# 감성분석1

## 사전기반

최 석 재

*lingua@naver.com*

# 감성분석


- 긍정 – 중립 – 부정 의 세 가지로 나누는 긍부정 분석과
- 기쁨, 슬픔, 분노, 놀람 등 사람의 감정으로 분류하는 감정 분석이 있다
- 일반적으로는 둘을 구분하지 않고 감정분석 혹은 감성분석이라고 한다

- Customer Service 분야에서는
- 전화, 게시판, 이메일 등에서 수집된 고객의 소리 중 (VOC)
- 특히 고객의 불만을 파악하여 서비스 개선하는 데 사용한다

- 빈도분석과 연결하여 사용되며,
- 전통적인 사전을 이용한 방법과
- 머신러닝 또는 딥러닝을 이용한 방법이 사용된다

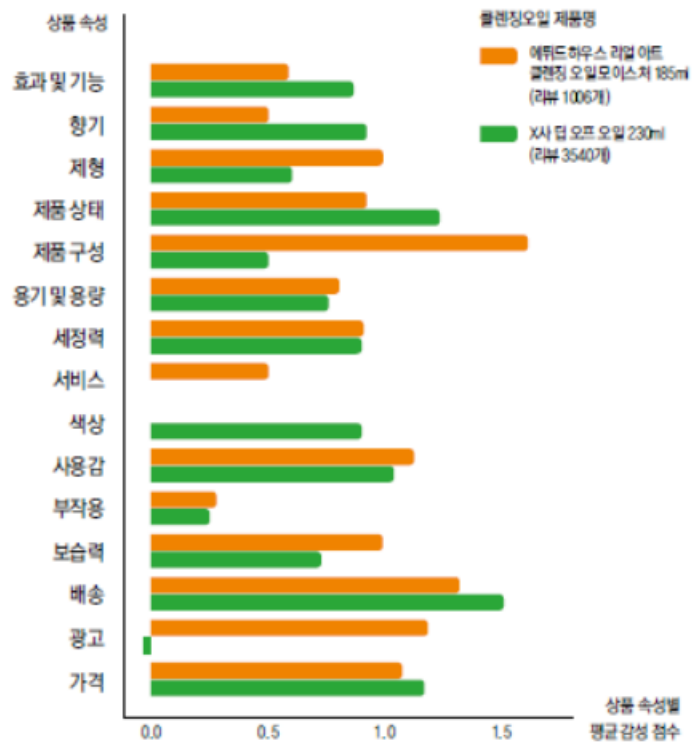
# 분석 감정의 종류

- 2분류: 긍정 – 부정
- 3분류: 긍정 – 중립 – 부정
- 4분류: neutral, happy, sad, anger
- 5분류: neutral, happy, SURPRISE(surprise+fear), sad, ANGER(disgust+anger)
- 6분류: neutral, happy, surprise, fear, sad, ANGER(disgust+anger)
- 7분류: neutral, happy, surprise, fear, sad, disgust, anger

# 에뛰드하우스 화장품 감성분석

- 에뛰드하우스와 경쟁 상품이 언급된 SNS 글을 수집
- 전반적인 감성분석은 물론, 속성별 감성분석으로 상품의 상대적 위치 파악

| Item_id | Name | Category_name | Review N | Senti mean |
|---|---|---|---|---|
| 4114273022 | A사퍼펙트 클렌징폼 120g | 클렌징폼 | 3,101 | 2.12 |
| 9164497192 | B사 블랙헤드 리무버 오일 젤 60ml | 클렌징젤 | 5,525 | 0.93 |
| 7828017344 | C사 클린 잇 180ml | 클렌징크림 | 4,021 | 1.7 |
| 2600059922 | D사딥 오프 오일 230ml | 클렌징오일 | 3,540 | 1.61 |
| 5709044152 | E사H2O 클렌징워터500ml | 클렌징워터 | 3,248 | 1.92 |
| 8253203439 | F사 클렌징 워터 500ml | 클렌징워터 | 3,040 | 1.67 |
| 4114214175 | G사 클렌징 오일 200ml | 클렌징오일 | 2,537 | 1.77 |
| 2600022289 | H사 클렌징 오일 200ml | 클렌징오일 | 2,407 | 1.95 |
| 8253075054 | I사 내추럴 클렌징 오일 150ml | 클렌징오일 | 2,225 | 1.68 |
| 10453358372 | H사 크레이지 폼클렌저 500ml | 클렌징폼 | 1,659 | 2.04 |
| 7832473489 | J사 그린티 클렌징 워터 300ml | 클렌징워터 | 1,643 | 1.75 |
| 10239039251 | K사 멀티 클렌징폼 175ml | 클렌징폼 | 1,487 | 2.16 |
| 6206154532 | L사 클렌징폼 175g | 클렌징폼 | 1,473 | 2.29 |
| 5720227580 | M사H2O 클렌징 오일 500ml | 클렌징워터 | 1,295 | 1.8 |
| 8699954195 | N사 티트리 클렌징 워터 300ml | 클렌징워터 | 1,287 | 1.77 |
| 7868312924 | O사 워터리 오일N 230ml | 클렌징오일 | 1,237 | 1.63 |
| 11932147834 | P사 모이스처 업 패드 70매 | 클렌징티슈 | 1,176 | 1.74 |
| 6168162950 | 에뛰드하우스 립앤 아이 리무버 250ml | 립앤아이리무버 | 1,079 | 1.85 |
| 8495081524 | 에뛰드하우스 리얼 아트 클렌징 오일 185ml | 클렌징오일 | 1,006 | 1.79 |



