

Nearest Neighbor Collaborative Filtering

쓸 수 있는 데이터

- User info
- Item info
- User – Item interaction
 - 별점, 좋아요 ...etc

사용할 데이터

- User – Item interaction
 - 별점, 좋아요 ...etc

목차

- User-User collaborative filtering
- Item-Item collaborative filtering
- Advanced topics

User-User collaborative filtering

An algorithmic framework for performing collaborative filtering

JL Herlocker, [JA Konstan](#), A Borchers, J Riedl - ACM SIGIR Forum, 2017 - dl.acm.org

8(10) → ③ → ◆ → + → † → ③ ↑ * → ⊕ → ↘ → * → > → > → < → ★ → ◆ → * → | → ★ → ③ → → → ◆ ↑ ○ → ▢ → → → ③ ↑ ○
→ ③ → → → ▢ → ○ → → → ↘ → ! 10 ▢ → → → | → ◆ → → → ⊕ → ◆ ↑ ↘ → ↘ → ◆ → > → → → ★ → ○ → → → ◆ → ◆ → * 10 ▢ →
> → ★ → X → ③ ↑ ↘ → * → * → ▢ → > → ! 10 ▢ ↑ √ → → → ★ → √ → ③ ↑ * → * 10 * → ↘ → → → ▢ → / → → → * → * → *
→ ③ > + → ★ → ③ → → → * → ⊕ → † → ◆ ↑ ○ → ③ → → → ◆ → * → * → + → ★ → ▢ → → → ③ ↑ ↘ → * ...
10 * → → → ↑ ○ → → → 10 * → * → ◆ ↑ ○ → ③ → → → * → ▢ → ★ → > → → → x ↘ ↑ ↘ → * → ③ ↑ ↘ → ↘ → ◆ → > → > → ○ → ▢ → ▢
↔ † → ③ → → → ◆ → ▢ → * → ▢ → * → 8 → → → ! → ③ ↑ ○ → ③ ↑ ↘ → ⊕ → ③ → → → ▢ → > → → → ③ ↑ ↘ → > → ↘ → / ...
☆ 3366회 인용 관련 학술자료 전체 6개의 버전

Herlocker, J. L., Konstan, J. A., Borchers, A., & Riedl, J. (2017, August). An algorithmic framework for performing collaborative filtering. In *ACM SIGIR Forum* (Vol. 51, No. 2, pp. 227-234). ACM.

Non-personalized collaborative filtering

- Average rating given to item i

$$S(u, i) = \frac{\sum_{v \in U} r_{vi}}{|U|}$$

Annotations for the equation above:

- $S(u, i)$: pred. score (indicated by a bracket pointing to the entire left side)
- u : user (indicated by an arrow pointing to u)
- i : item (indicated by an arrow pointing to i)
- $\sum_{v \in U}$: all users (indicated by an arrow pointing to the summation symbol and the set U)
- r_{vi} : rating (indicated by an arrow pointing to r_{vi})
- $|U|$: # users (indicated by an arrow pointing to $|U|$)

Non-personalized collaborative filtering

- Rating scale에는 개인차가 있다
 - 갑은 4점을 가장 많이 주고 5점 만점도 자주 준다
 - 을은 2점을 가장 많이 주고 5점 만점은 거의 주지 않는다
 - 정규화:

$$S(u, i) = \bar{r}_u + \frac{\sum_{v \in U} (r_{vi} - \bar{r}_v)}{|U|}$$

일반적인 정규화가 아닌 이유

- 유저의 표준편차를 구하기가 어렵다
사람들의 평점 데이터는 매우 self-selected

measures level of confidence in parenting among university graduates. Those who are proud of their parenting are more likely to want to talk about it, and therefore more likely to fill in the survey. There will be less representation from the graduates who are not confident in their parenting ability or who are ashamed of their track record.

- 경험적으로,
평균 0, 표준편차 1로 맞추는 것 보다
단순히 평균값을 빼는 것이 더 효과적이다

Non-personalized collaborative filtering

$$S(u,i) = \frac{\sum_{v \in U} r_{vi} \cdot w_{uv}}{\sum_{v \in U} w_{uv}}$$

← similarity

Non-personalized collaborative filtering

$$S(u,i) = \overline{r_u} + \frac{\sum_{v \in U} (r_{vi} - \overline{r_v}) * w_{uv}}{\sum_{v \in U} w_{uv}}$$

How to compute 가중치 w

- Compute 'similarity' between users U and V

How to compute 가중치 w

- Pearson Correlation
 - Degree to which things vary together as a relation to how much they vary individually

$$\text{Pearson Correl } w_{uv} = \frac{\sum_{i \in \mathcal{I}} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sigma_u \sigma_v}$$

