DEVIEW 2020

추천시스템 3.0: 딥러닝 후기시대에서 바이어스, 그래프, 그리고 인과관계의 중요성

김경민, 곽하녹, 박영진, 김민규 Clova, NAVER

CONTENTS

1. 추천시스템 배경

진화하는 개인화 추천시스템

DEVIEW 2020

1997
Collaborative Filtering

2006
Netflix Prize
Hybrid 접근법

2010~
Convolutional Neural Networks
Recurrent Neural Networks

2016~
Causal Inference

2017~
Graph Neural Networks

2020
현재

1단계 **2단계** **3단계**

개인화 추천시스템의 세 단계

	추천시스템 1.0	추천시스템 2.0	추천시스템 3.0
데이터 해석력	하	하	상
데이터 크기	하	상	상
계산복잡도	하	상	상

데이터 생성 과정에 대한 깊은 이해를 바탕으로 빅데이터 Manipulation이 중요하다

DEVIEW 2020

Positional Bias Reduction

Graph Construction

From Statistic To Causality

적용 서비스: LINE Wallet

LINE Wallet 탭 구성

LINE Pay

서비스
바로가기

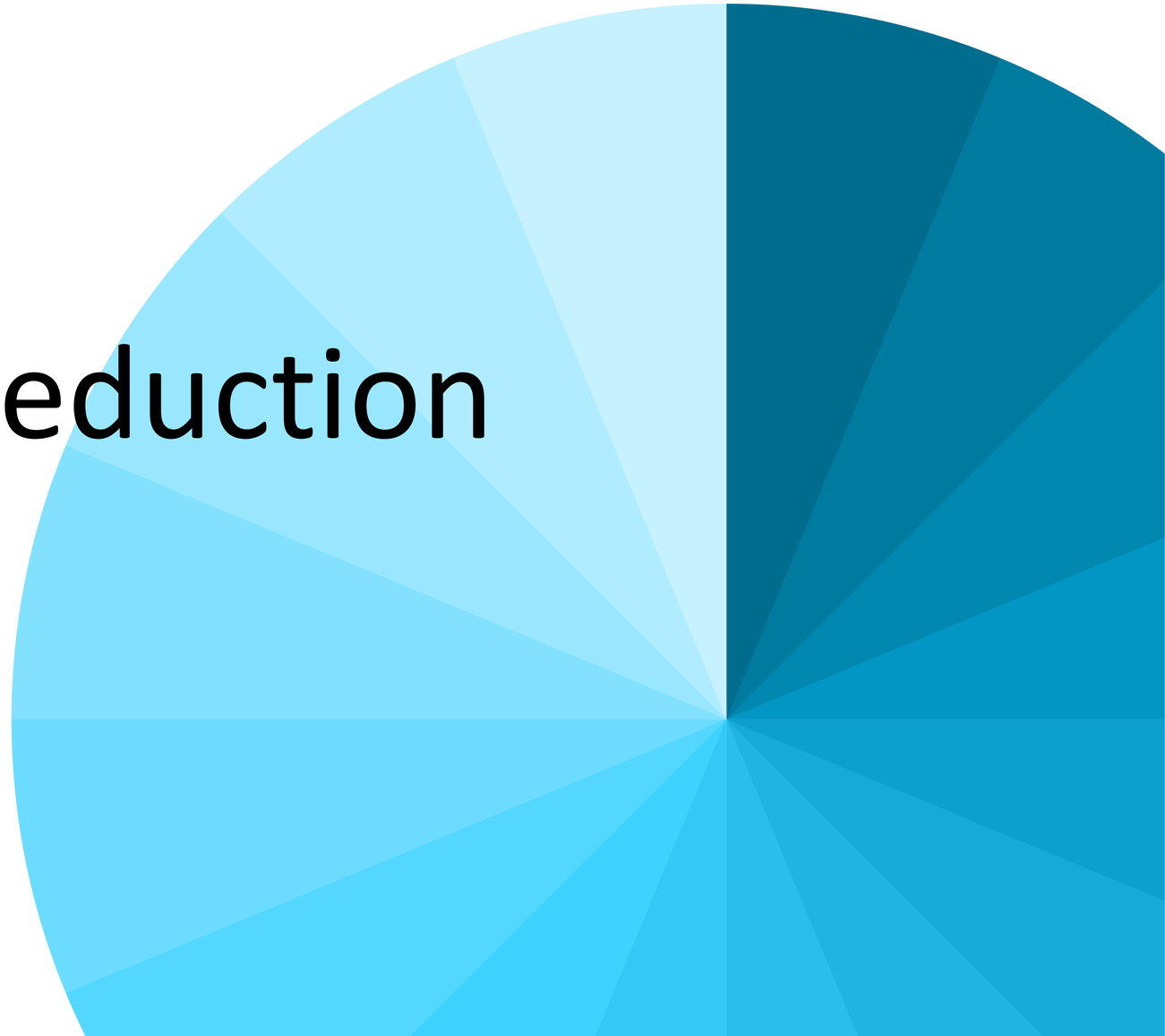
아이템

모듈
(서로 다른 콘텐츠)

모듈 예시

- Coupon
- MyCards
- Point
- Chirashi

2. Bias Reduction



UI추천: 내가 다음에 사용할 모듈은 UI 상단으로 미리 집합!

Prob. of next use 95%

Prob. of next use 80%

Four modules

- **Coupon**
- **MyCard**
- **Point**
- **Chirashi**

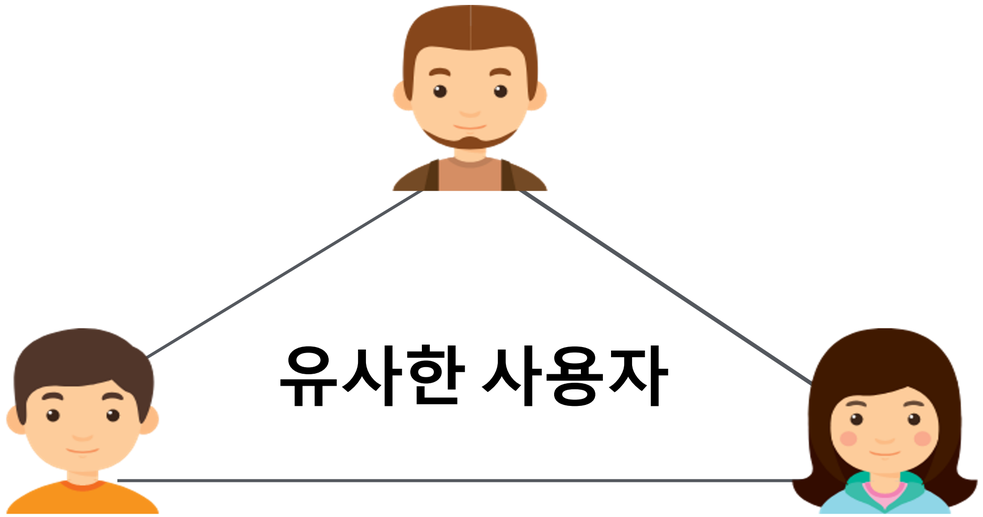
다음 사용 모듈을 어떻게 예측할까?
- 과거행동기반 예측알고리즘 -

사용자 과거 행동데이터를 모아서

월렛탭 행동로그 (클릭/뷰)

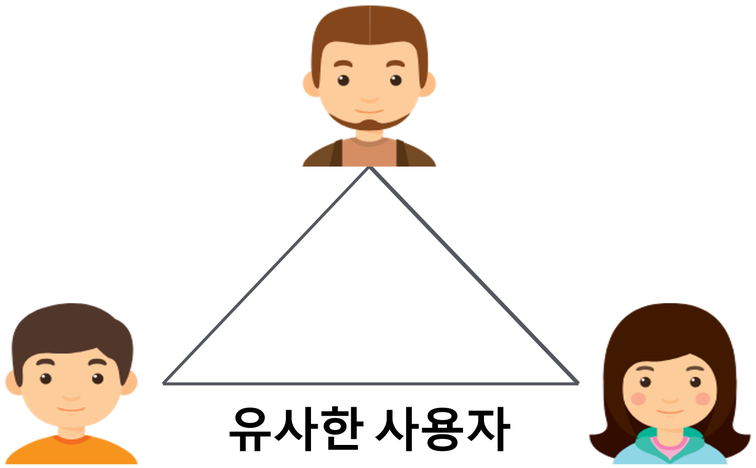
家計簿

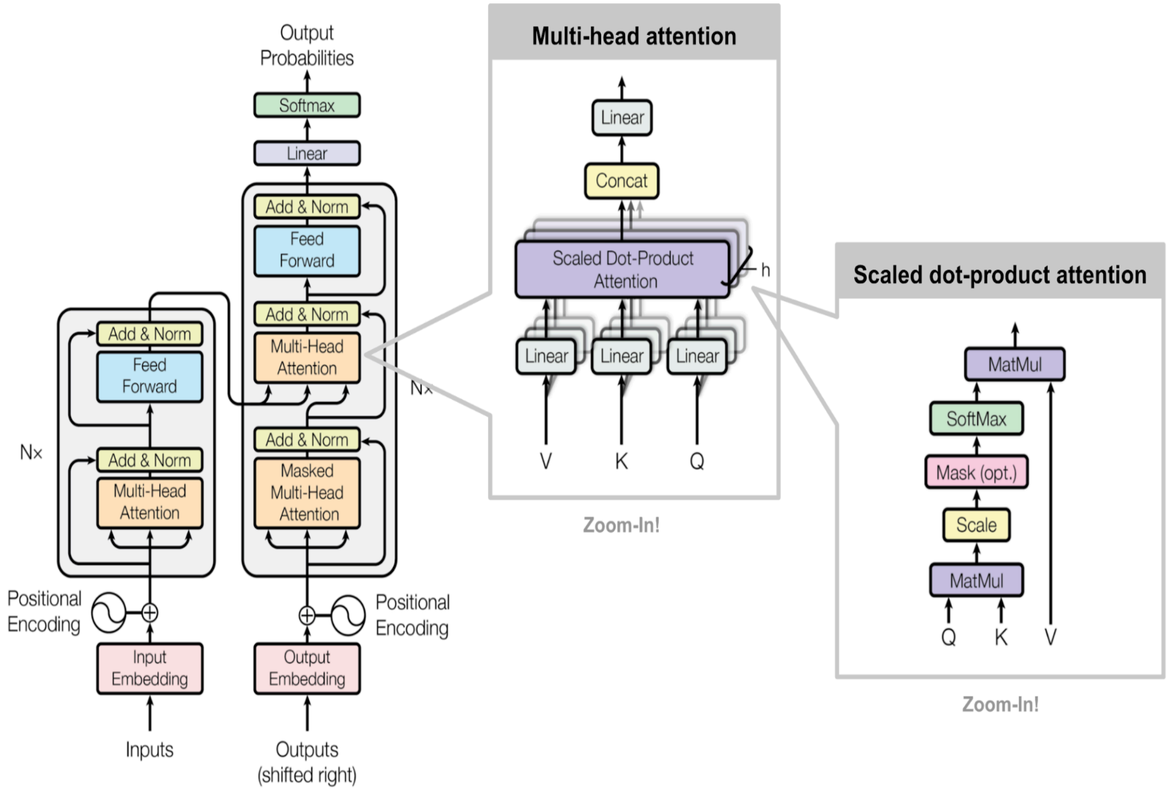
ビックカメラ
SHOPPING GO



학습 모델에 넣어주자

월렛탭 행동로그 (클릭/뷰)

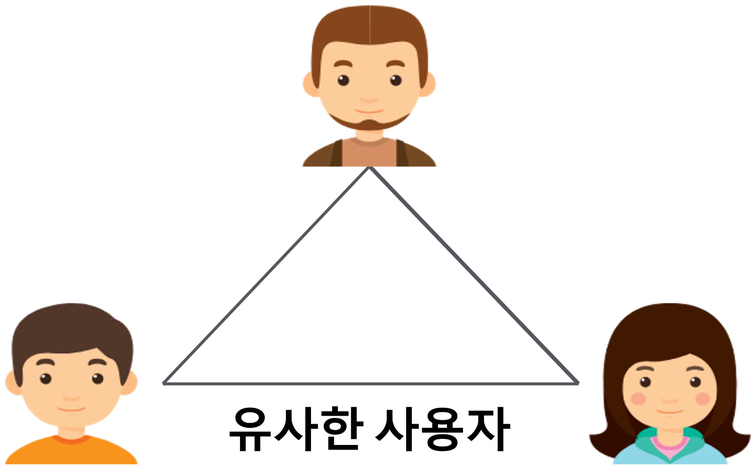




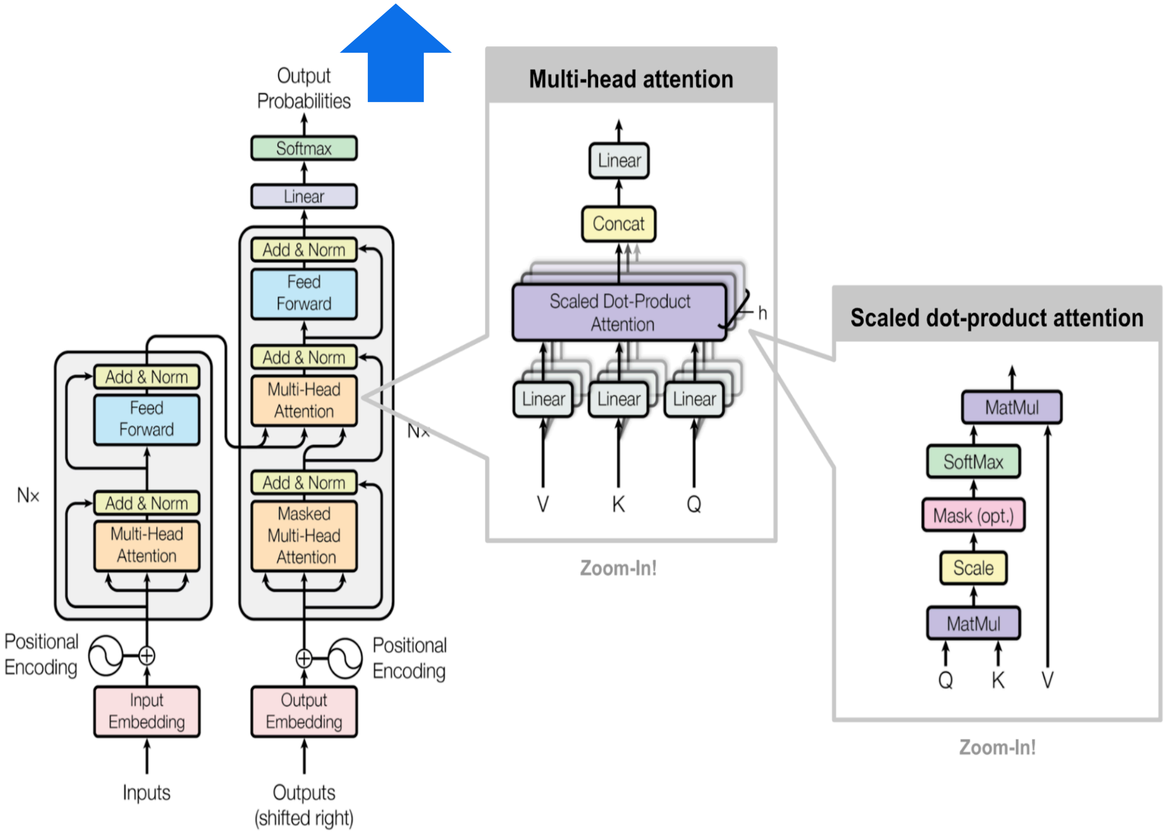
Transformer

학습 모델에 넣어주자

월렛탭 행동로그 (클릭/뷰)