# VOC 감성분석

# 감성단어 긍부정 분류

1. Encoder는 총 3인
2. VOC 사례 문서를 추출
3. 긍정, 중립, 부정 단어에 대한 개념을 공유
4. Encoder 3인이 각자 pilot 문서 내 소속 단어에 대해서 부정, 중립, 긍정으로 분류
5. 함께 일치, 불일치 여부를 논의하고 의견을 교환
6. 본격적으로 전체 단어에 대해 Encoder 3인이 각각 부정, 중립, 긍정으로 분류
7. 함께 모여 서로 일치하는 부분은 그대로 판정하고 감성단어 사전에 등록, 일치하지 않는 부분은 서로 상의 후 다시 부정, 중립, 긍정으로 분류
8. 재차 모여 일치하면 감성단어 사전에 등록하고, 불일치하는 경우 논의에 의하여 감성단어 사전에서 제외

# VOC용 감성단어 선정 결과

- 긍정단어 666개
- 부정단어 1,780개
- 중립단어 1,566개 추출

**긍정 단어 예**

축하, 축하되, 축하하, 충분, 충분되, 충분하, 충분히, 충성되, 충실, 충실도, 충실히, 충의되, 치국되, 치국하, 치료법, 치료제, 치안, 친교, 친목, 친밀, 친밀되, 친밀하, 친선되, 친절되, 침착, 침착되, 침착하, 칭찬, 칭찬되, 칭찬하, 카리스마, 쾌적, 쾌적되, 쾌적하, 쾌활, 크리스마스, 크리스마스트리, 클라이맥스, 타협하, 탁월하, 탐정, 태평, 태평되, 태평하, 통일되, 통일하, 통찰력, 트러스트, 특권, 튼튼되, 튼튼하, 편리, 평안하, 평온되, 평탄, 평탄하, 평화되, 평화하, 포옹, 풍부되, 풍부하, 풍족, 풍족되, 풍족하, 피크닉, 한결같, 할렐루야, 합리, 합리되, 합리주의자, 합리하, 해답되, 해답하, 핵심, 행복, 행복되, 행복하, 향긋, 향유, 향유되, 향유하, 헌신, 현대화되, 현대화하, 현명, 현명되, 협동되, 협동하, 협력되, 협력하, 협상되, 협상하, 화해하, 환대, 환대되, 환대하, 환락, 환영, 환희, 활기차, 활약되, 활약하, 황홀, 황홀경, 황홀되, 황홀하, 휭재, 효능, 훌륭, 휴식되, 휴양, 휴양되, 휴양하, 흥행사, 희망, 히트, 히트되, 히트하, 가뿐하, 간원하, 간절하, 갈망하, 감개무량하, 감명되, 감미롭, 감복하, 감흥일, 개운하, 경탄하, 고맙, 공감하, 귀엽, 그립, 끌리, 달갑, 담담하, 대견하, 도취하, 동감하, 든든하, 들뜨, 듬직하, 떳떳하, 만만하, 맘놓, 매료되, 매혹하, 뭉클하, 믿, 반갑, 벅차, 보람차, 뿌듯하, 사모하, 살맛나, 상쾌하, 설레, 속시원하, 시원하, 신나, 신명나, 신바람나, 안도하, 안심하, 안정하, 은혜롭, 의기양양하, 자긍하, 자부하, 자신만만하, 자신하, 재미있, 전율하, 정겹, 정가, 좋아하, 즐겁, 찡하, 친애하, 탄복하, 통쾌하, 편안하, 평온하, 호감가, 홀가분하, 황송하, 후련하, 흐뭇하, 흠모하, 흡족하, 흥겹, 흥나다, 흥미롭, 흥미진진하

