

DATATON PROJECT

오마카세 조

김현수

김상혁

박성환

최유빈

CONTENTS

1. INTRODUCE
2. 중고거래 비즈니스 및 분석 방향성
3. 데이터 현황 파악 및 전처리
4. Feature Engineering
5. 데이터 활용 및 판매 예상율
6. 판매 예상율 적용시, 기대효과

Team Member



김현수

부산의 제라드
오마카세조 리더
팀원 사기 증진 및 데이터 분석



김상혁

에릭
수학 전문가
데이터 분석 및 기획 및 검증



박성환

성환(영어 네임)
건축수학분석 전문가
데이터 분석 및 기획 및 검증



최유빈

케이티
데이터 기획 전문가
데이터 분석 및 기획





오마카세 TEAM

다양한 식재료들이 서로 상호 융합 보완 상승 효과를 하며 멋진 조화를
이룰때 맛있는 음식이 탄생하듯이

다양한 조원들의 생각과 경험이 모여 서로 의논하고 토의하는 과정을
거치며 결과적으로 멋있는 프로젝트를 만들자는 의미

주요 프로세스-1

중고거래 비즈니스 및 분석 방향성

Mercari Price Suggestion Challenge

160만개의 거대 데이터셋으로, 여러 변수간의 관계 분석 용이

NLP 데이터 분석에 대한 기대감

이용자/기업 대상 비즈니스 의사결정에 제안할 부분 많을 것으로 기대

메루카리 Business는 뭘까?



앱 주요 지표

- 연간 활성자수 : 1050만명
- 하루 이용자 : 71만명
- 연 거래액 : 2조 9153억원
- 연매출 : 2210억
- 일본 중고시장 점유율 : 60% 이상

주요 비즈니스

- 주요 비즈니스 : 중고거래 '당근마켓'과 유사하지만, 당근마켓은 직거래인 반면, 메루카리는 중고거래액의 10%라는 높은 수수료 구조로 수익성 강세
- 복잡한 중고거래 과정을 생략하고 사진과 글만 올리면 되는 간단한 프로세스
- 판매자가 가격을 미리 결정 후 구매자가 승인하면 거래 이루어지는 구조
- 10~30대 여성 타겟

이용자 관점

- 이 가격을 믿어도 되는지?
- 제품에 대한 설명이 잘 작성되어있는지?
- 이 상품의 퀄리티를 믿어도 되는지?
- 요즘 핫한 상품을 거래하려고 하는 것인지?
- 유료배송인지 무료 배송인지?

→ 좋은 물건을 적정한 가격에 구매

기업 관점

- 가격이 저렴해야 잘 팔리는지?
- 어떤 카테고리들이 많이 거래되는지?
- 브랜드가 작성되어있어야 잘 거래가 되는지?

→ 거래 활성화를 통한 수익 증대

판매 제품 특성 분석을 통해, 판매율 향상과 앱 활성화를 위한 비즈니스 전략 제안

중고거래 특징

▶ 중고거래 플랫폼 이용자들은 다양한 불편함을 경험하고 있으며,
특히 **거래 사기(46%)**, **상품 정보 불일치(32.4%)**, 주문 취소 및 환불 문제(13.5%) 등의 **신뢰도 문제를 가장 큰 불만 요인**으로
~~~ (이하 생략)

(경향신문, "유산균, 비타민 등은 당근마켓에서 거래하면 안됩니다", 202207)

▶ **판매자를 신뢰할 수 없어서(33.3%)** 중고거래 플랫폼 이용을 꺼리는 소비자도 많으며, **사기 방지 시스템 부족(37.1%)**과 ~~~~~  
~~~ (이하 생략)

(토익위원회, "고거래, 자주 이용하시나요? 중고거래 트렌드 설문조사 결과", 202212)

▶ 중고거래에 만족한 소비자들은 **저렴한 가격(73.4%)**와 **편리한 접근성(50.7%)**, 다양한 판매자 및 상품(24.9%) 등을 긍정적인 요소로
꼽고 있습니다. ~~ (이하 생략)

(토익위원회, "고거래, 자주 이용하시나요? 중고거래 트렌드 설문조사 결과", 202212)



중고거래 시장은 신뢰와 가격, 제품 퀄리티 등이 가장 중요, 데이터 기반 신뢰 보장 장치가 필요.
데이터 기반, 판매 예상을 시스템과 같은 서비스를 만들 수 있다면?

목차 리마인드

1. INTRODUCE
2. 중고거래 비즈니스 및 분석 방향성
3. 데이터 현황 파악 및 전처리
4. Feature Engineering
5. 데이터 활용 및 판매 예상율
6. 판매 예상율 비즈니스 적용시, 기대효과



주요 프로세스-2

**데이터 현황 파악 및
전처리**

데이터 확인

Train.tsv 선택

- 가격 예측 데이터라 test셋을 포함한 4개 데이터가 있었으나 Price 가격이 포함된 Train 데이터로 분석
- 총 148만 2535건 데이터, 8개의 컬럼

데이터

| | train_id | name | item_condition_id | category_name | brand_name | price | shipping | item_description |
|---------|----------|-------------------------------------|-------------------|---|------------|-------|----------|---|
| 0 | 0 | MLB Cincinnati Reds T Shirt Size XL | 3 | Men/Tops/T-shirts | NaN | 10.0 | 1 | No description yet |
| 1 | 1 | Razer BlackWidow Chroma Keyboard | 3 | Electronics/Computers & Tablets/Components & P... | Razer | 52.0 | 0 | This keyboard is in great condition and works ... |
| | ... | | ... | ... | | | ... | |
| 1482534 | 1482534 | Brand new lux de ville wallet | 1 | Women/Women's Accessories/Wallets | NaN | 22.0 | 0 | New with tag, red with sparkle. Firm price, no... |

1482535 rows x 8 columns

컬럼명 상세

| Column Name | 설명 |
|-------------------|--|
| train_id | 각 제품의 고유 ID |
| name | 제품명 |
| item_condition_id | 제품 상태 (1: 새 제품, 5: 많이 사용됨) |
| category_name | 제품 카테고리 (예: Women/Tops & Blouses/T-Shirts) |
| brand_name | 브랜드명 (없을 수도 있음) |
| price | ₩ 제품 가격 |
| shipping | 배송비 부담 (0: 구매자, 1: 판매자) |
| item_description | 제품 설명 (NLP 분석 가능) |

1. 결측치 대체 및 제거

```
[ ] # 각 컬럼별 결측치 개수 확인 총3개의 컬럼에서 결측치 발견
print("결측치 현황:")
print(train_df.isnull().sum())

# 중복 데이터 확인 중복데이터 없음
print("중복 행 수:", train_df.duplicated().sum())
```

→ 결측치 현황:

| | |
|-------------------|--------|
| train_id | 0 |
| name | 0 |
| item_condition_id | 0 |
| category_name | 6327 |
| brand_name | 632682 |
| price | 0 |
| shipping | 0 |
| item_description | 6 |
| dtype: int64 | |
| 중복 행 수: | 0 |

- Category_name과 Brand_name 결측치 多
- Category nan값은 unkown으로 대체
- Brand_name nan값은 unkown으로 대체
- Item_descriptio값은 no description으로 대체

```
[ ] # 결측치 처리: 결측값이 있는 컬럼별로 적절한 (도움을 주기 위해서 미리 입력되어 있는 텍스트))로 채우기

# category_name 컬럼: 결측치는 'unknown'으로 채움
train_df['category_name'] = train_df['category_name'].fillna('unknown')

# brand_name 컬럼: 결측치는 'unknown'으로 채움
train_df['brand_name'] = train_df['brand_name'].fillna('unknown')

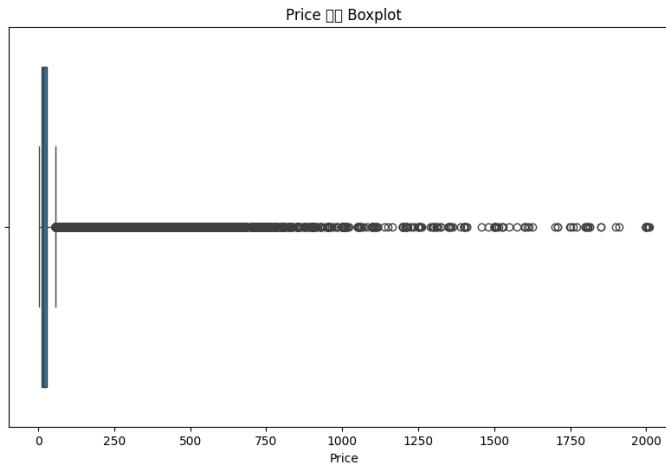
# item_description 컬럼: 결측치는 'no description'으로 채움
train_df['item_description'] = train_df['item_description'].fillna('no description')

# 결측치 처리 후 결과 확인
print("처리 후 결측치 현황:")
print(train_df.isnull().sum())
```

→ 처리 후 결측치 현황:

