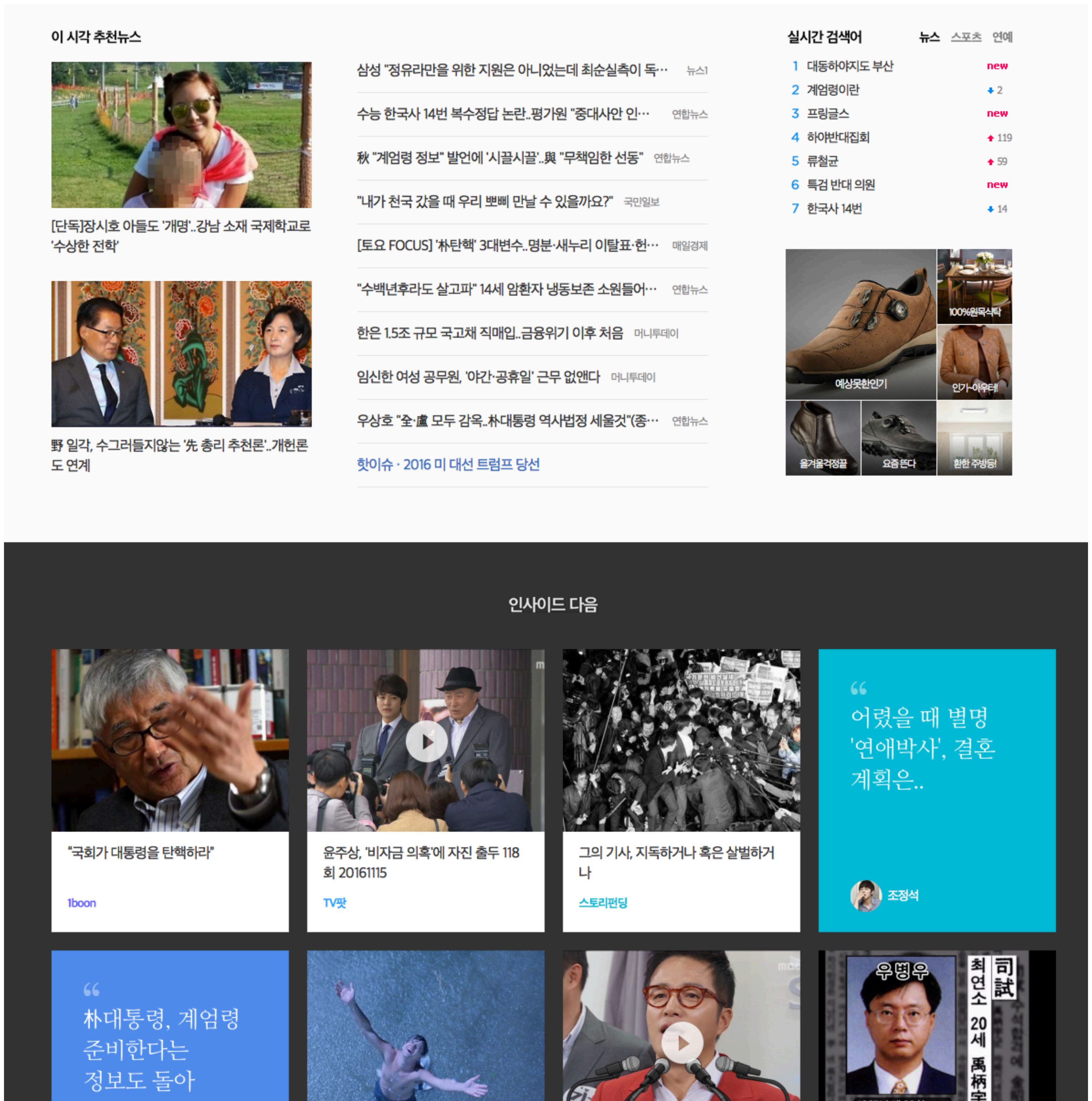
Recommender System in the Real World

Sanghyuk Chun
Researcher @CLAIR (Clova AI Research)

About Me...

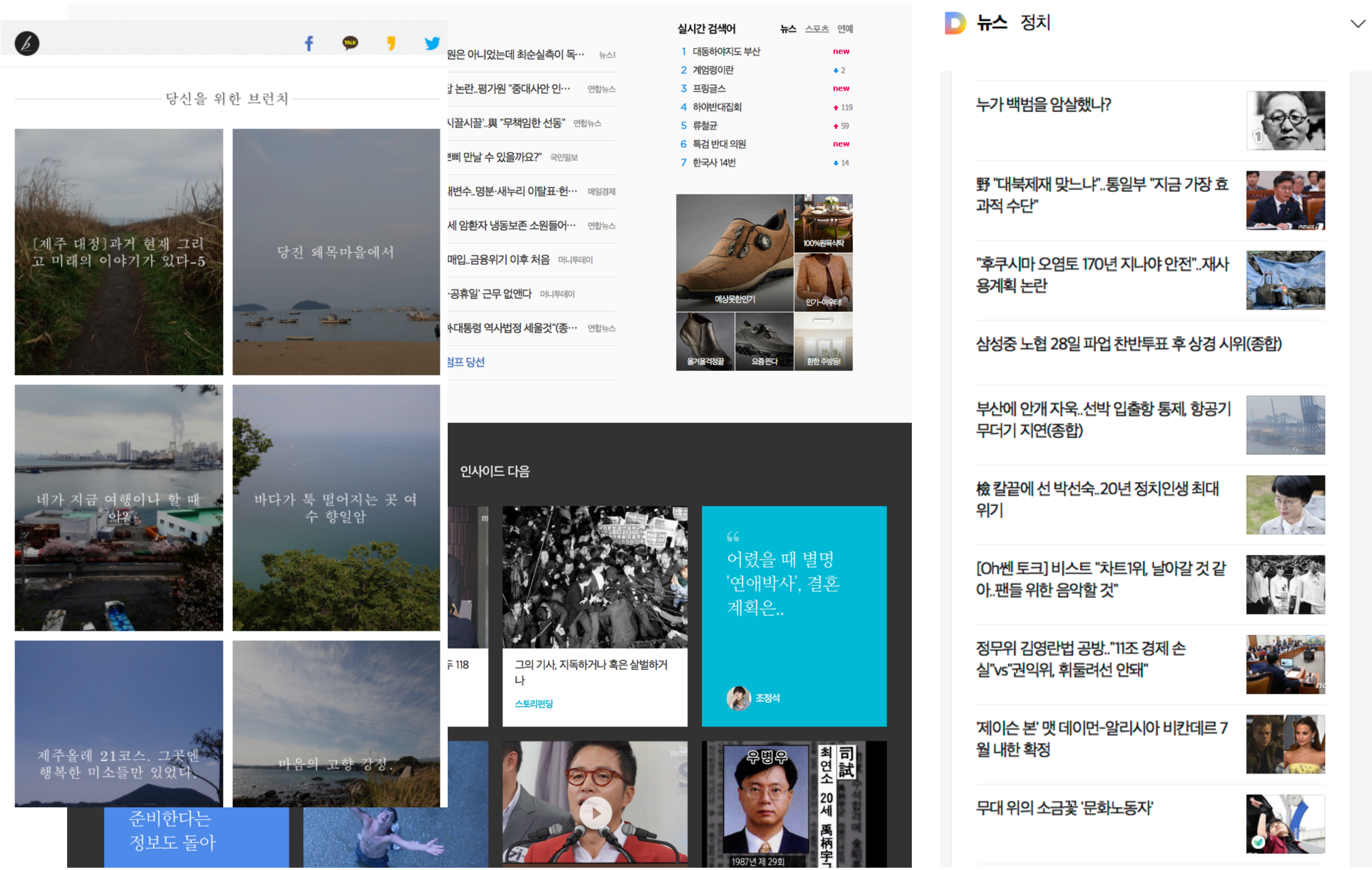
In Kakao...



이 시각 추천뉴스
[단독]장시호 아들도 '개명'..강남 소재 국제학교로 '수상한 전학'
野 일각, 수그러들지않는 '先 총리 추천론'..개헌론도 연계
삼성 "정유라만을 위한 지원은 아니었는데 최순실측이 독…
수능 한국사 14번 복수정답 논란..평가원 "중대사안 인…
秋 "계엄령 정보" 발언에 '시끌시끌'..與 "무책임한 선동"
"내가 천국 갔을 때 우리 뽀삐 만날 수 있을까요?"
[토요 FOCUS] '朴탄핵' 3대변수..명분·새누리 이탈표·헌…
"수백년후라도 살고파" 14세 암환자 냉동보존 소원들어…
한은 1.5조 규모 국고채 직매입..금융위기 이후 처음
임신한 여성 공무원, '야간·공휴일' 근무 없앤다
우상호 "全·盧 모두 감옥..朴대통령 역사법정 세울것"(종…
핫이슈 · 2016 미 대선 트럼프 당선
실시간 검색어
뉴스
스포츠
연예
1 대동하야지도 부산 new
2 계엄령이란 2
3 프링글스 new
4 하야반대집회 119
5 류철균 59
6 특검 반대 의원 new
7 한국사 14번 14
예상못한인기
100%원목식탁
인기-아우터
올겨울걱정끝
요즘뜬다
환한 주방등!
인사이드 다음
"국회가 대통령을 탄핵하라"
1boon
윤주상, '비자금 의혹'에 자진 출두 118회 20161115
TV팟
그의 기사, 지독하거나 혹은 살벌하거나
스토리펀딩
어렸을 때 별명 '연애박사', 결혼 계획은..
조정석
朴대통령, 계엄령 준비한다는 정보도 돌아
우병우
1987년 제 29회

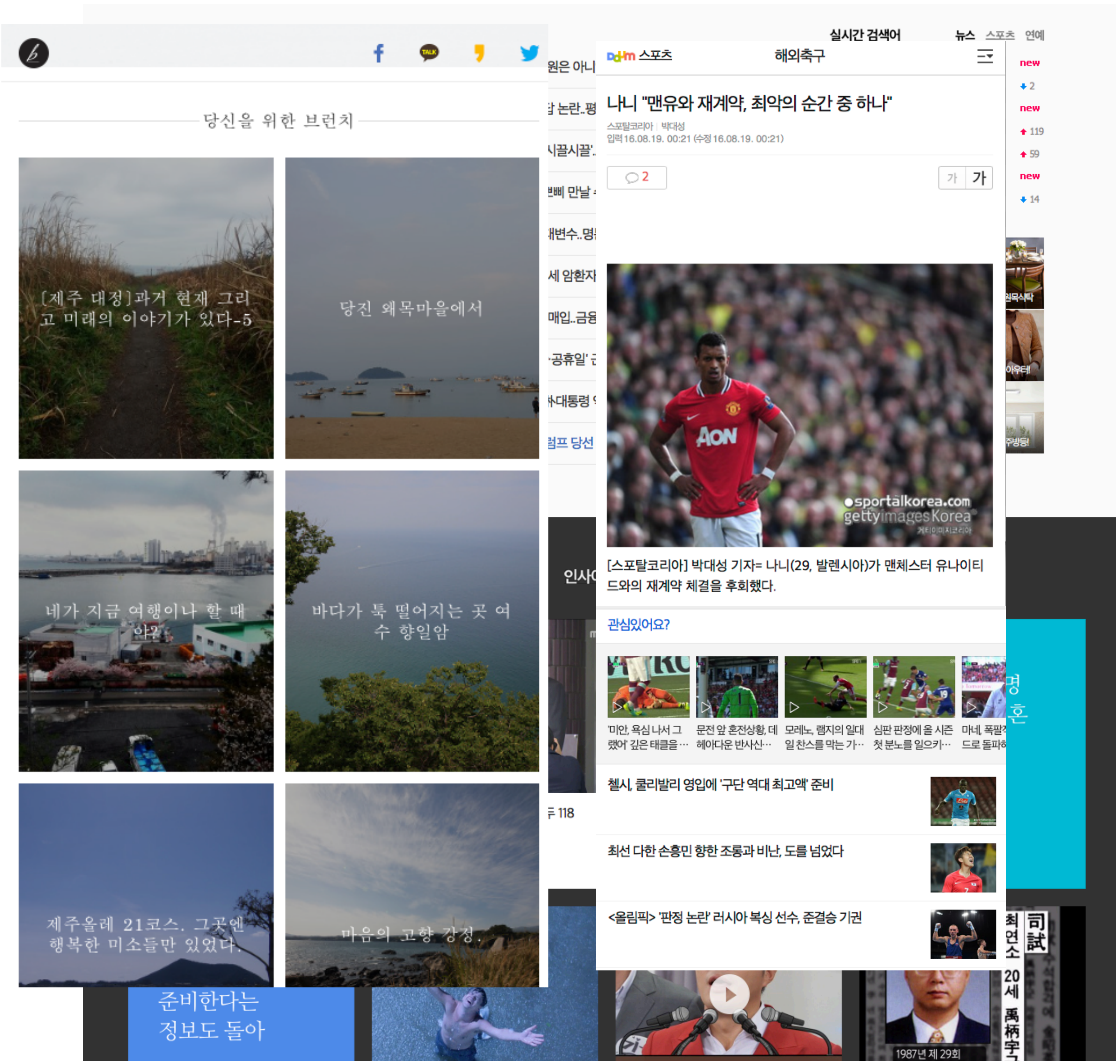
뉴스 정치
누가 백범을 암살했나?
野 "대북제재 맞느냐"..통일부 "지금 가장 효과적 수단"
"후쿠시마 오염토 170년 지나야 안전"..재사용계획 논란
삼성중 노협 28일 파업 찬반투표 후 상경 시위(종합)
부산에 안개 자욱..선박 입출항 통제, 항공기 무더기 지연(종합)
檢 칼끝에 선 박선숙..20년 정치인생 최대 위기
[Oh쎈 토크] 비스트 "차트1위, 날아갈 것 같아..팬들 위한 음악할 것"
정무위 김영란법 공방.."11조 경제 손실"vs"권익위, 휘둘려선 안돼"
'제이슨 본' 맷 데이먼-알리시아 비칸데르 7월 내한 확정
무대 위의 소금꽃 '문화노동자'

In Kakao...



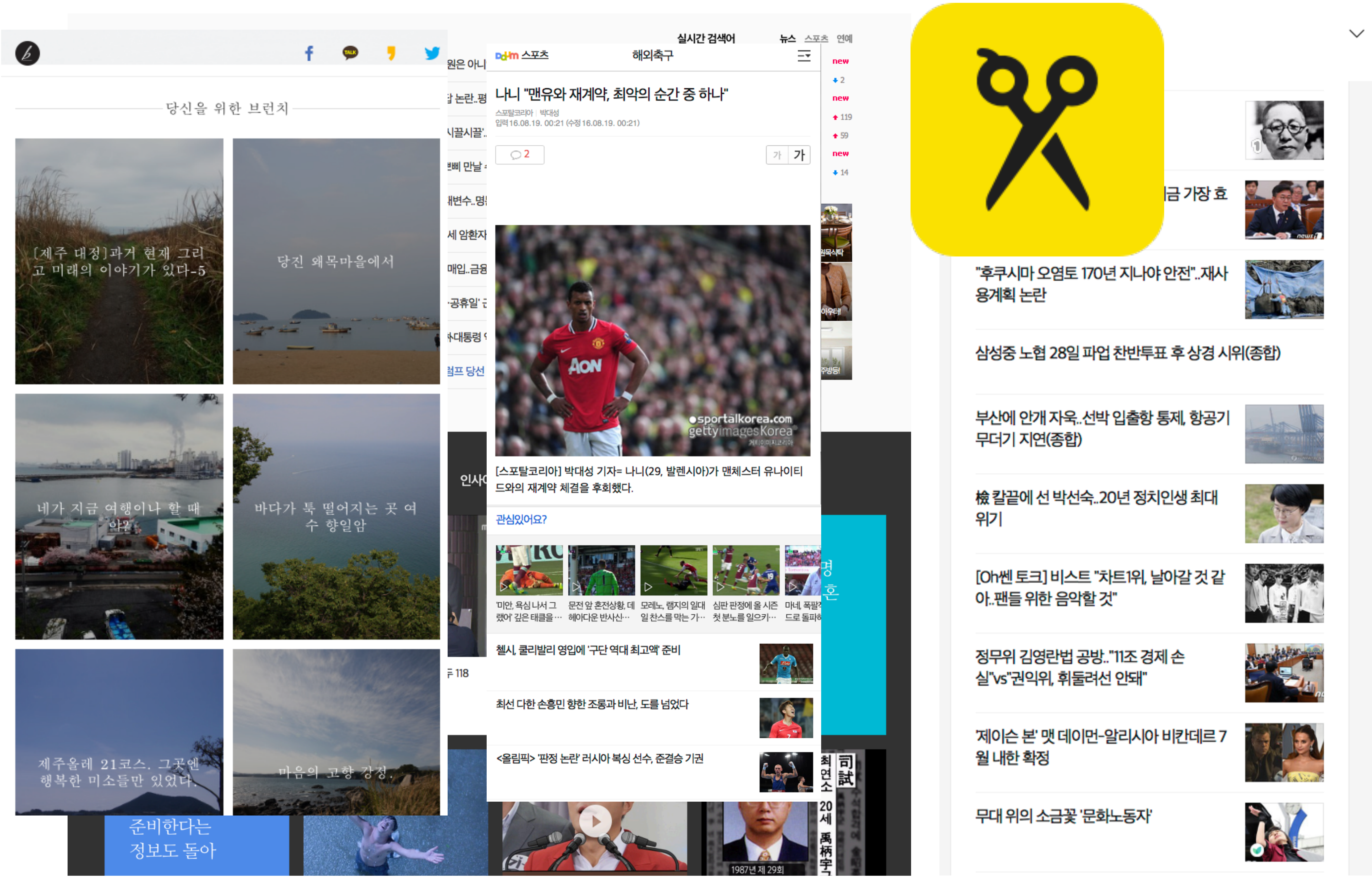
당신을 위한 브런치
[제주 대정]과거 현재 그리고 미래의 이야기가 있다-5
당진 왜목마을에서
네가 지금 여행이나 할 때 인가
바다가 툭 떨어지는 곳 여수 향일암
제주올레 21코스. 그곳엔 행복한 미소들만 있었다.
마음의 고향 강정.
준비한다는 정보도 돌아
실시간 검색어
뉴스 스포츠 연예
1 대동하야지도 부산 new
2 계엄령이란 2
3 프링글스 new
4 하야반대집회 119
5 류철균 59
6 특검 반대 의원 new
7 한국사 14번
인사이드 다음
그의 기사, 지독하거나 혹은 살벌하거나
스토리펀딩
어렸을 때 별명 '연애박사', 결혼 계획은..
조정석
우병우
뉴스 정치
누가 백범을 암살했나?
野 "대북제재 맞느냐"..통일부 "지금 가장 효과적 수단"
"후쿠시마 오염토 170년 지나야 안전"..재사용계획 논란
삼성중 노협 28일 파업 찬반투표 후 상경 시위(종합)
부산에 안개 자욱..선박 입출항 통제, 항공기 무더기 지연(종합)
檢 칼끝에 선 박선숙..20년 정치인생 최대 위기
[Oh쎈 토크] 비스트 "차트1위, 날아갈 것 같아..팬들 위한 음악할 것"
정무위 김영란법 공방.."11조 경제 손실"vs"권익위, 휘둘러선 안돼"
'제이슨 본' 맷 데이먼-알리시아 비칸데르 7월 내한 확정
무대 위의 소금꽃 '문화노동자'

In Kakao...


당신을 위한 브런치
[제주 대정]과거 현재 그리고 미래의 이야기가 있다-5
당진 왜목마을에서
네가 지금 여행이나 할 때 야?
바다가 툭 떨어지는 곳 여수 향일암
제주올레 21코스. 그곳엔 행복한 미소들만 있었다.
마음의 고향 강정.
준비한다는
정보도 돌아
실시간 검색어
뉴스 스포츠 연예
스포츠
해외축구
나니 "맨유와 재계약, 최악의 순간 중 하나"
스포탈코리아 | 박대성
입력 16.08.19. 00:21 (수정 16.08.19. 00:21)
2
가 가
sportalkorea.com
gettyimagesKorea
[스포탈코리아] 박대성 기자= 나니(29, 발렌시아)가 맨체스터 유나이티드와의 재계약 체결을 후회했다.
관심있어요?
첼시, 쿨리발리 영입에 '구단 역대 최고액' 준비
최선 다한 손흥민 향한 조롱과 비난, 도를 넘었다
<올림픽> '판정 논란' 러시아 복싱 선수, 준결승 기권
1987년 제 29회

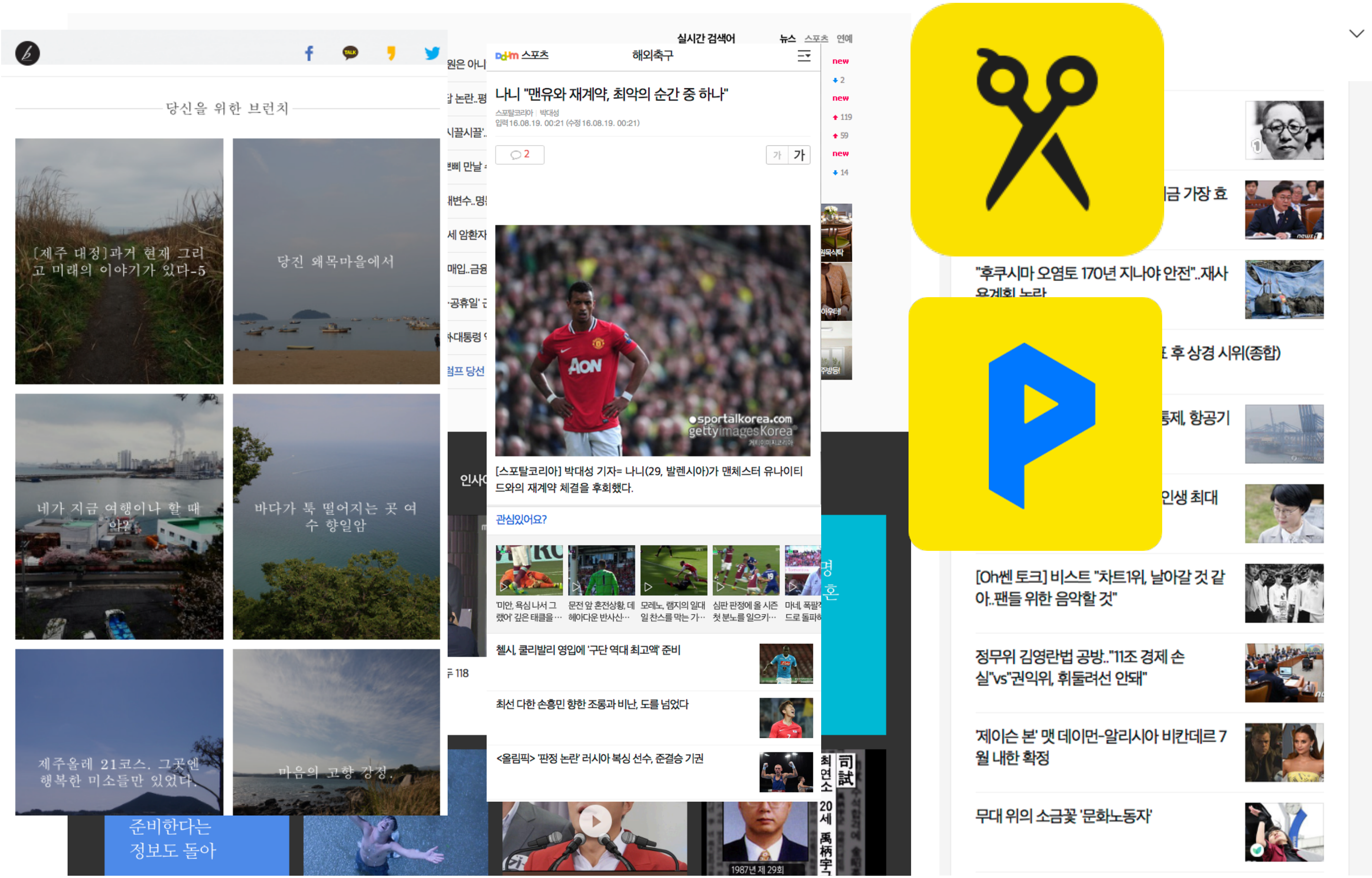

뉴스 정치
누가 백범을 암살했나?
野 "대북제재 맞느냐"..통일부 "지금 가장 효과적 수단"
"후쿠시마 오염토 170년 지나야 안전"..재사용계획 논란
삼성중 노협 28일 파업 찬반투표 후 상경 시위(종합)
부산에 안개 자욱..선박 입출항 통제, 항공기 무더기 지연(종합)
檢 칼끝에 선 박선숙..20년 정치인생 최대 위기
[Oh쎈 토크] 비스트 "차트1위, 날아갈 것 같아..팬들 위한 음악할 것"
정무위 김영란법 공방.."11조 경제 손실"vs"권익위, 휘둘려선 안돼"
'제이슨 본' 맷 데이먼-알리시아 비칸데르 7월 내한 확정
무대 위의 소금꽃 '문화노동자'

In Kakao...



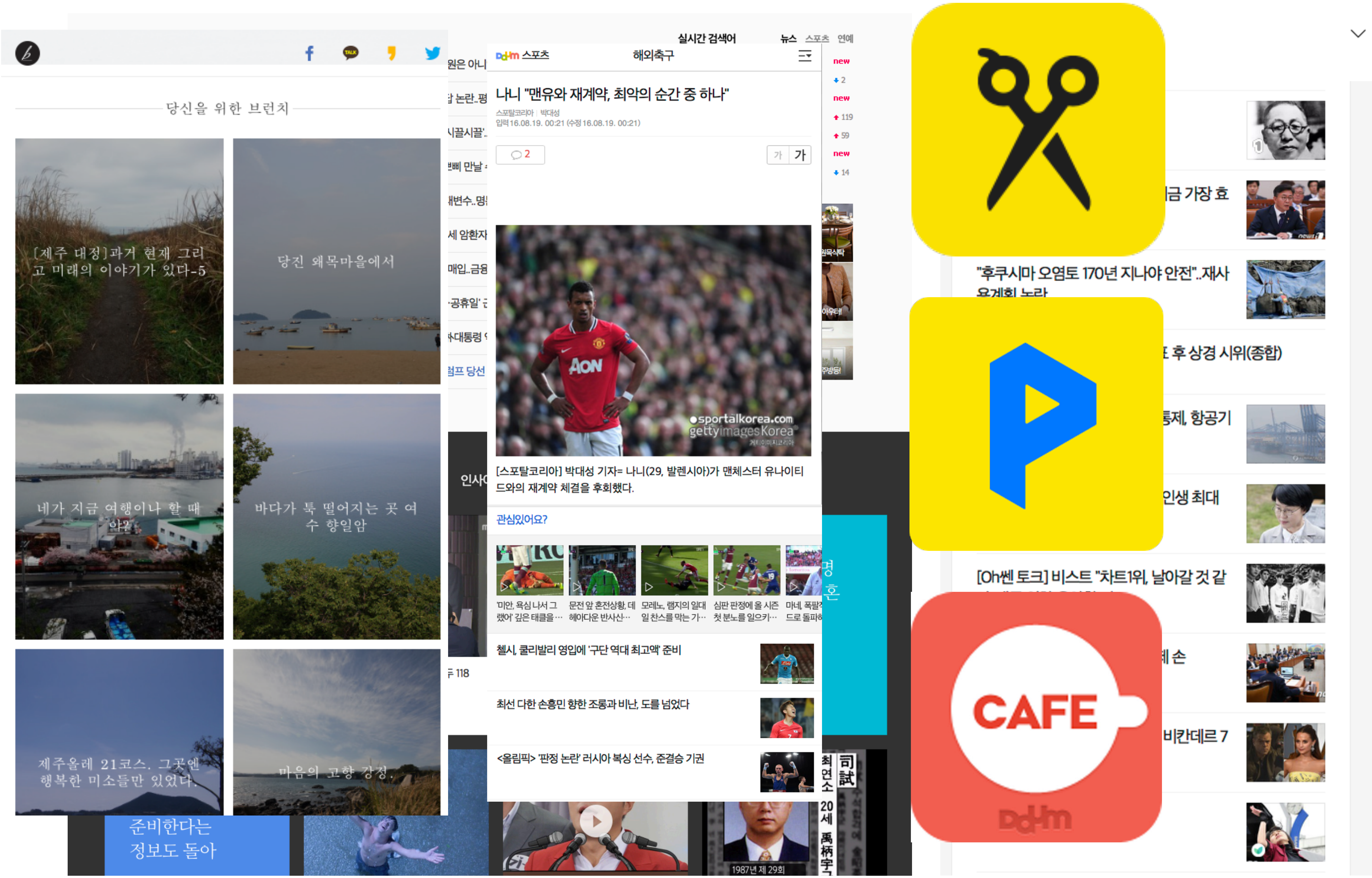
당신을 위한 브런치
[제주 대정]과거 현재 그리고 미래의 이야기가 있다-5
당진 왜목마을에서
네가 지금 여행이나 할 때야?
바다가 툭 떨어지는 곳 여수 향일암
제주올레 21코스. 그곳엔 행복한 미소들만 있었다.
마음의 고향 강정.
Daum 스포츠
해외축구
나니 "맨유와 재계약, 최악의 순간 중 하나"
[스포탈코리아] 박대성 기자= 나니(29, 발렌시아)가 맨체스터 유나이티드와의 재계약 체결을 후회했다.
관심있어요?
첼시, 쿨리발리 영입에 '구단 역대 최고액' 준비
최선 다한 손흥민 향한 조롱과 비난, 도를 넘었다
<올림픽> '판정 논란' 러시아 복싱 선수, 준결승 기권
실시간 검색어
"후쿠시마 오염토 170년 지나야 안전"..재사용계획 논란
삼성중 노협 28일 파업 찬반투표 후 상경 시위(종합)
부산에 안개 자욱..선박 입출항 통제, 항공기 무더기 지연(종합)
檢 칼끝에 선 박선숙..20년 정치인생 최대 위기
[Oh쎈 토크] 비스트 "차트1위, 날아갈 것 같아..팬들 위한 음악할 것"
정무위 김영란법 공방.."11조 경제 손실"vs"권익위, 휘둘려선 안돼"
'제이슨 본' 맷 데이먼-알리시아 비칸데르 7월 내한 확정
무대 위의 소금꽃 '문화노동자'

In Kakao...

