# RS: data preprocess and recommend

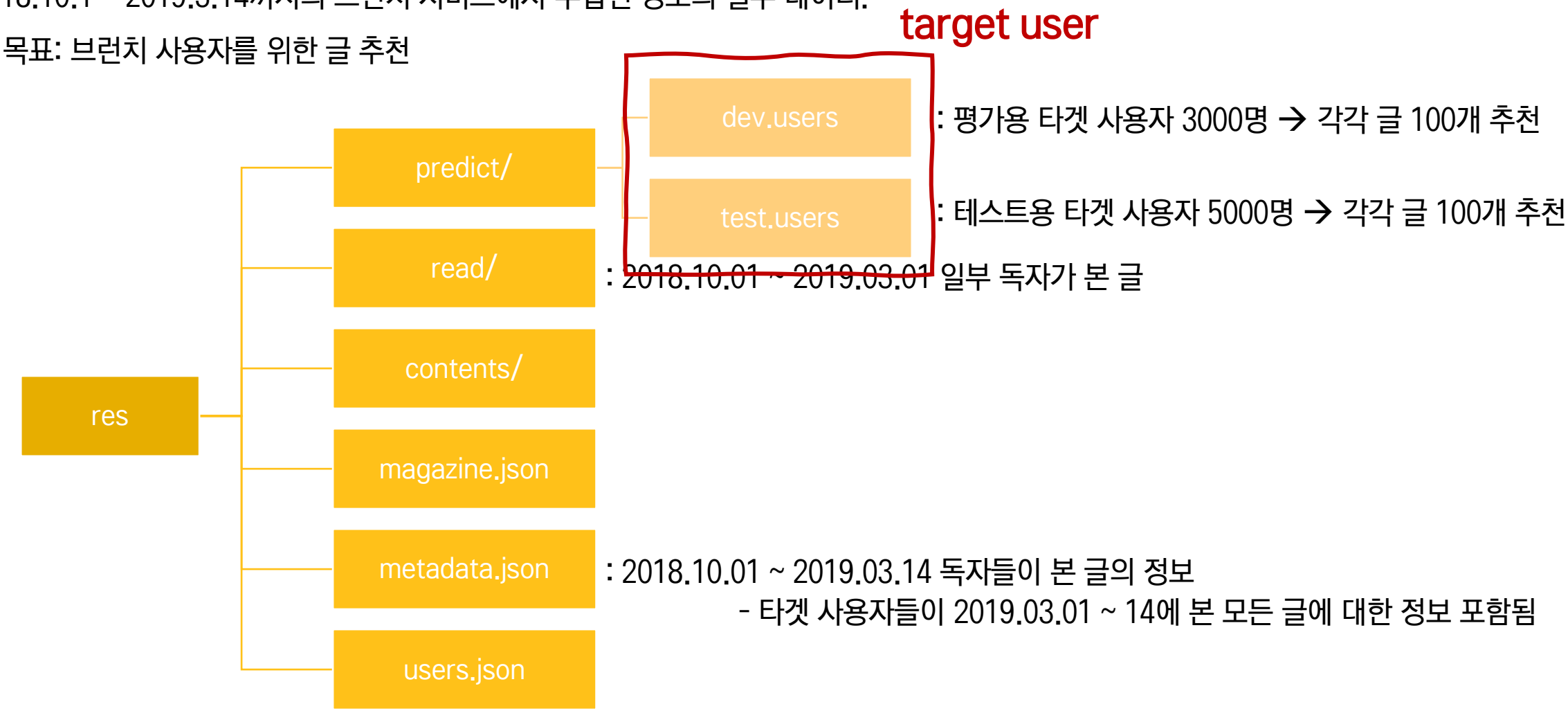
20.08.06

컴퓨터과학전공 1715237 이혜승

# Brunch data

2018.10.1 ~ 2019.3.14까지의 브런치 서비스에서 수집된 정보의 일부 데이터.

