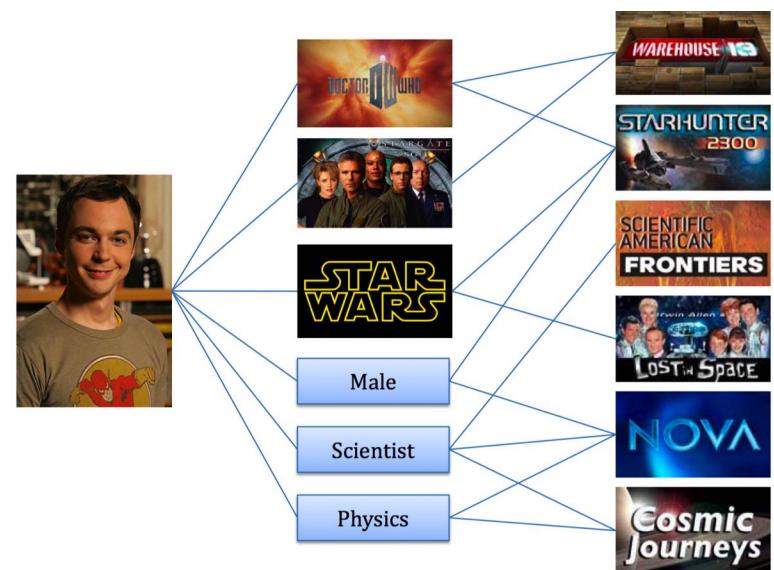
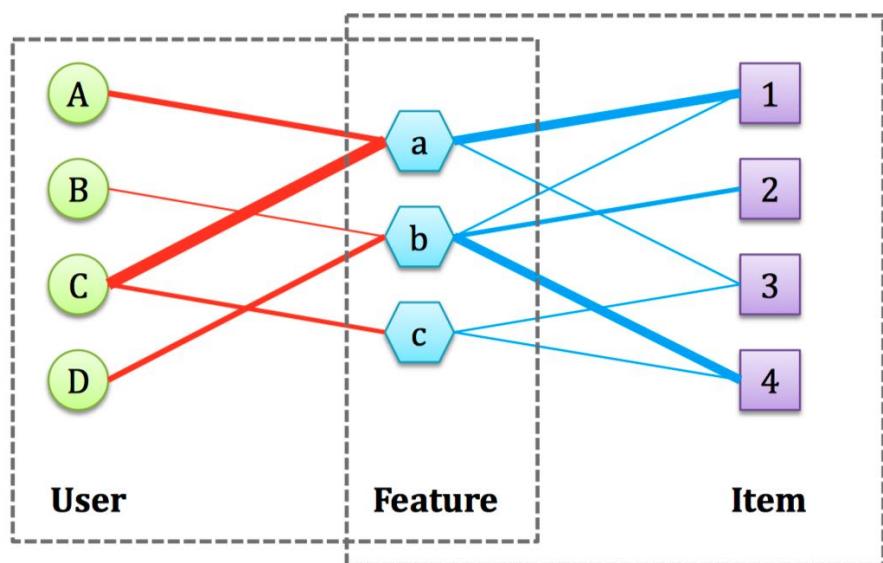
- Using content information
- Do not recommend

### ○ 이용자 콜드 스타트

- Non-personalization recommendation
  - Most popular items
  - Highly Rated items
- Using user register profile (Age, Gender...)

### ○ 시스템 콜드 스타트

- 이용자-콘텐츠 행렬(User-item Matrix)의 차원을 줄이는 방법
  - Latent Semantic Indexing(LSI)
  - Principal Component Analysis(PCA)
- Feature-based recommendation framework
  - Advantage:
    - » Architecture
    - » Heterogeneous data
    - » Reasonable Explanation
  - Disadvantage:
    - » Do not support user-based methods



## ■ How to get user interest quickly

- When new user comes, his feedback on what items can help us better understand his interest?
- Not very popular
- Can represent a group of items
- Users who like this item have different item

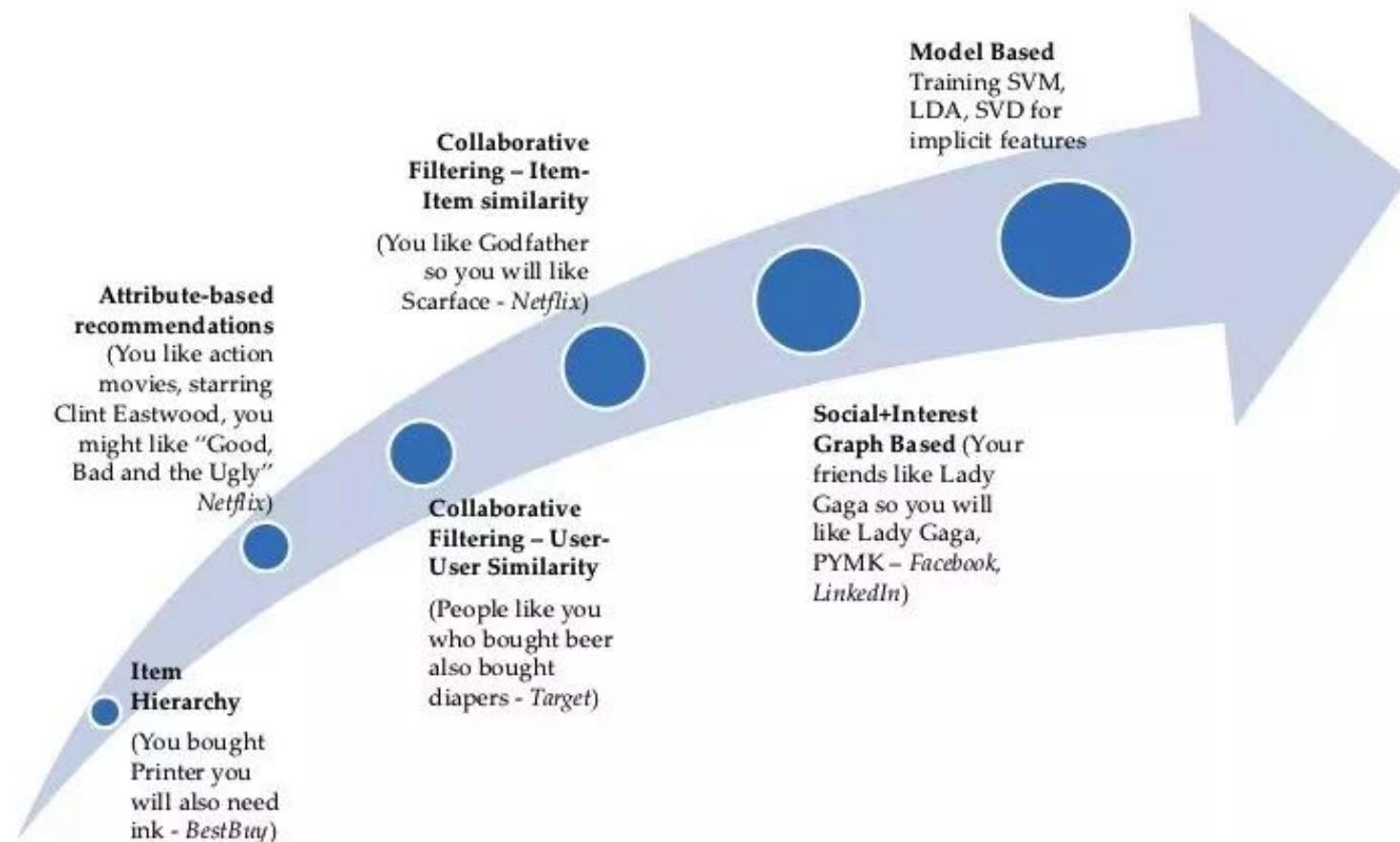
## ■ 모델 기반 협력 필터링(Model-based Collaborative Filtering algorithm)

