

| 올리브영 리뷰 데이터를 활용한 추천시스템 성능 개선

중간 발표

4조

이영현
강채원
김주은
임세은

목 차

1. Introduction

- 프로젝트 주제
- 무신사? 올리브영!

2. 진행 상황

- 데이터 수집
- 데이터 전처리

3. 향후 계획

- 리뷰 데이터 처리
- 모델링&성능 평가

Introduction

기존 추천 시스템 - ex) MovieLens 100k movie ratings

사용자 정보

id, 성별, 연령, 직업 등

아이템 정보

id, 제목, 개봉일,
장르별 원핫인코딩 컬럼

구매 정보

사용자 id, 아이템 id,
평점, timestamp

자연어 데이터를 크게 활용하지 않음

" 리뷰 데이터를 어떤 식으로 처리했을 때
추천 시스템 성능이 가장 좋은가 "

- KoBART를 이용한 요약
- Textrank를 이용한 요약
- Count Vectorizer를 이용한 임베딩
- TF-IDF를 이용한 임베딩
- ...

Introduction

프로젝트 주제 : 무신사 리뷰 데이터 분석 및 서비스 개발 (미정)

목적

- 리뷰자의 유형을 고려한 리뷰 데이터 분석
- 프로덕트 개선 or 상품 추천 알고리즘에 사용할 수 있는 분석 결과 도출

무신사 리뷰 데이터를 활용하는 이유

- 다양한 상품군 존재
- 리뷰 데이터 다양성
- 구매자별 리뷰 데이터 확인 가능
- 사용자 성비가 55:45로 균등한 편(2020년 7월 기준)



Introduction

무신사 리뷰



- 사용자 성비가 균등
- 구매자별 리뷰 데이터 확인 가능
- 다양한 상품군 존재



- 짧은 리뷰
- 관여도가 높은 사용자가 적음
(= 반복적으로 리뷰를 작성하는 사용자)

MUSINSA

E

LV.6달달한손기술

남성 · 168cm · 80kg · 신고



1993스튜디오

어센틱 아치 로고 스웨트셔츠_오프베이지

M 구매



여자친구랑 같이 커플로 입기 너무 좋아서 이번겨울 저것만 입었어요

사이즈 보통이에요

받기 보통이에요

색감 보통이에요

두께감 보통이에요

배송 빨라요

포장 꼼꼼해요



👍 도움돼요 0

😊 스타일 좋아요 0

Introduction

올리브영 리뷰

OLIVE  YOUNG



- 관여도가 높은 사용자가 다수 존재
- 구매자별 리뷰 데이터 확인 가능
- **긴 리뷰**



- 사용자 성비의 불균형
- 상품군이 한정적



17호봄브라이트

TOP 34

더모 코스메틱, 메이크업 분야 탐라뷰어
지성 · 볼륨톤 · 모공 · 블랙헤드

★★★★★ 2023.05.14

피부타입 지성에 좋아요 | 피부고민 보습에 좋아요 | 자극도 자극없이 순해요

지성용 토너로 사용하기 아주 좋아요.

발랐을 때 촉촉하고 흡수도 금방 잘 되는 편이라 바쁜 아침에도 듬뿍 듬뿍 사용하기 좋아요.

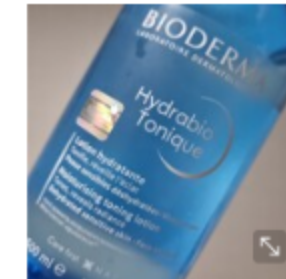
그리고 고보습 토너라고 되어있지만 바른 뒤에 유분감 있게 마무리되지 않고 수분만 필요한 만큼 깔끔하게 잘 채워줘서 여름에도 사용하기 아주 좋아요.

유분기 강한 스킨토너가 싫은 지성피부에게 추천하고 싶습니다.

아침에 화장 전엔 바이오더마 하이드라비오 토너 한번 바른 다음 바이오더마 연두색 세비엄 토너 한번 사용해주면 보습하면서 유분기관리까지 가능해서 두가지 같이 사용하는 것도 괜찮은 것 같습니다.