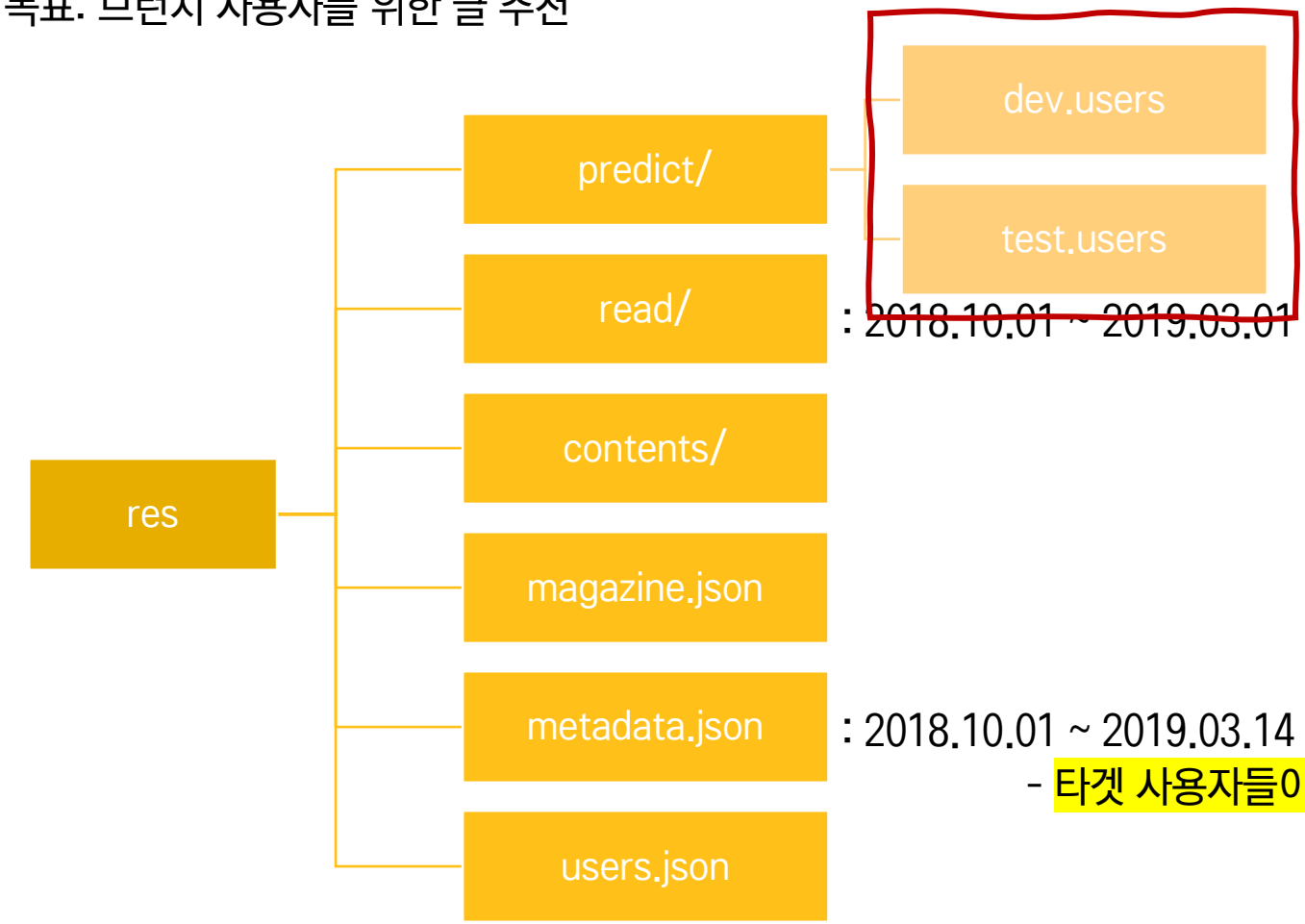
- 목표: 브런치 사용자를 위한 글 추천



# Brunch data

2018.10.1 ~ 2019.3.14까지의 브런치 서비스에서 수집된 정보의 일부 데이터.

