동국대학교
dongguk university

텍스트 마이닝

엄진영

텍스트 마이닝(Text mining)

- **문자로 된 데이터에서 가치 있는 정보를 얻어 내는 분석 기법**
- **SNS나 웹 사이트에 올라온 글을 분석해 사람들이 어떤 이야기를 나누고 있는지 파악할 때 활용**
  - 형태소 분석(Morphology Analysis) : 문장을 구성하는 어절들이 어떤 품사로 되어 있는지 분석

- **분석 절차**
  1. 형태소 분석
  2. 명사, 동사 형용사 등 의미를 지닌 품사 단어 추출
  3. 빈도표 만들기
  4. 시각화

힙합 가사 텍스트 마이닝

- 한글 자연어 분석 패키지인 koNLP(Korean Natural Language Processing)를 이용하면 한글 데이터로 형태소 분석가능
- KoNLP는 자바가 설치되어 있어야 사용가능
- Java 다운로드 및 설치
  - https://www.java.com/ko/download/manual.jsp

힙합 가사 텍스트 마이닝

- 패키지 설치 및 로드

```
# 패키지 설치
install.packages("rJava")
install.packages("memoise")
install.packages("KoNLP")

# 패키지 로드
library(KoNLP)

## Checking user defined dictionary!

library(dplyr)
```

- 패키지 로드 에러 발생할 경우 - java 설치 경로 확인 후 경로 설정

```
# java 폴더 경로 설정
Sys.setenv(JAVA_HOME="C:/Program Files/Java/jre1.8.0_111/")
```

KoNLP 오류 해결

- R이 계속해서 버전을 업그레이드 해서 구 패키지들이 작동이 안되는 경우가 있음
- 현재 KoNLP 패키지가 CRAN에서 삭제된 상태라 install.packages()를 이용해 설치할 수 없음

- 구패키지를 설치할 수 있는 방법
  1. CRAN 사이트에서 직접 패키지를 다운로드해서 경로를 지정
  2. 패키지 다운로드 받을 수 있는 CRAN사이트를 명시해서 다운

KoNLP 오류 해결

- java와 rJava 설치

```
install.packages("multilinguer")

library(multilinguer)

install_jdk()
```

- KoNLP 의존성 패키지 설치

```
install.packages(c('stringr', 'hash', 'tau', 'Sejong',
'RSQLite', 'devtools'), type = "binary")
```

- 깃헙 버전 KoNLP 설치

```
install.packages("remotes")

remotes::install_github('haven-jeon/KoNLP', upgrade =
"never", INSTALL_opts=c("--no-multiarch"))

library(KoNLP) #최종적으로 "KoNLP" 패키지를 불러옵니다
```

사전 설정 오류 해결


```
devtools::install_github('haven-jeon/NIADic/NIADic',
build_vignettes = TRUE)

# 설치한 JAVA version에 따라 달라집니다

Sys.setenv(JAVA_HOME='C:/Program Files
(x86)/Java/jre1.8.0_251')

# "woorimalsam" dic을 불러옵니다

buildDictionary(ext_dic = "woorimalsam")

useNIADic()  # "NIADic" dic을 불러옵니다
```