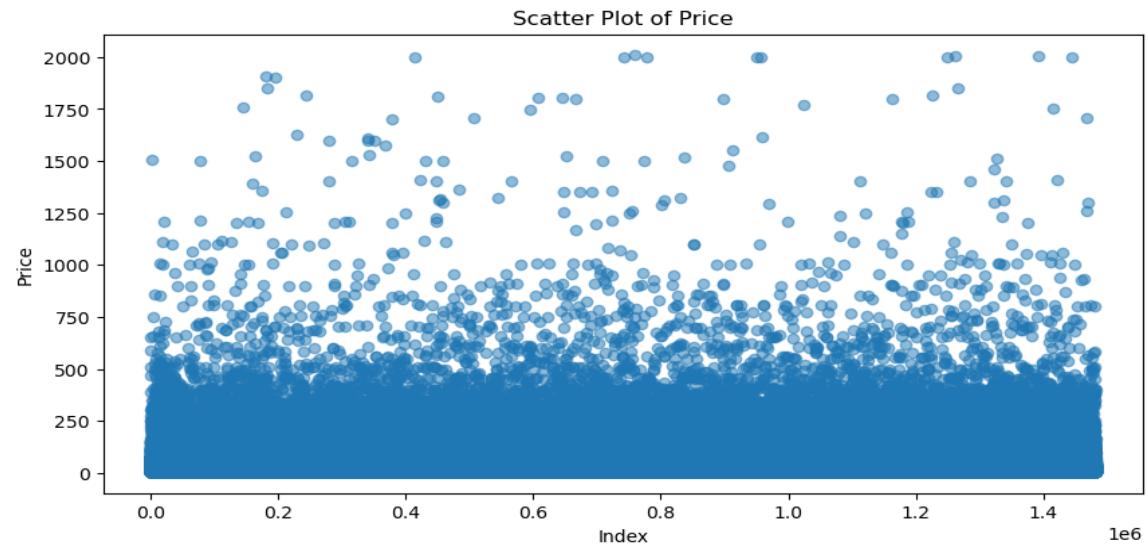
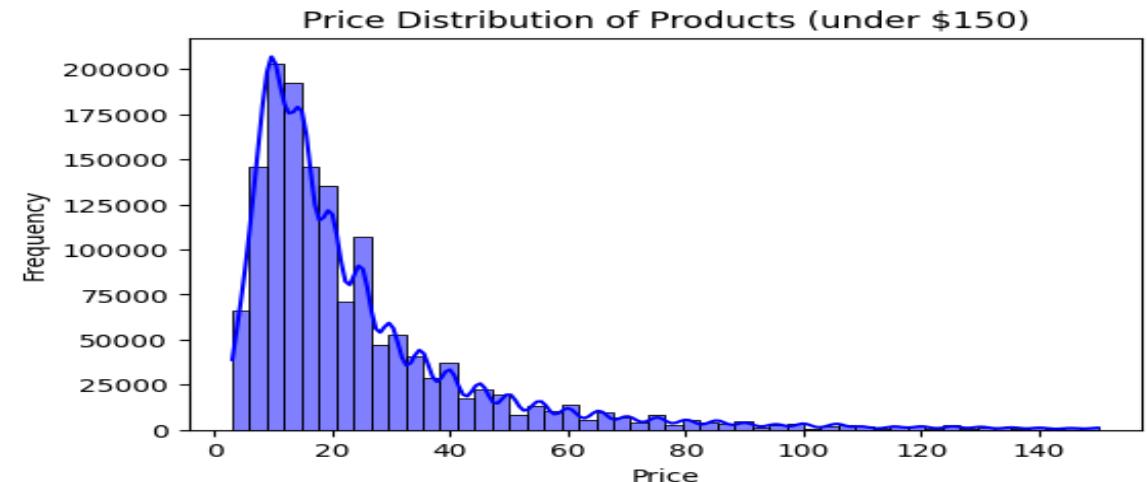
| | |
|-------------------|---|
| train_id | 0 |
| name | 0 |
| item_condition_id | 0 |
| category_name | 0 |
| brand_name | 0 |
| price | 0 |
| shipping | 0 |
| item_description | 0 |
| dtype: int64 | |

2-1. 세부 컬럼 확인_Price



```
train_df = train_df[train_df['price'] > 0] #price 0인거 제거
```

- Price = 0 값은 제외하였음
- 가격이 저가 구간에 몰려있는 경향성을 확인 (\$150 이하 多)
- 카테고리별 고가의 제품도 있기 때문에 이상치를 따로 제거X
- 가격 컬럼만 있을 뿐, 팔린 여부에 대한 컬럼은 없어 아쉬움



2-2. 세부 컬럼 확인_Brand_name

```
# 'brand name' 관련
```

```
train_df['brand_name'].unique() #뭐가 많다..
```

```
array(['unknown', 'razer', 'target', ..., 'astroglide', 'cumberland bay',
       'kids only'], dtype=object)
```

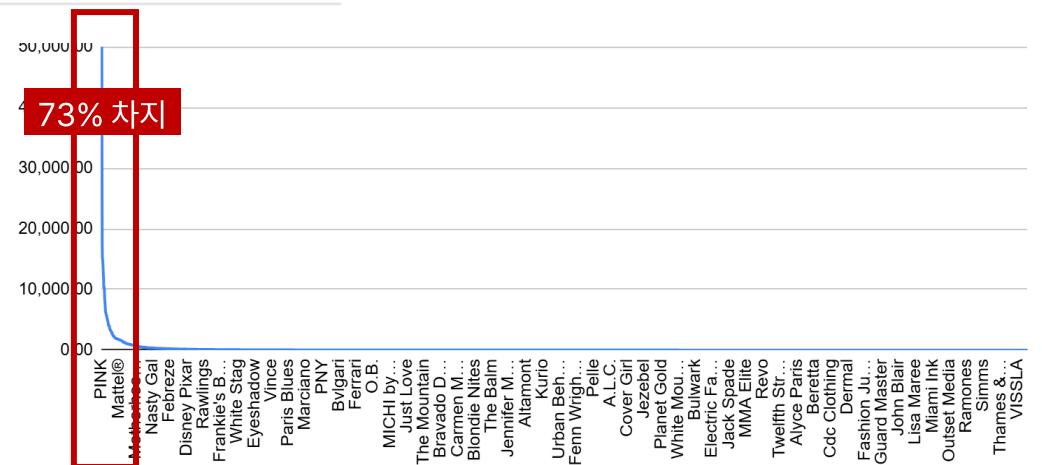
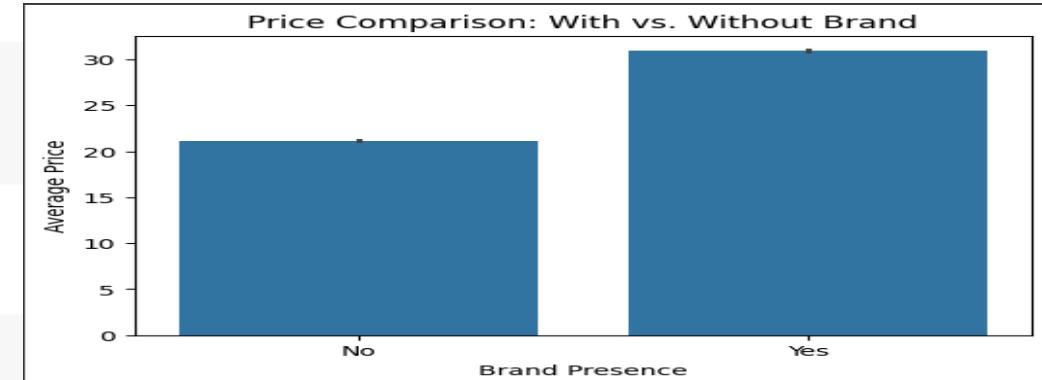
```
train_df[train_df['brand_name'] == '...'] #...는 이하 생략이였다.
```

| train_id | name | item_condition_id | category_name | brand_name | price | shipping | item_description |
|----------|------|-------------------|---------------|------------|-------|----------|------------------|
|----------|------|-------------------|---------------|------------|-------|----------|------------------|

```
train_df['brand_name'].nunique() #one hot 진행하기에는 너무 많다..
```

4810

- Brand_name의 unique값은 4810개
- 빈도수 상위에 있는 브랜드가 전체의 73%를 차지함
- 모든 브랜드를 사용하기보다 범주형으로 나눌 필요성이 보임
- 따라서 플랫폼 내 언급량 TOP 100 브랜드 여부에 따라 범주형 컬럼 생성



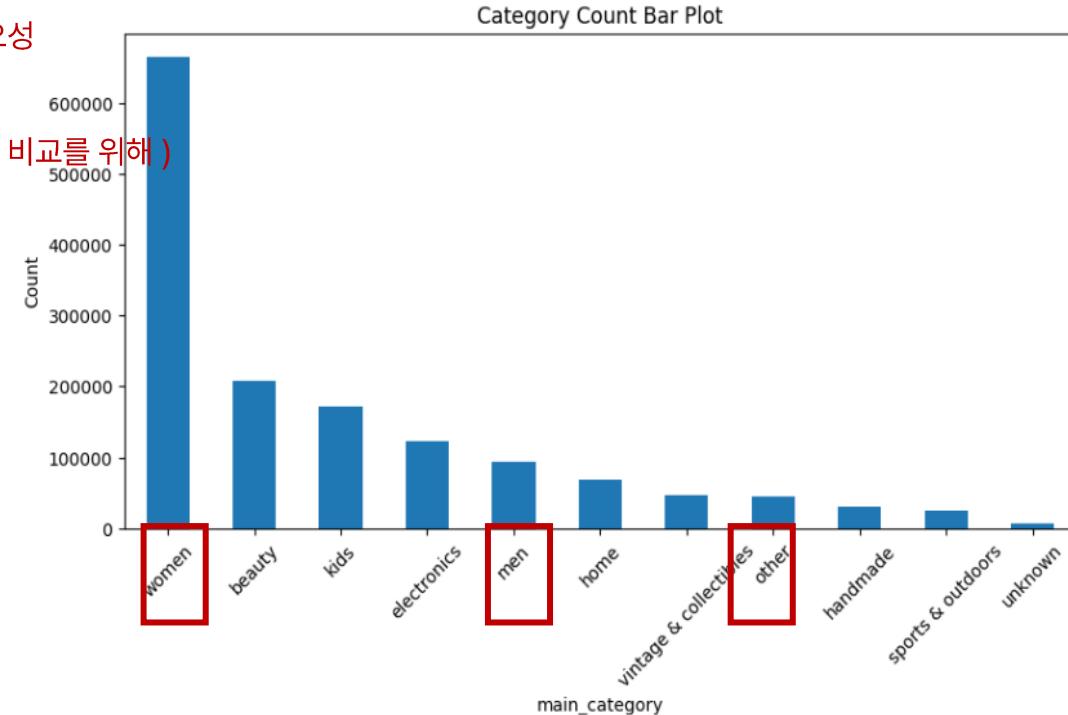
2-3. 세부 컬럼 확인_Category_name

| | category | count |
|----|---|-------|
| 0 | women/athletic apparel/pants, tights, leggings | 60152 |
| 1 | women/tops & blouses/t-shirts | 46349 |
| 2 | beauty/makeup/face | 34320 |
| 3 | beauty/makeup/lips | 29901 |
| 4 | electronics/video games & consoles/games | 26547 |
| 5 | beauty/makeup/eyes | 25200 |
| 6 | electronics/cell phones & accessories/cases, c... | 24668 |
| 7 | women/underwear/bras | 21254 |
| 8 | women/tops & blouses/tank, cami | 20270 |
| 9 | women/tops & blouses/blouse | 20269 |
| 10 | women/dresses/above knee, mini | 20068 |
| 11 | women/jewelry/necklaces | 19750 |
| 12 | women/athletic apparel/shorts | 19518 |
| 13 | beauty/makeup/makeup palettes | 19091 |
| 14 | women/shoes/boots | 18853 |