- 목표: 브런치 사용자를 위한 글 추천




목표: **2019.02.22 ~ 2019.03.14** 기간동안 조회할 글을 예측

- 2019.02.22 ~ 2019.02.28: 조회기록 기반 예측( item based CF 활용)
- 2019.03.01 ~ 2019.03.14 메타데이터 이용( content-based 활용)

: 2018.10.01 ~ 2019.03.01 일부 독자가 본 글

: 2018.10.01 ~ 2019.03.14 독자들이 본 글의 정보

- 타겟 사용자들이 2019.03.01 ~ 14에 본 모든 글에 대한 정보 포함됨

# Brunch data

2018.10.1 ~ 2019.3.14까지의 브런치 서비스에서 수집된 정보의 일부 데이터.

- 목표: 브런치 사용자를 위한 글 추천

목표: **2019.02.22 ~ 2019.03.14** 기간동안 조회할 글을 예측

- 2019.02.22 ~ 2019.02.28: 조회기록 기반 예측( item-based CF )
  : 소비 패턴 활용 → 타겟 사용자 소비할 글 예측

- 2019.03.01 ~ 2019.03.14 메타데이터 이용( content-based recommendation )
  : 사용자의 작가 선호도, 글 소비 성향, 작가 성향 등의 유사도 계산 → 타겟 사용자 소비할 글 예측

# Brunch data

- USER의 글 소비 경향 (전체 / 최근 소비 경향) 반영하여 추천
- 최근 = 추천기간(2019.02.22 ~ 2019.03.14) 1주 전부터 설정

# Brunch data

- USER의 글 소비 경향 (전체 / 최근 소비 경향) 반영하여 추천
- 최근 = 추천기간(2019.02.22 ~ 2019.03.14) 1주 전부터 설정

(방법)

## 1. Item based CF

- not cold-start users: 예측 기간 이전 1주일 포함한 기간(2019.02.15 ~ 28)동안 읽은 글의 수가 평균 이상인 사용자들
- Not long-tailed items: 예측 기간 이전 1주일 포함한 기간(2019.02.15 ~ 28)동안 조회수가 상위 5%인 글들

- item user matrix 생성
- item에 대해 cosine similarity 계산
- 가장 비슷한 100개 item의 weighted mean을 이용하여 predict

(각 user에 대해 weighted mean이 높은 상위 100개 article)

# Brunch data

(방법)

그렇다면, cold-start user / long-tailed item에 대해서는 어떻게 추천?

idea)

조회수(view), 최근 조회수(recent_view) 높은 글 추천

구독하는 작가 글 중 안 읽은 글 추천

구독하는 매거진의 글 추천

읽은 글의 태그와 같은 태그를 가진 글 추천

Thought)

Long-tailed item은 소외된 글(view, recent_view가 적은 글) – item based CF로 해결

Cold-start item은 작성 된 지 얼마 안된 글(reg_ts가 최신인 글) - ??

# Brunch data

(방법)

## 2. Popularity based

- 조회수 높은 상위 20%의 글들이 많이 소비됨 -> 조회수 높은 인기 글 추천
- Long-tailed, cold-start item은 소외됨.

## 3. Following based

- 구독하는 작가의 글을 많이 읽음 → 구독하는 작가의 글 추천
- 구독하는 작가가 없는 경우? ? ? (전체 사용자의 98%가 구독 중)

## 4. Magazine based

- 읽었던 매거진의  글들을 많이 읽음
- 조회한 기록이 있는 매거진의 글을 추천

## 5. Tag based

- 조회한 글의 태그를 **사용자의 관심 키워드(interest)**로 가정
- 사용자의 관심 키워드를 갖고있는 글을 추천
- # 그나마 long-tailed item을 꺼내 올 수 있을 것 같음.