쿠폰모듈 70%



Transformer

이렇게 끝나면 참 좋겠지만

난 일반적인 딥러닝 문제와는 달라~
편향된 데이터 다루기가 핵심이다

우리가 수집하는 로그에 비밀이 있으니

DEVIEW 2020

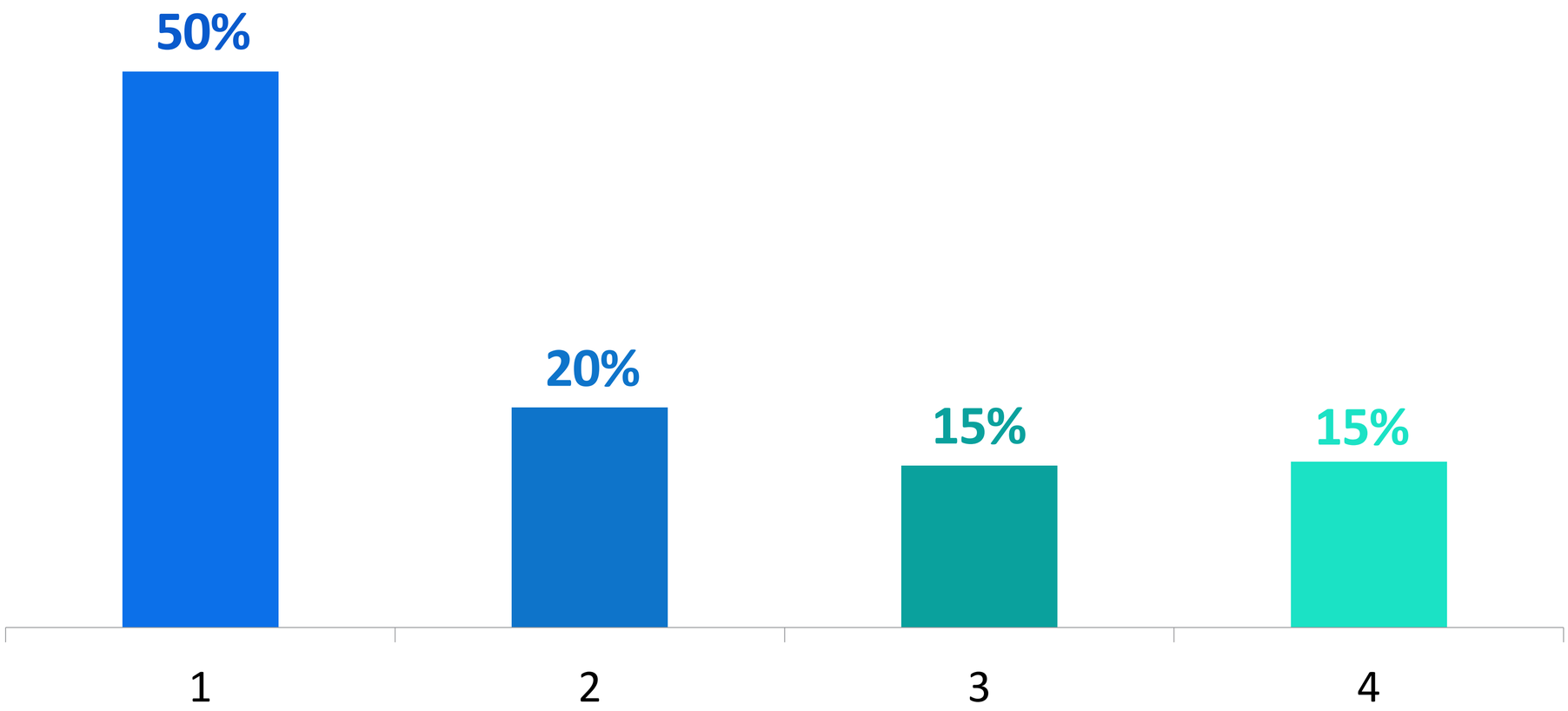
사용자가 주어지면

행동로그

클릭했던 모듈을 맞추도록 학습

클릭로그

컨텐츠 내용과 관련없이 월렛 사용자는 2번 중 1번 첫 번째 모듈을 클릭한다



노출 위치에도 빈익빈 부익부 현상이 생긴다

DEVIEW 2020

클릭 로그가 많음 — “다음에도 상단 배치”

클릭 로그가 적음 — “다음에도 하단 배치”

위치에 따른 이점을 보정하지 않으면
더 나은 모듈 순서를 놓치게 된다

방법 1: 랜덤 데이터로 학습하기

DEVIEW 2020

의도

- 일부 사용자들에게 모듈 순서를 랜덤으로 추천
- 해당 집단에서 구한 CTR은 Position Bias 약함

장점

- Position Bias가 없는 데이터
- 구현이 편함

단점

- 데이터 크기가 매우 작음
- 사용자에게 같은 순서로만 추천되어 기호성을 확인하기 어려움

방법 2: CTR Loss Weight

DEVIEW 2020

의도

- Loss를 CTR로 Weighting하여 높은 CTR을 갖는 모듈 추천 강화
- Position이 위로 갈수록 CTR이 낮아져서 debias 효과

장점

- Position Bias 완화
- Class Imbalance 문제 완화

단점

- CTR이 높은 모듈에 지나치게 편향됨
- Class Imbalance 문제에 취약

방법 3: CTR Mini-Batch

DEVIEW 2020

의도

› Loss Weighting 대신에 CTR에 비례하여 Mini-Batch를 구성

장점

› Position Bias 완화

› Class Imbalance 문제 완화

단점

› CTR이 높은 모듈에 지나치게 편향됨

› Class Imbalance 문제에 취약

방법 4: Explorative Labeling

DEVIEW 2020

의도

- 많은 사용자들이 한 번도 본적이 없는 모듈이 존재
- 안 본 모듈을 직접 보여주기 전에는 기호성 파악 불가능
- Thompson Sampling 기반으로 view가 적은 모듈을 일정 확률로 labeling

