RS: Explore data

20.07.23

컴퓨터과학전공 1715237 이혜승

Alibaba data

모바일 타오바오 추천 시스템의 클릭 데이터

- 목표: user의 과거 클릭 행동으로 다음 항목을 예측!

- Columns:

item_id		아이템의 primary key.
txt_vec	Item의 text feature	아이템의 정보를 모델에 의해 만들어진 real-valued vector로 표현함.
img_vec	Item의 image feature	
user_id		사용자의 primary key.
time	클릭 이벤트가 발생한 timestamp	0~1값. 순서 정보를 알 수 있음.
user_age_level		1 ~ 8의 categorical value
user_gender		F / M
user_city_level		1 ~ 6의 categorical value

Alibaba data

모바일 타오바오 추천 시스템의 클릭 데이터

- 목표: user의 과거 클릭 행동으로 다음 항목을 예측!

Training시에 활용하는 csv파일 3개 존재.(T=0일 때만 고려)

- Underexpose_item_feat.csv: 아이템 정보 데이터
  - item_id: col 0
  - txt_vec: col 1 ~ 128
  - img_vec: col 129 ~ 257

  # 벡터 정보는 추후 분석 때, word2vec을 활용하는듯?

underexpose_item_feat.csv

columns: item_id(아이템 번호: PK 역할), txt_vec(아이템의 text feature, 128차원), img_vec(아이템의 image feature, 128차원)

- 0번째 column: item_id
- 1 ~ 128번째 column: txt_vec
- 129 ~ 257번째: img_vec

```
In [23]: item_feat = pd.read_csv('./kdd_data/underexpose_item_feat.csv', header = None)
```

```
In [159]: item_feat
```

Out[159]:

	0	1	2	3	4	5	6	
0	42844	[4.514945030212402	-2.383720	0.500414	0.407068	-1.995229	0.109078	-0.69
1	67898	[-2.0029051303863525	-0.929881	0.790017	-1.380895	-0.510463	-1.810096	1.36
2	66446	[4.221673011779785	-1.497139	1.133570	-2.745607	-4.197045	-0.542392	-1.39
3	63651	[2.6579699516296387	-0.941863	1.121529	-5.109496	-0.279041	-0.351968	-1.08
4	46824	[3.192194938659668	-1.936676	1.199909	-2.562152	-2.573456	0.575841	-2.35
...	...	...	...	...	...	...	...	...
108911	79253	[2.1436519622802734	-1.591184	-0.283598	-2.186552	-1.505779	0.876601	1.30
108912	109138	[0.8901849985122681	0.042669	2.842594	-4.322702	-1.107593	-0.033230	2.83
108913	62184	[1.4589283466339111	-0.638677	-0.957509	-2.936515	-0.897658	-0.992379	0.57
108914	42748	[3.3835203647613525	-1.669863	1.264212	-2.128029	-2.129893	2.562061	-1.77
108915	61098	[2.4478535652160645	-0.469942	1.873896	-4.149315	-3.618531	-0.558433	0.16

Alibaba data

모바일 타오바오 추천 시스템의 클릭 데이터

- 목표: user의 과거 클릭 행동으로 다음 항목을 예측!

Training시에 활용하는 csv파일 3개 존재.(T=0일 때만 고려)

- Underexpose_user_feat.csv: 사용자 정보 데이터
  - user_id
  - user_age_level: 1~8 범주형
  - User_gender: F / M
  - user_city_level: 1~6 범주형

  # null값 존재 → 전처리 필요

underexpose_user_feat.csv

columns: user_id(사용자 번호: PK 역할), user_age_level, user_gender, user_city_level

```
In [58]: user_feat = pd.read_csv('./kdd_data/underexpose_user_feat.csv', header = None, names=[
```

```
In [59]: user_feat
```

Out[59]:

	user_id	user_age_level	user_gender	user_city_level
0	17	8.0	M	4.0
1	26	7.0	M	2.0
2	35	6.0	F	4.0
3	40	6.0	M	1.0
4	49	6.0	M	1.0
...	...	...	...	...
6784	35320	1.0	F	2.0
6785	35334	7.0	F	6.0
6786	35340	7.0	F	3.0
6787	35392	5.0	M	NaN
6788	35432	1.0	F	5.0

6789 rows × 4 columns

Alibaba data

모바일 타오바오 추천 시스템의 클릭 데이터

- 목표: user의 과거 클릭 행동으로 다음 항목을 예측!

Training시에 활용하는 csv파일 3개 존재.(T=0일 때만 고려)

- Underexpose_train_click-0.csv
  - user_id
  - item_id: 해당 row의 사용자가 클릭하여 조회한 item.
  - time: 사용자별로 그룹핑을 해서 확인하면, 해당 사용자가 어떤 순서로 click을 했는지 확인 가능.

