



실습 10강 감성분석

❖ KNU 한국어 감성사전

15	:(1
16	:'(1
17	:- (1
18	:)	-1
19	:-)	1
20	:/	-1
21	:	1
22	:D	-1
23	:-D	-1
24	:p	0
25	:-P	-1
26	;	1
27	;)	1
28	;-)	1
29	;^;	-1
30	^///^	1
31	^^	1
32	^^//	1
33	^-^;	-1
34	^-^;;	-1
35	^-^; ; ; ;	-1
36	-_-^	-1

37	<3	1
38	=)	1
39	8-)	0
40	B-)	0
41	XD	1
42	ㄱ ㅅ	1
43	ㄱ ㅅ ㄷ	1
44	가격미 싸다	1
45	가짜스로	0
46	가짜미 사귀며	1
47	가짜미하다	1
48	가꾸러뜨리다	-1
49	가꾸러트리다	-1
50	가난	-2
51	가난뱅이	-2
52	가난살이	-2
53	가난살이하다	-2
54	가난설음	-2
55	가난에	-2
56	가난에 꼬들려서	-2
57	가난하게	-2
58	가난하고	-2

라이브러리 불러오기

감성사전을 불러 오기 위한 라이브러리 설치와 불러오기

```
install.packages("readr")  
library(readr)
```

감성사전을 불러 오기 위한 라이브러리 설치와 불러오기

```
rev <- read_delim("review_dict.txt", delim="\t",  
  col_names=c("word", "score"))
```

```
> rev <- read_delim("review_dict.txt", delim="\t", col_names=c("word", "score"))  
Rows: 57 Columns: 2
```

-- Column specification -----

Delimiter: "\t"

chr (1): word

dbl (1): score

i Use `spec()` to retrieve the full column specification for this data.

i Specify the column types or set `show_col_types = FALSE` to quiet this message.

감성사전 확인

```
head(rev, 10)
```

상위 10개

```
> head(rev, 10)  
# A tibble: 10 x 2  
  word      score  
  <chr>    <dbl>  
1 감사합니다      2  
2 고생했네요    -2  
3 이하네요    -2  
4 끝내 주세요      2  
5 기가막히게      1  
6 늦었네요    -2  
7 만족합니다      2  
8 만족해요      2  
9 맛은 있네요      1  
10 맛있게      2
```

라이브러리 불러오기

```
# 감성분석을 위한 라이브러리 설치와 불러오기  
install.packages("SentimentAnalysis")  
library(SentimentAnalysis)
```

```
# sentdic에 감성분석 전용 사전의 단어에 따른 점수로 가중치를 매길 수  
있게 기준 설정  
sentdic <- SentimentDictionaryWeighted(words = rev$word,  
                                         scores = rev$score)
```

```
# 점수가 0보다 크면 긍정어(positive), 0보다 작으면 부정어(negative)  
로 기준 설정  
sentdic <- SentimentDictionary(rev$word[rev$score>0],  
                               rev$word[rev$score<0])
```

```
# 감성사전 기준 확인  
summary(sentdic)
```

```
> summary(sentdic)  
Dictionary type: binary (positive / negative)  
Total entries: 57  
Positive entries: 32 (56.14%)  
Negative entries: 25 (43.86%)
```

텍스트 전처리

미리 크롤링 해놓은 review.txt 파일을 txt에 저장

```
txt <- readLines("review.txt", encoding = "UTF-8")
```

```
head(txt)
```

```
> head(txt)
```

```
[1] "닭이 너무 맛있어요 최고!! 육질이 살아있음"  
[3] "기가막히게 맛있습니다. 사장님 감사합니다."  
[5] "완전 만족합니다. 재구매 각이네요."
```

```
"배송도 빠르고 상품도 좋습니다. ^^"
```

```
"닭이 너무 작아요! 양이 작은 편인데도 부족하네요. ><"
```

```
"삼계탕에 넣었는데 양이 기대 이하네요..."
```

stringr 패키지를 통해 텍스트 대체 실시

```
install.packages("stringr")
```

```
library(stringr)
```

마침표, 쉼표, 느낌표, 물음표 제거

```
txt_2 <- str_replace_all(txt, "([.,!?])", "")
```

```
head(txt_2)
```

```
> head(txt_2)
```

```
[1] "닭이 너무 맛있어요 최고 육질이 살아있음"  
[3] "기가막히게 맛있습니다 사장님 감사합니다"  
[5] "완전 만족합니다 재구매 각이네요"
```

```
"배송도 빠르고 상품도 좋습니다 ^^"
```

```
"닭이 너무 작아요 양이 작은 편인데도 부족하네요 ><"
```

```
"삼계탕에 넣었는데 양이 기대 이하네요"
```

txt_2 데이터 타입 확인

```
class(txt_2)
```

텍스트마이닝

문서형 데이터 형태 변환을 위해 tm 패키지 설치 및 라이브러리 불러오기

```
install.packages("tm")  
library(tm)
```

txt_2 데이터를 Corpus 형태로 변환

```
co_txt <- Corpus(VectorSource(txt_2))  
class(co_txt) # 데이터 타입 확인
```