상위 카테고리로 묶을 필요성

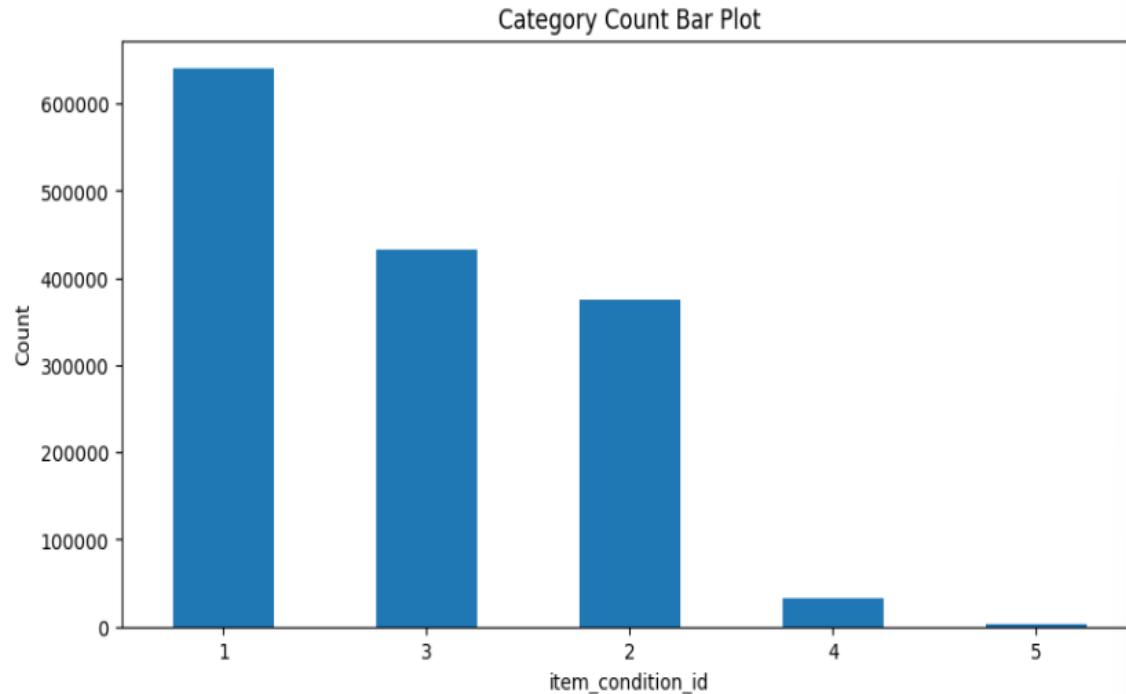
(카테고리별 트렌드 분석,

카테고리별 적정 가격인지 비교를 위해)



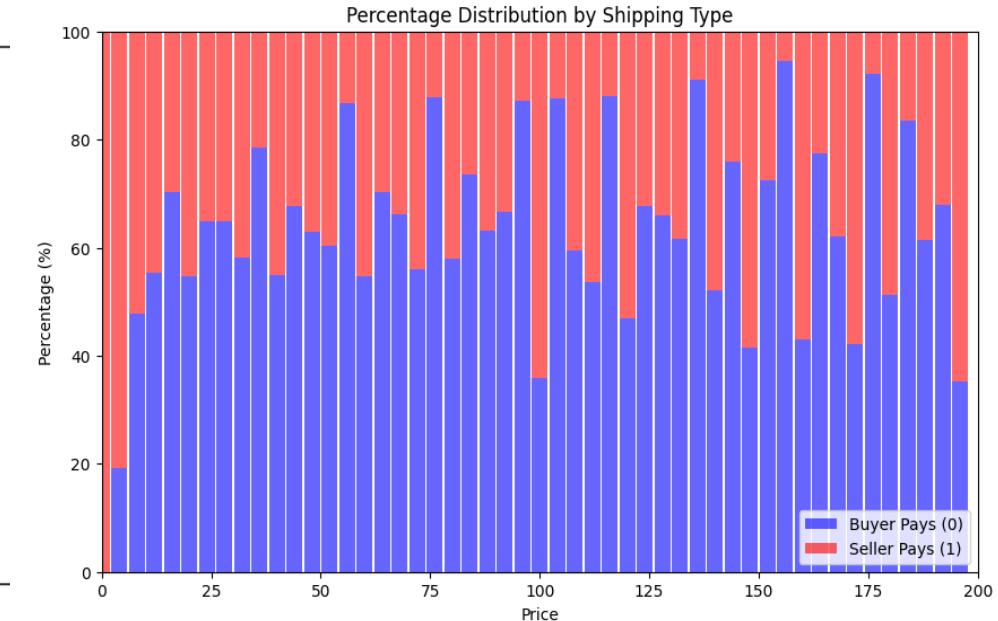
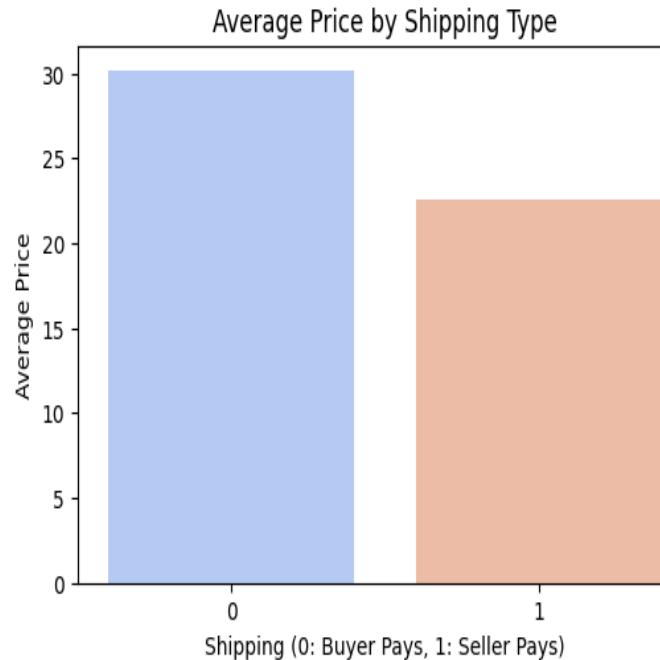
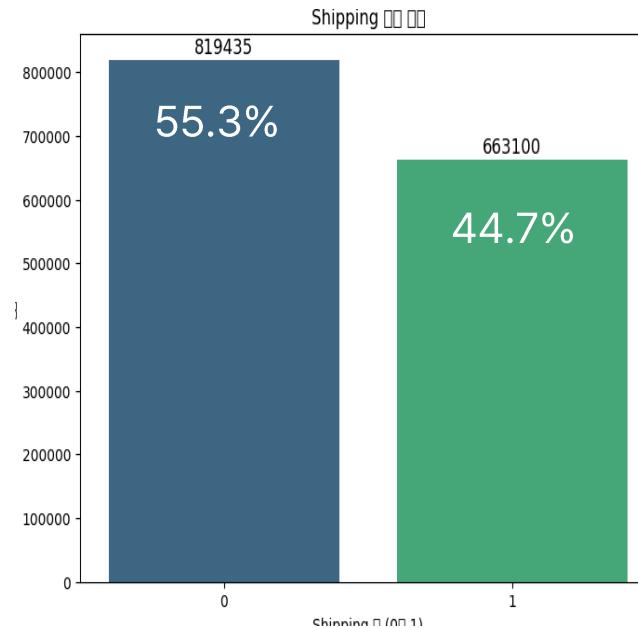
- Category명이 athletic apparel/pants,tights,leggings 같은 너무 세분화된 형식으로 이루어져 정제 필요하다 판단
- ' / ' 기준으로 split하여 맨 처음의 단어를 main_category를 생성
- women , men , others라는 모호한 카테고리명 존재. 2depth 컬럼을 가져와 상세화 (women's handbag 등)
- 카테고리별 트렌드 경향성 도 볼 수 있다면, 좋을 것 같다는 의견

2-4. 세부 컬럼 확인_Item_condition_id



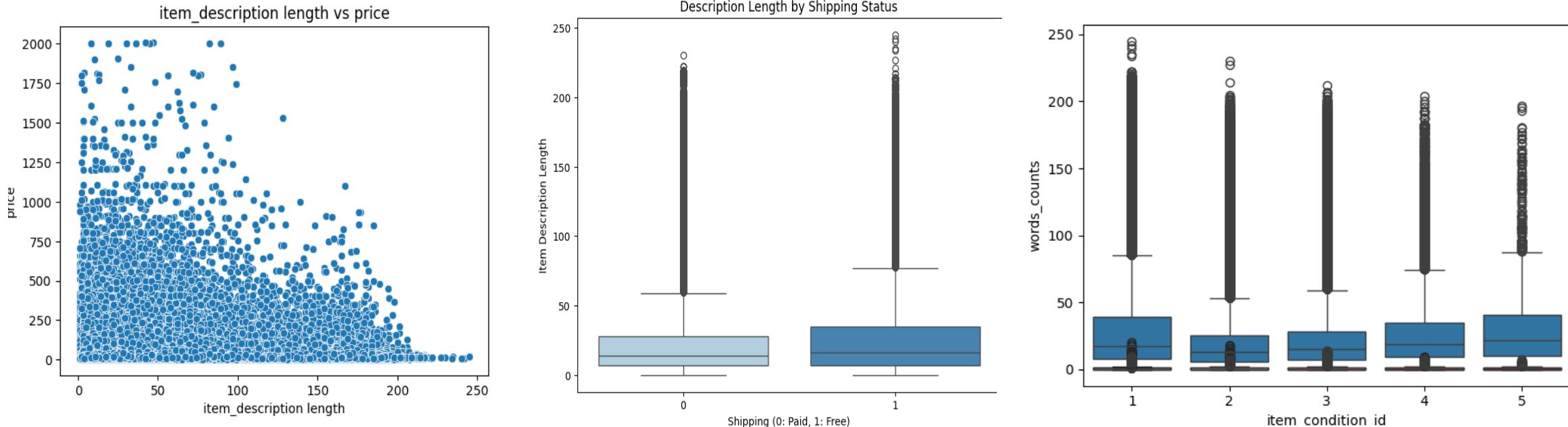
- 1번 값(새 상품급의 컨디션)의 개수가 가장 많음
- 4번, 5번 값이 너무 적어, 5번 값을 이상치로 제거 여부에 대해 논의하였으나
- 품질 가장 낮은 5번을 남기고 끊어서, 상 / 중 / 하 범주형 데이터를 생성 하는 것으로 결정

2-5. 세부 컬럼 확인_Shipping



- 0(구매자가 배송비 부담) 값이 55.3%로 높게 나타남
- 구매자가 배송비를 부담하는 상품의 평균 가격이 더 높게 나타남 → 구매자 배송비 부담 상품은 상대적 고가일 가능성 O
- 반면, 저가 상품은 판매자가 배송비를 부담하는 방식으로 판매량을 유지하려는 전략을 사용하는 것으로 보임

2-6. 세부 컬럼 확인_Item_description



- Item_description에서 null 값은 약 5%(8만건). 좌측 밀집되어 있어 대부분 짧은 설명이 많음
- 가격이 낮을 수록 설명이 짧은 경향. 일부 고가/저가의 상품은 긴 설명을 가지고 있는 경우도 있으나, 길이는 가격에 큰 영향 X
- 상품 상태에 따라 설명 길이 차이가 없음. 설명의 '길이'가 아닌 설명의 '내용'을 분석해 볼 필요있다 판단됨

앞으로 Feature Engineering 방향은?