  # fairness of exposure 문제와 가장 관련 있는 데이터.

  # 사용자의 행동을 click으로 판단 및 예측

  time column에 대한 생각

  1. 순서 O → 해당 사용자 클릭 행동 순서 정의 가능. (**click_order** 파생 변수)
  2. 순서로 인해, 다음 클릭을 확인하여 해당 사용자의 history와 next_item 확인 가능.
  3. 순서대로 정렬했을 때, 다음 클릭까지의 time term을 통해 얼마나 머물렀는지 확인 가능.
  4. Q: 같은 time값을 갖는 item_id가 여러 개 → 그만큼 짧게 머무른 건가? (ex.클릭 후 바로 이탈)
  5. 사용자가 데이터 수집 기간 중, 몇번의 click을 했는지 정의.(**click** 파생변수)

underexpose_train_click-0.csv

columns: user_id(사용자 번호: PK 역할), item_id, time(클릭 이벤트가 발생한 timestamp, 순서정보 존재)

fariness of exposure문제를 위한 가장 중요한 데이터로 여겨짐

```
In [132]: train_click = pd.read_csv('./kdd_data/underexpose_test_click-0.csv', header = None, na
```

```
In [163]: train_click
```

Out[163]:

	user_id	item_id	time	click	click_order	exposure_cnt	item_exposure_cnt
0	1133	221	0.983812	39	31	2	2
1	17864	253	0.983783	15	10	1	1
2	6941	309	0.983785	7	5	1	1
3	34089	358	0.983781	10	7	1	1
4	21659	536	0.983793	22	20	2	2
...	...	...	...	...	...	...	...
21211	836	116073	0.983791	36	22	17	17
21212	9218	116073	0.983790	52	48	17	17
21213	26433	116073	0.983794	13	8	17	17
21214	21131	116276	0.983783	8	7	1	1
21215	8415	116373	0.983794	38	25	2	2

21216 rows × 7 columns

Alibaba data

모바일 타오바오 추천 시스템의 클릭 데이터

- 목표: user의 과거 클릭 행동으로 다음 항목을 예측!

Training시에 활용하는 csv파일 3개 존재.(T=0일 때만 고려)

- Underexpose_train_click-0.csv
  - user_id
  - item_id: 해당 row의 사용자가 클릭하여 조회한 item.
  - time: 사용자별로 그룹핑을 해서 확인하면, 해당 사용자가 어떤 순서로 click을 했는지 확인 가능.

fariness of exposure 문제와 가장 관련 있는 데이터.
사용자의 행동을 click으로 판단 및 예측

노출 빈도 분포 탐색
Item_id가 얼마나 click되었는지 카운트하여, 각 아이템들의 노출 빈도를 확인해 봄.

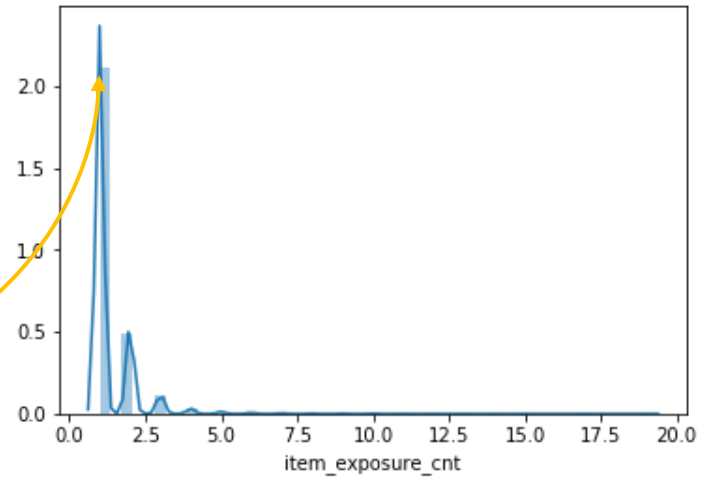
Item 전체 개수: 108,916개
사용자가 클릭한 item 개수: 15,670개(클릭 수가 1번인 것이 15,670개로 아이템 전체의 11%)
사용자의 클릭 수가 0번: 93,246개. (아예 노출 되지 않음. 아이템 전체의 85%)

```
In [154]: train_click['item_exposure_cnt'] = train_click.groupby(['item_id'])['time'].transform('
          item_exposure = train_click[['item_id','item_exposure_cnt']]
          item_exposure = item_exposure.drop_duplicates("item_id", keep="first")
          item_exposure
```

Out[154]:

	item_id	item_exposure_cnt
0	221	2
1	253	1
2	309	1
3	358	1
4	536	2
...	...	...
21201	112944	1
21206	114082	1
21207	114213	1
21209	115118	1
21214	116276	1

15670 rows × 2 columns



Alibaba data

모바일 타오바오 추천 시스템의 클릭 데이터

- 목표: user의 과거 클릭 행동으로 다음 항목을 예측!