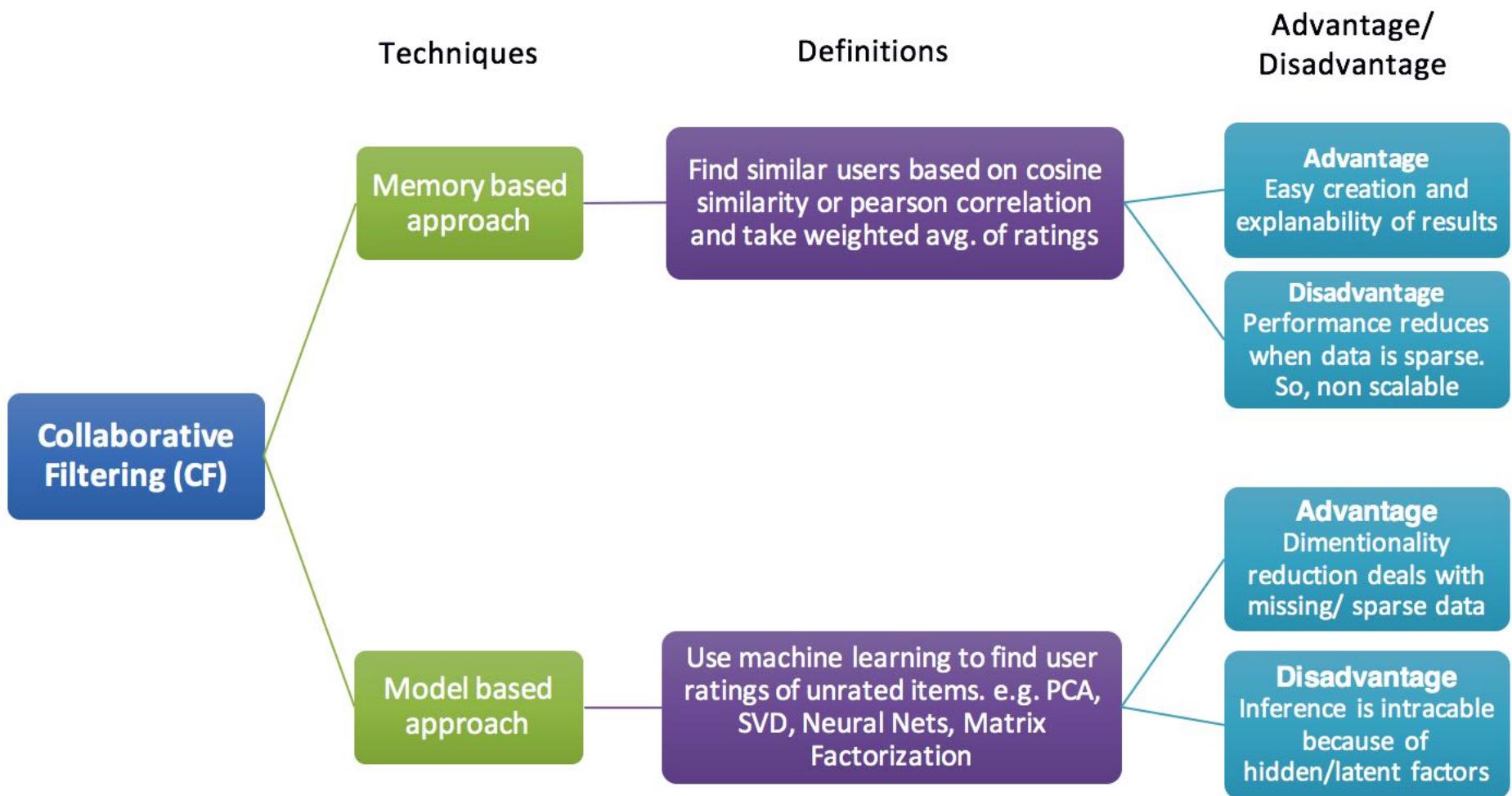
- 모델 기반 협력필터링은 기존 항목간 유사성을. 단순하게 비교하는. 것에서. 벗어나 자료안에 내재한. 패턴을 이용하는 기법
- 연관되는. 자료의. 크기를 동적으로. 변화시키는 방법
  - 예를들어. 영화를 추천하는 경우, '해리 포터' 시리즈 2편을 추천하기 위해서는 '해리 포터' 시리즈 1편, 단 한 편을 좋아했는가가 다른 무엇보다 중요한 요소. 하지만 <주토피아>를 추천하기 위해서는 많은 수의 유사한 영화를 고려해야 함
- 잠재(latent) 모델에 기반을 둔 방법
  - 잠재 모델이란 사용자가 특정 항목을 선호 하는 이유를 알고리즘적으로 알아내는 기법
  - 예를 들어 어느 사용자가 <태양의 후예>라는 드라마를 좋아하는 경우, 이 정보를 단순하게 그대로 사용하는 것이 아니라, 주위의 정보를 이용해 선호 이유를 유추하는 것. 그 사용자는 <태양의 후예>를 주연배우 때문에 좋아할 수도 있고, 드라마 OST가 좋아서 선호할 수도 있으며, 액션. 멜로 장르를 선호해서 선택할 수도 있음. 많은 양의 정보를 분석함으로써 이러한 이유를 알아내고, 이를 추천에 이용
  - 모델 기반 협력 필터링은 이러한 세부적 정보를 유추함으로써 높은 정확도로 항목을 추천 할 수 있다. 추천의 이유를 직관적으로 사용자에게 전달함으로써 추천의 신뢰성도 높일 수 있어, 현재 활발히 연구되고 있다. 하지만 이러한 모델을 만들어내는 데는 매우 많은 계산이 필요하고, 이에 따라 즉각적인 추천이 어려울 수 있다. 모델 기반 협력 필터링은 자료에 내재되어 있는 패턴을 알아내는 것이 핵심적인 기술이며, LDA(Latent Dirichlet Allocation), 베이지안 네트워크 (Bayesian Network) 등의 알고리즘이 사용된다.
  - 이와는 다른 방식으로, 최근 주목받는 딥러닝(Deep Learning) 기술에 기반을 둔 새로운 알고리즘이 여러 분야에서 놀라운 진전을 보이고 있다. 음악 서비스인 스포티파이(Spotify)가 협력 필터링에 딥러닝 기술을 적용한다고 알려져 있으며, 구글은 추천을 위한 텍스트를 자동으로 생성 하기 위해 딥러닝 기술을 사용하고 있다. 최근에 사람을 놀라게 한 알파고의 경우에서 보듯, 딥러닝에 기반을 둔 알고리즘이 개발될 것으로 예상된다.