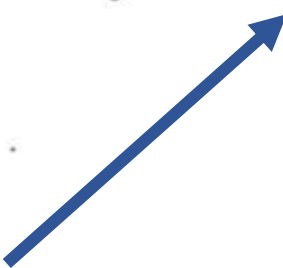
Pearson Correlation의 문제점

- Users U and V 사이에 겹치는 아이템 수가 적다면?
 - 둘 모두 평가한 아이템이 2개라면 둘 간의 선형관계를 추정할 수 있겠지만 이것이 얼마나 정확할까?
- 5점 척도 평점이 아닌 추천/비추천 데이터라면?
 - 0, 1, Null 이라면?

가중치 w

- Pearson correlation이 아닌 다른 값들로 대체할 수도 있다.

"neighborhood"

$$S_{(u,i)} = \bar{r}_u + \frac{\sum_{v \in U} (r_{vi} - \bar{r}_v) * w_{uv}}{\sum_{v \in U} w_{uv}}$$


- 모든 유저를 반영할 필요는 없다.
 - 유저 자신 제외
 - Limit total user size – 50~200명만 반영하고 싶을 때
 - Similarity가 안 좋을 수 있음
 - 최저 similarity 설정 – minimize noise
 - Neighbors 수가 적을 수 있음

정리

- Given a set of items I , and a set of users U , and a sparse matrix of ratings R ,
We compute the prediction $s(u,i)$ as follows:
 - For all users $v \neq u$, compute w_{uv}
 - similarity metric (e.g., Pearson correlation)
 - Select a neighborhood of users $V \subset U$ with highest w_{uv}
 - may limit neighborhood to top-k neighbors
 - may limit neighborhood to $\text{sim} > \text{sim_threshold}$
 - may use sim or $|\text{sim}|$ (risks of negative correlations)
 - may limit neighborhood to people who rated i (if single-use)
 - Compute prediction:

$$s(u, i) = \bar{r}_u + \frac{\sum_{v \in V} (r_{vi} - \bar{r}_v) * w_{uv}}{\sum_{v \in V} w_{uv}}$$

Implementation Issues

- Given $m = |U|$ users and $n = |I|$ items:
 - Computation can be a Bottleneck
 - Correlation between two users is $O(n)$
 - All correlations for a user is $O(mn)$
 - All pairwise correlations is $O(m^2n)$
 - Recommendations at least $O(mn)$
 - Lots of ways to make more practical
 - More persistent neighborhoods ($m \rightarrow k$)
 - Cached or incremental correlations

Core assumption

- Base assumption #1:
 - 개개인의 취향은 독립적이고 stable하거나
 - 서로 sync를 맞춰서 변화한다
- Base assumption #2:
 - Our system is scoped within a domain of agreement
 - People who agree on movies tend to agree on movies
 - But people who agree on postings about humor may not agree on recipes
 - 따라서 각 도메인에 따라 다른 relationship을 만들어야 한다

정리

- Find people most like you
- Scale their thoughts adequately in relation to how much they are like you (similarity)
- => prediction of what you may like

Customization and design
decisions

조절할 수 있는 것들

- Selecting neighborhoods
- Scoring items from neighborhoods
- Normalizing data
- Computing similarities
 - Algorithms
 - Tweaks

Selecting neighborhoods

- 모든 neighbors
 - 데이터나 유저 수가 적을 때
- Similarity/distance 역치
- 랜덤 neighbors
 - 유저 수가 많을 때
 - 주로 다른 기법과 혼합해서 사용
- Top-N neighbors by similarity/distance
- 클러스터 안의 neighbors

How many neighbors?

- 이론상으로는, 다다익선
 - 좋은 similarity metric이 있다면
- 실제로는, 비슷하지 않은 neighbor들의 noise가 유용성을 저하한다
- 25~100을 가장 자주 사용
 - 30~50 often good for movies

Scoring from neighborhoods

- 평균
- Weighted 평균
 - Common, simple, works well
- 다중선행회귀

Normalizing data

- 유저들은 다르게 평점을 준다
 - 갑은 전반적으로 높게, 을은 전반적으로 낮게
 - 갑은 3~4점 위주로, 을은 5점 척도를 전부 사용하여
- 따라서 정규화 필요

Common normalizations

- Subtract user mean rating
- Z-score로 변환
- Subtract item or item-user mean
- 계산이 끝나면 정규화를 reverse 해야 한다.

