Ch1-2. '비건' 키워드 연관어 분석

4 조

2022. 5. 13

트위터 '비건' 연관어 분석

'비건' 하면 함께 언급되는 단어들을 가지고, 단어간 관계를 나타나는 규칙을 찾아보겠습니다.

가장 관계가 높은 규칙을 가진 연관어들을 시각화하여, 비건 트렌드를 확인하고자 합니다.

---

1. 트위터 추출한 raw data 전처리 (1)

```
#트위터 추출한 raw data 전처리 (줄단위 데이터 구조 유지)
#트위터에서 추출한 raw data
load("C:/Rstudy/miniprj/textmining/vegan_raw.RData")

head(vegan,2)
```

```
## [[1]]
## [1] "cute_aram: RT @donghaemul_kr: 오늘 저녁엔 참치 없는 '비건 참치' 먹고, 탈육식 함께해요!\n\n#세계참치의날 #참치 #참치마요 #비건참치 #식물성참치 #대체육 #탈육식 #비건세상만들기 #동물해방물결 #느끼는모두에게자유를 https://t.co/r…"
##
## [[2]]
## [1] "poorhungryant: RT @MX_rang: 나 이 mbb 알아 비건하고 불독티 좋아하고 해장을 런닝으로 하는 몽베베자나 https://t.co/zqYr4kVOVV"
```

```
#getText() : 텍스트추출, sapply() : 함수 적용하여 벡터로 변환
vegan_word <- sapply(vegan, function(t) t$getText())

library(stringr)
#str_replace_all() : 해당 하는 문자를 치환(특수문자를 공백으로 바꿈)
vegan_word <- str_replace_all(vegan_word, "\\W", " ")
```

```
#불용어 제거를 위해 vegan_gsub.txt 파일에 있는 텍스트 읽어들임
txt <- readLines("vegan_gsub.txt", encoding="UTF-8")
cnt_txt <- length(txt) #텍스트 라인 개수 확인
#i <- 1
for(i in 1:cnt_txt){
  vegan_word <- gsub((txt[i]), "", vegan_word)
}
# cnt_txt 개수만큼 반복문 실행.
# gsub("바꾸고자하는 대상의 text 또는 패턴","대체할 text",text 객체)
# vegan_word 에서 txt 를 ""으로 치환하여, 다시 vegan_word 에 할당.

head(vegan_word,2) #불용어 제거된 줄 단위 텍스트.
```

```
## [1] "RT  donghaemul_kr  저녁엔 참치 없는  비건 참치  먹고  탈육식 함께해요
세계참치의날  참치  참치마요  비건참치  식물성참치  대체육  탈육식  비건세상만들기
동물해방물결  느끼는모두에게자유 https  t co r "
## [2] "RT  MX_rang  나 이 mbb 알아 비건하고 불독티 좋아하고 을  하는 나 https
t co zqYr4kVOVV"
```