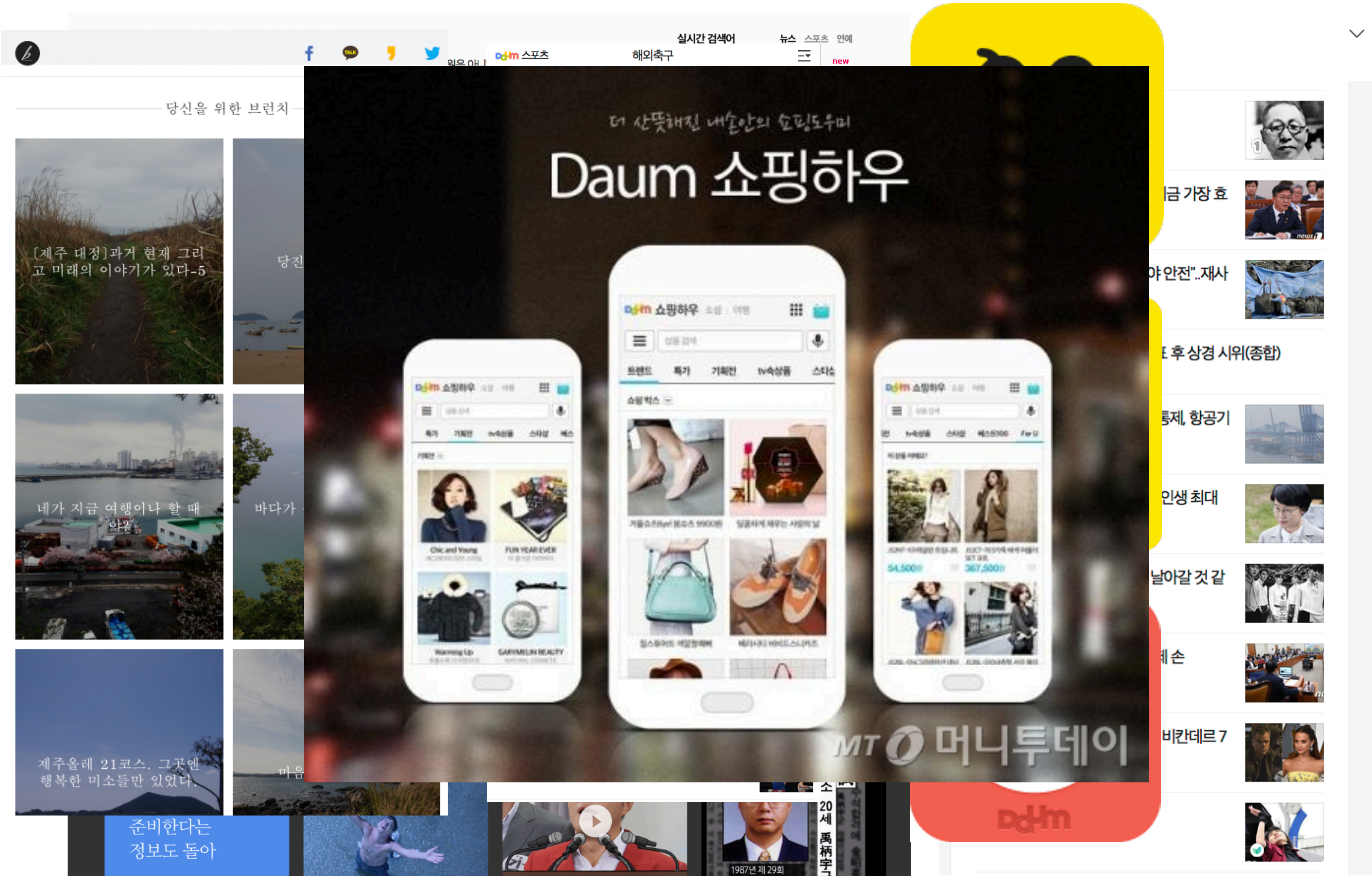


당신을 위한 브런치
[제주 대정]과거 현재 그리고 미래의 이야기가 있다-5
당진 왜목마을에서
네가 지금 여행이나 할 때 알?
바다가 툭 떨어지는 곳 여수 향일암
제주올레 21코스. 그곳엔 행복한 미소들만 있었다.
마음의 고향 강정.
준비한다는
정보도 돌아
실시간 검색어
뉴스 스포츠 연예
스포츠
해외축구
나니 "맨유와 재계약, 최악의 순간 중 하나"
입력 16.08.19. 00:21 (수정 16.08.19. 00:21)
[스포탈코리아] 박대성 기자= 나니(29, 발렌시아)가 맨체스터 유나이티드와의 재계약 체결을 후회했다.
관심있어요?
첼시, 쿨리발리 영입에 '구단 역대 최고액' 준비
최선 다한 손흥민 향한 조롱과 비난, 도를 넘었다
<올림픽> '판정 논란' 러시아 복싱 선수, 준결승 기권
"후쿠시마 오염토 170년 지나야 안전"..재사용계획 논란
[Oh쎈 토크] 비스트 "차트1위, 날아갈 것 같아..팬들 위한 음악할 것"
정무위 김영란법 공방.."11조 경제 손실"vs"권익위, 휘둘려선 안돼"
'제이슨 본' 맷 데이먼-알리시아 비칸데르 7월 내한 확정
무대 위의 소금꽃 '문화노동자'
1987년 제 29회

In Kakao...

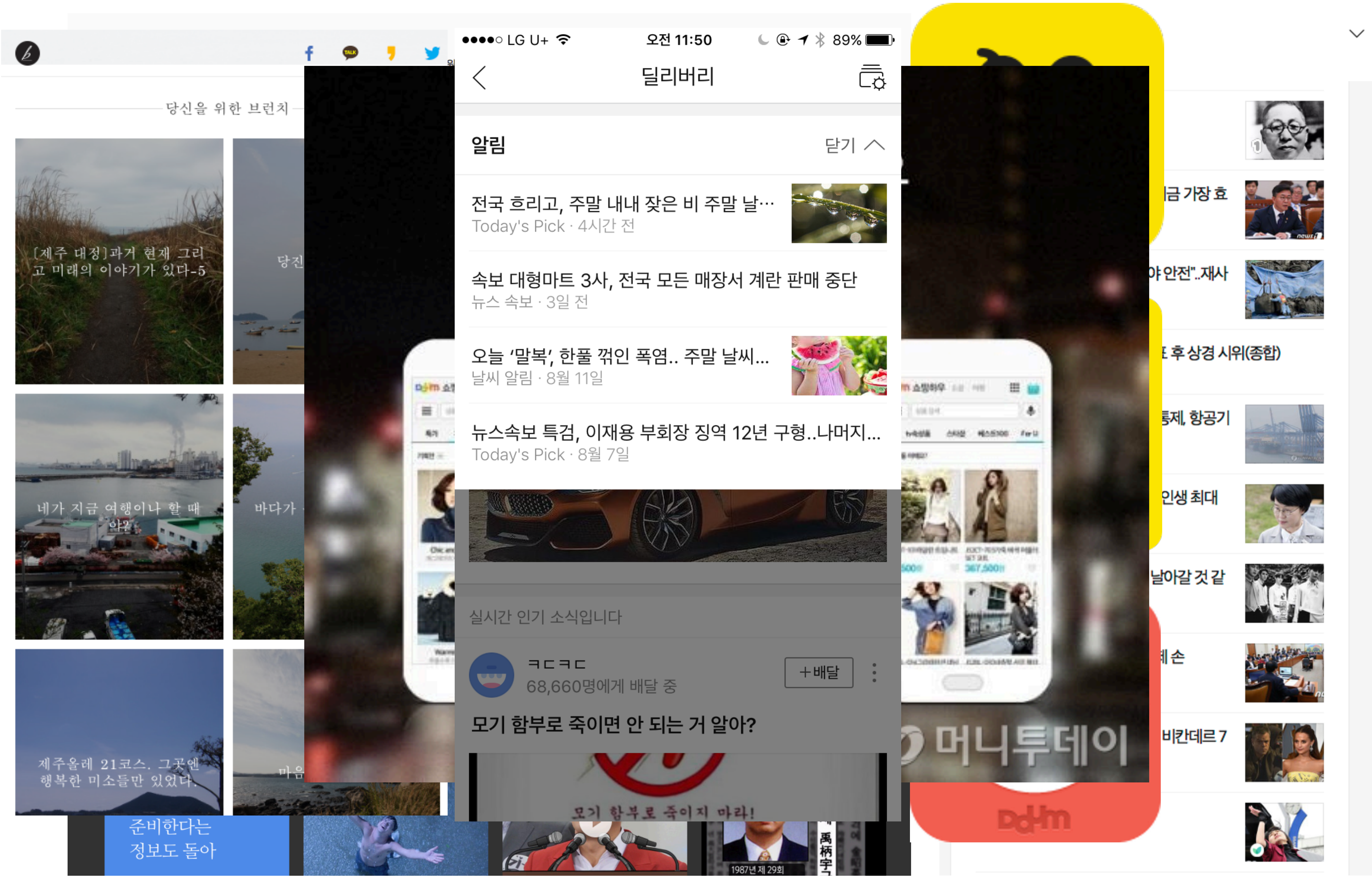

당신을 위한 브런치
[제주 대정]과거 현재 그리고 미래의 이야기가 있다-5
당진 왜목마을에서
네가 지금 여행이나 할 때야?
바다가 툭 떨어지는 곳 여수 향일암
제주올레 21코스. 그곳엔 행복한 미소들만 있었다.
마음의 고향 강정.
준비한다는 정보도 돌아
실시간 검색어
스포츠
해외축구
나니 "맨유와 재계약, 최악의 순간 중 하나"
[스포탈코리아] 박대성 기자= 나니(29, 발렌시아)가 맨체스터 유나이티드와의 재계약 체결을 후회했다.
관심있어요?
첼시, 쿨리발리 영입에 '구단 역대 최고액' 준비
최선 다한 손흥민 향한 조롱과 비난, 도를 넘었다
<올림픽> '판정 논란' 러시아 복싱 선수, 준결승 기권
"후쿠시마 오염토 170년 지나야 안전"..재사
[Oh쎈 토크] 비스트 "차트1위, 날아갈 것 같
CAFE

In Kakao...



당신을 위한 브런치
[제주 대정]과거 현재 그리고 미래의 이야기가 있다-5
네가 지금 여행이나 할 때 야?
제주올레 21코스. 그곳엔 행복한 미소들만 있었다.
실시간 검색어
해외축구
더 산뜻해진 내손안의 쇼핑도우미
Daum 쇼핑하우
MT 머니투데이
준비한다는
정보도 돌아

In Kakao...



당신을 위한 브런치
[제주 대정]과거 현재 그리고 미래의 이야기가 있다-5
네가 지금 여행이나 할 때 야?
제주올레 21코스. 그곳엔 행복한 미소들만 있었다.
준비한다는 정보도 돌아
LG U+
오전 11:50
89%
딜리버리
알림
닫기
전국 흐리고, 주말 내내 잦은 비 주말 날…
Today's Pick · 4시간 전
속보 대형마트 3사, 전국 모든 매장서 계란 판매 중단
뉴스 속보 · 3일 전
오늘 '말복', 한풀 꺾인 폭염.. 주말 날씨...
날씨 알림 · 8월 11일
뉴스속보 특검, 이재용 부회장 징역 12년 구형..나머지...
Today's Pick · 8월 7일
실시간 인기 소식입니다
ㅋㄷㅋㄷ
68,660명에게 배달 중
+배달
모기 함부로 죽이면 안 되는 거 알아?
모기 함부로 죽이지 마라!
머니투데이
후 상경 시위(종합)
인생 최대

In Kakao...

당신을 위한 브런치
[제주 대정]과거 현재 그리고 미래의 이야기가 있다-5
제주올레 21코스. 그곳엔 행복한 미소들만 있었다.
LG U+
오전 11:50
89%
딜리버리
알림
닫기
kakao mini
실시간 인기 소식입니다
ㅋㄷㅋㄷ
68,660명에게 배달 중
+배달
모기 함부로 죽이면 안 되는 거 알아?
모기 함부로 죽이지 마라!
머니투데이
Daum

In Kakao...

Melon

Contents

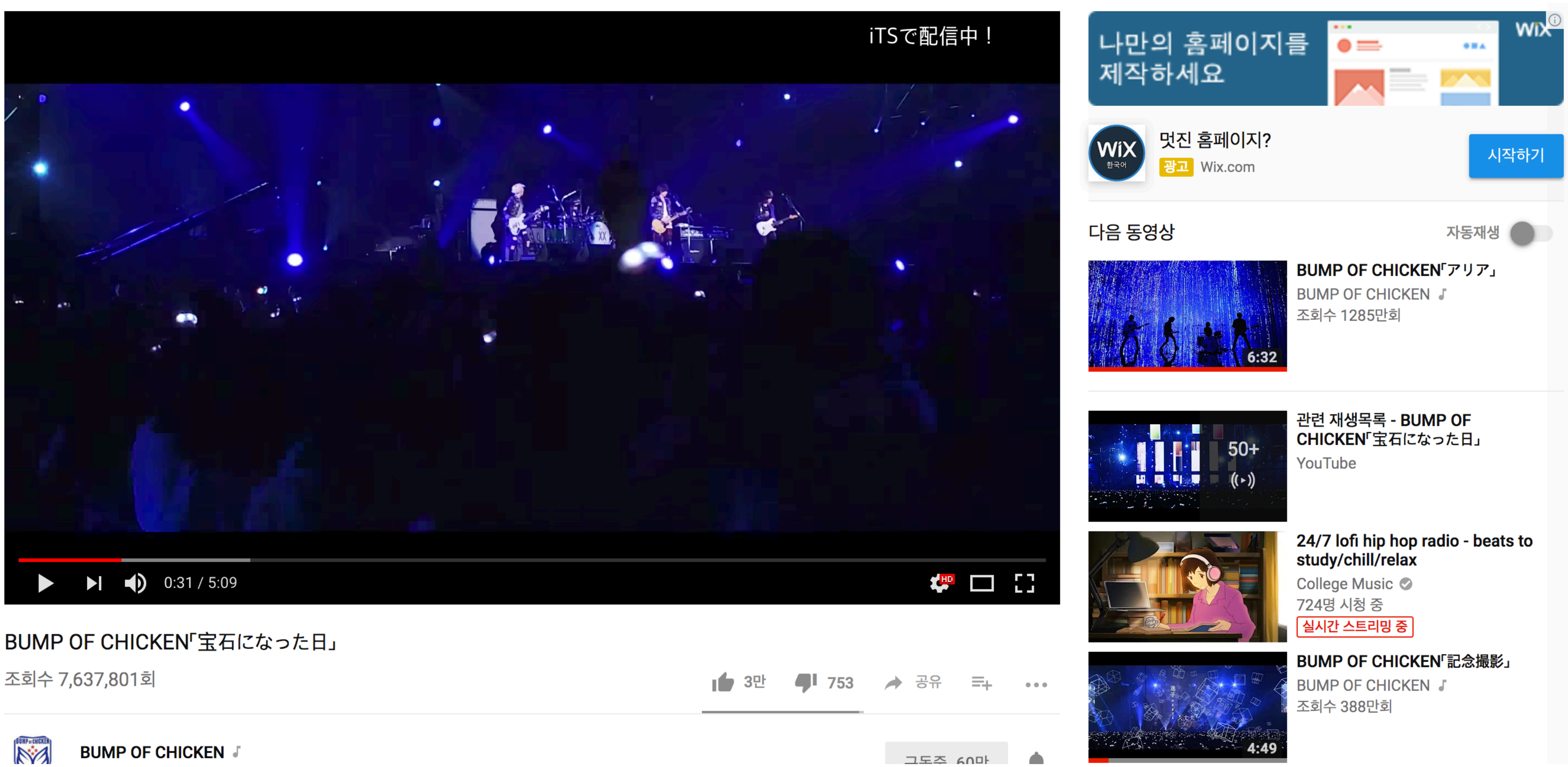
Introduction to Recommender System

What is a recommender system?

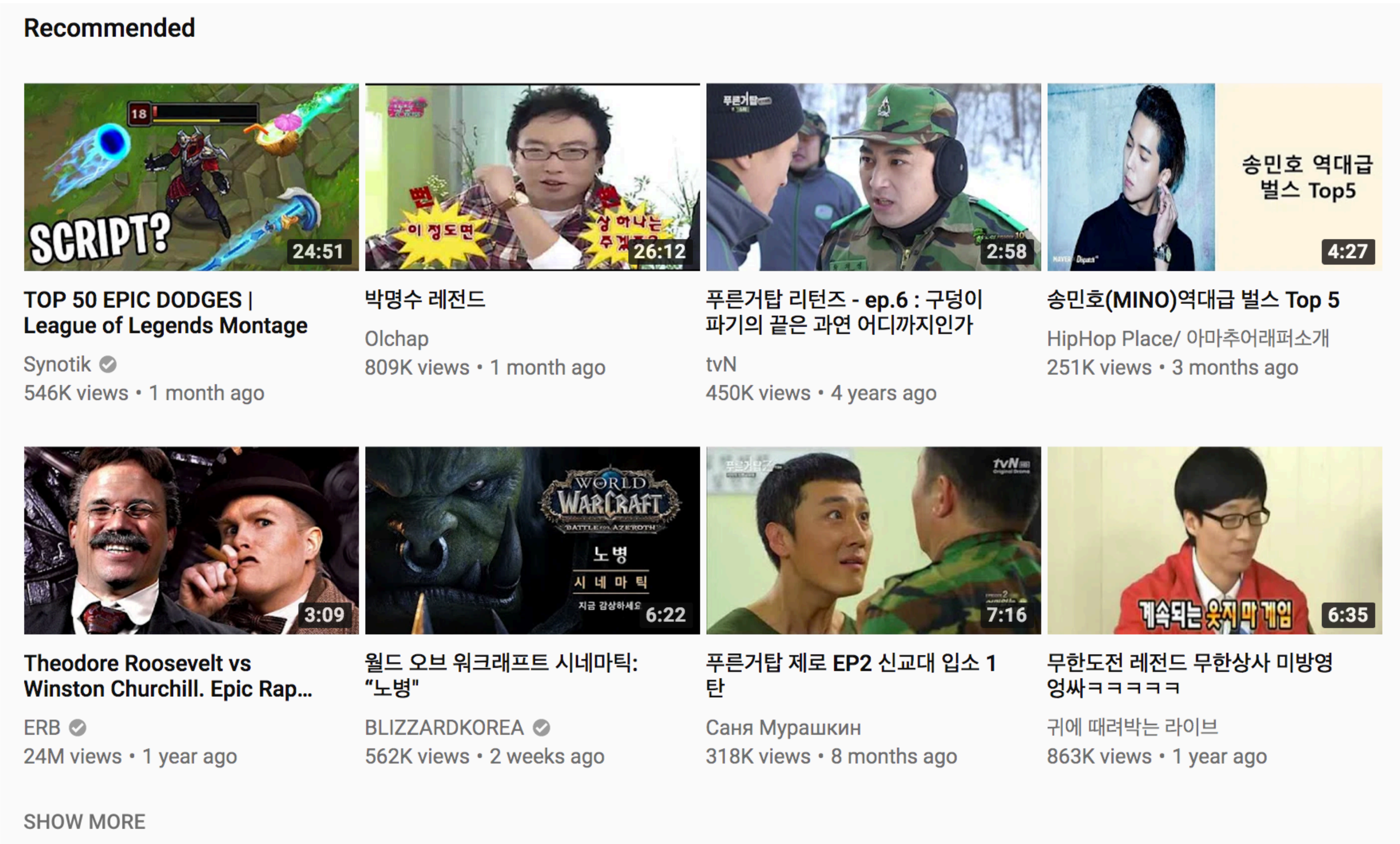
A recommender system recommends **items** to **users** to optimize a utility composed of one or more **objectives**

Reference: RecSys 2016: Tutorial on Lessons Learned from Building Real-life Recommender Systems

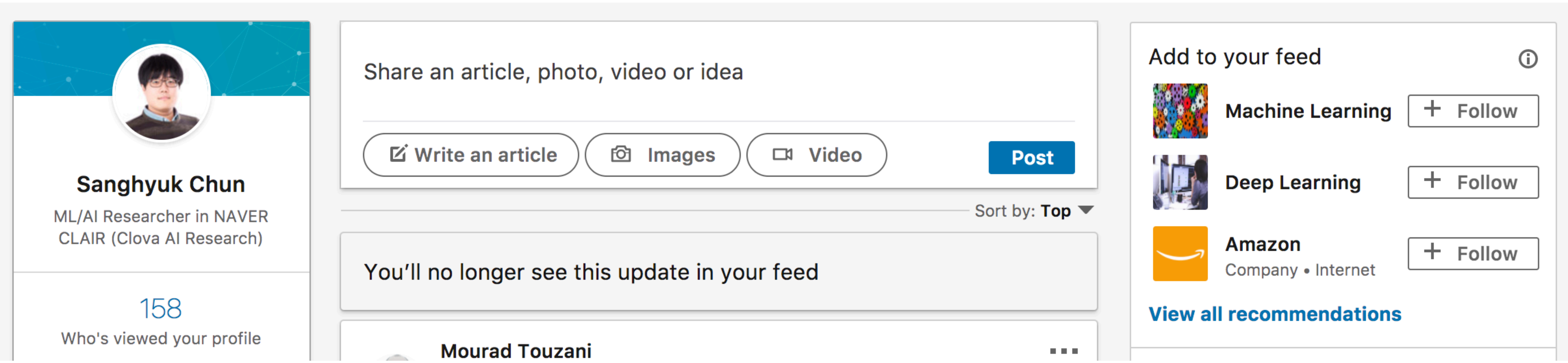
RecSys Example 1: YouTube Up next



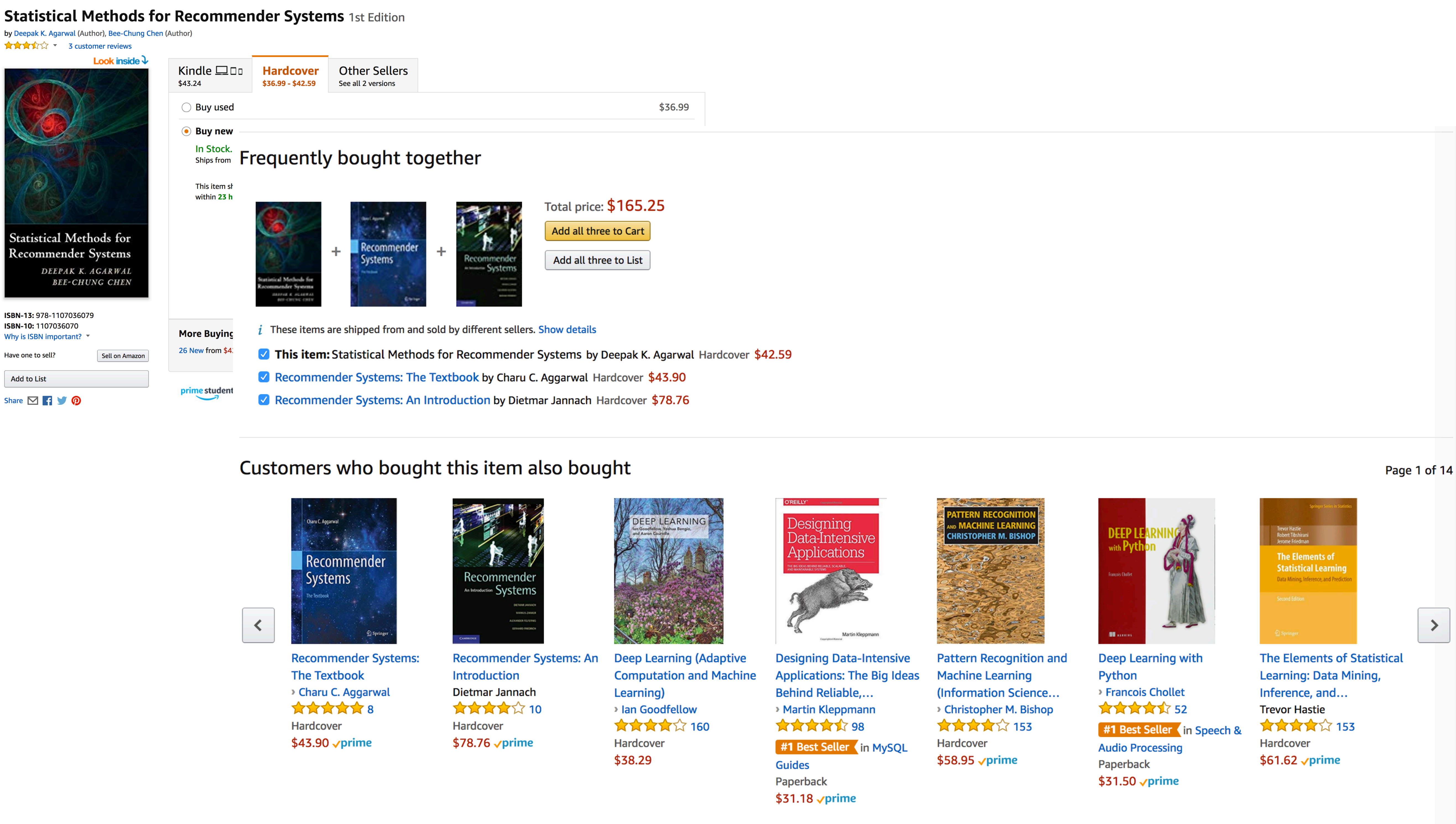
RecSys Example 2: YouTube Recommended



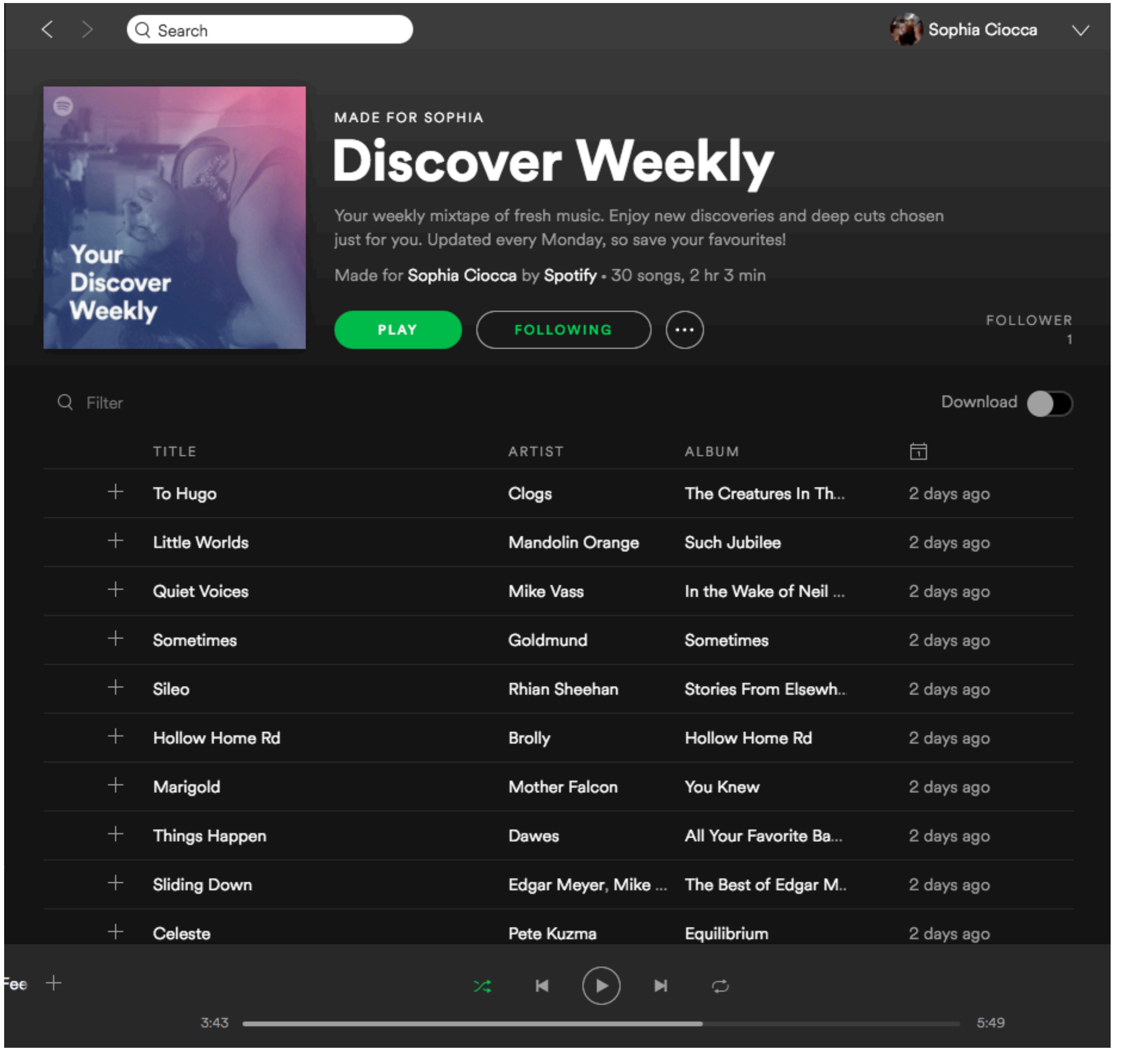
RecSys Example 3: LinkedIn Personalized Feed



RecSys Example 4: Amazon item recommendation

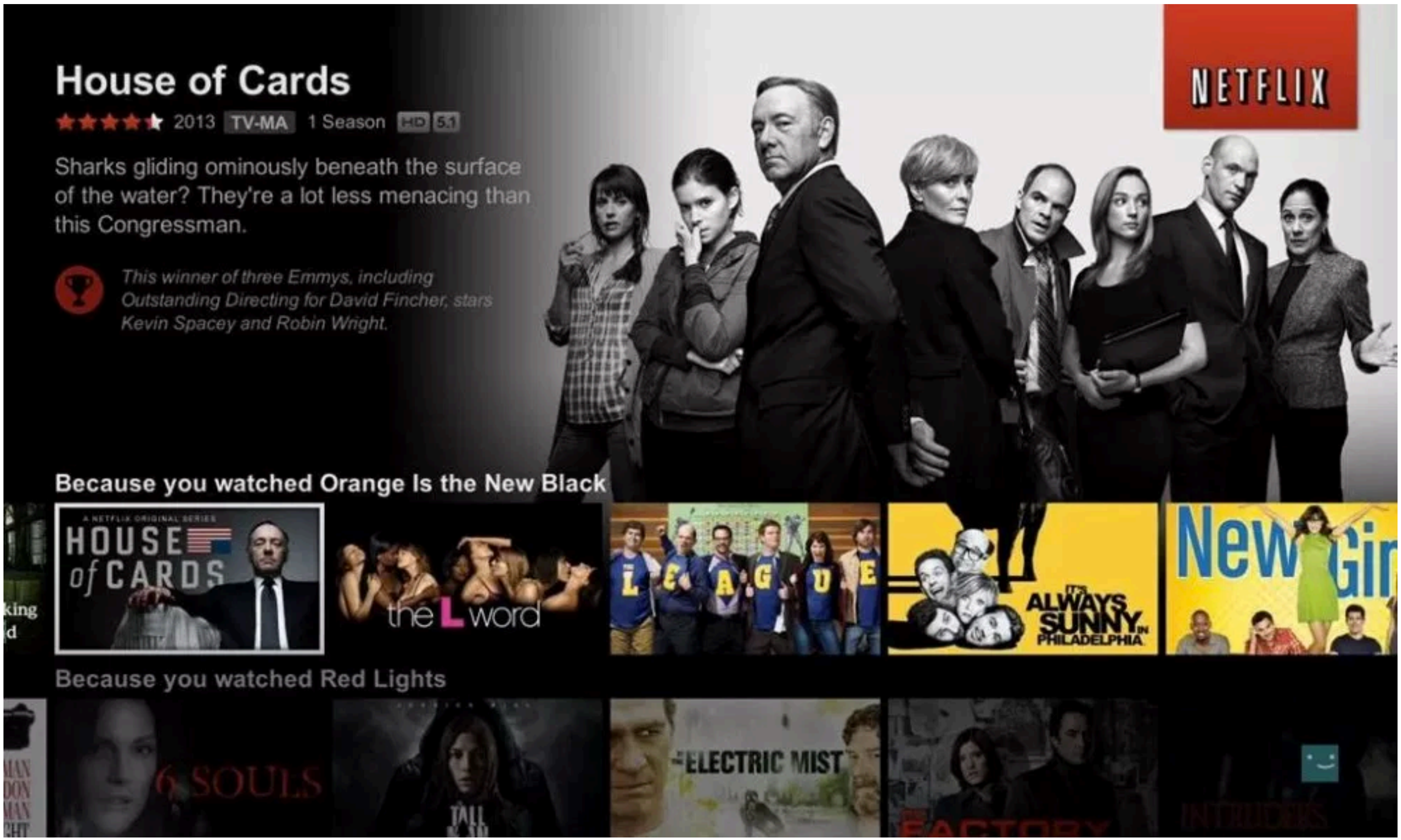


RecSys Example 5: Spotify Discover



https://medium.com/s/story/spotifys-discover-weekly-how-machine-learning-finds-your-new-music-19a41ab76efe

RecSys Example 6: Netflix Recommendation



http://www.shalomeir.com/2014/11/netflix-prize-1/

Amazon: 35% of the purchases are from recommendation

Alibaba: up to 20% growth of conversion rate from personalized landing pages (during Chinese shopping festival)