Computing similarity

- Pearson correlation

$$sim(u, v) = \frac{\sum (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum (r_{ui} - \bar{r}_u)^2} \sqrt{\sum (r_{vi} - \bar{r}_v)^2}}$$

- Usually only over ratings in common
- User normalization not needed
- Spearman rank correlation is Pearson applied to *ranks*
- Hasn't been found to work as well

문제점: 데이터가 적을 때

- 유저 갑과 유저 을이 모두 평가한 상품이 단 한 개일 때
 - Pearson correlation = 1
 - 유저 갑과 을은 유사한가?

해결: Weighting similarity

$$sim(u, v) = \frac{\sum (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum (r_{ui} - \bar{r}_u)^2} \sqrt{\sum (r_{vi} - \bar{r}_v)^2}}$$

- 분자: 갑이 평가한 상품 n 을이 평가한 상품
- 분모: 갑이 평가한 상품 U 을이 평가한 상품
- 분모는 큰데 분자가 작으면 similarity가 낮아진다

Good baseline configuration

- Top N neighbors (~30개)
- Weighted averaging
- User-mean or z-score normalization
- Cosine similarity over normalized ratings

Influence limiting and attack resistance

What can go wrong in collaborative filtering

Fake ratings

리얼

REAL, 2016

관람객       4.42 기자·평론가      2.67

네티즌      4.24 내 평점      등록 >

개요 액션, 느와르 한국 137분 2017.06.28 개봉

감독 이사랑

출연 김수현(수트 장태영/뿔테 장태영), 이성민(최진기), 성동일(… [더보기](#) >

등급 [국내] 청소년 관람불가



다운로드

♡ 9,326



Fake ratings

★★★★★ 10 이 영화를 본 제가 별 10개를 받아야겠습니다

오레몬(audi****) | 2017.06.28 03:01 | 신고



공감

13338



바공감

503

★★★★★ 10 나만 당할수는 없다!!

김용훈(abak****) | 2017.06.28 00:35 | 신고



공감

10881



바공감

681

★★★★★ 7 저희 할머니가 이걸보고 류마티스 관절염을 극복하셨습니다. 감사합니다.

즉결심판(tige****) | 2017.06.28 00:31 | 신고



공감

8706



바공감

600

별점 그래프

평균 ★1.1
(1만명)



코멘트 2500+



이동진 평론가



★ 1.0

대체 어떻길래,라는 심정으로 보았다가...레
알.