---

2. 줄단위 단어 전처리 (2)

```
library(KoNLP)

# 줄 단위 단어 추출
lword <- Map(extractNoun, vegan_word) #Map(extractNoun,변수) : 변수에서 명사단
위로 추출
length(lword) #Length() 함수 : 데이터 개수 확인 / 출력값 : [1] 1000
```

```
## [1] 1000
```

```
lword <- unique(lword) #unique() 함수 : 빈 block 필터링
length(lword) # 중복제거 후 출력값 : [1] 441
```

```
## [1] 441
```

```
#str(lword) #List of 441
head(lword, 1)
```

```
## [[1]]
##  [1] "RT"                  "donghaemul"          "kr"
##  [4] "저녁"                "참치"                "참치"
##  [7] "육식"                "세계참치의날"        "참치"
## [10] "참치"                "비건"                "참치"
## [13] "식물성"              "참치"                "대체"
## [16] "육"                  "육식"                "비건"
## [19] "세상"                "만들기"              "동물"
## [22] "해방"                "물결"                "느끼는모두에게자유"
## [25] "https"               "t"                   "co"
## [28] "r"
```

```
# 단어 필터링 함수 정의
# : 길이가 2 개 이상 4 개 이하 사이의 문자 길이로 구성된 단어
#   is.hangul() 함수 : 영어 단어 필터링
filter1 <- function(x){
  nchar(x) >= 2 && nchar(x) <=4 && is.hangul(x)
}
filter2 <- function(x){
  Filter(filter1,x)
}

# 줄 단위로 추출된 단어 전처리
lword <- sapply(lword, filter2) # 단어 길이 1 이하 또는 5 이상인 단어 제거
head(lword,1) # 단어길이 2~4 사이의 단어 출력 (1 이하 또는 5 이상 제거, 한글외 언어 제거)
```

```
## [[1]]
##  [1] "저녁"   "참치"   "참치"   "육식"   "참치"   "참치"   "비건"   "참치"
##  [9] "식물성" "참치"   "대체"   "육식"   "비건"   "세상"   "만들기" "동물"
## [17] "해방"   "물결"
```

---

3. 연관분석

- 연관분석을 위해서는 추출된 단어를 대상으로 **트랜잭션 형식**으로 자료구조 변환이 필요함.

- **arules::apriori()** 함수를 사용해 연관규칙을 찾음. 파라미터 값으로 적정한 지지도와 신뢰도를 입력함. -> 20~40 개 사이의 연관 규칙이 나오도록 파라미터 조정함)
- **support** : 지지도. 좋은 규칙(빈도가 많은, 구성비가 높은)을 찾거나 불필요한 연산 줄일 때 기준 사용.
- **conf** : 신뢰도. 신뢰도가 높을 수록 유용한 규칙일 가능성이 높다고 할수있음.
- **lift** : 향상도. 향상도가 1 보다 크거나 작으면 우연적 기회보다 우수함 의미. (list=1 이면 서로 독립관계)

```
library(arules)  #arules 연관분석 패키지 로드

#트랜잭션 생성
wordtran <- as(lword, "transactions") #as(data, "transactions") : 트랜잭션으로 변환
wordtran  # 출력값 : 441 transactions (rows) and 1943 items (columns)

## transactions in sparse format with
##  441 transactions (rows) and
##  1943 items (columns)

#교차표 작성
wordtable <- crossTable(wordtran) #crossTable() : 교차테이블 생성
#head(wordtable) #유사단어들이 함께 있는 형태로 출력

#단어간 연관 규칙 산출
transrules <- apriori(wordtran,
                      parameter = list(support=0.015, conf=0.08))

## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##        0.08    0.1    1 none FALSE            TRUE       5   0.015      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
```

```
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 6
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[1943 item(s), 441 transaction(s)] done [0.00s].
## sorting and recoding items ... [40 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 done [0.00s].
## writing ... [27 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

 # 출력값 : writing ... [27 rule(s)] done [0.00s]. (27 개 규칙을 찾음)

#apriori() : 연관분석 기능 적용하여 규칙성을 찾는 함수. (transaction 자료 구조에서 실행)

#연관규칙 생성 결과 보기

```
head(inspect(transrules))
```

```
##      lhs                rhs        support    confidence coverage   lift
## [1]  {}              => {논비건}   0.08390023 0.08390023 1.00000000
1.0000000
## [2]  {}              => {비건}     0.44444444 0.44444444 1.00000000
1.0000000
## [3]  {지향}          => {비건}     0.01587302 0.77777778 0.02040816
1.7500000
## [4]  {출장}          => {카페}     0.02267574 1.00000000 0.02267574
13.7812500
## [5]  {카페}          => {출장}     0.02267574 0.31250000 0.07256236
13.7812500
## [6]  {출장}          => {비건}     0.02267574 1.00000000 0.02267574
2.2500000
## [7]  {친구}          => {비건}     0.01587302 0.63636364 0.02494331
1.4318182
## [8]  {메뉴}          => {비건}     0.01814059 0.80000000 0.02267574
1.8000000
## [9]  {크림}          => {비건}     0.01587302 0.43750000 0.03628118
0.9843750
## [10] {식당}          => {비건}     0.02494331 0.57894737 0.04308390
1.3026316
## [11] {음식}          => {비건}     0.02721088 0.66666667 0.04081633
```

