2020년 전산통계 프로젝트

1. 개요

2. 신문사별 실시간 뉴스의 단어 출현 빈도 시각화

3. 중심어와 주제를 통한 신문사별 비교

4. 부록(R 코드 첨부)

0조
정보통계 · 보험수리학과
20000000 000
20000000 000
20151109 이찬솔

1. 개요

○ 웹 크롤링 (web crawling), 스크래핑 (scraping), 파싱 (parsing) 방법을 활용하여, 6개 신문사의 실시간 뉴스를 수집하고 전처리를 거쳐서 **상위 50개 단어**의 출현 빈도를 시각화 및 비교.
○ 자료 수집을 위한 포털사이트의 태그는 각 신문사 홈페이지의 실시간 뉴스 헤드라인들을 기준으로 함.
○ 실시간 뉴스는 **2020년 12월 4일 20시 15분을 기준**으로 함.
○ 조원 3인이 각각 2개의 언론사를 담당하여 파싱, 전처리, 시각화를 진행하였으며, 도출된 단어구름에 대해 조원 전원이 파악 및 비교를 진행함.
○ 실행한 R코드는 하나로 취합하여 파일로 보관. (부록#2)

2. 신분사별 실시간 뉴스의 단어 출현 빈도 시각화

○ **한국일보**

- 12.03일 펼쳐진 **수능**과 관련된 “문제”, “수능”, “수험생”의 단어 빈도가 각각 20, 12, 10으로서 중심어임을 알 수 있음.
- 그 다음으로 **코로나 현황**과 관련된 “확진자”, “경기”, “정부”, “감염”의 단어들 역시 각각 15, 14, 14, 10의 높은 빈도를 보임.
- **수능, 재난지원금, 시각장애인 안내견등**의 주제들에 공통적으로 관련되어있는 “택시”는 12의 높은 빈도를 보임.
- 그 외의 단어들로 **법무부와 검찰**과 관련된 “검찰”, “후보자”, “ 장관”, “사건”, **총리측근의혹과** 관련된 “대표”, “혐의”, “수사”, 12.04일 펼쳐진 **해외축구와 손흥민**에 관련된 “리그”, “시즌”, “퍼포먼스”등이 단어구름內 존재.

○ **중앙일보**

- **총리측근 의혹**과 관련된 “수사”, “관련”, “이낙연” “대표”, “측근”들의 빈도가 각각 8, 7, 5, 5로서 단어구름의 중심어임을 확인 할 수 있음.
- **법무부와 검찰**과 관련된 “검찰”, “법원”, “법무부”, “박은정”등의 단어들 또한 빈도가 각각 8, 5, 6, 5로서 중심어임을 알 수 있음.
- **코로나 현황**과 **수능**에 관련된 “검사”, “수능”등의 단어들은 상대적으로 빈도가 낮음.
- 그 외의 단어들로 **월성원전**과 관련된 “월성”, “원전”, **성범죄**과 관련된 “강간”등이 단어구름內 존재.

○ **조선일보**

- **법무부와 검찰**과 관련된 “윤석열”, “추미애”, “법무부”, “직무정지”등의 단어들이 각각 33, 22, 15, 15의 압도적으로 높은 빈도를 가짐.
- **총리측근 의혹**과 관련된 “측근”, “이낙연”등의 단어들이 각각 15, 11로서 뒤를 이음.
- 그 외의 단어들로 **월성원전**과 관련된 “월성”, “원전”, **미군 합참의장의 발언**과 관련된 “북한”, “발사”. “합참의장”등이 단어구름內 존재.
- **코로나 현황**과 **수능, 성범죄**에 관련된 단어들은 단어구름內 포함되지 않음.

○ **동아일보**

- **법무부와 검찰**과 관련된 “윤석열”이 “이용”이 중심어임을 확인 할 수 있음.
- 그 외의 중심어로서 **총리측근 의혹**과 관련된 “이낙연”, “측근”, “논란”등이 있으며, **수능**과 관련된 “이용”, “수험생”, **정부 지지율**과 관련된 “대통령”, “지지율”등이 있음.
- 그 외의 단어들로 **코로나 현황**과 **성범죄,** 12.04일 2700을 돌파한 **코스피지수**들의 주제들과 관련된 “사망”, “여성”, “강간”, “경찰”, “코스피”, 등이 단어구름內 존재.