👍 1571 💬 104

좋아요



영사남

★ 0.5

ㅅㅂ..내 2시간 18분
ㅅㅂ..내 4500원
괜히 궁금해서 건드렸다가 큰 낭패를 봤다
역시 사람의 말은 잘 들어야 돼 젠장

👍 608 💬 25

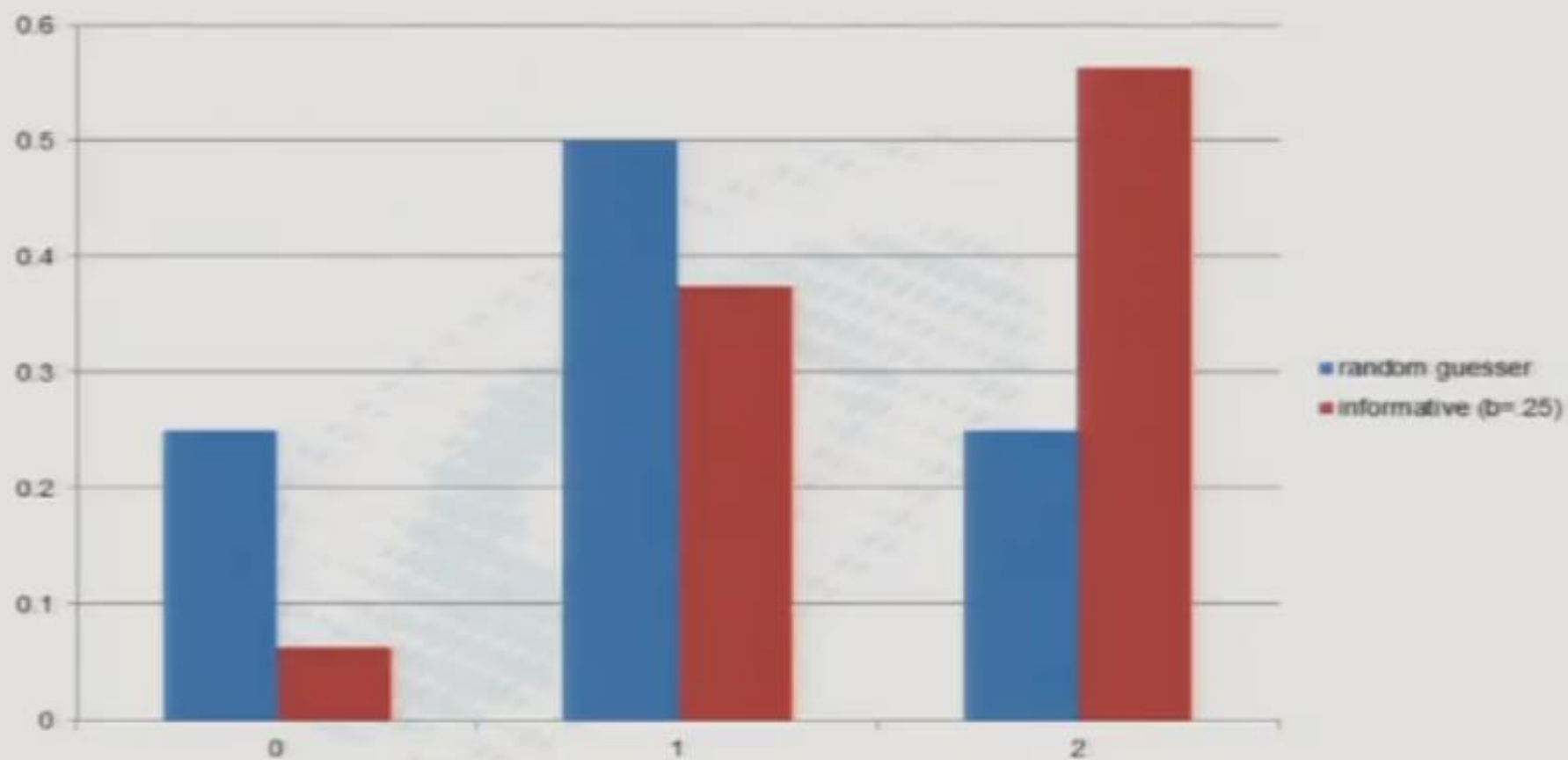
좋아요

Detecting fake accounts