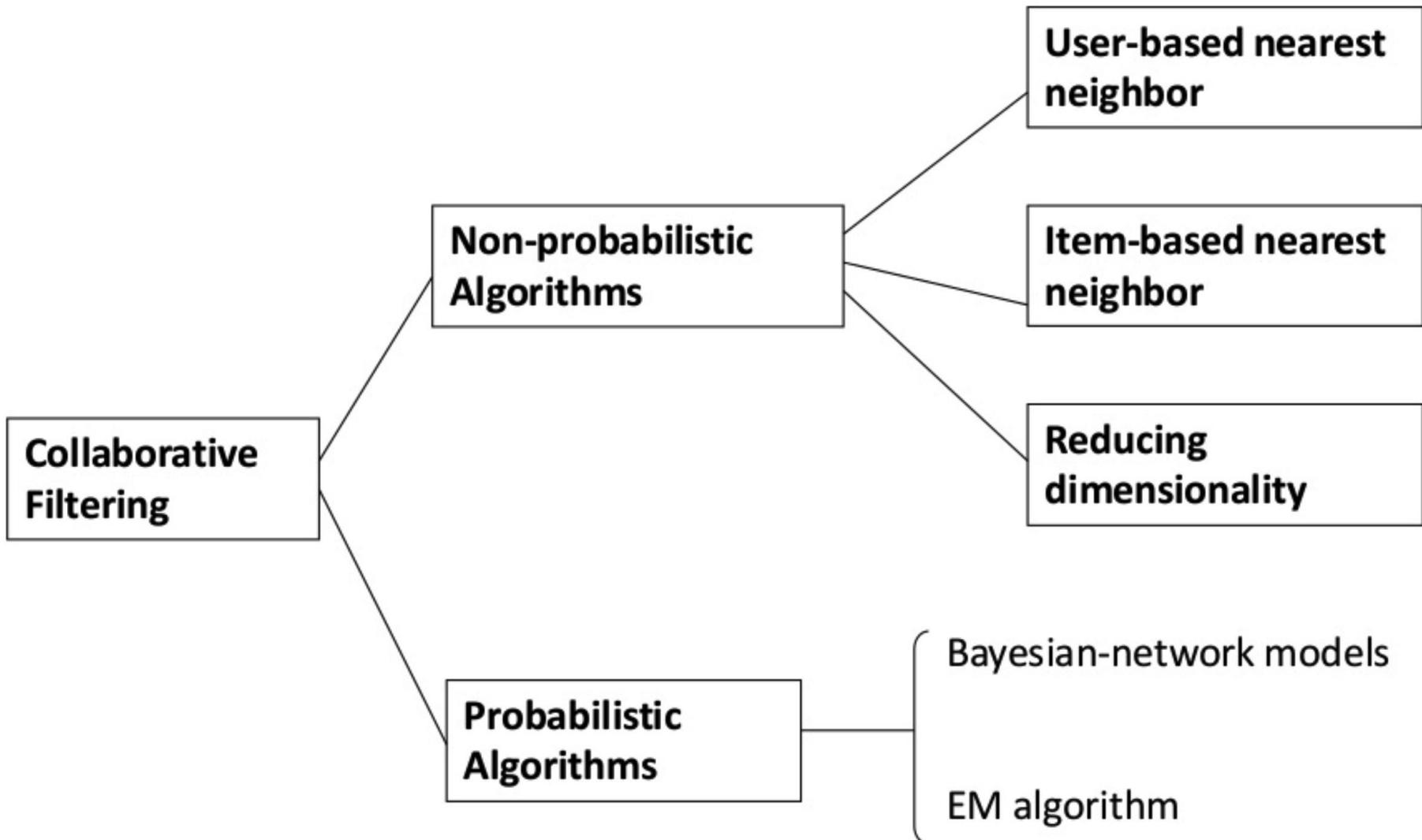
## ■ Hybrid

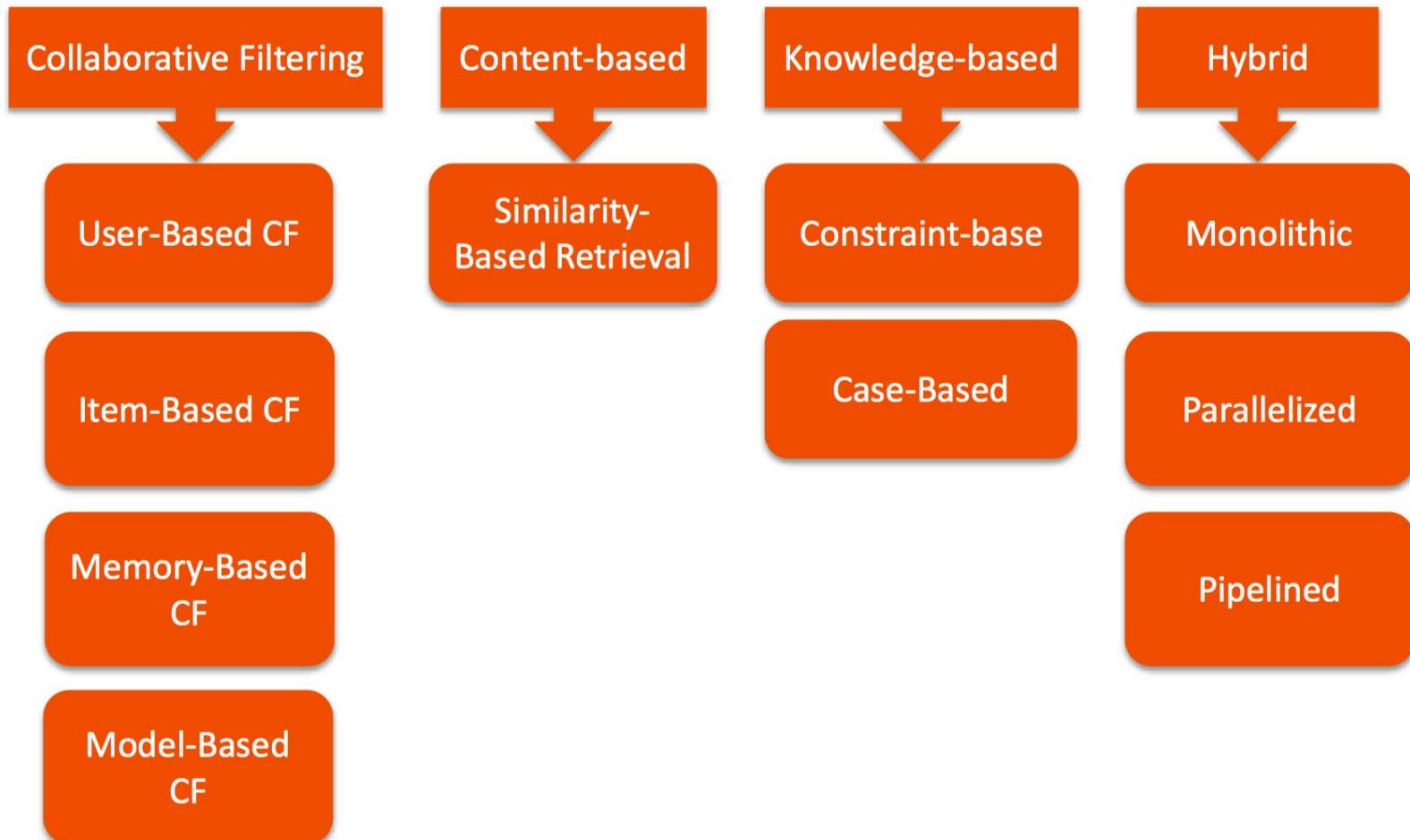
- 콘텐츠 기반 필터링과 협업 필터링을 결합한 다양한 혼합(Hybrid) 알고리듬들이 고안
  - (Balabanović & Shoham, 1997; Basilico & Hofmann, 2004; Basu, Hirsh, & Cohen, 1998; Good et al., 1999; Melville, Mooney, & Nagarajan, 2002; Park et al., 2006; Popescul et al., 2001; Schein, Popescul, Ungar, & Pennock, 2002).
  - 이용자의 선호도에 따라 내용 기반 또는 협업 필터링의 가중치를 주는 방법과 데이터가 없는 초기 콜드 스타트 상황에서 내용기반의 추천에 가중치를 두고, 추후 데이터가 많아질 경우 자동적으로 협업 필터링 기반의 추천에 더 많은 가중치를 두는 방법 등이 제안
- Hybrid 기법도 행동 데이터와 이용자 정보가 전혀 없는 새로운 이용자에게 추천을 제공하는데 어려움 (Park et al., 2006).
  - 새로운 이용자를 위한 일반적인 추천 방법 중 하나는 해당 시점에 가장 인기 있는 콘텐츠를 추천.
    - 예를 들어, 야후에서는 칼먼 필터 (Kalman Filter)를 이용하여 실시간으로 뉴스의 인기도를 측정 (Agarwal et al., 2009). 이후 야후는 이용자의 선호도와 멀티암드밴 딧(Multi-Armed Bandit, MAB)을 결합한 컨텍스츄얼 벤딧 (Contextual MAB) 알고리듬을 고안(Li et al., 2010).