Training시에 활용하는 csv파일 3개 존재.(T=0일 때만 고려)

- Underexpose_item_feat.csv: 아이템 정보 데이터
  - item_id: col 0
  - txt_vec: col 1 ~ 128
  - img_vec: col 129 ~ 257 # 벡터 정보는 추후 분석 때, word2vec을 활용하는듯?
- Underexpose_user_feat.csv: 사용자 정보 데이터
  - user_id
  - user_age_level
  - user_gender
  - user_city_level
- Underexpose_train_click-0.csv
  - user_id
  - item_id: 해당 row의 사용자가 클릭하여 조회한 item.
  - time: 사용자별로 그룹핑을 해서 확인하면, 해당 사용자가 어떤 순서로 click을 했는지 확인 가능.

item_id	
txt_vec	Item의 text feature
img_vec	Item의 image feature
user_id	
time	클릭 이벤트가 발생한 timestamp
user_age_level	
user_gender	
user_city_level	

Alibaba data

모바일 타오바오 추천 시스템의 클릭 데이터

- 목표: user의 과거 클릭 행동으로 다음 항목을 예측!

Training시에 활용하는 csv파일 3개 존재.(T=0일 때만 고려)

- Underexpose_item_feat.csv: 아이템 정보 데이터
  - item_id: col 0
  - txt_vec: col 1 ~ 128
  - img_vec: col 129 ~ 257 # 벡터 정보는 추후 분석 때, word2vec을 활용하는듯?
- Underexpose_user_feat.csv: 사용자 정보 데이터
  - user_id
  - user_age_level
  - user_gender
  - user_city_level
- Underexpose_train_click-0.csv
  - user_id
  - item_id: 해당 row의 사용자가 클릭하여 조회한 item.
  - time: 사용자별로 그룹핑을 해서 확인하면, 해당 사용자가 어떤 순서로 click을 했는지 확인 가능.

item_id	
txt_vec	Item의 text feature
img_vec	Item의 image feature
user_id	
time	클릭 이벤트가 발생한 timestamp
user_age_level	
user_gender	
user_city_level	

Brunch data

2018.10.1 ~ 2019.3.14까지의 브런치 서비스에서 수집된 정보의 일부 데이터.

- 목표: 브런치 사용자를 위한 글 추천

- 파일 구성:

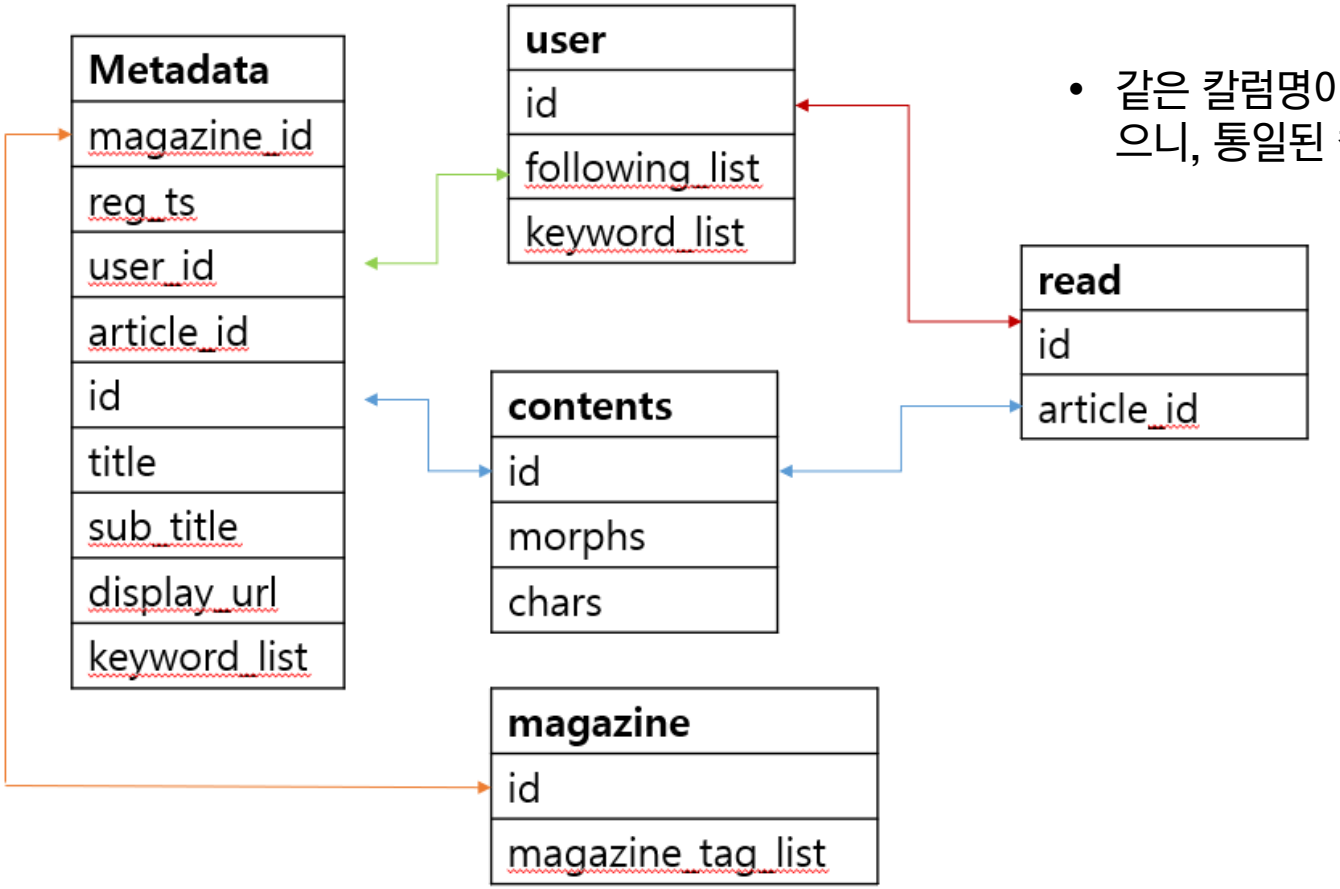
		#사용자id @읽은글들 @순서대로
read.tar	본(read) 글의 정보	- 읽은 (클릭한) 글이 순서대로 나열(사용자의 클릭 행동 순서O) - 머문 시간의 정보 or timestamp 존재하지 않기때문에 바로 이탈했는지 여부 알 수 없음.
metadata.json	글의 메타데이터	작가의 글들의 메타데이터 확인
/contents	글의 본문 정보	형태소 분석 결과 등의 본문 정보
users.json	사용자 정보	독자 및 작가의 사용자 정보
magazines.json	매거진 정보	