```
1.5000000
## [12] {일기}           => {나의}     0.05215420 1.00000000 0.05215420
17.6400000
## [13] {나의}           => {일기}     0.05215420 0.92000000 0.05668934
17.6400000
## [14] {일기}           => {비거니즘} 0.05215420 1.00000000 0.05215420
16.9615385
## [15] {비거니즘}       => {일기}     0.05215420 0.88461538 0.05895692
16.9615385
## [16] {논비건}         => {비건}     0.02721088 0.32432432 0.08390023
0.7297297
## [17] {나의}           => {비거니즘} 0.05668934 1.00000000 0.05668934
16.9615385
## [18] {비거니즘}       => {나의}     0.05668934 0.96153846 0.05895692
16.9615385
## [19] {비거니즘}       => {비건}     0.01587302 0.26923077 0.05895692
0.6057692
## [20] {카페}           => {비건}     0.06122449 0.84375000 0.07256236
1.8984375
## [21] {비건}           => {카페}     0.06122449 0.13775510 0.44444444
1.8984375
## [22] {출장, 카페}     => {비건}     0.02267574 1.00000000 0.02267574
2.2500000
## [23] {비건, 출장}     => {카페}     0.02267574 1.00000000 0.02267574
13.7812500
## [24] {비건, 카페}     => {출장}     0.02267574 0.37037037 0.06122449
16.3333333
## [25] {나의, 일기}     => {비거니즘} 0.05215420 1.00000000 0.05215420
16.9615385
## [26] {비거니즘, 일기} => {나의}     0.05215420 1.00000000 0.05215420
17.6400000
## [27] {나의, 비거니즘} => {일기}     0.05215420 0.92000000 0.05668934
17.6400000
##      count
## [1]   37
## [2]  196
## [3]    7
## [4]   10
## [5]   10
## [6]   10
## [7]    7
```

```
## [8]    8
## [9]    7
## [10]  11
## [11]  12
## [12]  23
## [13]  23
## [14]  23
## [15]  23
## [16]  12
## [17]  25
## [18]  25
## [19]   7
## [20]  27
## [21]  27
## [22]  10
## [23]  10
## [24]  10
## [25]  23
## [26]  23
## [27]  23
```