YouTube: 70% of the watching is from recommendation

Netflix: 75% of what people are watching on Netflix comes from recommendations + Employing a recommender system enables Netflix to save around $1 billion each year

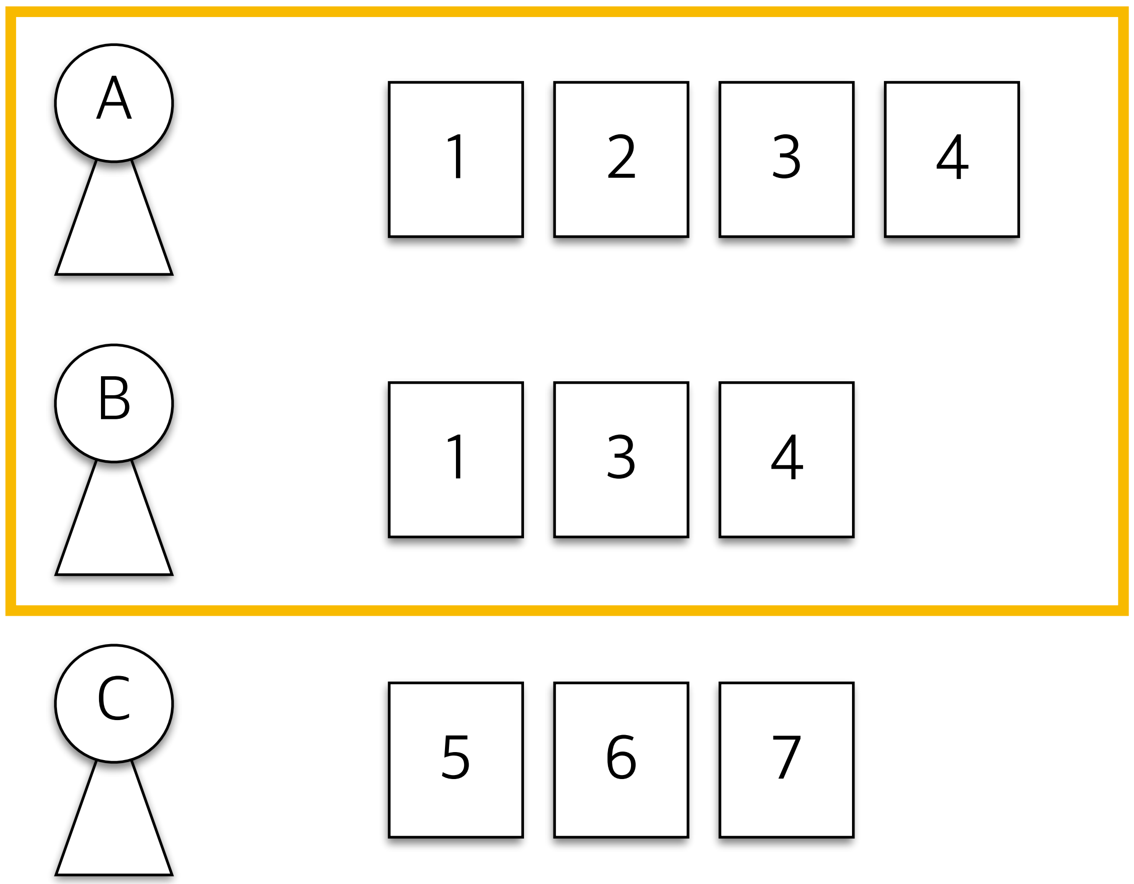
https://tryolabs.com/blog/introduction-to-recommender-systems/

Traditional Methods

- CF / CB
- Netflix Problem

Traditional Recommendations

Collaborative Filtering (CF)



https://en.wikipedia.org/wiki/Collaborative_filtering#/media/File:Collaborative_filtering.gif

https://en.wikipedia.org/wiki/Collaborative_filtering#/media/File:Collaborative_filtering.gif

Traditional Recommendations

Contents Based Filtering (CB)

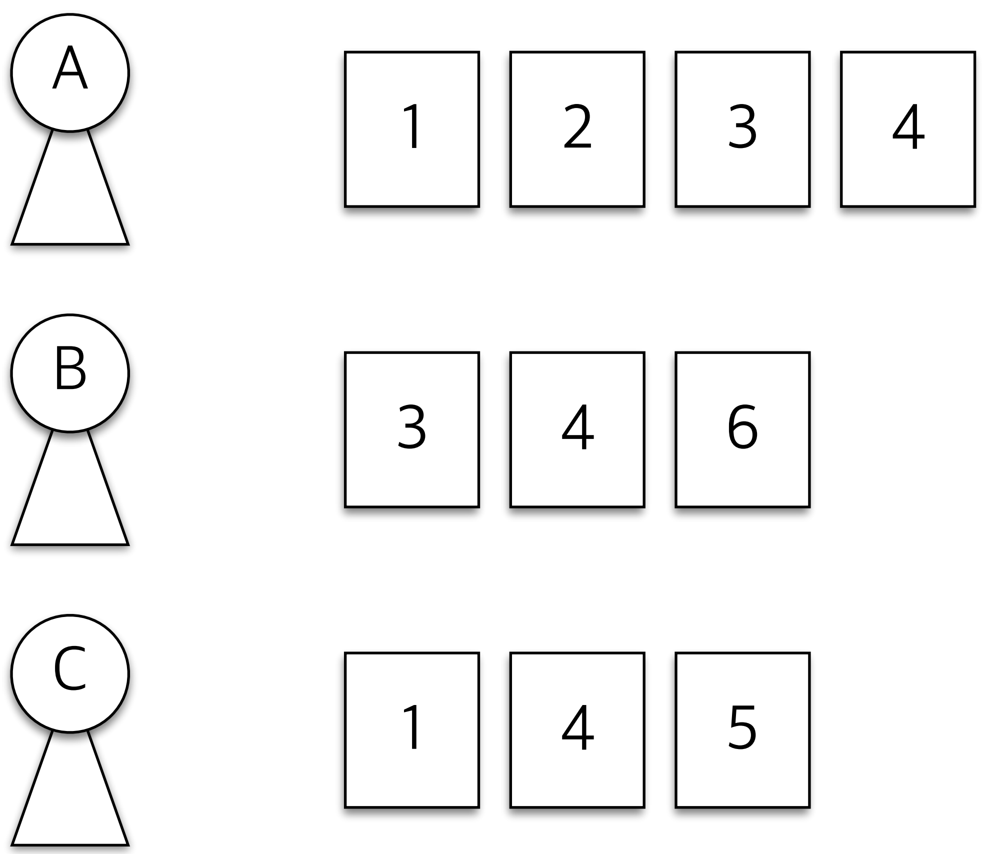
BUMP OF CHICKEN「Hello,world!」LIVE...
BUMP OF CHICKEN
조회수 144만회 • 2년 전

BUMP OF CHICKEN feat. HATSUNE MIKU「ray」
BUMP OF CHICKEN
조회수 1295만회 • 4년 전

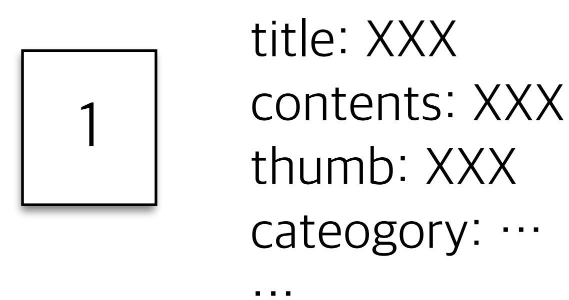
RecSys 2016: Tutorial on Lessons Learned from...
ACM RecSys
조회수 2.1천회 • 1년 전

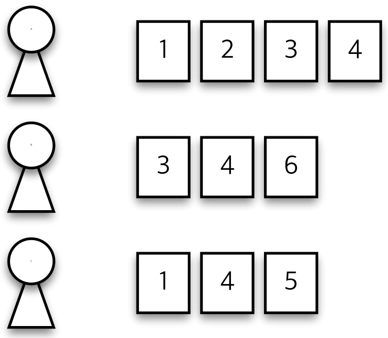
Data for Recommendation

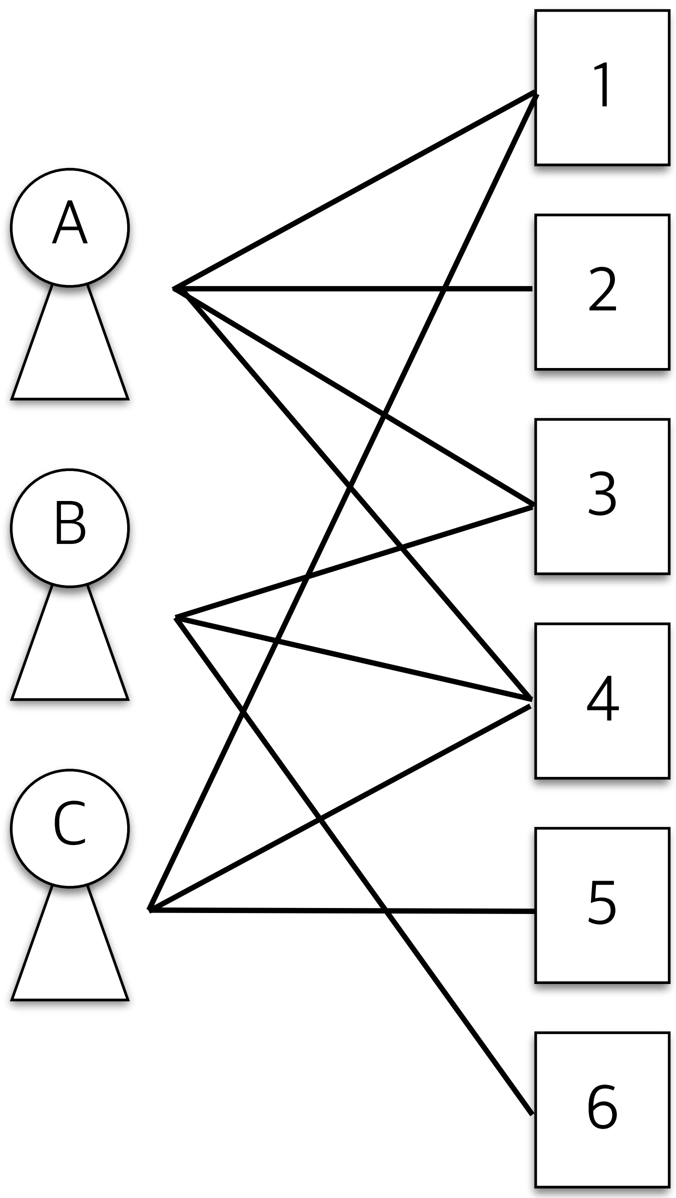
User History
(rating, view, purchase, ...)



Content data
(metadata, raw data, ...)







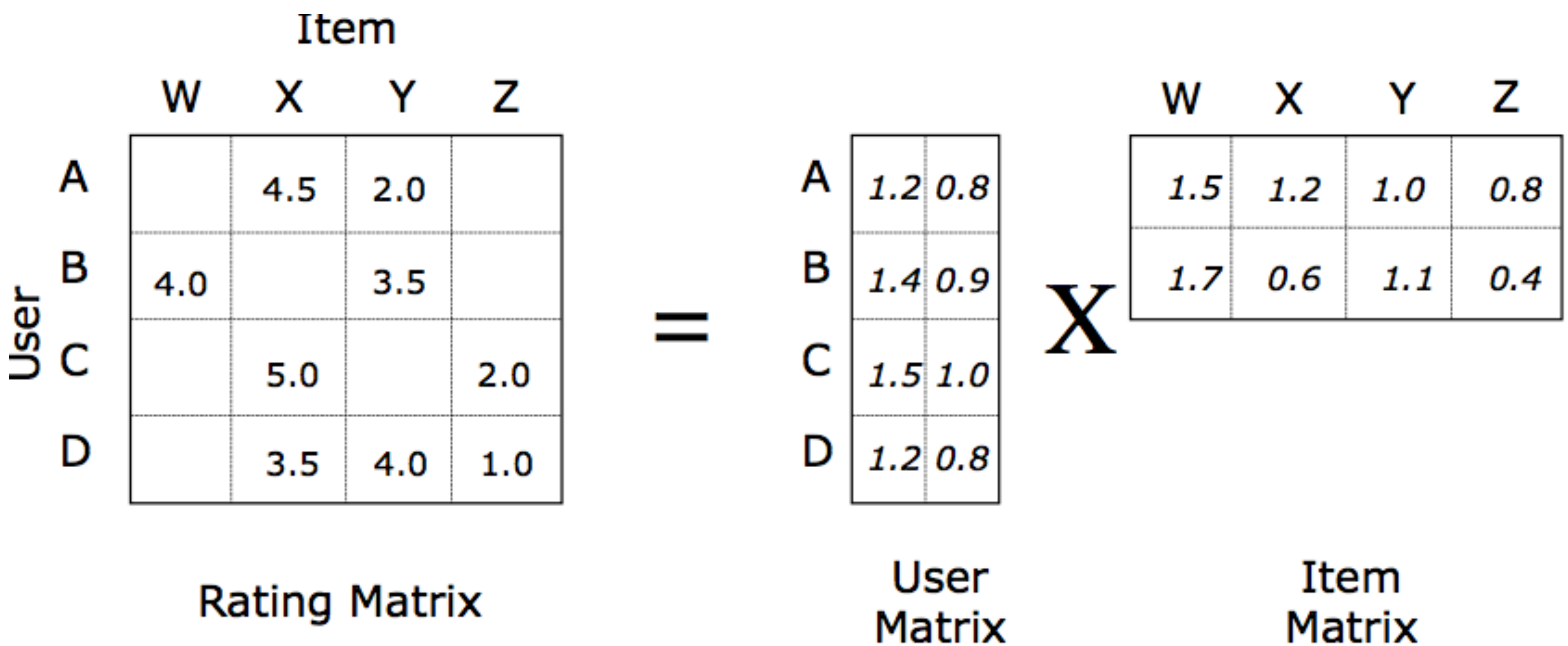
Graph Form

	A	B	C
1	1		1
2	1		
3	1	1	
4		1	1
5			1
6		1	

Matrix Form

Traditional CF methods

- Low Rank Matrix Factorization



Brief Overview of Matrix Factorization

$$\min_{\hat{R}} \sum_{u,i \in \kappa} (r_{ui} - \hat{r}_{ui})^2 \quad \text{s.t.} \quad \text{rank}(\hat{R}) = k.$$

$$\min_{P,Q} \sum_{u,i \in \kappa} (r_{ui} - p_u \cdot q_i)^2 + \lambda(\|p_u\|_2^2 + \|q_i\|_2^2).$$

$$\min_{x_\star, y_\star} \sum_{u,i} c_{ui}(p_{ui} - x_u^T y_i)^2 + \lambda \left(\sum_u \|x_u\|^2 + \sum_i \|y_i\|^2 \right)$$

More readings

- **[ALS + implicit feedback]** Hu, Yifan, Yehuda Koren, and Chris Volinsky. "Collaborative filtering for implicit feedback datasets." Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on. Ieee, 2008.

- **[Overview paper]** Koren, Yehuda, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems." Computer 8 (2009): 30-37.

- **[PMF]** Mnih, Andriy, and Ruslan R. Salakhutdinov. "Probabilistic matrix factorization." Advances in neural information processing systems. 2008.

- **[Logistic MF]** Johnson, Christopher C. "Logistic matrix factorization for implicit feedback data." Advances in Neural Information Processing Systems 27 (2014).

- **[BPR-MF]** Gantner, Zeno, et al. "Personalized ranking for non-uniformly sampled items." Proceedings of KDD Cup 2011. 2012.

- **[AutoEncoder]** Wang, Hao, Naiyan Wang, and Dit-Yan Yeung. "Collaborative deep learning for recommender systems." Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2015.

Recommendation using features

Item-to-item recommendation

- return "most K-similar items" to the query item

Personalized recommendation

- return items observed by "most K-similar users" to the query user

Limitation of CF

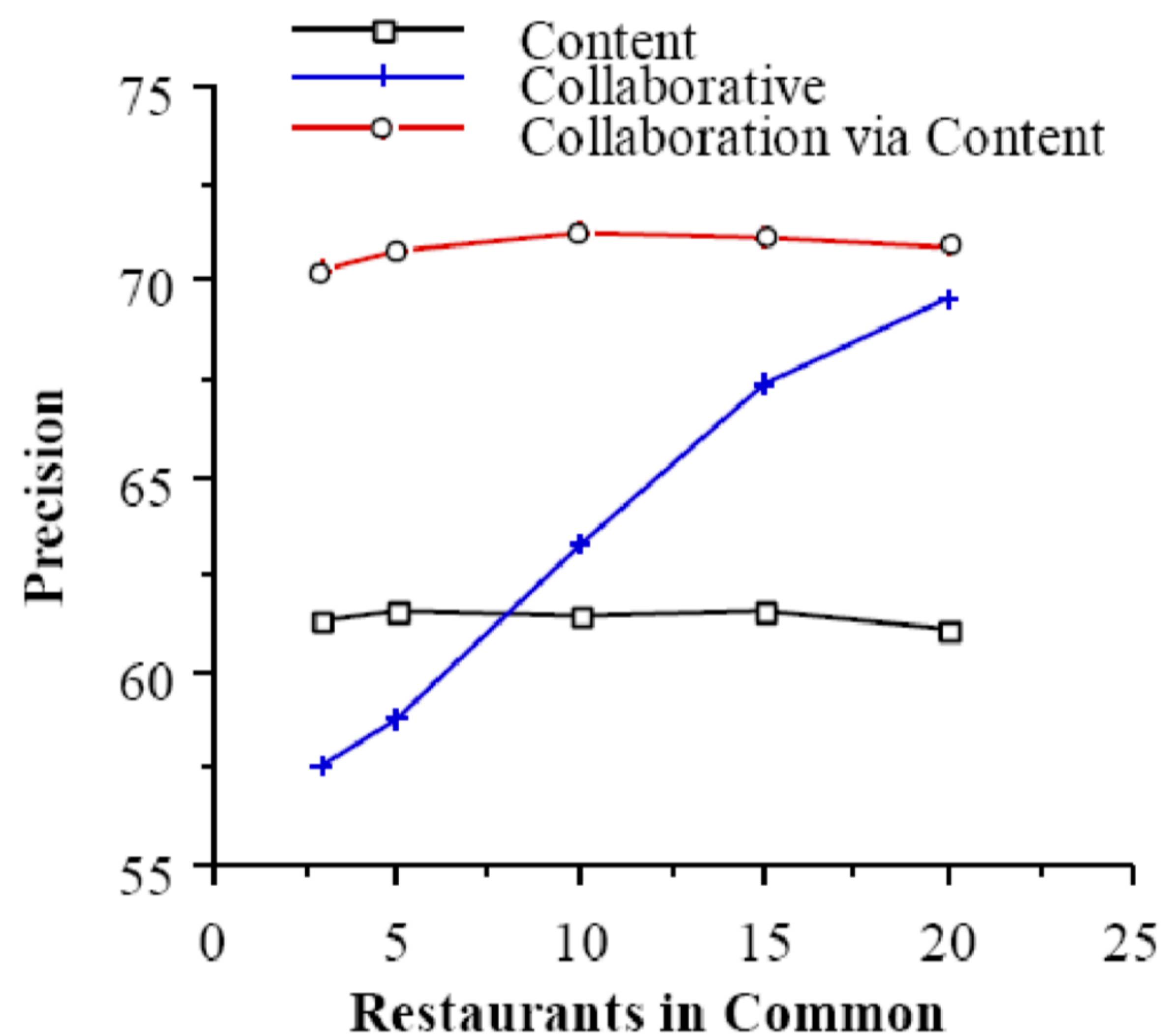
- Cold start: There needs to be enough other users already in the system to find a match. New items need to get enough ratings.

- Popularity bias: Hard to recommend items to someone with unique tastes. Tends to recommend popular items (items from the tail do not get so much data)

https://www.slideshare.net/xamat/recommender-systems-machine-learning-summer-school-2014-cmu

CF vs. CB

- Content–based recommendation with Bayesian classifier
- Collaborative is standard using Pearson correlation
- Collaboration via content uses the content-based user profiles



Averaged on 44 users

Precision computed in top 3 recommendations

Ensemble methods

Hybridization Method	Description
Weighted	Outputs from several techniques (in the form of scores or votes) are combined with different degrees of importance to offer final recommendations
Switching	Depending on situation, the system changes from one technique to another
Mixed	Recommendations from several techniques are presented at the same time
Feature combination	Features from different recommendation sources are combined as input to a single technique
Cascade	The output from one technique is used as input of another that refines the result
Feature augmentation	The output from one technique is used as input features to another
Meta-level	The model learned by one recommender is used as input to another

https://www.slideshare.net/xamat/recommender-systems-machine-learning-summer-school-2014-cmu

Netflix Prize

2006.10 ~ 2009.07

Improve by 10% RMSE = $ 1M!
(Winner Takes ALL!)

Baseline algorithm (Cinematch): 0.9525

$$\sqrt{\frac{1}{|\mathcal{X}|}\sum_{i,j\in\mathcal{X}}(r_{ij}-\widehat{r}_{ij})^2}$$

Winners of Netflix Prize

Grand Prize: team "BellKor’s Pragmatic Chaos"

- 2007 Winner: team "BellKor" (Bell & Koren)
(improved by **8.26%**)

- 2008 Winner: team "BellKor in Chaos"
(Union of team BellKor and team Big Chaos)
(improved by **9.44%**)

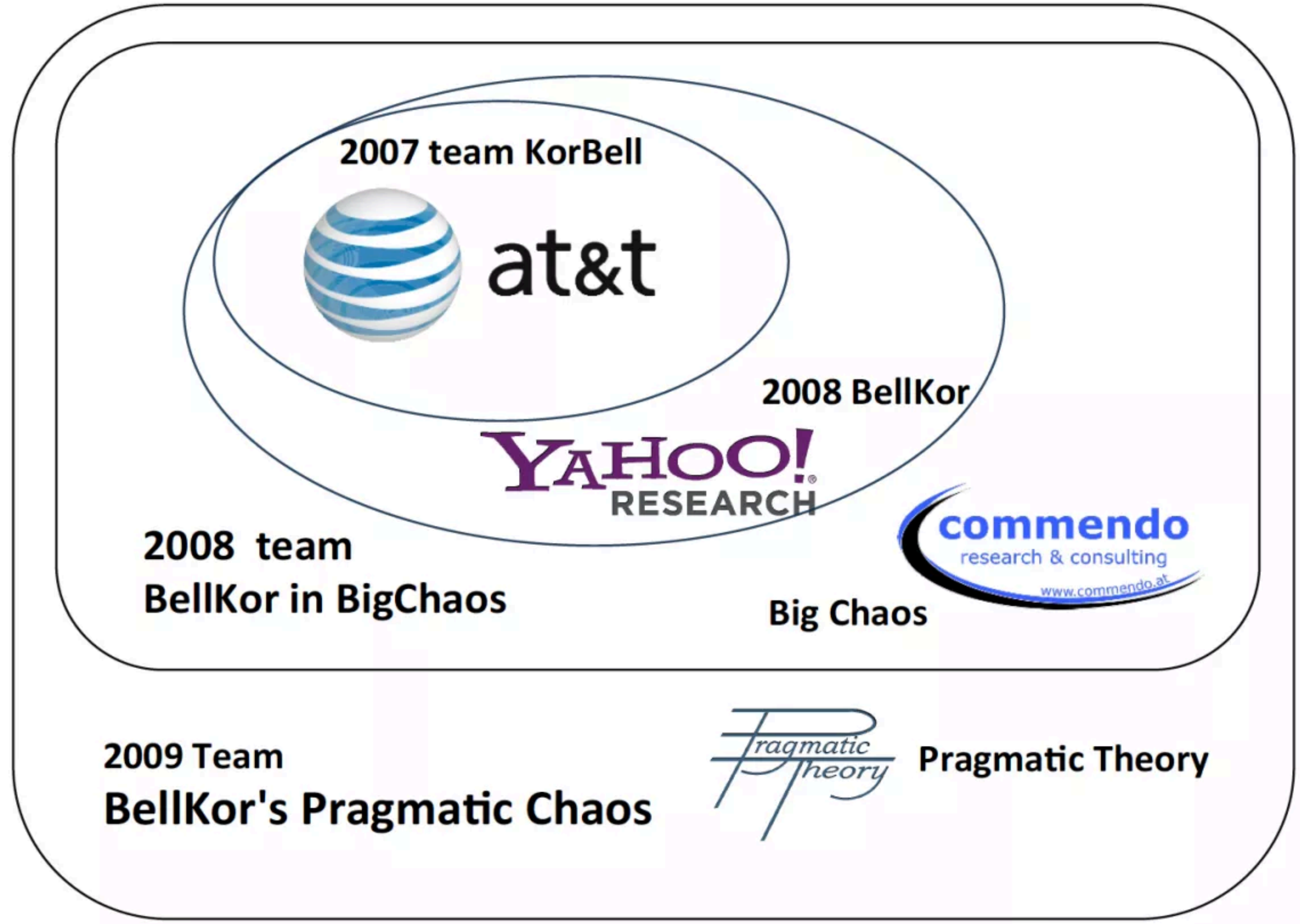
Winners of Netflix Prize

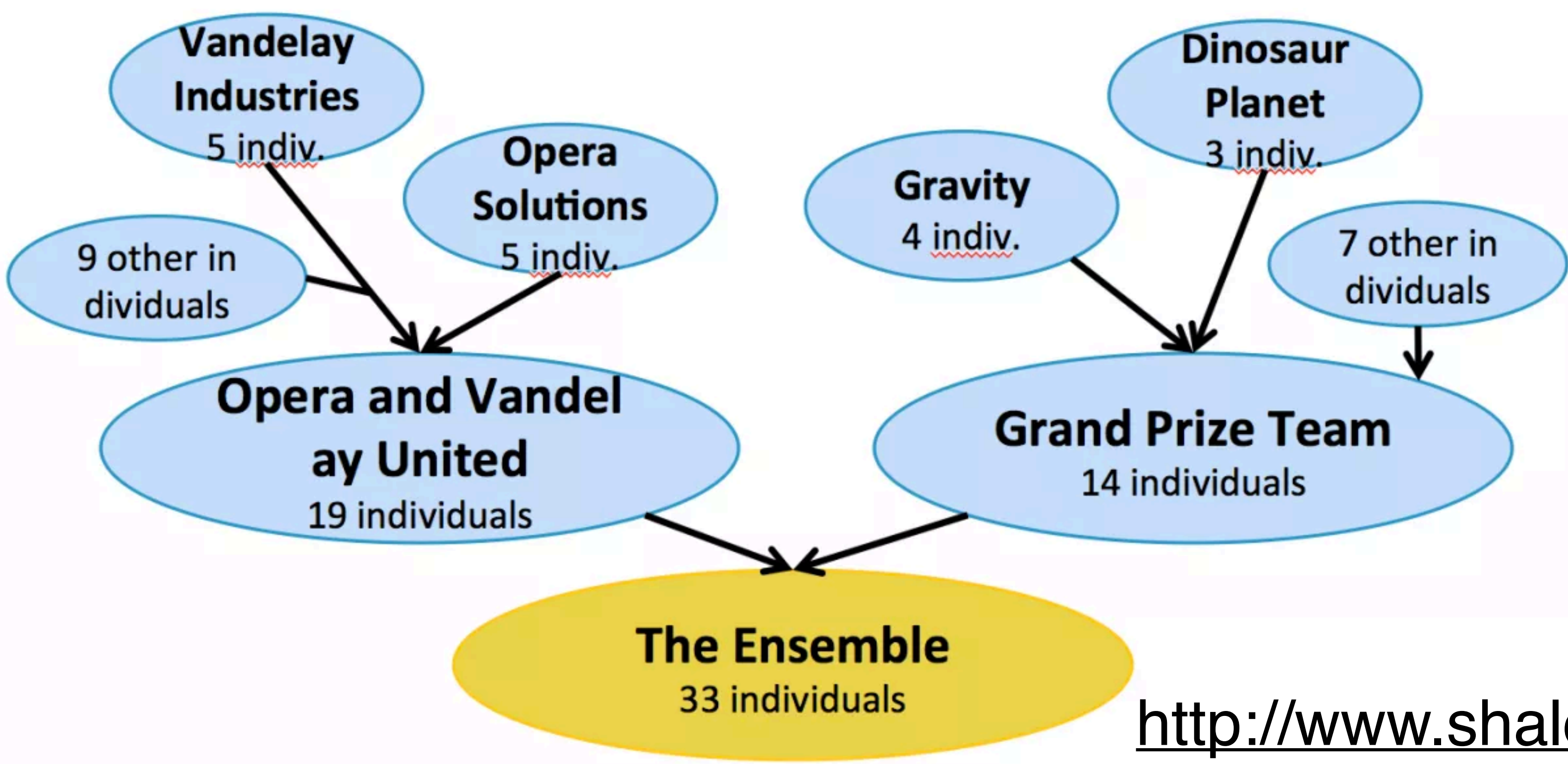
- Final Winner: "BellKor's Pragramatic Chaos" (Union of team BellKor, team Big Chaos and Pragmatic Theory) (improved by **10.06%**)

For achieving **10% improvement**,
it takes about **3 years!**

Rank	Team Name	Best Test Score	% Improvement	Best Submit Time
Grand Prize - RMSE = 0.8567 - Winning Team: BellKor's Pragmatic Chaos				
1	BellKor's Pragmatic Chaos	0.8567	10.06	2009-07-26 18:18:28
2	The Ensemble	0.8567	10.06	2009-07-26 18:38:22
3	Grand Prize Team	0.8582	9.90	2009-07-10 21:24:40
4	Opera Solutions and Vandelay United	0.8588	9.84	2009-07-10 01:12:31
5	Vandelay Industries !	0.8591	9.81	2009-07-10 00:32:20
6	PragmaticTheory	0.8594	9.77	2009-06-24 12:06:56
7	BellKor in BigChaos	0.8601	9.70	2009-05-13 08:14:09
8	Dace_	0.8612	9.59	2009-07-24 17:18:43
9	Feeds2	0.8622	9.48	2009-07-12 13:11:51
10	BigChaos	0.8623	9.47	2009-04-07 12:33:59
11	Opera Solutions	0.8623	9.47	2009-07-24 00:34:07
12	BellKor	0.8624	9.46	2009-07-26 17:19:11

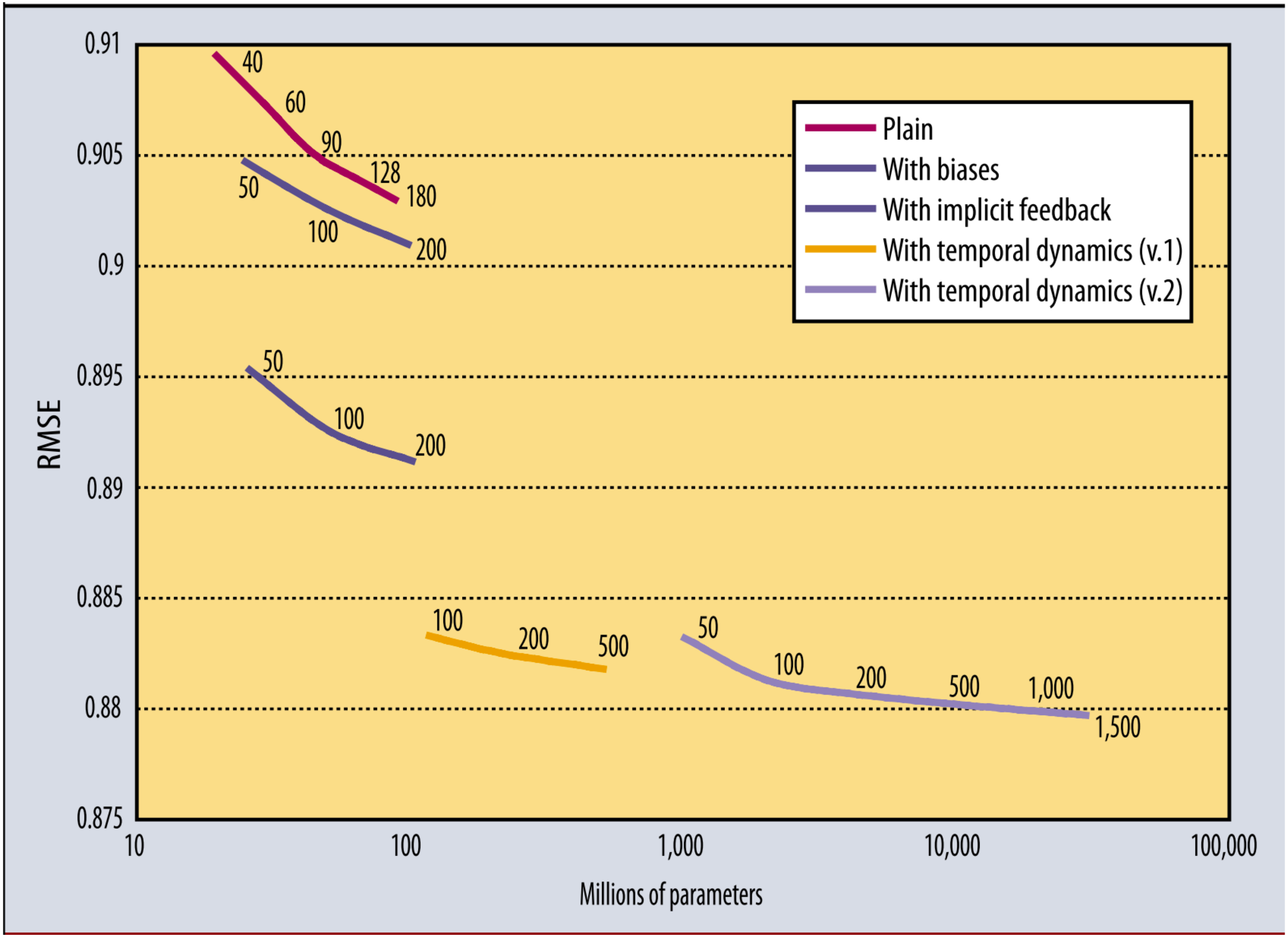
"That 20 minutes was worth a million dollar"





http://www.shalomeir.com/2014/12/netflix-prize-3/

Recap: Cinematch (0.9525)



Koren, Yehuda, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems." Computer 8 (2009): 30-37.