Price

가격이 저가에 몰려 있음

Category

상위 카테고리로 나눌 필요성

Brand Name

빈도수 TOP 100 브랜드가 전체 73%

Condition

컨디션을 범주형 변수로 묶어본다면?

Item description

단어수는 경향성 나타나지 않음

카테고리별 평균 가격과 가격을 비교하는 컬럼을 추가하여 판매자가 제시한 가격이 적정 가격인지 확인해보자.

카테고리별 판매 가격 평균이나 카테고리별 경향성 파악 필요. 또는 카테고리별 트렌드 경향성도 볼 수 있다면?

TOP 100에 있는 브랜드 여부에 따른 범주형 컬럼 생성 필요

1(새상품급)과 5(안좋음)의 비중 차이가 워낙 커, 정확한 인사이트 도출을 위해 상/중/하 범주형 컬럼 생성 필요

단어수 보다 단어의 내용을 반영할 수 있는 긍부정어 관련 컬럼 생성 필요

주요 프로세스-3

**Feature
Engineering**

1. 파생 컬럼_긍·부정 Count

불용어 제거

```
[17] df['item_description'][1]
```

출력 결과

this slime is approximately 1.5 ounces, very soft, it is kind of sticky, this item also comes in an air tight container for easy storage.

→ nltk 패키지로 문장 토큰화 진행

→ This, is(be동사), 1.5(수치), 부사 등 제거 필요성 多
→ nltk의 stopwords 를 이용해 불용어 제거 진행

긍부정어 사전 구축

* LLM이나 머신러닝 기반 해보고 싶었지만
중고 거래 특성을 더 반영한 사전 기반으로 진행.

[(+)]Positive list]

- 제품 상태 : *flawless, perfect, great, excellent, clean etc..*
- 사용감 및 편안함 : *flawless, perfect, great, excellent, clean etc..*
- 디자인 및 스타일 : *flawless, perfect, great, excellent, clean etc..*
- 감성 표현 : *flawless, perfect, great, excellent, clean etc..*

[(-)]Negative list]

- 제품 상태 : *demaged, broken, defective, scratched etc..*
- 사용감 및 편안함 : *faulty, weak, flawed etc..*
- 디자인 및 스타일 : *ugly, cluncky, messy, unpolished etc..*
- 감성 표현 : *disappointing, annoying, questionable, smelly, horrible etc..*

→ TF-IDF를 통한 중요 단어 도출 기반 / GPT 도움을 받아 각 100개 리스트업
→ 어원화를 통한 mapping 진행

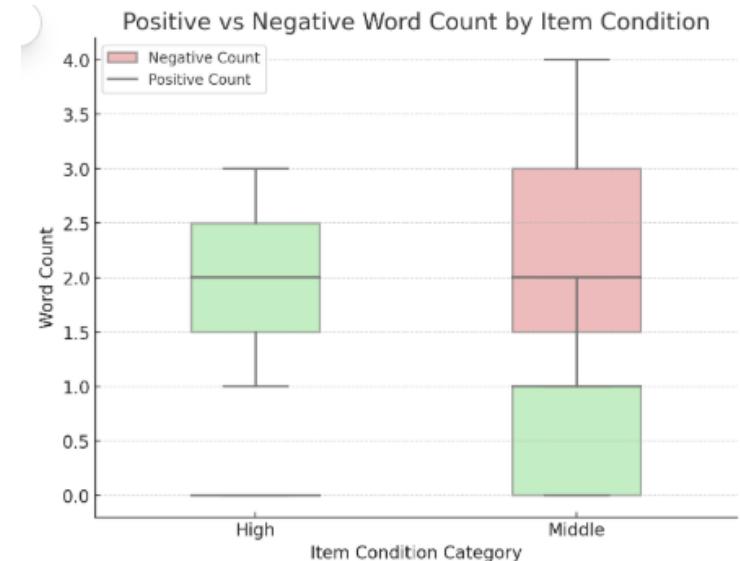
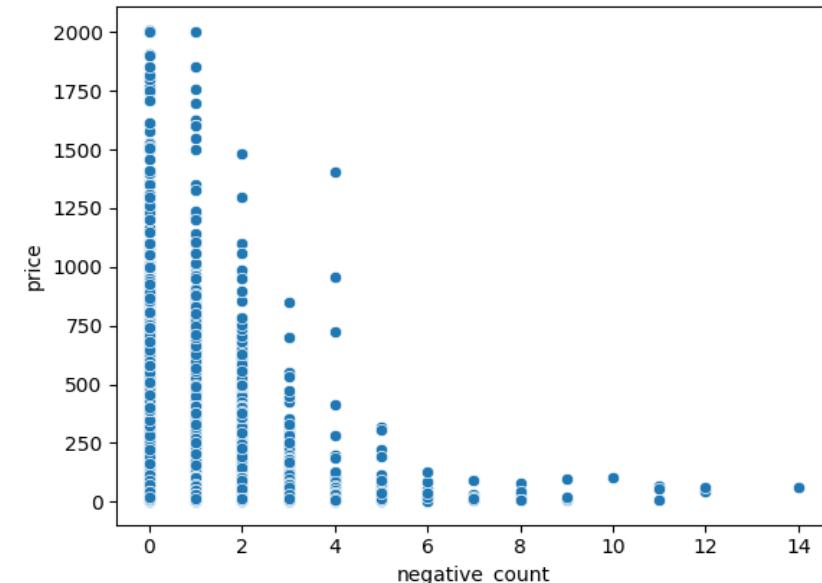
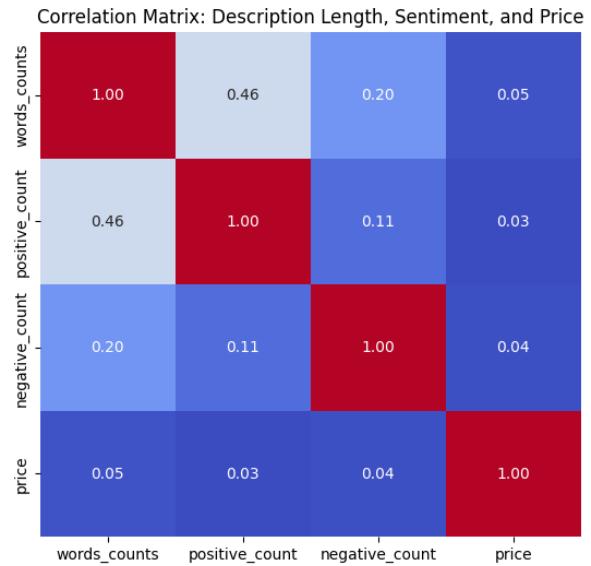
컬럼 결과

| positive_count | positive_words | negative_count | negative_words |
|----------------|-------------------|----------------|----------------|
| 2 | ['super', 'cute'] | 0 | 0 |
| 0 | 0 | 1 | ['worn'] |

| positive_count | positive_words | negative_count | negative_words |
|----------------|-------------------|----------------|----------------|
| 2 | ['super', 'cute'] | 0 | 0 |
| 0 | 0 | 1 | ['worn'] |

1. 파생 컬럼_긍·부정 Count

그래프



- 긍부정 단어수와 가격간의 높은 상관관계는 없지만, 부정어와 가격간의 상관관계가 소폭 높음
- 부정적인 단어가 많아질 수록 가격이 낮아지며, 고가 상품일 수록 부정어가 줄어드는 경향을 확인
- 상품 상태가 좋을수록 긍정적인 표현이 많이 사용되나, 중간 이하일 수록 부정적인 표현이 다수 발견됨

2. 파생 컬럼_main_category

기존 카테고리(1,288개)

| Count |
|---|
| women/athletic apparel/pants, tights, leggings |
| women/tops & blouses/t-shirts |
| beauty/makeup/face |
| beauty/makeup/lips |
| electronics/video games & consoles/games |
| beauty/makeup/eyes |
| electronics/cell phones & accessories/cases, c... |
| women/underwear/bras |
| women/tops & blouses/tank, cami |
| women/tops & blouses/blouse |
| women/dresses/above knee, mini |
| women/jewelry/necklaces |
| women/athletic apparel/shorts |
| beauty/makeup/makeup palettes |
| women/shoes/boots |
| beauty/frAGRANCE/women |
| beauty/skin care/face |
| women/women's handbags/shoulder bag |
| men/tops/t-shirts |
| women/dresses/knee-length |
| women/athletic apparel/shirts & tops |
| women/shoes/sandals |
| women/jewelry/bracelets |

/로 split 하여 총 5개 컬럼 도출

| main_category | sub1_category | sub2_category | sub3_category | sub4_category |
|----------------|---------------|---------------|---------------|---------------|
| tops & blouses | tops | t-shirts | unknown | unknown |