힙합 가사 텍스트 마이닝

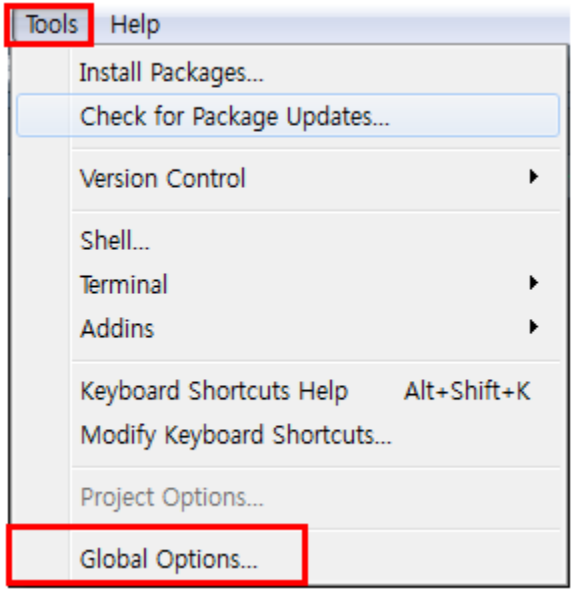
- 사전 설정하기

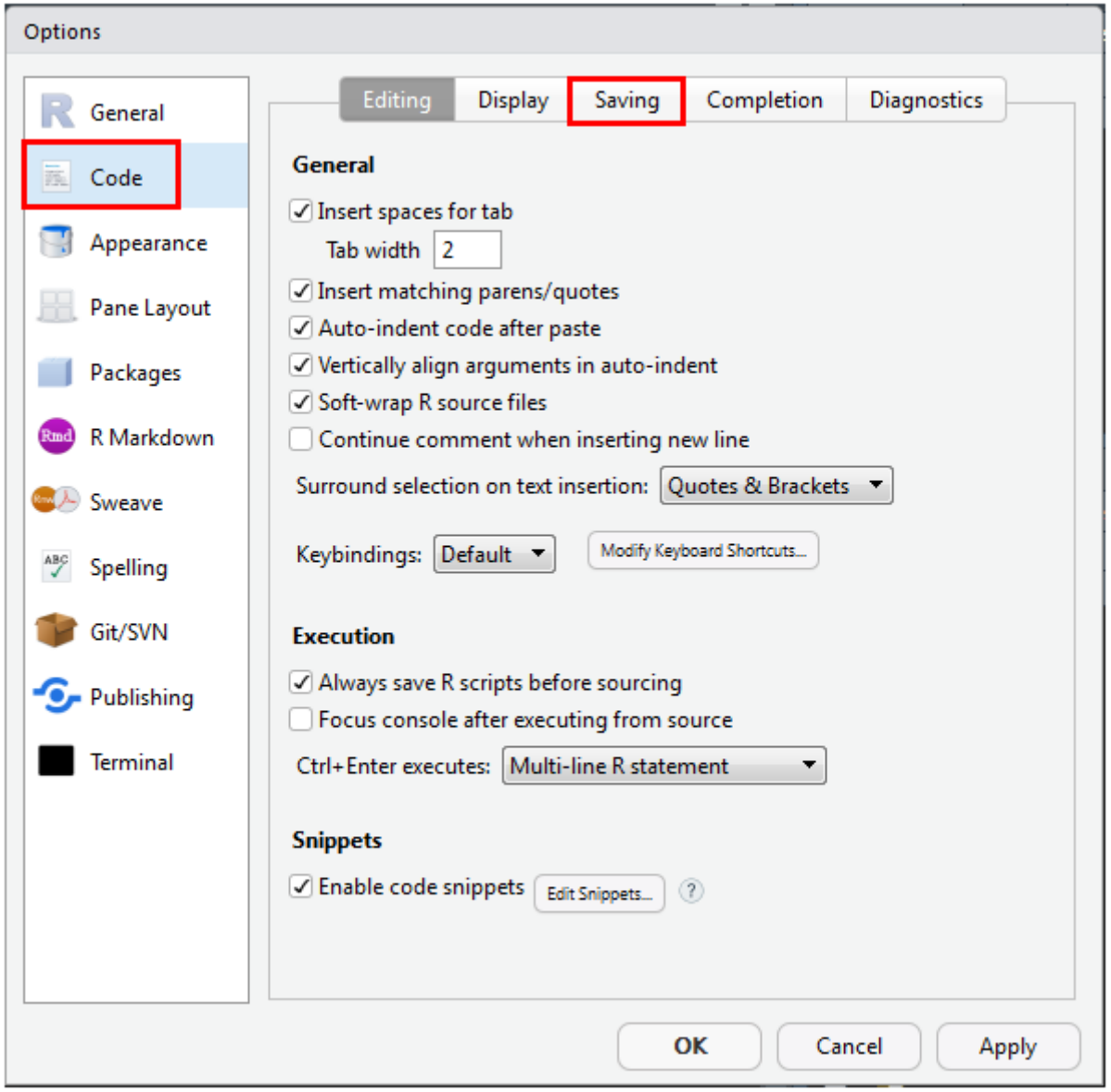
```
useNIADic()

## Backup was just finished!
## 983012 words dictionary was built.
```

- 데이터 준비
  - 깃허브(bit.ly/doit_rd)에서 hiphop.txt 파일을 다운로드

```
# 데이터 불러오기
txt <- readLines("hiphop.txt")

## Warning in readLines("hiphop.txt"): incomplete final line
found on
## 'hiphop.txt'

head(txt)

## [1] "\"보고 싶다"                    "이렇게 말하니까 더 보고 싶다"
## [3] "너희 사진을 보고 있어도"        "보고 싶다"
## [5] "너무 야속한 시간"               "나는 우리가 밉다"
```