**부정단어 예**

파멸하, 파문, 파산, 패배, 패배되, 패배주의, 패배하, 편견, 편모, 편애, 편애되, 편애하, 편파, 편파되, 편파하, 폐결핵, 폐기, 폐렴, 폐병, 폐색되, 폐색하, 포기, 포기되, 포기하, 포악, 포악되, 포악하, 포위되, 폭격, 폭로, 폭로되, 폭로하, 폭발하, 표류되, 표류하, 피곤, 피난, 피난되, 피난하, 피로, 피로되, 피로하, 피부염, 핑계, 하위, 하자, 학대, 학대되, 학대하, 학살, 학살되, 학살하, 학질, 한심, 함정, 항복되, 항복하, 항소, 항의, 항의되, 항의하, 항의하니, 항진, 항진되, 항진하, 해고, 해이, 해적, 해적선, 핵, 핵무기, 허술, 허술하, 허위, 허탕되, 허탕하, 허튼소리, 헐뜯기, 험담, 험악되, 험악하, 헛되, 현혹, 현혹되, 현혹하, 혈우병, 혈전증, 혐기, 혐기되, 혐오, 혐오되, 혐오하, 협박자, 협심증, 협잡하, 협착, 형벌되, 형벌하, 호로, 호색, 호색되, 호색하, 호소하, 호통되, 호통하, 혼돈, 혼돈되, 혼돈하, 혼동, 혼동되, 혼동하, 혼란, 혼란되, 혼란스러워서, 혼란하, 혼잡, 혼잡되, 혼잡하, 홀리, 홍수, 홍진, 화, 화나, 화딱지, 화상, 화재, 환불 , 환불요청, 황폐, 황폐되, 회원탈퇴, 회의론자, 회피, 회피되, 회피하, 횡령되, 횡령하, 횡포, 횡포되, 후두염, 후퇴되, 후퇴하, 후회, 후회되, 후회하, 훔쳐내, 훔치, 훼방, 훼방되, 훼방하, 훼손, 훼손되, 훼손하, 흉내, 흐느낌, 흠잡, 흡연되, 흡연하, 흡혈귀, 흥청망청, 희롱, 희롱되, 희롱하, 희생, 희생자, 힘들, ㅠㅜ, ㅠㅠㅠㅠ, 가련하, 가소롭, 가엽, 가증스럽, 가책하, 갈등하, 갑갑하, 같잖, 개탄하, 거북하, 걱정하, 겁나, 격분하, 격양하, 격하,

# VOC 자동 감성분석

- 감성사전에 의하여 VOC 문장에 대한 긍부정성을 판정

| 입력문 | 감정점수 | 감정 | 부정점수 | 긍정점수 | 부정단어 | 긍정단어 |
|---|---|---|---|---|---|---|
| 수수료 결제자 변경 안녕하세요. 당사의 은행 조회서 수수료 결제자를 회계법인으로 변경 하고 싶은데 현재 시스템상 변경 불가라고 나옵니다. 어떻게 해결해야 하는지 방안 가이드 부탁드립니다. 감사합니다. | 0 | 중립 | -1 | 1 | 불가 | 감사하 |
| 인터넷청약시 청약신청에서 다음으로 안넘어 가네요 급합니다. 청약코자 하는 사람입니다. 제한구역이 아니라 1순위로 되는줄 알고 있으나 로그인해서 잘 넘어가다가 청약신청에서 주택형 선택시 활성화가 되지 않습니다. 왜 그런가요? 오늘 마감이라 빠른 답변 부탁합니다. | -1 | 부정 | -1 | 0 | 제한 | |
| 부적격 여부가 왜 전산상 반영이 안되는지 여부 1순위 청약하고 청약발표 관련해서 문의드립니다. 제 지인이 청약 당첨되었다가 청약조건 오기입력으로 부적격 처리되어 당첨 취소되었습니다. 본인 스스로 그렇게 알고 있으나 아파트투유에서 조회하면 부적격 사실이 나타나지 않습니다. 혹시 착오가 있어서 안내를 잘못한게 아닌지 생각됩니다. 제 지인의 부적격 여부에 대해서 확인 좀 해주시기 바랍니다. 당사자가 인터넷과 주택청약에 서툴러 제가 대신 동의를 얻어 확인요청합니다 | -4 | 부정 | -4 | 0 | 부적격,취소,착오,잘못 | |