Brunch data

2018.10.1 ~ 2019.3.14까지의 브런치 서비스에서 수집된 정보의 일부 데이터.

- 목표: 브런치 사용자를 위한 글 추천

- columns:



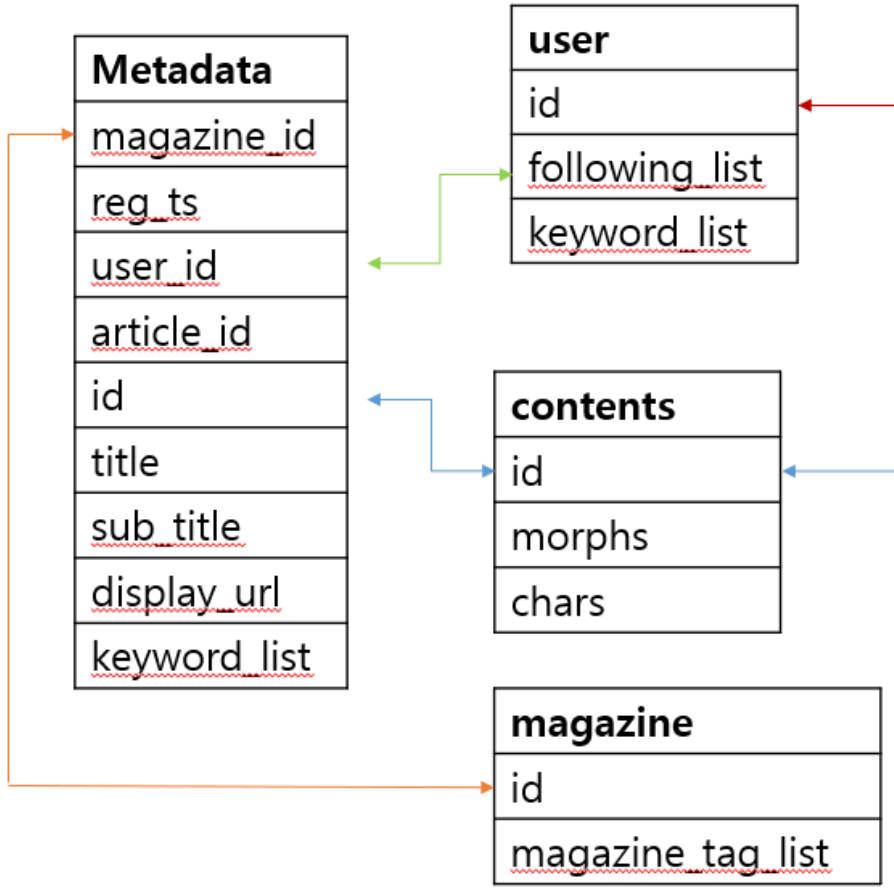
- 같은 칼럼명이어도, 각 파일에서 다른 정보들을 담고 있으니, 통일된 칼럼명으로 바꾼 후에 조인하려고 함.

Brunch data

2018.10.1 ~ 2019.3.14까지의 브런치 서비스에서 수집된 정보의 일부 데이터.