Evaluation of Recommender System

Offline evaluation

- RMSE / MAE, ...

- precision / recall / AUC, ...

- ranking metrics: NDCG, MAP, MRR, ...

Not directly related to real world user behaviors

Evaluation of Recommender System

Online evaluation

- CTR (Click-Through-Ratio)

- Cost per action, cost per click, ...

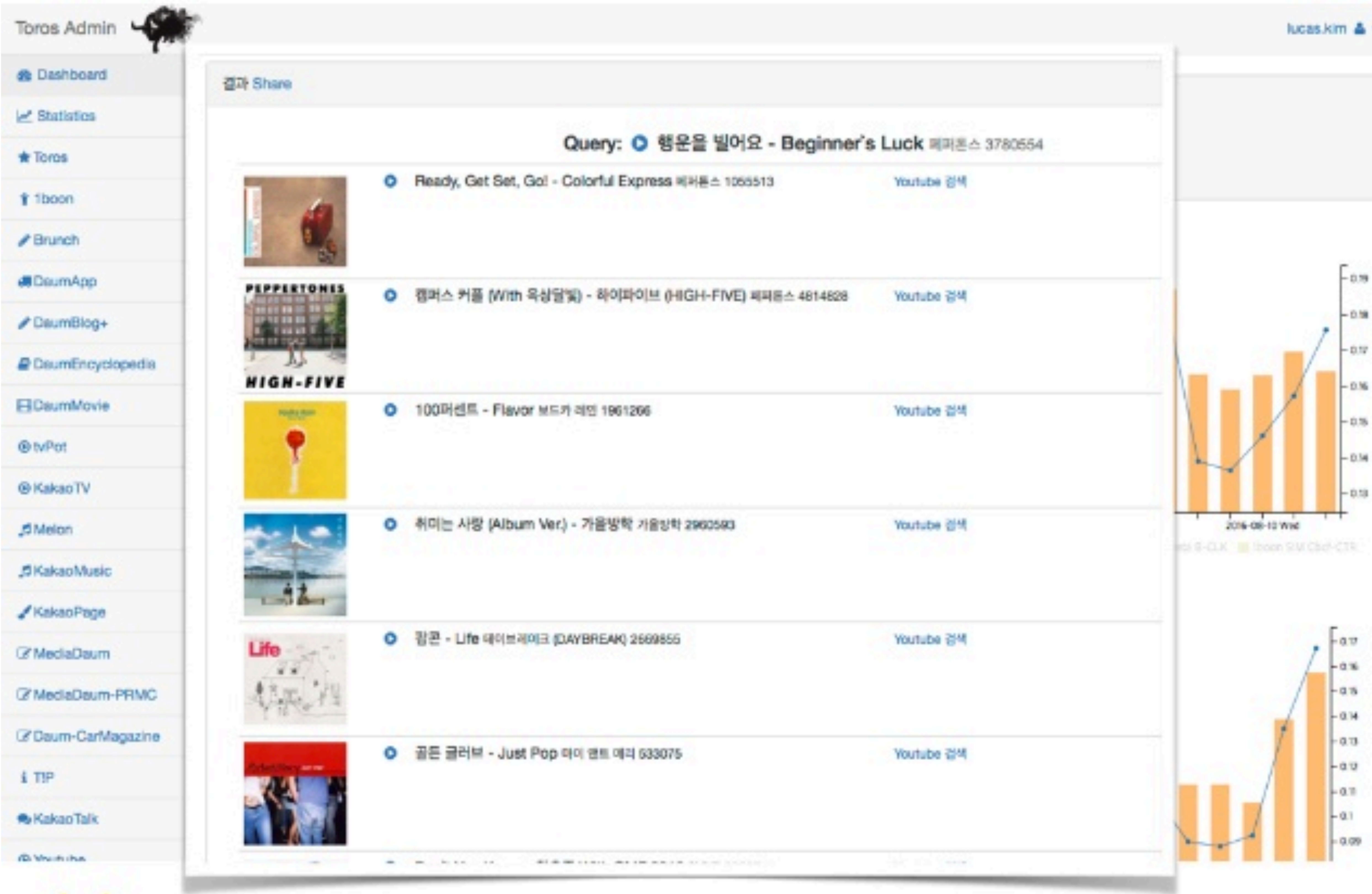
- PV / UV / ....

'Expensive' A/B test is required (also it is noisy)

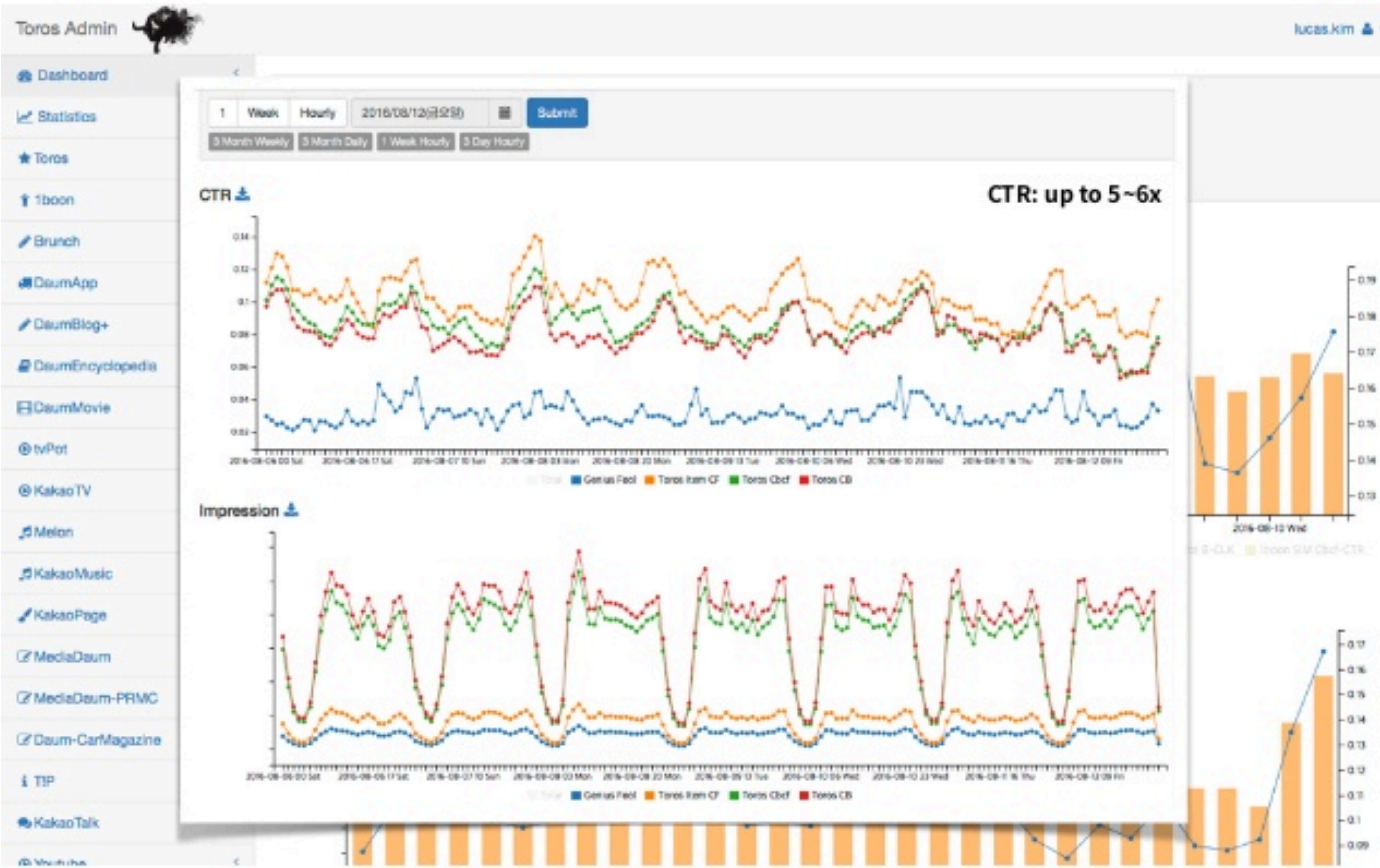
kakao

© Kakao Corp.

https://www.slideshare.net/kimkwangseop/toros-python-framework-for-recommender-system



https://www.slideshare.net/kimkwangseop/toros-python-framework-for-recommender-system



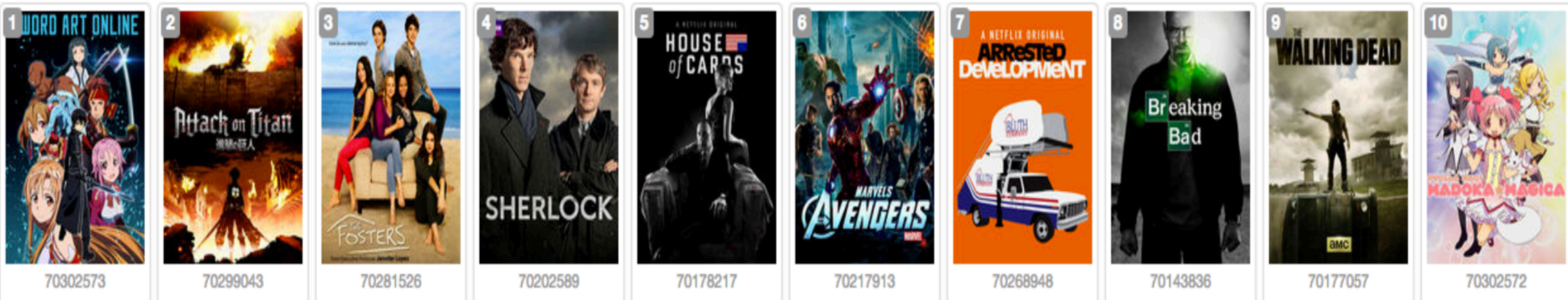
kakao

https://www.slideshare.net/kimkwangseop/toros-python-framework-for-recommender-system

More Novel Methods

Learning to Rank
Explore / Exploit
Session Based Recommendation
Deep Learning

It's the RANKING, stupid



4.7 4.6 4.5 4.5 4.5 4.5 4.5 4.5 4.5 4.5

Niche titles

High average ratings… by those who would watch it

NETFLIX

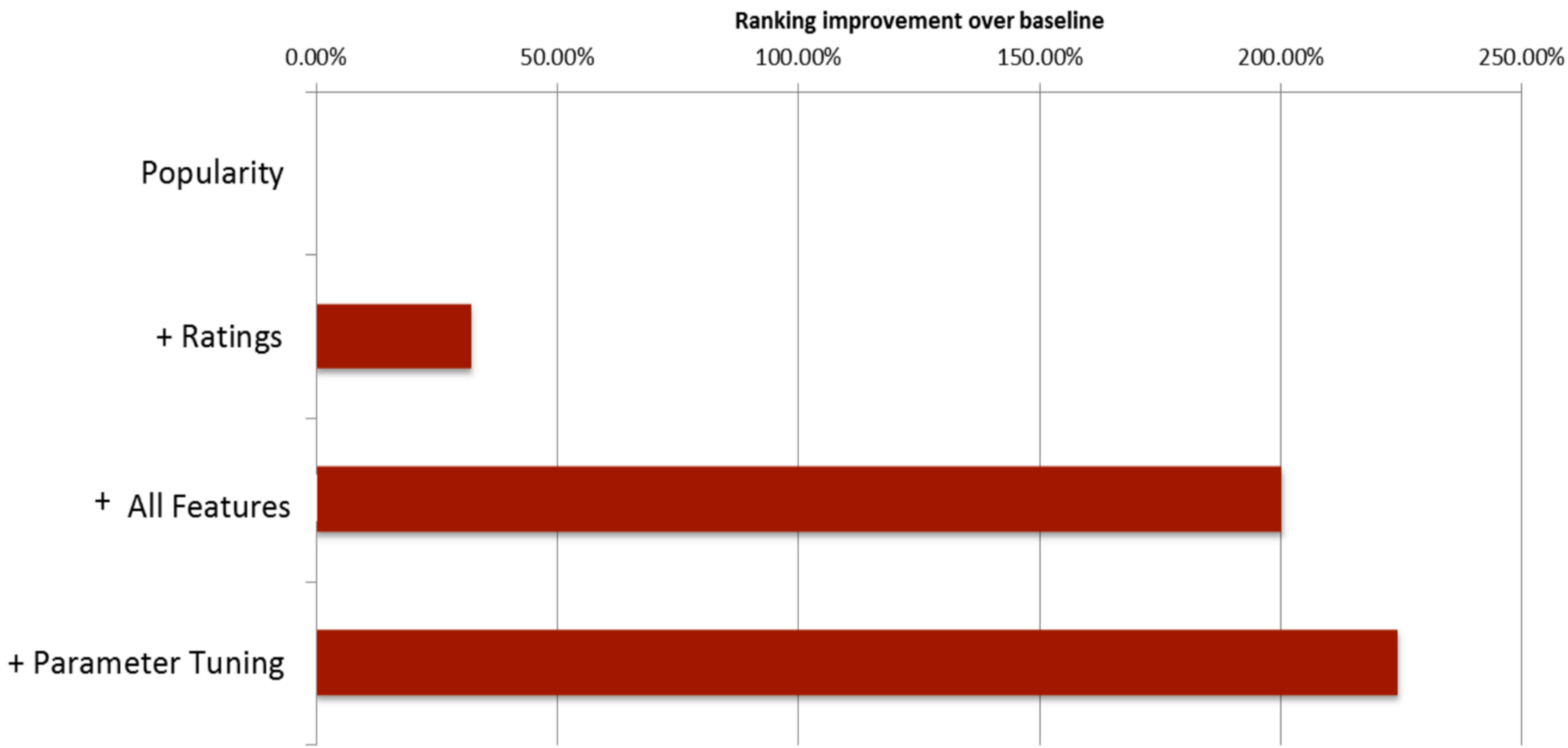
Xavier Amatriain – July 2014 – Recommender Systems

https://www.slideshare.net/xamat/recommender-systems-machine-learning-summer-school-2014-cmu

Lessons from RMSE

Rating (explicit feedback) is noisy than preference (implicit feedback) in many cases

Ranking by rating is not best ranking



NETFLIX

https://www.slideshare.net/xamat/recommender-systems-machine-learning-summer-school-2014-cmu

Learning to rank

Learning to rank is the machine learning problem to optimize 'ranking' from ranked data

Generally, training data is partially observed!

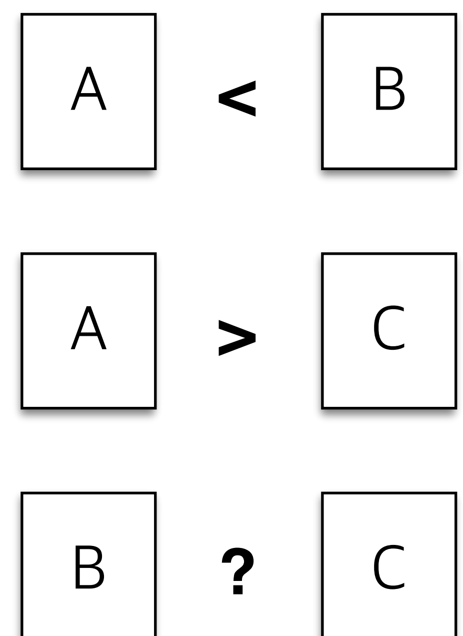
By constructing ranking model directly, one can reconstruct ranking of the unobserved data

Learning to rank (pairwise)

Data: pairwise observation (preference)

Example model: Bradley–Terry (BTL) model

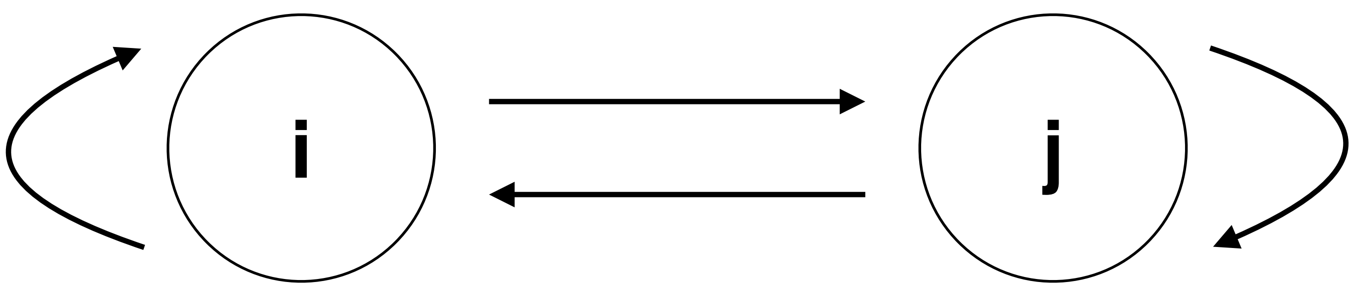
$$P(i > j) = \frac{p_i}{p_i + p_j}$$



Example algorithm for BTL model: Rank Centrality

Build graph from pairwise data as the following

j selected / (# i selected + # j selected)



$$P_{ij} = \begin{cases} \frac{1}{d_{\max}} A_{ij} & \text{if } i \neq j\,, \\ 1 - \frac{1}{d_{\max}} \sum_{k \neq i} A_{ik} & \text{if } i = j\,. \end{cases}$$

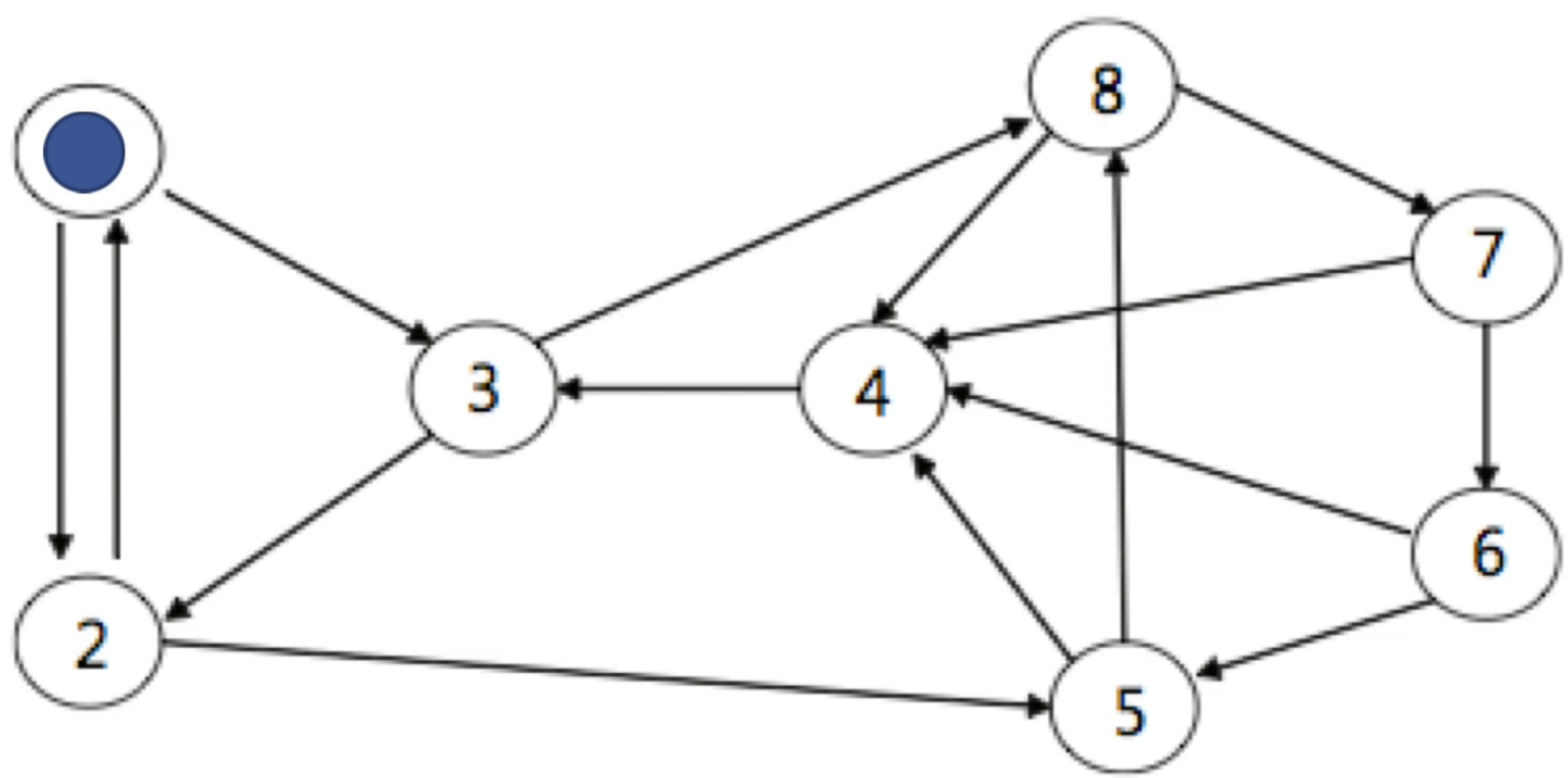
i selected / (# i selected + # j selected)

Negahban, Sahand, Sewoong Oh, and Devavrat Shah. "Rank centrality: Ranking from pairwise comparisons. Operations Research 65.1 (2016): 266-287.

Rank Centrality

Algorithm: random walk on the graph until converged (Markov chain, similar to PageRank)

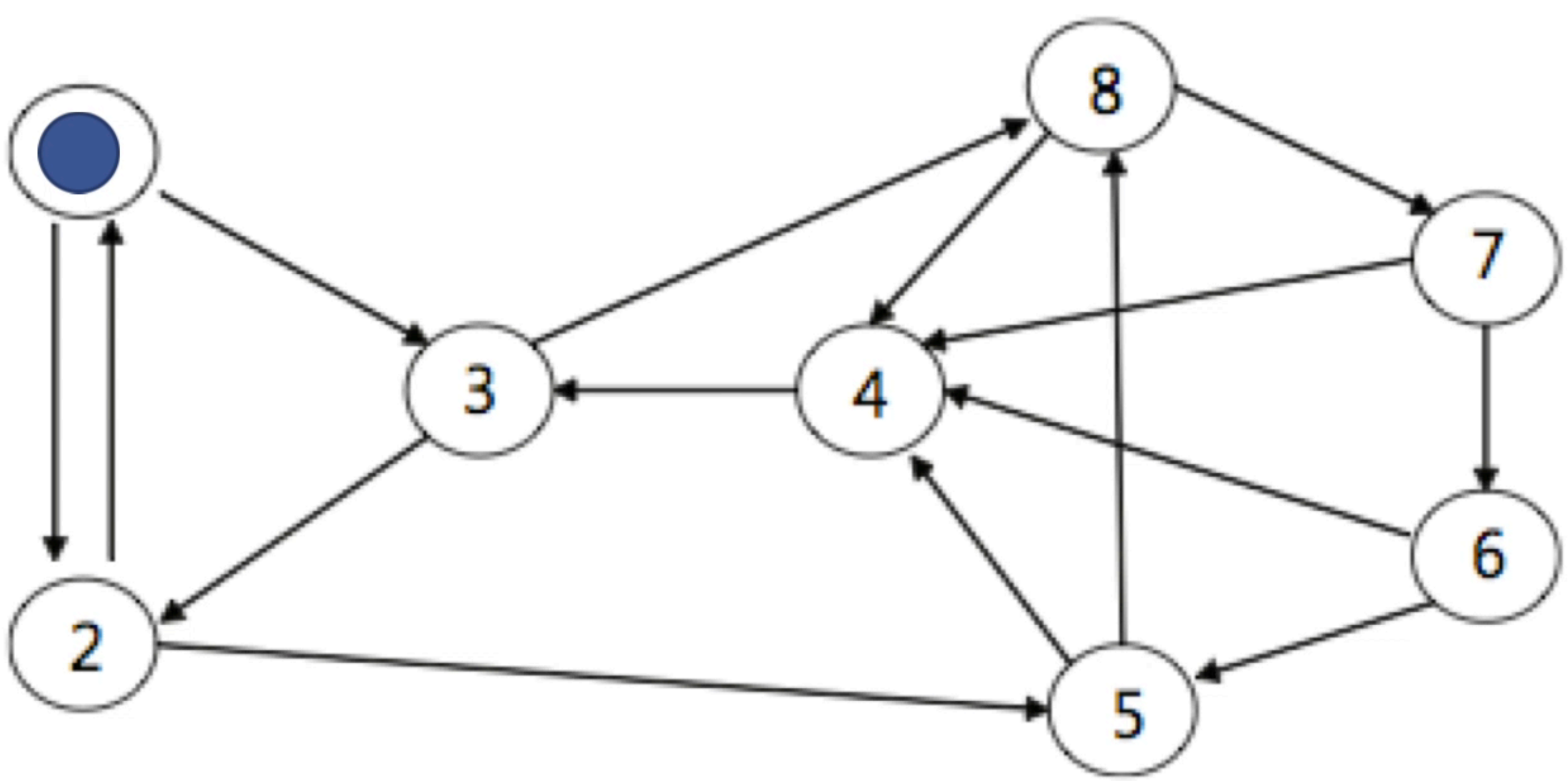
=> stationary distribution equals to rank!



Rank Centrality

Algorithm: random walk on the graph until converged (Markov chain, similar to PageRank)

=> stationary distribution equals to rank!



