# 인물소개

- 유튜브 크롤링
- 자연어처리
- Power BI 시각화
- 프로젝트 발표

김창연

- 유튜브 크롤링
- 자연어처리
- Power BI 시각화
- 프로젝트 발표

김보경

- 유튜브 크롤링
- 자연어분석처리
- 프로젝트 발표

이상은

# 인물소개

## 강명진

- 통계자료 시각화

## 박상준

- 유튜브 크롤링
- 게시글 빈도수 시각화
- 주가분석
- 보고서 작성

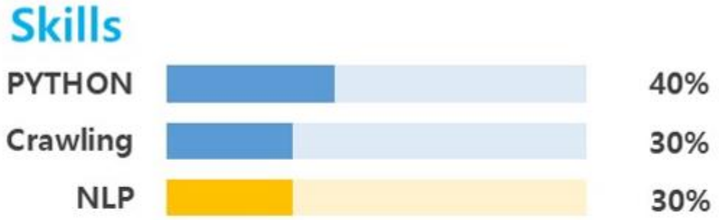
## 김소연

- 유튜브 크롤링
- 게시글 빈도수 시각화
- 자연어처리

# NLP 기반의
# 신제품 출시를 위한 분석 모델


**유통령 :**

강명진, 김보경, 김소연, 김창연, 박상준, 이상은



# 목차

PART. 1

# 배경

01

1. 현황
2. 목적 및 주제
3. 필요성

# 1. 현황

KBS WORLD RADIO

뉴스 주제별 엔터테인먼트

경제

## 코로나19에 라면 매출 최대...'집콕'에 봉지면 늘고 컵라면 줄고

Write: 2020-08-20 10:36:35 Update: 2020-08-20 10:39:43

출처: 농림축산식품부 (2018-2019)

아시아경제 경제 | '코로나19로 세계는 집콕'...미국도 한국도 집에서 신라면 먹었다

일반

## '코로나19로 세계는 집콕'...미국도 한국도 집에서 신라면 먹었다

최종수정 2020.08.22 09:30 기사입력 2020.08.22 09:30 댓글 1

올해 국내 상반기 라면시장 1조1300억 사상 최대, 온라인 매출 2배↑

신라면, 짜파게티 등 두 자릿수 성장...집콕 확산에 봉지라면 인기

중앙일보 오피니언 정치 경제 사회 국제 문화 스포츠

## 경제

경제정책 산업 금융증권 부동산 과학미래 글로벌경제 고용노동

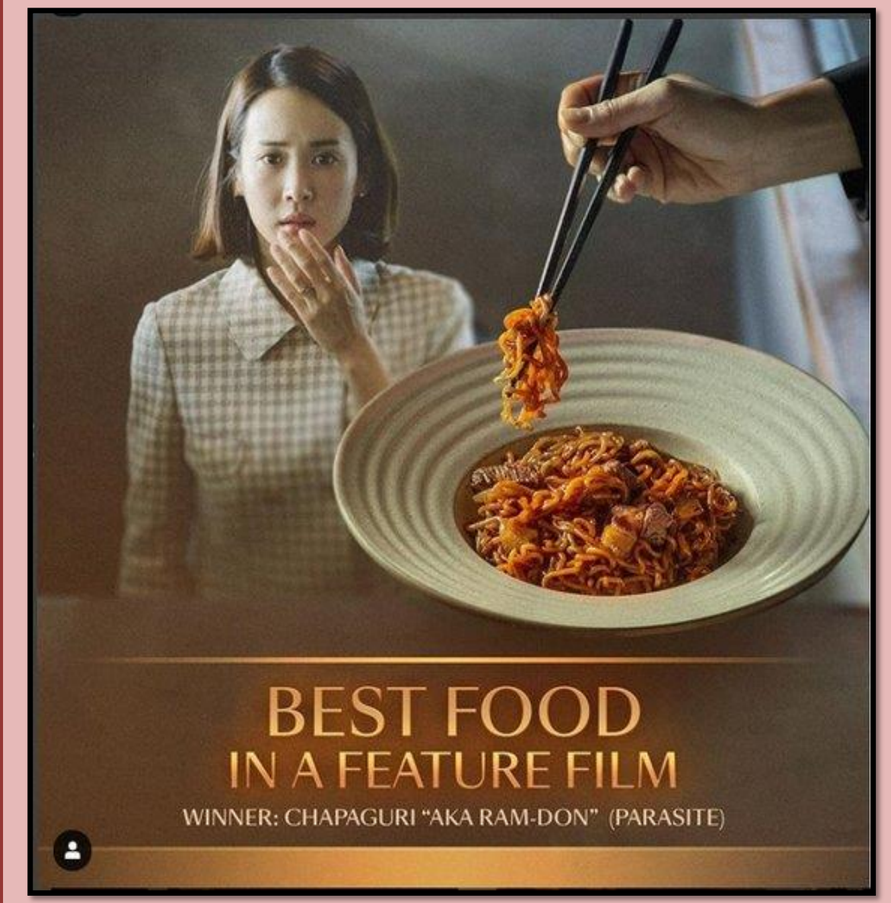
## 코로나19도 뚫은 한국 라면 인기...K-농식품 수출 4.4% 증가

[중앙일보]입력 2020.07.02 11:00

임성빈 기자

[ 그래프2 ] 주로 구매하는 간편식 품목

# 1. 현황




**라면**이 식품 트렌드를 반영한
대표적인 시장

# 2. 목적

트렌드를 기반으로 기획된 상품의 시장성 분석

# 2. 주제

NLP 기반의 신제품 출시를 위한 분석 모델

# 3. 필요성

1

트렌드를 반영한 신제품이나 출시한 브랜드가
실제로 어떤 성과를 내고 장기적으로 매출에 긍정적인 영향을 끼칠지 가늠

2

각종 그래프로 분석결과를 이해하기 쉽게 시각화

3

수천개의 온라인 게시글과 100,000여개의 유튜브 댓글을
기반으로 해 소비자 입장을 생생하게 반영한 의사결정 지원 모델

PART.2

# 프로젝트 소개

02

1. 개발환경
2. 분석 프로세스
3. 분석 모델링

# 1. 개발환경

## Language

Python

## Front end

Power BI


## Back end

TensorFlow

pandas

NumPy

KoNLPy

## Database

# 2. 분석 프로세스

## [ 주제선정 ]

NLP 기반의
신제품 출시를
위한 분석모델

## [ 정보수집 ]

BeautifulSoup
Request

blog
YouTube
Daum
TISTORY

## [ 개발구현 ]

Tensorflow
Konlpy
SQLite3

3. OKT 토큰화와 필요없는 조사 제거

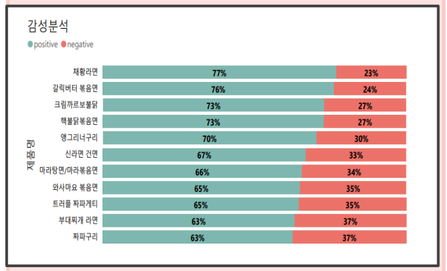
```
In [41]: from konlpy.tag import Okt
okt=Okt()
# mecab 사용형 토큰화, 합글자 결거, stop_word 정거
stop_word = ['에서','에서','이다','에게','으로','이랑','까지','부터','하다']
stopPos = ['Determiner','Adverb','Conjuctions','Josa','PreEomi','Eomi','Suffix',
           'Punctuation','Foreign','Alpha','Number','Unknown','KoreanParticle']
def token_okt(text):
    text = okt.pos(text)
    text = [i for i in text if len(i[0])>1]
    text = [i for i in text if i[0] not in stop_word]
    text = [i[0] for i in text if i[1] not in stopPos]
    return text

In [42]: # 토큰화 + 토큰리스트 생성
def make_tokens(df):
    df['tokens'] = ''
    tokens_list=[]
    for i, row in df.iterrows():
        if i%5000==0:
            print(i,'/',len(df))
        token = token_okt(df['Comments'][i])
        df['tokens'][i] = ' '.join(token)
    return df

df = make_tokens(df)
```

## [ BI 분석·시각화 ]

Matplotlib
Seaborn
Power BI




## [ 모델링]

신제품
출시를 위한
의사결정
지원 모델

# 2. 분석 프로세스

[ 정보 수집 ]

## [ Selenium으로 유튜브 댓글 크롤링 ]

**1. 필요한 라이브러리 불러오기**

In [8]:
```
from selenium import webdriver as wd
from bs4 import BeautifulSoup
import time
import requests
from bs4 import BeautifulSoup
import pandas as pd
```

**2. 페이지 내 댓글 크롤링하는 함수**

In [3]:
```
def youcrawl(url):

    driver = wd.Chrome(executable_path="../chromedriver.exe")
    driver.get(url)
    last_page_height = driver.execute_script("return document.documentElement.scrollHeight")

    while True:
        driver.execute_script("window.scrollTo(0, document.documentElement.scrollHeight);")
        time.sleep(3.0)
        new_page_height = driver.execute_script("return document.documentElement.scrollHeight")

        if new_page_height == last_page_height:
            break
        last_page_height = new_page_height

    html_source = driver.page_source

    driver.close()

    soup = BeautifulSoup(html_source, 'lxml')
    youtube_comments = soup.select('yt-formatted-string#content-text')

    str_youtube_comments = []

    for i in range(len(youtube_comments)):
        str_tmp = str(youtube_comments[i].text)
        str_tmp = str_tmp.replace('\n', '')
        str_tmp = str_tmp.replace('\t', '')
        str_tmp = str_tmp.replace(' ', '')

        str_youtube_comments.append(str_tmp)
    return str_youtube_comments
```