# Recommender Approaches









## ■ Memory-Based Collaborative Filtering

- Item-Item Collaborative Filtering: "Users who liked this item also liked ..."
- User-Item Collaborative Filtering: "Users who are similar to you also liked ..."
- **non parametric ML approaches like KNN (clustering)**
- The key difference of memory-based approach from the model-based techniques (*hang on, will be discussed in next paragraph*) is that we are not learning any parameter using gradient descent (or any other optimization algorithm). The closest user or items are calculated only by using **Cosine similarity or Pearson correlation coefficients**, which are only based on arithmetic operations.

# USER & ITEM

	TV	Camera	iPod	iPhone	iPad	Macbook	Headphone
1							
2							
3							
4							
5							
6							

# ORDER DATA

	TV	Camera	iPod	iPhone	iPad	Macbook	Headphone
1							
2							
3							
4							
5							
6							

## ORDER DATA (cont.)

	 TV	 Camera	 iPod	 iPhone	 iPad	 Macbook	 Headphone
1	1	1		1		1	
2	1		1		1		1
3	1			1			1
4		1		1		1	
5		1	1		1		
6	1	1			1		

## ORDER DATA (cont.)

	 TV	 Camera	 iPod	 iPhone	 iPad	 Macbook	 Headphone
1	1	1	0	1	0	1	0
2	1	0	1	0	1	0	1
3	1	0	0	1	0	0	1
4	0	1	0	1	0	1	0
5	0	1	1	0	1	0	0
6	1	1	0	0	1	0	0

# VECTOR & DIMENSION

	TV	Camera	iPad	iPhone	iPad	Macbook	Headphone
1	1	1	0	1	0	1	0
2	1	0	1	0	1	0	1
3	1	0	0	1	0	0	1
4	0	1	0	1	0	1	0
5	0	1	1	0	1	0	0
6	1	1	0	0	1	0	0

# VECTOR & DIMENSION

Vector	TV	Camera	Pet	iPhone	iPad	Macbook	Headphone
	1	1	0	1	0	1	0
	1	0	1	0	1	0	1
	1	0	0	1	0	0	1
	0	1	0	1	0	1	0
	0	1	1	0	1	0	0
	1	1	0	0	1	0	0

# VECTORS

	 TV	 Camera	 iPod	 iPhone	 iPad	 Macbook	 Headphone
1	1	1	0	1	0	1	0
2	1	0	1	0	1	0	1
3	1	0	0	1	0	0	1
4	0	1	0	1	0	1	0
5	0	1	1	0	1	0	0
6	1	1	0	0	1	0	0

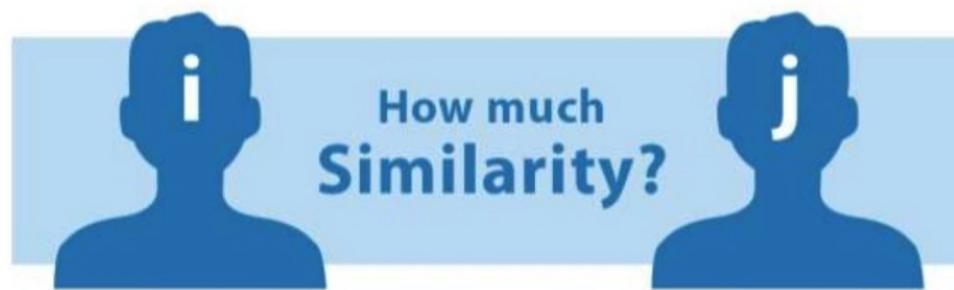
# VECTORS

---

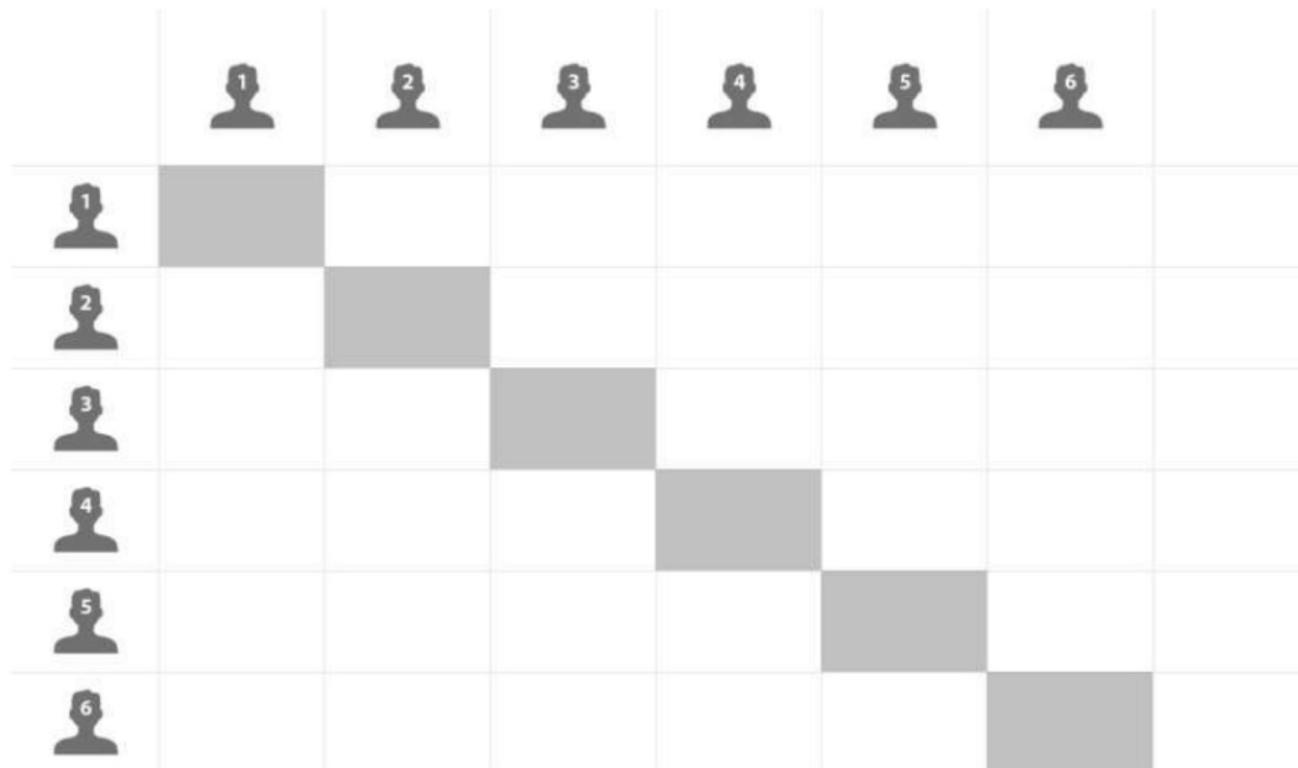
$$\begin{array}{l} \text{1} = \left( \begin{array}{cccccccc} 1 & , & 1 & , & 0 & , & 1 & , & 0 & , & 1 & , & 0 \end{array} \right) \\ \text{2} = \left( \begin{array}{cccccccc} 1 & , & 0 & , & 1 & , & 0 & , & 1 & , & 0 & , & 1 \end{array} \right) \\ \text{3} = \left( \begin{array}{cccccccc} 1 & , & 0 & , & 0 & , & 1 & , & 0 & , & 0 & , & 1 \end{array} \right) \\ \text{4} = \left( \begin{array}{cccccccc} 0 & , & 1 & , & 0 & , & 1 & , & 0 & , & 1 & , & 0 \end{array} \right) \\ \text{5} = \left( \begin{array}{cccccccc} 0 & , & 1 & , & 1 & , & 0 & , & 1 & , & 0 & , & 0 \end{array} \right) \\ \text{6} = \left( \begin{array}{cccccccc} 1 & , & 1 & , & 0 & , & 0 & , & 1 & , & 0 & , & 0 \end{array} \right) \end{array}$$