향은 바이오더마 특유의 호불호 거의 갈리지 않을, 어느 한쪽에 치우치지 않은 향이라서 토너 사용할 때 거부감도 없어서 더 좋아요.

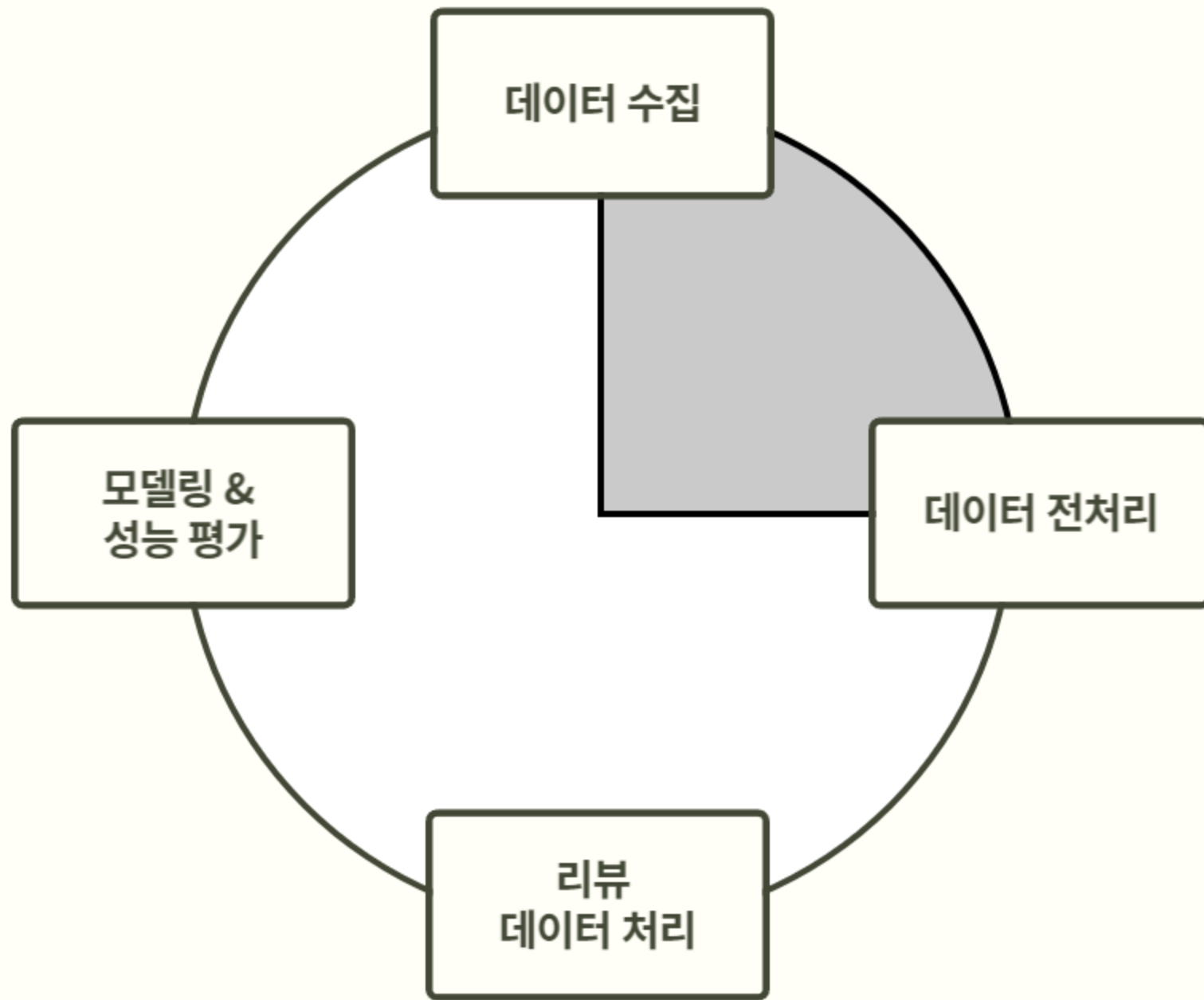
23년 초에 구입했을 때 25년도까지만 제품이 와서 유통기한 역시 아주 넉넉해 두고두고 사용하기도 좋아요.