장점

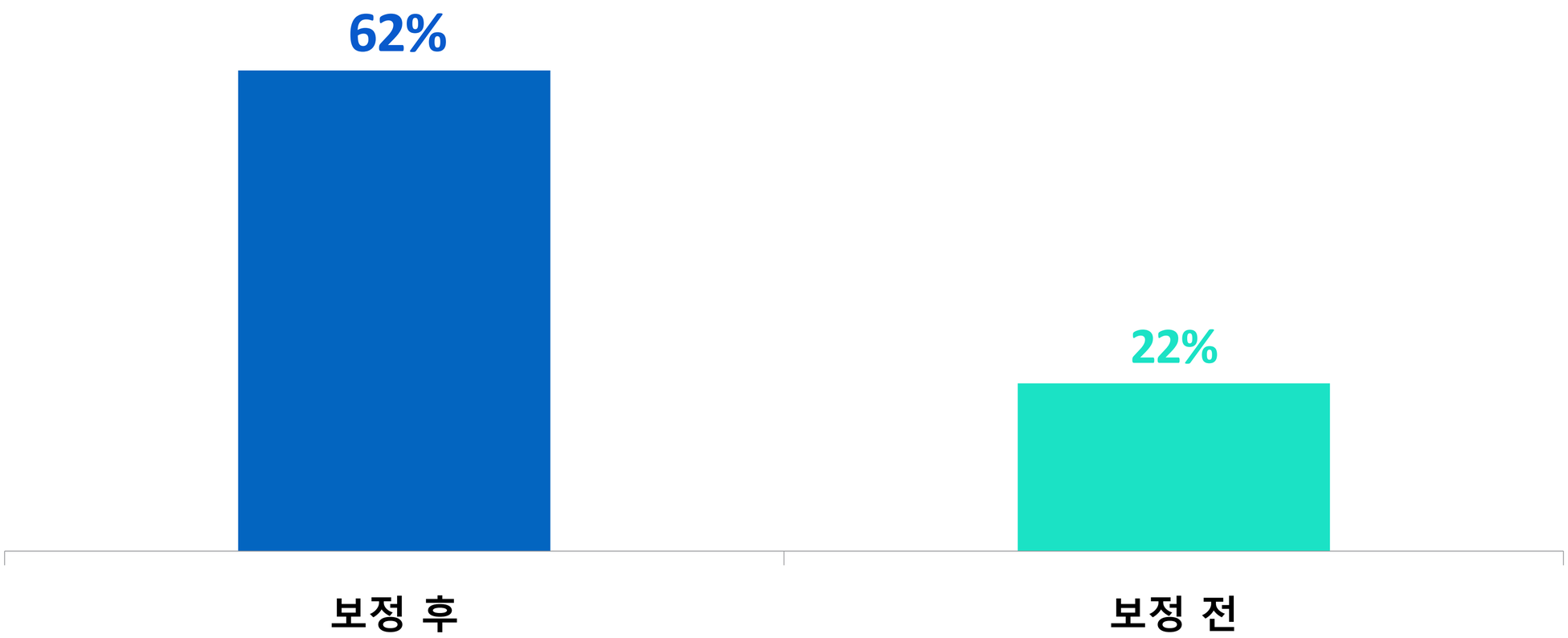
- Exploration 효과
- 진짜 기호성 파악

단점

- 관심 없는 모듈을 추천할 수 있음
- 탐색 시간이 오래 걸리고 중도 이탈 가능

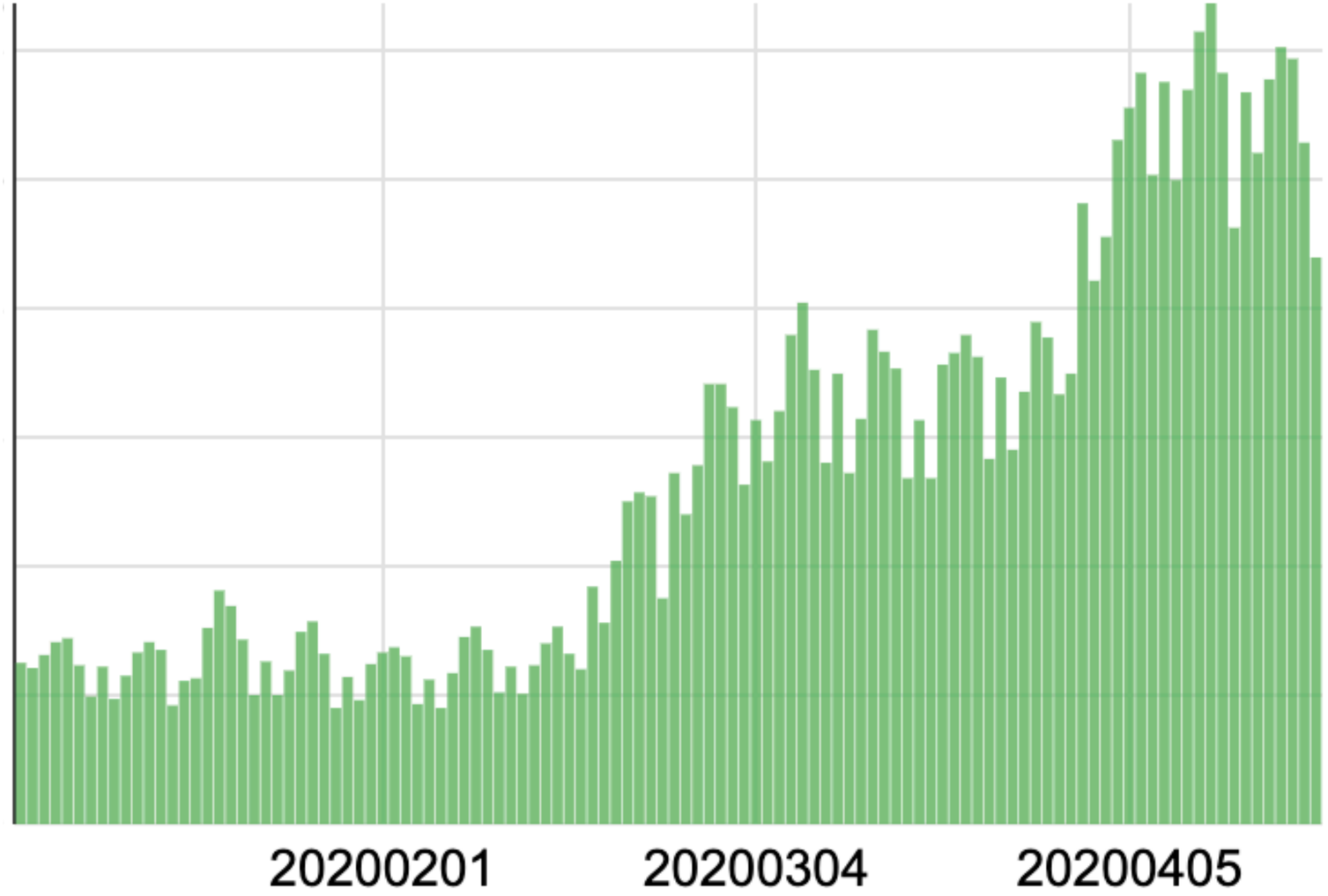
보정해주니 모듈 CTR이 더 오른다

DEVIEW 2020

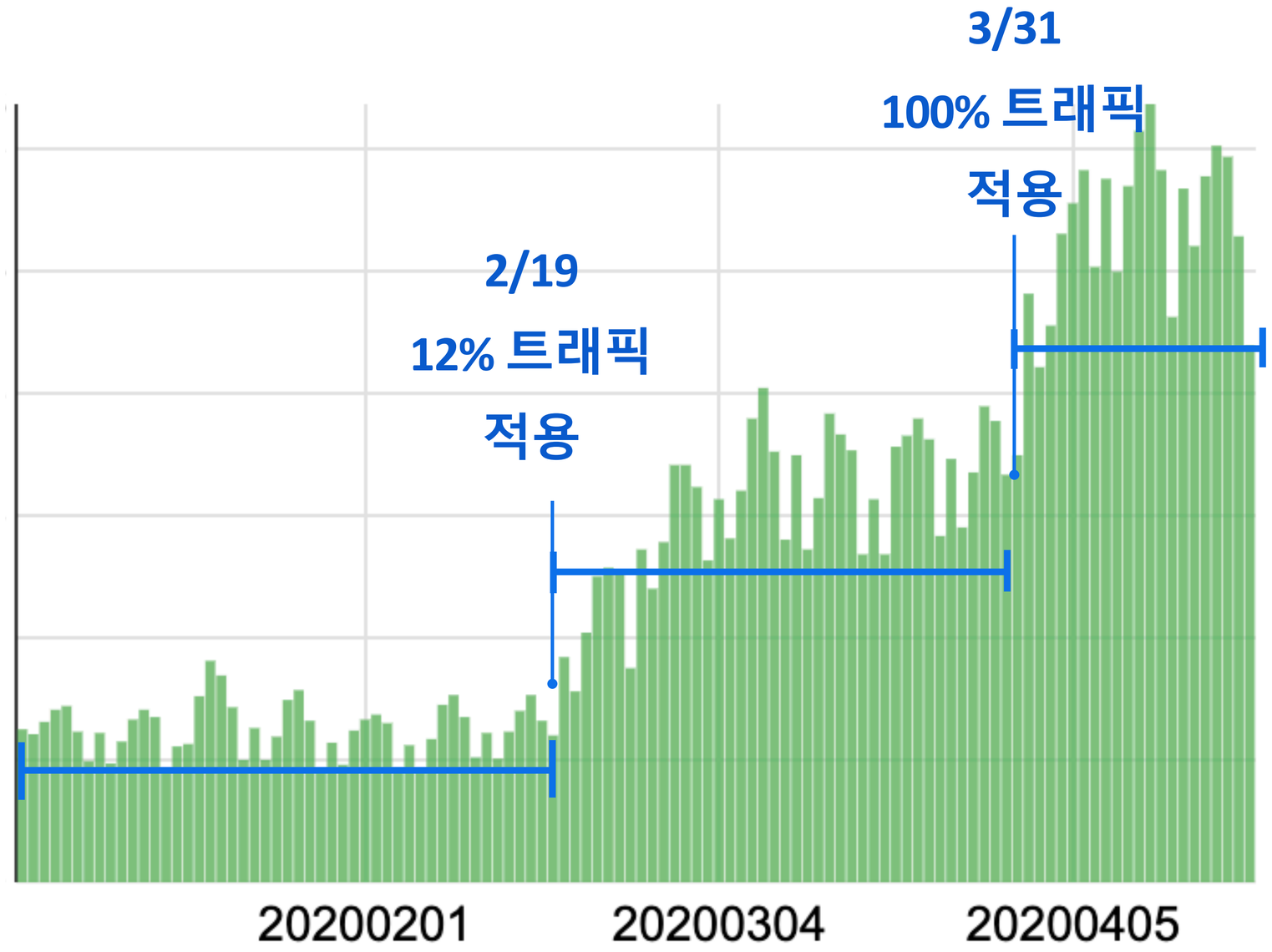


그래서 CTR, 얼마나 올랐나?

월렛모듈 CTR의 계단식 상승세



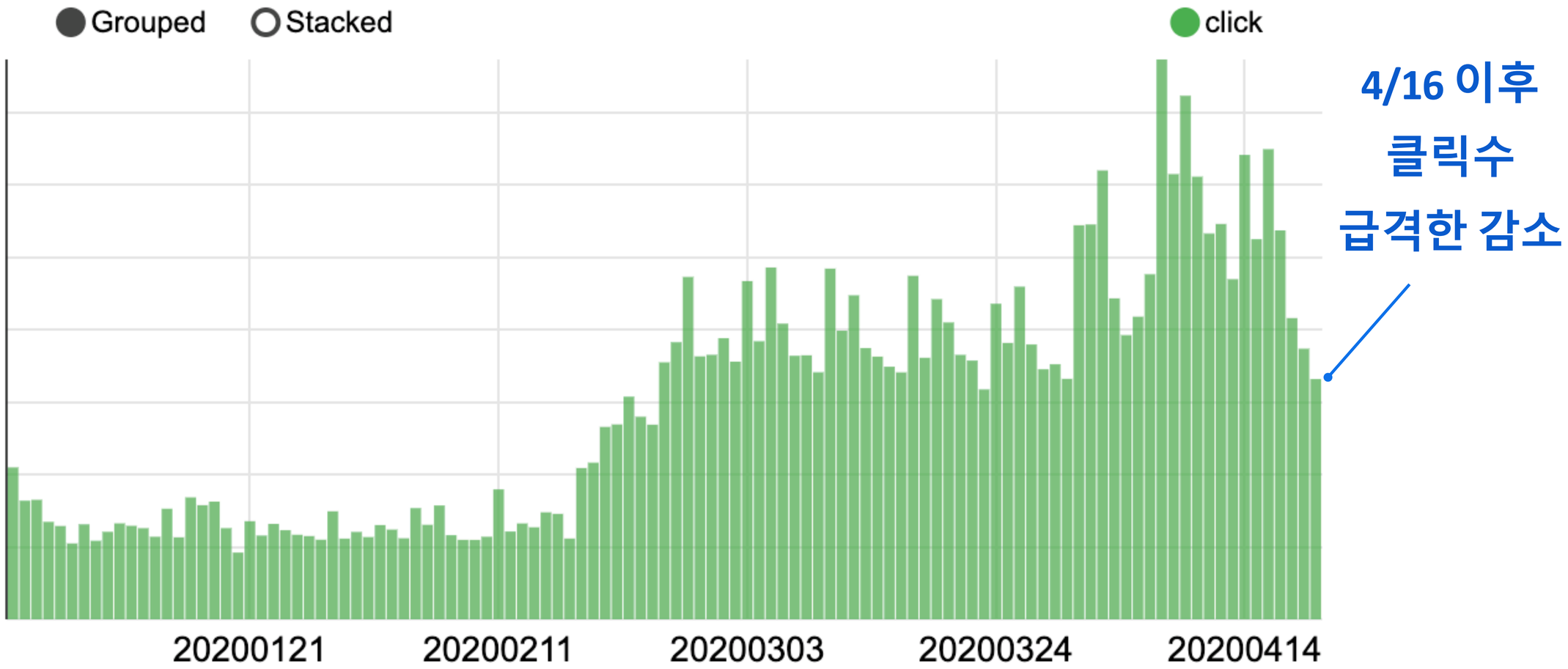
월렛모듈 CTR의 계단식 상승세



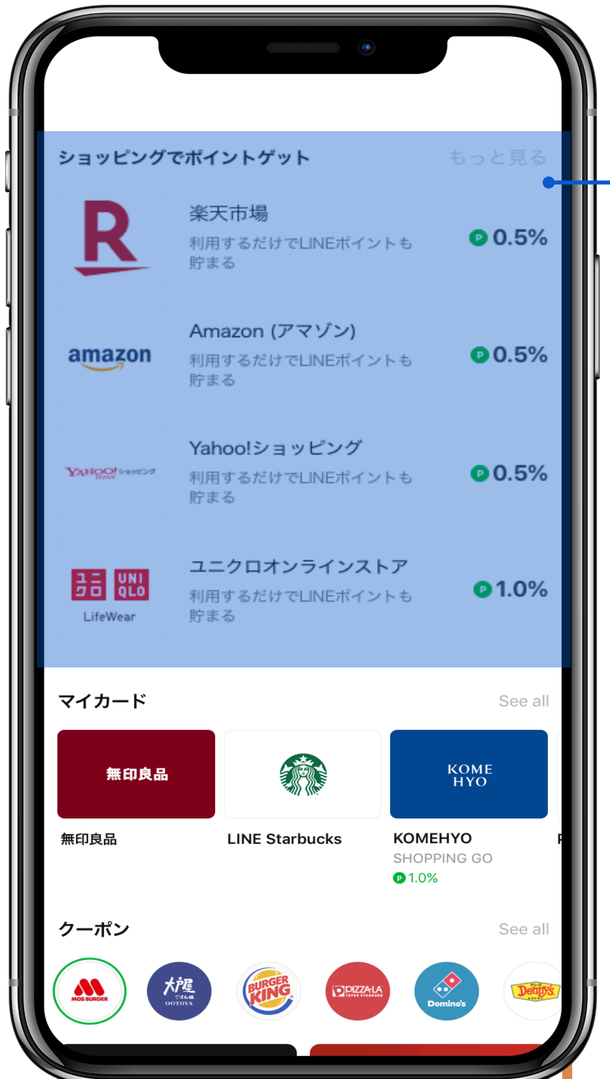
기획 vs 추천, 누가 이길까?

잘 나가던 월렛탭에 무슨 일이 생겼을까?