크롤링 데이터

1. 유튜브 제품 리뷰 댓글
2. 네이버 블로그 리뷰수
3. 온라인 쇼핑몰 상품평 갯수

Out [23] :

| | Comments |
|---|---|
| 0 | 왜들 미역가지고 난리들이랴..미역 어디서 싸게 나왔나 |
| 1 | 진짜쫄면 소스인듯.....미역을 왜케 ;;;;;; |
| 2 | gs 25 왕의 밥상 도시락이 궁금합니다. \n고물 대왕님 리뷰 부탁 좀 .. 가격... |
| 3 | 컨텐츠도 리뷰도 제일 상세하고 여타 다른 유투버보다 세밀해서좋은데 파이리~~~만빼면... |
| 4 | 이쯤되면 펜톤에서 올해의색상을 발표하는것처럼 ㅋㅋㅋ어딘가에서 올해의 재료나 음식을 ... |
| ... | ... |
| 1972 | 물티슈로 입을 좀 수시로 닦아주세요^^ |
| 1973 | 3:06 5:36 ㅋㅋㅋ |
| 1974 | 한입주셔유 |
| 1975 | 😍😍 |
| 1976 | 증말 |

Out [25] :

| | Comments |
|---|---|
| 0 | 왜들 미역가지고 난리들이랴 미역 어디서 싸게 나왔나 |
| 1 | 진짜쫄면 소스인듯 미역을 왜케 |
| 2 | s 왕의 밥상 도시락이 궁금합니다 고물 대왕님 리뷰 부탁 좀 가격은 ... |
| 3 | 컨텐츠도 리뷰도 제일 상세하고 여타 다른 유투버보다 세밀해서좋은데 파이리 만빼면... |
| 4 | 이쯤되면 펜톤에서 올해의색상을 발표하는것처럼 ㅋㅋㅋ어딘가에서 올해의 재료나 음식을 ... |
| 5 | 잘 먹었습니다 미역이라니 살짝 미친듯 |
| 6 | 착 뜯고서 뭔가 눈빛이ㅋㅋ 멋진척 인건가요 고물님 귀여워 미역 그 그만 |
| 7 | 마지막에 매운맛 확 올라와서 허어 하는게 너무 귀여워 게이 될거같아 |
| 8 | 무난한 비빔면 |
| 9 | 삼양은 나온다면 클 또닭비빔면 또역에디션 이런 거일려나요 ㅋㅋㅋ |

In [42] :

```
# 토큰화 + 토큰리스트 생성
def make_tokens(df):
    df['tokens'] = ''
    tokens_list=[]
    for i, row in df.iterrows():
        if i%500==0:
            print(i,'/',len(df))
        token = token_okt(df['Comments'][i])
        df['tokens'][i] = ' '.join(token)
    return df

df = make_tokens(df)
```

한글 제외한 다른 문자제거

# 2. 분석 프로세스

## [ 감정분석을 위해 LSTM으로 자동 라벨링하는 머신러닝 모델 구현 ]

```
In [ ]: from tensorflow.keras.layers import Embedding, Dense, LSTM
        from tensorflow.keras.models import Sequential
        from tensorflow.keras.models import load_model
        from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint
```

```
In [ ]: model = Sequential()
        model.add(Embedding(vocab_size, 100))
        model.add(LSTM(128))
```

```
In [ ]: loaded_model = load_model('best_model.h5')
        print("\n 테스트 정확도: %.4f" % (loaded_model.evaluate(X_test, y_test)[1]))
```

```
12/12 [==============================] - 0s 7ms/step - loss: 0.5467 - acc: 0.7880

 테스트 정확도: 0.7880
```

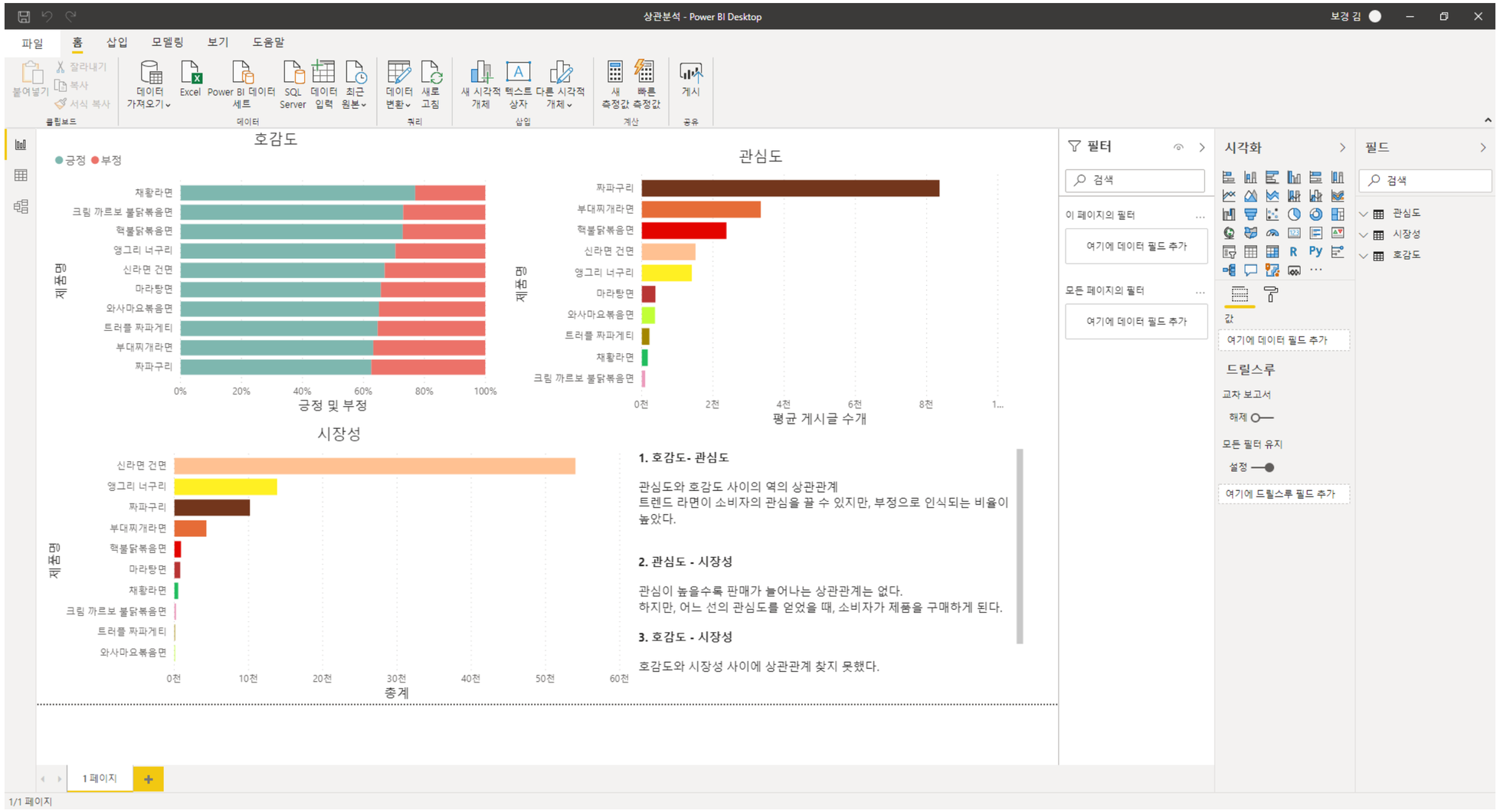
테스트 정확도: 0.7880

```
15/15 [==============================] - 1s 63ms/step - loss: 0.6278 - acc: 0.7216 - val_loss: 0.6674 - val_acc: 0.5694
Epoch 2/15
14/15 [============================>.] - ETA: 0s - loss: 0.5278 - acc: 0.7357
Epoch 00002: val_acc improved from 0.56944 to 0.60648, saving model to best_model.h5
15/15 [==============================] - 0s 32ms/step - loss: 0.5314 - acc: 0.7320 - val_loss: 0.6353 - val_acc: 0.6065
Epoch 3/15
14/15 [============================>.] - ETA: 0s - loss: 0.4261 - acc: 0.7929
Epoch 00003: val_acc improved from 0.60648 to 0.71296, saving model to best_model.h5
15/15 [==============================] - 0s 31ms/step - loss: 0.4264 - acc: 0.7912 - val_loss: 0.5860 - val_acc: 0.7130
Epoch 4/15
13/15 [=========================>....] - ETA: 0s - loss: 0.3249 - acc: 0.8577
Epoch 00004: val_acc improved from 0.71296 to 0.75463, saving model to best_model.h5
15/15 [==============================] - 1s 34ms/step - loss: 0.3207 - acc: 0.8619 - val_loss: 0.5595 - val_acc: 0.7546
Epoch 5/15
14/15 [============================>.] - ETA: 0s - loss: 0.2543 - acc: 0.9083
Epoch 00005: val_acc did not improve from 0.75463
15/15 [==============================] - 0s 30ms/step - loss: 0.2565 - acc: 0.9095 - val_loss: 0.5354 - val_acc: 0.7407
Epoch 6/15
13/15 [=========================>....] - ETA: 0s - loss: 0.2112 - acc: 0.9333
Epoch 00006: val_acc did not improve from 0.75463
15/15 [==============================] - 1s 39ms/step - loss: 0.2093 - acc: 0.9327 - val_loss: 0.5684 - val_acc: 0.7546
Epoch 7/15
15/15 [==============================] - ETA: 0s - loss: 0.1882 - acc: 0.9339
Epoch 00007: val_acc did not improve from 0.75463
15/15 [==============================] - 0s 32ms/step - loss: 0.1882 - acc: 0.9339 - val_loss: 0.6147 - val_acc: 0.7454
Epoch 8/15
13/15 [=========================>....] - ETA: 0s - loss: 0.1300 - acc: 0.9564
Epoch 00008: val_acc improved from 0.75463 to 0.75926, saving model to best_model.h5
15/15 [==============================] - 0s 33ms/step - loss: 0.1320 - acc: 0.9548 - val_loss: 0.6015 - val_acc: 0.7593
Epoch 9/15
15/15 [==============================] - ETA: 0s - loss: 0.1218 - acc: 0.9582
Epoch 00009: val_acc did not improve from 0.75926
15/15 [==============================] - 1s 33ms/step - loss: 0.1218 - acc: 0.9582 - val_loss: 0.6935 - val_acc: 0.7315
Epoch 00009: early stopping
```