- 목표: 브런치 사용자를 위한 글 추천

- columns:



read.tar

Tar파일 압축 풀면, '시작일_종료일' 형태의 파일들로 구성됨.

각 파일에는 파일명에 기록된 시간동안 해당 사용자가 본 글들이 순서대로 나열되어 있음.

Columns:

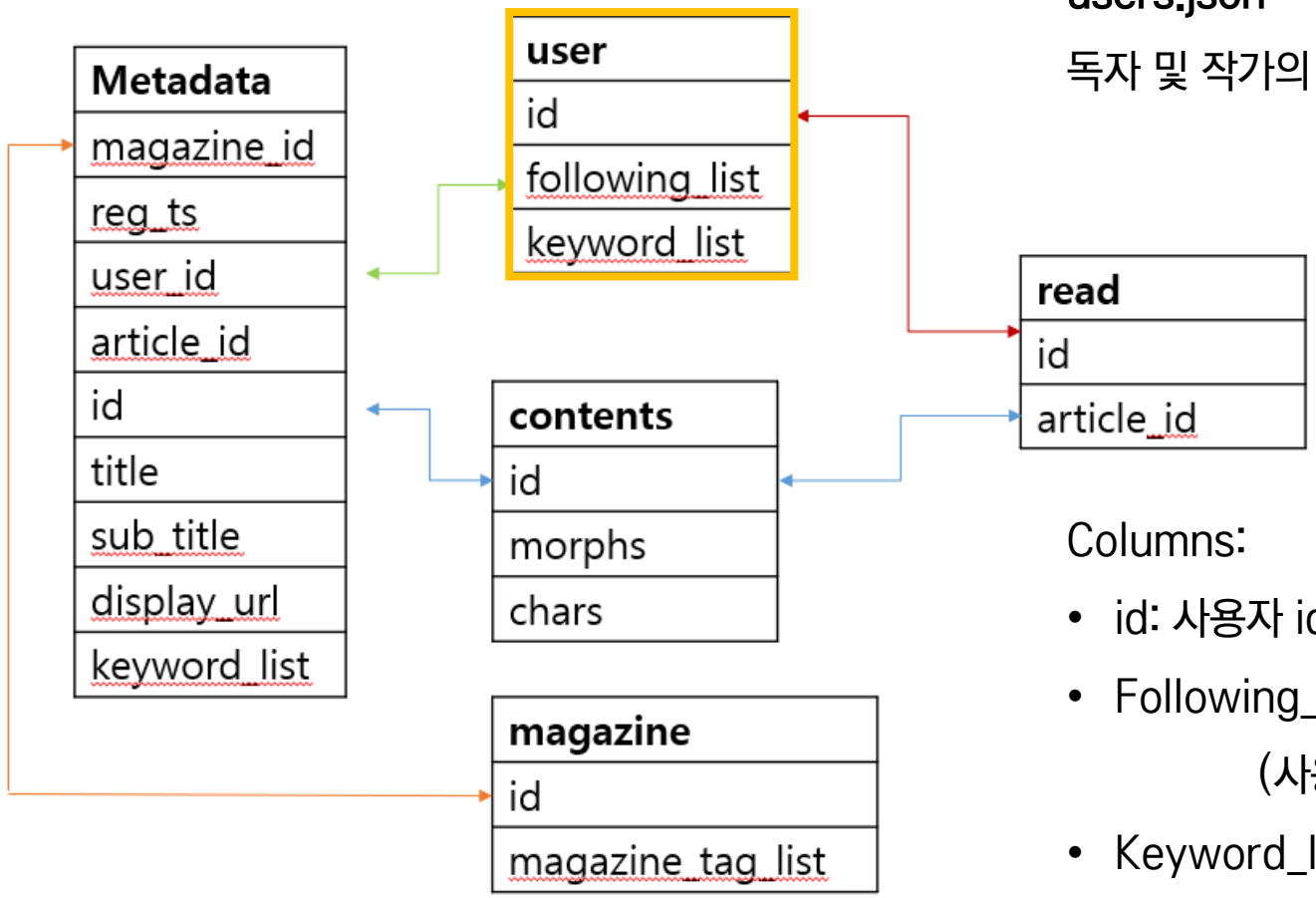
- id: 사용자 id
- article_id: 글 id

Brunch data

2018.10.1 ~ 2019.3.14까지의 브런치 서비스에서 수집된 정보의 일부 데이터.

- 목표: 브런치 사용자를 위한 글 추천

- columns:



users.json

독자 및 작가의 사용자 정보를 담고 있는 파일.