- 짧은 시간에 너무 많은 rating을 생산한 계정
- 자신보다 몇 초 전에 생성된 계정과 rating 패턴이 지나치게 유사한 계정
- Fake account의 패턴과 특징을 찾아 잡아낸다

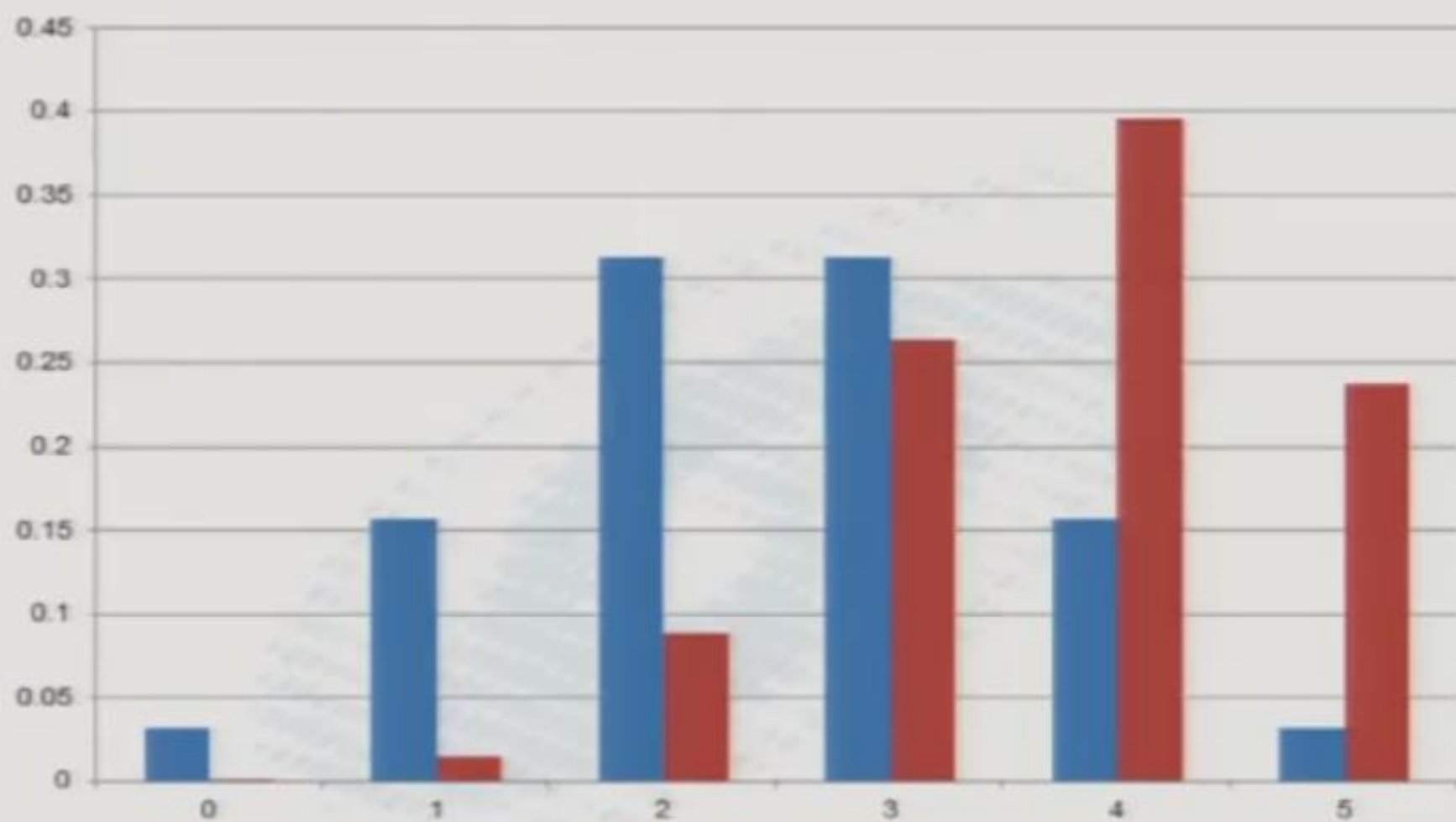
Separating informative and uninformative