※ 해당 리뷰는 원칙적으로 기본 상품이 동일한 단품 사용 후 작성된 것이며, 개별 상품에 따라 용량 내지 일부 구성(기, 기획상품 등)이 상이할 수 있음을 안내드립니다.

→ 올리브영의 리뷰 데이터가 적합하다고 판단

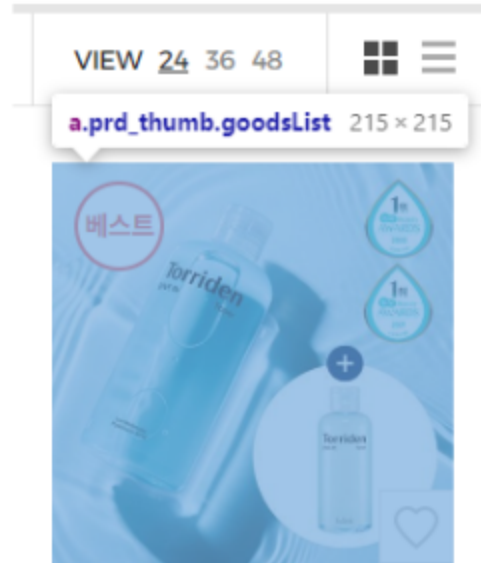
진행 상황



- 데이터 수집 : 토너/로션/올인원 상품에 한정
 - 상품 세부사항 페이지로 이동할 수 있는 key
 - 상품 세부사항
 - 사용자 리뷰 페이지로 이동할 수 있는 key
 - 사용자별 리뷰
- 데이터 전처리
 - 사용자 정보
 - 상품 정보

데이터 수집

상품 세부 페이지로 이동할 수 있는 key



토리든
[단독기획] 토리든 다이브인 저분자
히알루론산 토너 300ml 기획(+100m..
21,000원 **15,700원**
세일 오늘드림

```
<li criteo-goods="A000000170266001" class data-index="3">
  <div class="prd_info ">
    <a href="https://www.oliveyoung.co.kr/store/goods/getGoodsDetail.do?goodsNo=A0_010008&trackingCd=Cat100000100010008_MID&curation&egcode&rccode&egrcode" data-ref="goodsno" data-attr="카테고리상세^검색결과상품_인기순^[단독기획] 토리든 다이브인 저분자 히알루론산 토너 300ml 기획(+100ml 추가 증정)^4" data-ref-dispcatno="100000100010008" data-ref-itemno="001" data-trk="/Cat100000100010008_MID" data-impression="A000000170266^카테고리상세_검색결과상품_인기순^4" onclick="javascript: gtm.goods.callGoodsGtmInfo('A000000170266', '', 'ee-productClick', '카테고리상세_검색결과상품', '4');" data-impression-visibility="1"> == $0
      <span class="thumb_flag best">베스트</span>
      
    </a>
    <div class="prd_name">...</div>
    <button class="btn_zzim jeem" data-ref-goodsno="A000000170266">...</button>
    <p class="prd_price">...</p>
    <p class="prd_flag">...</p>
    <p class="prd_point_area tx_num">...</p>
    <p class="prd_btn_area">...</p>
  </div>
</li>
```

```
# 올리브영 > 토너/로션/올인원
### 모든 상품의 goodsno 가져오기
goodsno = list()

base_url = "https://www.oliveyoung.co.kr/store/display/getMCategoryList.do?dispCatNo=100000100010008&fltDispCatNo=&prdSort=01&pageIdx=({})"
for i in range(0,16):
    url = base_url.format(i+1)
    resp = requests.get(url)
    soup = BeautifulSoup(resp.content, 'html.parser', from_encoding = 'utf-8')
    if i != 15: #마지막 페이지가 아니라면
        for j in range(0,12): # 4 * 12 = 48개 상품에 대해 goodsno 크롤링
            temp = soup.find('div', id='Contents').find_all('ul', 'cate_prd_list gtm_cate_list')[j]
            no = temp.find_all('div', "prd_info")
            for m in range(0,4):
                goodsno.append(no[m].find('a')['data-ref-goodsno'])
    else: #마지막 페이지라면
        for j in range(0,9): # 4*9-1 = 35개 상품에 대해 goodsno 크롤링
            temp = soup.find('div', id='Contents').find_all('ul', 'cate_prd_list gtm_cate_list')[j]
            no = temp.find_all('div', "prd_info")
            if j != 8:
                for m in range(0,4):
                    goodsno.append(no[m].find('a')['data-ref-goodsno'])
            else: #마지막 페이지의 마지막 줄에는 상품 3개만 존재
                for m in range(0,3):
                    goodsno.append(no[m].find('a')['data-ref-goodsno'])
print(i,"번째 페이지 goodsno 스크래핑 완료")
```