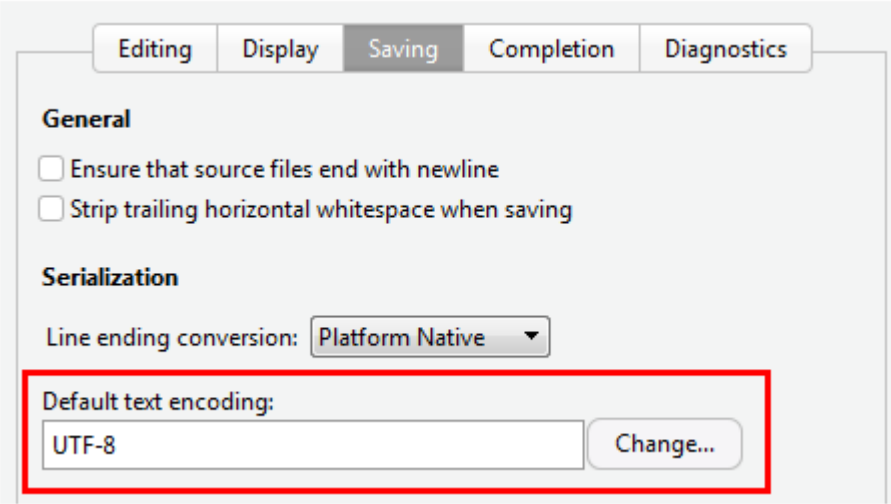
한글 깨짐 현상 없애기

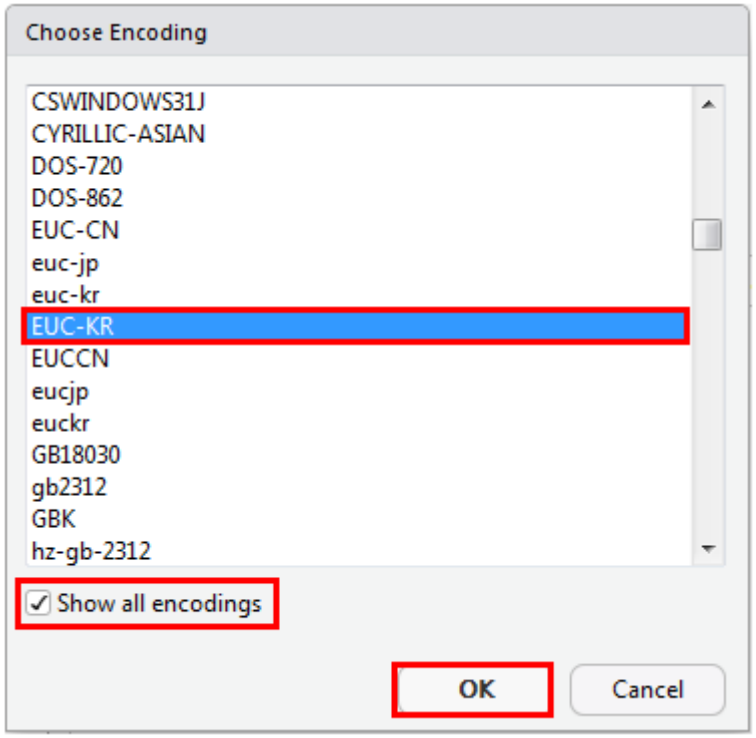

Tools
Help
Install Packages...
Check for Package Updates...
Version Control
Shell...
Terminal
Addins
Keyboard Shortcuts Help
Alt+Shift+K
Modify Keyboard Shortcuts...
Project Options...
Global Options...


Options
General
Code
Appearance
Pane Layout
Packages
R Markdown
Sweave
Spelling
Git/SVN
Publishing
Terminal
Editing
Display
Saving
Completion
Diagnostics
General
Insert spaces for tab
Tab width
2
Insert matching parens/quotes
Auto-indent code after paste
Vertically align arguments in auto-indent
Soft-wrap R source files
Continue comment when inserting new line
Surround selection on text insertion:
Quotes & Brackets
Keybindings:
Default
Modify Keyboard Shortcuts...
Execution
Always save R scripts before sourcing
Focus console after executing from source
Ctrl+Enter executes:
Multi-line R statement
Snippets
Enable code snippets
Edit Snippets...
OK
Cancel
Apply