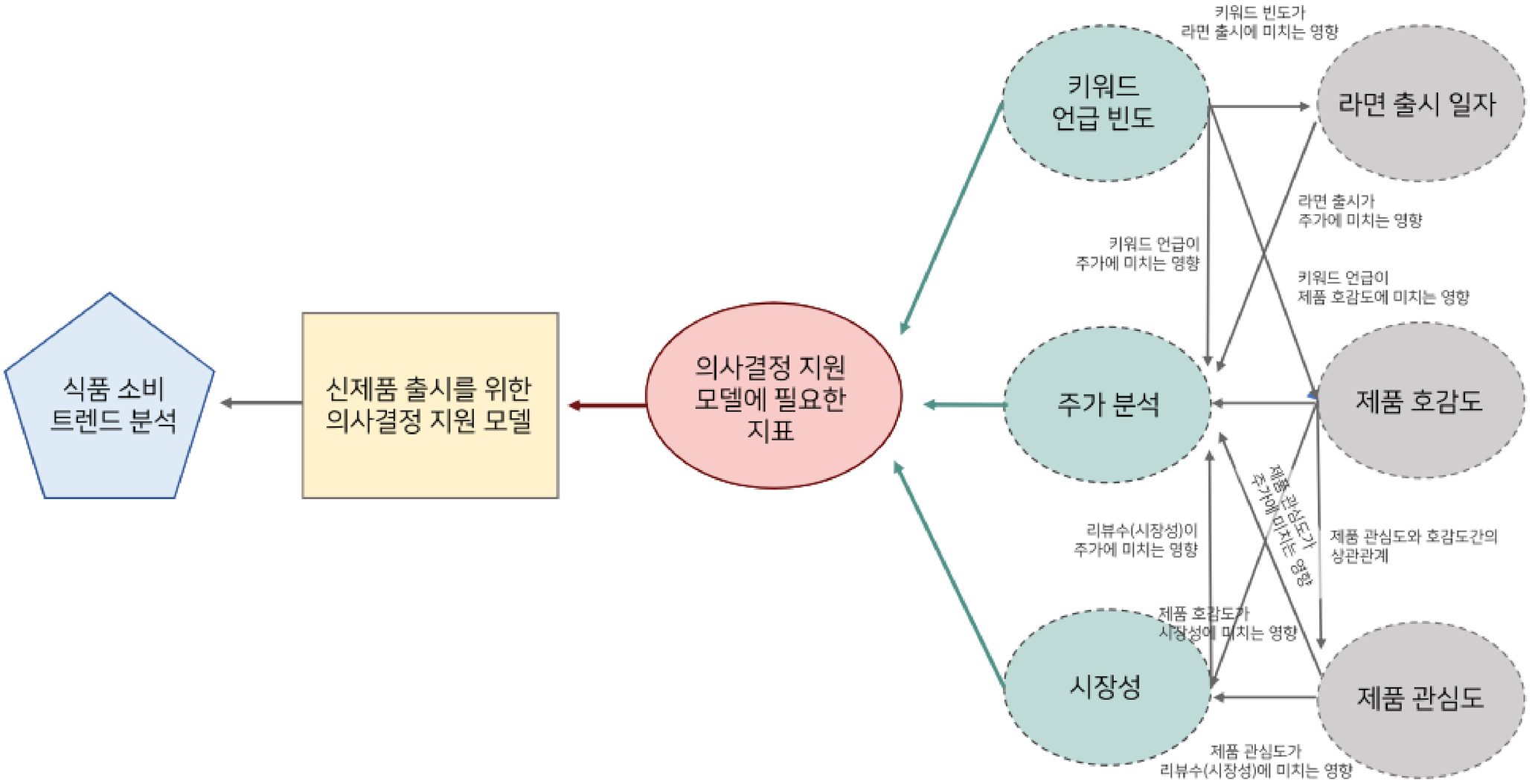
Epochs = 15
데이터를 총 15번
학습 시킴

# 2. 분석 프로세스

[ BI 분석 · 시각화 ]



# 3. 분석 모델링


식품 소비
트렌드 분석
신제품 출시를 위한
의사결정 지원 모델
의사결정 지원
모델에 필요한
지표
키워드
언급 빈도
라면 출시 일자
주가 분석
제품 호감도
시장성
제품 관심도
키워드 빈도가
라면 출시에 미치는 영향
키워드 언급이
주가에 미치는 영향
라면 출시가
주가에 미치는 영향
키워드 언급이
제품 호감도에 미치는 영향
리뷰수(시장성)이
주가에 미치는 영향
제품 관심도가
주가에 미치는 영향
제품 관심도와 호감도간의
상관관계
제품 호감도가
시장성에 미치는 영향
제품 관심도가
리뷰수(시장성)에 미치는 영향