With 2 Items of History



Distribution of number of ratings agreeing with target

With 5 Items of History



With 30 Items of History



Separating informative and uninformative

- 최소 n 개의 평점이 쌓이기 전까지는 반영하지 않는다
- n 개 기준을 보수적으로 잡을 수록 더 많은 informative rater들의 데이터를 버리게 될 수 있다
- 하지만 attacker라면?
 - 첫 30개는 informative한 척 하고 후에 fake rating을 생산하기 시작한다면?

Fundamental tradeoff

- Resistance to more attackers -> more loss of info from genuine raters
 - (n, c) -robust
 - preventing attackers from doing more than (c) bits of damage
 - $\Omega(n \log(n/c))$ information loss
- 해결방안
 - Limit n (number of attacker accounts)
 - Allow larger damage c and try to repair it
 - Keep attackers uninformed about other's ratings

Trust-based recommendation

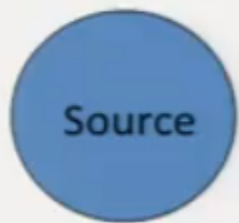
Epinions.com

Trust-based recommendation

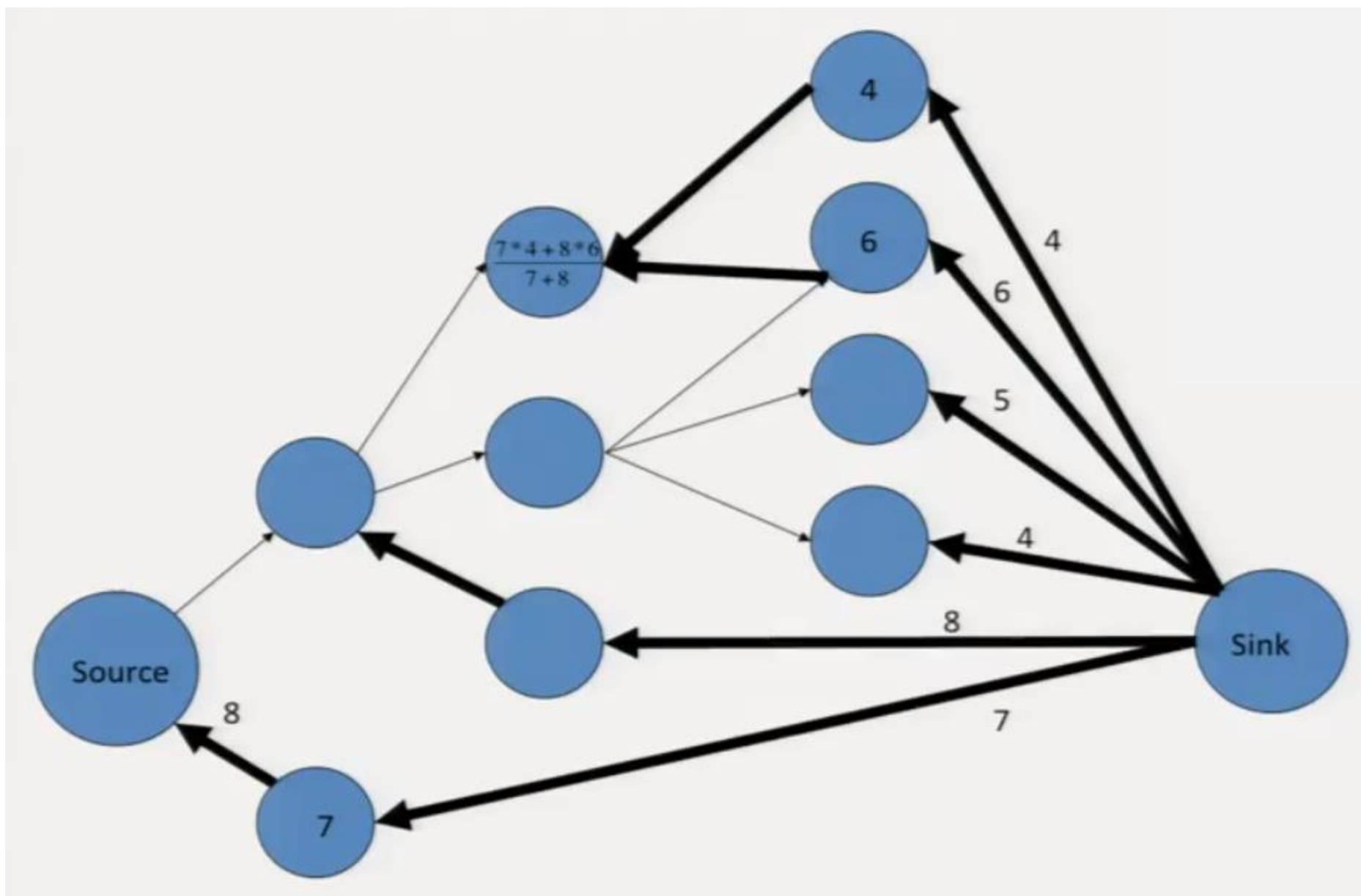
- User-user collaborative filtering:
Rating * Weight(similarity)
나와 비슷한 사람에게 가중치를 준다
- Trust-based recommendation:
Rating * Weight(trust rating - explicit/implicit)
내가 신뢰하는 사람에게 가중치를 준다