DEVIEW 2020

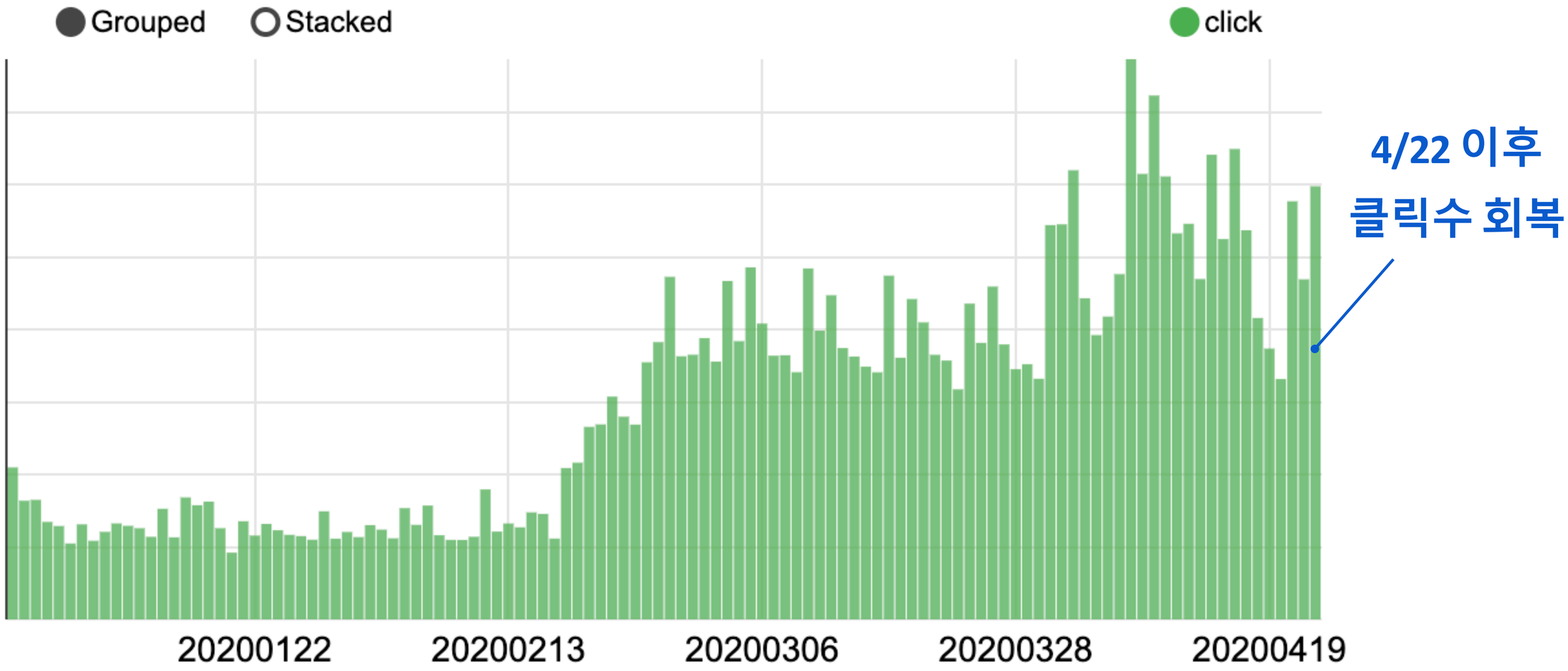


기획 모듈, 28% 트래픽의 월렛탭 최상단에 쇼핑모듈 노출

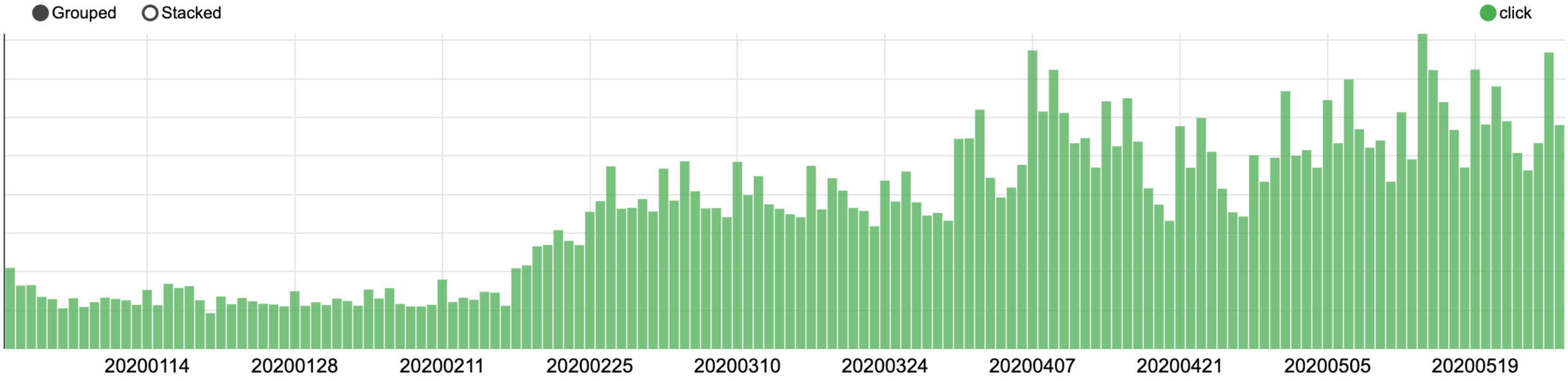


새로운 쇼핑모듈

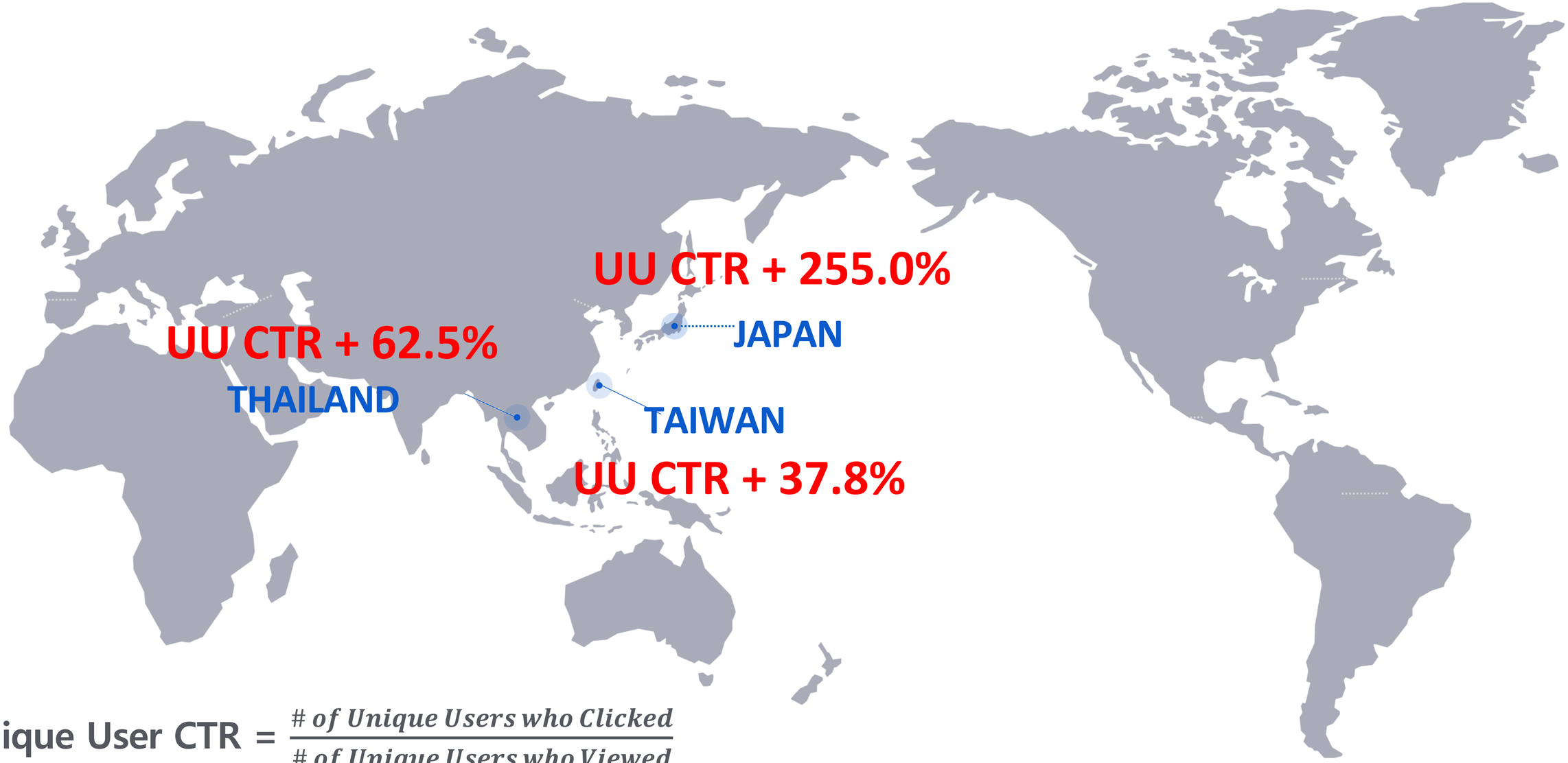
쇼핑모듈을 최상단에서 내리자 회복된다



5월까지 추세


Grouped
Stacked
click
20200114
20200128
20200211
20200225
20200310
20200324
20200407
20200421
20200505
20200519

일본, 대만, 태국



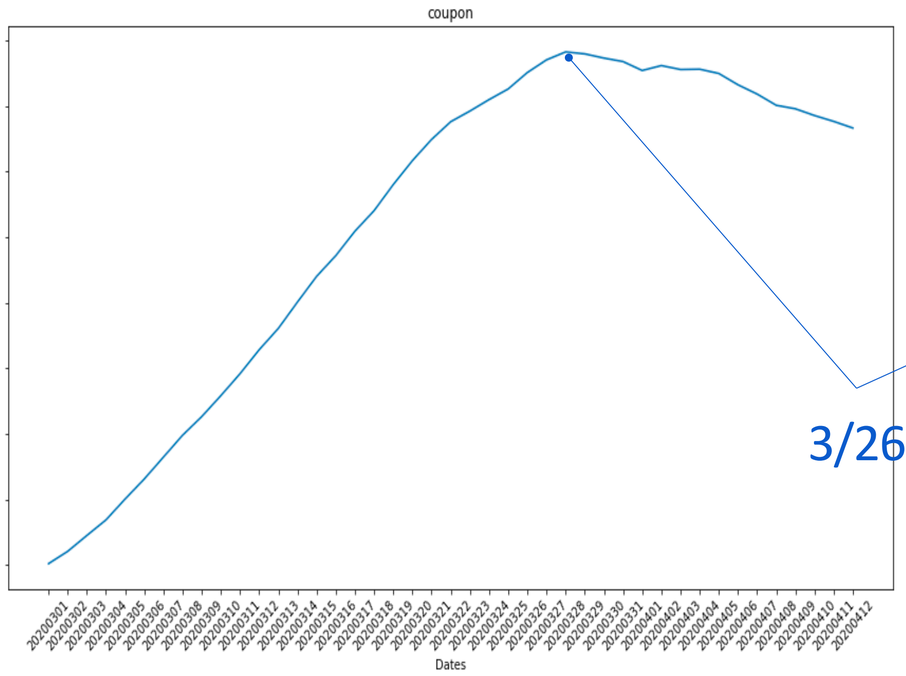
$$\text{Unique User CTR} = \frac{\#\ of\ Unique\ Users\ who\ Clicked}{\#\ of\ Unique\ Users\ who\ Viewed}$$