Columns:

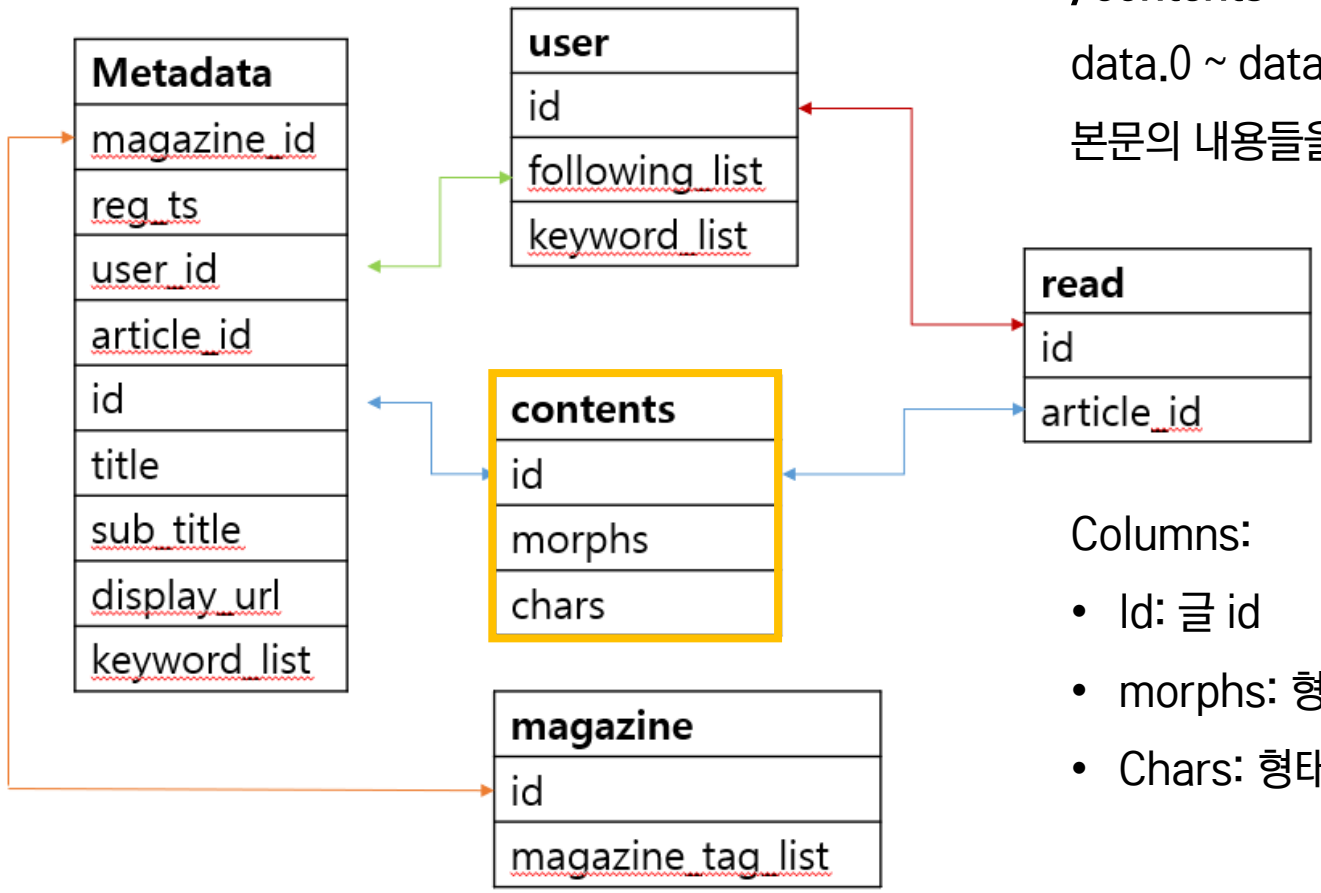
- id: 사용자 id
- Following_list: 사용자가 구독중인 작가리스트
  (사용자의 interest)
- Keyword_list: 최근 며칠 간 작가 글에 유입된 검색 키워드
  (사용자가 작가라면, 해당 작가 글에 유입된 정보)

Brunch data

2018.10.1 ~ 2019.3.14까지의 브런치 서비스에서 수집된 정보의 일부 데이터.

- 목표: 브런치 사용자를 위한 글 추천

- columns:



/contents

data.0 ~ data.6 총 7개의 파일이 존재.

본문의 내용들을 형태소 분석을 통해 추출한 것임.

Columns:

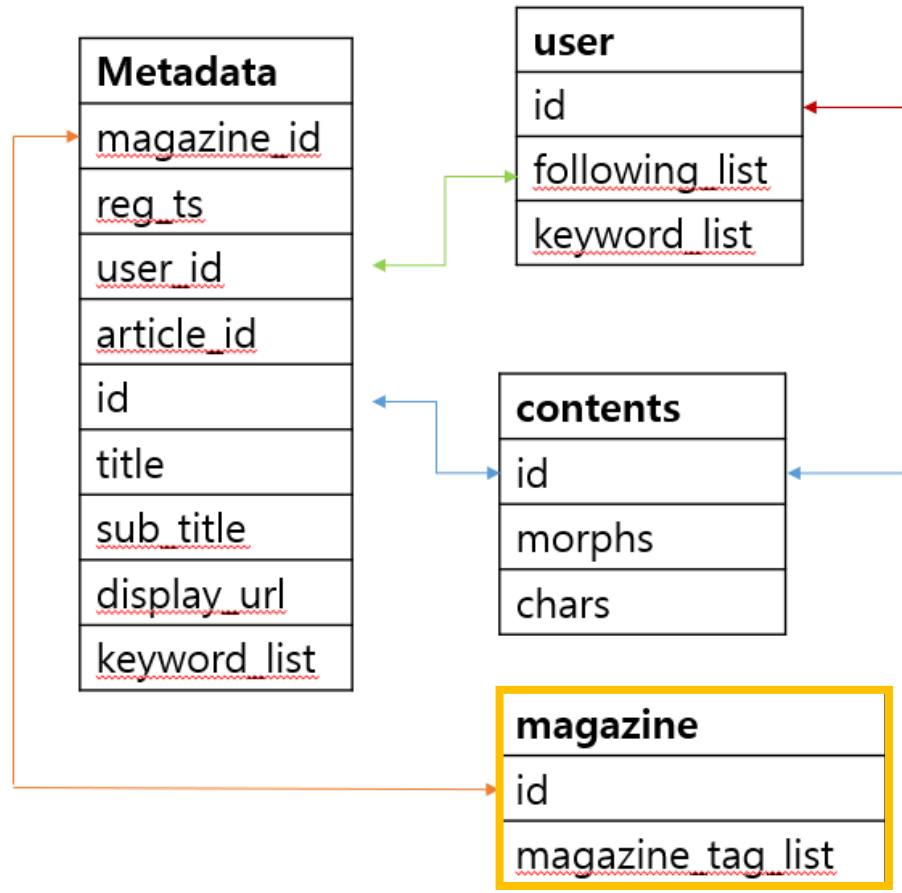
- Id: 글 id
- morphs: 형태소 분석 결과
- Chars: 형태소 분석 결과

Brunch data

2018.10.1 ~ 2019.3.14까지의 브런치 서비스에서 수집된 정보의 일부 데이터.

- 목표: 브런치 사용자를 위한 글 추천

- columns:



magazines.json

매거진: 비슷한 글들을 묶어서 잡지(목록)를 만든 것. 1명의 작가가 본인 글로 매거진을 만들 수도, 최대 20명의 작가가 참여해서 만들 수도 있음.

Columns:

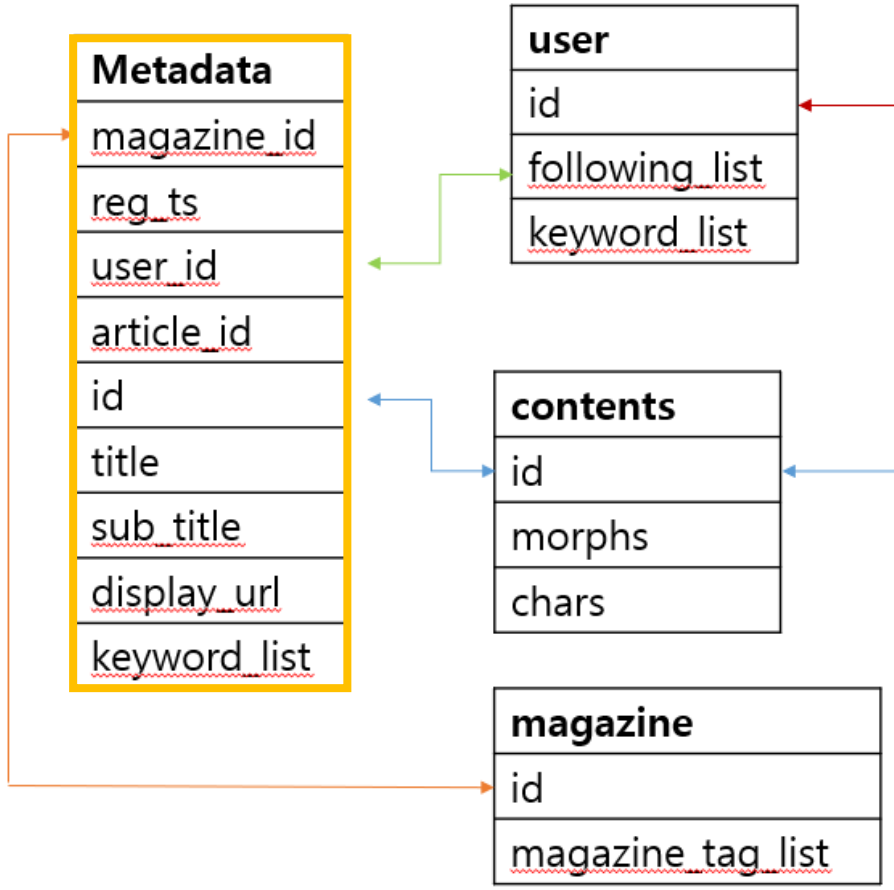
- id: 매거진 id
- magazine_tag_list: 작가가 부여한 매거진 태그 정보

Brunch data

2018.10.1 ~ 2019.3.14까지의 브런치 서비스에서 수집된 정보의 일부 데이터.