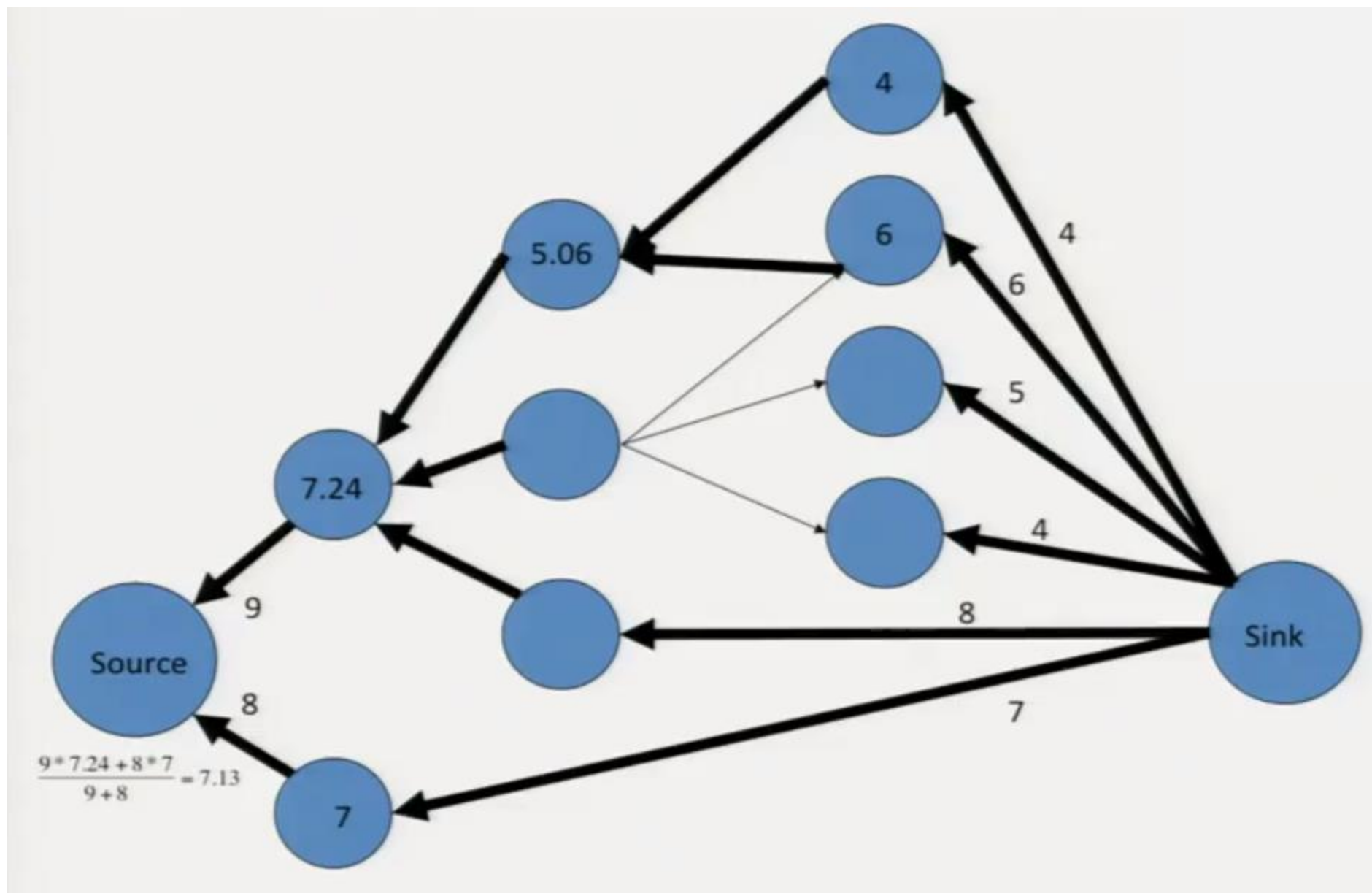
Computing trust

- Social network
- Trust
 - Reputation – global, general, 누구에게나 항상 같은 값
 - Personal trust – 개인마다 다른 값
 - Ex) 진보적인 사람에게 보수적인 사람의 정치적 견해는 신뢰도가 높지 않은 정보









Privacy issues

- Trust ratings has to remain private

장점

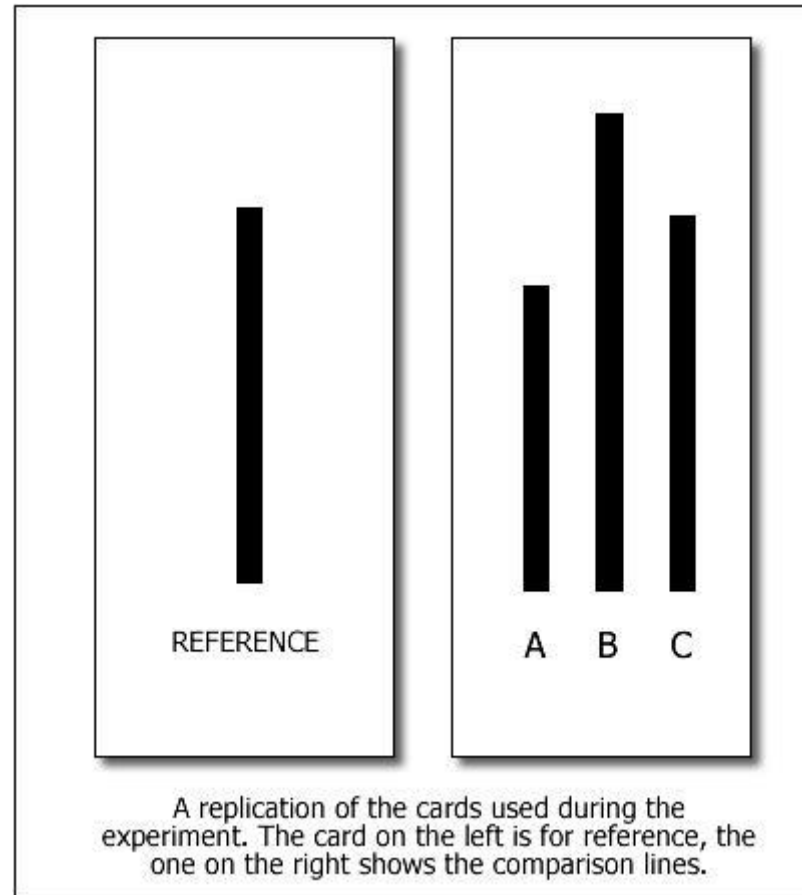
- 유저가 평균 유저와 매우 다를 때
 - "시계 태엽 오렌지" – 다수 유저에게 호평을 받았지만 나는 불호
 - Collaborative filtering
 - 높은 예상평점 (잘못된 추천)
 - Trust recommendation
 - similarity와 trust를 동시에 반영
 - 다른 영화들에 대한 평점은 비슷하지만 한 작품에 대해 심하게 평점이 엇갈릴 때, 서로의 신뢰도가 내려간다