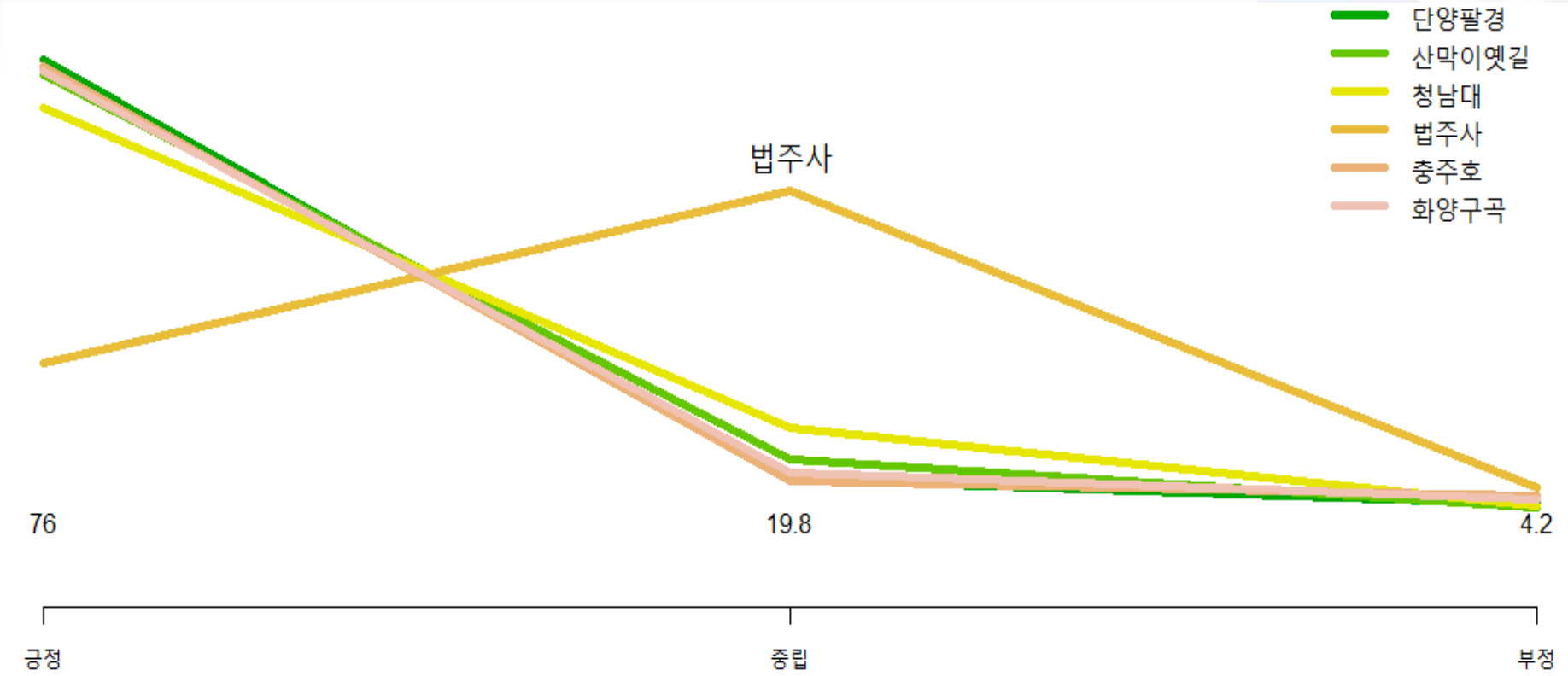
빅데이터 기획연구 총서 17-03호

# 충북지역 대표 관광 콘텐츠 전략화 방안 제시를 위한 빅데이터 분석

충북연구원
충 북 미 래 기 획 센 터

## 2.2 관광지별 감성분석

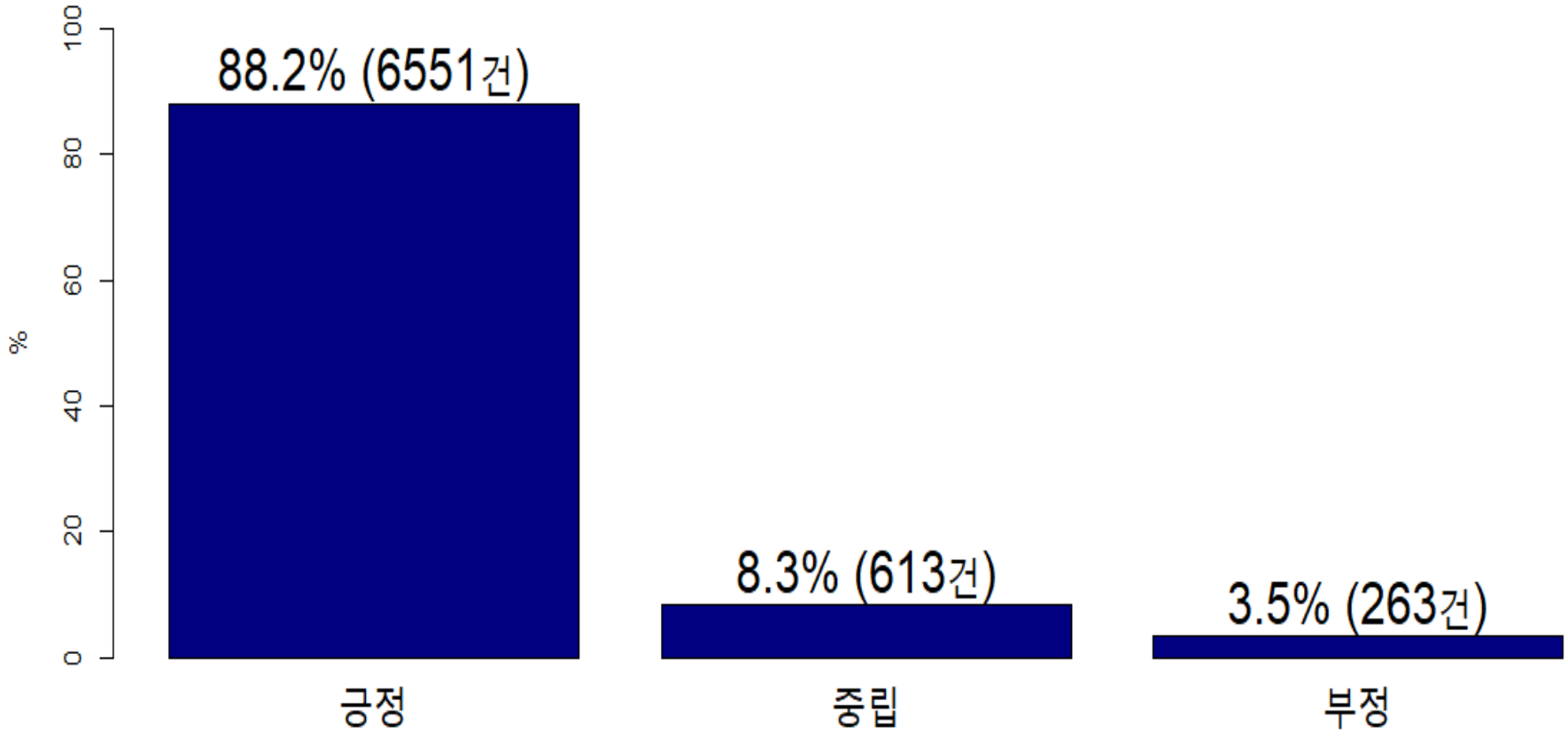
- 전체 감성분포




**“법주사”를 제외하고는 긍정이 가장 많음 (76%)**

## 2.2 관광지별 감성분석 : 1) 단양팔경

- 단양팔경 감성분포



## 2.2 관광지별 감성분석 : 1) 단양팔경

- **긍정문** 예문
  - "(13) 단양이 마늘로 유명하다며 마늘 정식집들이 많았는데 이 식당 역시 마늘 정식집 ..."
  - "(35) 단양은 마늘이 유명해서 인지 시장에 마늘 짱많고 모든 음식이름에 마늘이 ..."
  - "(227) 도담삼봉은 일출 명소로도 유명하다고 합니다 "
  - "(351) 그 팔경 중에서도 가장 유명하다고 할 수 있는 제1경도담삼봉을 찾았다"
- **부정문** 예문
  - "(5553) 남편은 소머리국밥을 시키고 나는 산채비빔밥을 시켰는데 가격도 비싸고 반찬도 없고"