→ 총 755개 상품 key 수집

데이터 수집

상품 세부사항

아누아 > 스킨케어 > 토너/로션/올인원 > 스킨/토너

[단독기획] 아누아 어성초 77 수딩 토너 350ml 기획(+토너40ml+패드2매+선크림10ml 증정)

30,500원 **22,800원** 혜택 정보

세일 오늘드림

75명이 보고있어요

내용물의 용량 또는 중량	토너350ml+패드2매+토너40ml+선크림10ml
제품 주요 사양	모든 피부용.

총 6,136 건

4.7 점

★★★★★

80% 15% 4% 1% 1%

5점 4점 3점 2점 1점

최고

피부타입	피부고민	자극도
건성에 좋아요 23%	보습에 좋아요 22%	자극없이 순해요 77%
복합성에 좋아요 63%	진정에 좋아요 78%	보통이에요 23%
지성에 좋아요 14%	주름/미백에 좋아요 1%	자극이 느껴져요 0%

```
#개별 상품 페이지 크롤링
driver = webdriver.Chrome(service=service, options=chrome_options)

for i,j in enumerate(goods.loc[:, 'url']): #나눠서 돌리기
    if pd.isnull(goods.loc[i, "용량"]): #크롤링 안 된 부분부터
        resp = requests.get(j)
        if resp.status_code == requests.codes.ok: #status_code == 200인 경우만
            driver.get(j) #개별 상품 페이지로 이동

            seed = np.random.randint(100)
            np.random.seed(seed)
            a = np.random.randint(5)
            time.sleep(a) #생성한 난수만큼 sleep

            item = driver.find_element(By.CLASS_NAME, 'prd_name').text
            goods.loc[i, '상품명'] = item
            brand = driver.find_element(By.CLASS_NAME, "prd_brand").text
            goods.loc[i, '브랜드'] = brand
            category = driver.find_elements(By.CLASS_NAME, "cate_y")[-1].text
            goods.loc[i, '카테고리'] = category #카테고리 : 마지막 카테고리만 가져오기
            price = driver.find_element(By.CLASS_NAME, "price-2").text
            goods.loc[i, '가격'] = price #가격

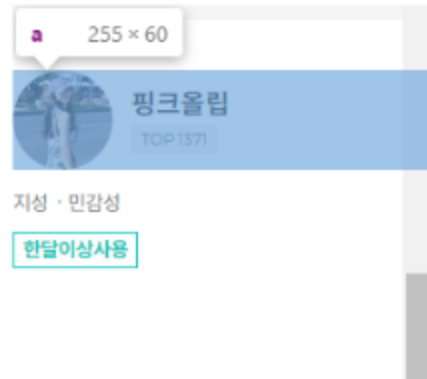
            info = "var element = document.querySelector('.goods_buyinfo');element.click();"
            driver.execute_script(info) #구매 정보 클릭
            driver.implicitly_wait(20)
            amount = driver.find_element(By.CSS_SELECTOR, "#artcInfo > dl:nth-child(2) > dd").text
            goods.loc[i, '용량'] = amount
            spec = driver.find_element(By.CSS_SELECTOR, "#artcInfo > dl:nth-child(3) > dd").text
            goods.loc[i, '주요사양'] = spec
            contain = driver.find_element(By.CSS_SELECTOR, "#artcInfo > dl:nth-child(8) > dd").text
            goods.loc[i, '성분'] = contain
```