좋은 모듈, 나쁜 모듈?
그리고 아이템추천의 필요성

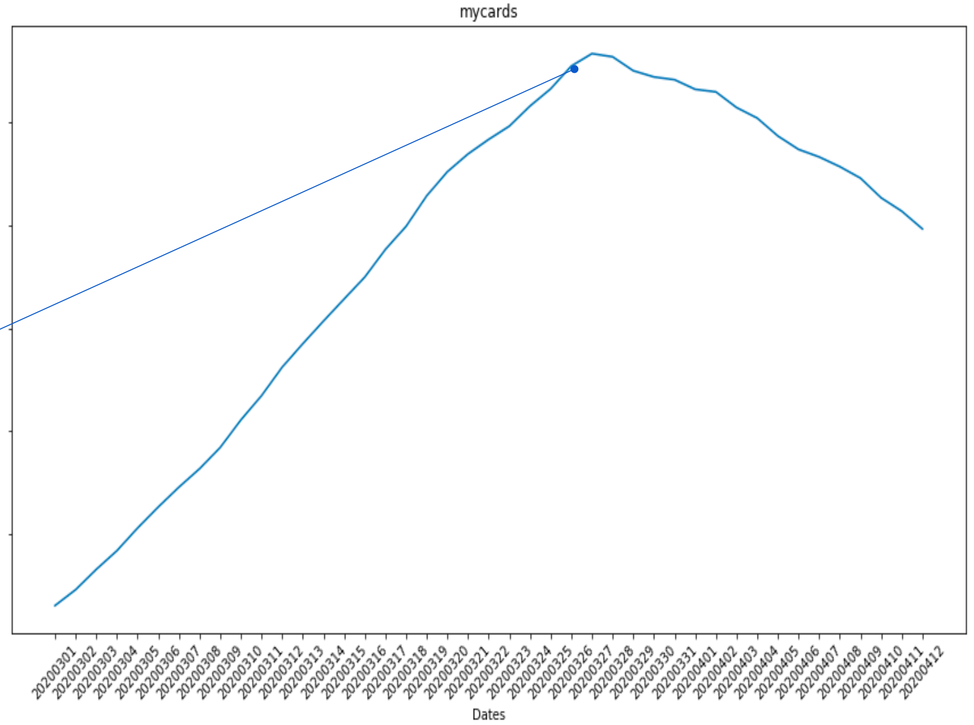
최근 일본 코로나의 영향으로

DEVIEW 2020

Outdoor 상품이 인기가 없다



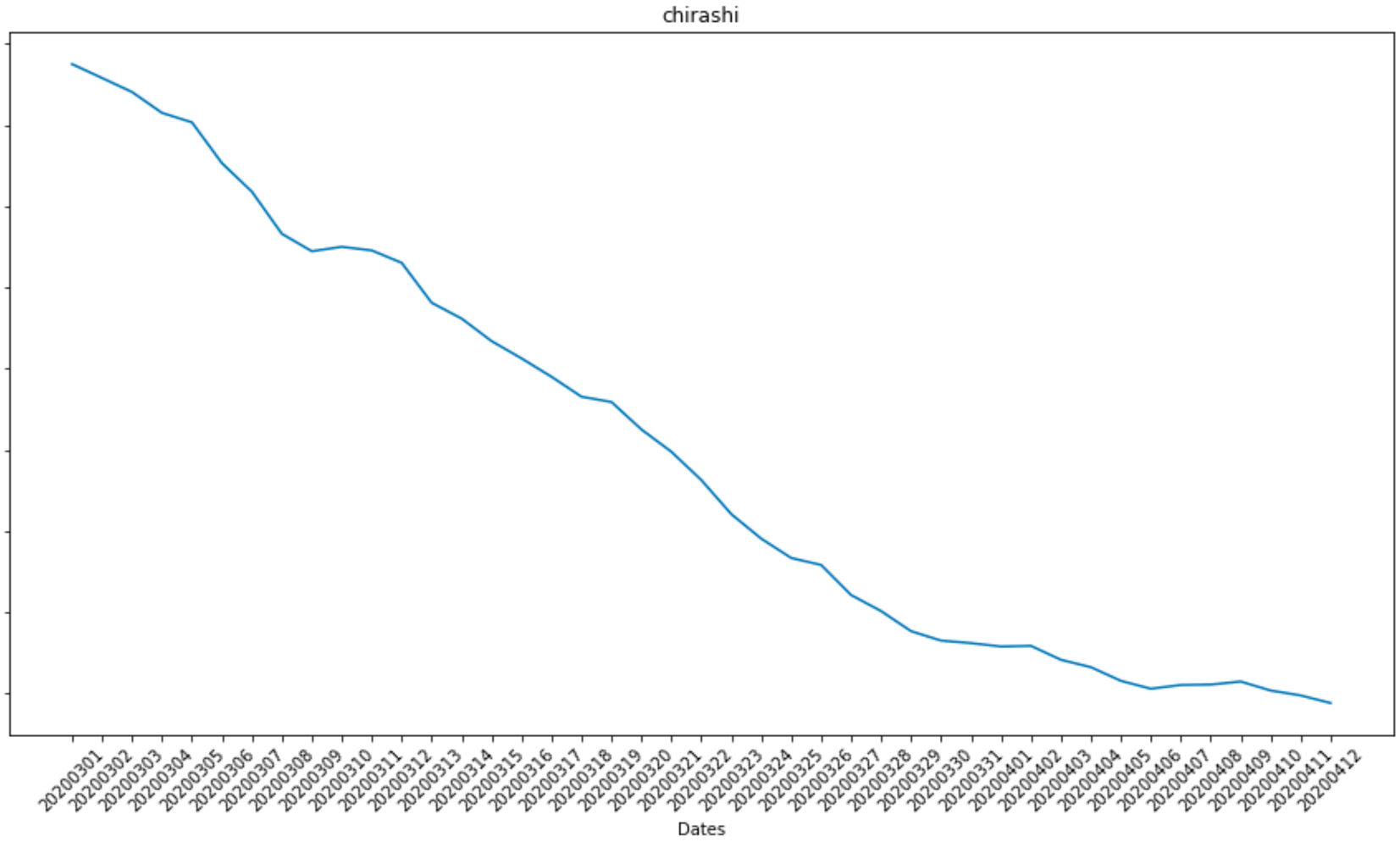
3/26



Coupon 사용자 수

MyCard 사용자 수

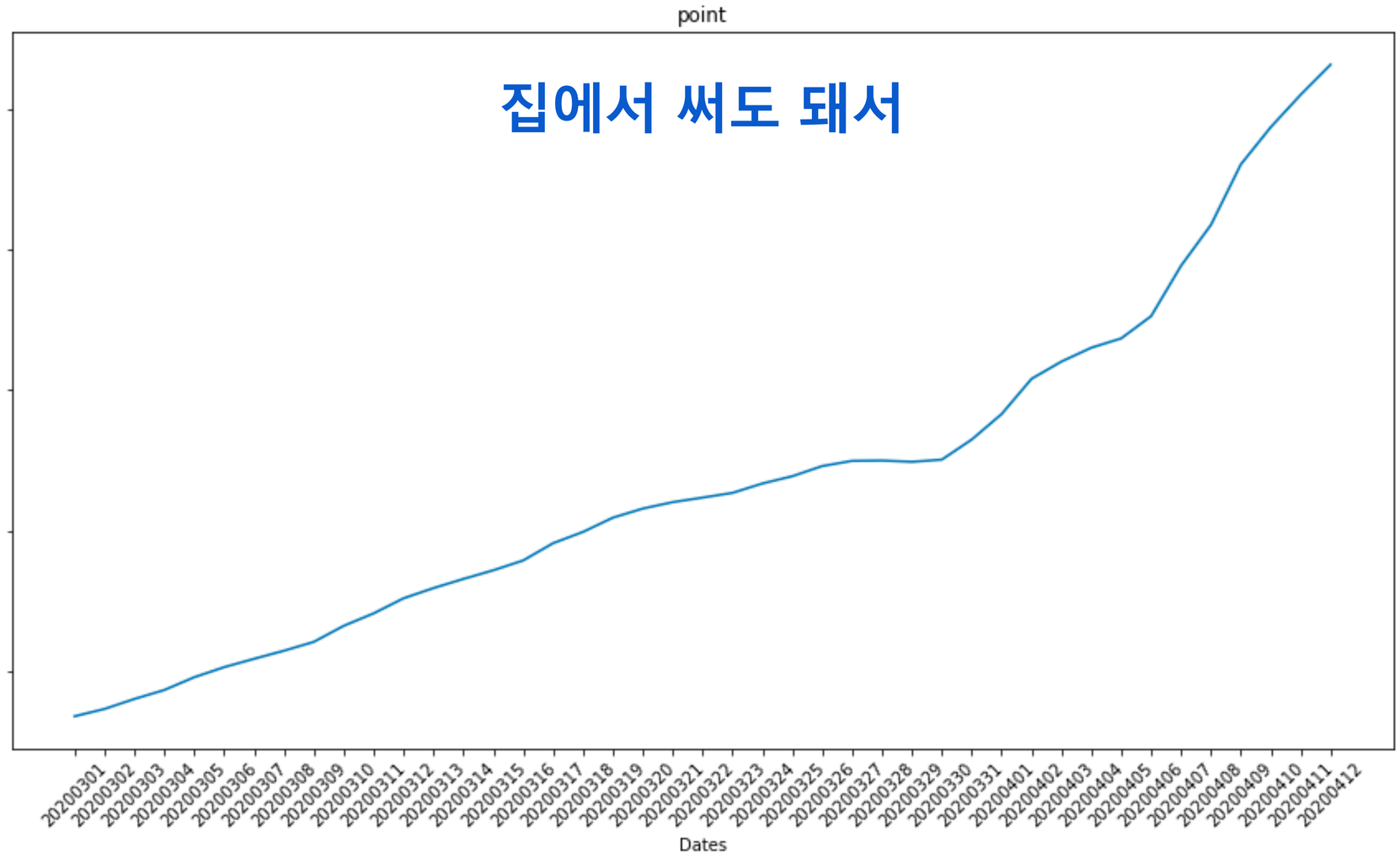
Chirashi 사용자는 계속 감소추세



Chirashi 사용자 수

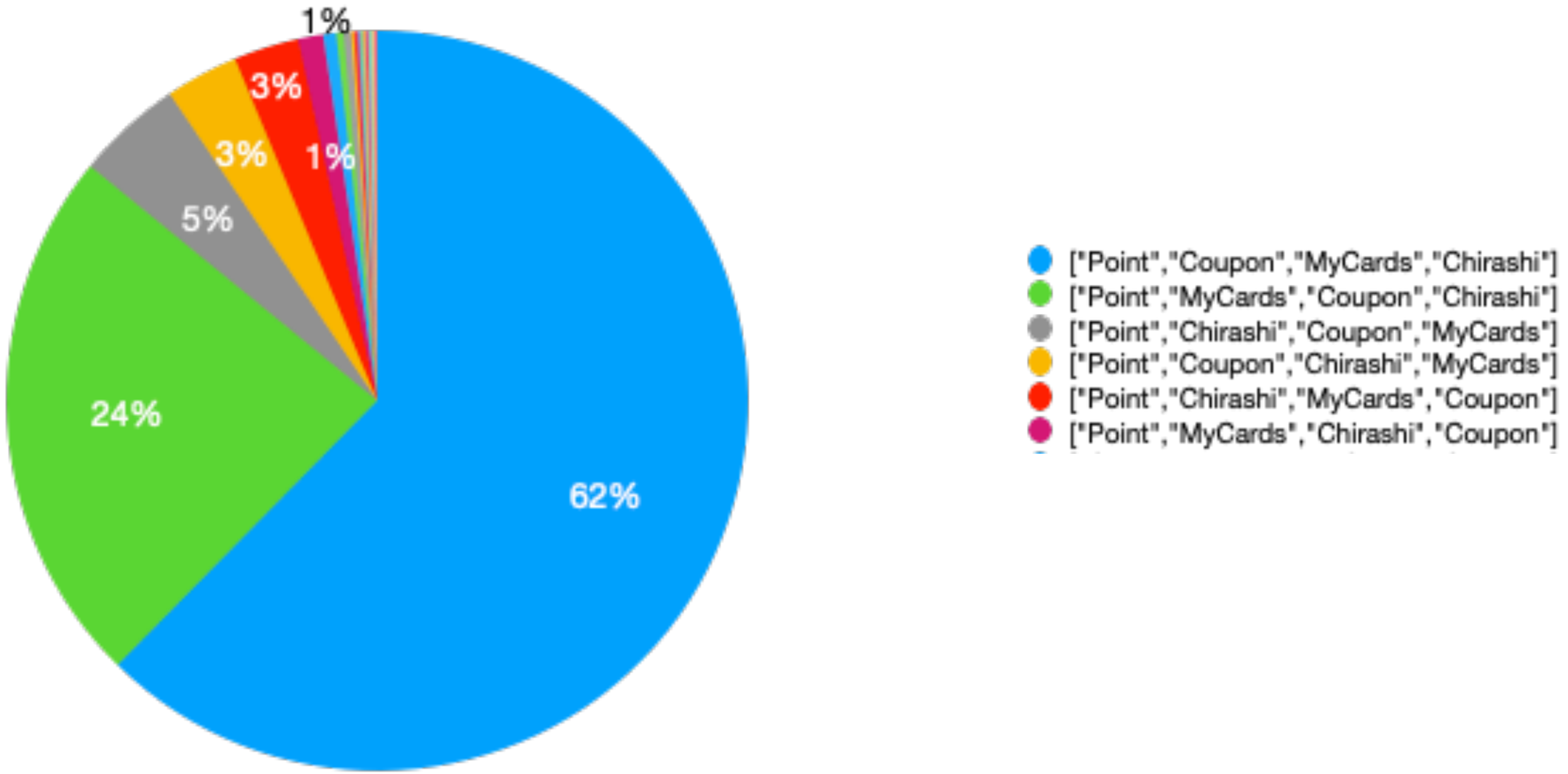
반면 Points 사용자는 증가한다

DEVIEW 2020



Points 사용자 수

Lesson: Points는 좋고 Chirashi는 나쁘다?



트래픽의 99%가 최상단에 Points를,

86%가 최하단에 Chirashi를 배치

아이템 추천이 같이 이뤄져야
다른 모듈도 활성화

3. Graph Neural Network for Recommendation

LINE Coupon

500+ items, 30+ brands

Coupon 추천의 시작

Android & iOS

2020.05 ~

개인화 추천

버거러버 영희

빵돌이 철수

사용 기록

추천

딥러닝 기반 추천 시스템

Click History

Demographics

. . .

User

X_u

Image

Category

. . .

Coupon

X_c

$$p_{uc} = \frac{1}{1 + \exp(-\mathbf{x}_u^T \mathbf{x}_c)}$$

Dot Product → CTR Prediction

0: Non-click
1: Click

Cold Start Problem & Diversity

New Users

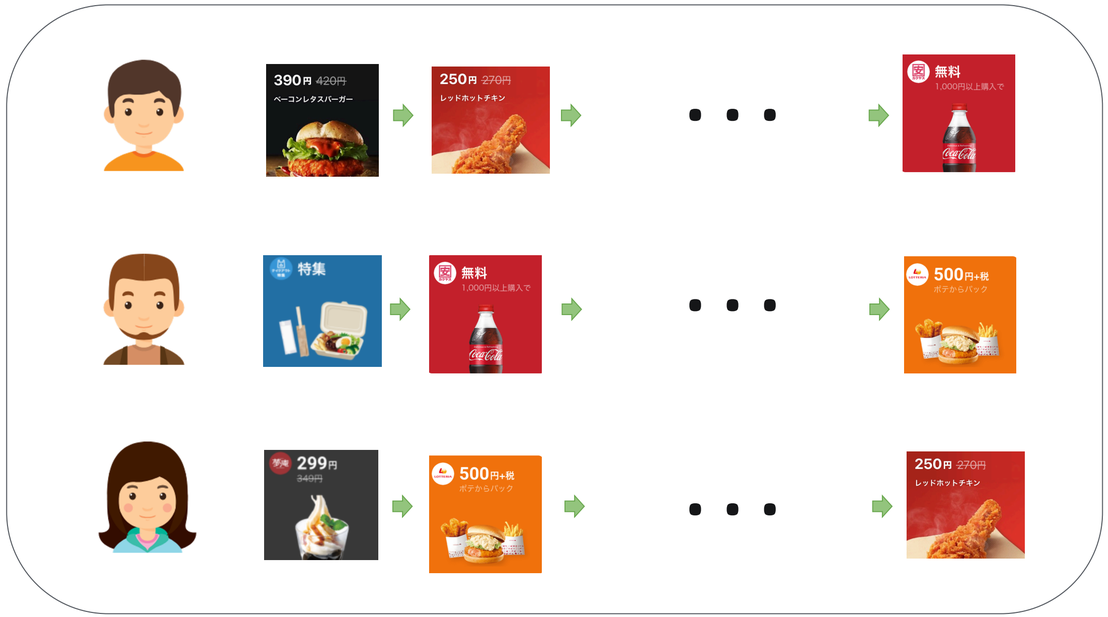
New Coupons

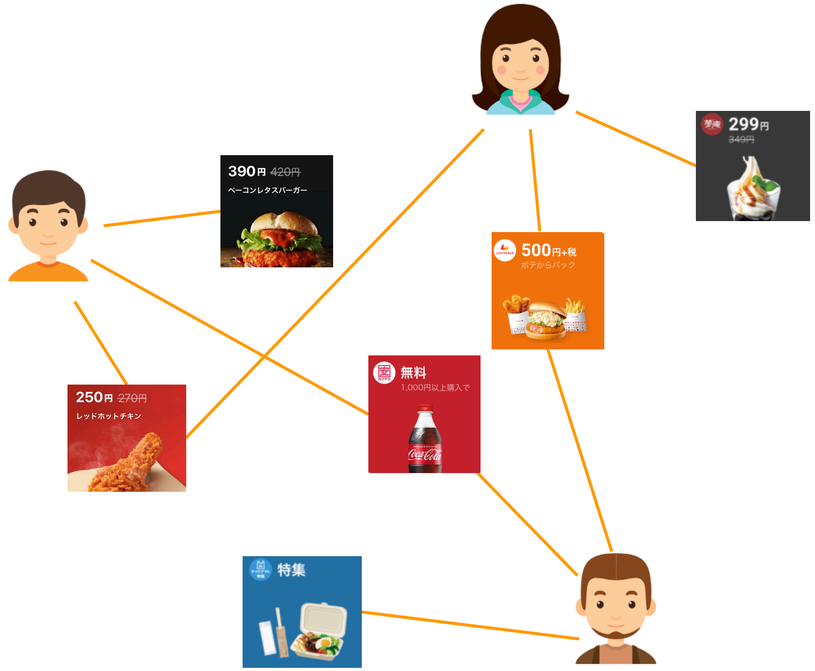
Our Goal: CTR과 Diversity, 두 마리 토끼 잡기

그래프로 모든 로그를 엮어라!