# SIMILARITY CALCULATION

---



# USER SIMILARITY MATRIX



# SIMILARITY CALCULATION

---

## Cosine similarity

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

## SIMILARITY CALCULATION

1	1	1	0	1	0	1	0
2	1	0	1	0	1	0	1

$$\text{Sim}(u1, u2) = \frac{1 \times 1 + 1 \times 0 + 0 \times 1 + 1 \times 0 + 0 \times 1 + 1 \times 0 + 0 \times 1}{\sqrt{1^2 + 1^2 + 0^2 + 1^2 + 0^2 + 1^2 + 0^2} \times \sqrt{1^2 + 0^2 + 1^2 + 0^2 + 1^2 + 0^2 + 1^2}} \\ = 0.25$$

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

# OTHER SIMILARITY MEASURES

---

## Euclidean Distance

$$D_E(\vec{t}_a, \vec{t}_b) = \left( \sum_{t=1}^m |w_{t,a} - w_{t,b}|^2 \right)^{1/2}$$

## Pearson Correlation Coefficient

$$D_E(\vec{t}_a, \vec{t}_b) = \left( \sum_{t=1}^m |w_{t,a} - w_{t,b}|^2 \right)^{1/2}$$

## Jaccard Coefficient

$$SIM_J(\vec{t}_a, \vec{t}_b) = \frac{\vec{t}_a \cdot \vec{t}_b}{|\vec{t}_a|^2 + |\vec{t}_b|^2 - \vec{t}_a \cdot \vec{t}_b}.$$

AND MORE...

## SIMILARITY CALCULATION EXAMPLE

	1	2	3	4	5	6
1		0.25	0.577	0.866	0.289	0.577
2	0.25		0.577	0	0.577	0.577
3	0.577	0.577		0.333	0	0.333
4	0.866	0	0.333		0.333	0.333
5	0.289	0.577	0	0.333		0.667
6	0.577	0.577	0.333	0.333	0.667	

## K-NEAREST-NEIGHBOR

	1	2	3	4	5	6
1	0.25	0.577	0.866	0.289	0.577	
2	0.25		0.577	0	0.577	0.577
3	0.577	0.577		0.333	0	0.333
4	0.866	0	0.333		0.333	0.333
5	0.289	0.577	0	0.333		0.667
6	0.577	0.577	0.333	0.333	0.667	

# K-NEAREST-NEIGHBOR



0.866



0.577



0.577



0.289



0.25

**K = 4**

~ Only get 4 most similar users (nearest neighbor)

# NEIGHBORS' ORDER

**K = 4**

~ Only get 4 most similar users (nearest neighbor)



0.866



0.577



0.577



0.289



0.25



# REMOVE BOUGHT ITEMS



0.866



0.577



0.577



0.289

**K = 4**

~ Only get 4 most similar users (nearest neighbor)



# CALCULATING FINAL SCORE

K = 4

~ Only get 4 most similar users (nearest neighbor)



Smartphone = 0.866

Headphones = 0.577

Speaker = 0.289



0.577

0.577

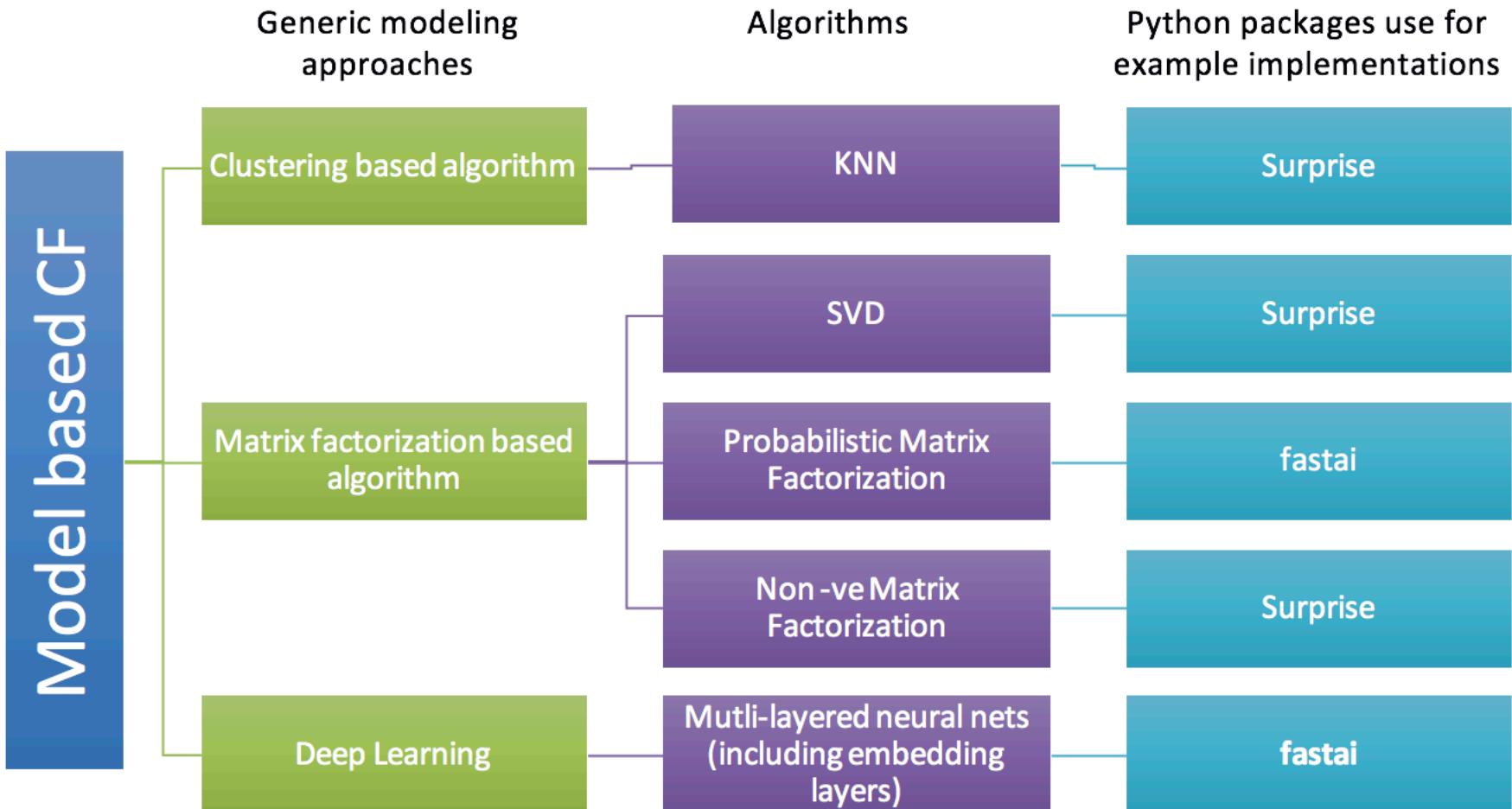
0.289

0.289

- User-based vs. Item-based

	User-based	Item-based
Scalability	Bad when user size is large	Bad when item size is large
Explanation	Bad	Good
Novelty	Bad	Good
Coverage	Bad	Good
Cold start	Bad for new users	Bad for new items
Performance	Need to get many users history	Only need to get current user's history

Items are simpler, users have multiple tastes



# OBJECT

---



A music track

# OBJECT INFORMATION

---



A music track

- Singer
- Composer
- Bitrate
- Length
- Instrument
- Genre
- Language
- Year
- Chord
- Subject
- ...