Main_category 뒤
women / men / others 등 모호한 컬럼 정제



Sub1_category 를 가져와 구체화



공통으로 묶일 수 있는 카테고리 통합

Tops & Blouse → Tops 로 통일

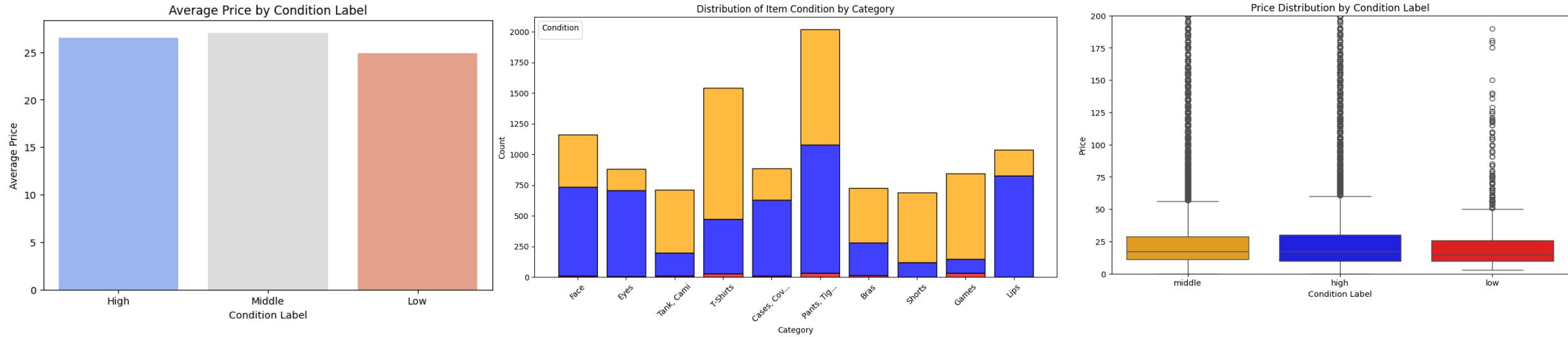
Girls 4+, boys 6+ 등 → Kids 로 통일

최종 카테고리(37개)

| main_category |
|------------------------|
| athletic apparel |
| automotive |
| beauty |
| books |
| coats & jackets |
| daily & travel items |
| diapering |
| dresses |
| electronics |
| feeding |
| gear |
| handmade |
| home |
| jeans |
| jewelry |
| kids |
| maternity |
| men's accessories |
| nursery |
| office supplies |
| other |
| pants |
| pet supplies |
| shoes |
| shorts |
| skirts |
| sports & outdoors |
| suits |
| sweaters |
| sweats & hoodies |
| swimwear |
| tops & blouses |
| toys |
| underwear |
| vintage & collectibles |
| women's accessories |
| women's handbags |

3. 파생 컬럼_Condition_상중하

Condition 그래프

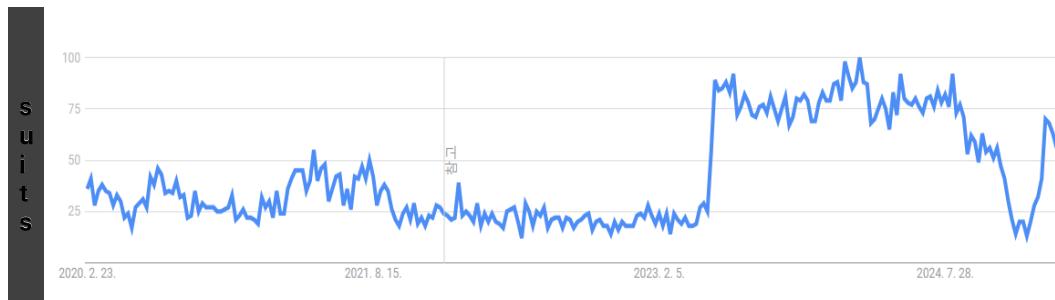
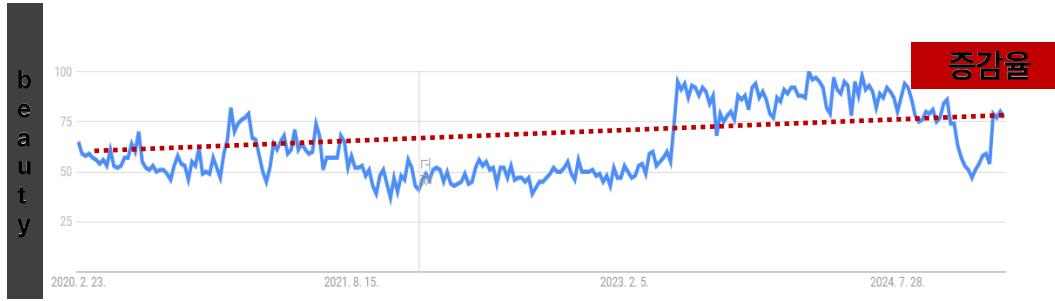


- 1(새거) → 'High' // 2와 3 → Middle // 4와 5 → Low로 변경하여 총 High / Middle / Low로 범주형 컬럼 추가
- 상품 상태가 좋을수록 가격이 높아지는 경향이 있지만, 중간 상태의 제품이 오히려 평균 가격이 가장 높은 경우도 나타남
- 카테고리별 상품 상태 분포도를 확인 해보니 High는 뷰티, Middle은 의류, Low는 전체적으로 적은 편으로 파악 됨
- 상품 상태와 가격의 관계를 분석하니 High 상태의 제품이 항상 비싼 것은 아님. Middle 상태의 제품이 더 높은 평균 가격을 보이는 경우도 존재함

4. 파생 컬럼_트렌드 지수

트렌드 지수

37개 main category별 5년전부터 현재까지 검색량 비중 증감율을 통해 사람들의 해당 카테고리의 관심도를 반영한 트렌드 지수 컬럼을 추가



출처 : 구글트렌드

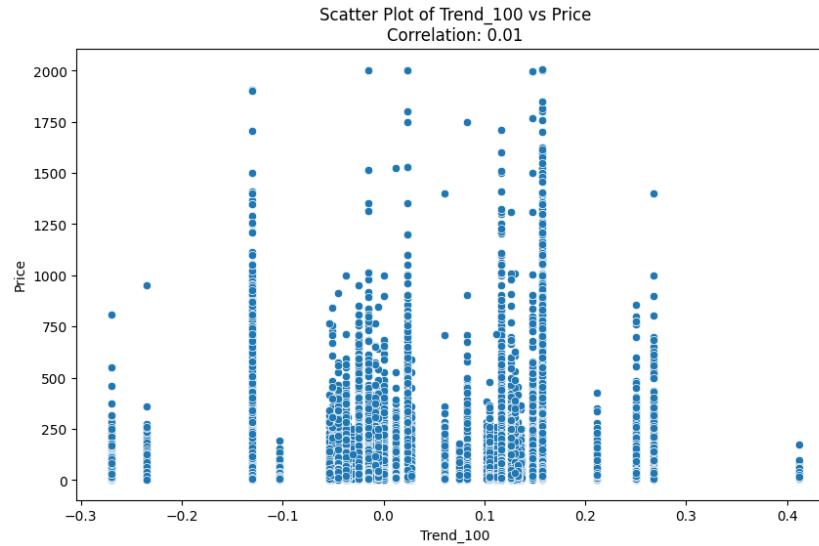
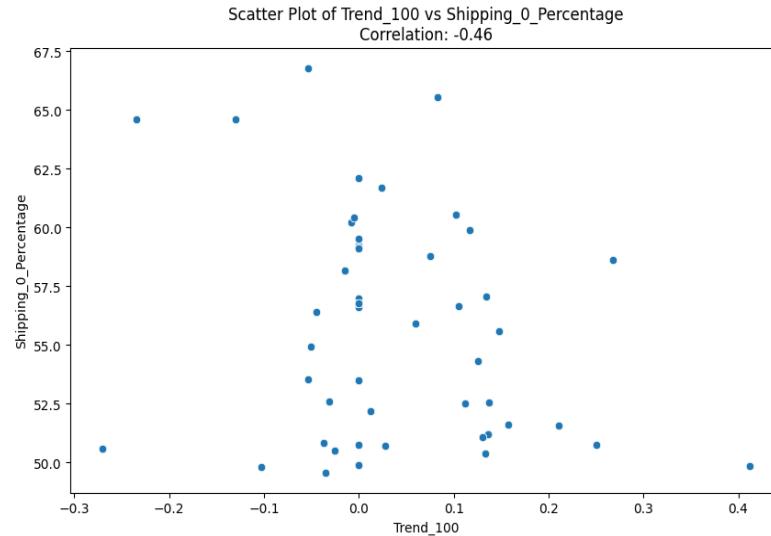
트렌드 지수 상위 카테고리

Athletic apparel 카테고리가 가장 증감율이 높게 나타남
실제로, 애슬레저 룩 등 스포츠웨어 브랜드의 인기도가 반영된 것으로 판단됨

| no | category | per |
|-----|------------------------|-------|
| 1 | athletic apparel | 0.25% |
| 2 | sweats & hoodies | 0.21% |
| 3 | women's handbags | 0.18% |
| 4 | women's accessories | 0.15% |
| 5 | shorts | 0.14% |
| 6 | gear | 0.14% |
| 7 | pet supplies | 0.13% |
| 8 | jeans | 0.13% |
| 9 | coats & jackets | 0.13% |
| 10 | men's accessories | 0.13% |
| 11 | vintage & collectibles | 0.12% |
| 12 | pants | 0.11% |
| 13 | kids | 0.11% |
| 14 | ... | ... |
| ... | ... | ... |