PART.3

# 분석

03

1. 분석대상 선정 근거
2. 분석 대상 선정
3. 요소 선정 및 검증
4. 요소 간의 상관관계

# 라면 키워드를 추출하여 출시 당시 트렌드였는지 확인


크림

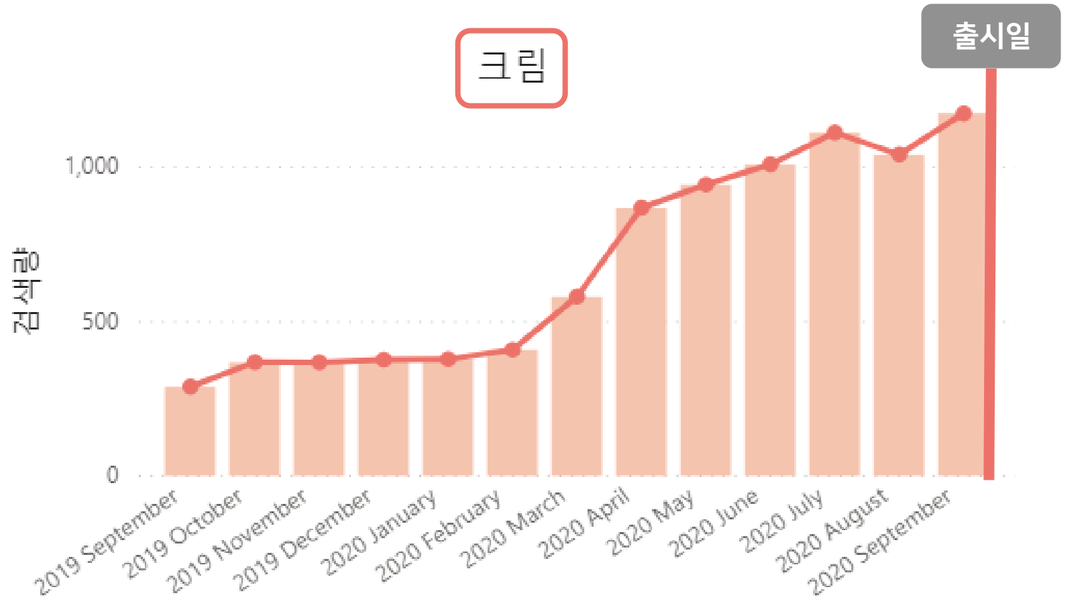

마라

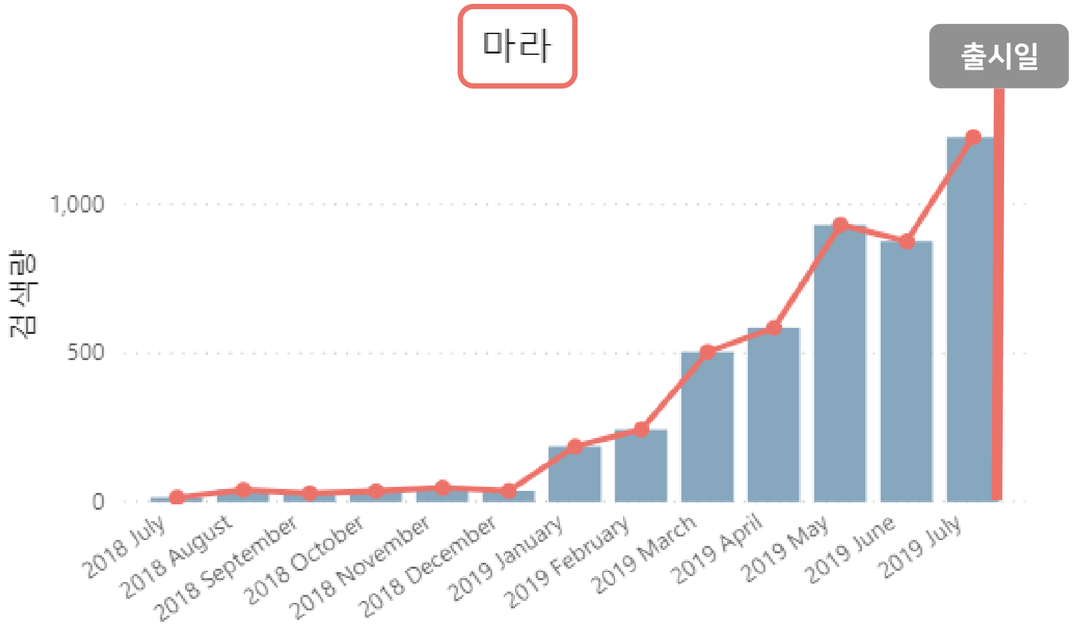
# 1. 선정근거

## 키워드의 검색량 상승

- 키워드 검색량이 라면이 출시되기 전 1년간 20% 이상 상승하면 이 키워드가 해당 시기 트렌드였다고 판단

- 트렌드를 반영한 라면 10개 선정



# 2. 분석대상 선정

**건강**

농심 신라면 건면

**마라**

삼양 마라탕면

**매운**

- 농심 앵그리 RtA
- 삼양 핵불닭볶음면

**비건**

오뚜기 채황라면

**와사비마요네즈**

삼양 와사마요볶음면

**짜파구리**

농심 짜파구리

**부대찌개**

농심 부대찌개면

**크림**

삼양
크림까르보불닭볶음면

**트러플**

농심 트러플 짜파게티

# 3. 요소선정 및 검증

## 1) 검증요소 선정 : 분석 요인을 네 가지로 선정

요인 1
**관심도**

네이버 블로그 게시글 수
추이를 크롤링 하여
기획라면에 대한
소비자들의 관심도 분석

요인 2
**호감도**

NLP 감정분석을 통한
소비자들의 기획라면에 대한
호감도 분석

요인 3
**시장성**

온라인 쇼핑몰 상품평 수를
통한 기획라면의 시장성 분석

요인 4
**주식**

기획라면의 출시했을 때,
주식의 연관성 분석

# 3. 요소선정 및 검증

**분석요인 ① 관심도**

**2) 검증 : 기획 라면에 대한 소비자들의 관심을 파악 하고자 함.**
**이를 위해  네이버 블로그 게시글 수 언급량을 통해 소비자들의 관심도를 검증**

예측

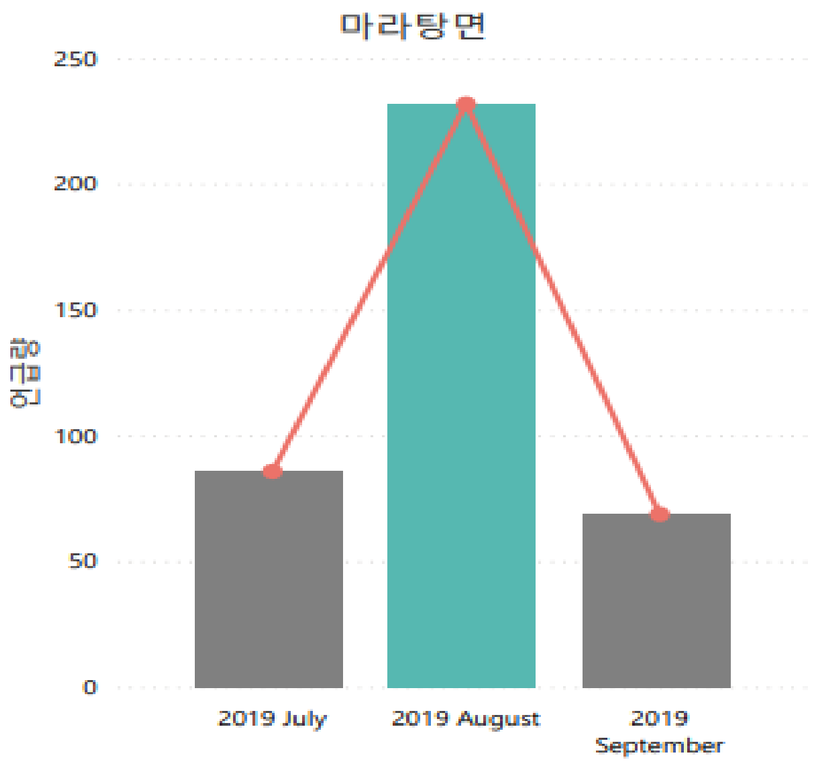

‘네이버 블로그 게시글’
에서  라면 이 출시된 후,
언급량이  많을 수록
‘관심‘ 이 높다 라는 예측

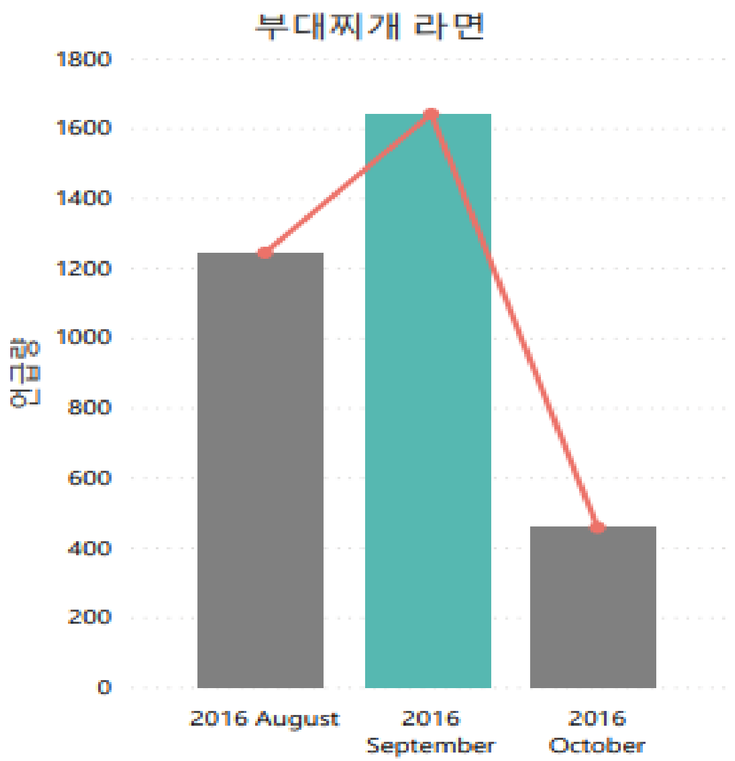
검증

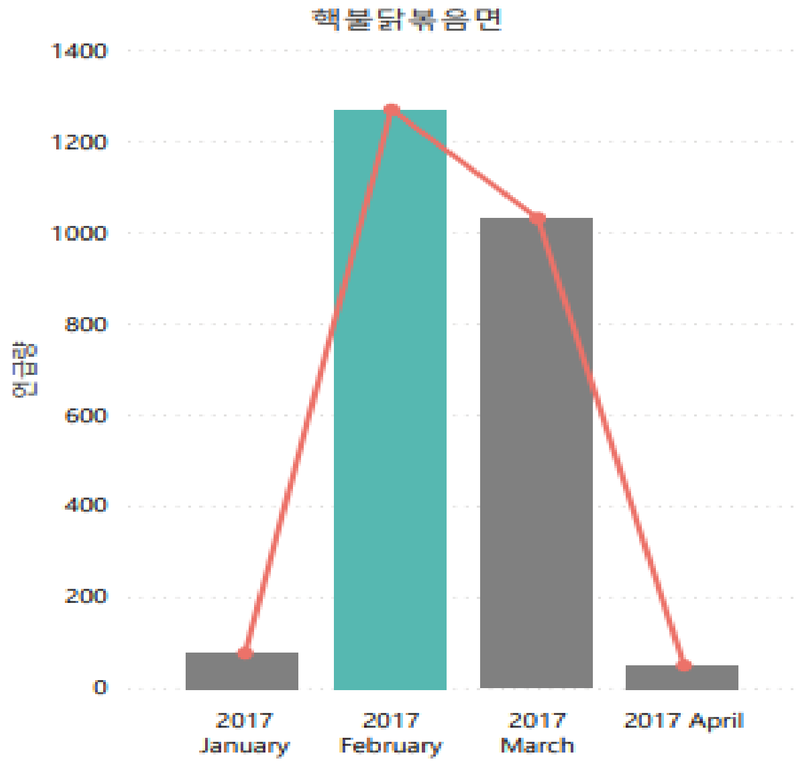

네이버 블로그를 통해서
해당 라면 이 들어간  게시글들을
크롤링 하여  90일 간
‘언급된 라면의 블로그 게시글 수를 ’ 의
총 합을 구하였음.