# FEATURE SET

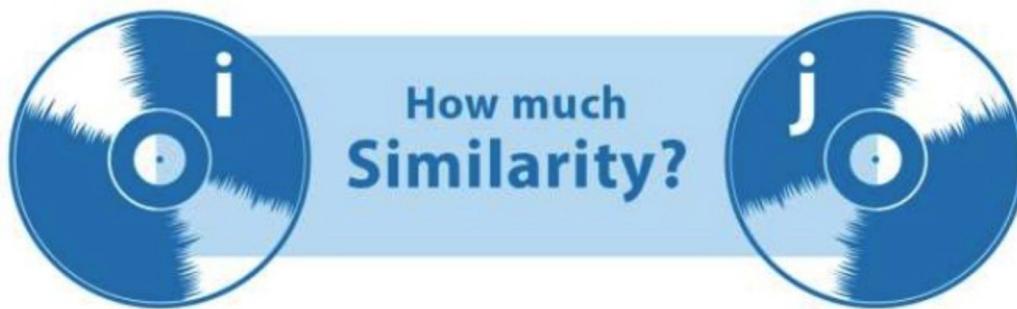


A music track

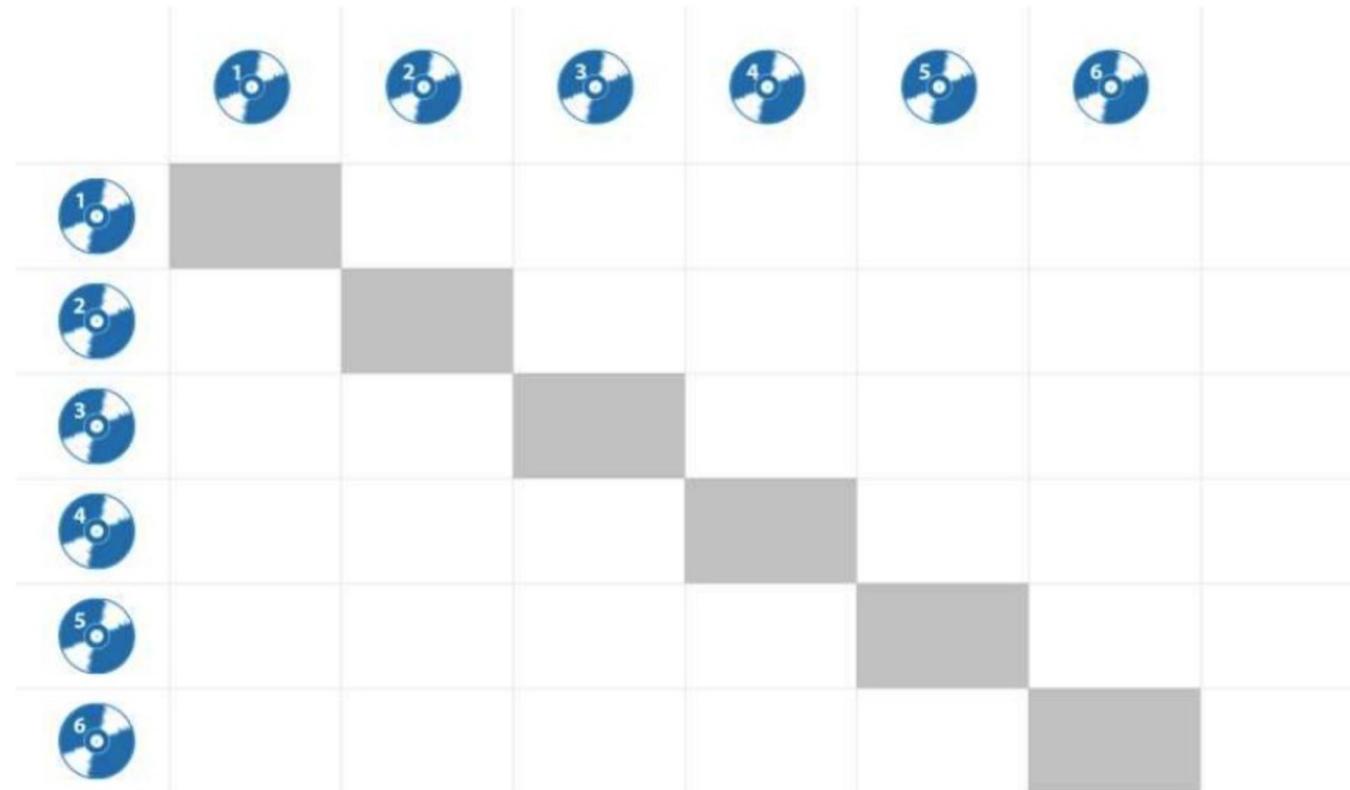
- Singer
- Composer
- Bitrate
- Length
- Instrument
- Genre
- Language
- Year
- Chord
- Subject
- ...



**FEATURES**



# SIMILARITY MATRIX



# SIMILARITY MEASURE



# SIMILARITY MEASURE

---

$$\begin{aligned} S(O_i, O_j) = & \omega_1 f(A_{1i}, A_{1j}) + \omega_2 f(A_{2i}, A_{2j}) \\ & + \cdots + \omega_n f(A_{ni}, A_{nj}) \end{aligned}$$

# SIMILARITY MATRIX

	1	2	3	4	5	6
1	0.25	0.577	0.866	0.289	0.577	
2	0.25	0.577	0	0.577	0.577	
3	0.577	0.577	0.333	0	0.333	
4	0.866	0	0.333	0.333	0.333	0.333
5	0.289	0.577	0	0.333	0.667	
6	0.577	0.577	0.333	0.333	0.667	

# SIMILARITY SORTING

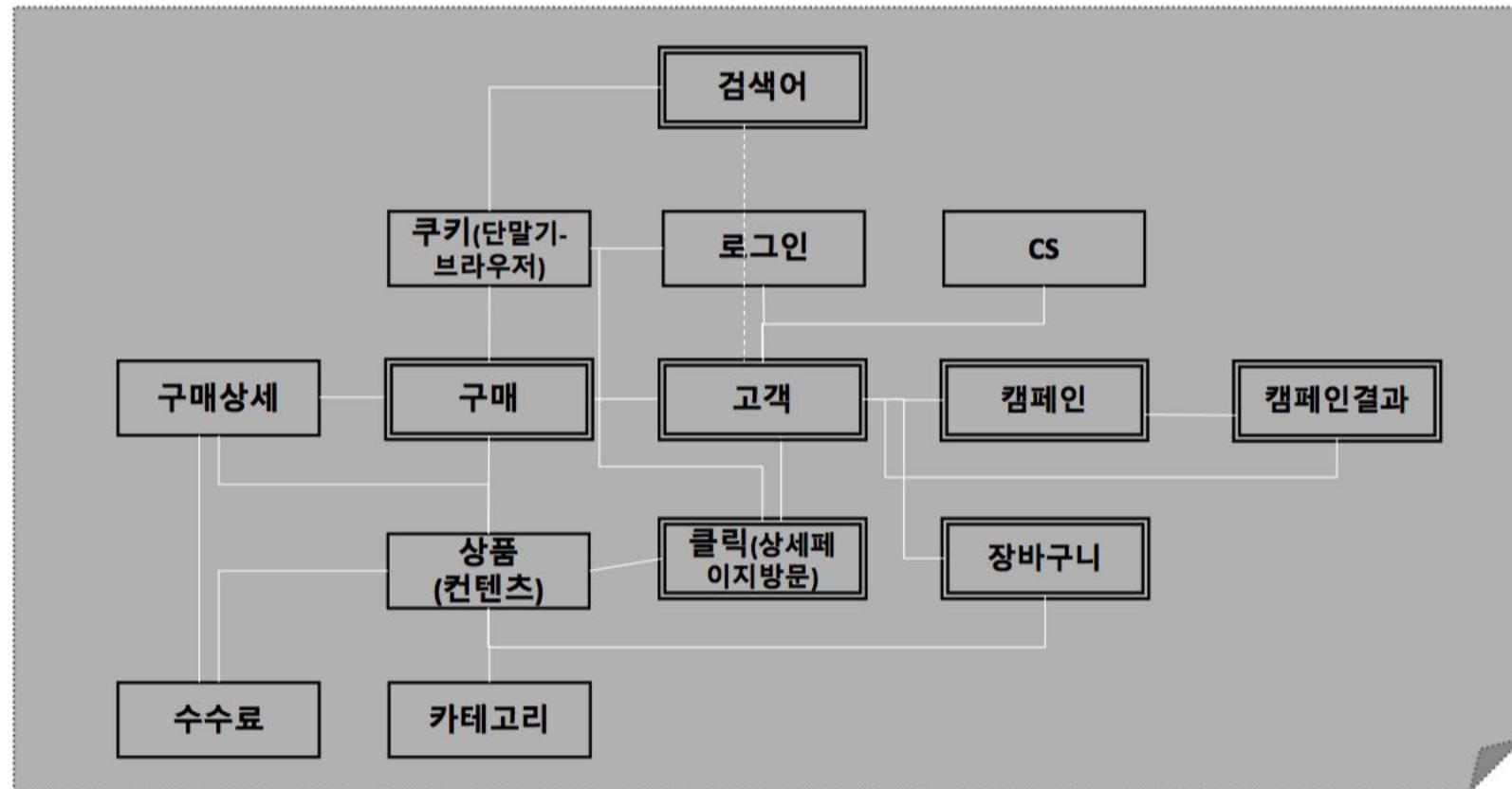
	1	2	3	4	5	6
1	0.25	<b>0.577</b>	<b>0.866</b>	<b>0.289</b>	<b>0.577</b>	
2	0.25	0.577	0	0.577	0.577	
3	0.577	0.577	0.333	0	0	0.333
4	0.866	0	0.333	0	0.333	0.333
5	0.289	0.577	0	0.333	0	0.667
6	0.577	0.577	0.333	0.333	0.667	