4. 파생 컬럼_트렌드 지수

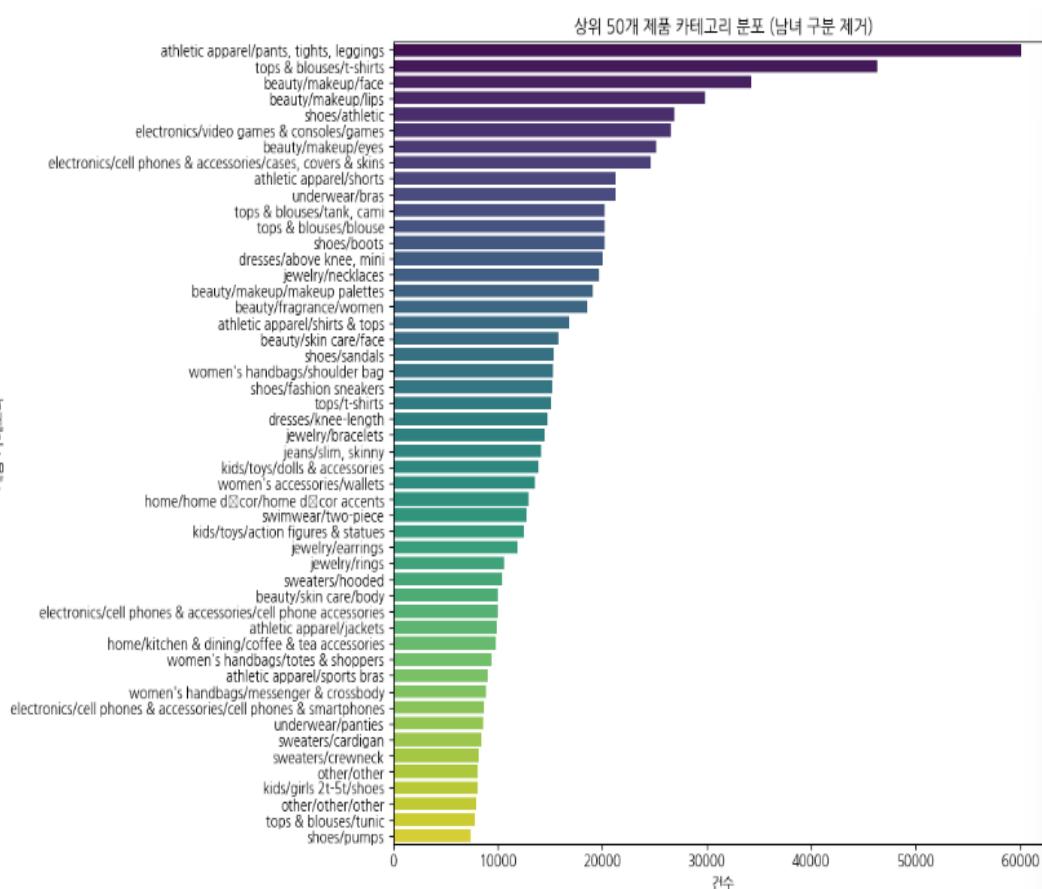
트렌드 지수 그래프



- 트렌드와 가격의 상관관계(완)은 0.01로 제품의 트렌드 변화와 가격 간의 관계 X. 즉 트렌드가 변화해도 큰 영향 없음
- 배송비와의 상관관계(-0.46)가 높음. 제품의 트렌드가 증가하면 구매자가 배송비를 지불할 확률이 줄어듬
- 인기있는 제품일 수록 가격은 크게 변하지 않지만, 구매자가 배송비를 지불하지 않으려는 경향이 강해진다고 해석 가능

5. 파생 컬럼_Category per Price 평균

Category별 Price 평균



가격이 적정하게 책정되었는지를 얻기 위해,
비교 컬럼으로서 Category별 Price 평균을 구함.

Main_category로는 범위가 너무 넓어 값이 왜곡될 수 있어,
Category_name으로 범위를 좁혀 Price 평균을 도출함

왼쪽의 그래프는 Category_name별 빈도수로서, 정규성을 떨 것으로 판단됨

분석 총 의견

공부정어

공부정 단어수와 가격간의 높은 상관관계는 없지만,
부정어가 많아질 수록 가격이 낮아지며, 고가일 수록 부정어가 줄어드는 경향 확인

Main_Category

최종 카테고리 37개로 확정,
기타 변수들과 함께 크로스 분석 이용

Condition_상중하

상품 상태가 좋을 수록 가격이 높아지는 경향이 있음.
High는 뷰티, Middle은 의류 카테고리가 많았음.

트렌드 지수

트렌드가 변화해도 가격은 큰 영향은 없으나,
트렌드 경향성이 높을 수록 구매자가 배송비를 지불하지 않으려고함

Price 평균

카테고리별 Price 평균을 구해, 데이터 상 책정된 가격이 적정 가격인지 비교하고자함

주요 프로세스-4

데이터 활용 및 판매 예상율

지금까지 만든 데이터 기반으로,
판매 가능한지 예상 할 수 있다면?

1. 판매 예상율 분석_Price

```

import matplotlib.pyplot as plt

# 전체 개수 대비 비율(%) 계산
category_counts_percent = (category_counts / category_counts.sum()) * 100

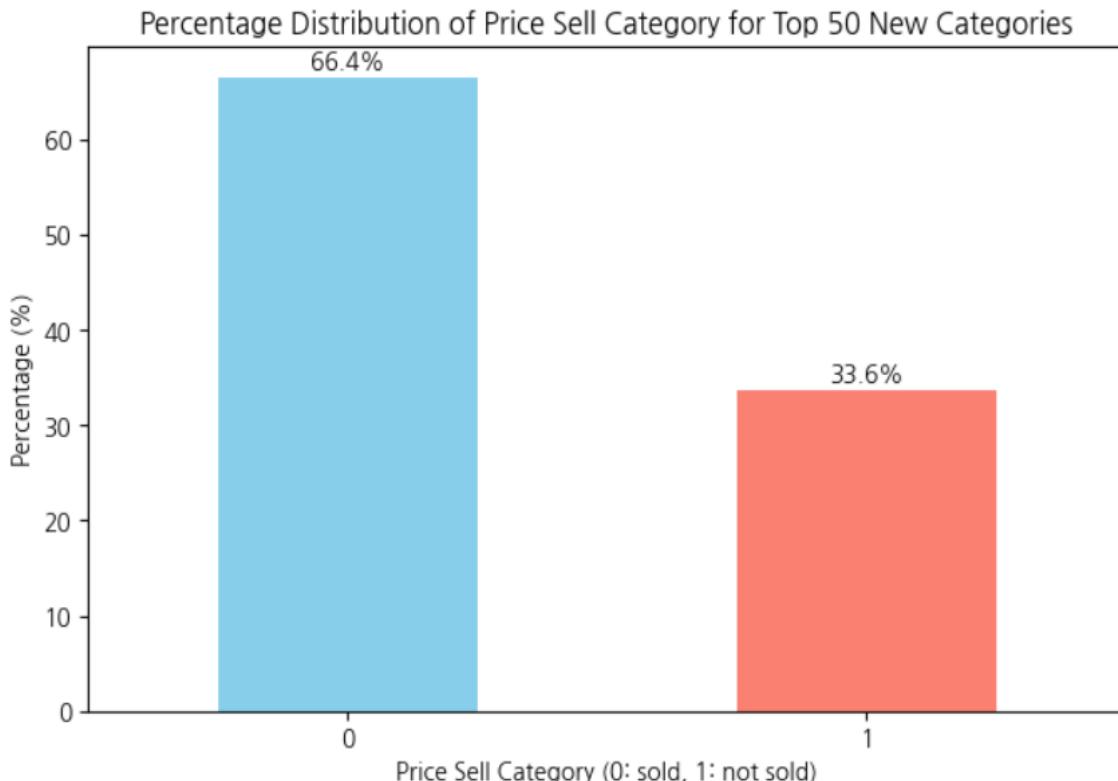
# 그래프 그리기
plt.figure(figsize=(8, 5))
category_counts_percent.plot(kind='bar', color=['skyblue', 'salmon'])

# 라벨 및 제목 설정
plt.xlabel("Price Sell Category (0: sold, 1: not sold)")
plt.ylabel("Percentage (%)")
plt.title("Percentage Distribution of Price Sell Category for Top 50 New Categories")
plt.xticks(rotation=0)

# y축 값 퍼센트 표시
for index, value in enumerate(category_counts_percent):
    plt.text(index, value + 1, f"{value:.1f}%", ha='center', fontsize=10)

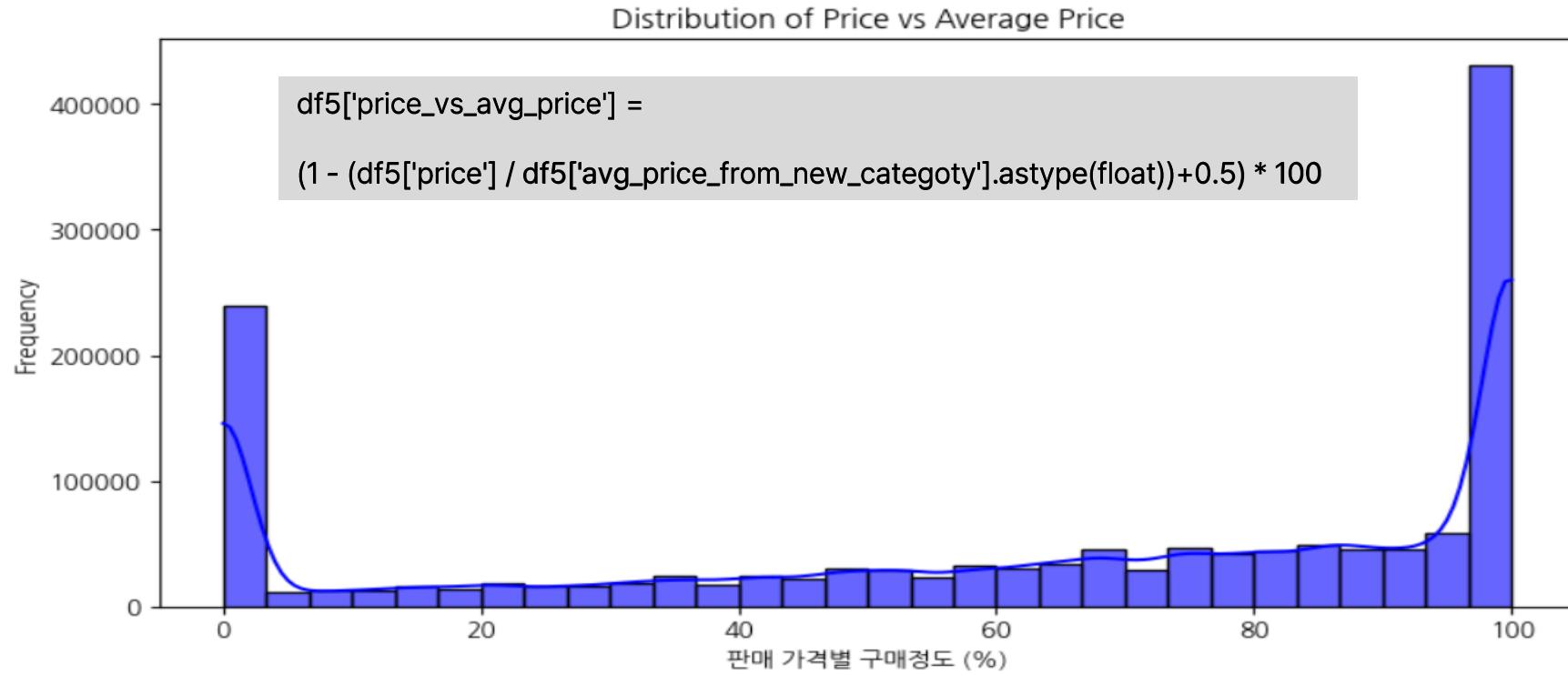
# 그래프 출력
plt.show()

```



- 이전 카테고리 분류를 기준으로 각 카테고리별 price의 평균 값 분석 진행.
- 카테고리 평균값보다 작거나 같은 값은 판매될 것이라고 이분법적으로 가정하고 판매 여부를 그래프로 확인.
- Price 기준으로 판매될 가능성이 66퍼센트임이 확인. 카테고리별 가격 분포가 왼쪽으로 치우친 구조일 것임이 예상됨.