  - "(6931) 마늘이 워낙 비싸서 음식 가격은 조금 나가는 것 같습니다"
  - "(612) 석문가는길은 가파르고 많은 계단을 올라가야해요"
  - "(750) 석문으로 올라가는 나무데크길은 길고 가파라서 숨을 가득 차게 합니다"

## 2.2 관광지별 감성분석 : 정리 2 – 지역별 주요 긍정/부정 요소

- 단양팔경
  긍정요소: 단양팔경, 도담삼봉, 드라이브, 마늘, 음식
  부정요소: 비싼 음식, 가파른 석문 계단, 힘든 도담삼봉의 교통편
- 산막이옛길
  긍정요소: 괴산호, 산막이옛길, 호수, 코스, 음식
  부정요소: 배의 줄이 길, 무서운 출렁다리
- 청남대
  긍정요소: 역대 대통령 별장, 삼겹살, 송어, 국화축제
  부정요소: 제약적인 사진촬영, 예약입장, 긴 코스
- 법주사
  긍정요소: 속리산과 법주사의 풍경, 계곡, 단풍
  부정요소: 비싼 입장료
- 충주호
  긍정요소: 충주호, 청풍호, 드라이브, 코스, 캠핑
  부정요소: 가파른 계단, 비싼 유람선
- 화양구곡
  긍정요소: 시원한 계곡, 물놀이
  부정요소: 힘든 산행, 가파른 계단