○ **한겨레**

- **법무부와 검찰**과 관련된 “장관”, “후보자”가 18과 10의 높은 빈도를 가짐으로 중심어임을 확인 할 수 있음.
- 그 외의 중심어로서 **코로나 현황**과 **성범죄**와 관련된 “방역”, “경찰”, “여성”, “백신”등이 있음.
- **중국의 한한령 완화**와 관련된 “드라마”, “중국”의 단어들도 높은 빈도를 기록, 단어 구름의 중심부에 위치함.
- **총리측근 의혹**과 **정부 지지율**에 관련된 단어들은 단어구름에 존재하지 않음.

○ **경향신문**

- **법무부와 검찰**과 관련된 “검찰”과 “개혁”, “윤석열”이 중심어임을 확인 할 수 있음.
- 그 외의 중심어로서 **코스피지수**와 관련된 “코스피”, “돌파”, **코로나 현황**과 관련된 “확진”, “검사”, “격리”등의 있음.
- 그 외의 단어들로 **해외축구와 손흥민**에 관련된 “스포츠”, “손흥민”, **수능**과 관련된 “수능”, 문제“등이 단어구름內 존재.
- **총리측근 의혹**과 **정부 지지율**에 관련된 단어들은 단어구름에 존재하지 않음.

3. 중심어와 주제를 통한 신문사별 비교

○ **코로나 현황** (“확진자”, “감염”등)
- 조선일보를 제외한 5개의 신문사 모두 단어구름內 포함하고 있거나, 중심어로 가지고 있음.

○ **수능** (“수능”, “수험생”등)
- 조선일보와 한겨례를 제외한 4개의 신문사 모두 단어구름內 포함하고 있거나, 중심어로 가지고 있음.

○ **법무부와 검찰** (“윤석열”, “검찰”, “장관”등)
- 6개 신문사중 한국일보를 제외한 모든 나머지 신문사들이 해당 주제에 대한 단어들을 중심어로 가지는 모습을 알 수 있으며, 5개 신문사 모두 단어구름內 “윤석열”을 가지고 있음.
- 특히 조선일보와 동아일보는 “윤석열”이 다른 단어들에 비해 상대적으로 출현 빈도수가 압도적으로 큰 것을 단어구름을 통해 알 수 있음.

○ **총리측근 의혹** (“측근”, “이낙연”등)
- 위의 5개의 신문사중 조선일보, 중앙일보, 동아일보는 중심어로서 **총리측근 의혹**에 대한 단어들을 중심어로 같이 가지고 있거나, 단어구름內 포함.
- 반면 한겨례, 경향신문은 그렇지 않음.

○ **월성원전** (“월성”등)
- **총리측근 의혹**과 관련된 단어를 중심어로 가지고 있는 위의 3개 신문사(조선일보, 중앙일보, 동아일보)중 2개의 신문사(중앙일보, 조선일보)가 **월성원전**과 관련된 단어들을 단어구름內 포함하고 있음.
- 반면 한겨례, 경향신문은 그렇지 않음.

○ **해외축구** (“손흥민”등)
- 한국일보와 경향신문만이 단어구름內 해당주제에 대한 단어를 포함.

4. 부록

○ 각 신문사 홈페이지에서 추출된 단어 집계행렬中 신문명(예 : “동아”)과, “더보기”, “이전”, “다음”등을 비롯한 불필요한 단어들은 wordResult지정 과정에서 제거함.

○ 단어 집계행렬中 존재하는 중복단어들 (예: “이낙연”, “이낙연측”)은 wordResult지정 과정에서 동일 단어로 지정 및 빈도수를 병합함.

○ 단어 집계행렬中 존재하는 글자가 일부 손실된 단어들은 wordResult지정 과정에서 names(wordResult)[ ] 코드로 수정함. (예: “코스” ⇒ “코스피”)

○ 태그 수집과 전처리 과정중 조선일보의 것만 중복된 문장이 연속으로 3개가 나오는 것을 확인하였고, 정확한 빈도수 비교를 위하여 wordResult지정 과정에서 일괄적으로 3으로 나눔.