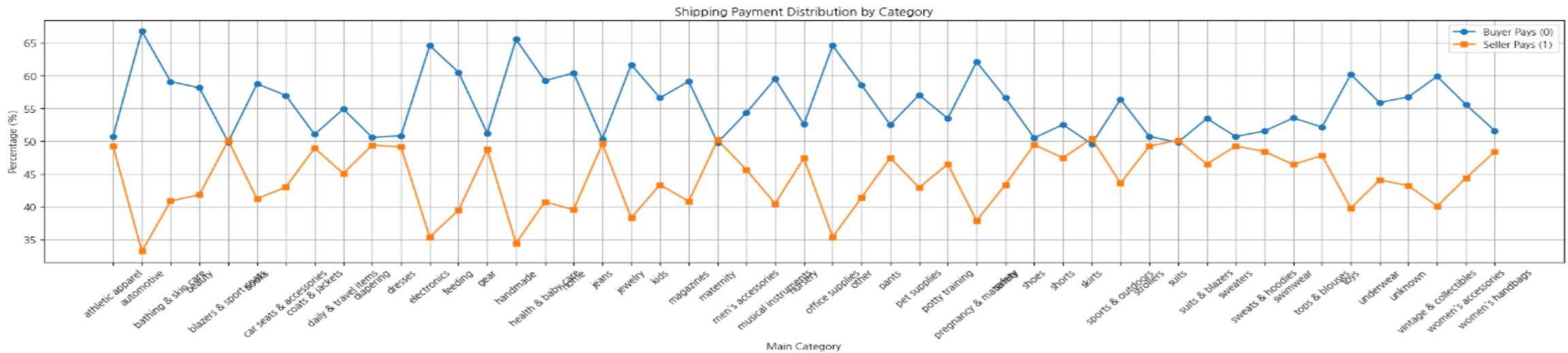
2. 판매 예상율 분석_Price



- 단어 전처리 과정을 통해 확인한 제품 설명란에 긍정적인 단어와 부정적인 단어의 개수를 활용하여 긍정적인 단어가 많으면 좋은 제품일 것이라는 척도를 만듦.
- 긍정적 단어 개수/(긍정단어개수 + 부정단어개수) 식을 통해 긍정 단어 비율을 확인.

3. 판매 예상을 분석_Shipping

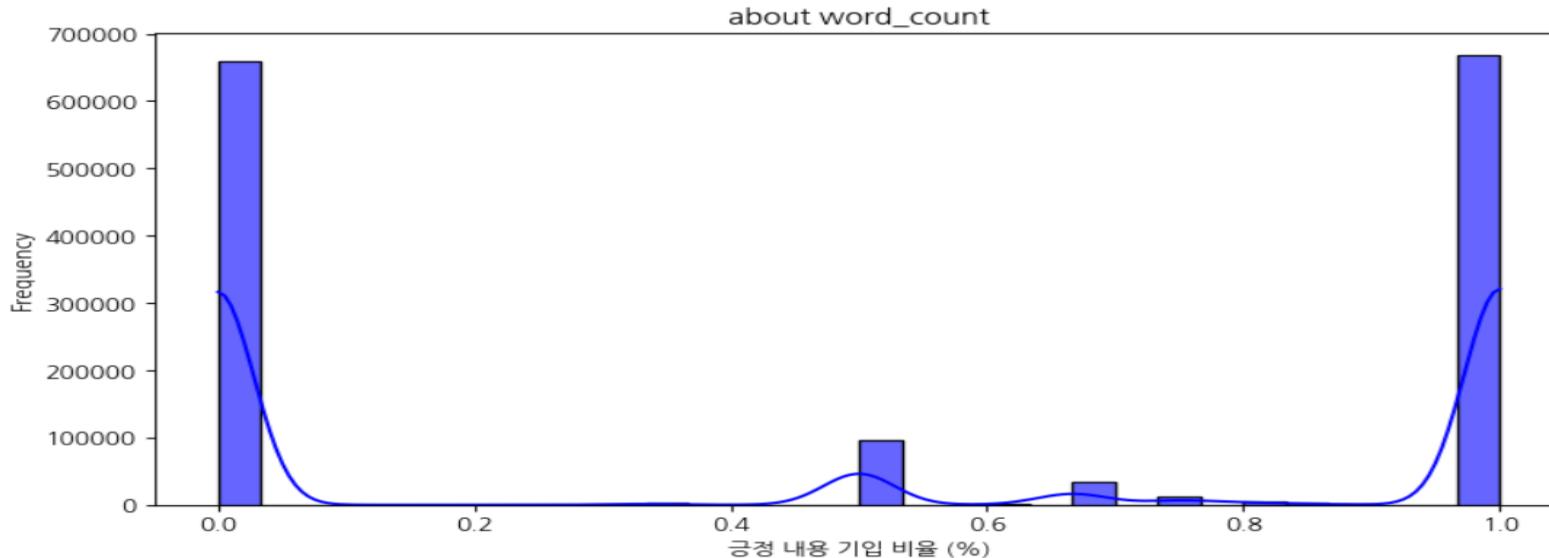
```
df5['selected_shipping_percentage'] =  
df5.apply(lambda row: row['shipping_0_percentage'] if row['shipping'] == 0 else row['shipping_1_percentage'], axis=1)
```



- 카테고리별 Shipping의 가격 평균을 퍼센테이지로 변환하여 구매자 부담(0,파랑) 과 판매자 부담(1, 주황)으로 카테고리별 shipping 판매가격 분포. 0인 값이 큰 경향성을 볼 수 있음
- 이는 shipping이 0이면 값이 크다고 해도 구매할 가능성이 상대적으로 클 것임을 의미하며 정도를 퍼센테이지로 구하였기에 판매하고자 하는 제품의 값이 0인지, 1인지에 대한 판매율 가능성의 척도로 사용.

4. 판매 예상을 분석_Words Counts

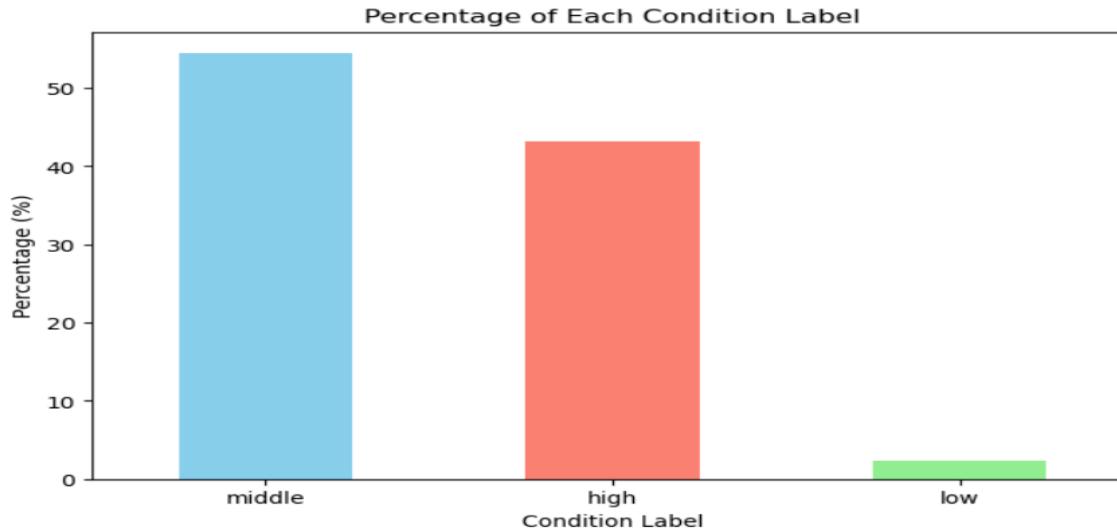
```
df5['words_counts_sum'] = df5['positive_count'] / ((df5['positive_count'] + df5['negative_count']).replace(0, 1))
```



- 단어 전처리 과정을 통해 확인한 제품 설명란에 긍정적인 단어와 부정적인 단어의 개수를 활용하여 긍정적인 단어가 많으면 좋은 제품일 것이라는 척도를 만듦
- 긍정적 단어 개수/(긍정단어개수 + 부정단어개수) 식을 통해 긍정 단어 비율을 확인