- 목표: 브런치 사용자를 위한 글 추천

- columns:



metadata.json

글들의 정보를 메타데이터로 담고 있음.

조회되지 않은 글들의 정보도 갖고 있다고 생각.

Columns:

- Magazine_id: 이 글의 브런치 매거진 아이디(없으면 0)
- Reg_ts: 글 등록 시간(unix시간으로 표기하여, 순서 정보 파악 가능)
- User_id: 작가 id
- Article_id: 글 id
- Id: 글 식별자(#'user_id'_'article_id'형태
- Title / Sub_title / Display_url
- Keyword_list: 작가가 등록한 태그 정보

Brunch data

2018.10.1 ~ 2019.3.14까지의 브런치 서비스에서 수집된 정보의 일부 데이터.

- 목표: 브런치 사용자를 위한 글 추천

Think

- 방법론은 잘 모르겠지만, metadata.json에 나온 '글'들과 read디렉토리 하위 파일들의 '글'과 비교하면,
  cold-start 문제 / long-tailed한 item(글)을 찾을 수 있지 않을까

- 매거진 → 독자의 interest, 성향 파악
- 구독하는 작가가 다른 작가와 매거진을 함께 한다면 유사하다고 판단 → 추천해줄 수 있을 듯.
- reg_ts: 유닉스 시간(1970.1.1 00:00:00으로 부터 경과한 시간을 초 단위로 나타냄). 새로운 아이템 여부 확인 가능.

Brunch data

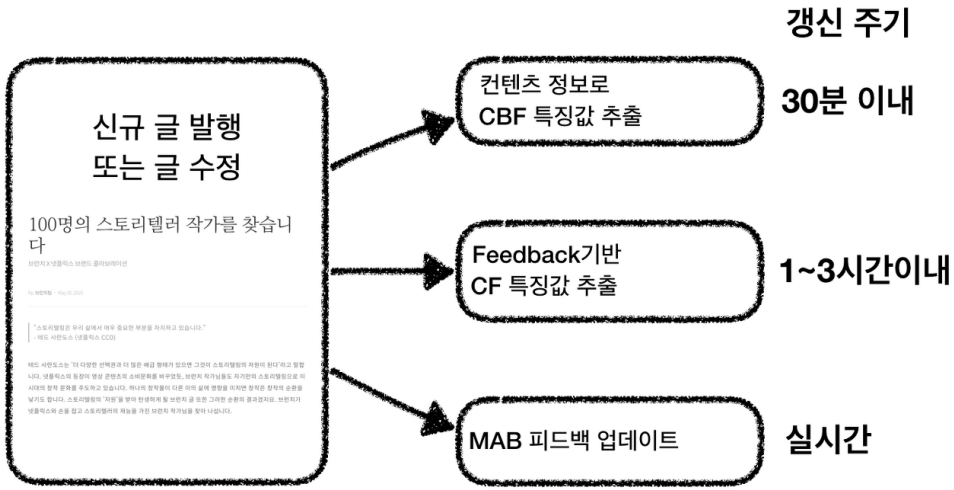
2018.10.1 ~ 2019.3.14까지의 브런치 서비스에서 수집된 정보의 일부 데이터.

- 목표: 브런치 사용자를 위한 글 추천

알리바바 데이터와 비교

- 작가가 글을 등록한 시간(reg_ts)이 주어짐.
  알리바바: 사용자가 아이템을 클릭/조회한 시간정보가 주어짐.
  (+) 새로운 글(new item / 노출이 되지않은 글 = cold-starter item)의 정보를 얻어낼 수 있음.

- Read디렉터리 하위의 data.0~6: 사용자의 history알 수 있음
  알리바바: train_click-0.csv의 time column 역할을 대체.
  next item 예측하기에 활용할 수 있을 듯
  다른 점: 하나의 글에 한 명의 사용자가 여러 번 접속하기도 함.



브런치에서 실제로 아이템 cold-start문제 해결하는 방법: 신규 item(글)에 대해 feature vec 추출(30분 이내) → 추천에 활용

어떠한 방법으로 해결할 수 있을 지는 더 고민!

reference

- Kakao arena 참여 팀 깃헙 [github: hyeonho1028]: 데이터 loading에 참고
- Kakao Arena 2nd Competition[Arena: Brunch Article Recommedations]
- Kakao arena 깃헙[github:kakao arena]: 베이스라인 제공
- About brunch competition [tistory: TEAM EDA]
- 브런치 데이터의 탐색과 시각화[brunch: 카카오 정책산업 연구]
- 알리바바 데이터 탐색[참가자 코드]