Example:
StarCraft + Rank Centrality

전체	리그	시즌	리그	날짜	구분	승자	승자종족	패자	패자종족
1	1	EVER 2003	정규	03.03.01 (	개인	베르트랑	Terran	홍진호	Zerg
3	3	EVER 2003	정규	03.03.01 (	개인	박신영	Zerg	장진수	Zerg
4	4	EVER 2003	정규	03.03.01 (	개인	김현진	Terran	김정민	Terran
6	6	EVER 2003	정규	03.03.01 (	개인	서지훈	Terran	성학승	Zerg
7	7	EVER 2003	정규	03.03.08 (	개인	박경락	Zerg	이현승	Terran
9	9	EVER 2003	정규	03.03.08 (	개인	최수범	Terran	나도현	Terran
10	10	EVER 2003	정규	03.03.08 (	개인	김성제	Protoss	윤정민	Terran
12	12	EVER 2003	정규	03.03.08 (	개인	임요환	Terran	전태규	Protoss
13	13	EVER 2003	정규	03.03.15 (	개인	송병석	Protoss	이창훈	Zerg
15	15	EVER 2003	정규	03.03.15 (	개인	임요환	Terran	이윤열	Terran
16	16	EVER 2003	정규	03.03.15 (	개인	조정현	Terran	김현진	Terran
18	18	EVER 2003	정규	03.03.15 (	개인	베르트랑	Terran	성학승	Zerg
19	19	EVER 2003	정규	03.03.22 (	개인	주진철	Zerg	최수범	Terran
21	21	EVER 2003	정규	03.03.22 (	개인	이현승	Zerg	전태규	Protoss
22	22	EVER 2003	정규	03.03.22 (	개인	이재훈	Protoss	박정석	Protoss
24	24	EVER 2003	정규	03.03.22 (	개인	서지훈	Terran	박경락	Zerg
25	25	EVER 2003	정규	03.03.29 (	개인	송병석	Protoss	조병호	Protoss
27	27	EVER 2003	정규	03.03.29 (	개인	이윤열	Terran	주진철	Zerg
28	28	EVER 2003	정규	03.03.29 (	개인	박성훈	Protoss	서지훈	Terran
30	30	EVER 2003	정규	03.03.29 (	개인	박동욱	Protoss	이재훈	Protoss
31	31	EVER 2003	정규	03.04.05 (	개인	박정석	Protoss	백영민	Protoss
33	33	EVER 2003	정규	03.04.05 (	개인	성학승	Zerg	박경락	Zerg
34	34	EVER 2003	정규	03.04.05 (	개인	최연성	Terran	베르트랑	Terran
36	36	EVER 2003	정규	03.04.05 (	개인	기욤	Protoss	김성제	Protoss
37	37	EVER 2003	정규	03.04.12 (	개인	홍진호	Zerg	최인규	Terran
39	39	EVER 2003	정규	03.04.12 (	개인	이윤열	Terran	서지훈	Terran
40	40	EVER 2003	정규	03.04.12 (	개인	주진철	Zerg	장진수	Zerg
42	42	EVER 2003	정규	03.04.12 (	개인	베르트랑	Terran	윤정민	Terran
43	43	EVER 2003	정규	03.04.19 (	개인	임요환	Terran	박경락	Zerg

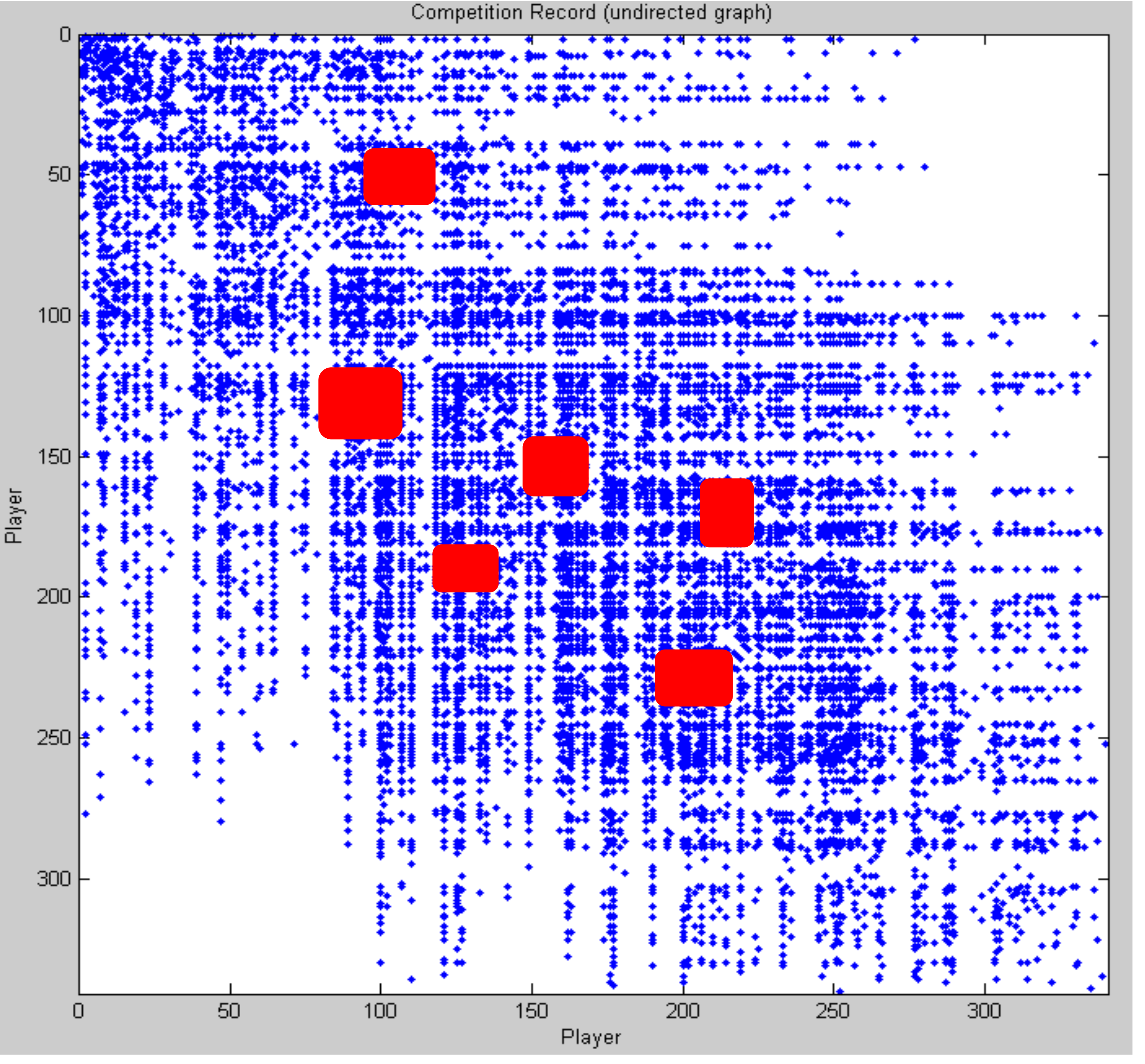
베르트랑	홍진호
박신영	장진수
김현진	김정민
서지훈	성학승
박경락	이현승
최수범	나도현
김성제	윤정민
임요환	전태규
송병석	이창훈
임요환	이윤열
조정현	김현진
베르트랑	성학승
주진철	최수범
이현승	전태규
이재훈	박정석
서지훈	박경락
송병석	조병호
이윤열	주진철
박성훈	서지훈
박동욱	이재훈
박정석	백영민
성학승	박경락
최연성	베르트랑
기욤	김성제
홍진호	최인규
이윤열	서지훈

0	1	1
2	3	1
4	5	1
6	7	1
8	9	1
10	11	1
12	13	1
14	15	1
16	17	1
14	18	1
19	4	1
0	7	1
20	10	1
9	15	1
21	22	1
6	8	1
16	23	1
18	20	1
24	6	1
25	21	1
22	26	1
7	8	1
27	0	1
28	12	1
1	29	1
18	6	1

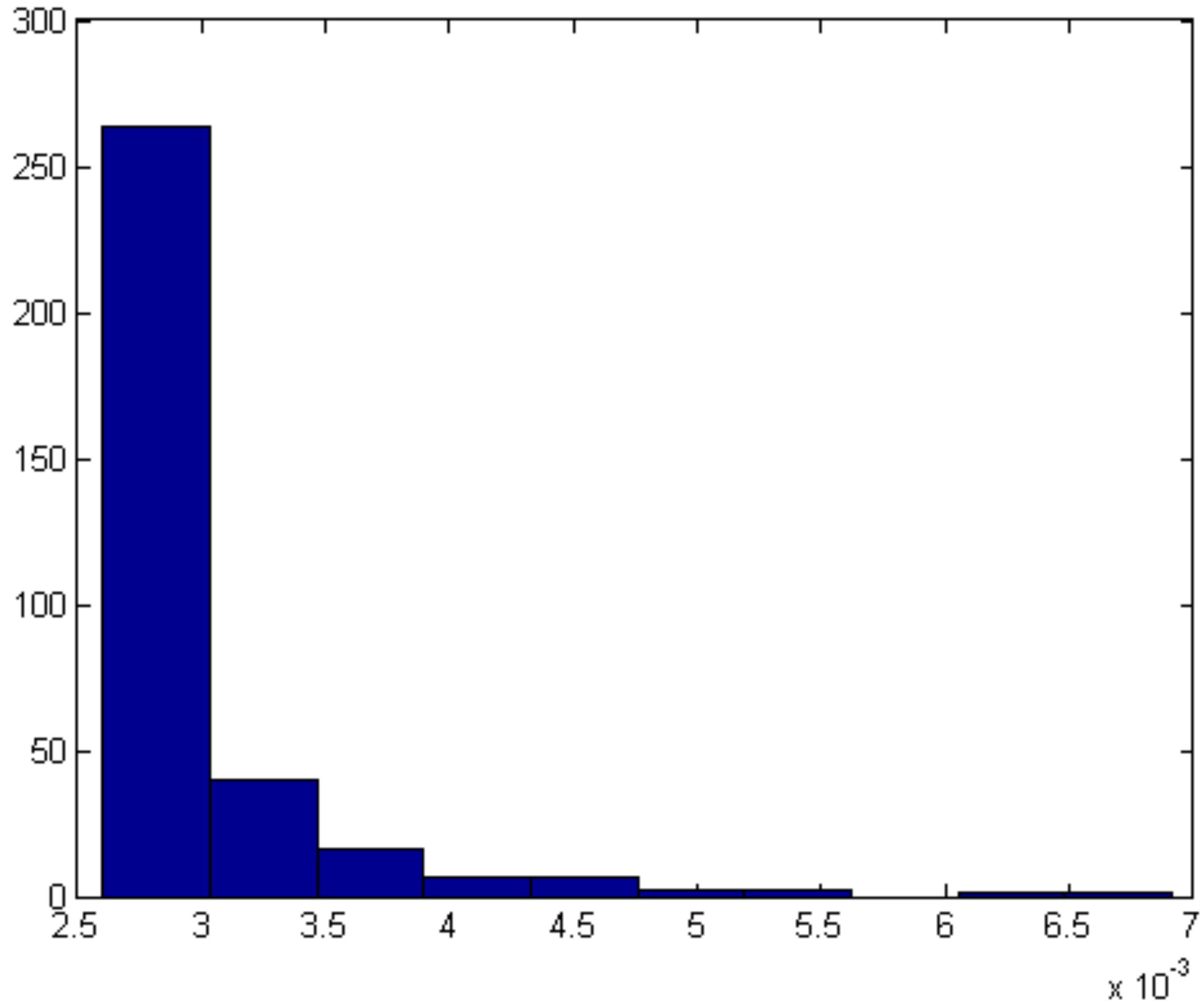
Total 340 player, 9100 match

for 11 years

Edge list of match-up record



Competition record(undirected)



Top 10
Young-ho Lee
Jea-Dong Lee
Teak-Yong Kim
Byoung-Goo Song
Myeong-Hoon Jeong
Bo-Sung Yeom
Yong-Tae Yoon
Jae-Ho Lee
Sang-Moon Shin
Myeoung-Woon Kim

Metric for Ranking

NDCG (Normalized Discounted Cumulative Gain)

Mean Average Precision (MAP)

Mean Reciprocal Rank (MRR)

....

The measurements are NOT differentiable

Models for learning to rank

Pointwise Models

estimate ranking by computing score for each example then sorting by the score (based on regression or classification)

data: (i, j, relevance score)

- Logistic regression, PRank, MCRank ...

Models for learning to rank

Pairwise Models

constructing pairwise rank model by minimizing inversions in the given pairs (the problem is transformed into a binary classification)

data: (i, j, preference)

- RankNet, SVMRank, AdaRank, LambdaMART...

Models for learning to rank

Listwise Models

Directly (non-differential) optimizing rank metric such as NDCG, MAP

Could be solved by genetic algorithm, simulated annealing, relaxation ...

Learning to rank

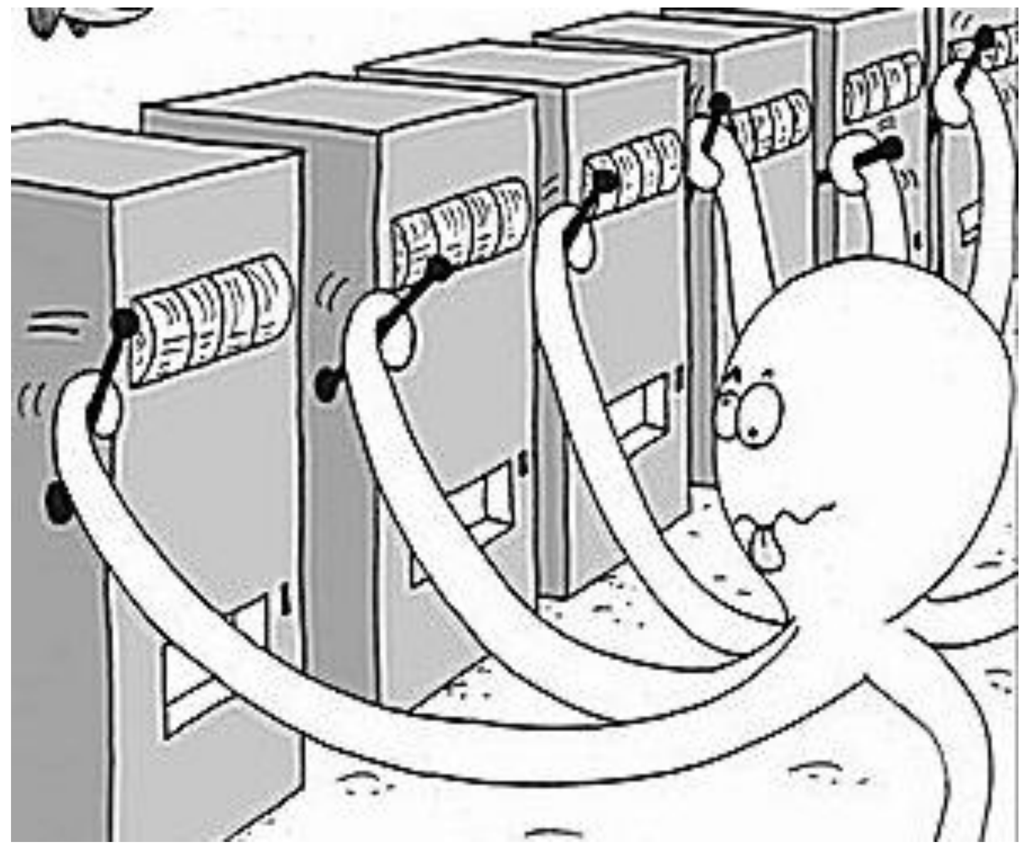
Directly optimize ranking in **offline**

Even though learning to rank optimize ranking, it still does not optimize **'profit' (CTR)** directly

More Novel Methods

Learning to Rank
Explore / Exploit
Session Based Recommendation
Deep Learning

Multi-armed Bandit Problem



- K arms with unknown reward distributions
- Maximize reward (or minimize regret) over time T
- Each time, a policy select a single arm and receive reward

Example: Bernoulli bandit

	A	B	C
Reward: 0	50%	30%	80%
Reward: 1	50%	70%	20%

Select: A A B B B B B C A B B B ..

Reward: 1 0 1 0 0 1 1 0 1 1 1 1 ..

Exploration vs. Exploitation

Exploration: Since we have no information of each arm, we have to ‘explore’ unknown arms repeatedly

Exploitation: Play the best arm (empirically) to get large reward

Trade off between **exploration** and **exploitation**

Algorithm (Epsilon greedy)

with probability 1 - e, Play best arm

with probability e, Play a random arm

After enough large exploration, epsilon greedy play randomly with probability epsilon

Algorithm (UCB)

UCB (Upper Confidence Bound)

$$i = \arg\max_i \mu_i + P_i$$

(UCB 1) $$i = \arg\max_i \bar{x}_i + \sqrt{\frac{2\ln t}{n_i}}.$$

Algorithm (Thompson Sampling)

Algorithm 2 Thompson sampling for the Bernoulli bandit

Require: α, β prior parameters of a Beta distribution
  $S_i = 0, F_i = 0, \quad \forall i$. {Success and failure counters}
  for $t = 1, \ldots, T$ **do**
    for $i = 1, \ldots, K$ **do**
      Draw θ_i according to Beta$(S_i + \alpha, F_i + \beta)$.
    end for
    Draw arm $\hat{\imath} = \arg\max_i \theta_i$ and observe reward r
    if $r = 1$ **then**
      $S_{\hat{\imath}} = S_{\hat{\imath}} + 1$
    else
      $F_{\hat{\imath}} = F_{\hat{\imath}} + 1$
    end if
  end for

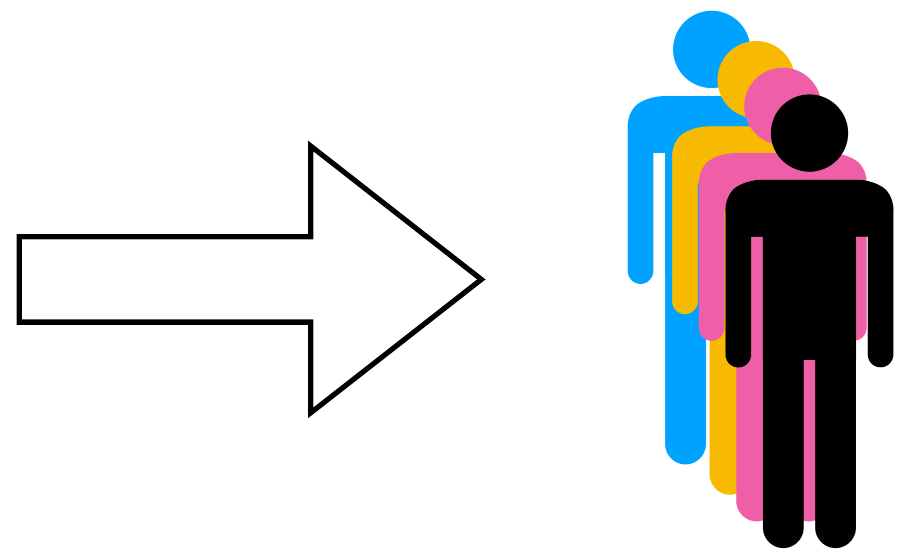
Chapelle, Olivier, and Lihong Li. "An empirical evaluation of thompson sampling." Advances in neural information processing systems. 2011.

Bandit in Recommender System

Oracle CTR

5%

8%



2%

1%

Question:
Which item should we serve to users?
How can we find such items 'online'?

Bandit in Recommender System

Think each **item** as the **arm** of the bandit

Than, **reward** is **click** and
reward distribution is equal to **CTR**

Now, we can find the item with best CTR while dealing with explore / exploit trade-off!

Limitation of MAB in real world

Stochastic bandit assumes the following

- Each time, bandit only choose single arm
  (i.e., observe only single item)

- Immediately feedback (most of cases, users don't click item directly)

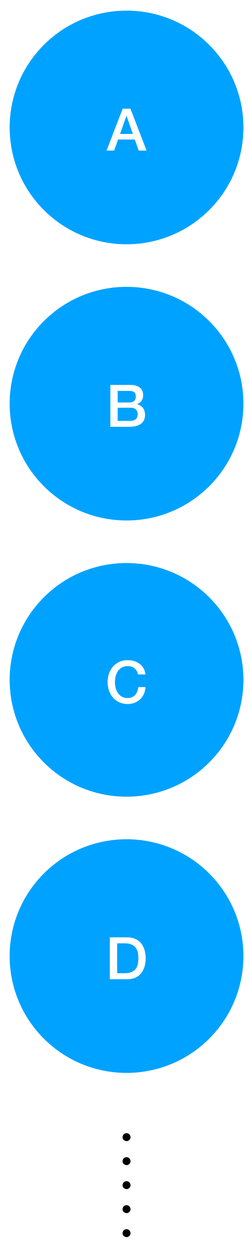
- Number of arms are finite and fixed
  (in real world, there is a 'life-cycle' of item
  + new item appears frequently)

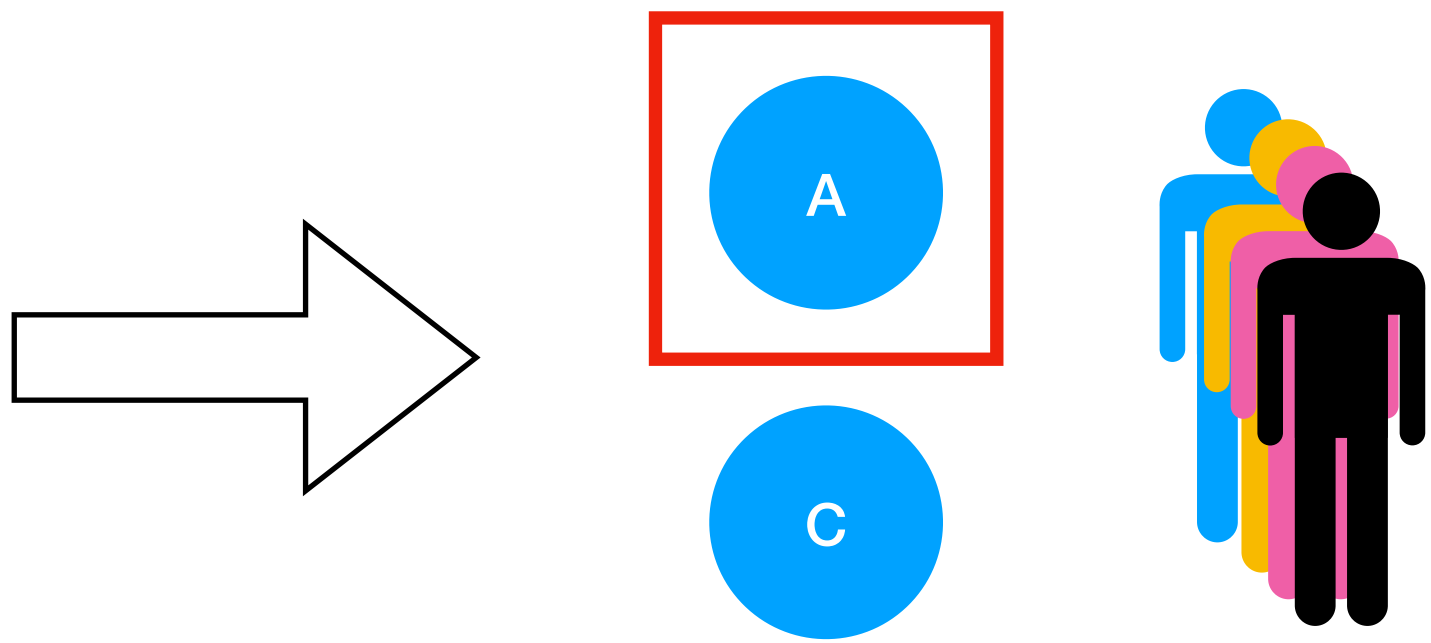
Limitation of MAB in real world

Stochastic bandit assumes the following

- (cont) arm is stationary (there is 'positional bias' and CTR of each item is affected by co-recommend items)

Modified MAB for RecSys





Multi-play Bandit!
(sorted by expected CTR, displayed items)
+ discounted penalty for lower rank item

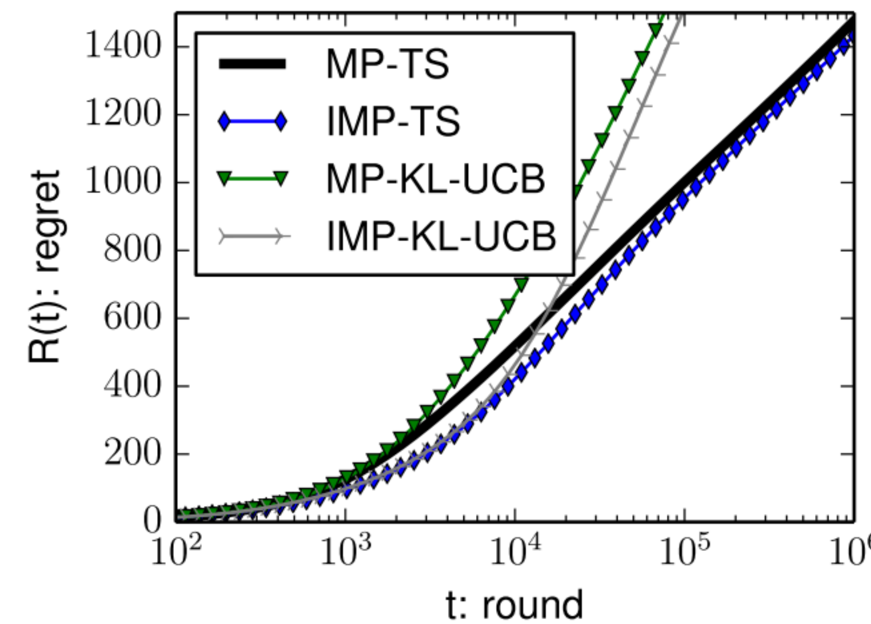
Arms: candidates for recommendation (by CF, CB, learning to rank, ...)

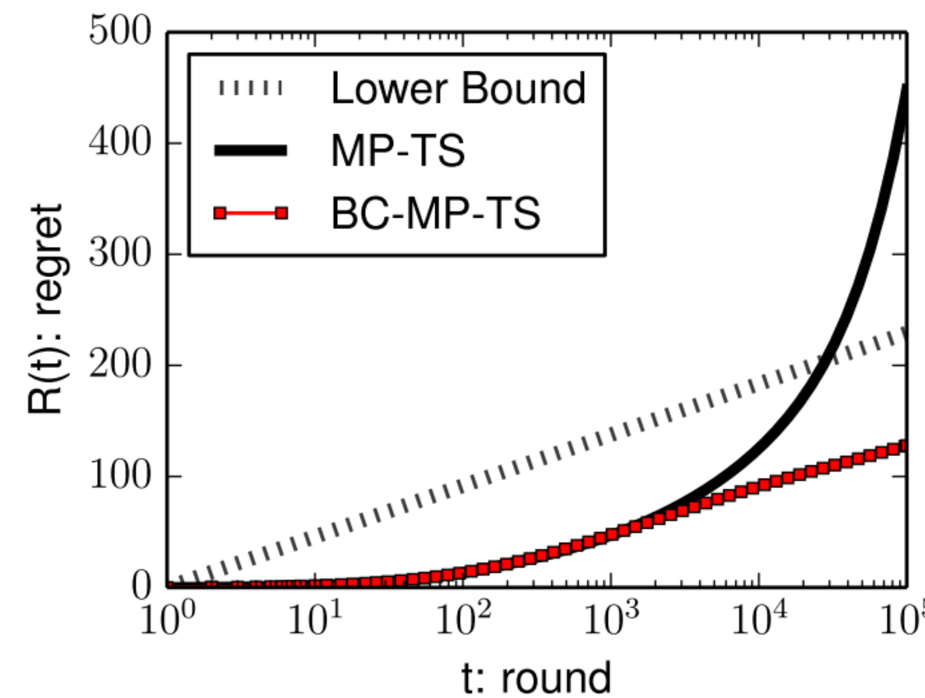
IMP-TS (improved MP-TS)

Algorithm 1 Multiple-play Thompson sampling (MP-TS) for binary rewards

Input: # of arms K, # of selection L
for $i = 1, 2, \ldots, K$ **do**
  $A_i, B_i = 1, 1$
end for
$t \leftarrow 1$.
for $t = 1, 2, \ldots, T$ **do**
  for $i = 1, 2, \ldots, K$ **do**
    $\underline{\theta_i(t) \sim \mathrm{Beta}(A_i, B_i)}$
  end for
  $I(t) =$ top-L arms ranked by $\theta_i(t)$.
  for $i \in I(t)$ **do**
    if $X_i(t) = 1$ **then**
      $A_i \leftarrow A_i + 1$
    else
      $B_i \leftarrow B_i + 1$
    end if
  end for
end for

For K-1 items,
instead of sampling from Beta,
sorted by empirical mean





Komiyama, Junpei, Junya Honda, and Hiroshi Nakagawa. "Optimal regret analysis of thompson sampling in stochastic multi-armed bandit problem with multiple plays." ICML 2015

Personalization using MAB

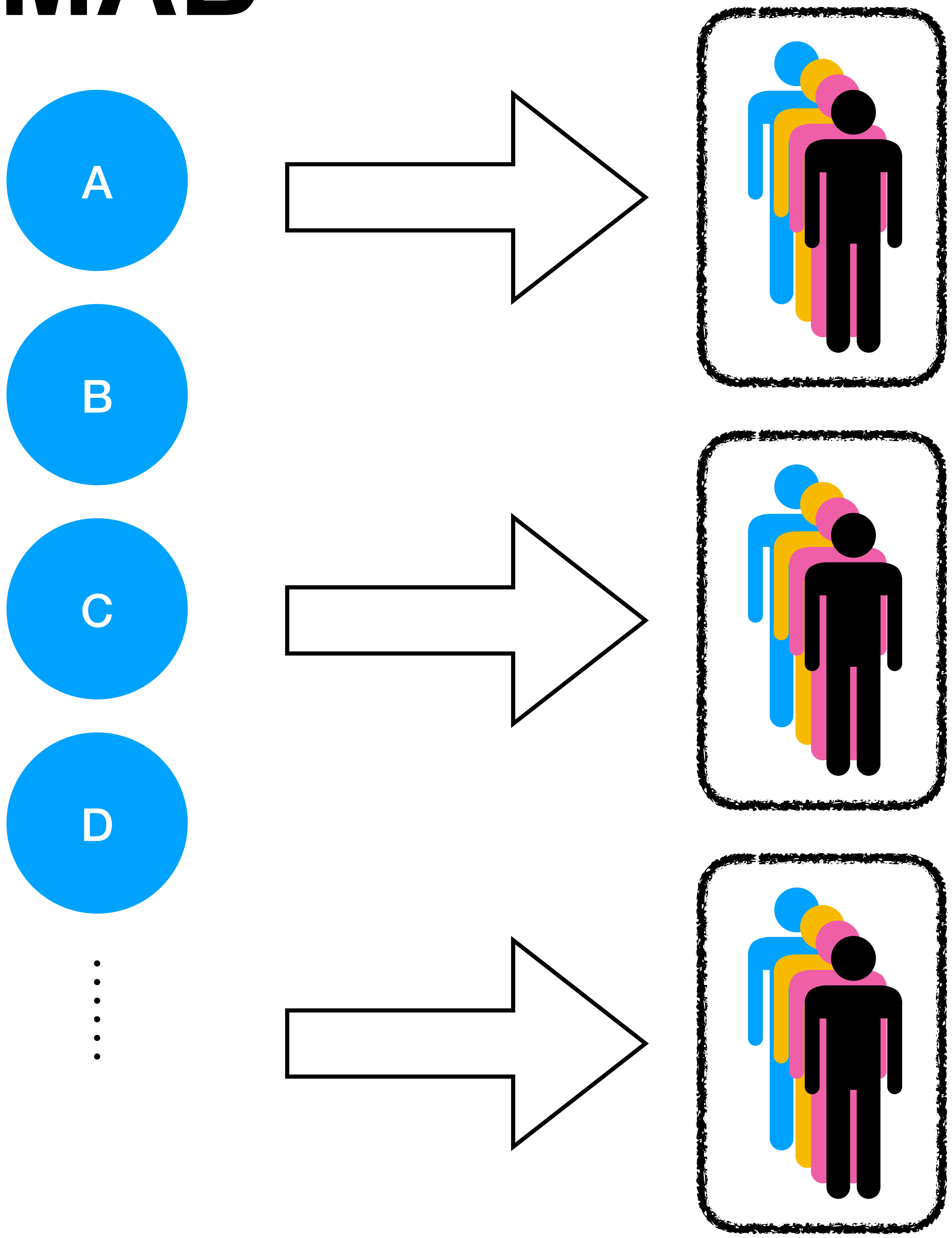
Note that MAB requires MANY experiments until converge to near optimal

Theorem 1. (Regret upper bound of MP-TS) *For any sufficiently small $\epsilon_1 > 0, \epsilon_2 > 0$, the regret of MP-TS is upper-bounded as*

$$\mathbb{E}[\mathrm{Reg}(T)] \leq \sum_{i \in [K] \setminus [L]} \left(\frac{(1+\epsilon_1)\Delta_{i,L} \log T}{d(\mu_i, \mu_L)} \right) + C_a(\epsilon_1, \mu_1, \mu_2, \ldots, \mu_K) + C_b(T, \epsilon_2, \mu_1, \mu_2, \ldots, \mu_K),$$

One bandit for one person = random reco.

Semi-personalization using MAB



One bandit for one user cluster
= better than random

Questions)

- **How to clustering?**
- **Still not fully 'personalized'**

Contextual Bandit

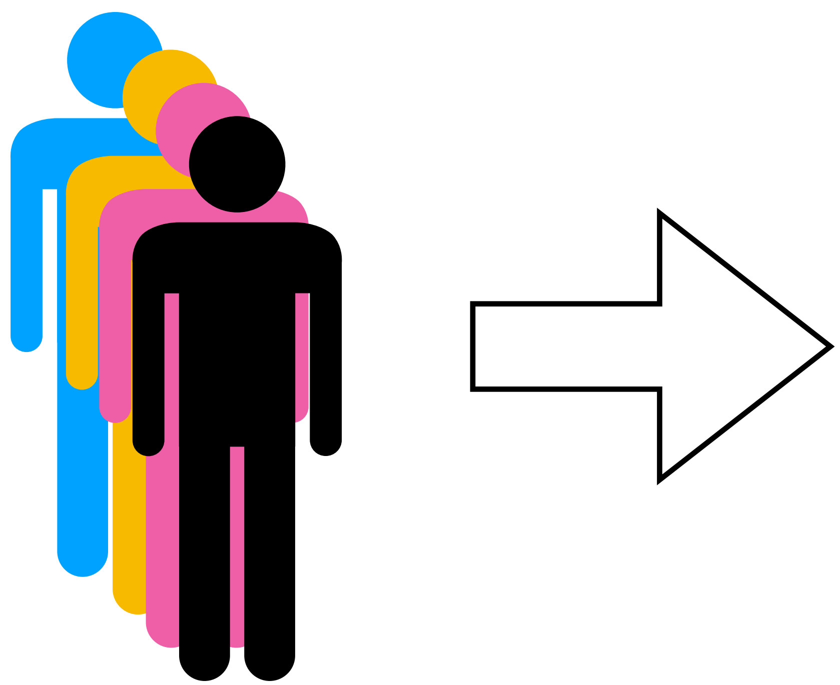


Figure 1: A snapshot of the "Featured" tab in the Today Module on Yahoo! Front Page. By default, the article at F1 position is highlighted at the story position.

For each time, contextual vector $x_{t,a}$ is observed (related to both user and item)

Li, Lihong, et al. "A contextual-bandit approach to personalized news article recommendation." Proceedings of the 19th international conference on World wide web. ACM, 2010.