→ 총 376개 상품 세부 사항 수집

데이터 수집

사용자 리뷰 페이지로 이동할 수 있는 key

상품별 리뷰 페이지에서 수집



```
<!-- ## 리뷰 고도화 1차 ## -->
<!-- 상품명 등록제한 카테고리 안내 문구 -->
<script type="text/javascript"></script>
<div class="reviewCate" id="gdasRecommKeyword" style=""></div>
<!-- 상품명 리스트 start -->
<div class="review_list_wrap">
  <ul class="inner_list" id="gdasList">
    <li>
      <div class="info">
        <div class="user clrfix">
          <a href="javascript:;" onclick="goods.gdas.goReviewerProfile('Vkk1RTkwT2wyd09oWgkyTHdSRExTdZ09')">
            data-attr="상품상세^리뷰어프로필^프로필이미지 또는 닉네임 클릭">
              <!--## 리뷰 고도화 1차 ## : top, id 위치 변경 및 마크업 변경 -->
```

```
def reviewer_id():
    for k in range(0,10):
        item = driver.find_element(By.CLASS_NAME, 'prd_name').text
        items.append(item) #상품명
        time.sleep(1)
        try:
            user_id = driver.find_elements(By.CLASS_NAME, 'id')[k]
            onclick = user_id.get_attribute('onclick')
            id_key = onclick.split("'")[-2]
        except IndexError:
            id_key = np.nan
        user.append(id_key) #사용자 id
```

```
for _,j in enumerate(goods.loc[131:, 'url']):
    resp = requests.get(j)
    if resp.status_code == requests.codes.ok: #status_code == 200인 경우만

        driver.get(j) #개별 상품 페이지로 이동
        time.sleep(0.5)
        driver.execute_script("window.scrollTo(0, document.body.scrollHeight);") #스크롤 끝까지 내리기
        try:
            review = driver.find_element(By.CSS_SELECTOR, '#reviewInfo')
            review.click() #리뷰 클릭
            driver.implicitly_wait(10)
            temp = driver.find_element(By.CSS_SELECTOR, "#gdasSort > li.is-layer.on")
            temp.click() #리뷰 유용한 순 클릭
            driver.execute_script("window.scrollTo(0, document.body.scrollHeight);") #스크롤 끝까지 내리기

            try: #리뷰가 있는 경우
                reviewer_id() #첫페이지 리뷰 정보 가져오기
                for i in range(2,11): #페이지 이동 : 10페이지까지
                    next = driver.find_element(By.CSS_SELECTOR, "#gdasContentsArea > div > div.pageing > a:nth-child({})".format(i))
                    next.send_keys(Keys.ENTER)
                    driver.implicitly_wait(3)
                    reviewer_id()
            except NoSuchElementException: #리뷰 없는 경우 : 리뷰가 10개 이하인 경우도 같은 메시지
                print(_, "리뷰가 없습니다")
        except NoSuchElementException: #페이지 로딩 안 되는 경우
            print(_, "로딩 실패")
        except: #기타 에러 발생한 경우 프록시 바꾸기
            print(_, "프록시 변경 필요")
            driver.quit()
            break
```

→ 총 8047명의 key 수집

데이터 수집

사용자별 리뷰



핑크올립

TOP1371

지성 · 원톤 · 민감성

364
도움

1
팔로워

1
팔로잉

팔로우

컬렉션 2



남남



패드맛집

누적 리뷰 183

최근작성순 ▾



필리밀리

필리밀리아쿠아핏 여성용 면도기 (색상랜덤발송)