단점

- Trust data를 얻는 것이 어렵다
 - Explicit하게 묻거나
 - Estimate
- 사람들은 trust 평가를 잘 못한다
 - 특정 domain에 한정하여 평가한다는 것을 어려워한다
 - 내 동생을 믿지만 개가 추천하는 영화는 내 취향이 아니야
 - 사기꾼 갑을 믿지 않지만 영화 취향은 참 잘 맞아
 - Low rating을 주는 것을 불편해한다
 - Trust와 agreement의 차이
 - 너와 나의 취향은 다르지만 너는 내 취향에 맞게 추천해 줄 것이라 믿는다 (X)
 - 너와 나의 취향이 비슷하므로 너의 평점을 신뢰할 수 있다 (O)

Impact of bad ratings

Interface influences ratings

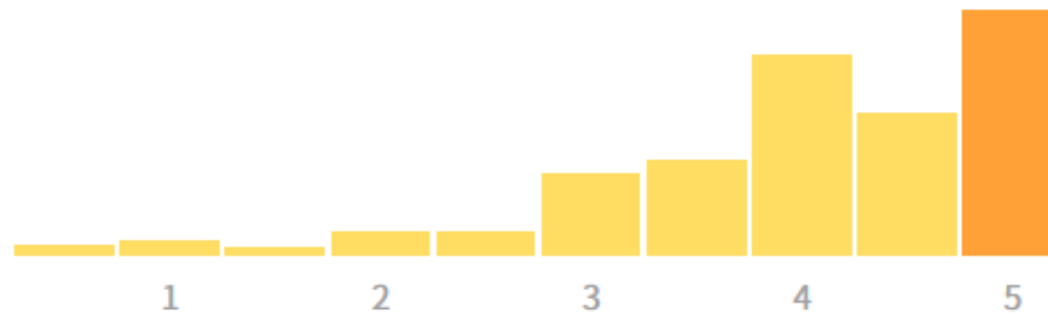


- Asch, S. E. (1956). Studies of independence and conformity: I. A minority of one against a unanimous majority. *Psychological monographs: General and applied*, 70(9), 1.

Interface influences ratings

별점 그래프

평균 ★4.0
(53만명)



Interface influences ratings






A little experiment

- 오래 전에 평가한 영화를 다시 평가
- 예상 평점 X
 - Same: 60%, lower: 20%, higher: 20%
- 예상 평점 = 기존 평점
 - Same: 70%, lower: 15%, higher: 15%
- 예상 평점 = 기존 평점 + 1
 - Same: 10%, lower: 60%, higher: 30%
- 예상 평점 = 기존 평점 - 1
 - Same: 30%, lower: 60%, higher: 10%
- 평점이 보여진 방향으로(예상 평점대로) 움직인다

Bad(fake/manipulated) ratings

- 사람들은 보여진 대로 움직이는 경향이 있기 때문에 Rating을 조작할 수 있다
 - 하지만 사람들은 알아챈다!

140자평 | 총 22,298건

  패널평만 보기 | 최신순 

★★★★★ 10 다른말 필요없고 지금 1점대 주는 사람들 아이디를 클릭해서 보길 바란다과연 알바인지 아닌지 특이한건 다른영화들에대해서 똑같은 평점대를 준다는 것이다 이게 알바일까 실질적인 영화관람인들인가 판단하길 바란다
lone**** | 2012.12.06 | [신고](#)

★★★★★ 1 폭동영화라서 재미가 없음
heas**** | 2012.12.06 | [신고](#)

★★★★★ 1 왜만들었냐
leec**** | 2012.12.06 | [신고](#)

★★★★★ 1 이거보면 시간이 아깝다
madu**** | 2012.12.06 | [신고](#)

초장에 형성된 분위기가 계속 간다

- 사람들은 보여지는 것 (원래 있던 것)에 영향을 받고 그 방향으로 움직이는 경향이 있다
- 원래 있던 것, 초기에 있는 콘텐츠가 향후 서비스 자체의 분위기와 방향성을 좌우한다