Contextual Bandit (LinUCB)

$$\mathbb{E}\left[r_{t,a} | x_{t,a}\right] = x_{t,a}^{\top} \theta_a^*$$

- at time t, user u_t observes arm a with context vector $x_{t,a}$
- context vector $x_{t,a}$ summarizes information of both the user u_t and arm a
- θ_a is a learning parameter for each arms
- Note: if x is constant, it is exactly same as stochastic bandit

Contextual Bandit (LinUCB)

How to UCB?

- Exploit only: select maximum expectation

$$\mathbb{E}\left[r_{t,a} | x_{t,a}\right] = x_{t,a}^{\top} \theta_a^*$$

- UCB: consider variance

$$a_t \stackrel{\text{def}}{=} \arg\max_{a \in \mathcal{A}_t} \left(\mathbf{x}_{t,a}^{\top} \hat{\boldsymbol{\theta}}_a + \alpha \sqrt{\mathbf{x}_{t,a}^{\top} \mathbf{A}_a^{-1} \mathbf{x}_{t,a}} \right) \quad \mathbf{A}_a \stackrel{\text{def}}{=} \mathbf{D}_a^{\top} \mathbf{D}_a + \mathbf{I}_d.$$

Theorem 4.1 *Suppose the rewards* $r_{t,a}$ *are independent random variables with means* $\mathrm{E}[r_{t,a}] = x_{t,a}^{\top}\theta^*$, *let* $\epsilon = \sqrt{\frac{1}{2}\ln\frac{2TK}{\delta}}$ *and* $A_t = D_t^{\top} D_t + \mathrm{I}_d$ *then with probability* $1 - \delta/T$, *we have*

$$|x_{t,a}^{\top}\hat{\theta}_t - x_{t,a}^{\top}\theta^*| \leq (\epsilon + 1)\sqrt{x_{t,a}^{\top} A_t^{-1} x_{t,a}}$$

Contextual Bandit (LinUCB)

Problem: how to choose context vector x?

Recall: a context vector x summaries both arm a (article) and user u

Short Answer: Run user clustering using article-related feature make context vector (6-dimensional vector), i.e., we have to run clustering as number of articles

LinUCB: Context (details)

- Article feature: 83D categorical feature
  - URL categories: tens of classes
  - editor categories: tens of topics tagged by human
- User feature 1193D categorical feature
  - Demographic categories: 2 gender * 5 age band
  - Geographic features: about 200 locations
  - Behavioral categories: about 1000 binary categories that summarize the user consumption history within Yahoo

LinUCB: Context (details)

- To dimension reduction, project user feature onto article categories and then cluster users
- First, fit bilinear logistic regression to CTR using user/article feature $\phi_u^\top W \phi_a$
- Project user feature to article feature by $\psi_u = \phi_u^\top W$
- Run k-means onto ψ_u with k=5
- Final 6D user feature x:
  cluster indicator 5D + constant 1

More readings

- **[Survey]** Burtini, Giuseppe, Jason Loeppky, and Ramon Lawrence. "A survey of online experiment design with the stochastic multi-armed bandit." arXiv preprint arXiv:1510.00757 (2015).

- **[Empirical Study]** Chapelle, Olivier, and Lihong Li. "An empirical evaluation of thompson sampling." Advances in neural information processing systems. 2011.

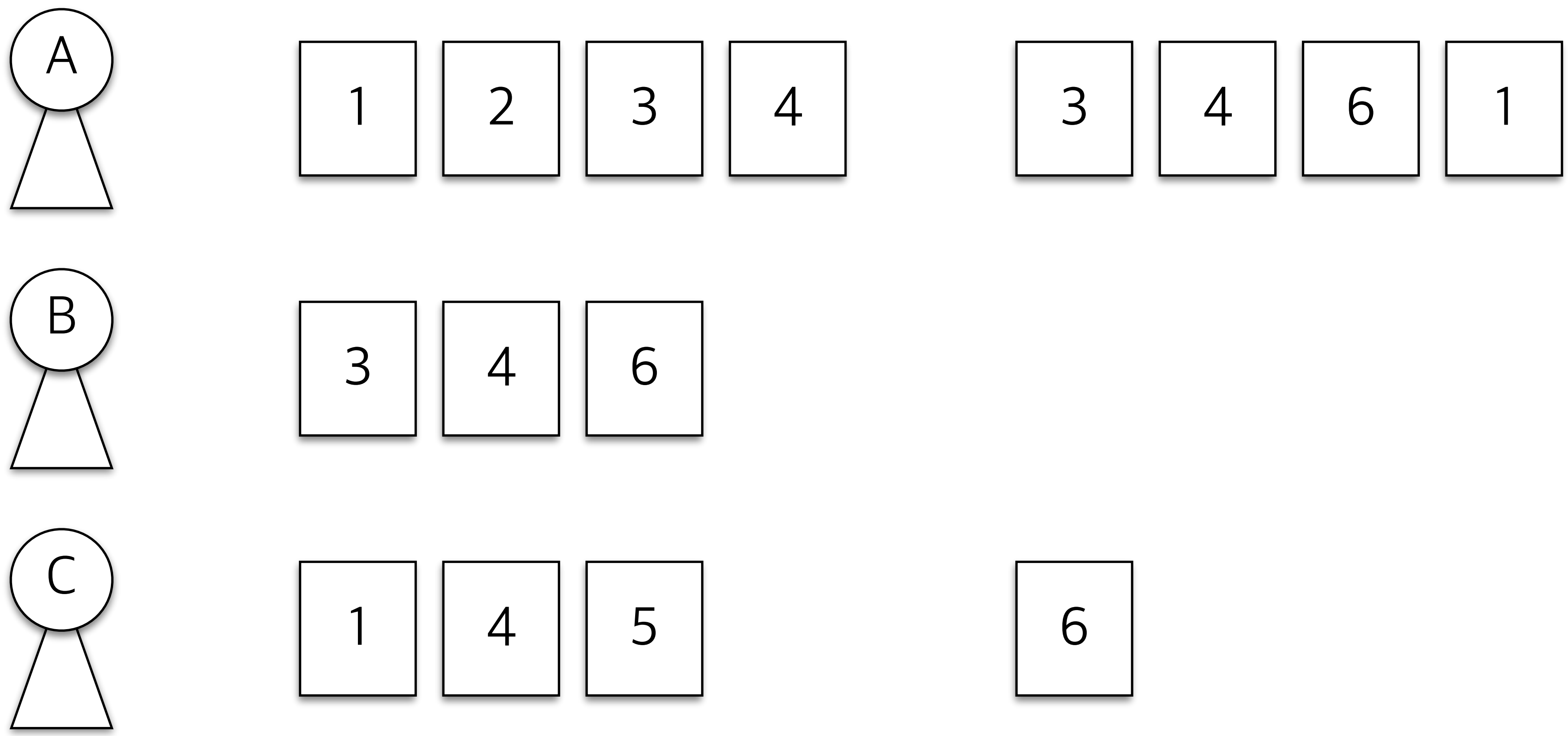
- **[Advanced contextual bandit for recommendation]** Vanchinathan, Hastagiri P., et al. "Explore-exploit in top-n recommender systems via gaussian processes." Proceedings of the 8th ACM Conference on Recommender systems. ACM, 2014.

- **[Book]** Agarwal, Deepak K., and Bee Chung-Chen. "Statistical methods for recommender systems." (2016).

More Novel Methods

Learning to Rank
Explore / Exploit
Session Based Recommendation
Deep Learning

Define 'Session'



Why session based?

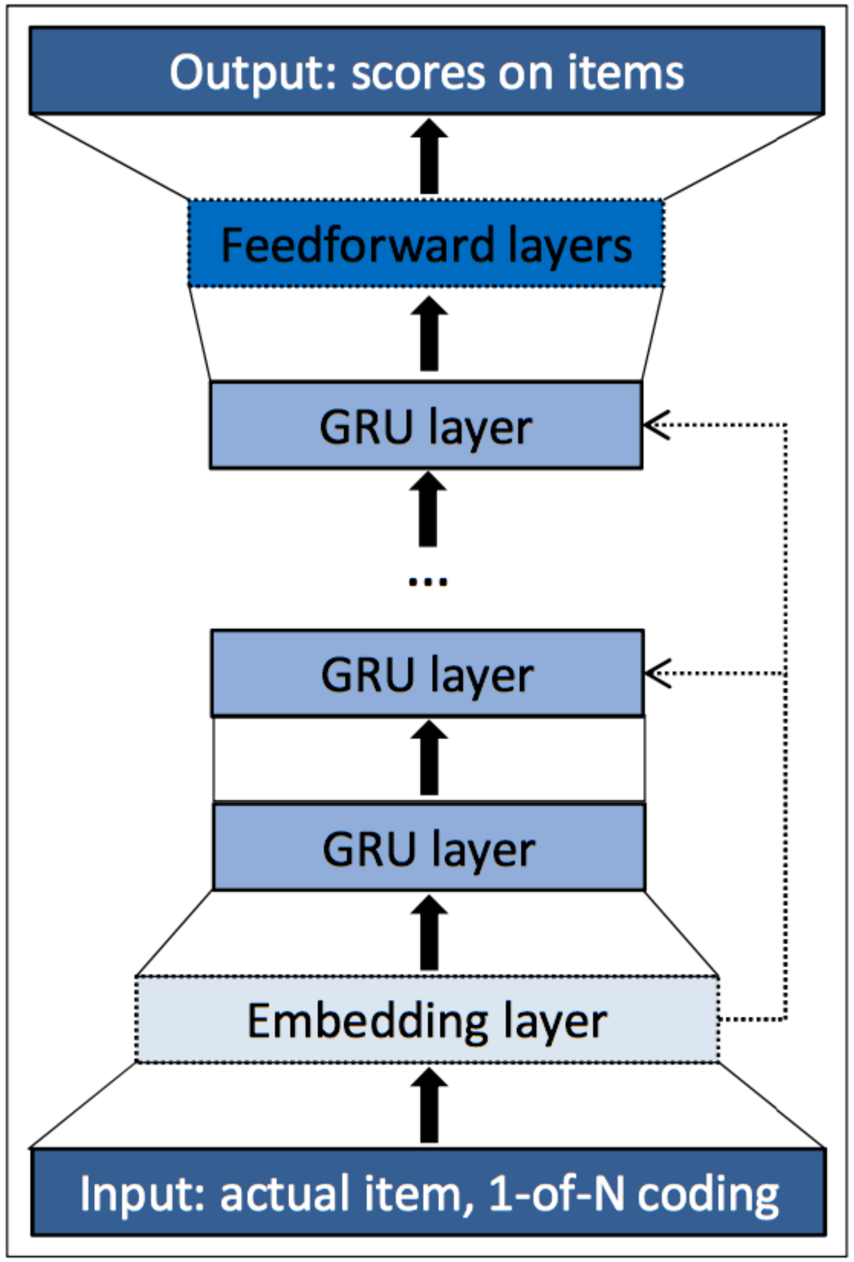
Subsequent sessions of the same user should be handled independently

Favorites of users could be changed by time

Maybe session contains "context" information

(Practically) session data could be handled in 'incremental' way while matrix data couldn't

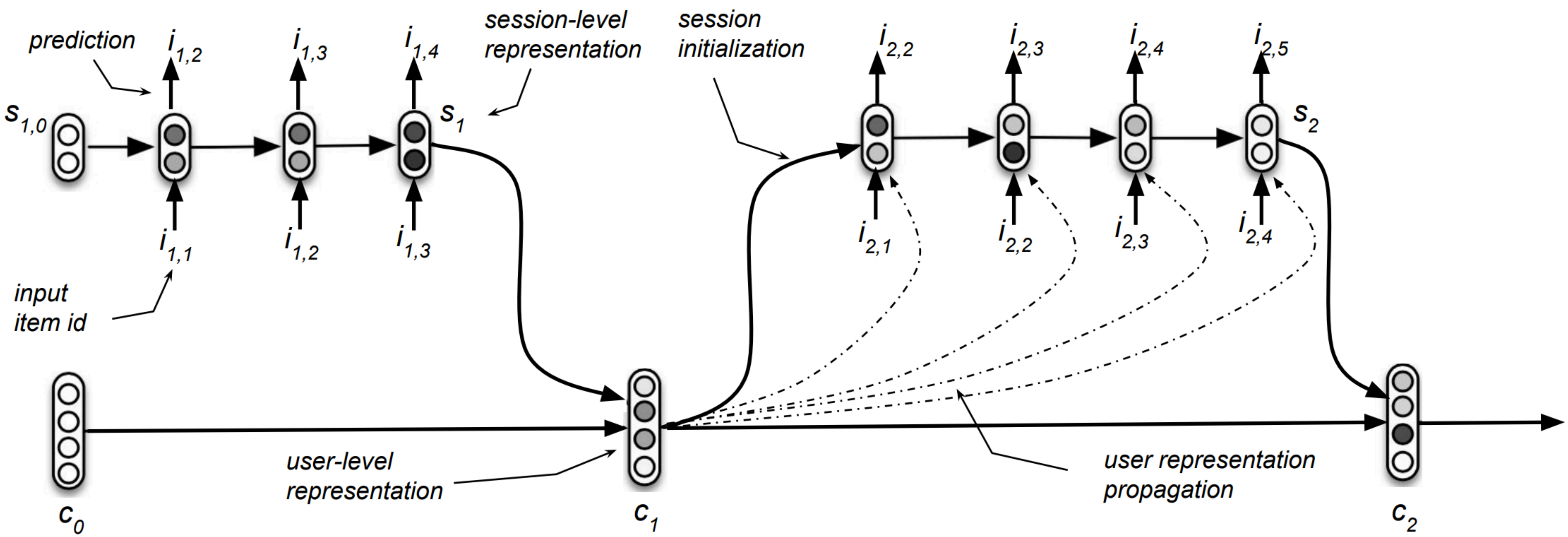
Session-based recommendations with recurrent neural networks



in a nutshell: Next-item prediction

Hidasi, Balázs, et al. "Session-based recommendations with recurrent neural networks." ICLR 2016

Personalizing Session-based Recommendations with Hierarchical Recurrent Neural Networks



Quadrana, Massimo, et al. "Personalizing session-based recommendations with hierarchical recurrent neura networks." Proceedings of the Eleventh ACM Conference on Recommender Systems. ACM, 2017.

Neural Attentive Session-based Recommendation

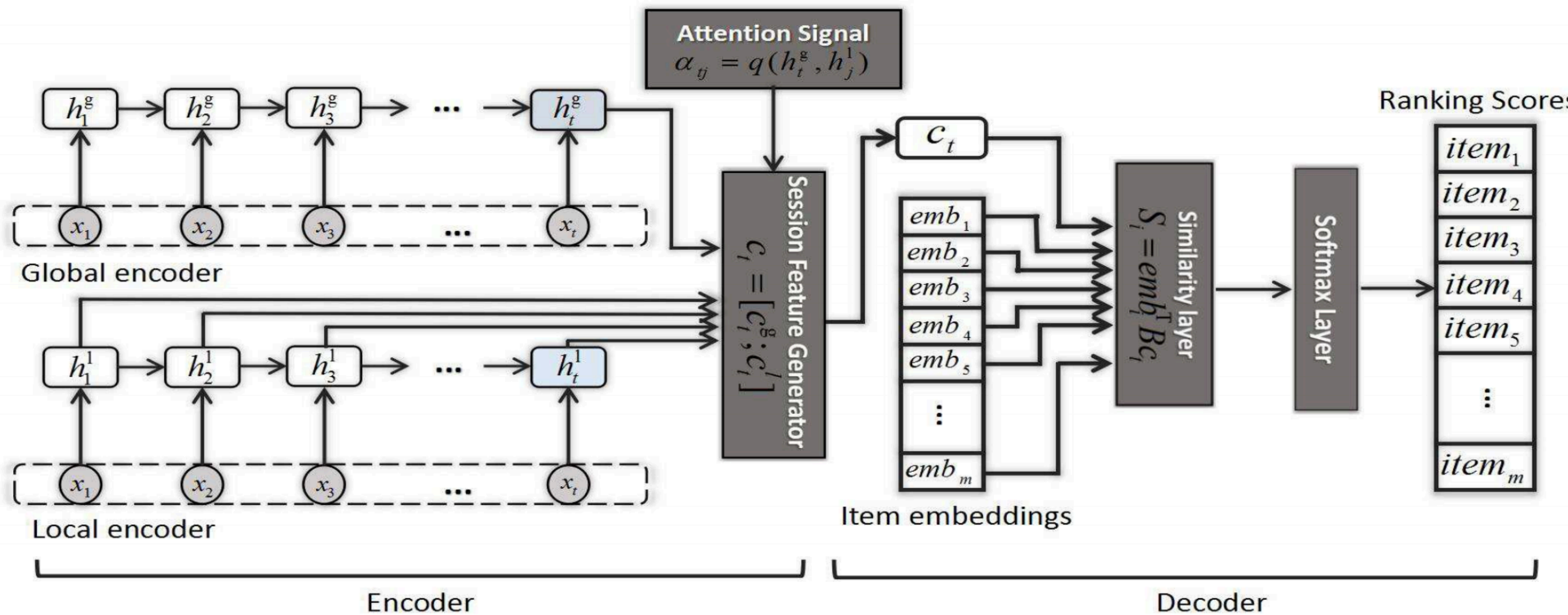


Figure 4: The graphical model of NARM, where the session feature c_t is represented by the concatenation of vectors c_t^g and c_t^l (as computed in equation (5) and (6)). Note that h_t^g and h_t^l play different roles, while they have the same values. The last hidden state of the global encoder h_t^g plays a role to encode the entire input clicks while the last hidden state of the local encoder h_t^l is used to compute attention weights with the previous hidden states.

Li, Jing, et al. "Neural attentive session-based recommendation." *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 2017.

More Novel Methods

Learning to Rank
Explore / Exploit
Session Based Recommendation
Deep Learning

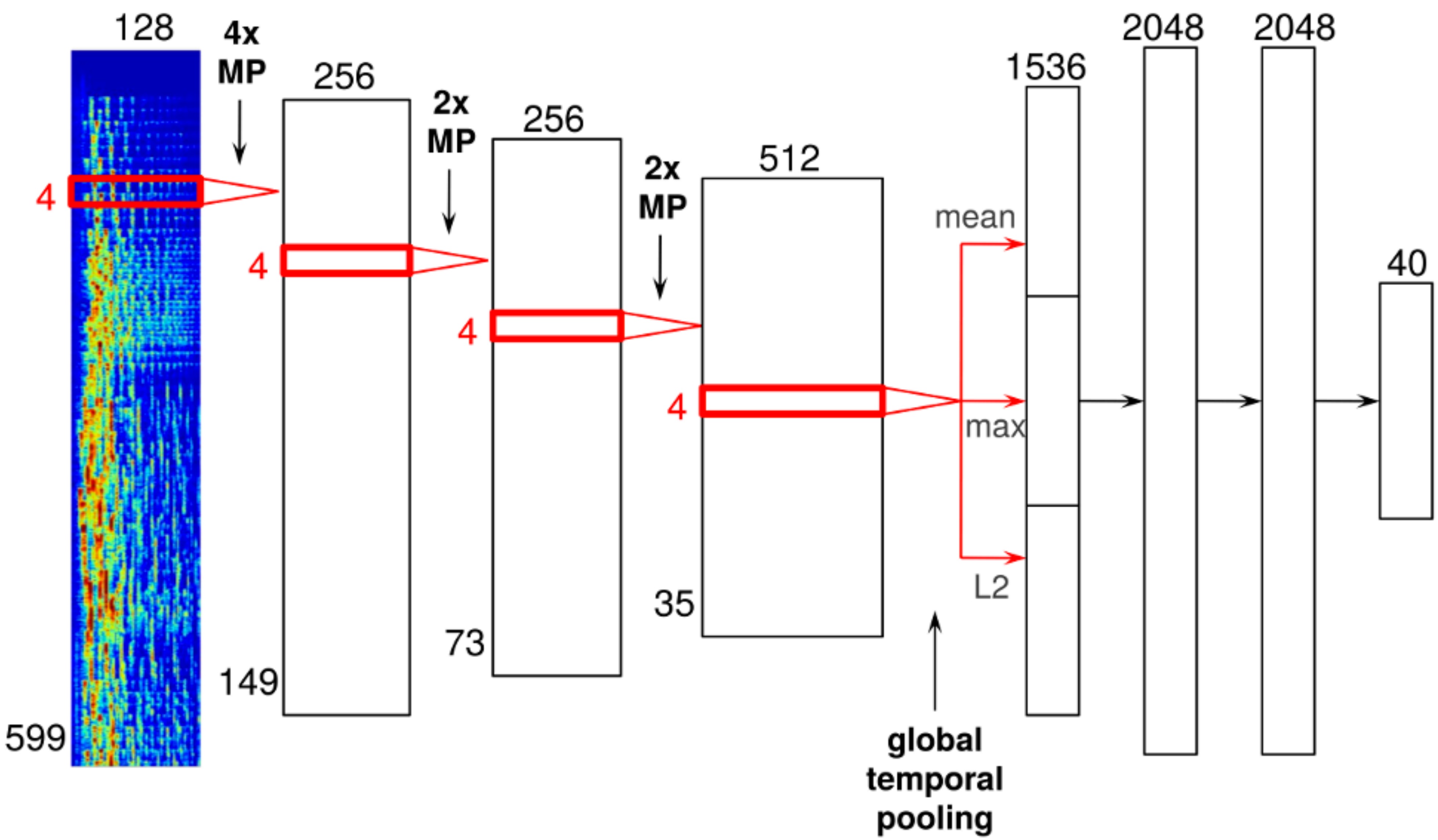
Deep learning for CB (without user history)

Similarity measure using pre-trained deep models (VGG, ResNet, ...)

Any deep model could be used for measuring 'similarity' of the given items https://www.researchgate.net/figure/t-SNE-visualization-of-clothing-items-visual-features-embedding-Distinctive-classes-of_fig5_322355346

t-SNE visualization of clothing items' visual features embedding. Distinctive classes of objects, e.g. those that share visual similarities are clustered around the same region of the space.

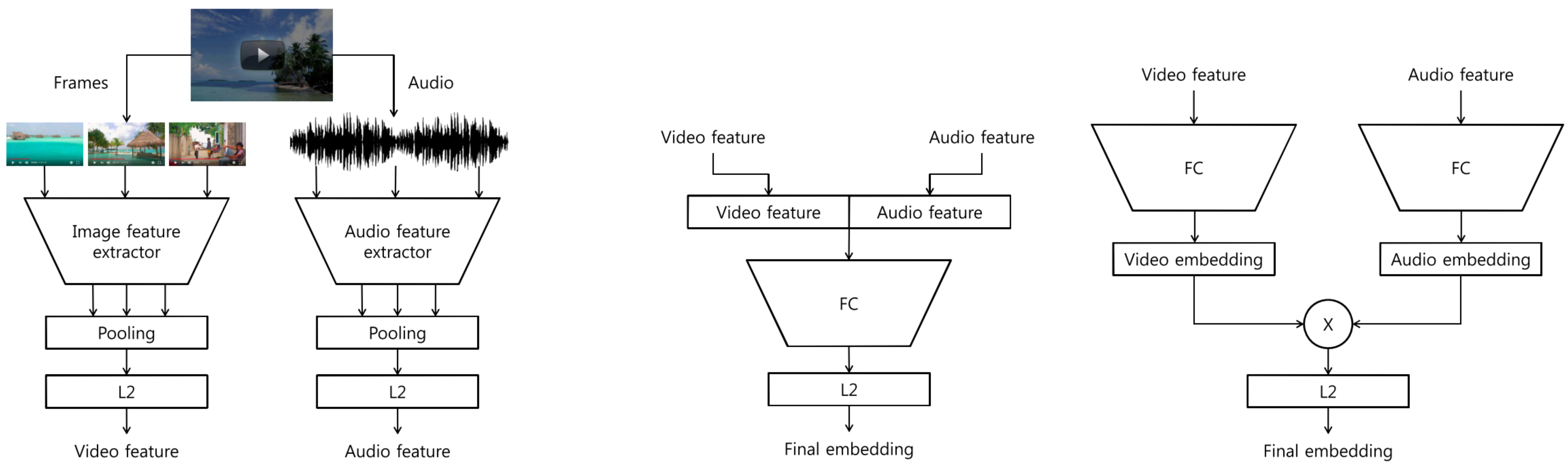
Deep content-based music recommendation



CF item feature

http://benanne.github.io/2014/08/05/spotify-cnns.html

Collaborative Deep Metric Learning For Video Understanding

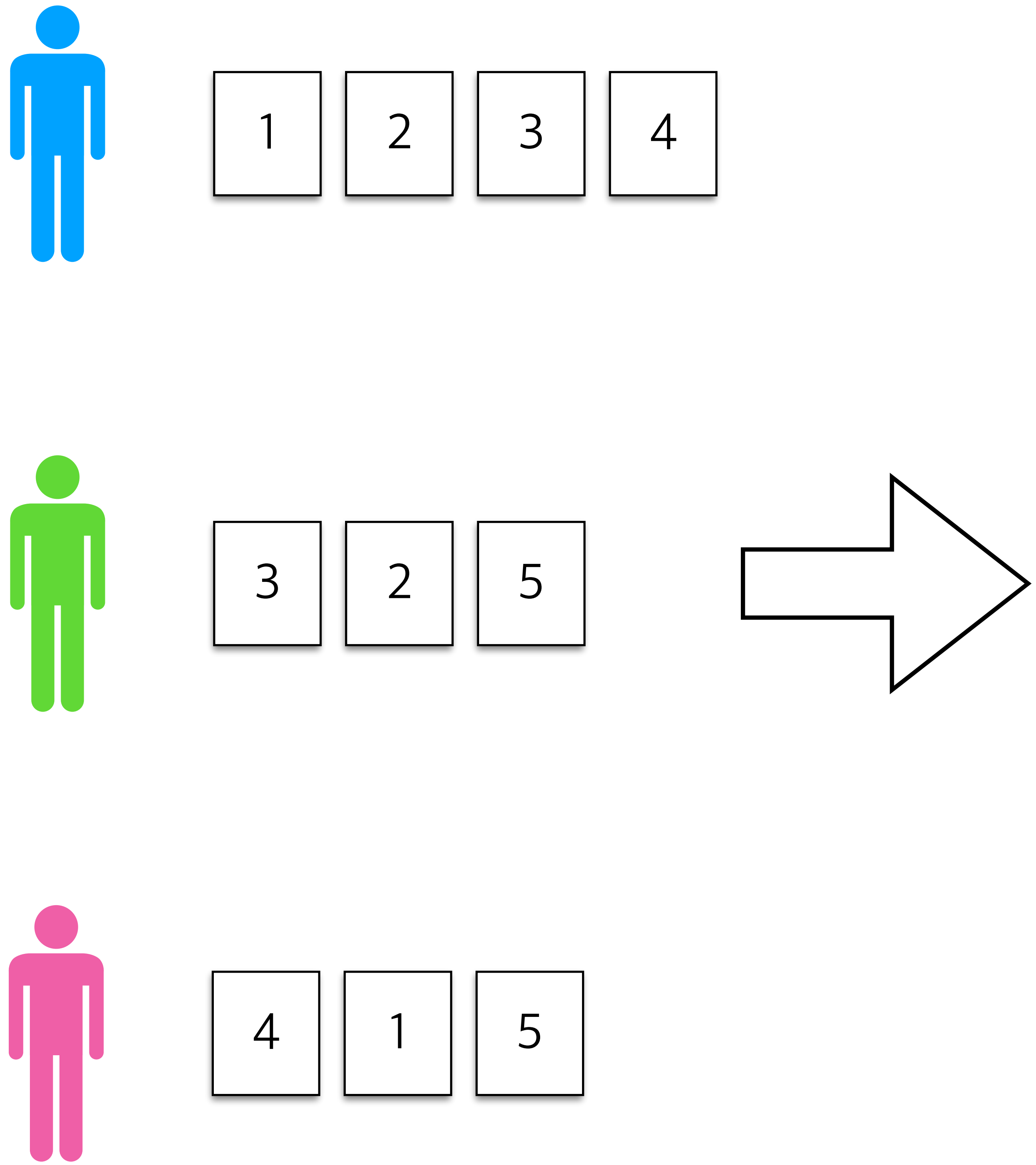


(a) **Feature extraction**. From sampled frames and low-level audio signals, we extract video and audio features with pre-trained models.

(b) Two possible variants of **embedding network** f. Video and audio features are combined, either from the beginning (early fusion; *left*) or with element-wise multiplication after two separate towers for each of them (late fusion; *right*).

Target: triplet loss (positive: co-watch, co-clicked, ...)

Lee, Joonseok, et al. "Collaborative Deep Metric Learning for Video Understanding." (2018).



Could embed users into feature space using their watching history

$$\max_{v \in V - Q} \frac{1}{|Q|} \sum_{q \in Q} \cos(f(\mathbf{x}_q), f(\mathbf{x}_v)),$$

$$\max_{v \in V - Q} \max_{q \in Q} \cos(f(\mathbf{x}_q), f(\mathbf{x}_v)).$$

Lee, Joonseok, et al. "Collaborative Deep Metric Learning for Video Understanding." (2018).

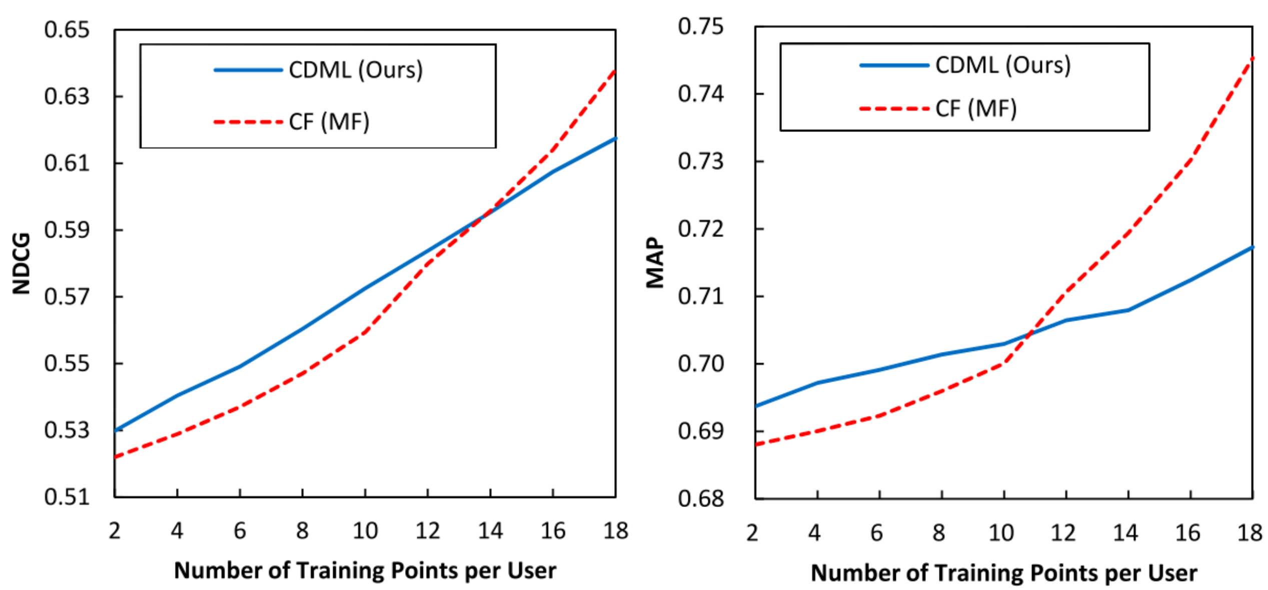


Figure 5: Cold-start recommendation performance with different number of training data points per user, in NDCG (*left*) and MAP (*right*). We see that our CDML is relatively stronger for colder start cases.