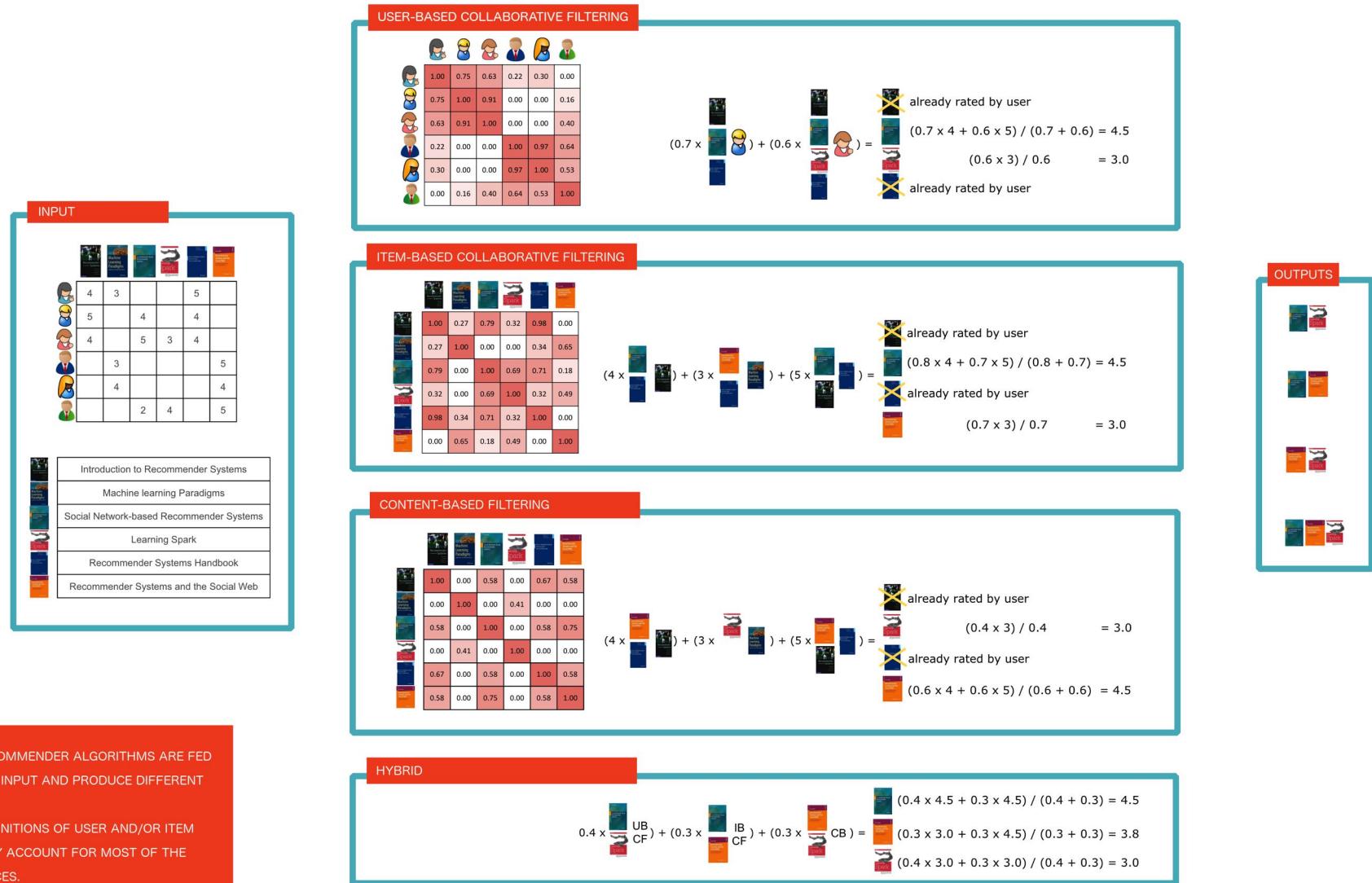
# K-NEAREST NEIGHBOR (knn)



**K = 4**

~ Only get 4 most similar products (nearest neighbor)





FOUR RECOMMENDER ALGORITHMS ARE FED THE SAME INPUT AND PRODUCE DIFFERENT OUTPUTS.  
 THEIR DEFINITIONS OF USER AND/OR ITEM SIMILARITY ACCOUNT FOR MOST OF THE DIFFERENCES.

## CBF의 Meta 정보 예시

- 웹 페이지
  - words, hyperlinks, images, tags, comments, titles, URL, topic
- 음악
  - genre, rhythm, melody, harmony, lyrics, meta data, artists, bands, press releases, expert reviews, loudness, energy, time, spectrum, duration, frequency, pitch, key, mode, mood, style, tempo
- 사용자
  - age, sex, job, location, time, income, education, language, family status, hobbies, general interests, Web usage, computer usage, fan club membership, opinion, comments, tags, mobile usage
- 상황
  - time, location, mobility, activity, socializing, emotion

## ■ 추천 시스템에서 로그가 필요한 이유

- 고객의 방문 성향 분석

고객이 얼마의 주기로 방문을 하는지, 얼마의 시간동안 쇼핑을 하는지, 주로 접속하는 시간이 언제인지, 주로 접속하는 요일이 언제인지 등

- 고객의 관심 상품 분석

고객이 접속해서 어떤 상품을 조회하는지 확인. 고객이 상품의 특정 카테고리를 클릭하거나, 특정 상품의 상세 페이지를 클릭한다면, 해당 상품에 관심이 있다고 가정

