Text Mining For Korean

이번 섹션에서는 트위터 데이터를 활용한 텍스트 마이닝을 소개한다.
주로 소개된 내용은 아래와 같다.

- 데이터 획득
- 전처리
- 핵심어 추출, 단어간의 관계 파악
- 워드 클라우드
- 트위터 클러스터링

데이터 획득

twitteR 패키지를 활용해 트위터 데이터를 가져온다.

교육 목적상 본문 분석에서는 필자가 직접 제공하는 데이터를 사용하고, 트위터 데이터 패치 해오는 코드만을 살펴본다.

```
library(twitteR)
#
n <- 200

keyword <- "삼성전자"
#
keyword <- enc2utf8(keyword)
#
rdmTweets <- searchTwitter(keyword, n)

load(url("http://dl.dropbox.com/u/8686172/twitter.RData"))

nDocs <- length(rdmTweets)
```

사실 텍스트 전처리는 데이터 상황에 따라 가변적이다. 따라서 로(raw) 데이터를 먼저 확인해보고 본인이 어떤 목적으로 분석을 하는지 그 방향과 데이터가 맞는지 확인하고 방향과 일시키기 위해서 어떻게 전처리를 해야 되는 고민이 필요하다.

일단 필자는 아래와 같은 전처리 계획을 세웠다.

1. @ 트윗 태그 제거
2. URL 제거
3. 명사 추출
4. 문장 부호 제거
5. 숫자 제거
6. 영어 소문자화
7. 불용어 제거

이를 위해 KoNLP, tm 패키지가 필요하다.

```
library(KoNLP)
```

```
## Error: there is no package called 'KoNLP'
```

```
library(tm)


df <- do.call("rbind", lapply(rdmTweets, as.data.frame))

removeTwit <- function(x) {
    gsub("@[[:graph:]]*", "", x)
}

df$ptext <- sapply(df$text, removeTwit)

removeURL <- function(x) {
    gsub("http://[[:graph:]]*", "", x)
}

df$ptext <- sapply(df$ptext, removeURL)
useSejongDic()
```

```
## Error: 함수 "useSejongDic"를 찾을 수 없습니다
```

```
df$ptext <- sapply(df$ptext, function(x) {
    paste(extractNoun(x), collapse = " ")
})
```

```
## Error: 함수 "extractNoun"를 찾을 수 없습니다
```

```

# build corpus
myCorpus_ <- Corpus(VectorSource(df$ptext))
myCorpus_ <- tm_map(myCorpus_, removePunctuation)
myCorpus_ <- tm_map(myCorpus_, removeNumbers)
myCorpus_ <- tm_map(myCorpus_, tolower)
```

```
## Error: 'utf8towcs'에 잘못된 입력 'I just ousted 축컴백벤자민 ������ as the
## mayor of 삼성전자서비스센터 on '가 있습니다
```

```
myStopwords <- c(stopwords("english"), "rt")
myCorpus_ <- tm_map(myCorpus_, removeWords, myStopwords)
```

tm패키지는 R에서 텍스트 마이닝을 위해 가장 빈번히 사용되는 패키지이다. 특히나 이 패키지는 Corpus라는 자료구조를 기반으로 분석을 수행하기 때문에 Corpus로 데이터를 변형하기 위한 과정이 필요하다.
이 Corpus내에서 단어에 대한 집계와 빈도수 그리고 단어간의 관계에 대한 코드를 소개한다.

- 단어-트윗 간의 매트릭스
- 10이상의 빈도수를 가진 단어들
- "lg" 단어에 대한 관련 단어

```
myTdm <- TermDocumentMatrix(myCorpus, control = list(wordLengths = c(2, Inf)))

# inspect frequent term
findFreqTerms(myTdm, lowfreq = 10)
```

```
##  [1] "lg"                "lte"               "oled"
##  [4] "pc"                "tv"                "가처분"
##  [7] "가칭"              "개국에서"          "갤럭시"
## [10] "결정"              "경기대회"          "공개"
## [13] "공모"              "공장"              "국내"
## [16] "권오현"            "기능"              "김치냉장고"
## [19] "냉각"              "노트"              "다양"
## [22] "대용량"            "대표이사부회장가" "메탈"
## [25] "미국"              "반대"              "반도체"
## [28] "부정적"            "부회장"            "산시성"
## [31] "삼성"              "삼성전자와"        "서울뉴시스김민기"
## [34] "서울연합뉴스"      "세계"              "소송"
## [37] "스마트폰"          "스마트폰인"        "시안"
## [40] "시장"              "아이폰"            "아이폰가"
## [43] "아이폰에"          "애플"              "연속"
## [46] "영향"              "이동통신"          "이유"
## [49] "전국"              "전략"              "전망"
## [52] "전자"              "제공"              "제품"
## [55] "중국"              "증권"              "진행"
## [58] "착공식"            "참석"              "철회"
## [61] "출시"              "출장"              "침해"
## [64] "태블릿"            "특허"              "판매금지"
## [67] "폰앤케이스"        "프리미엄"          "하기"
## [70] "하반기"            "한국투데이"        "한투블로그기자"
## [73] "현대"              "확실"              "확정"
## [76] "후원"
```