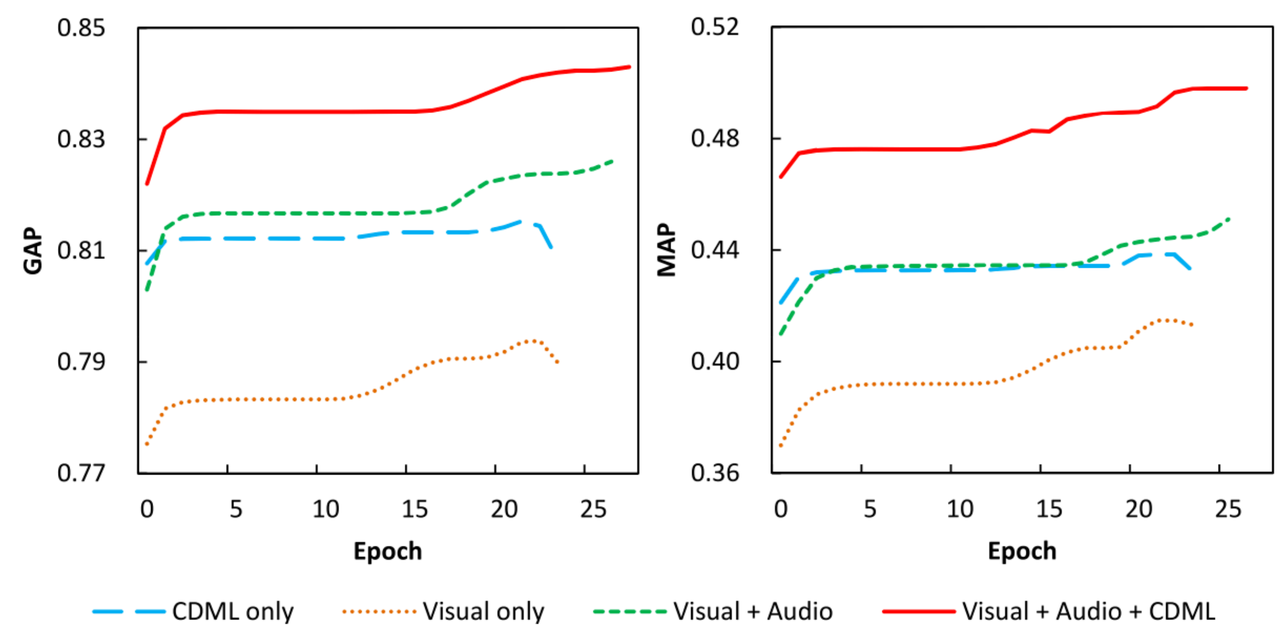


Figure 6: YouTube-8M video classification training curve comparing different features. Adding CDML features improves classification accuracy, by bringing the complimentary user behavior information to the content information.

Lee, Joonseok, et al. "Collaborative Deep Metric Learning for Video Understanding." (2018).

More readings

- Cheng, Heng-Tze, et al. "Wide & deep learning for recommender systems." Proceedings of the 1st Workshop on Deep Learning for Recommender Systems. ACM, 2016.

- Covington, Paul, Jay Adams, and Emre Sargin. "Deep neural networks for youtube recommendations." Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016.

Real World RecSys

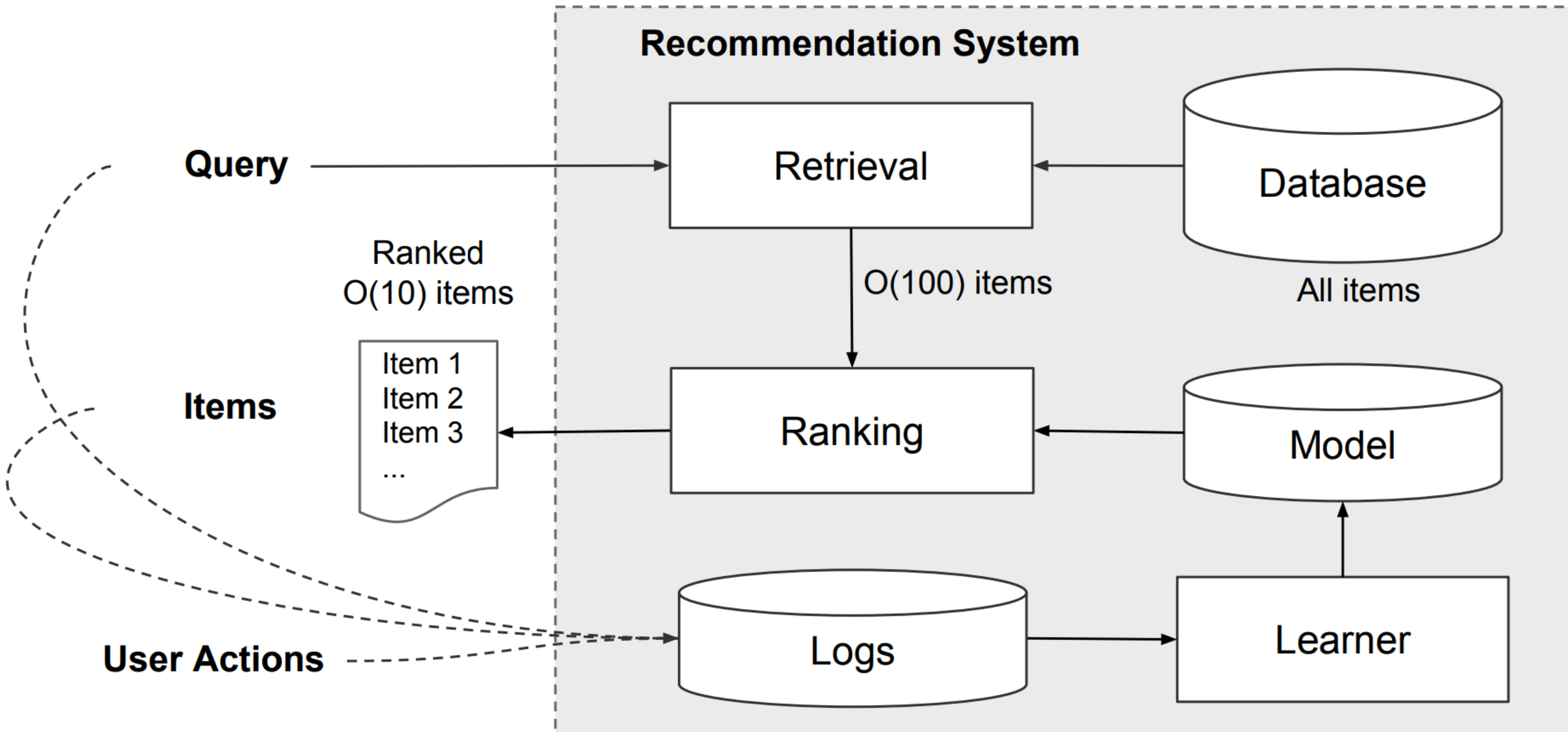
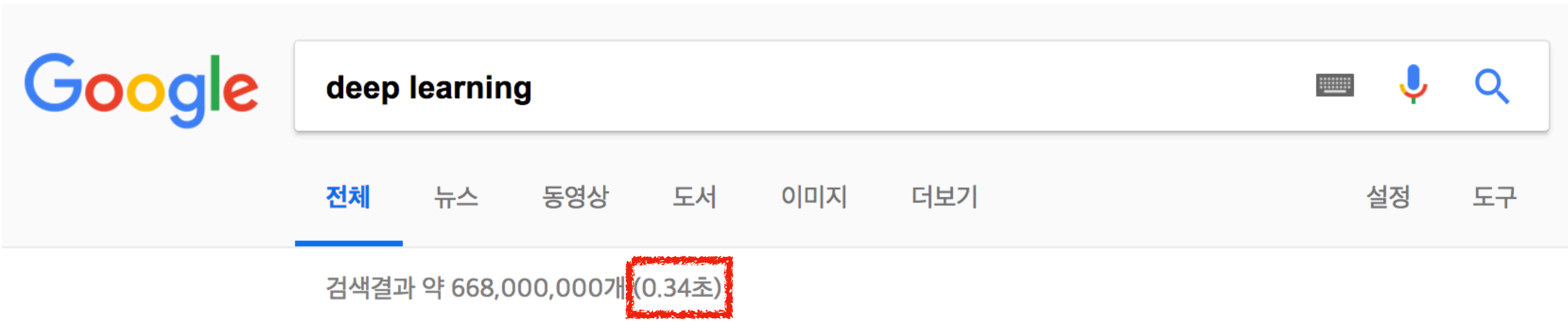
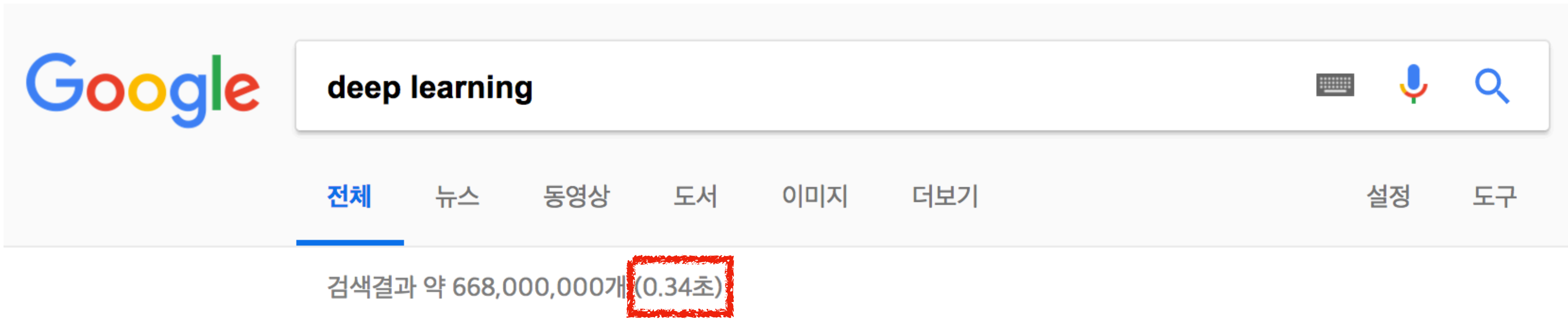


Figure 2: Overview of the recommender system.

Cheng, Heng-Tze, et al. "Wide & deep learning for recommender systems." Proceedings of the 1st Workshop on Deep Learning for Recommender Systems. ACM, 2016.



https://github.com/jcjohnson/cnn-benchmarks



Network	Layers	Top-1 error	Top-5 error	Speed (ms)	Citation
AlexNet	8	42.90	19.80	14.56	[1]
Inception-V1	22	-	10.07	39.14	[2]
VGG-16	16	27.00	8.80	128.62	[3]
VGG-19	19	27.30	9.00	147.32	[3]
ResNet-18	18	30.43	10.76	31.54	[4]
ResNet-34	34	26.73	8.74	51.59	[4]
ResNet-50	50	24.01	7.02	103.58	[4]
ResNet-101	101	22.44	6.21	156.44	[4]
ResNet-152	152	22.16	6.16	217.91	[4]
ResNet-200	200	21.66	5.79	296.51	[5]

https://github.com/jcjohnson/cnn-benchmarks

ResNet-200 Benchmark

GPU	cuDNN	Forward (ms)	Backward (ms)	Total (ms)
Pascal Titan X	5.1.05	104.74	191.77	296.51
Pascal Titan X	5.0.05	104.36	201.92	306.27
Maxwell Titan X	5.0.05	170.03	320.80	490.83
Maxwell Titan X	5.1.05	169.62	383.80	553.42
Maxwell Titan X	4.0.07	203.52	356.35	559.87
Pascal Titan X	None	314.77	519.72	834.48
Maxwell Titan X	None	497.57	953.94	1451.51
CPU: Dual Xeon E5-2630 v3	None	8666.43	13758.73	22425.16

ResNet-200 Benchmark

GPU	cuDNN	Forward (ms)	Backward (ms)	Total (ms)
Pascal Titan X	5.1.05	104.74	191.77	296.51
Pascal Titan X	5.0.05	104.36	201.92	306.27
Maxwell Titan X	5.0.05	170.03	320.80	490.83
Maxwell Titan X	5.1.05	169.62	383.80	553.42
Maxwell Titan X	4.0.07	203.52	356.35	559.87
Pascal Titan X	None	314.77	519.72	834.48
Maxwell Titan X	None	497.57	953.94	1451.51
CPU: Dual Xeon E5-2630 v3	None	8666.43	13758.73	22425.16

Sponsored

Nvidia Tesla v100 16GB

by PNY

$8,439.00 + $75.00 shipping

Only 2 left in stock - order soon.

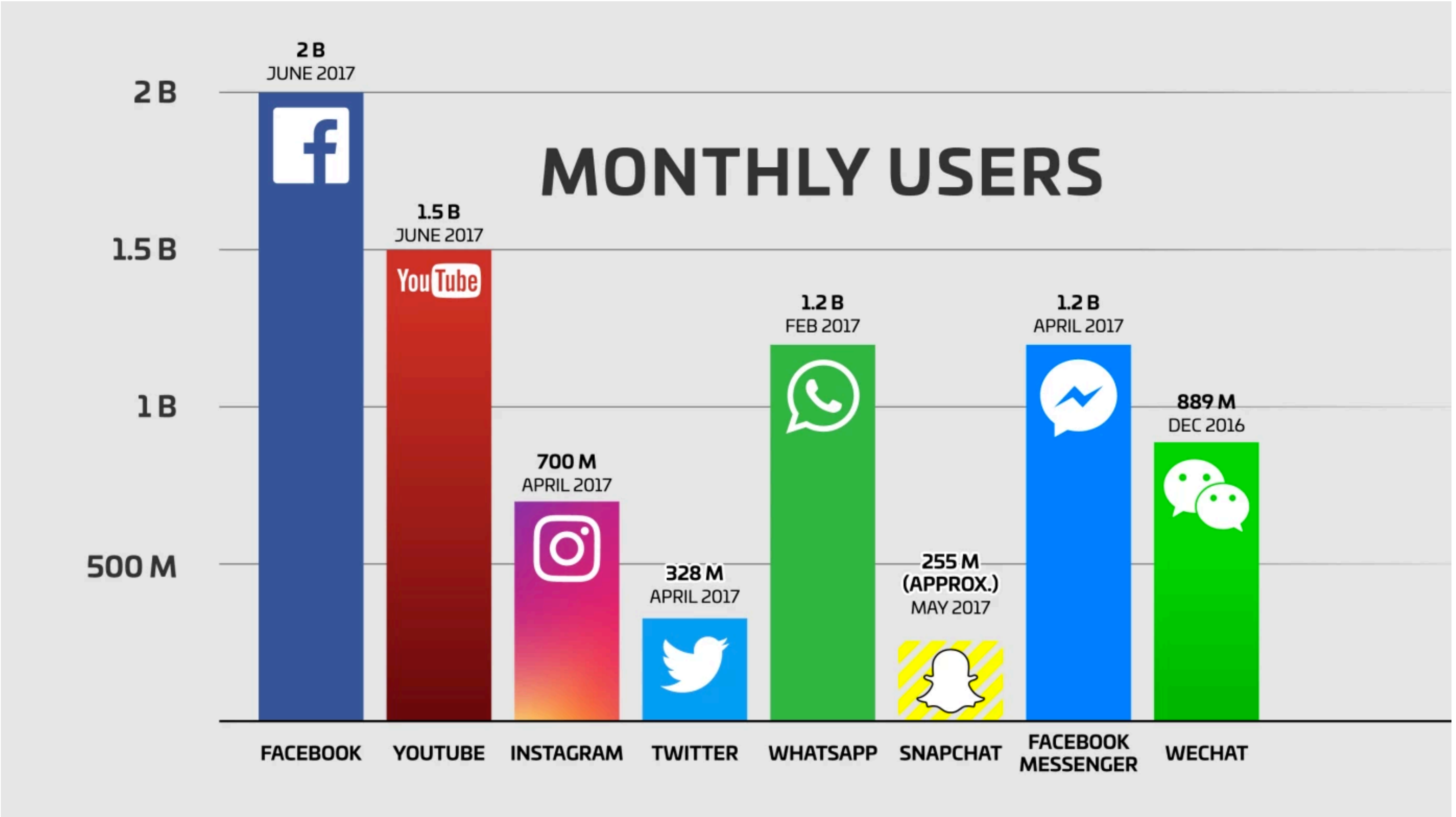
NVIDIA 900-2G610-0000-000 TESLA P40 24GB

by NVIDIA

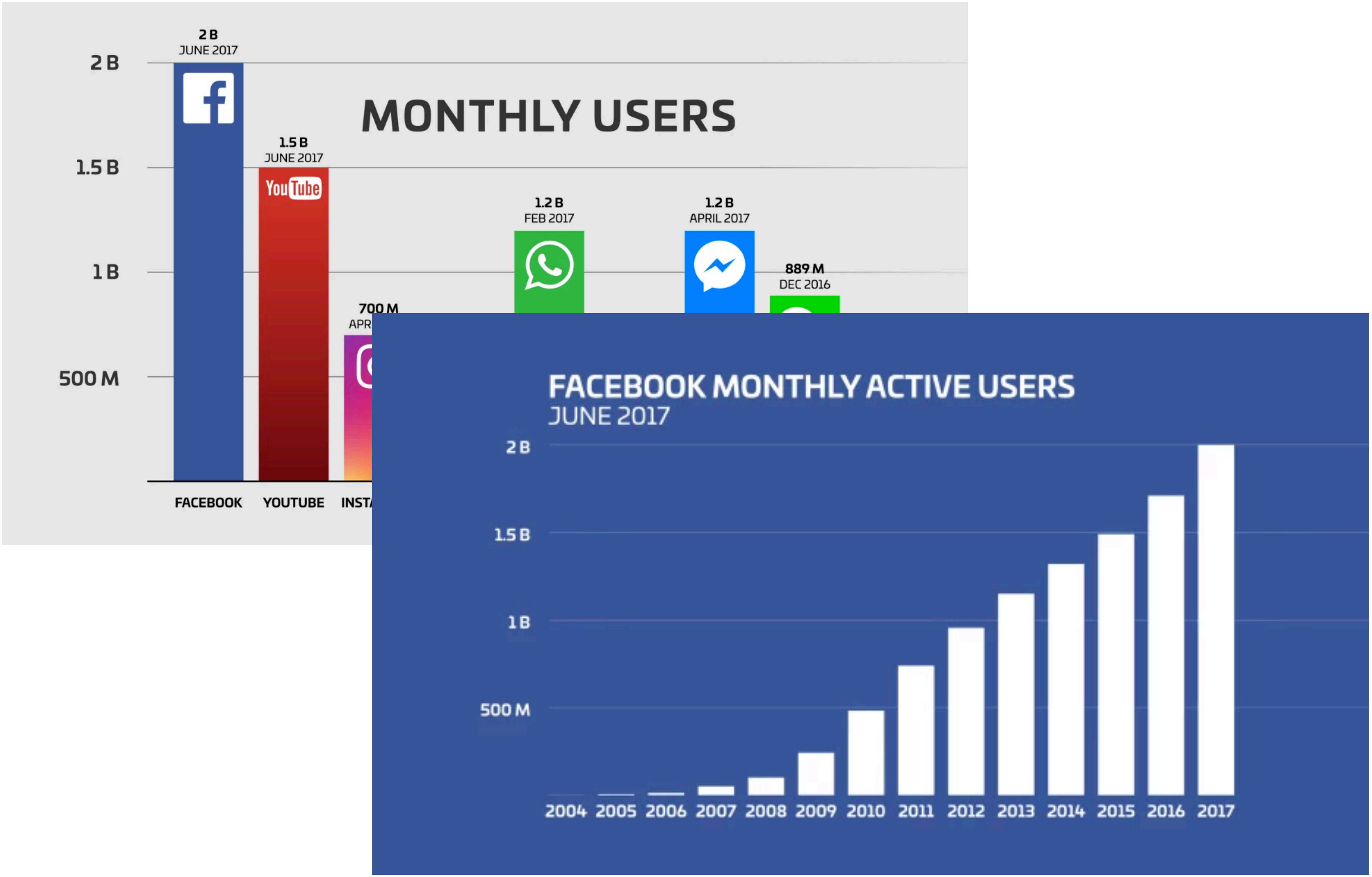
Eligible for Shipping to Korea, Republic of

$5,545.00

Only 1 left in stock - order soon.



https://techcrunch.com/2017/06/27/facebook-2-billion-users/



https://techcrunch.com/2017/06/27/facebook-2-billion-users/

As YouTube receives more than 90 PB of videos data every year

It has more than 7 billion videos available out of 5 billion videos watched every day by more than 30 million users

It will take more than 199771 Year to watch all videos available on YouTube.

All data are stored at the Google modular Data center located at different locations.

https://www.quora.com/How-many-videos-are-on-YouTube-2017-1

Technical difficulty: Scalability

Real-world RecSys should be able to handle super super many queries (5B queries per day)

Real-world RecSys could compute similarity over all-items (over 7B videos), even KNN takes hopelessly long time

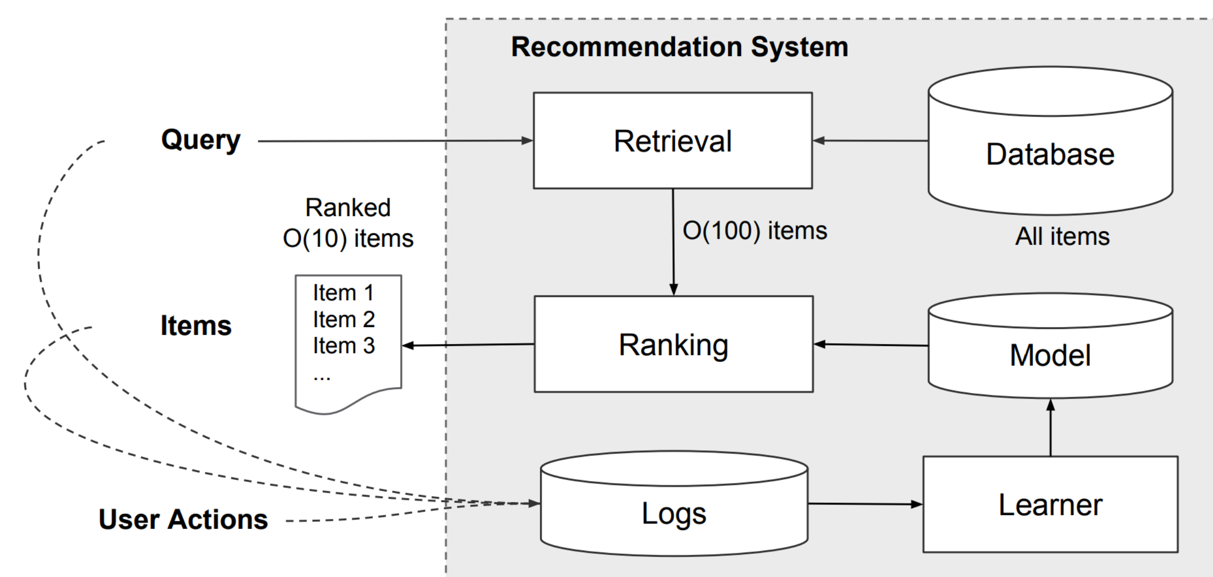


Figure 2: Overview of the recommender system.

KNN

```
import time
import numpy as np
```

```
dim = 100
N = 10000000   # 10M

topk = 10
```

```
x = np.random.random(dim)
D = np.random.random((N, dim))
```

```
# assume that x and D are normalized
t = time.time()
print(-np.argsort(D.dot(x))[:topk])
print(time.time() - t)
```

```
[-6106596 -5824451 -5924596 -6692504 -1097690 -1103777 -2550588 -6209227
 -1983432  -893117]
6.813840866088867
```

How can we make faster KNN?

complexity of Naïve method: $O(d^2N)$

Easy way: reduce d or N (d -> 20, 10, filtering N)

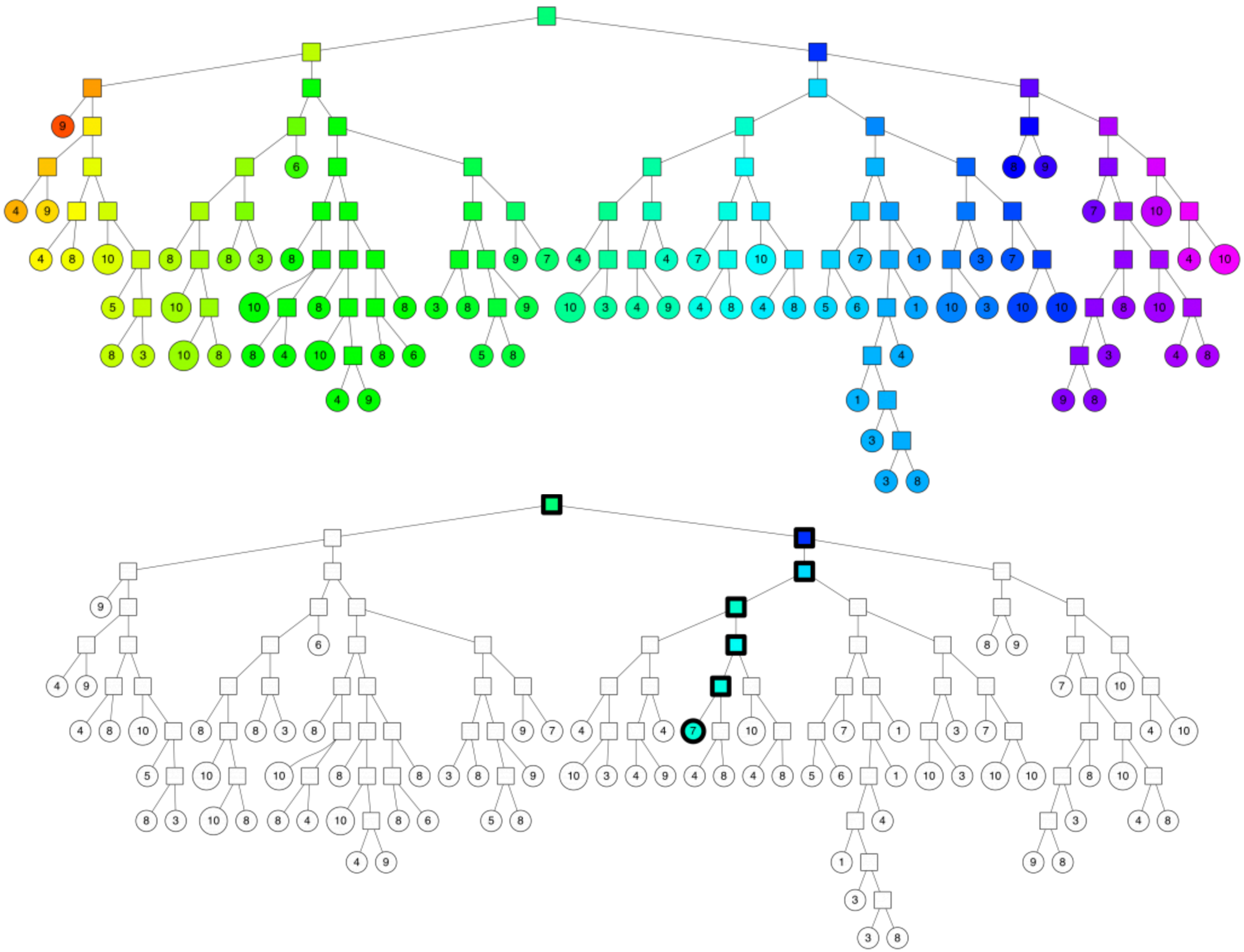
Other ways:

- use data structure (tree, graph)

- quantization + indexing

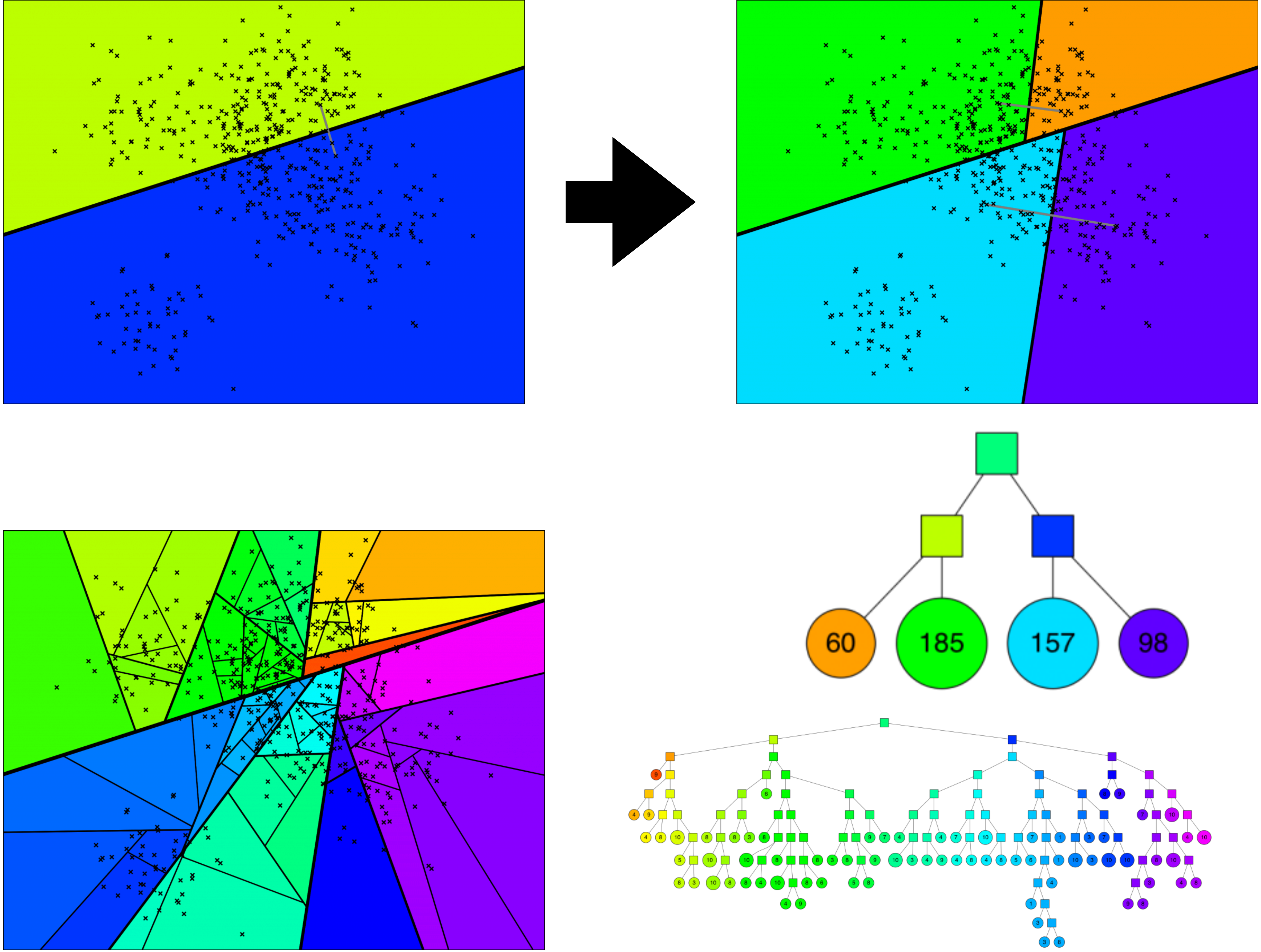
- .....