5. 판매 예상을 분석_제품 퀄리티

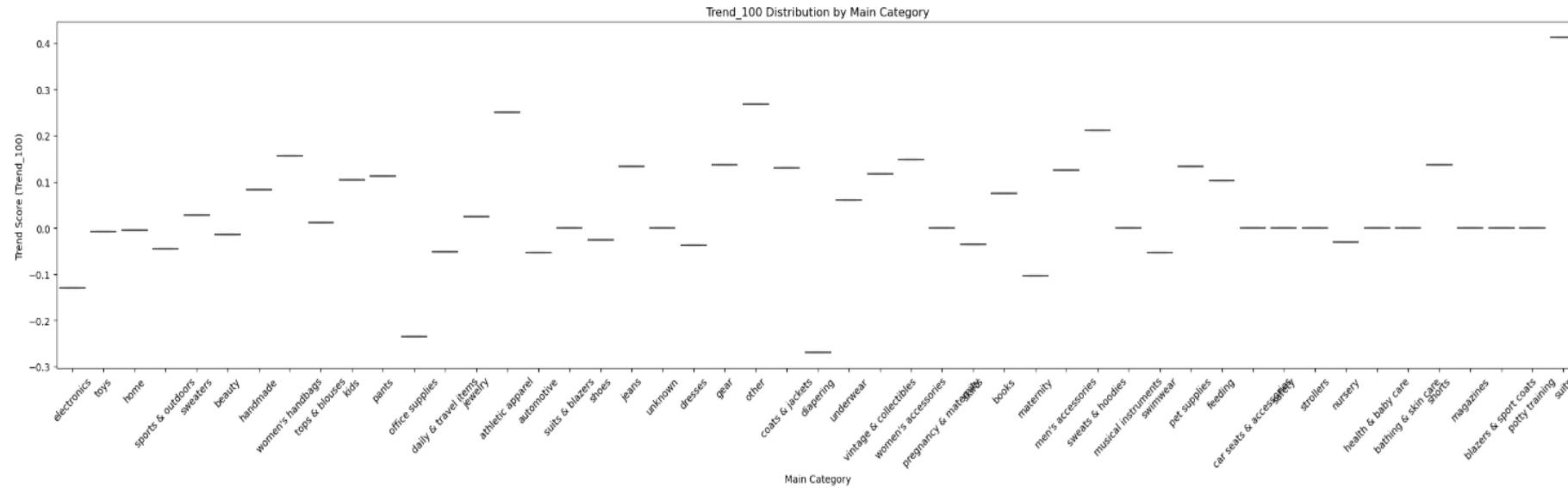
```
df5['condition_label_num'] = df5['condition_label'].map({ 'high' : 1, 'middle' : 2/3, 'low' : 1/3 })
```



- 제품 퀄리티를 high를 최고점 1, middle은 중간점 2/3, low를 최저점 1/3으로 수치화함

6. 판매 예상을 분석_Trend

```
df5['Trend_100'] = df5['Trend_100'].clip(-10, 10)
```



- 카테고리별 트렌드 분포는 최댓값이 10이 나타나도록 진행 (10 이상의 값은 10으로 치환)
- - 에 대한 부분을 0~10 값으로 수치 변환 진행하지 않았음 (이유 기억안남. 성환님께 물어봐야...)

판매 예상을 계산

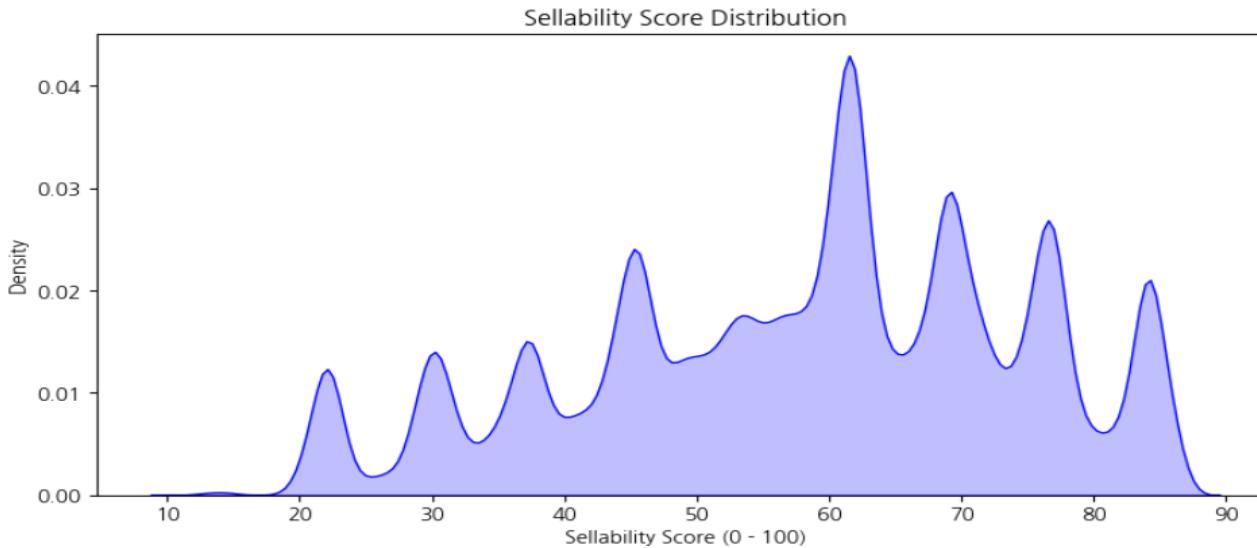
판매 예상을 =

$$\{ \text{Price} * 0.4 + \text{상세 설명} * 0.15 + \text{제품 퀄리티} * 0.25 \\ + \text{Trend} * 0.1 + \text{shipping 여부} * 0.1 \}$$

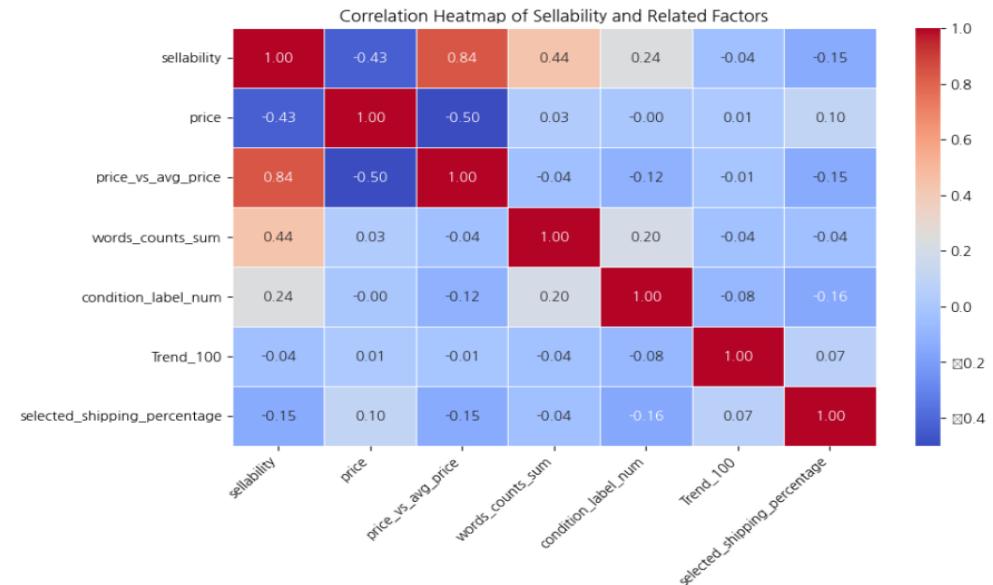
- 카테고리별 가중치에 들어가는 값은 최대가 100이 되어야함
- 제품이 판매되기 위해선 Price가 제일 중요하다 판단. 그 다음으로 제품 퀄리티가 중요하며 다음순서로 상세설명이 중요할 것임을 가중치로 표현함
- 트렌드와 shipping은 동일한 위치의 중요도를 가질 것임을 가정함

판매 예상을 _INSIGHT

판매 예상을 분포도



판매 예상을과 변수간 상관성



- 오른쪽 그래프는 판매 예상을 분포도이며, 특정 구간에서 급상승하는 경향이 보여짐
- 판매 예상을과 Price_avg(0.84)가 높은 상관관계를 보이며, 가격이 평균보다 낮을 수록 더 잘 팔릴 가능성이 높음
- Price(-0.4)로 가격이 높을수록 판매 가능성이 낮아, 중고거래에서 합리적인 가격이 중요함
- 이는 중고거래 시장에서 경쟁력있는 가격이 중요한 요소라는 것을 반증
- 상세한 상품 설명이 소비자의 신뢰를 높이고 구매 결정을 유도할 가능성이 높음
- 제품상태, 트렌드 도 미미하나 상관관계 있음

주요 프로세스-5

판매 예상을 적용시,
기대효과

판매율 예상을 적용시 이용자/기업 관점 기대효과



기업 관점

- 판매 예상을 통해 잘 팔릴 가능성이 높은 상품을 추천하여 판매 촉진
- 예상을 통해 판매율이 낮은 상품에 대한 프로모션을 제안함으로 정책 최적화
- 판매율 높은 상품은 소비자의 만족도를 높이고, 거래 성공률을 증가시켜, 플랫폼의 신뢰도를 상승

이용자 관점

- 이용자 관점에서 어떤 상품을 신뢰할 수 있는지 미리 확인하고 거래 위험 감소
- 어떤 상품이 잘 팔릴 수 있는지 안다면, 더 빠르게 의사결정 가능함
- 소비자의 더 합리적인 가격과 신뢰할 수 있는 정보를 바탕으로 구매 가능

판매 예상을 시스템 도입을 통해 플랫폼 신뢰도를 높이고, 소비자 만족도도 동시에 향상 가능

회고

- # 헌수
- 이번 프로젝트를 진행하면서 데이터 분석에 대한 이해가 조금 더 깊어졌다. 이전에는 단순히 코드 작성에 집중했다면, 프로젝트를 통해 데이터 분석의 흐름과 팀워크의 중요성을 직접 경험할 수 있었다. 팀원들과의 소통을 통해 다양한 시각을 배우고, 협업의 가치도 다시금 깨닫게 되었다. 오마카세 팀원 여러분 정말 고생 많으셨습니다! 정말 많이 배웠어요!! 감사합니다!!!
- # 성환
- 이번 프로젝트를 통해서 팀원들 각각의 뛰어난 능력을 알았으며, 혼자 했다면 1년 정도 시간?이 소요되는 심도 깊은 대화와 의사소통의 중요성, 코딩 버전업관리, 해당작업의 공동 목표라는 중요성을 배운것 같다..
- # 상혁
- 데이터 분석에 있어서 시간이 부족한 느낌이 들었다. 분석 진행을 통해 데이터의 이해도가 늘어나면서 나타나는 추가 의문들을 모두 진행하지 못해서 보다 퀄리티 있는 결과를 보여주지 못하고 마무리된 점이 아쉽다.
- # 유빈
- 열심히 했는데, 항상 아쉬운 분석이다. 메인 카테고리의 분류의 정확성을 높이고, 자연어처리 긍부정어 분류를 좀 더 정교하게 해보고 싶다. 팀원들의 도움이 너무 컸다. 부족한 부분을 채워주는 팀원을 만난 것이 행운이었고 밤샘의 의미를.....

Thank you for listening.