DEVIEW 2020

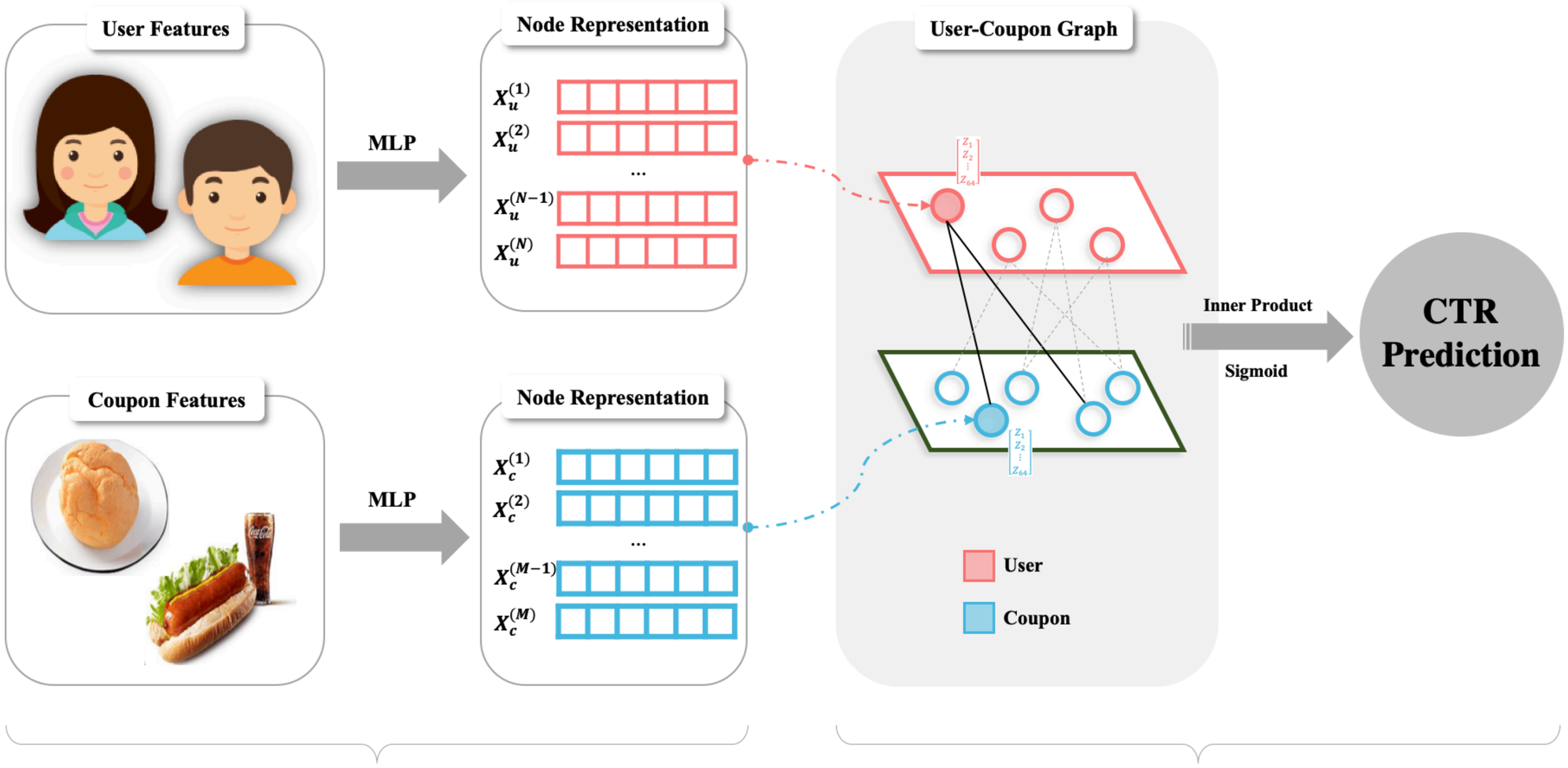
과거의 쿠폰사용로그



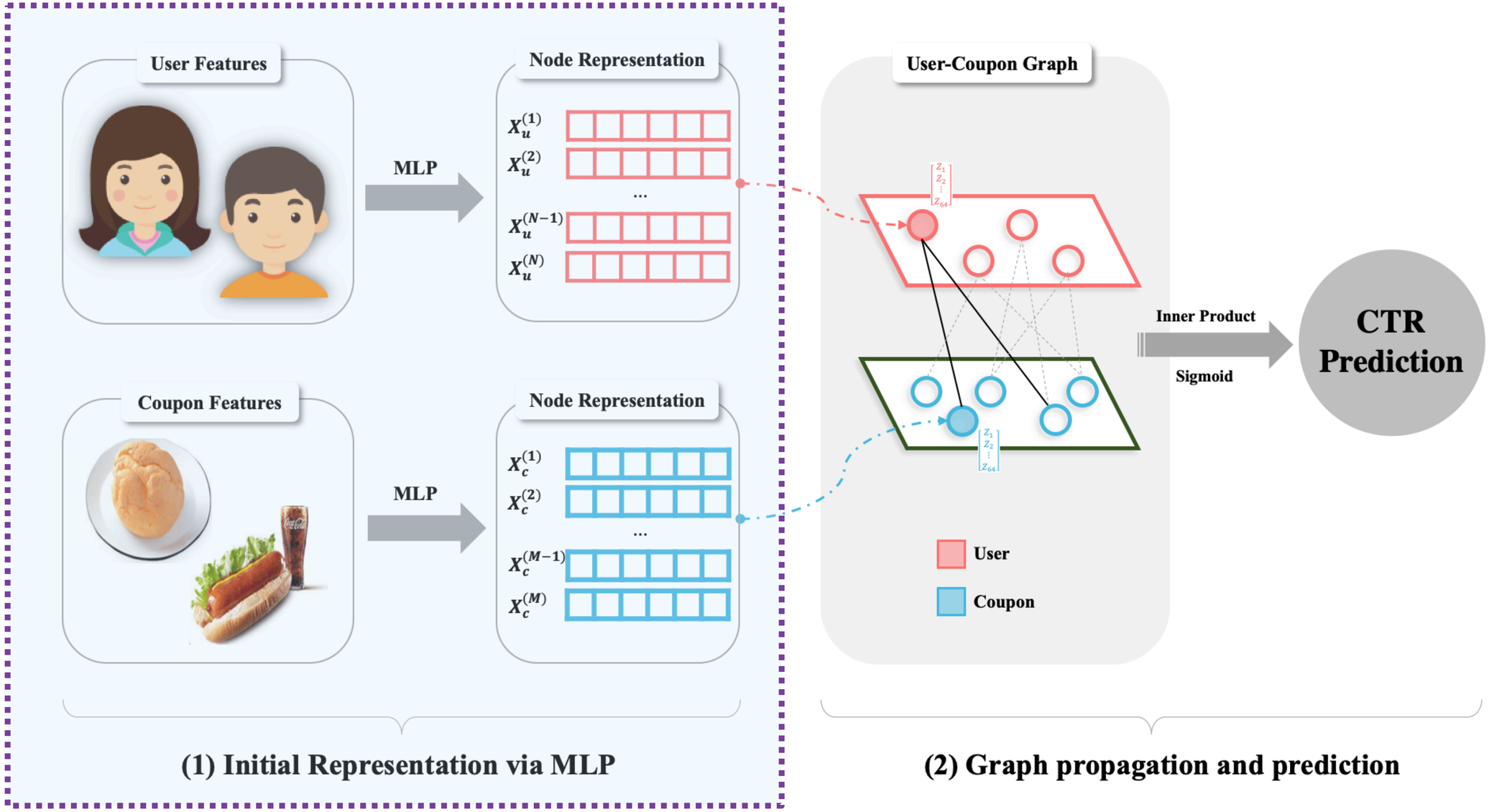


모든 것은 연결되어 있다...

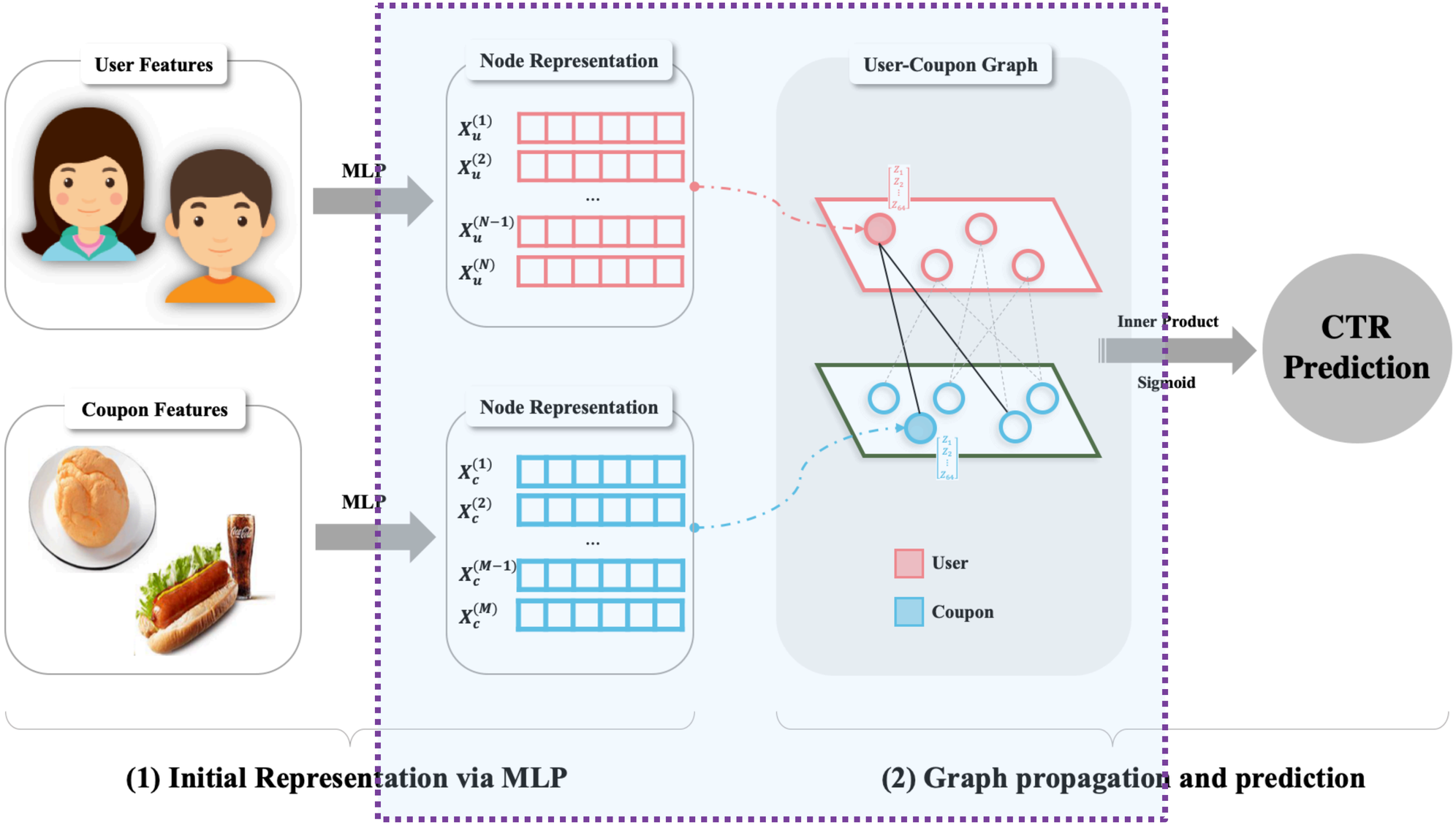
그래프 기반 추천 시스템


User Features
MLP
Node Representation
$X_u^{(1)}$
$X_u^{(2)}$
...
$X_u^{(N-1)}$
$X_u^{(N)}$
Coupon Features
MLP
Node Representation
$X_c^{(1)}$
$X_c^{(2)}$
...
$X_c^{(M-1)}$
$X_c^{(M)}$
User-Coupon Graph
Z_1 Z_2 ⋮ Z_{64}
User
Coupon
Inner Product
Sigmoid
CTR Prediction
(1) Initial Representation via MLP
(2) Graph propagation and prediction

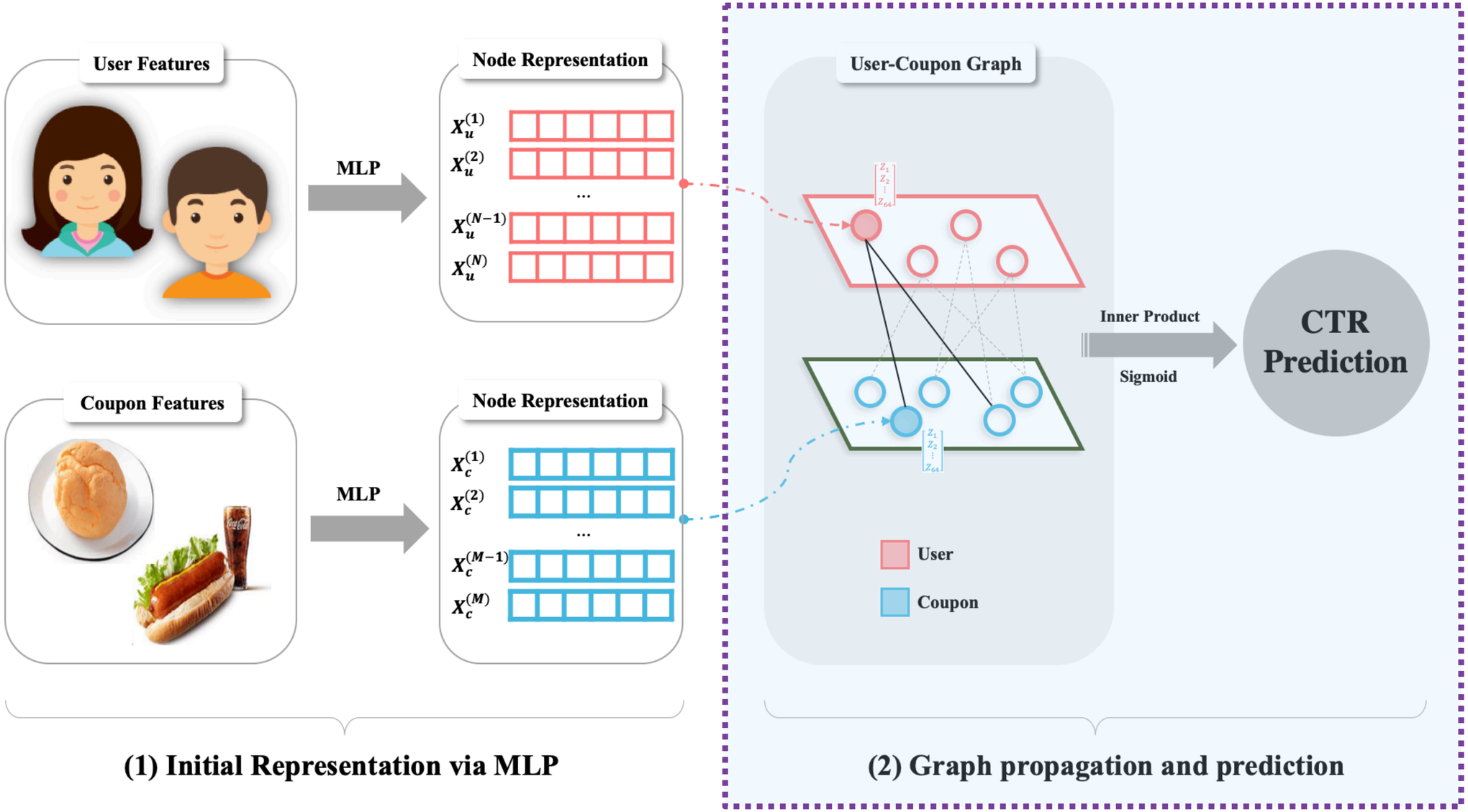
그래프 기반 추천 시스템


User Features
Node Representation
MLP
$X_u^{(1)}$
$X_u^{(2)}$
...
$X_u^{(N-1)}$
$X_u^{(N)}$
Coupon Features
Node Representation
MLP
$X_c^{(1)}$
$X_c^{(2)}$
...
$X_c^{(M-1)}$
$X_c^{(M)}$
User-Coupon Graph
Inner Product
Sigmoid
CTR Prediction
User
Coupon
(1) Initial Representation via MLP
(2) Graph propagation and prediction

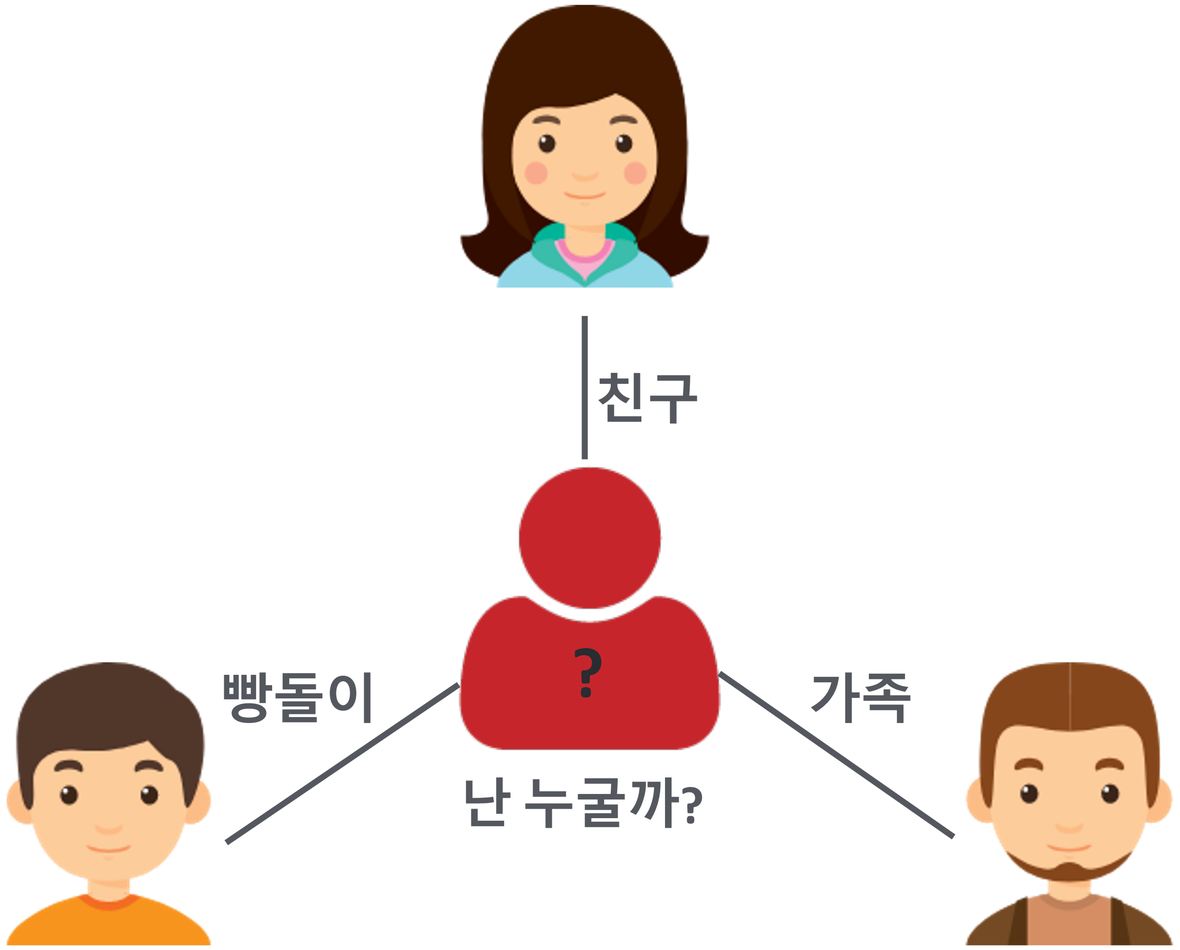
그래프 기반 추천 시스템


User Features
Coupon Features
MLP
MLP
Node Representation
Node Representation
$X_u^{(1)}$
$X_u^{(2)}$
...
$X_u^{(N-1)}$
$X_u^{(N)}$
$X_c^{(1)}$
$X_c^{(2)}$
...
$X_c^{(M-1)}$
$X_c^{(M)}$
User-Coupon Graph
User
Coupon
Inner Product
Sigmoid
CTR Prediction
(1) Initial Representation via MLP
(2) Graph propagation and prediction