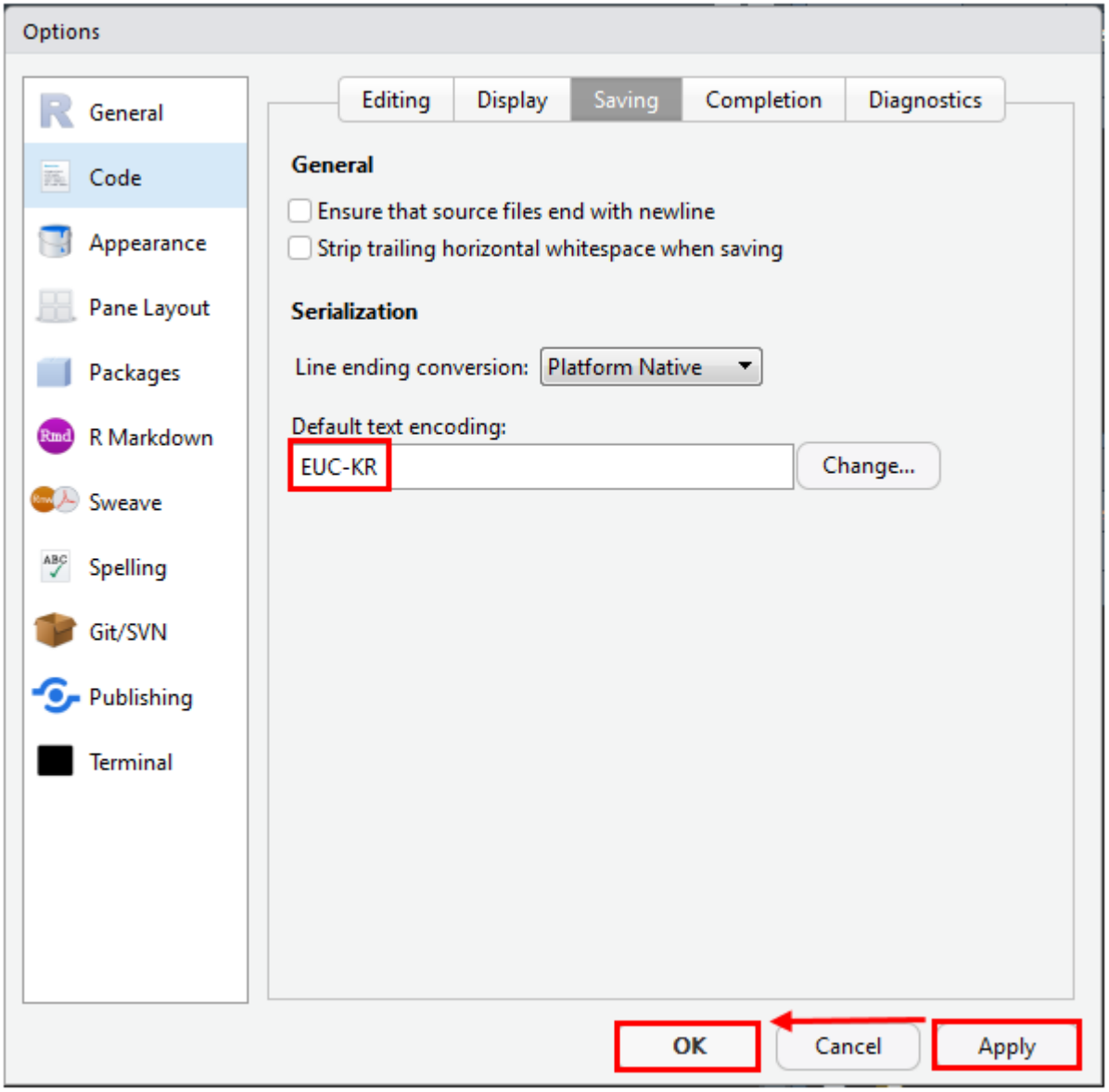
한글 깨짐 현상 없애기

- 기본 값이 UTF-8로 되어있는데 EUC-KR로 수정해보자





한글 깨짐 현상 없애기


Options
General
Code
Appearance
Pane Layout
Packages
R Markdown
Sweave
Spelling
Git/SVN
Publishing
Terminal
Editing
Display
Saving
Completion
Diagnostics
General
Ensure that source files end with newline
Strip trailing horizontal whitespace when saving
Serialization
Line ending conversion:
Platform Native
Default text encoding:
EUC-KR
Change...
OK
Cancel
Apply

힙합 가사 텍스트 마이닝

- 멜론 차트 랩/힙합 부문 2017년 3월 둘째주 (SongList.xlsx)