# 감성분석 *Sentiment Analysis*

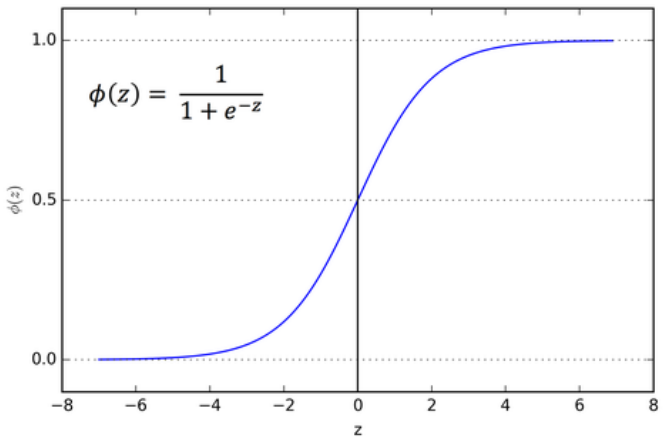
# 사전을 이용한 감성분석

# 감성분석 방법

- 문서에서 긍정적 단어가 나타나면 +1, 부정적 단어가 나타나면 -1을 쓰는 것과 같은 방법으로 감성점수를 계산

- 감성점수 > 0 이면 긍정적인 문서,
- 감성점수 < 0 이면 부정적인 문서,
- 감성점수 = 0 이면 중립적인 문서로 봄

- Sigmoid 함수 사용하여 값을 0~1 사이로 정규화 하는 방법도 있음



# 구글 드라이브와 연결

- 다음의 코드를 입력한 뒤, 절차에 따라 진행한다
- verification code 입력 후, authorization code를 입력한다
- 일정 시간(약 3시간) 이후에는 끊기므로, 매번 이 작업을 해주어야 한다

```
# from google.colab import auth
# auth.authenticate_user()
```

- from google.colab import drive
- drive.mount('/content/gdrive')

최종 화면

# 형태소분석기 설치

- !apt-get update
- !apt-get install g++ openjdk-8-jdk

- !pip install JPype1
- !pip install rhinoMorph

# 경로 변경

# 파일이 있는 곳으로 경로를 변경한다

- !cd /content/gdrive/My₩ Drive/pytest/

※ 별도의 코드 셀에서 진행해야 한다
※ 파일이 잘 읽어지지 않으면
- 가장 처음의 구글 드라이브와의 연결을 다시 한다(drive.mount('/content/gdrive'))
- '!'와 '₩'는 떼어도 보고, 붙여도 본다 (colab 문제)
- 그래도 안되면 ' ! '를 ' % '로 바꾸어본다
- 그래도 안되면 data = read_data('/content/gdrive/My Drive/pytest/ratings_small.txt', encoding='cp949')
- 그래도 안되면 !pip install -U -q PyDrive 를 실행해본다

※ PyCharm에서 진행한다면 아래의 코드로 경로를 변경한다

```
import os
os.chdir("C:/pytest")
```

# 데이터 로딩

```
def read_data(filename, encoding='cp949'):                    # 읽기 함수 정의
  with open(filename, 'r', encoding=encoding) as f:
    data = [line.split('Wt') for line in f.read().splitlines()]
    data = data[1:]                          # txt 파일의 헤더(id document label)는 제외하기
  return data

def write_data(data, filename, encoding='cp949'):             # 쓰기 함수도 정의
  with open(filename, 'w', encoding=encoding) as f:
    f.write(data)

#data = read_data('/content/gdrive/My Drive/pytest/ratings_small.txt', encoding='cp949')
data = read_data('ratings.txt', encoding='cp949')             # (긍정 10만, 부정 10만)
```

※ full data를 읽는다

# 전체 데이터 형태소 분석

```
import rhinoMorph
rn = rhinoMorph.startRhino()

morphed_data = ''
for data_each in data:
    morphed_data_each = rhinoMorph.onlyMorph_list(rn, data_each[1],
        pos=['NNG', 'NNP', 'VV', 'VA', 'XR', 'IC', 'MM', 'MAG', 'MAJ'])
    joined_data_each = ' '.join(morphed_data_each) # 문자열을 하나로 연결
    if joined_data_each:                           # 내용이 있는 경우만 저장하게 함
        morphed_data += data_each[0]+"₩t"+joined_data_each+"₩t"+data_each[2]+"₩n"

# 형태소 분석된 파일 저장
write_data(morphed_data, 'ratings_morphed.txt', encoding='cp949')
```