# 3. 요소선정 및 검증

## 분석요인 ① 관심도
2) 검증



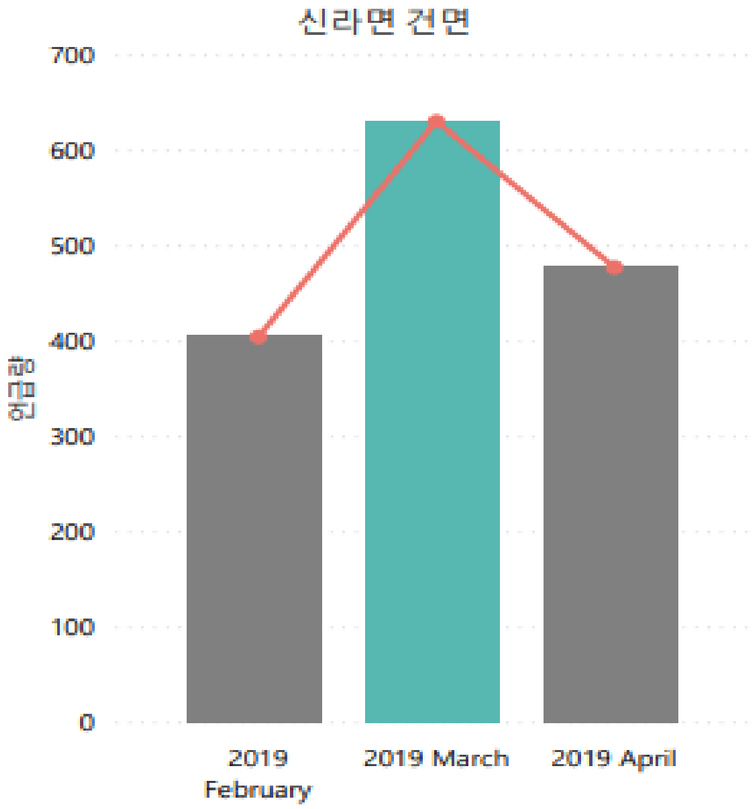



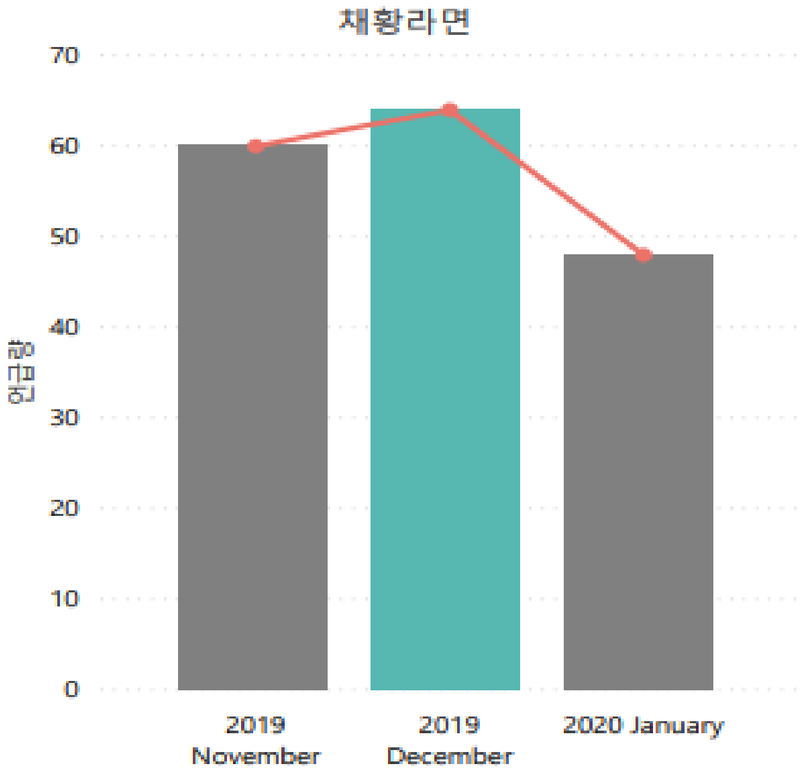



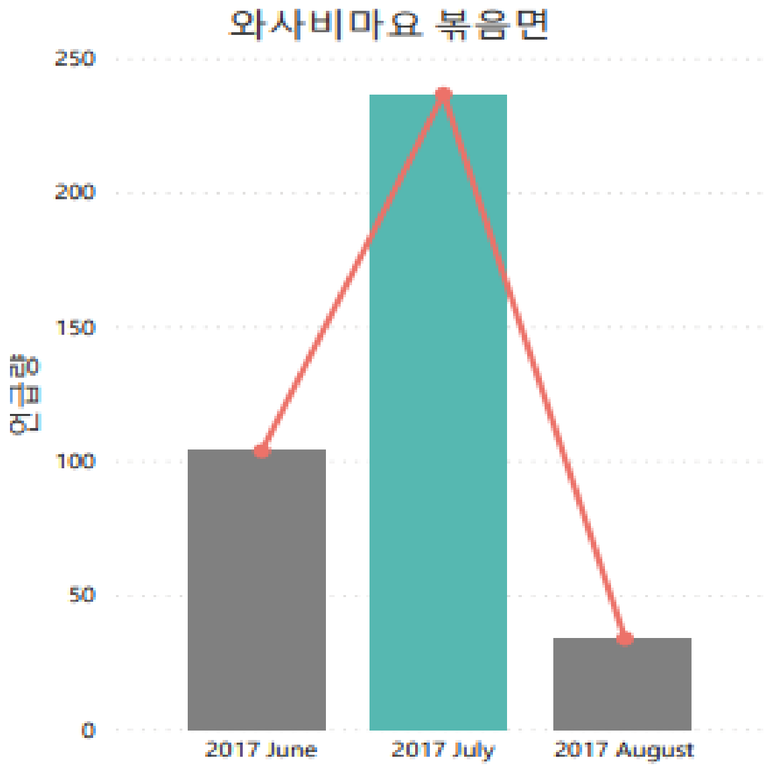

출시후. 한달 동안 단기적 급 상승세를 보이고 그 후 다시 하락세

# 3. 요소선정 및 검증

## 분석요인 ① 관심도
2) 검증



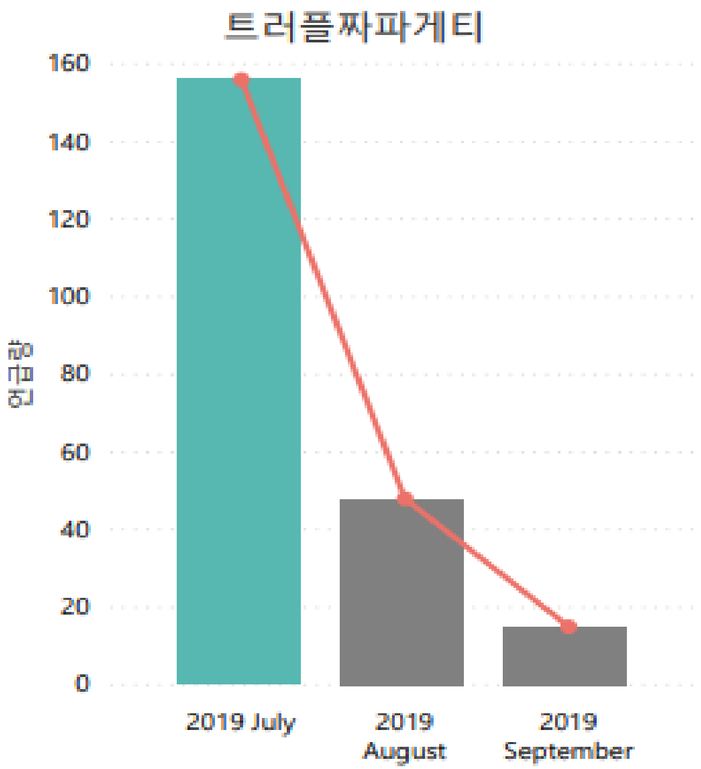



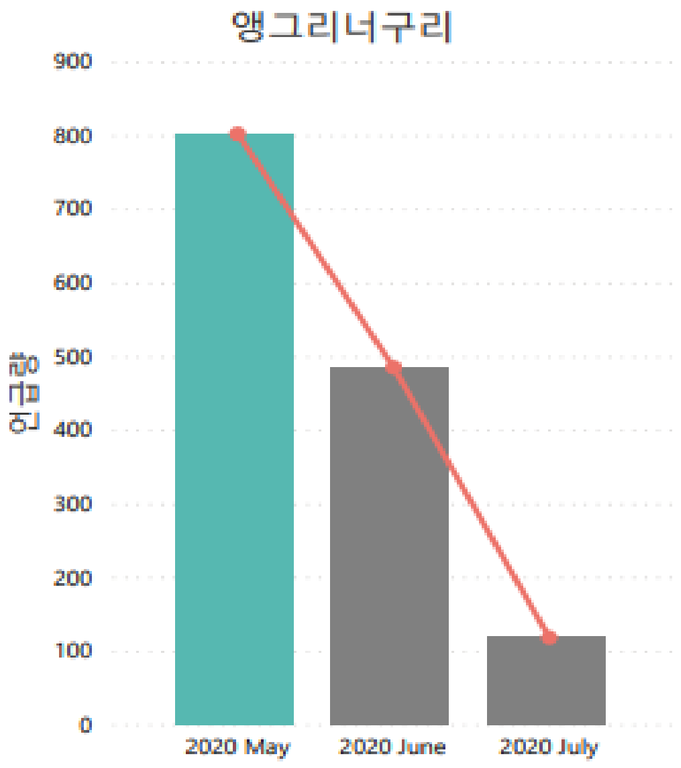



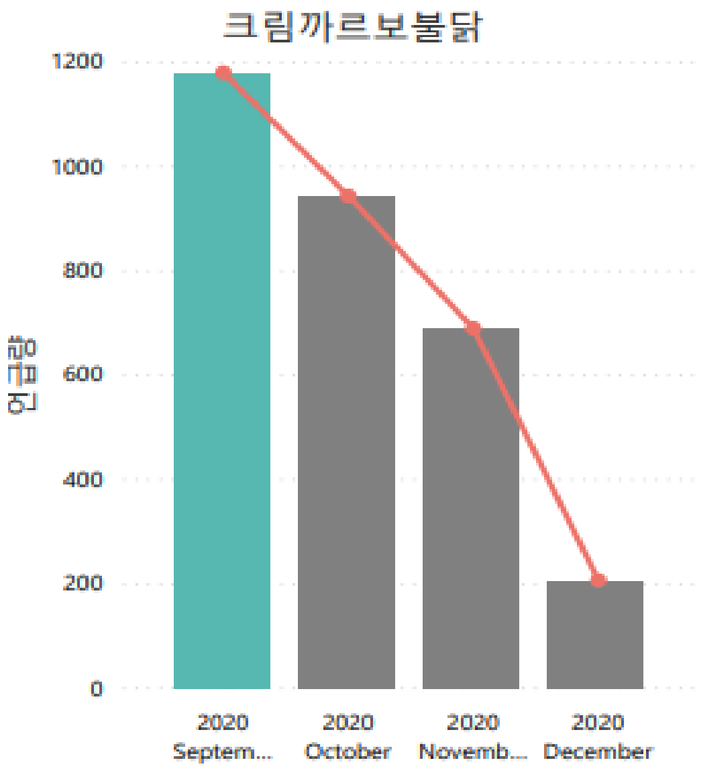

출시후. 한달 동안 단기적 급 상승세를 보이고 그 후 다시 하락세

# 3. 요소선정 및 검증

## 분석요인 ① 관심도
2) 검증



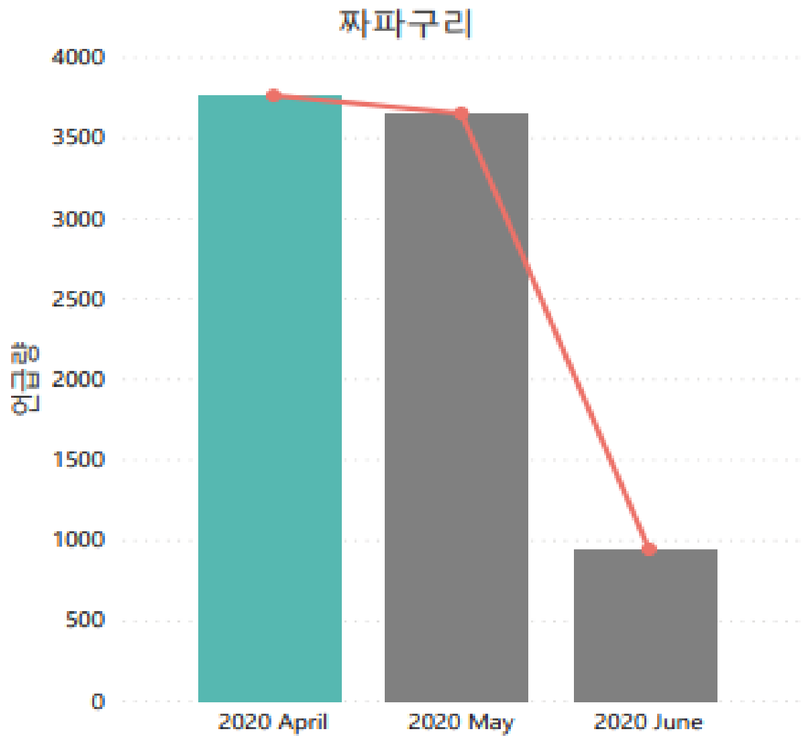




출시후. 지속 적인 하락

# 3. 요소선정 및 검증

## 분석요인 ① 관심도
2) 검증




‘기생충’ 영화의 영향으로
짜파구리 라면은
두 달간 급 상승세 -> 급 하락.