★★★★★ 작성일자 2023.05.22

한달이상사용

물로만 면도가 가능하고 5중 면도날이여서 깔끔하게 제모를 해줍니다

도움이 되었어요 1

신고하기

```
# 유저 반복문 추가하기
for m,n in enumerate(df['리뷰자']):
    if pd.isnull(df.loc[m, '상품명']): #크롤링 안 된 부분부터
        if m % 20 == 0: # 인덱스가 20의 배수인 경우만
            url = base_url.format(n)
            try:
                driver.get(url)
                driver.implicitly_wait(5)
                for _ in range(2): #스크롤 2번 실행
                    driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
                    time.sleep(1)

                resp = driver.page_source #현재 페이지 파싱
                soup = BeautifulSoup(resp, "html.parser")
                for i in range(0,20): #상품명 20개 크롤링
                    try:
                        type = soup.find('ul','profile-keyword-list').find_all('li','list-item')
                        if len(type) == 4: #타입 정보가 전부 있을 경우
                            df.loc[m+i, 'type1'] = type[0].get_text()
                            df.loc[m+i, 'type2'] = type[1].get_text()
                            df.loc[m+i, 'type3'] = type[2].get_text()
                            df.loc[m+i, 'type4'] = type[3].get_text()
                        else: #타입 정보가 일부만 있을 경우
                            for j in range(len(type)):
                                df.loc[m+i, 'type{}'.format(j+1)] = type[j].get_text()
                    except AttributeError: #타입 정보가 없을 경우
                        df.iloc[m+i, 1:5] = np.nan #결측값으로 채우기
                try:
                    df.loc[m+i, '브랜드'] = soup.select('p.rw-box-figcaption__brand')[i].get_text()
                    df.loc[m+i, '상품명'] = soup.select('p.rw-box-figcaption__name')[i].get_text()
                    df.loc[m+i, '평점'] = soup.select('span.point')[i].get_text()
                    df.loc[m+i, '작성일자'] = soup.select('span.review_point_text')[i].get_text()
                    df.loc[m+i, '본문'] = soup.select('p.rw-box__description')[i].get_text()
                except IndexError: #리뷰한 상품 개수가 20개보다 적을 경우
                    df.iloc[m+i, 5:] = np.nan #결측값으로 채우기
            if m % 1000 == 0:
                print(m, "번째 크롤링 완료")
        except: #에러 발생 시 멈춤
            print(m, "번째에서 에러 발생")
            break
```

→ 총 89933개 리뷰 수집 (토너/로션/올인원 제품에 대한 리뷰 5427개)

데이터 전처리

사용자 정보

사용자 key	type1	type2	type3	type4	랭킹
eHNCNmFVY1ladXNQ V21xcS9zT0J1QT09	지성	웜톤	모공	트러블	1792
ckN0NIJxTy9WU2lyM jZQWHQ3dWpzUT09	건성	웜톤	주름	탄력	0
U3Ey2xHVytUcFVYa jdSbkJVaUV3UT09	민감성	겨울쿨톤	각질	다크서클	0
T3VnTkYrTnFocnkzdk txVVg3VFJuQT09	복합성	웜톤	주름	탄력	0
ekdrWEZHc1dYb2ZJT EpDVXI3Sm1adz09	건성	쿨톤	각질	모공	0

- type1: 피부타입(건성, 지성, 복합성...)
- type2 : 퍼스널컬러(웜톤, 쿨톤, 겨울쿨톤...)
- type3 & 4 : 피부고민