※ cloud에서는 2분 정도 소요

# 형태소 분석된 데이터 로딩

```
data = read_data('ratings_morphed.txt' , encoding='cp949')
print(len(data))
print(len(data[0]))
print(data[0])
```

```
197559
3
['8132799', '디자인 배우 학생 외국 디자이너 일구 전통 통하 발전 문화 산업 부럽 사실 우리나라 그 어렵 시절 끝 열정 지키 노라노
```

# 감정사전 읽기

```
data_id = [line[0] for line in data]          # 데이터 id
data_text = [line[1] for line in data]        # 데이터 본문
data_senti = [line[2] for line in data]       # 데이터 긍부정 부분

positive = read_data('positive.txt')          # 긍정 감정사전 읽기
negative = read_data('negative.txt')          # 부정 감정사전 읽기

print("positive:", positive)
print("negatvie:", negative)

pos_found = []                                # 각 문장에서 발견될 긍정어의 개수
neg_found = []                                # 각 문장에서 발견될 부정어의 개수
```

```
positive: [['가능'], ['가능하'], ['가락'], ['가치'], ['간단하'],
negatvie: [['가로막'], ['가로지르'], ['가리'], ['가명'], ['가시']
```

# 감정단어 파악

```
def cntWordInLine(data, senti):
    senti_found = []
    for onedata in data:
        oneline_word = onedata.split(' ')          # 한 줄의 데이터를 공백 단위로 분리하여 리스트로 저장
        senti_temp = 0                             # 그 줄에서 발견된 감정단어의 수를 담는 변수
        for sentiword in senti:                    # 감정사전의 어휘
            if sentiword[0] in oneline_word:       # sentiword[0] 하여 리스트 원소를 추출 (문자열)
                senti_temp += 1                    # 현재의 감정단어와 일치하면 숫자를 하나 올려 줌 (중복X)
        senti_found.append(senti_temp)             # 현재의 줄에서 찾은 감성단어의 숫자를 해당 위치에 저장
    return senti_found
```

```
data_senti_poscnt = cntWordInLine(data_text, positive)      # 발견된 긍정 단어의 숫자 파악
data_senti_negcnt = cntWordInLine(data_text, negative)      # 발견된 부정 단어의 숫자 파악

print(data_senti_poscnt[:20])
print(data_senti_negcnt[:20])
```

```
[5, 1, 0, 0, 2, 1, 0, 0, 0, 1, 1, 1, 0, 1, 2,
[1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1,
```

※ Cloud에서는 1분 정도 소요

# 감정점수 계산

```
# Pandas 데이터프레임으로 저장
import pandas as pd
newdata = pd.DataFrame({'id':data_id, 'text':data_text, 'original':data_senti,
                        'pos':data_senti_poscnt, 'neg':data_senti_negcnt})
senti_score = newdata['pos'] - newdata['neg']       # 긍정개수에서 부정개수를 뺌
newdata['senti_score'] = senti_score                # 그 수를 senti_score 컬럼에 저장

newdata.loc[newdata.senti_score > 0, 'new'] = 1     # 새로운 긍부정 기호
newdata.loc[newdata.senti_score <= 0, 'new'] = 0    # 새로운 긍부정 기호

# 처음에 기록된 긍부정과 새로 계산된 긍부정이 같은지 여부를 matched 컬럼에 저장
# original 컬럼은 문자로 되어 있으므로 숫자로 변환 뒤 비교
newdata.loc[pd.to_numeric(newdata.original) == newdata.new, 'matched'] = 'True'
newdata.loc[pd.to_numeric(newdata.original) != newdata.new, 'matched'] = 'False'
```

# 원점수와 비교 및 저장

```
score = newdata.matched.str.count('True').sum() / (newdata.matched.str.count('True').sum()
                                + newdata.matched.str.count('False').sum()) * 100
print(score)
```

※ 62.5 %

```
newdata.to_csv('newfile.csv', sep=',', encoding='cp949', index=False)    # csv 저장
newdata.to_csv('newfile2.txt', sep='₩t', encoding='cp949', index=False) # 또는 txt 저장
```