- 접속 세션별 관심 상품 분석

동일한 고객이라도 세션별로는 다른 목적을 가지고 쇼핑몰에 접속했다고 볼 수 있음. 고객이 어제 접속 시에는 의류와 관련된 쇼핑을 주로 했지만, 오늘 접속시에는 모자를 주로 봤다면, 같은 고객이지만, 두 개의 세션은 다른 목적을 가지고 있다고 볼 수 있음. 이 경우 한 명의 고객이지만 두 명의 고객으로 분류하여 데이터를 학습한다면, 향후 다른 고객이 모자에 관심 있는 고객이 접속 시 모자에 관심있던 세션에 대한 학습 데이터 기반으로 추천 하는 것이 구매 확률을 높일 수 있음

- A/B 테스트 결과 검증

추천 시스템을 개발할 때는 한 개의 모델을 만드는 것이 아님. 다양한 추천 모델을 만들어 검증을 하고, 해당 도메인의 전문가에게 추천 결과에 대한 피드백을 받고, 문제가 없다면 그 다음에 고객 대상으로 A/B 테스트. 이 때 A 모델과 B 모델을 고객에게 노출 시 어떤 추천 모델을 더 선호하는지 확인하기 위해 서도 웹 로그가 필요

- 추천 모델 적용 실적 분석

최종적으로 선택된 모델을 적용 후 추천 결과에 대한 실적을 평가. 이 때는 노출 대비 얼마나 클릭했는지, 화면의 어느 영역에 추천 상품을 노출시 효과가 있는지 분석하는데 활용.

Behavior	User	Size
Page view	All user	Very Large
Watch video	All user	Large
Favorite	Register user	Middle
Vote	Register user	Middle
Add to playlist	Register user	Small
Facebook like	Register user	Small
Share	Register user	Small
Review	Register user	Small

## ■ Data Structure

- User ID
- Item ID
- Behavior Type
- Behavior Content
- Context
  - Timestamp
  - Location

## ■ Which data is most important

- Main behavior in the website
- All user can have such behavior
- Cost
- Reflect user interests on items

상품 상세 페이지 조회 시 대체재로 다른 제품 추천 (상품 상세 페이지를 보는 시점에는 해당 상품에 대해서 구매 의사가 확실하지 않기 때문에 대체재를 보여줄 경우에 관심도가 낮을 수 있음)

Samsung Galaxy S8 Unlocked 64GB - US Version (Midnight Black) - US Warranty  
Available from these sellers.

See all buying options

Compare to similar items

Samsung Galaxy S8 Unlocked by Samsung Samsung U.S. Limited Warranty*	Samsung Galaxy S8 64GB Unlocked Phone - International Version (Arctic Silver)	Sony Xperia XA1 Ultra 6" Factory Unlocked Phone - 32GB - White (U.S. Warranty)	Essential Phone 128 GB Unlocked with Full Display, Dual Camera - Pure White
This item Samsung Galaxy S8 Unlocked 64GB - US Version (Midnight Black) - US Warranty			
Add to Cart	Add to Cart	Add to Cart	Add to Cart

- ◆ 대체재 : 서로 대신 쓸 수 있는 관계에 있는 두 가지의 재화  
(쌀과 밀가루, 만년필과 연필, 버터와 마가린)
- ◆ 보완재 : 서로 보완 관계에 있는 재화

Shopping Cart

	Price	Quantity
Apple MacBook Air 13.3-Inch Laptop (Intel Core i5 1.6GHz, 128GB Flash, 8GB RAM, OS X El Capitan) by Apple	\$884.00	1
Only 16 left in stock - order soon. Shipped from: DigitalandMore Gift options not available. Learn more		
<a href="#">Delete</a>   <a href="#">Save for later</a>		

Subtotal (1 item): \$884.00

The price and availability of items at Amazon.com are subject to change. The Cart is a temporary place to store a list of your items and reflects each item's most recent price. [Learn more](#)

Do you have a gift card or promotional code? We'll ask you to enter your claim code when it's time to pay.

Sign in to turn on 1-Click ordering.

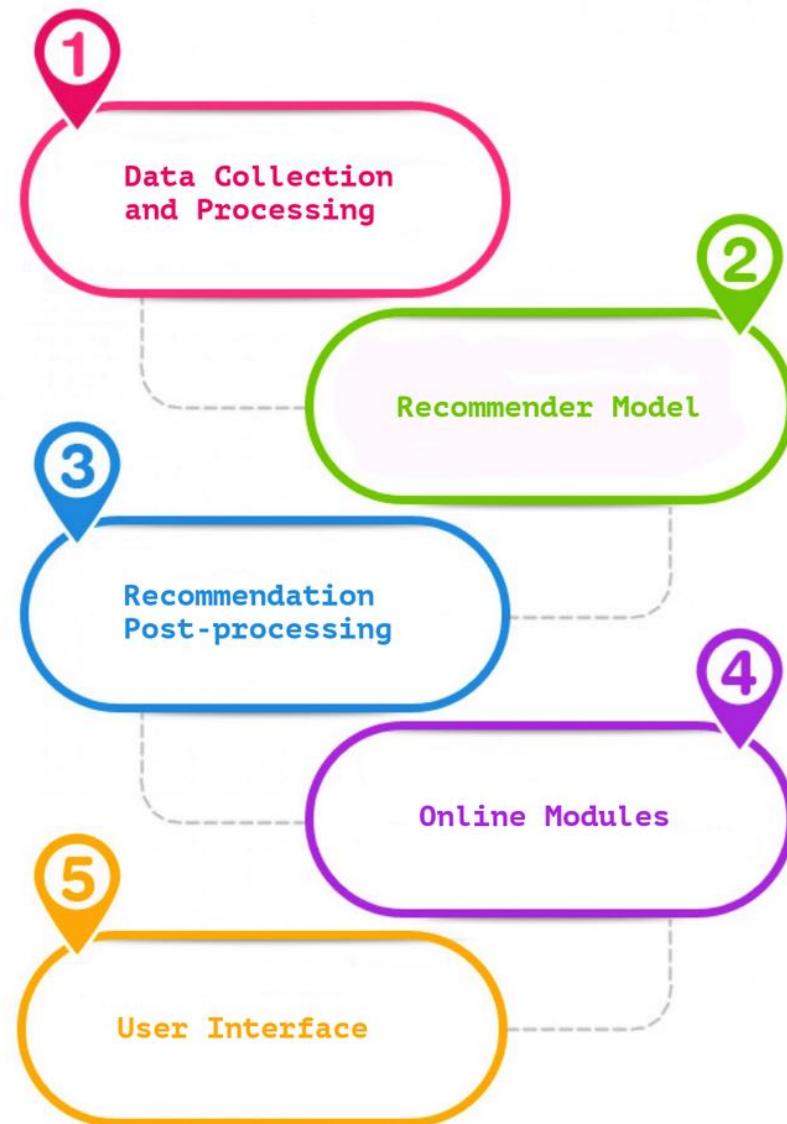
Sponsored Products related to items in your cart

KayondHerringbone... ★★★★★ 9 \$13.99
AmazonBasics... ★★★★★ 9,927 \$11.50

[See all buying options](#)

[See all buying options](#)

맥북을 카트에 담을 시 보완재로 노트북 가방과 파우치가 추천



## ■ Important points before building your own recommendation system:

- If you have a large database and you make recommendations from it online, the best way would be to divide this problem into 2 subproblems: 1) choosing top-N candidates and 2) ranking them.
- How do you measure the quality of your model? Along with the standard quality metrics, there are some metrics specially for recommendation problems: [Recall@k and Precision@k](#), Average Recall@k, and Average Precision@k. Also look at [the great description of metrics for recommendation systems](#).
- If you are solving recommendation problems with classification algorithms, you should think about generating negative samples. If a user bought a recommended item, you should not add it as a positive sample, and others as negative samples.
- Think about the online-score and offline-score of your algorithm quality. A training model only on historical data can lead to primitive recommendations because the algorithm won't know about new trends and preferences.

■ <http://files.grouplens.org/papers/ml-100k.zip>