순위	제목	가수	가사
1	봄날	방탄소년단	보고 싶다
2	에라 모르겠다	BIGBANG	No I don't wanna go too fast
3	우리집을 못 찾겠군요(Feat. 볼빨간사춘기)	매드클라운	착해 빠진 게 독한 소리 할 때
4	기다렸다 가	다이나믹 듀오, 첸(CHEN)	일이 피곤했나
5	당신의 밤(Feat. 오혁)	황광희 X 개코	별 하나에 추억과
6	저 별	헤이즈(Heize)	혹시 저 별도 나를 보고 있을까
7	마에스트로(Maestro)	창모(CHANGMO)	다섯살때부터 나는
8	피 땀 눈물	방탄소년단	내 피 땀 눈물 내 마지막 춤을
9	돌아오지마(Feat. 용준형 Of 비스트)	헤이즈(Heize)	아직도 비가 내리면
10	Not Today	방탄소년단	All the underdogs in the world
11	BERMUDA TRIANGLE(Feat. Crush, DEAN)	지코(ZICO)	손목에 Rolex 이젠 Boring
12	휘파람	BLACKPINK	Hey boy
13	반창고	MC 몽, 허각	어서 빨리 어른이 되고 싶은
14	쏘아	하하 X MINO	우린 거북선
15	GIRLFRIEND	BIGBANG	어쩜 그렇게 예쁠 수가 있을까
16	A Supplementary Story : You Never Walk Alone	방탄소년단	예 신은 왜 자꾸만
17	Outro : Wings	방탄소년단	Take me to the sky
18	Day Day(Feat. 박재범)(Prod. by GRAY)	BewhY(비와이)	한 번 돌아가 보자구
19	Lost	방탄소년단	눈을 감고 아직 여기 서 있어
20	puzzle	씨잼(C Jamm), BewhY(비와이)	I'll Take 1 I'll take 2
21	아름다워	창모(CHANGMO)	널 이제 놓아줘야 될것같애
22	And July(Feat. DEAN, DJ Friz)	헤이즈(Heize)	약이라도 타놓은 걸까 yeah
23	만세	양세형 X BewhY	우리는 단 한 가지만
24	남아있어(Feat. Crush)	로꼬	유난히 뜨거웠던 지난 여름과
25	둘! 셋!(그래도 좋은 날이 더 많기를)	방탄소년단	꽃길만 걷자

힙합 가사 텍스트 마이닝

- 특수문자 제거

```
install.packages("stringr")

library(stringr)

# 특수문제 제거
txt <- str_replace_all(txt, "\\W", " ")
```

str_sub (fruit, start, end) <- replace

fruit	banana
↓	str_sub (fruit, 1, 3) <- "str"
fruit	strana

str_replace(string, pattern, replacement)

fruit	banana
↓	str_replace (fruit, "a", "s")
	bsnana

str_replace_all(string, pattern, replacement)

fruit	banana
↓	str_replace_all (fruit, "a", "s")
	bsnsns

힙합 가사 텍스트 마이닝

가장 많이 사용된 단어 알아보기

```
# 명사 추출하기
extractNoun("대한민국의 영토는 한반도와 그 부속도서로 한다")

## [1] "대한민국" "영토"     "한반도"   "부속도서" "한"

# 가사에서 명사추출
nouns <- extractNoun(txt)

# 추출한 명사 list를 문자열 벡터로 변환, 단어별 빈도표 생성
wordcount <- table(unlist(nouns))
```

힙합 가사 텍스트 마이닝

- 자주 사용된 단어 빈도표 만들기

```
# 데이터 프레임으로 변환
df_word <- as.data.frame(wordcount, stringsAsFactors = F)

# 변수명 수정
df_word <- rename(df_word,
                  word = Var1,
                  freq = Freq)

# 두 글자 이상 단어 추출
df_word <- filter(df_word, nchar(word) >= 2)

top_20 <- df_word %>%
  arrange(desc(freq)) %>%
  head(20)
```

힙합 가사 텍스트 마이닝


```
top_20
```

```
##    word freq
## 1   you   89
## 2    my   86
## 3   YAH   80
## 4    on   76
## 5  하나   75
## 6  오늘   51
## 7   and   49
## 8  사랑   49
## 9  like   48
## 10 우리   48
## 11  the   43
## 12 시간   39
## 13 love   38
## 14   to   38
## 15   we   36
## 16   it   33
## 17   em   32
## 18  not   32
## 19 역사   31
## 20 flex   30
```

워드 클라우드 만들기

- 패키지 준비하기

```
# 패키지 설치
install.packages("wordcloud")

# 패키지 로드
library(wordcloud)

## Loading required package: RColorBrewer

library(RColorBrewer)
```