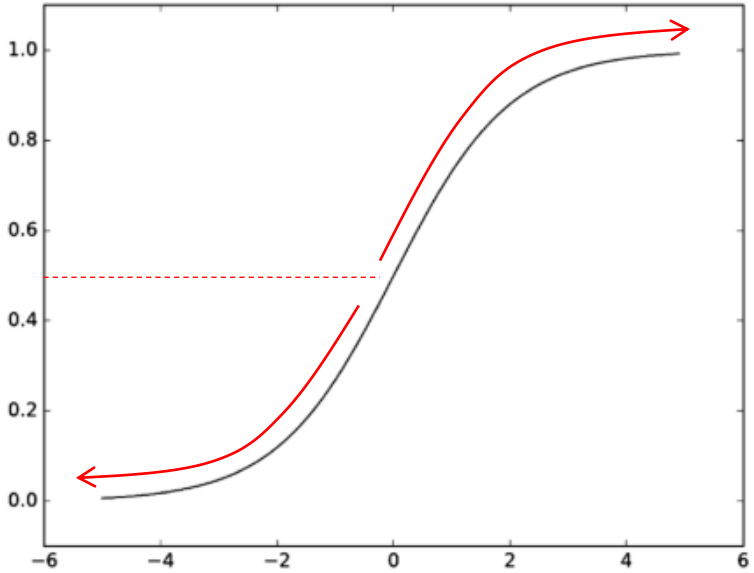
| | id | text | original | pos | neg | senti_score | new | matched |
|---|---|---|---|---|---|---|---|---|
| 0 | 8132799 | 디자인 배우 학생 외국 디자이너 일구 전통 통하 발전 문화 산업 부럽 사실 우리나라... | 1 | 5 | 1 | 4 | 1.0 | True |
| 1 | 4655635 | 폴리스스토리 시리즈 뉴 없 최고 | 1 | 1 | 0 | 1 | 1.0 | True |
| 2 | 9251303 | 와 연기 진짜 쩔 지루 생각하 몰입 그래 이런 진짜 영화 | 1 | 0 | 1 | -1 | 0.0 | False |
| 3 | 10067386 | 안개 자욱 하 밤하늘 뜨 초승달 같 영화 | 1 | 0 | 0 | 0 | 0.0 | False |
| 4 | 2190435 | 사랑 해보 사람 처음 끝 웃 있 영화 | 1 | 2 | 0 | 2 | 1.0 | True |

※ newdata.head()

※ 어떠한 단어가 업데이트 되어야 할지 살펴본다

# 시그모이드 점수 계산

- 시그모이드 함수는 모든 값을 0~1 사이로 변경해준다
- 따라서 각 문장에 긍정 혹은 부정 단어가 아무리 많아도 값을 정규화하는 효과를 갖는다
- 다음과 같이 하여 'sigmoid' 컬럼에 계산 결과를 넣을 수 있다




- import math
- def sigmoid(x):
  return 1 / (1 + math.exp(-x))
- newdata['sigmoid'] = newdata.senti_score.apply(sigmoid)

X의 값이 양수일 때는 클수록 작아짐 → 전체 결과는 1에 가까워짐
X의 값이 양수일 때는 작을수록 1에 가까워짐 → 전체 결과는 0.5에 가까워짐
X의 값이 음수일 때는 클수록 커짐 → 전체 결과는 0에 가까워짐
X의 값이 음수일 때는 작을수록 1에 가까워짐 → 전체 결과는 0.5에 가까워짐

math.exp(-1) = 1/2.71828... = 0.3678...
math.exp(-2) = 1/7.38905 = 0.1353...
math.exp(1) = 2.71828...
math.exp(2) = 7.38905...

# 결과 확인

- newdata.head()

| | id | text | original | pos | neg | senti_score | new | matched | sigmoid |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 8132799 | 디자인 배우 학생 외국 디자이너 일구 전통 통하 발전 문화 산업 부럽 사실 우리나라... | 1 | 5 | 1 | 4 | 1.0 | True | 0.982014 |
| **1** | 4655635 | 폴리스스토리 시리즈 뉴 없 최고 | 1 | 1 | 0 | 1 | 1.0 | True | 0.731059 |
| **2** | 9251303 | 와 연기 진짜 쩔 지루 생각하 몰입 그래 이런 진짜 영화 | 1 | 0 | 1 | -1 | 0.0 | False | 0.268941 |
| **3** | 10067386 | 안개 자욱 하 밤하늘 뜨 초승달 같 영화 | 1 | 0 | 0 | 0 | 0.0 | False | 0.500000 |
| **4** | 2190435 | 사랑 해보 사람 처음 끝 웃 있 영화 | 1 | 2 | 0 | 2 | 1.0 | True | 0.880797 |