# 3. 요소선정 및 검증

## 분석요인 ① 관심도
### 3) 검증 결과

6개의 데이터
(마라탕면, 부대찌개, 핵불닭, 신라면 건면, 채황라면, 와사비 마요 등)

→

출시 후, 한 달 동안 게시글 수의 상승이 있었고,
그 후는 하락세

→

소비자들은 기획라면의 출시 후, 한 달간 관심도가 가장 높다는 것을 알 수 있었음

3개 데이터
(트러플, 앵그리,크림,짜파구리)

→

출시 후, 지속적인 하락세

→

트러플 짜파게티, 짜파구리는 미디어(영화, 예능) 등의 영향으로 출시 직후에만 단기적 관심도를 보임

# 3. 요소선정 및 검증

## 분석요인 ② 호감도

2)검증 : 기획 라면에 대하여 소비자들이 얼마나 긍정적인지 부정적인지 검증 하고자 함
이를위해 NLP 감성분석을 통하여 , 소비자들의 유투브 리뷰 댓글을 크롤링 하여 소비자들의 기획라면의 호감도를 검증

### 예측

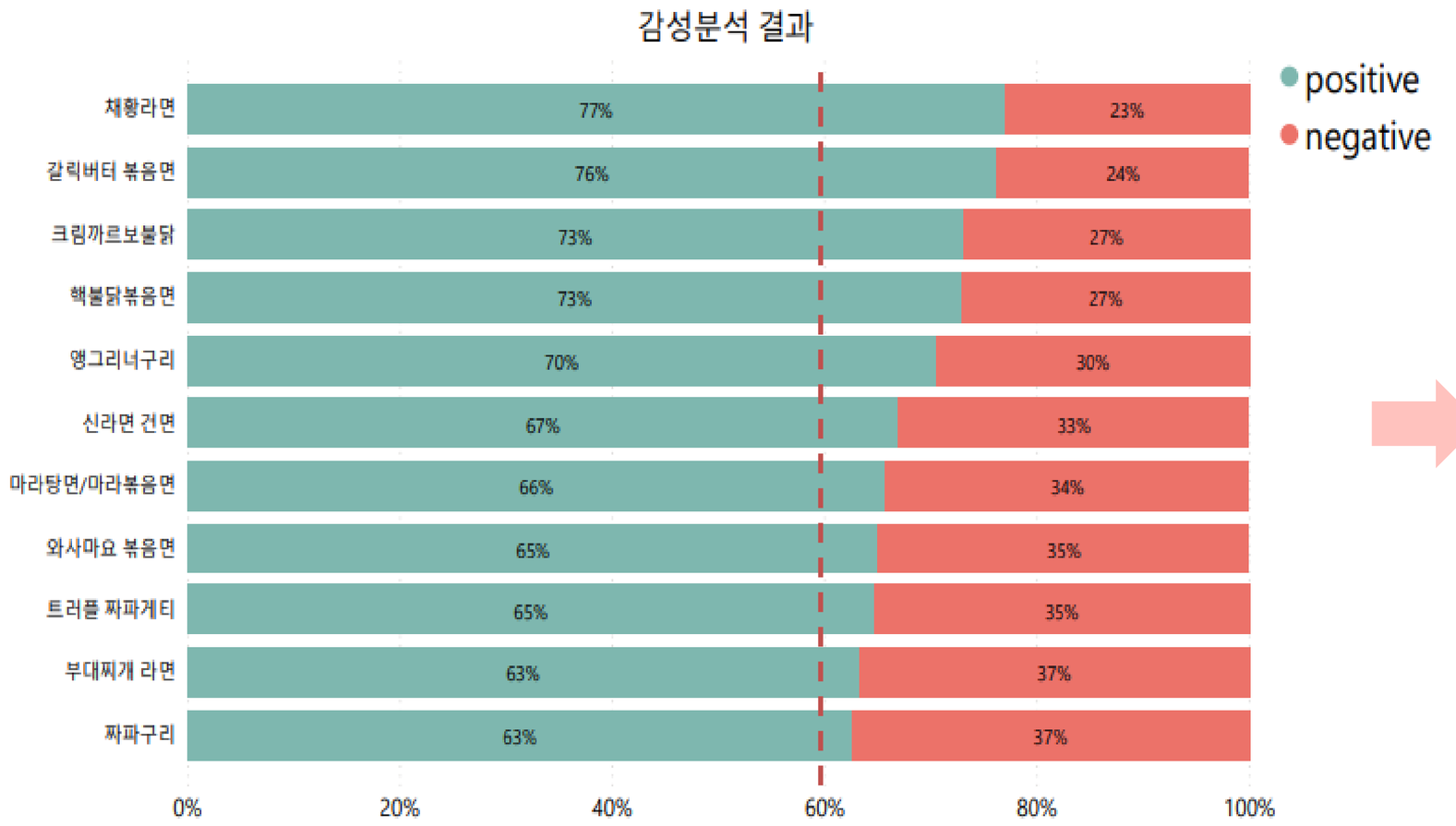

라면 10개 데이터가
실제로 소비자들의 '호감' 이 있었는지 알고자 함.
유튜브 긍정 댓글이 60퍼센트가 넘어가면
'호감도'가 높은 라면으로 선정

### 검증


소비자들의 유튜브 댓글을 크롤링 후,
NLP 감성분석을 통하여
기획라면에 대한 소비자들의
긍정 부정을 알아보고
호감도를 검증

# 3. 요소선정 및 검증

## 분석요인 ② 호감도
### 3)검증 결과




선정한 10개 라면
모두 60% 이상 호감도를 보임

# 3. 요소선정 및 검증

## 긍정 키워드와 부정 키워드

25000개 유튜브 댓글 기반 워드 클라우드

```
pos_dataset = pos_dataset.lower()
create_word_cloud(pos_dataset)
```


긍정

```
In [197]: neg_dataset = neg_dataset.lower()
          create_word_cloud(neg_dataset)
```


부정

# 3. 요소선정 및 검증

## 분석요인 ③ 시장성

2) 검증 : 우리가 뽑은 라면 10개 데이터가 실제로 시장성이 있는지 파악하고자함 .
이를 위해, 온라인 쇼핑몰의 상품평 추이를 보고 소비자들의 '매출'로 이어졌는지 분석