```
##        lhs          rhs    support confidence   coverage     lift count
## [1]     {} => {논비건} 0.08390023 0.08390023 1.00000000  1.00000    37
## [2]     {} =>   {비건} 0.44444444 0.44444444 1.00000000  1.00000   196
## [3] {지향} =>   {비건} 0.01587302 0.77777778 0.02040816  1.75000     7
## [4] {출장} =>   {카페} 0.02267574 1.00000000 0.02267574 13.78125    10
## [5] {카페} =>   {출장} 0.02267574 0.31250000 0.07256236 13.78125    10
## [6] {출장} =>   {비건} 0.02267574 1.00000000 0.02267574  2.25000    10
```

---

4. 연관어 시각화

생성한 연관규칙 27 개를 그래프로 시각화하였으며, **거리가 가까울수록 높은 연관성**을 가지고 있음을 보여준다.

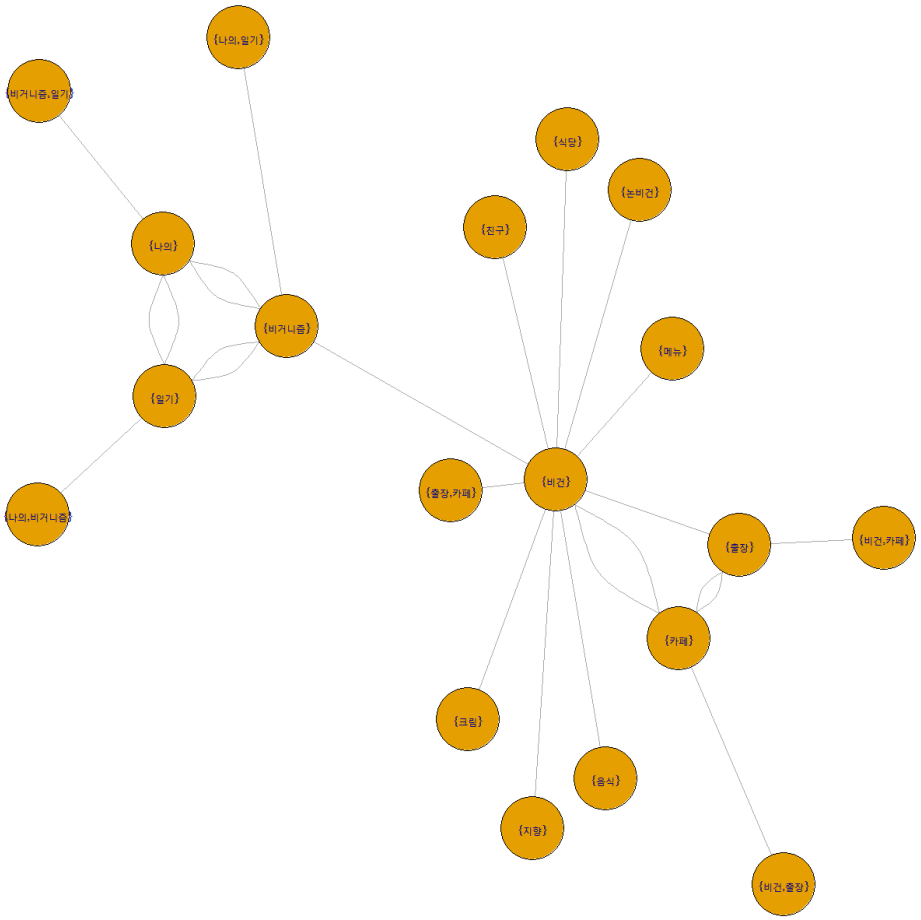
```
#연관 단어 시각화 위해 자료 구조 변경
rules <- labels(transrules, ruleSep = " ") #연관규칙 레이블을 " "으로 분리리
head(rules,4)
```