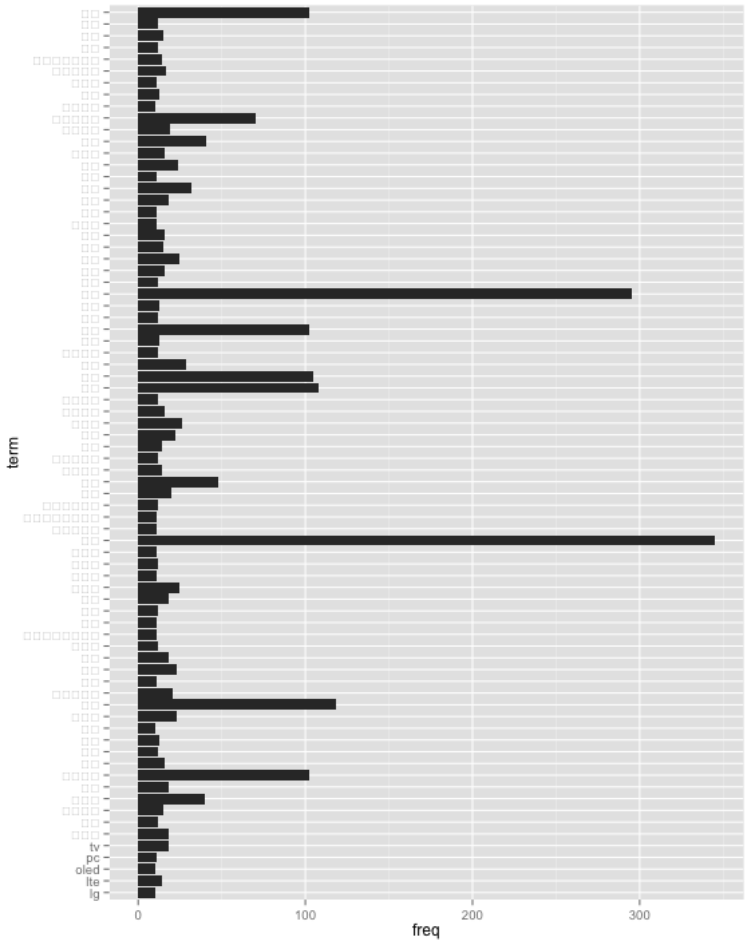
```
# inspect associations
findAssocs(myTdm, "lg", 0.25)
```

```
##             lg           가전           기용       김연아엔           맞수
##           1.00           0.90           0.90           0.90           0.90
##           모델           소녀         손연재           위해       이미지를
##           0.90           0.90           0.90           0.90           0.90
## 점입가경인데요           시대           구축       프리미엄           문제
##           0.90           0.63           0.51           0.39           0.36
##           발열         대용량           신경           하기           oled
##           0.36           0.35           0.34           0.34           0.31
##         경쟁이           궤도   발광다이오드           양산           유기
##           0.31           0.31           0.31           0.31           0.31
##         차세대             tv           제품
##           0.31           0.28           0.28
```

이들에 대한 플로팅을 해본다.

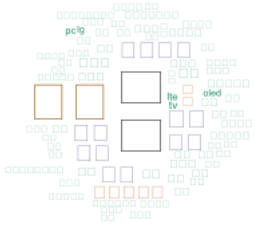
ggplot2를 이용한 막대그림

```
library(ggplot2)

termFrequency <- rowSums(as.matrix(myTdm))
termFrequency <- subset(termFrequency, termFrequency >= 10)

ggplot(data.frame(term = names(termFrequency), freq = termFrequency), aes(term,
    freq)) + geom_bar() + coord_flip()
```