Content-based recommendation:
사용자의 글 조회(rating) 기반이 아닌 content 기반으로 추천.

# Brunch data

타겟 사용자들의 정보를 담은 데이터프레임(**target_df**)을 생성

1. users에 있는 user_id + users에 없는 user_id 추가
2. 타겟 사용자에 대한 read 정보 추가
    a. 전체 기간동안 target user가 본 article 저장( **target_df['read']** )
    b. 2주간 target user가 본 article 저장( **target_df['recent']** )
3. 타겟 사용자가 본 following 빈도수 저장(구독하는 작가의 글을 몇 개나 보았는 지 → 구독 작가에 대한 충성도(?))
4. 타겟 사용자가 본 magazine 빈도수 저장(읽은 글의 magazine 아이디와, 해당 매거진 조회 수 → 매거진에 대한 interest파악)
5. 타겟 사용자가 본 글의 태그 빈도수 저장(→ 읽은 글의 태그와 같은 태그를 가진 글을 추천 가능)
6. 타겟 사용자의 태그에서 빈도수가 높은 상위 N개 저장(→ 사용자의 관심 키워드 및 관심사 알고 추천해주기 위함)
    : 유사한 article들을 뽑아내면, long-tailed item / cold-start item을 끌어낼 수 있지 않을까?
    : Q) 같은 태그를 가진 글 중, 새로운 글 추천-> cold start 해결 / 조회수가 낮은 글 추천 -> long-tailed 해결. But 그 글들이 양질의 글인지 어떻게 알까?
7. 타겟 사용자의 글 소비성향 저장