```
## [1] "{} {논비건}"   "{} {비건}"     "{지향} {비건}" "{출장} {카페}"
```

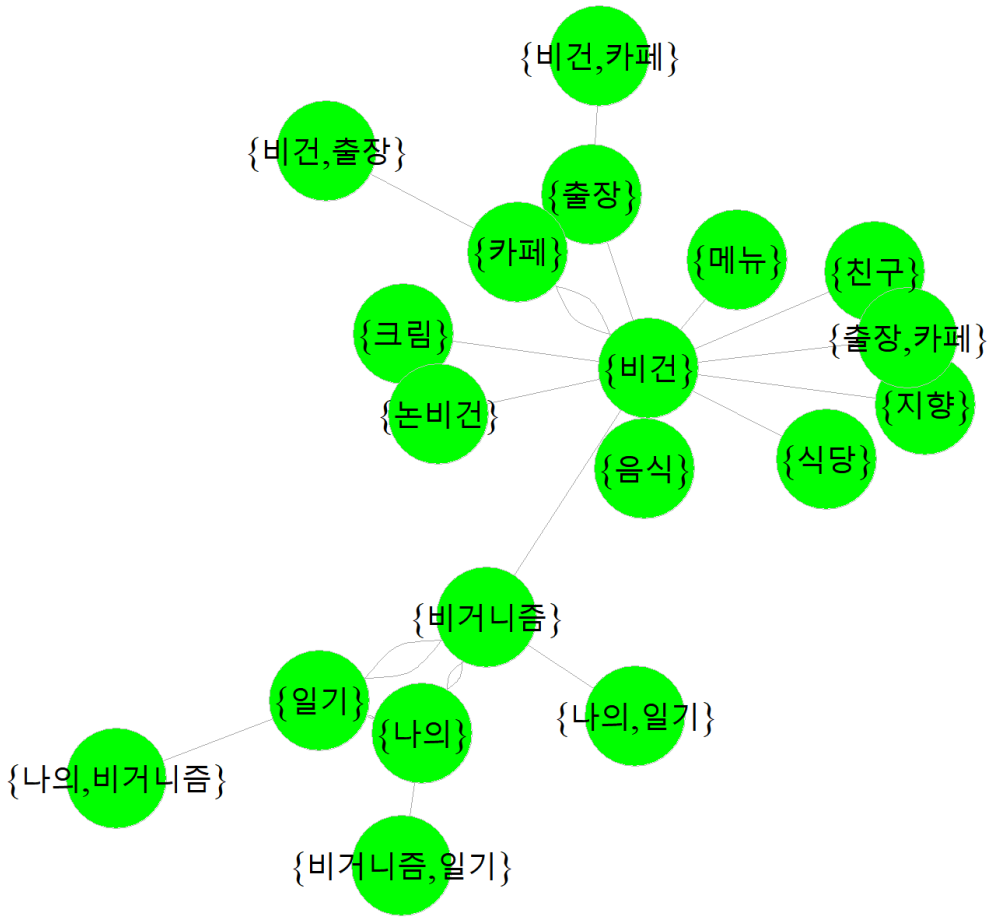
```r
#문자열로 묶인 연관 단어를 행렬 구조 변경
rules <- sapply(rules, strsplit, " ", USE.NAMES = F)
# class(rules) #출력값 : [1] "list"

# 행 단위로 묶어서 matrix 로 반환 (do.call)
rulemat <- do.call("rbind", rules)
# class(rulemat) #출력값 : [1] "matrix" "array"

# 연관어 시각화를 위한 igraph 패키지 설치
library(igraph)

# edgelist 보기 - 연관 단어를 정점(vertex) 형태의 목록 제공(matrix 형태의 자료형을 전달 받게 되어 있음)

relueg <- graph.edgelist(rulemat[c(3:27),], directed = F)  #[c(1:2)] - "{}"제외
relueg
```

```
## IGRAPH d7227eb UN-- 19 25 --
## + attr: name (v/c)
## + edges from d7227eb (vertex names):
##  [1] {지향}    --{비건}          {출장}    --{카페}
##  [3] {출장}    --{카페}          {비건}    --{출장}
##  [5] {비건}    --{친구}          {비건}    --{메뉴}
##  [7] {비건}    --{크림}          {비건}    --{식당}
##  [9] {비건}    --{음식}          {일기}    --{나의}
## [11] {일기}    --{나의}          {일기}    --{비거니즘}
## [13] {일기}    --{비거니즘}      {비건}    --{논비건}
## [15] {나의}    --{비거니즘}      {나의}    --{비거니즘}
## + ... omitted several edges
```

```r
# edgelist 시각화
plot.igraph(relueg)
```




```
# edgelist 시각화 2.
plot.igraph(relueg, vertex.label=V(relueg)$name, vertex.label.cex=3, vertex.l
abel.color='black',
            vertex.size=25, vertex.color='green', vertex.frame.color='gray')
```


{비건,카페}
{비건,출장}
{출장}
{카페}
{메뉴}
{친구}
{크림}
{출장,카페}
{비건}
{논비건}
{지향}
{음식}
{식당}
{비거니즘}
{일기}
{나의}
{나의,일기}
{나의,비거니즘}
{비거니즘,일기}