inspect(co_txt) # co_txt 형태 살펴보기

```
> inspect(co_txt) # co_txt 형태 살펴보기
```

```
<<SimpleCorpus>>
```

```
Metadata: corpus specific: 1, document level (indexed): 0
```

```
Content: documents: 30
```

```
[1] 닭이 너무 맛있어요 최고 육질이 살아있음  
[2] 배송도 빠르고 상품도 좋습니다 ^^  
[3] 기가막히게 맛있습니다 사장님 감사합니다  
[4] 닭이 너무 작아요 양이 작은 편인데도 부족하네요 ><  
[5] 완전 만족합니다 재구매 각이네요  
[6] 삼계탕에 넣었는데 양이 기대 이하네요  
[7] 배송이 늦었네요 보내주신 아이스팩이 터져서 치운다고 고생했네요 포장에 신경써야 할듯  
[8] 닭이 엄청 크고 실하네요 들어갈만한 냄비가 없어서 큰 냄비를 샀네요 맛도 좋습니다  
[9] 아이스박스가 터져서 왔네요 비닐도 벗겨지고 다시 구매하고 싶지 않네요 그나마 닭이 상하지 않은게 다행인듯  
[10] 닭에 기름이 너무 많네요 기름제거가 너무 힘들었네요 ㅠㅠ  
[11] 살이 너무 찝찝해요 요리를 잘 못한 건지 닭 자체가 그런건지 잘 모르겠네요  
[12] 너무 신선합니다 살도 너무 부드러워요 오랜만에 가족들과 맛있는 삼계탕 해먹었네요 감사합니다  
[13] 가격이 저렴한데도 불구하고 닭이 너무 신선하고 맛있네요 닭에 또 구매할 생각입니다
```

DTM 생성

```
# Corpus 상태에서 DocumentTermMatrix 형태로 변환  
dtm_txt <- DocumentTermMatrix(co_txt)
```

```
# DocumentTermMatrix 형태 살펴보기  
inspect(dtm_txt)
```

```
> inspect(dtm_txt)  
<<DocumentTermMatrix (documents: 30, terms: 214)>>  
Non-/sparse entries: 268/6152  
Sparsity : 96%  
Maximal term length: 7  
Weighting : term frequency (tf)  
Sample :  
Terms  
Docs 건지 것 너무 닭이 만족해요 먹었습니다 양이 작은 잘 큰  
11 1 0 1 0 0 0 0 0 2 0  
14 0 0 1 1 0 0 0 0 0 0  
15 1 0 1 0 0 1 0 0 1 0  
16 0 0 1 0 0 1 1 1 1 0  
20 0 0 0 0 0 0 0 0 0 1  
22 0 0 0 0 0 0 1 0 0 1  
27 0 1 0 0 0 0 0 0 1 0  
30 0 0 0 0 0 0 0 0 0 0  
8 0 0 0 1 0 0 0 0 0 1  
9 0 0 0 1 0 0 0 0 0 0
```

감성사전을 이용한 분석

```
# dtm_txt를 위에서 만든 감성 기준 사전 sentdic을 이용해 분석
res <- analyzeSentiment(dtm_txt, language="korean",
  rules=list("sentiment"=list(ruleSentiment, sentdic)))
```

```
# 결과확인
head(res)
```

```
> head(res)
  sentiment
1  0.3333333
2  0.5000000
3  0.7500000
4 -0.4285714
5  0.5000000
6 -0.2000000
```

선정된 문

감성점수 계산

sentiment가 0보다 크면 긍정, 0이면 중립, 0보다 작으면 부정으로 표시

```
res$pn <- ifelse(res$sentiment>0,"Positive",  
                 ifelse(res$sentiment==0,"Neutral","Negative"))
```

```
head(res)
```

```
> head(res)  
  sentiment      pn  
1  0.3333333 Positive  
2  0.5000000 Positive  
3  0.7500000 Positive  
4 -0.4285714 Negative  
5  0.5000000 Positive  
6 -0.2000000 Negative
```

결과 요약해서 보기

```
table(res$pn)
```

```
> table(res$pn)
```

```
Negative  Neutral Positive  
      13         4      13
```

시각화

```
# 결과 별도 저장하되 데이터 프레임 형태로 변환  
df_res <- as.data.frame(table(res$pn))
```

```
# 데이터 프레임 열이름 별도 지정  
names(df_res) <- c("res", "freq")
```

```
# 파이차트에 퍼센트 표시를 위해 pct 열 생성  
df_res$pct <- round(df_res$freq/sum(df_res$freq)*100, 1)  
df_res
```

```
> df_res  
  res freq  pct  
1 Negative  13 43.3  
2 Neutral   4 13.3  
3 Positive  13 43.3
```

↑
반응값

퍼센트 식

순수값
첫번째

시각화

파이차트로 감성 분석 결과 확인하기

```
pie(df_res$freq, labels = paste(df_res$res, df_res$pct, "%"),  
    main = "생닭 판매 고객 리뷰 감성 분석 결과")
```