단어 빈도수에 기반한 워드 크라우드

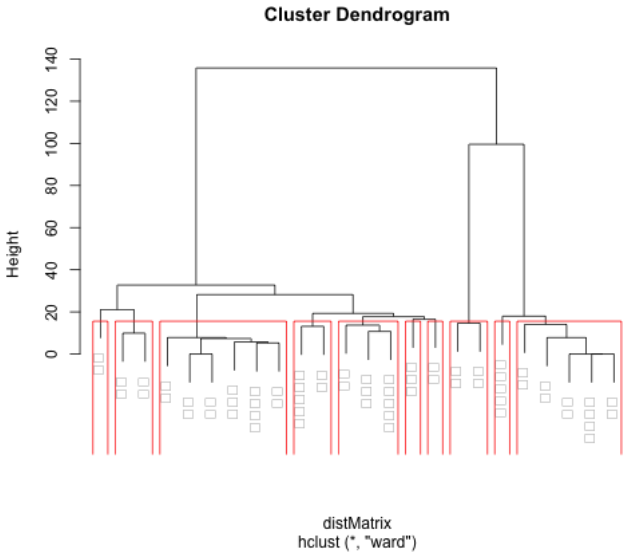
```
# Word Cloud

library(wordcloud)

m <- as.matrix(myTdm)

wordFreq <- sort(rowSums(m), decreasing = TRUE)

set.seed(375)

pal <- brewer.pal(8, "Dark2")

wordcloud(words = names(wordFreq), freq = wordFreq, min.freq = 10, random.order = F,
    rot.per = 0.1, colors = pal)
```



단어 기반 계층적 클러스터링

1. 어느정도 다양한 트윗에서 존재하는 존재하는 단어들만 추린다.
2. 스케일링
3. 거리 행렬 계산
4. 덴드로그램(dendrogram) 플로팅, 10개의 클러스터 만을 추린다.

```
myTdm2 <- removeSparseTerms(myTdm, sparse = 0.95)
m2 <- as.matrix(myTdm2)

distMatrix <- dist(scale(m2))

fit <- hclust(distMatrix, method = "ward")

plot(fit)

rect.hclust(fit, k = 10)
```




```
# (groups<-cutree(fit,k=10))
```

k-means 클러스터링

```
m3 <- t(m2)
k <- 4
kmres <- kmeans(m3, k)

round(kmres$centers, digits = 3)
```

```
##   갤럭시 경기대회  공개  기능 김치냉장고  다양  삼성  세계  소송  시장
## 1  0.014        0 0.007 0.014      0.147 0.112 1.168 0.035 0.021 0.098
## 2  0.195        0 0.366 0.000      0.000 0.024 1.512 0.366 1.098 0.024
## 3  2.143        0 0.000 1.000      0.000 0.071 1.000 0.000 0.000 0.500
## 4  0.000        1 0.000 1.000      0.000 0.000 1.000 0.000 0.000 0.000
##   아이폰 아이폰가  애플  연속  영향 전국  전자  진행  출시 태블릿  특허
## 1  0.028    0.007 0.077 0.021 0.000    0 1.007 0.007 0.133   0.00 0.014
## 2  0.537    0.366 2.366 0.000 0.707    0 0.854 0.366 0.000   0.22 0.951
## 3  0.000    0.000 0.000 0.000 0.000    0 1.000 0.000 0.929   0.50 0.000
## 4  0.000    0.000 0.000 1.000 0.000    1 1.000 0.000 0.000   0.00 0.000
##   폰앤케이스 한국투데이 후원
## 1      0.000      0.000    0
## 2      0.000      0.000    0
## 3      0.000      0.000    0
## 4      0.686      0.167    1
```

```
for (i in 1:k) {
    cat(paste("cluster ", i, " : ", sep = ""))
    s <- sort(kmres$centers[i, ], decreasing = T)
    cat(names(s)[1:3], "\n")
    # print(head(rdmTweets[which(kmres$cluster ==i)],n=3))
}
```

```
## cluster 1 : 삼성 전자 김치냉장고
## cluster 2 : 애플 삼성 소송
## cluster 3 : 갤럭시 기능 삼성
## cluster 4 : 경기대회 기능 삼성
```

Silhouette Plot을 보여준다.

```
library(fpc)
```

```
## Error: there is no package called 'fpc'
```

```
pamResult <- pamk(m3, metric = "manhattan")
```

```
## Error: 함수 "pamk"를 찾을 수 없습니다
```

```
(k <- pamResult$nc)
```

```
## Error: 개체 'pamResult'이 없습니다
```

```
pamResult <- pamResult$pamobject
```

```
## Error: 개체 'pamResult'이 없습니다
```

```
# print cluster medoids

for (i in 1:k) {
    cat(paste("cluster", i, ":"))
    cat(colnames(pamResult$medoids)[which(pamResult$medoids[i, ] == 1)], "\n")
    # print tweets in cluster i print(rdmTweets[pamResult$clustering==i])
}
```

```
## cluster 1 :
```

```
## Error: 개체 'pamResult'이 없습니다
```

```
# plotclusteringresult
layout(matrix(c(1, 2), 2, 1))  #settotwographsperpage
plot(pamResult, color = F, labels = 4, lines = 0, cex = 0.8, col.clus = 1, col.p = pamResult$clustering)
```

```
## Error: 개체 'pamResult'이 없습니다
```

```
layout(matrix(1))
```

Reference

- R Data Mining