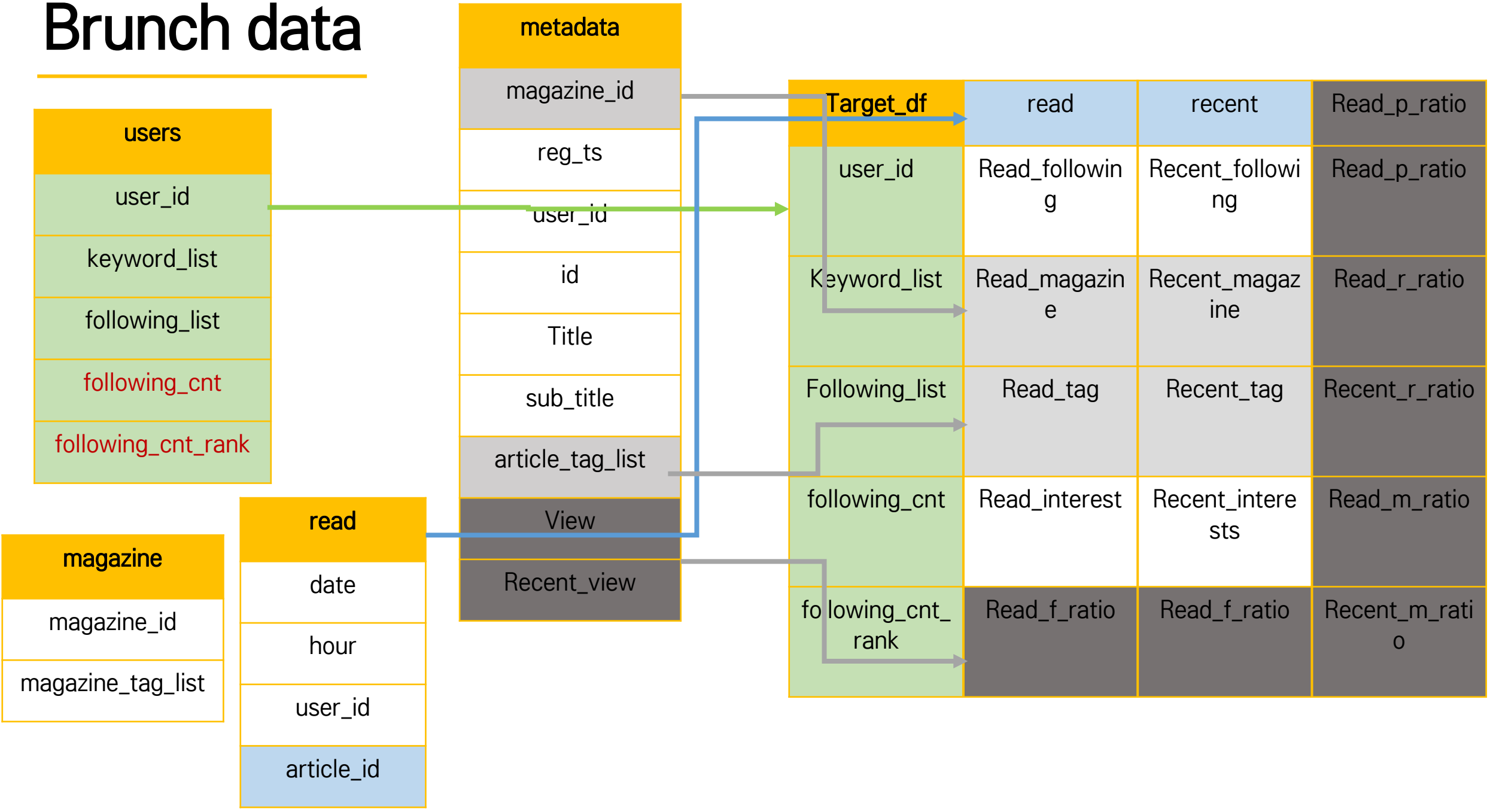
# Brunch data

타겟 사용자들의 정보를 담은 데이터프레임(**target_df**)을 생성

Columns:

- 'keyword_list'
- 'following_list'
- 'user_id'
- 'following_cnt'
- 'following_cnt_rank'
- 'read' / 'recent' : 전체/일정 기간동안 타겟 사용자가 읽은 글 목록
- 'recent_following' / 'read_following' : 전체/일정 기간동안 타겟 사용자가 본 구독작가의 글 개수
- 'read_magazine' /' recent_magazine' : 전체/일정 기간동안 타겟 사용자가 본 매거진 개수
- 'read_tag' / 'recent_tag' : 전체/일정 기간동안 읽은 글의 태그 정보
- 'read_interest' / ' recent_interest' : 전체/일정 기간동안 태그 빈도수가 높은 상위 N개로 관심 키워드설정
- 'read_f_ratio' / 'read_m_ratio' / 'read_p_ratio' / 'read_r_ratio' / 'recent_f_ratio',
- 'recent_m_ratio' / 'recent_p_ratio' / 'recent_r_ratio'

# Brunch data


users
user_id
keyword_list
following_list
following_cnt
following_cnt_rank
magazine
magazine_id
magazine_tag_list
read
date
hour
user_id
article_id
metadata
magazine_id
reg_ts
user_id
id
Title
sub_title
article_tag_list
View
Recent_view
Target_df
read
recent
Read_p_ratio
user_id
Read_following
Recent_following
Read_p_ratio
Keyword_list
Read_magazine
Recent_magazine
Read_r_ratio
Following_list
Read_tag
Recent_tag
Recent_r_ratio
following_cnt
Read_interest
Recent_interests
Read_m_ratio
following_cnt_rank
Read_f_ratio
Read_f_ratio
Recent_m_ratio

# reference

- Kakao arena 참여 팀 깃헙1 [github: hyeonho1028]: 데이터 loading에 참고
- Kakao Arena 2nd Competition[Arena: Brunch Article Recommedations]
- Kakao arena 깃헙[github:kakao arena]: 베이스라인 제공
- 참여팀 깃헙2[github: jihoo-kim]
- 참여팀 깃헙3[github: yeonmin]