Annoy

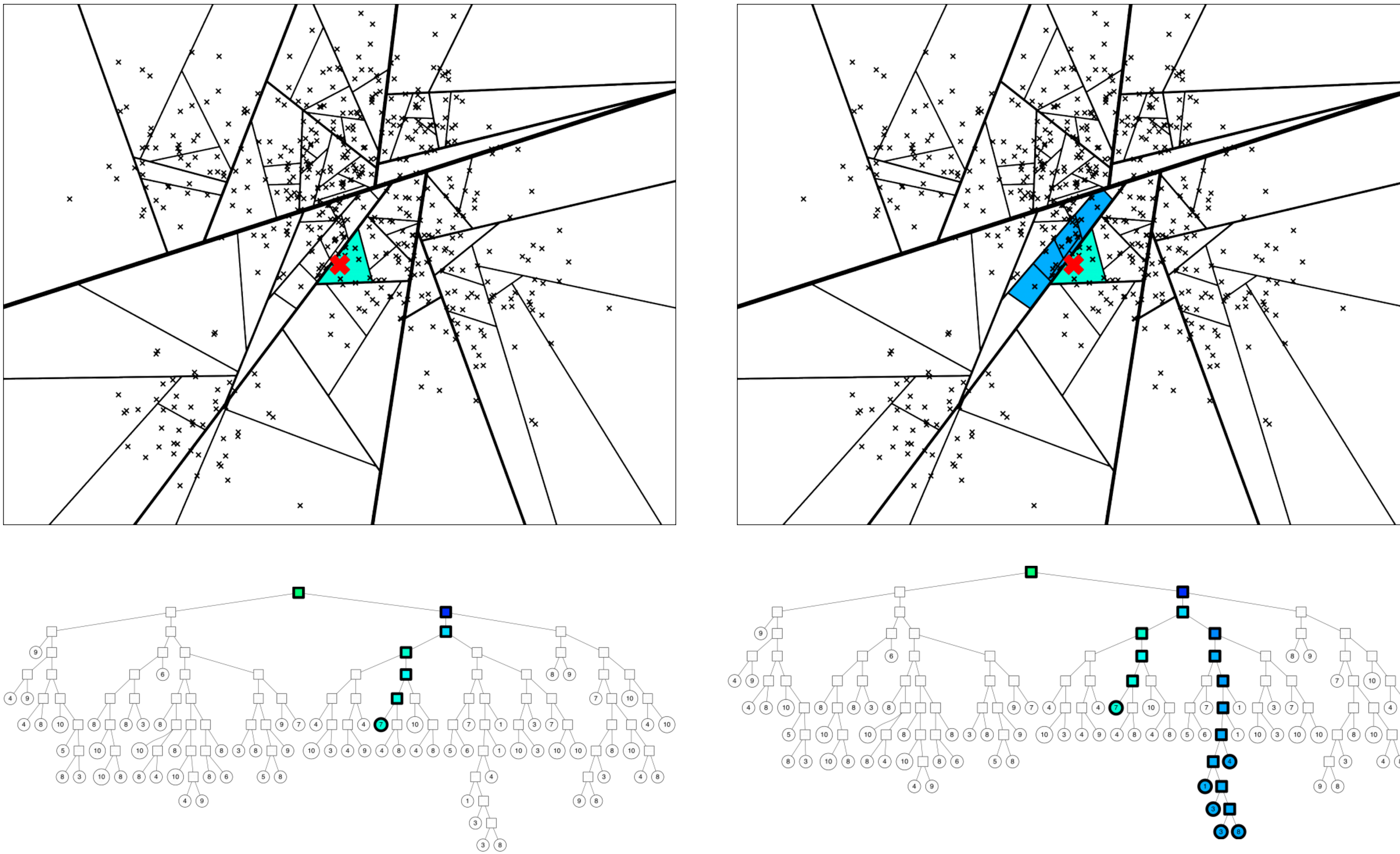


Build tree by random projection

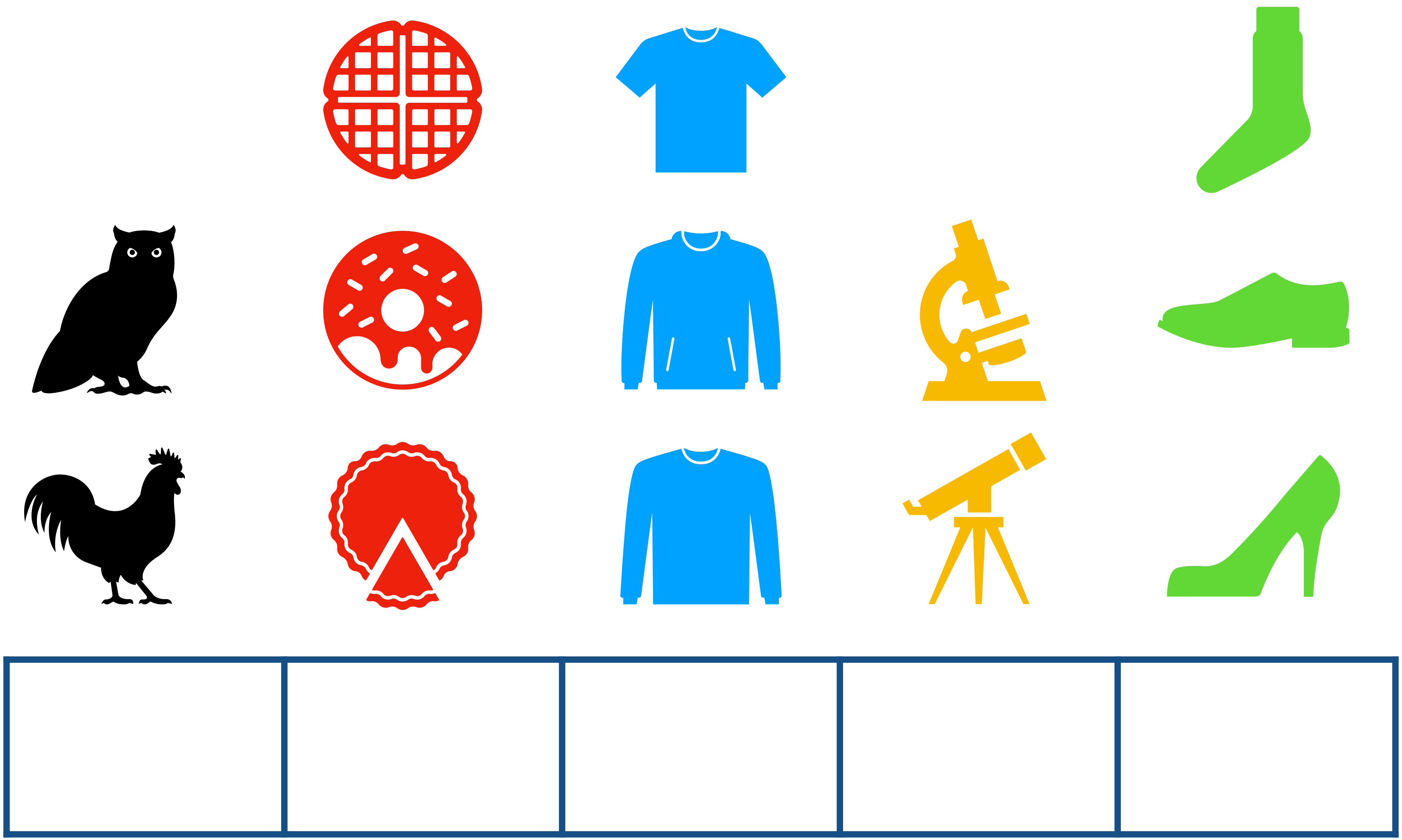
https://erikbern.com/2015/10/01/nearest-neighbors-and-vector-models-part-2-how-to-search-in-high-dimensional-spaces.html


60
185
157
98

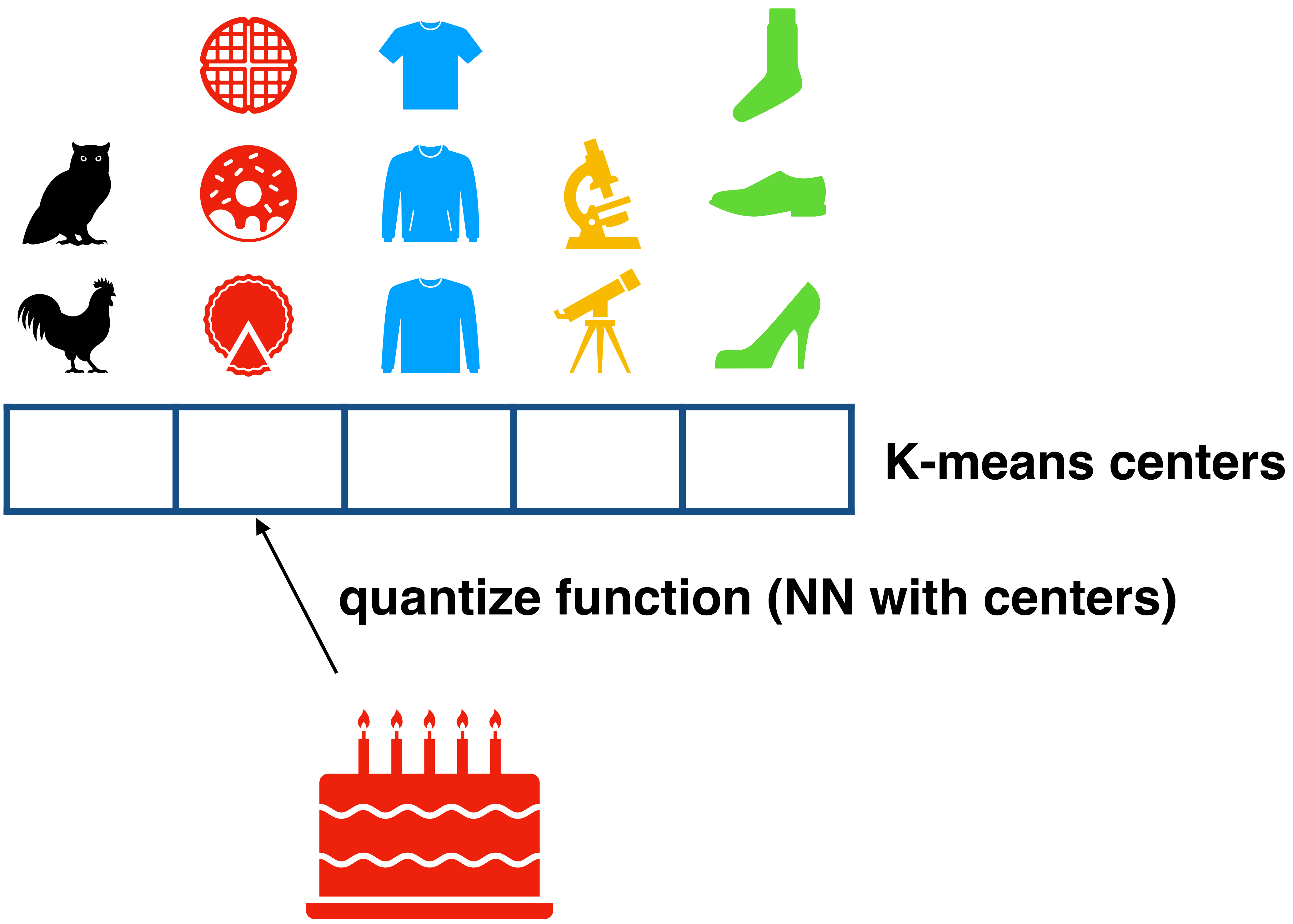
Priority Queue for ensuring "K"



Feature quantization + indexing



VQ (vector quantization)

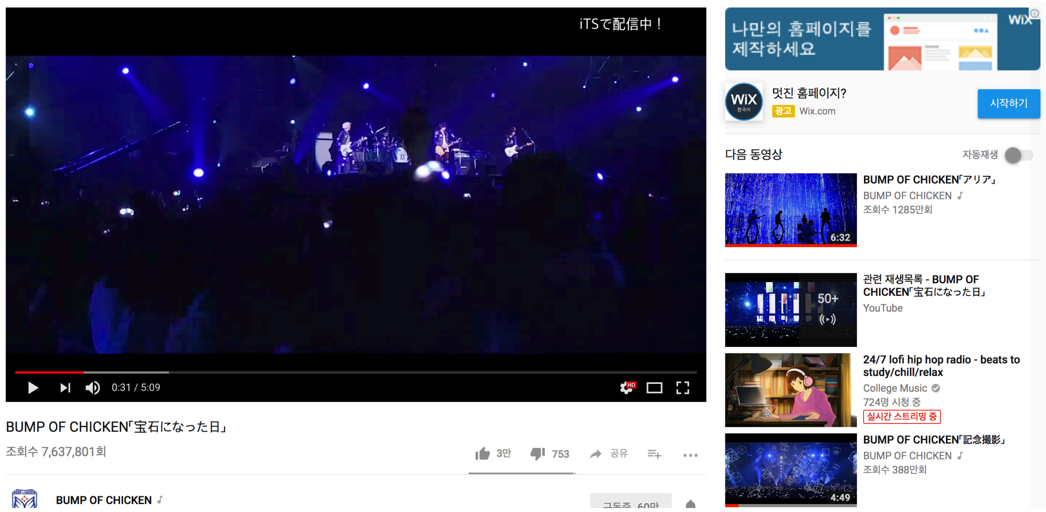


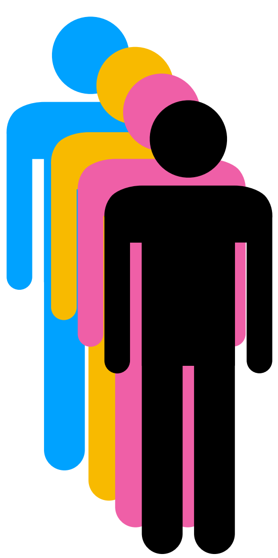
More readings

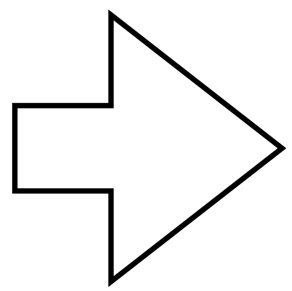
- **[Graph based ANN]** Malkov, Yu A., and Dmitry A. Yashunin. "Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World graphs." arXiv preprint arXiv:1603.09320(2016).

- **[end-to-end deep quantization method]** Jeong, Yeonwoo, and Hyun Oh Song. "Efficient end-to-end learning for quantizable representations." ICML 2018

Dealing with streaming data

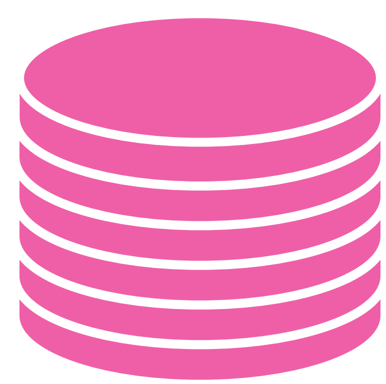






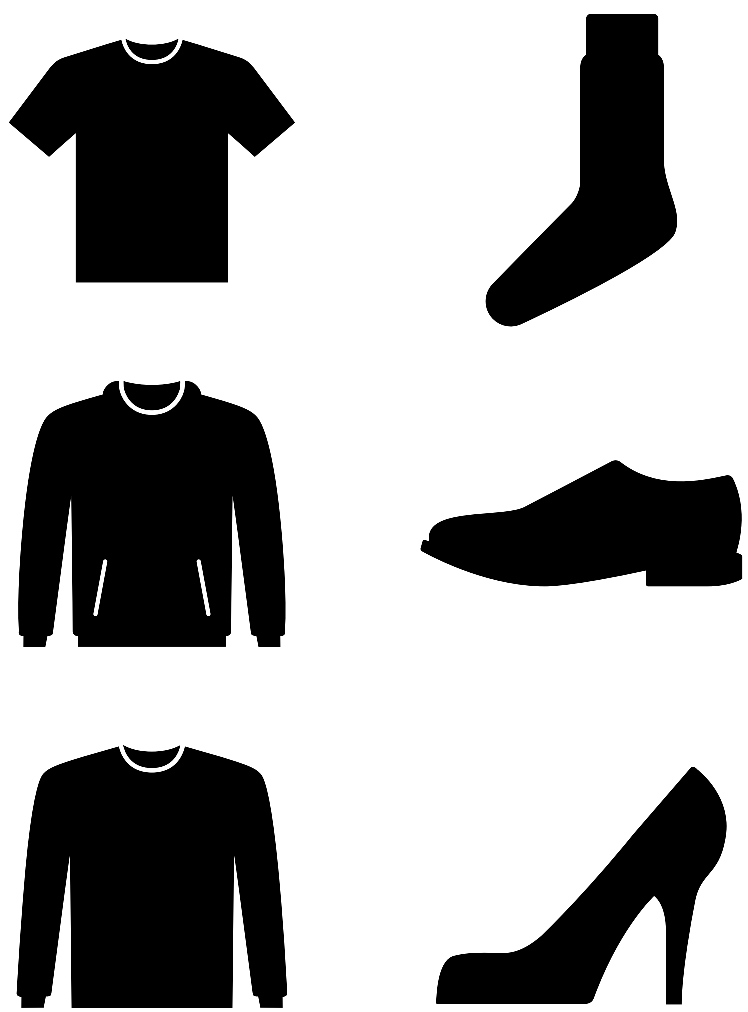
timestamp, user, item, referer

08-25 12:00, A , item1, -
08-25 12:10, A , item2, item1
08-25 12:11, A , item3, item2
08-25 12:11, B , item2, -
08-25 12:12, A , item4, item3
08-25 12:13, B , item1, item2
....

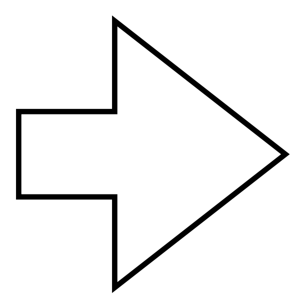


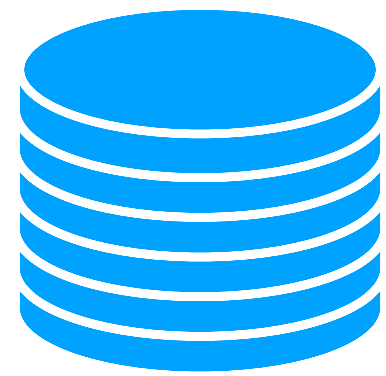
User history storage

Dealing with streaming data



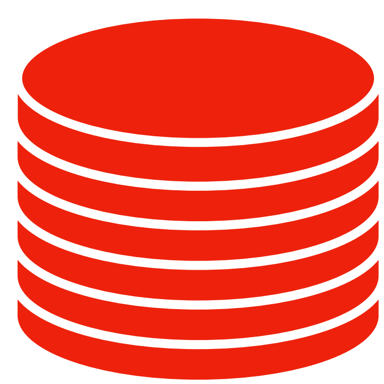
New items





Model storage

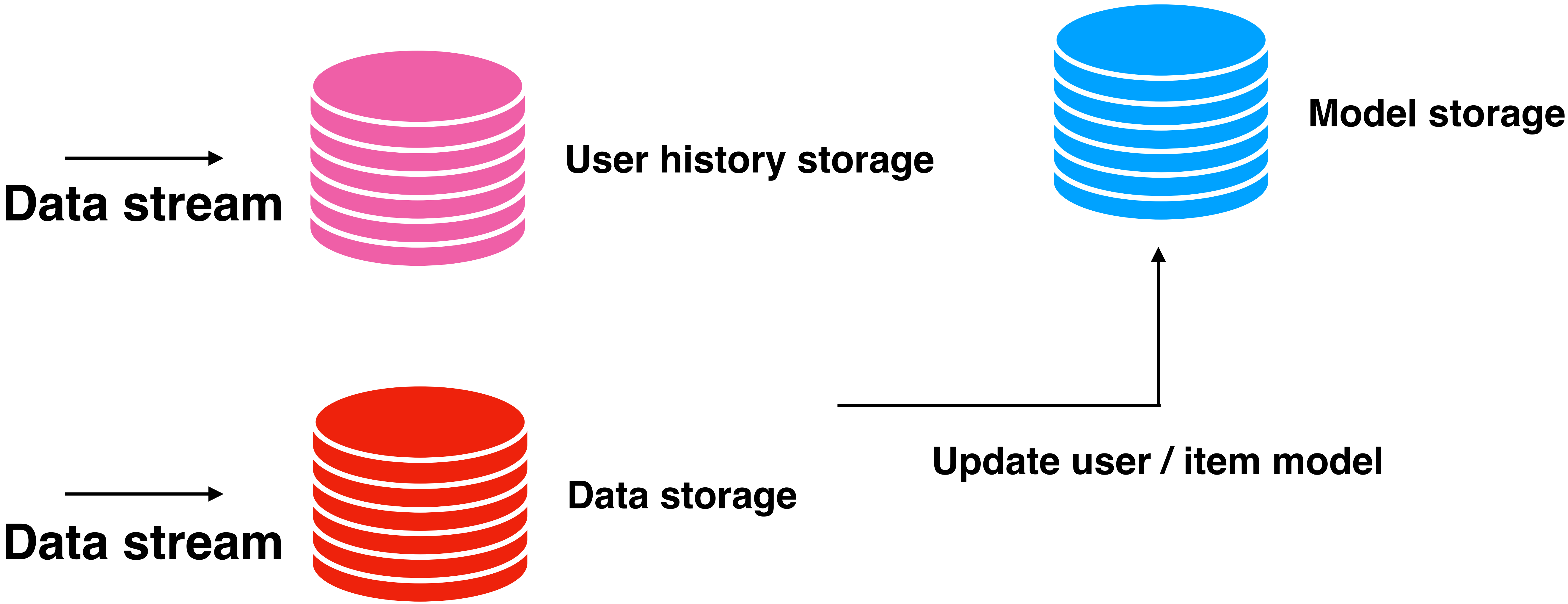
Feature extraction via various models (e.g. CF, CB, ...)



Data storage

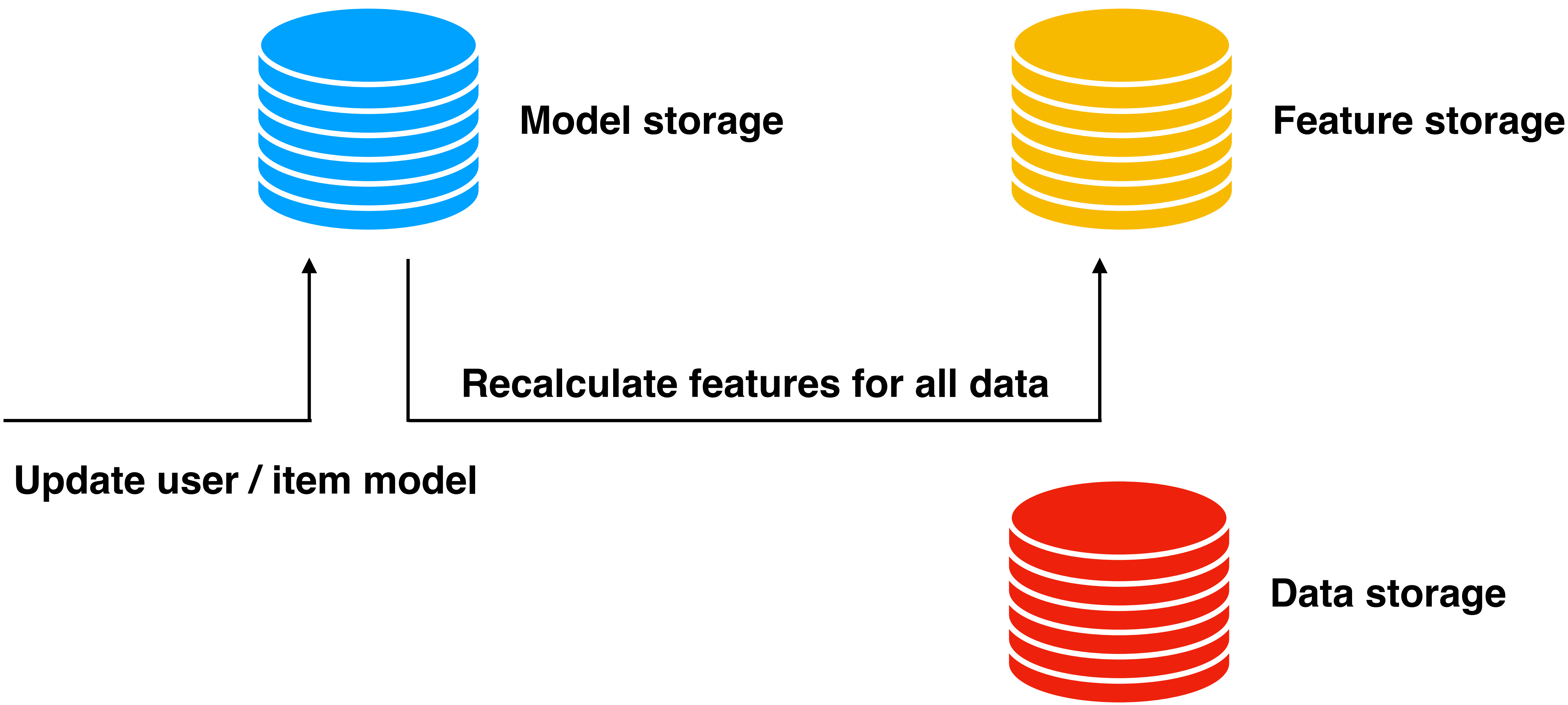
Feature storage

Incremental learning issue

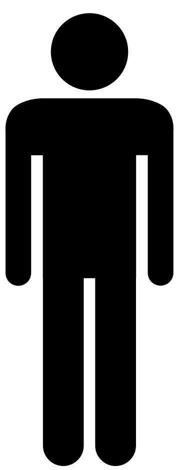


ETL (Extract / Transform / Load)

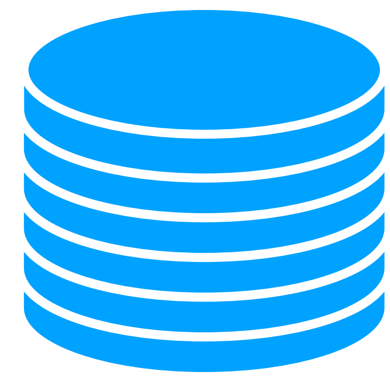
Incremental learning issue



How to serve?



1. Compute user /item feature
2. Choose A/B bucket
3. Get recommendation results from various models
4. Mix all of them up into single list
5. Re-ranking using ranking algorithm



Model storage

Feature storage

Serve recommendations to users

Be updated frequently!

- incremental model learning (batch)
- dealing with NEW item / user (online)

Computation issue

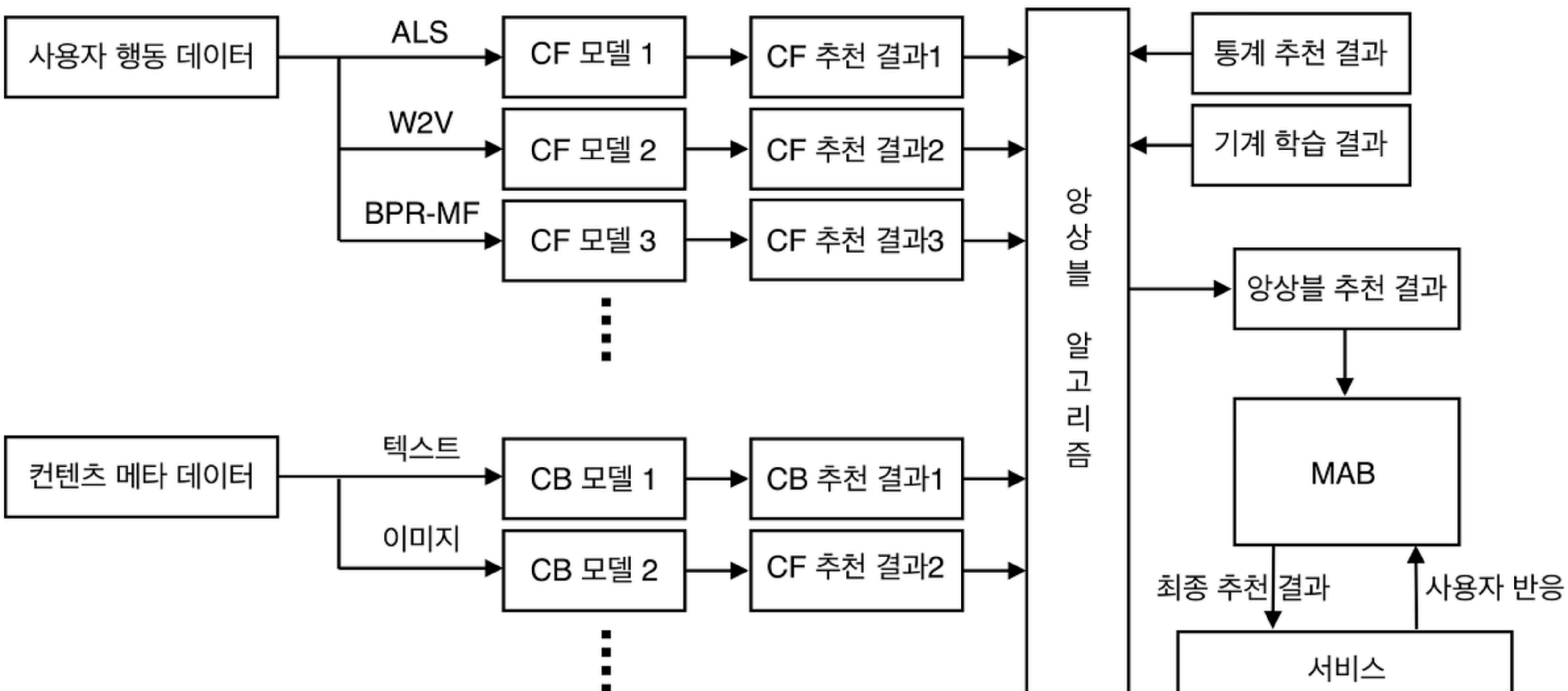
Inference using Machine learning models sometimes suffers from high computation cost

Possible solutions

- Caching & Indexing (i.e., save all recommendations to databases)

- Use more light and fast models (not 'deep' model)

토로스는 CF, CB, 통계 모델, 일반적인 기계학습 모델 등 다양한 모델들에서 추천 결과를 뽑고 뽑은 추천 결과를 앙상블하여 하나의 추천 결과로 병합한다. 만들어진 추천 결과가 사용자들에게 노출되기 시작하면 MAB를 사용해 가장 좋은 추천 결과가 무엇일지 찾아낸다.



https://brunch.co.kr/@kakao-it/72

References

- Introduction to Recommender Systems in 2018, tryo labs
- How Does Spotify Know You So Well? (Medium)
- NETFLIX PRIZE – 다이나믹 했던 알고리즘 대회 (3), shalomeir's blog
- RecSys 2016: Tutorial on Lessons Learned from Building Real-life Recommender Systems
- Recommender Systems @ML Summer School 2014 in Pittsburgh
- Koren, Yehuda, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems." Computer 8 (2009): 30-37.
- Negahban, Sahand, Sewoong Oh, and Devavrat Shah. "Rank centrality: Ranking from pairwise comparisons." Operations Research 65.1 (2016): 266-287.
- TOROS: Python Framework for Recommender System

References

- Chapelle, Olivier, and Lihong Li. "An empirical evaluation of thompson sampling." Advances in neural information processing systems. 2011.
- Komiyama, Junpei, Junya Honda, and Hiroshi Nakagawa. "Optimal regret analysis of thompson sampling in stochastic multi-armed bandit problem with multiple plays." ICML 2015
- Hidasi, Balázs, et al. "Session-based recommendations with recurrent neural networks." ICLR 2016
- Quadrana, Massimo, et al. "Personalizing session-based recommendations with hierarchical recurrent neural networks." Proceedings of the Eleventh ACM Conference on Recommender Systems. ACM, 2017.

References

- Van den Oord, Aaron, Sander Dieleman, and Benjamin Schrauwen. "Deep content-based music recommendation." Advances in neural information processing systems. 2013.
- Lee, Joonseok, et al. "Collaborative Deep Metric Learning for Video Understanding." (2018).
- Cheng, Heng-Tze, et al. "Wide & deep learning for recommender systems." Proceedings of the 1st Workshop on Deep Learning for Recommender Systems. ACM, 2016.
- [카카오AI리포트] 내 손안의 AI 비서 추천 알고리즘