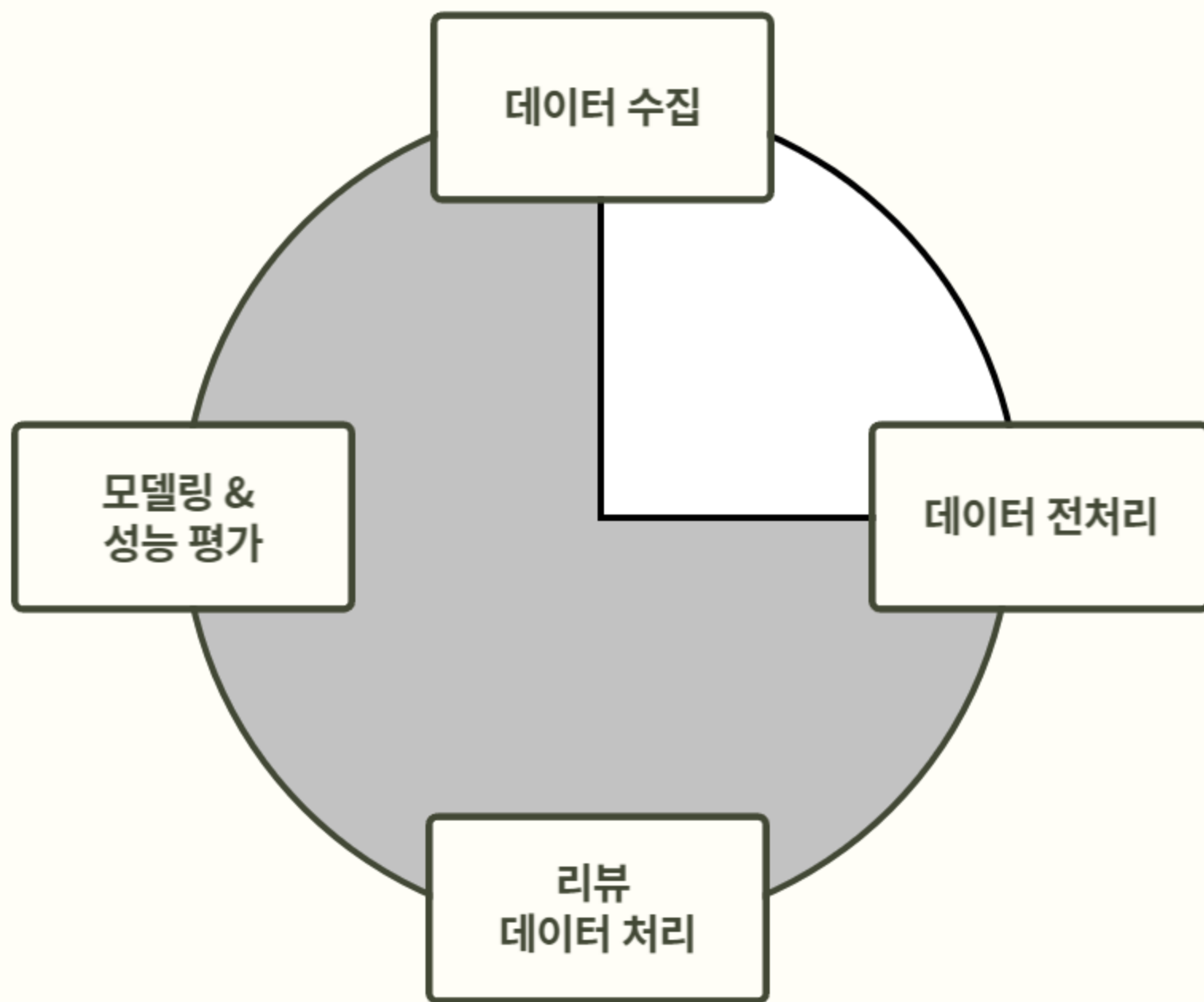
데이터 전처리

상품 정보

goodsno	상품명	카테고리	(중략)	가격	용량	증정 여부
A000000006564	우르오스 스킨로션 200ml	올인원	...	23,700	200ml	0
A000000006565	우르오스 스킨밀크 200ml	올인원		23,700	200ml	0
A000000137180	대용량] 라운드랩 1025 독도 토너 (본품500ml+100ml 추가 증정)	스킨/토너		27,000	본품]라운드랩 1025 독도 토너 500ml+증정품]라운드랩 1025 독도 토너...	1
A000000170266	단독기획] 토리든 다이브인 저분자 히알루론산 토너 300ml 기획(+100ml 추...	스킨/토너		15,700	본품] 다이브인 토너 300ml 증정] 다이브인 토너 100ml	1

- 상품 및 용량 정보 이용, 증정품 제공 여부 컬럼 생성

향후 계획



- 리뷰 데이터 처리 : 각 처리 방법에 대해 성능 평가
 - Text Summerization : 사용자별 리뷰 요약
 - KoBART, Textrank ...
 - Embedding
 - Count Vectorizer, TF-IDF ...
- 모델링 & 성능 평가
 - 사용자 - 상품 간 구매 여부 노드 구성
 - 노드 분리 : train set, test set
 - GNN 구축 및 학습
 - test set에 대해 성능 평가

참고 자료

- <https://news.edupang.com/news/article.html?no=99638>
- https://biz.chosun.com/site/data/html_dir/2020/11/13/2020111301702.html

감사합니다

4조

이영현
강채원
김주은
임세은