워드 클라우드 만들기

- 단어 색상 목록 만들기

```
pal <- brewer.pal(8,"Dark2")   # Dark2 색상 목록에서 8개 색상 추출
```

- 워드 클라우드 생성

```
set.seed(1234)                          # 난수 고정
wordcloud(words = df_word$word,         # 단어
          freq = df_word$freq,          # 빈도
          min.freq = 2,                 # 최소 단어 빈도
          max.words = 200,              # 표현 단어 수
          random.order = F,             # 고빈도 단어 중앙 배치
          rot.per = .1,                 # 회전 단어 비율
          scale = c(4, 0.3),            # 단어 크기 범위
          colors = pal)                 # 색깔 목록
```

워드 클라우드 만들기


whatever
Baby 믿음 right
거리 위인들처
호랑나비 doctor 거짓말 몸몸몸 자신
drum lov BANG hands 하지 빠라바빠
BOOM 그대 하루 walk All 어디 UH Hoga
그림 이상 사람 re 생각 역사 필요 young
say 화살 To 누구 가사 be wa 남아있
전화 여자 하나 toda bass
maestr Don wanna 소리 머리
never in the 어제 no your 그것
ready Girl gir 언제 yea
give yo 영 to YAH not Not 여기
친구 조준 들이 let 무대
Am 눈물 love 고민 bab
때론 Yea on 펴 노래
chic tha 하늘 you like 날개 흥 모르겠 목소 분위기
one madeleine ba Let it day 사랑 don 시계
마리 자유 goes 이름 ban 너 Yes 취향 of 이미 for hear 가슴
시작 thi worl em 오늘 Hulk 마음 mysel
why get 아 my up spread
Lost Tic dow want 몸몸몸매
Love 기억 순간 ma 시간 me go that
kno 무리 우린 bo fl so bang this
Whatever You and 세상 등 can NO
We got fly 얼굴 flex we Wron
ladies are Hey 미안 이젠 know put
out 사실 당신 is 파라 But 주먹
And just min girlfrien da 아이 need dance
미래 So nee YA boy 지금 바람 서로
BAN BOOMBAYA 겨울 때문 인생 마지막

워드 클라우드 만들기

- 단어 색상 바꾸기

```
pal <- brewer.pal(9,"Blues")[5:9]   # 색상 목록 생성
set.seed(1234)                      # 난수 고정

wordcloud(words = df_word$word,     # 단어
          freq = df_word$freq,      # 빈도
          min.freq = 2,             # 최소 단어 빈도
          max.words = 200,          # 표현 단어 수
          random.order = F,         # 고빈도 단어 중앙 배치
          rot.per = .1,             # 회전 단어 비율
          scale = c(4, 0.3),        # 단어 크기 범위
          colors = pal)             # 색상 목록
```

워드 클라우드 만들기


YAH
하나
onyou
my
사랑
오늘
and
we
you
like
the
to
in
love
시간
세상
girlfrien
flex
Hey
know
me
up
it
I
나
너
아
이름
눈물
하늘
기억
우린
미안
이젠
지금
Not
not
your
no
toda
wa
re
생각
역사
누구
사람
All
hands
BANG
doctor
거짓말
whatever
right
위인들처
믿음
Baby
거리
호랑나비
drum
lov
BOOM
그대
하루walk
어디
UH
Hoga
young
필요
빠라바빠
몸몸몸
자신
그림
이상
say
화살
전화
maestr
Don
never
gir
ready
give
yo
친구
Am
때론
tha
chic
one
madeleine
ba
Let
goes
마리자유
시작
thi
worl
why
get
Lost
Tic
Love
kno
무리
Whatever
We
got
fly
ladies
are
out
사실
And
당신
girlfrien
min
미래
So
nee
YA
boy
BOOMBAYA
겨울
때문
인생
마지막
da
아이
need
dance
바람
져로
But
파라
put
주먹
Wron
this
can
NO
bang
so
fl
bo
얼굴
You
ma
순간
dow
want
몸몸몸매
go
that
spread
mysel
for
hear
시계
가슴
don
yes
취향of
모르겠
록소
분위기
bap
도래
페
무대
let
여기
yea
그것
소리
머리
bass
남아있
여자
wan
언제
조준
늘
이
Yea
day
ban

국정원 트윗 텍스트 마이닝

- **국정원 계정 트윗 데이터**
  - 국정원 대선 개입 사실이 밝혀져 논란이 됐던 2013년 6월, 독립 언론 뉴스타파가 인터넷을 통해 공개
  - 국정원 계정으로 작성된 3,744개 트윗

국정원 트윗 텍스트 마이닝

- 데이터 준비하기
  - 깃허브(bit.ly/doit_re)에서 twitter.csv파일을 다운로드

```
# 데이터 로드
twitter <- read.csv("twitter.csv",
                    header = T,
                    stringsAsFactors = F,
                    fileEncoding = "UTF-8")

# 변수명 수정
twitter <- rename(twitter,
                  no = 번호,
                  id = 계정이름,
                  date = 작성일,
                  tw = 내용)

# 특수문자 제거
twitter$tw <- str_replace_all(twitter$tw, "\\W", " ")
```