예측

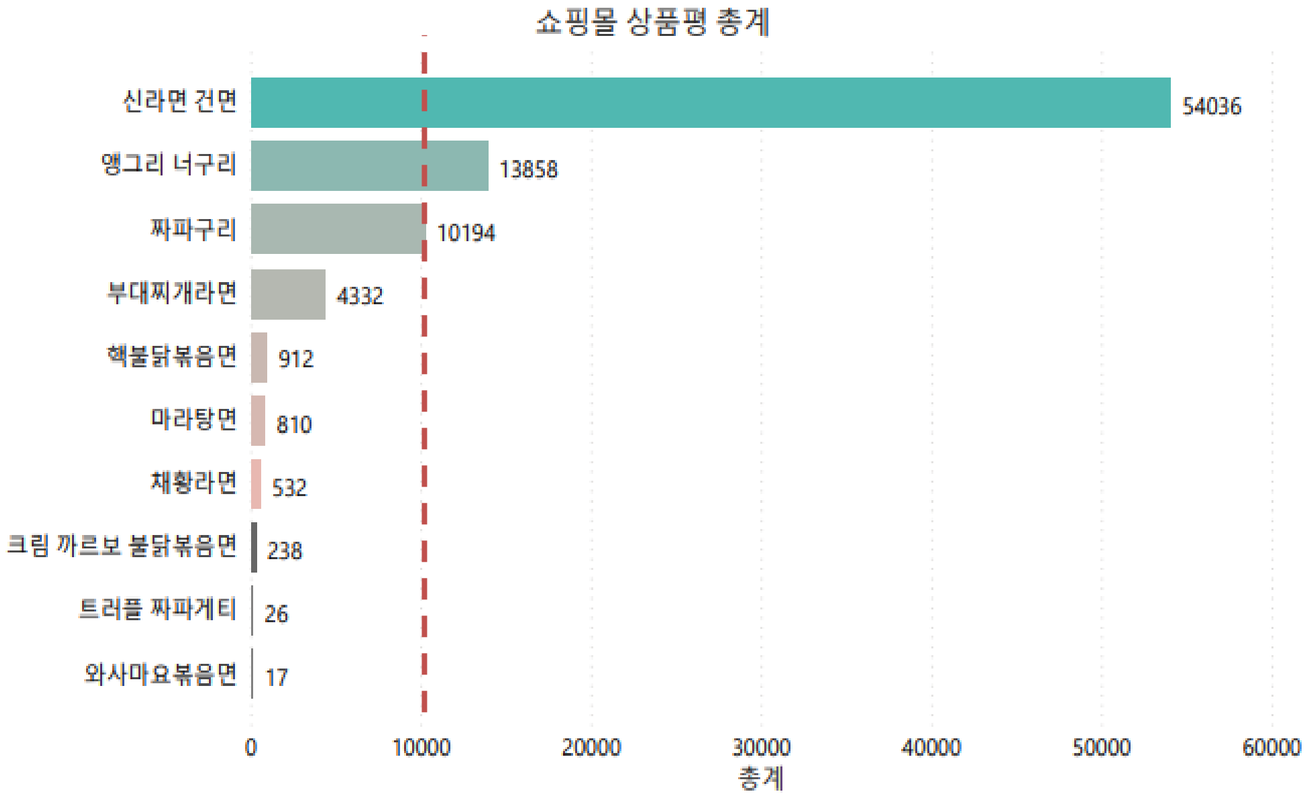

소비자들의
온라인 쇼핑몰 상품평 추이를 파악하여
상품평이 많을 수록 시장성이 높다고 예측

검증


온라인 쇼핑몰(쿠팡, 이마트몰 등)에서
상품후기 댓글을  크롤링하여
각 라면의 상품평의 총 합을 더하여
시장성을 검증

# 3. 요소선정 및 검증

## 분석요인 ③ 시장성
### 3) 검증 및 결과




결과

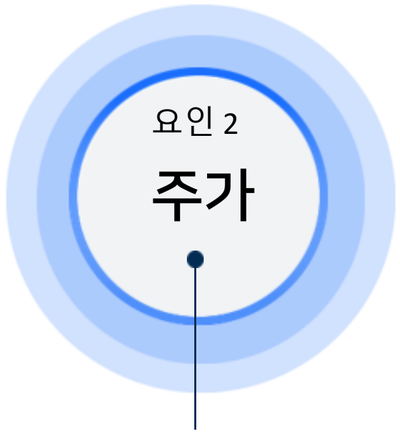
신라면 건면이 압도적으로 높은 시장성.
상품평이 10000건 이상인
앵그리너구리와 짜파구리가 시장성이 높음.

나머지 라면들은 시장성이 낮음

# 3. 요소선정 및 검증

## 분석요인 ④주가
## 2)검증

예측



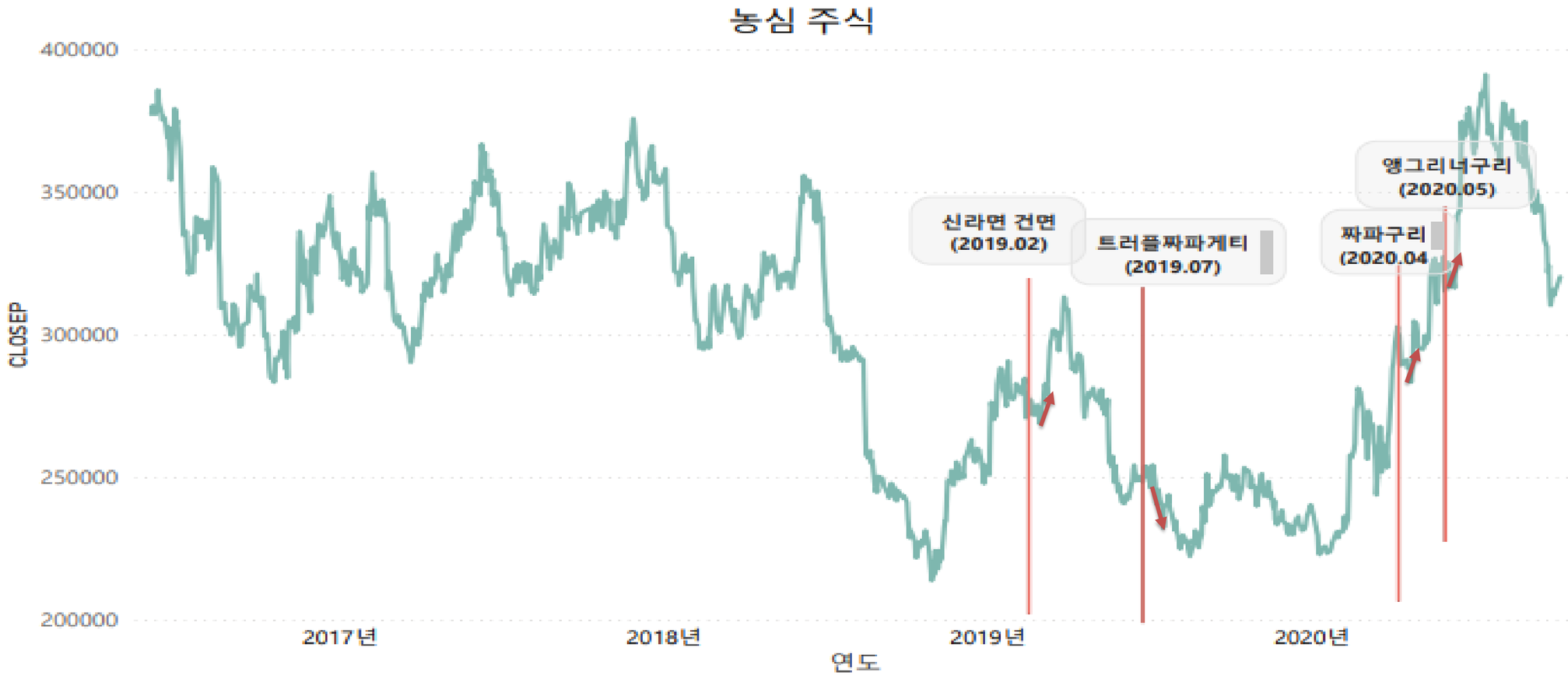

기획라면이 '출시' 할때
해당 기업의 주가에도 영향이 있을 수도 있다는
예측을 했음.