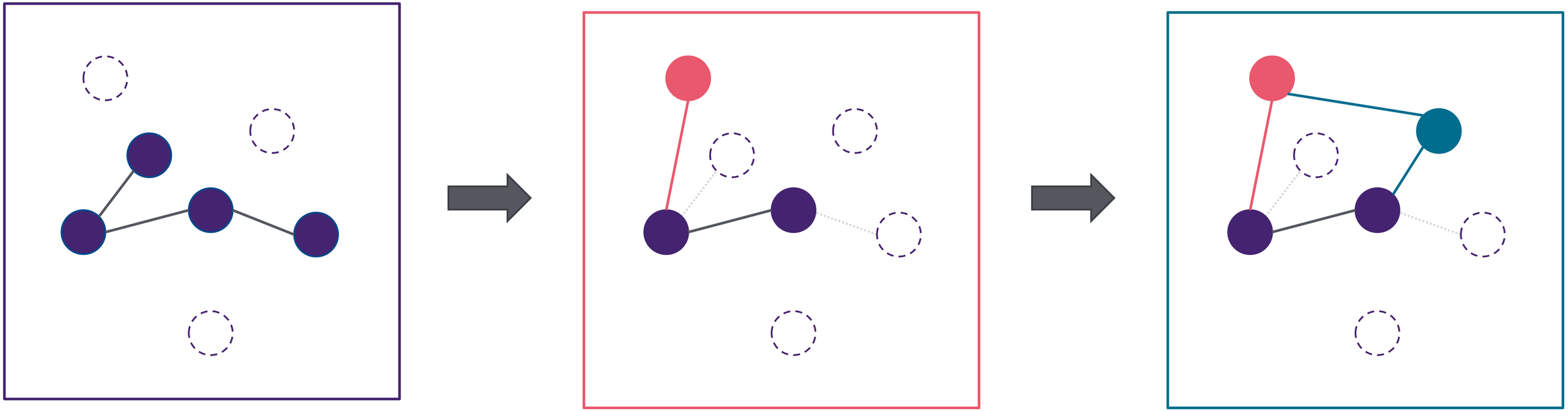
그래프 기반 추천 시스템


User Features
MLP
Node Representation
$X_u^{(1)}$
$X_u^{(2)}$
...
$X_u^{(N-1)}$
$X_u^{(N)}$
Coupon Features
MLP
Node Representation
$X_c^{(1)}$
$X_c^{(2)}$
...
$X_c^{(M-1)}$
$X_c^{(M)}$
User-Coupon Graph
Inner Product
Sigmoid
CTR Prediction
User
Coupon
(1) Initial Representation via MLP
(2) Graph propagation and prediction

“연결”을 학습합니다.



NEW 사용자/쿠폰에
더 효과적으로 적응할 수 있습니다.



Day 1

Day 2

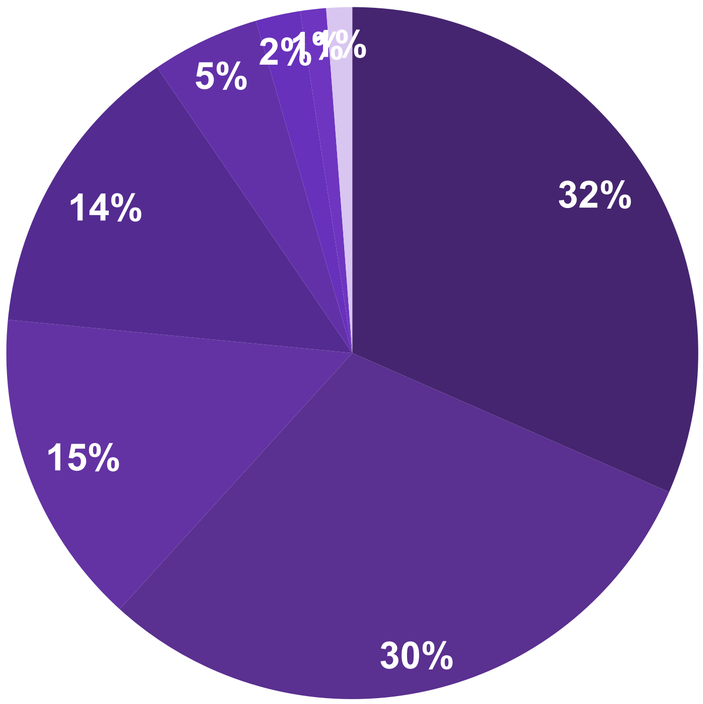
Day 3

그럼, 추천 성능은?

추천 결과에서 과거 인기도 비율

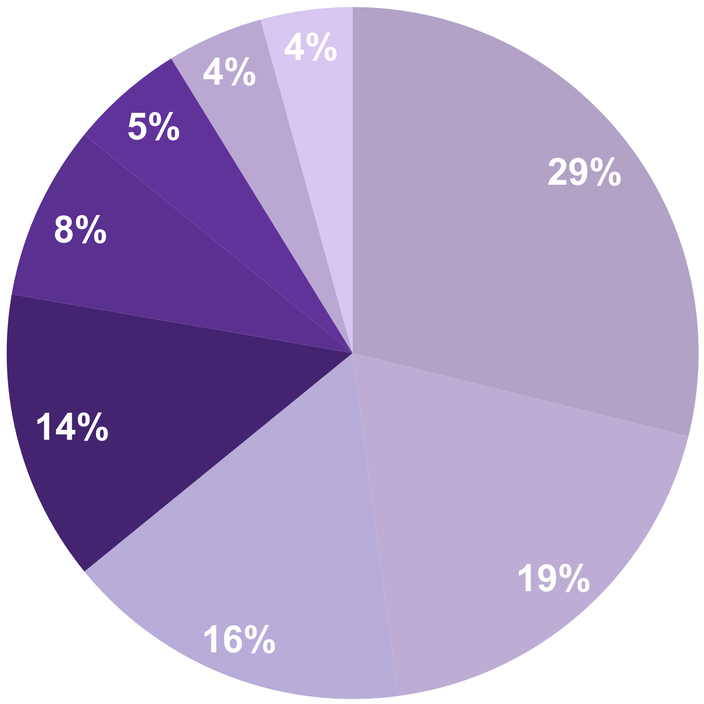
DEVIEW 2020

과거 2주간 인기도



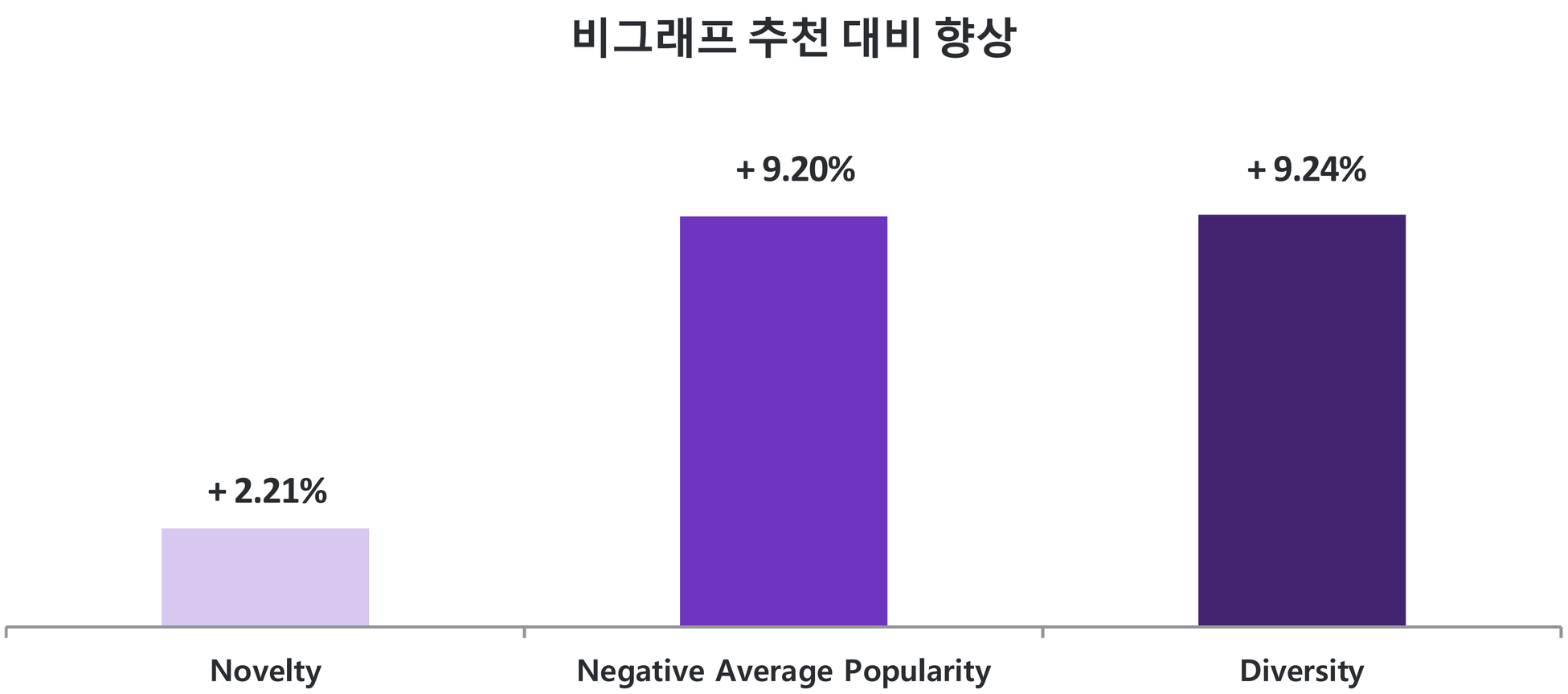
비그래프 모델

추천 비율

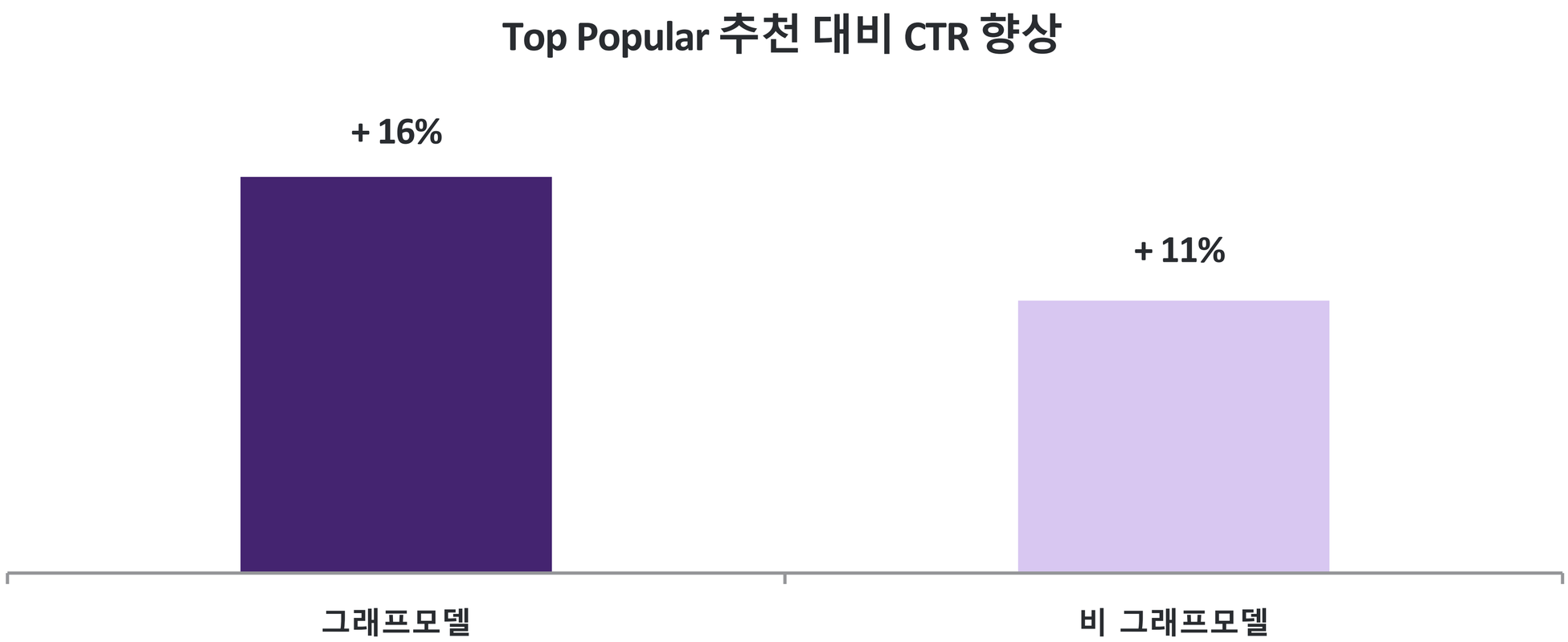


그래프 모델

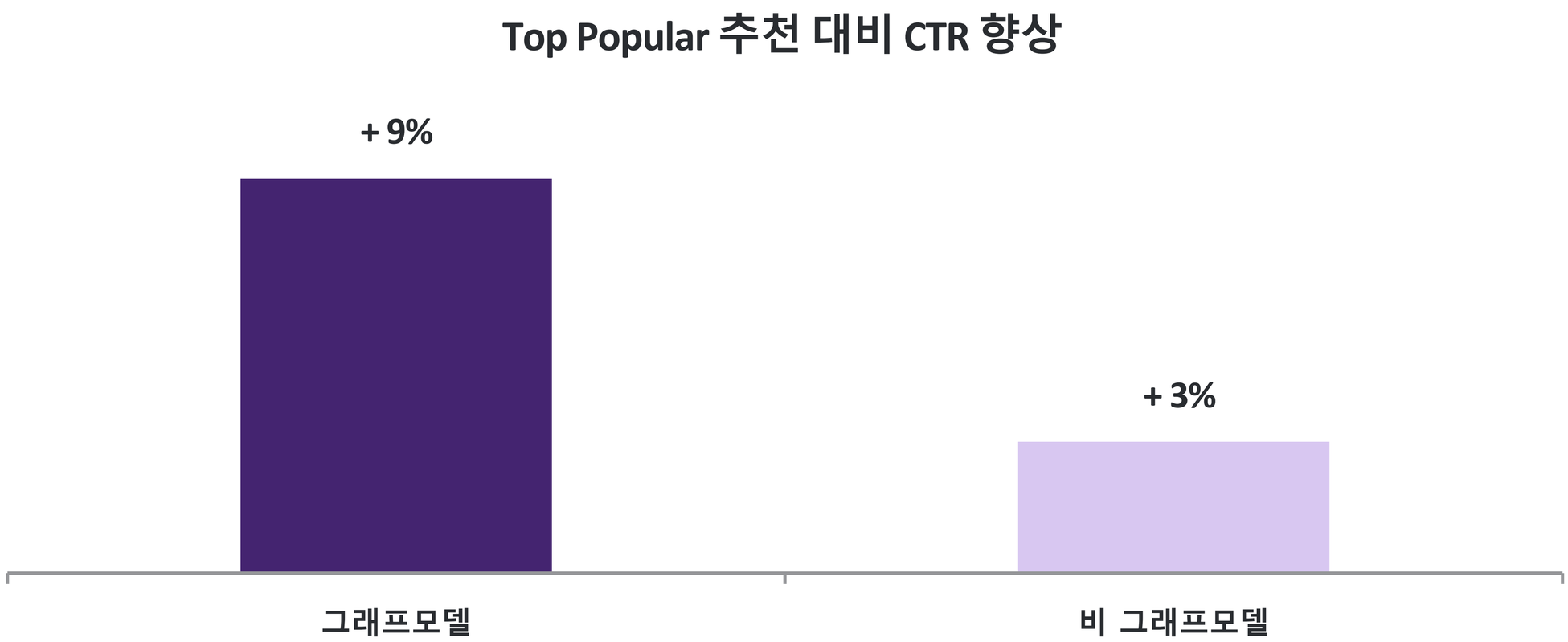
추천 성능 – Diversity



추천 성능 – CTR



Cold User 추천 성능 – CTR



Summary

- GNN 기반 개인화 추천 시스템
- Cold Start 문제를 해결 → CTR과 Diversity

[1] Tripartite Heterogeneous Graph Propagation for Large-scale Social Recommendation, RecSys 2019 Late-Breaking Result