국정원 트윗 텍스트 마이닝


```
head(twitter$tw)

## [1] "민주당의 ISD관련 주장이 전부 거짓으로 속속 드러나고있다  미국이 ISD를 장악하고 있다고 주장하지만 중재인 123명 가운데 미국인은 10명뿐이라고 한다 "
## [2] "말로만  미제타도  사실은  미제환장  김정일 운구차가 링컨 컨티넬탈이던데 북한의 독재자나 우리나라 종북들이나 겉으로는 노동자  서민을 대변한다면서 고급 외제차  아이팟에 자식들 미국 유학에 환장하는 위선자들인거죠"
## [3] "한나라당이 보수를 버린다네요 뭔가착각하는모냥인에 국민들이보수를싫어하는게 아니라뻘짓거리하는분들을싫어하는겁니다야당이진보어쩌고저쩌고한다고해서그들을조아한다고생각하면대착각"
## [4] "FTA를 대하는 현명한 자세  사실 자유주의 경제의 가장 큰 수해자는 한국이죠  농어업분야 피해를 줄이는 정부대안을 최대한  보완하고 일자리 창출 등 실익을 최대화해 나가는게 현실적인 대처자세일듯 "
## [5] "곽노현씨 갈수록 가관입니다  뇌물질에 아들 병역 의혹까지  도대체 아이들이 뭘 보고 배우겠습니까  이래도 자리 연연하시겠습니까 "
## [6] "과거 집권시 한미FTA를 적극 추진하던 세력이 이제 집권하면 폐기하겠다고 주장합니다  어이없어 말도 안 나오네요  표만 얻을 수 있다면 국가 안보나 경제가 어떻게 되든 상관없다는 무책임한 행태들  우리 정치의 후진성을 드러내는 거죠 "
```

국정원 트윗 텍스트 마이닝

- 단어 빈도표 만들기

```
# 트윗에서 명사추출
nouns <- extractNoun(twitter$tw)

# 추출한 명사 list를 문자열 벡터로 변환, 단어별 빈도표 생성
wordcount <- table(unlist(nouns))

# 데이터 프레임으로 변환
df_word <- as.data.frame(wordcount, stringsAsFactors = F)

# 변수명 수정
df_word <- rename(df_word,
                  word = Var1,
                  freq = Freq)
```

국정원 트윗 텍스트 마이닝

- 두 글자 이상으로 된 단어 추출, 빈도 상위 20개 단어 추출

```
# 두 글자 이상 단어만 추출
df_word <- filter(df_word, nchar(word) >= 2)

# 상위 20개 추출
top20 <- df_word %>%
  arrange(desc(freq)) %>%
  head(20)
```