#1 (R 코드)

```
library(RSQLite, KoNLP, tm, wordcloud, httr, rvest, XML)
# URL 요청 (한국부터 경향일보 순으로 모든 객체에 3부터 8 까지 번호 부여)
Hankook <- "https://www.hankookilbo.com/" ; web3 <- GET(Hankook)
Joongang <- "https://joongang.joins.com/" ; web4 <- GET(Joongang)
Chosun <- "https://www.chosun.com/" ; web5 <- GET(Chosun)
Donga <- "https://www.donga.com/" ; web6 <- GET(Donga)
Hani <- "http://www.hani.co.kr/" ; web7 <- GET(Hani)
Khan <- "http://www.khan.co.kr/" ; web8 <- GET(Khan)
# HTML 파싱
html3 <- htmlTreeParse(web3, useInternalNodes = T,
                       trim = T, encoding = "utf-8") ; rootNode3 <- xmlRoot(html3)
                                                    ...
                                                    ...
html8 <- htmlTreeParse(web8, useInternalNodes = T,
                       trim = T, encoding = "utf-8") ; rootNode8 <- xmlRoot(html8)
# 태그 자료수집 (12월 4일 20시 15분 기준)
khan <- paste(xpathSApply(rootNode8, "//div[@class='spotMain']", xmlValue),
              xpathSApply(rootNode8, "//div[@class='section_mediastory']", xmlValue),
              xpathSApply(rootNode8, "//div[@class='section_wrap']", xmlValue))
hani <- xpathSApply(rootNode7, "//div[@class='type1']", xmlValue)
dong <- xpathSApply(rootNode6, "//div[@class='headline_type2']", xmlValue)
chos <- xpathSApply(rootNode5, "//div[@class='grid grid__container grid__container-centered']", xmlValue)
joong <- paste(xpathSApply(rootNode4, "//div[@class='float_left']", xmlValue),
               xpathSApply(rootNode4, "//div[@class='list_vertical']", xmlValue))
hank <- xpathSApply(rootNode3, "//div[@class='container main']", xmlValue)
# 수집한 자료 전처리
khan_pre <- gsub('[\r\n\t]', ' ', khan) ; khan_pre <- gsub('[[:punct:]]', ' ', khan_pre) # 특수문자 제거
khan_pre <- gsub('[a-z]+', ' ', khan_pre) ; khan_pre <- gsub('[A-Z]+', ' ', khan_pre) # 영문제거
khan_pre <- gsub('[[:cntrl:]]', ' ', khan_pre) ; khan_pre <- gsub('\\d+', ' ', khan_pre) # 문장부호, 숫자 제거
khan_pre <- gsub('\\s+', ' ', khan_pre) ; khan_pre # 2개 이상 공백 교체 및 전처리 결과 확인
                                                    ……
                                                    ……
                                                    ……
hank_pre <- gsub('\\s+', ' ', hank_pre) ; hank_pre
```

각 언론사 별 단어추출 ⇒ 말뭉치 생성 ⇒ 단어vs문서 집계행렬 생성 및 data.frame화 ⇒ 단어 출현 빈도 와 그에 따른 상위 50개 단어에 대한 wordcloud생성

```
news_noun3 <- extractNoun(hank_pre) newsCorpus3 <- Corpus(VectorSource(news_noun3))
TDM3 <- TermDocumentMatrix(newsCorpus3, control = list(wordLengths = c(4,16)))
tdm.df3 <- as.data.frame(as.matrix(TDM3)) ; wordResult3 <- sort(rowSums(tdm.df3), decreasing = TRUE)
wordResult3 <- wordResult3[wordResult3 > 1] ; head(wordResult3,50)
wordResult3 <- wordResult3[-3] ; wordResult3 <- head(wordResult3, 50)
myNames3 <- names(wordResult3) ; df3 <- data.frame(word=myNames3, freq=wordResult3)
pal <- brewer.pal(8, "Paired")
wordcloud(df3$word, df3$freq, min.freq=2, random.order=F, scale =c(5, 0.3), rot.per=0.1, colors=pal, family="malgun")
                                                    ……
                                                    ……
                                                    ……
wordcloud(df8$word, df8$freq, min.freq=2, random.order=F, scale =c(5, 0.3), rot.per=0.1, colors=pal, family="malgun")
```

#2 (R 파일)

첨부파일 (**#2 (R파일) 7조.R**) 1개. 끝.