검증

기획라면이 '출시일'을 기점으로
해당 기업의 주가를 크롤링 하여
출시 후, 한달동안 주가가 상승폭이 컸는지
하락폭이 컸는지 검증

# 3. 요소선정 및 검증

## 분석요인 ④주가
### 3) 검증 결과



# 3. 요소선정 및 검증

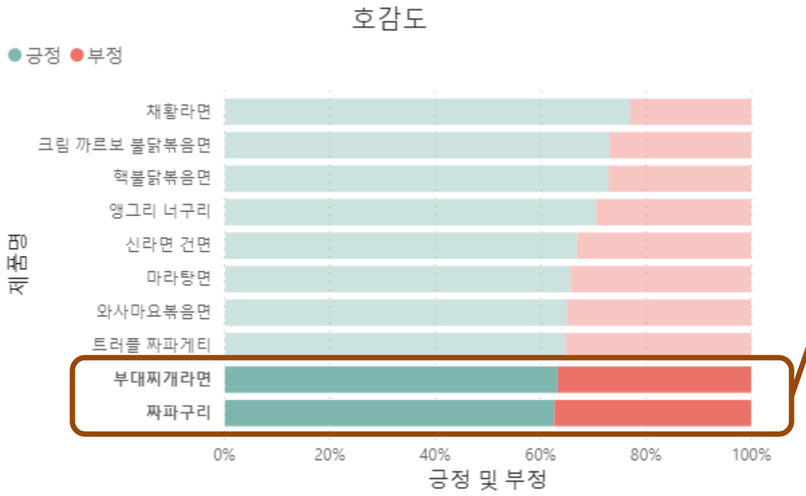
## 분석요인 ④주가
### 3) 검증 결과

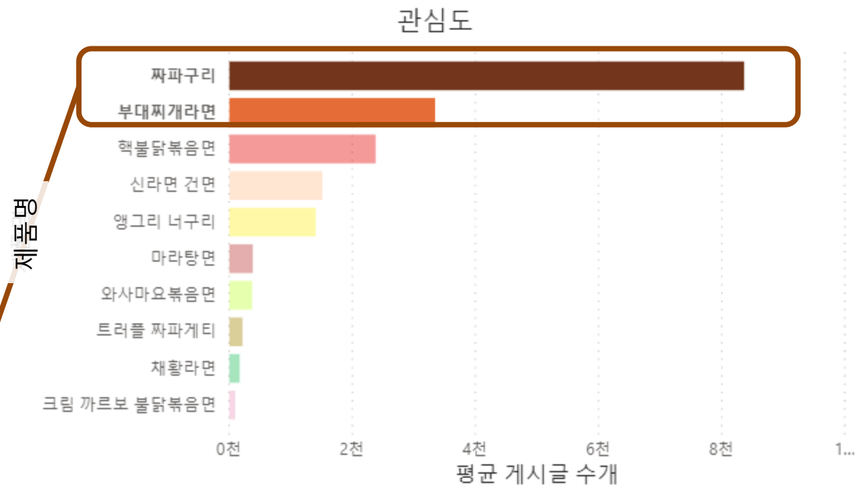
# 3. 요소선정 및 검증

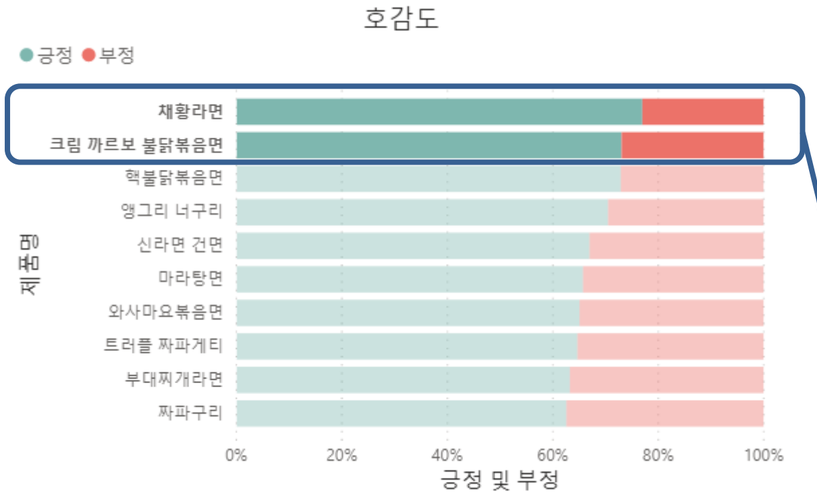
## 분석요인 ④주가
### 3) 검증 결과

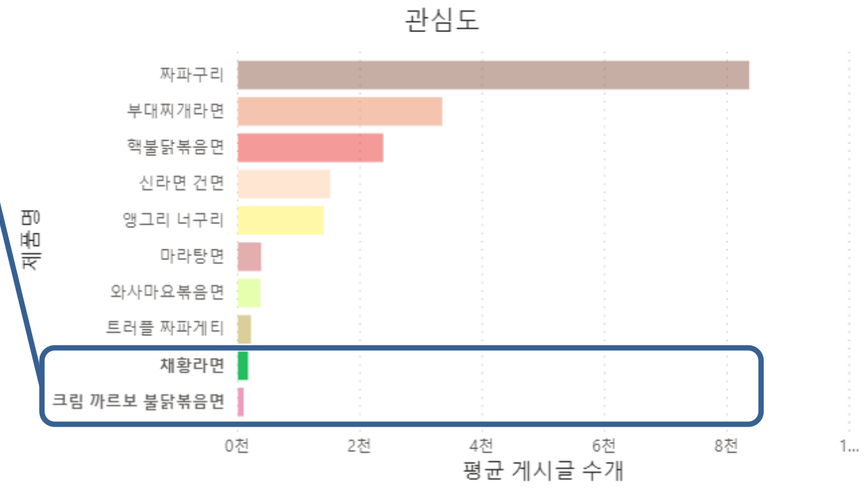
# 4. 요소간의 상관관계

## 1. 호감도 - 관심도



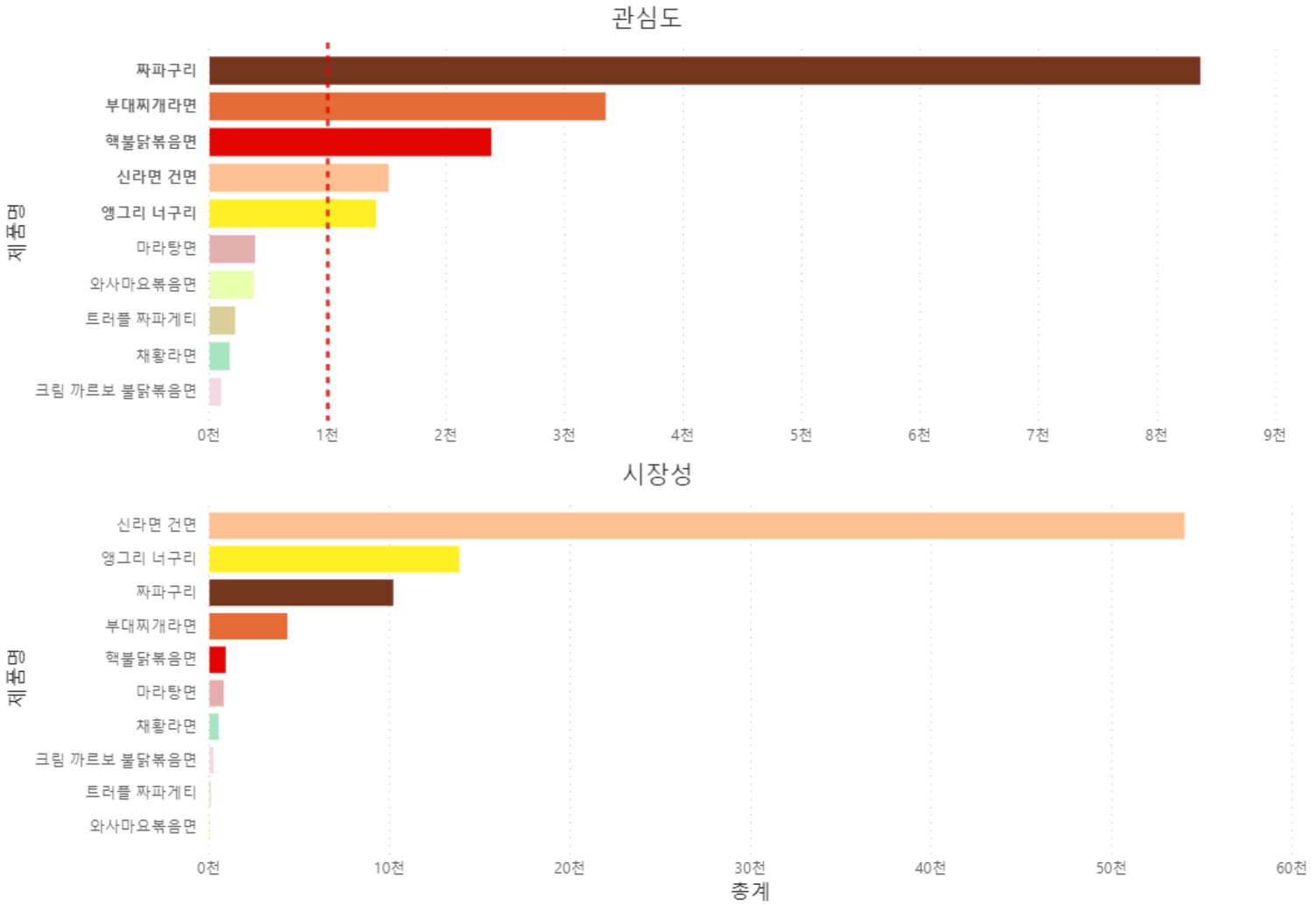




- 관심도와 호감도 사이의 역의 상관관계
- 호감도 하위 2개 제품이 관심도 상위 2개 제품
- 따라서, 트렌드 라면이 소비자의 관심을 끌 수 있지만, 부정으로 인식되는 비율이 높았다.

# 4. 요소간의 상관관계

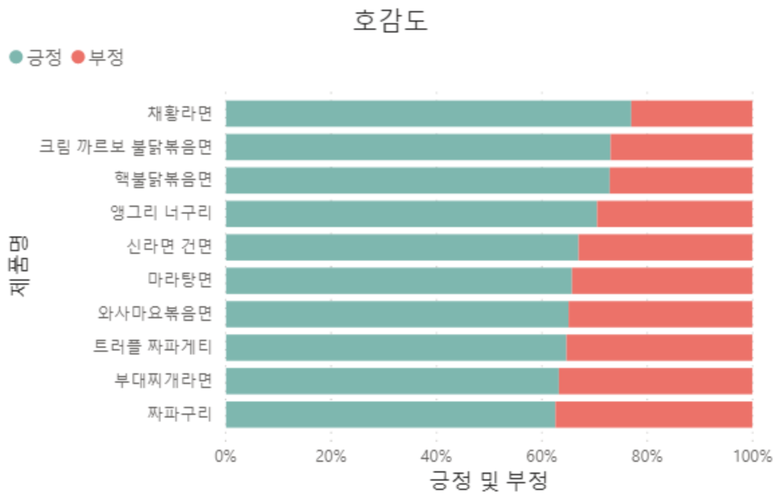
## 2. 관심도 - 시장성

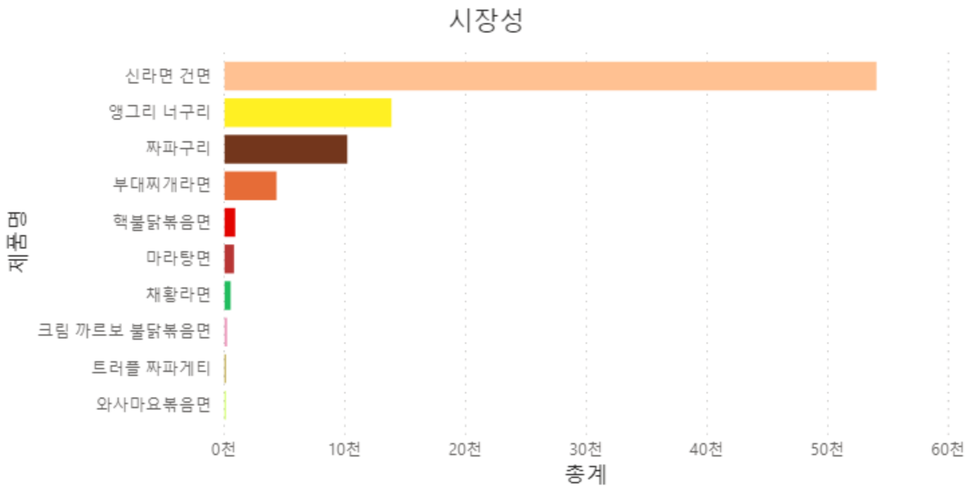



- 관심도가 높을수록 판매가 늘어나는 양의 상관관계는 없다.
- 하지만, 일정 선의 관심도를 얻었을 때, 제품 구매로 이어지게 된다.

# 4. 요소간의 상관관계

## 3. 호감도 - 시장성




- 호감도와 시장성 사이에는 특별한 상관관계를 찾지 못하였다.

PART.4

# 결론

04

1. 결론
2. 한계점 및 어려움

# 4-1. 결론

## No pattern but trend

라면은 관심도, 호감도, 시장성, 주식 등의 측면에서 일정한 패턴을 발견할 수는 없었음.

- 라면은  대중식품으로서 대중의 반응과 트랜드의 흐름이 중요함
- 우리가 크롤링한 데이터들(SNS,유튜브,온라인 쇼핑몰)는  대중 트랜드와 입맛을 파악하는 데에는 적절한 방법
- 키워드로 변경시, 다른 산업군에도 적용할 수 있는 분석 플랫폼으로 나아가 사용자들 에게 서비스를 제공할 수 있음

# 4-2. 한계점 및 어려움

## [ 내부 데이터의 부재 ]

판매량 및 매출액
데이터가 없어 리뷰수로
상품의 시장성 판단

## [ 상품 출시 – 주가간의 상관관계 ]

상품 출시 외에도 다양한
요인들이 주가에 영향 미침
(예: 코로나)

## [ 워드클라우드 ]

감정분석 결과에서
유의미한 결과를 찾으려
했지만 어려움 느낌

# 감사합니다

CONTACT
UNICORN@MMU.COM
010 1234 5678