국정원 트윗 텍스트 마이닝

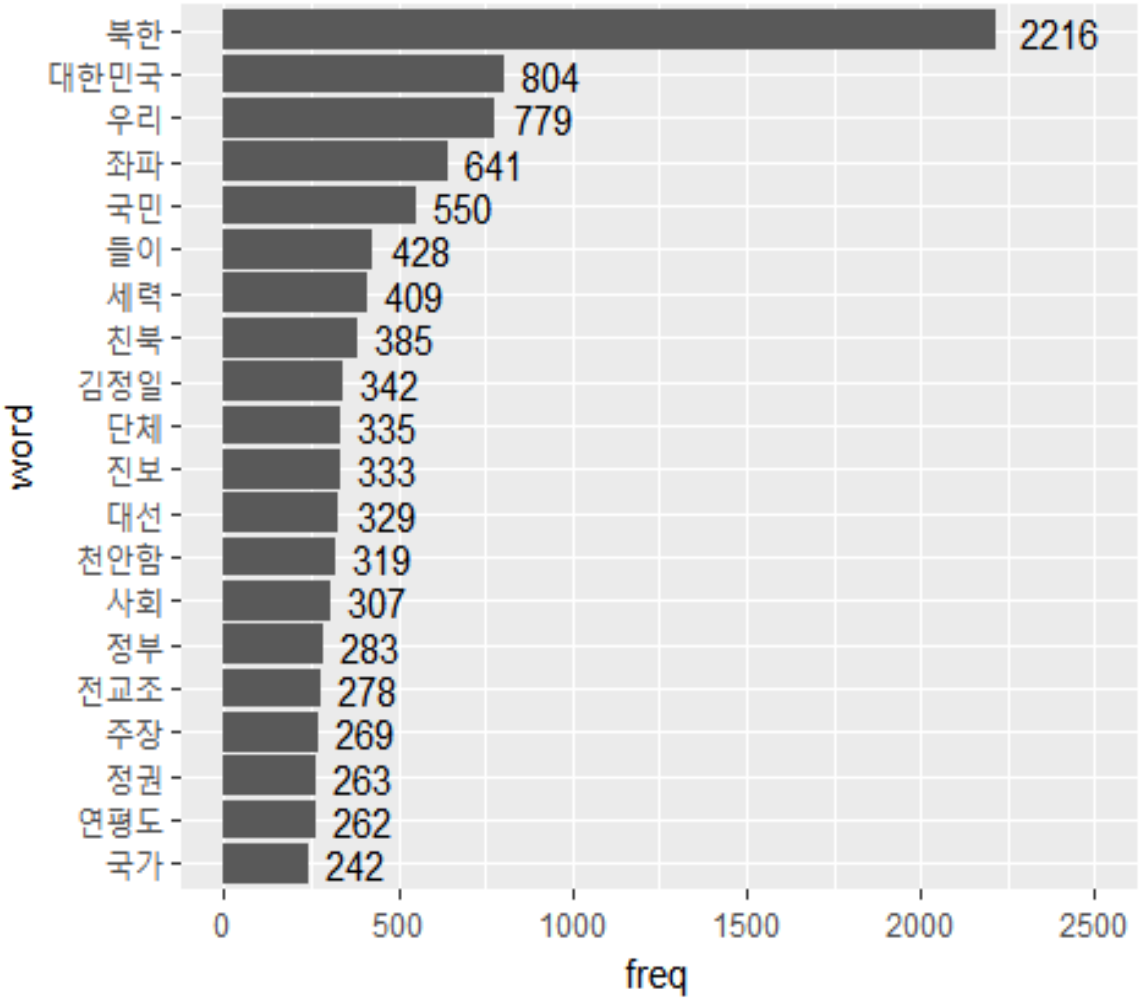

```
top20

##         word freq
## 1        북한 2216
## 2  대한민국  804
## 3        우리  779
## 4        좌파  641
## 5        국민  550
## 6        들이  428
## 7        세력  409
## 8        친북  385
## 9      김정일  342
## 10       단체  335
## 11       진보  333
## 12       대선  329
## 13     천안함  319
## 14       사회  307
## 15       정부  283
## 16     전교조  278
## 17       주장  269
## 18       정권  263
## 19     연평도  262
## 20       국가  242
```

국정원 트윗 텍스트 마이닝

- 단어 빈도 막대 그래프 만들기

```
library(ggplot2)

order <- arrange(top20, freq)$word   # 빈도 순서 변수 생성

ggplot(data = top20, aes(x = word, y = freq)) +
  ylim(0, 2500) +
  geom_col() +
  coord_flip() +
  scale_x_discrete(limit = order) + # 빈도 순서 변수 기준 막대 정렬
  geom_text(aes(label = freq), hjust = -0.3)      # 빈도 표시
```

국정원 트윗 텍스트 마이닝



국정원 트윗 텍스트 마이닝

- 워드 클라우드 만들기

```
pal <- brewer.pal(8,"Dark2")          # 색상 목록 생성
set.seed(1234)                        # 난수 고정

wordcloud(words = df_word$word,       # 단어
          freq = df_word$freq,        # 빈도
          min.freq = 10,              # 최소 단어 빈도
          max.words = 200,            # 표현 단어 수
          random.order = F,           # 고빈도 단어 중앙 배치
          rot.per = .1,               # 회전 단어 비율
          scale = c(6, 0.2),          # 단어 크기 범위
          colors = pal)               # 색상 목록
```

국정원 트윗 텍스트 마이닝

동국대학교
dongguk university

국정원 트윗 텍스트 마이닝

- 색깔 바꾸기

```
pal <- brewer.pal(9,"Blues")[5:9]   # 색상 목록 생성
set.seed(1234)                      # 난수 고정

wordcloud(words = df_word$word,     # 단어
          freq = df_word$freq,      # 빈도
          min.freq = 10,            # 최소 단어 빈도
          max.words = 200,          # 표현 단어 수
          random.order = F,         # 고빈도 단어 중앙 배치
          rot.per = .1,             # 회전 단어 비율
          scale = c(6, 0.2),        # 단어 크기 범위
          colors = pal)             # 색상 목록
```

국정원 트윗 텍스트 마이닝