[2] Graphs, Entities, and Step Mixture, ICML 2020 Workshop on Graph Representation Learning and Beyond

[3] Hop Sampling: A Simple Regularized Graph Learning for Non-Stationary Environments, KDD 2020 Workshop on Mining and Learning With Graphs

[4] Multi-Manifold Learning for Large-scale Targeted Advertising System, KDD 2020 Workshop, AdKDD

[5] div2vec: Diversity-Emphasized Node Embedding, RecSys 2020 Workshop on the Impact of Recommender Systems

4. Counterfactual Evaluation

앞선 방법들로 잘 동작하는 모델을 만들었다

우리는 모델을 과거 데이터를 이용해서 학습하고 튜닝했다.

하지만,

- 이 모델이 온라인에서 잘 동작할 지 어떻게 알지?
- 만약 온라인에서 잘 하지 못한다면 어떻게 하지?

온라인 – 오프라인 편차

온라인 플랫폼을 서비스하면 필연적으로 생기는 현상.
오프라인 테스트에서 좋았던 모델이 온라인에 나가면 힘을 쓰지 못한다.

두 가지 해결책이 있다.

1. 온라인 – 오프라인 테스트의 싱크를 맞추자
2. 온라인에서 직접 좋은 모델을 선택하자

온라인 – 오프라인 테스트 싱크를 맞추자

온라인 지표에 영향을 주는 요소를 알아내서, 오프라인과 온라인의 싱크를 맞추자

오프라인 테스트로 온라인 성능 최적화하기

DEVIEW 2020

온라인 CTR에 영향을 줄 만한 (cause) 요소를 골라서 테스트해보자

모델	온라인	기존 metric	IPS1	IPS2	IPS3	IPS4	IPS5	CAB
model1	1	3	4	3	4	2	2	1
model2	2	4	5	4	5	3	3	3
model3	3	1	3	5	3	5	5	5
model4	4	2	2	1	2	1	1	2
model5	5	5	1	2	1	4	4	4
Kendall Tau Correlation	N/A	0.2	-0.8	-0.2	-0.8	0.2	0.2	**0.4**

오프라인 테스트로 온라인 성능 최적화하기

이제 보다 온라인 환경에 가까운 모델 평가를 할 수 있다.

- 이를 이용해서 hyperparameter 최적화 등의 튜닝도 가능하다

이렇게 해도 여전히 간극은 있고, 온라인 자체에서 모델 테스트가 필요하다.

효율적으로 온라인에서 모델을 테스트해보자

DEVIEW 2020

추천 시스템 2.0 (A/B 테스트 방식)

- 하나의 모델을A/B 테스트를 통해 기존 모델보다 좋은 지 테스트
- 하나의 모델을 점검하는 데 오래 걸림

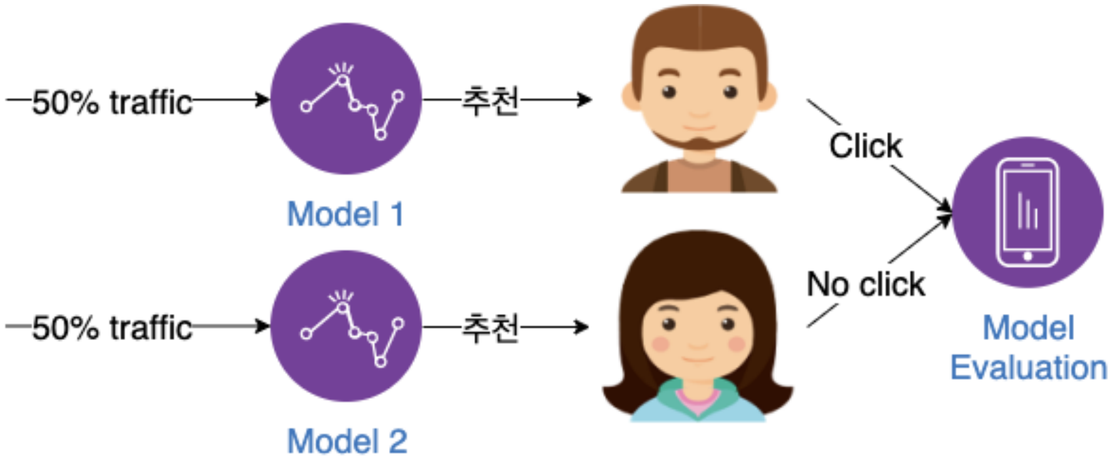
추천 시스템 3.0 (A/B/n 모델 서빙 방식)

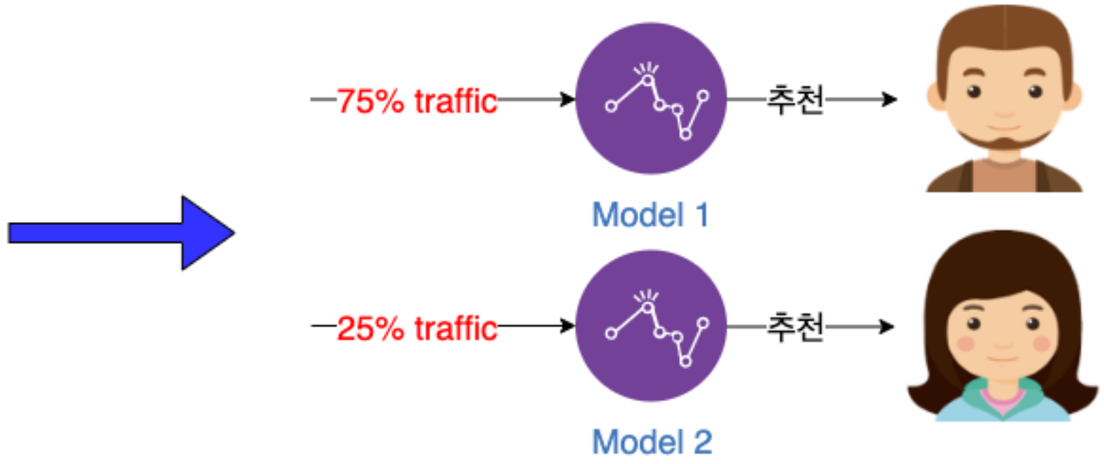
- n 개의 모델을 동시에 서빙
- 다양한 최신 모델 테스트
- 그리고, 가장 좋은 모델에게 트래픽을 몰아줘서 전체 시스템의 성능도 향상한다

온라인 – 오프라인 테스트를 합쳐보자

DEVIEW 2020

온라인 모델 테스트 (a.k.a 1부리그)

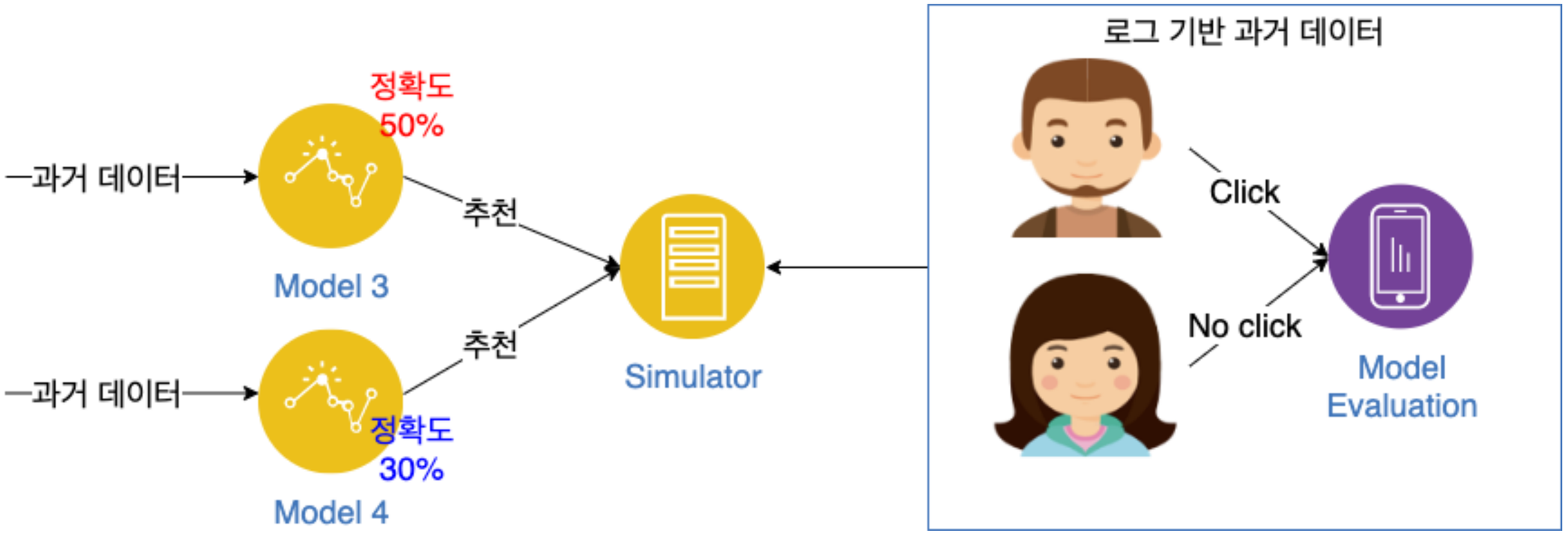




온라인 – 오프라인 테스트를 합쳐보자

DEVIEW 2020

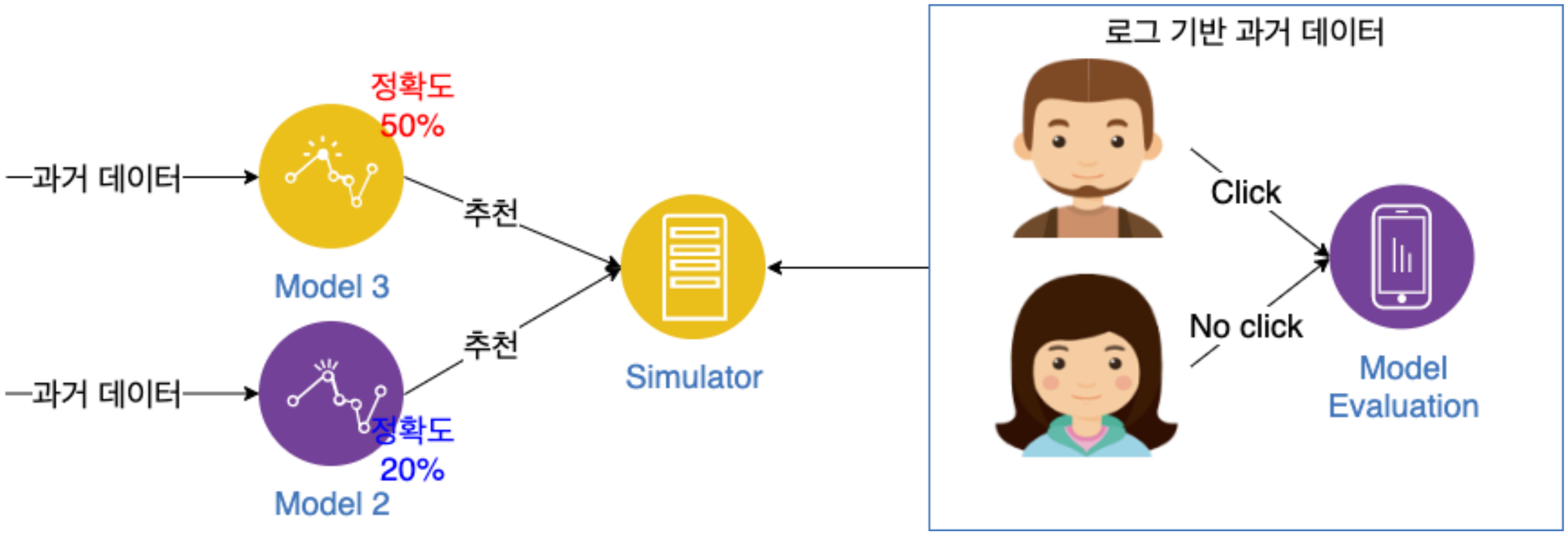
오프라인 모델 테스트 (a.k.a 2부 리그)



온라인 – 오프라인 테스트를 합쳐보자

DEVIEW 2020

모델 업데이트 (a.k.a 승격강등전)



마지막으로

온라인 성능을 기반으로 모델 선택 및 hyperparameter 최적화도 할 수 있다면?

변화하는 환경에 스스로 적응하는 모델 완성

여기까지 오면

DEVIEW 2020

모델러가 적당한 구조의 모델을 작성해서 주면, 시스템이

- Causality를 고려한 metric에 맞게 모델을 튜닝해주고,
- 시뮬레이션에서 좋은 모델을 온라인에 서빙해주며,
- 온라인 결과에 맞도록 모델을 다시 튜닝시켜주고,
- 더 좋은 모델에게 트래픽을 몰아줘서 시스템 성능을

높인다.

여기까지 오면

DEVIEW 2020

- Bias reduction을 통해 사용자가 진짜 좋아할 것을 추천하고,
- 그래프를 통해 새로운 사용자 / 아이템에도 강인하며,
- 오프라인에서 온라인성능을 최적화할 뿐만 아니라 스스로 변화하는 환경에 적응하며 운영 코스트를 줄인

추천시스템 3.0 완성!

Q & A

Thank You