

[R] R로 하는 비정형 데이터 처리 (facebook 데이터를 통한 긍정/부정 나누기)

노트북: [TIL-MY]

만든 날짜: 2020-08-05 오전 8:16

URL: <https://continuous-development.tistory.com/51?category=793392>

나무늘보의 개발 블로그

홈

태그

방명록

R

[R] R로 하는 비정형 데이터 처리 (facebook 데이터를 통한 긍정/부정 나누기)

· by 꾸까꾸 · 2020. 8. 3. · 수정 · 삭제

비정형 데이터 처리

일단 기본적인 패키지들을 install 하자

분류 전체보기 

Python

Database

ASP.NET

Algorithm

Deep learning

```

304
305 # 비정형 데이터 처리(텍스트 마이닝)
306 # 단어 빈도를 나타내는 시각화(wordcloud, koNLP, tm)
307
308
309 install.packages(c("hash", "tau", "Sejong",
310                   "RSQLite", "devtools", "bit",
311                   "rex", "lazyeval", "htmlwidgets",
312                   "crosstalk", "promises", "later",
313                   "sessioninfo", "xopen", "bit64",
314                   "blob", "DBI", "memoise", "plogr",
315                   "covr", "DT", "rcmdcheck", "rversions"),
316                   type = "binary")
317
318 # github 버전 설치
319 install.packages("remotes")
320 # 64bit에서만 동작합니다.
321 remotes::install_github('haven-jeon/KoNLP',
322                         upgrade = "never",
323                         INSTALL_opts=c("--no-multiarch"))
324
325 |
326
327 #감성분석
328 # service_data_facebook_bigdata.txt
329
330 fbook <- file(file.choose(),encoding="UTF-8")
331 fbook_read <- readLines(fbook)
332 head(fbook_read)
333 str(fbook_read)
334

```

여기서는 페이스북의 데이터를 가져왔다. 데이터의 내용은 아래와 같다.

이런 식으로 각 행에 대해서 문장형 데이터가 들어가 있었다.

```

> fbook_read <- readLines(fbook) # 원 문장형 데이터 처리
> head(fbook_read)
[1] "스마트폰 기기와 3D 프린터 등의 하드웨어는 데이터의 출원 디디고 빠르게 발전한다. 다음 그림은 2013년에 미국에서 60초 동안 얼마나 많은 말이 말해지는지를 나타낸 그림이다. Facebook에서는 1.8억의 글이 4만 원의 글로 구성되어 있다. 글의 길이는 100만 글자 정도이다. 데이터는 3560만 개이다. 이런 데이터를 실시간으로 분석하면 사용자들의 행동을 파악하거나 의사결정을 할 수 있는 등 다양한 용도로 사용될 수 있을 것입니다."
[2] "빅데이터를 처리하는 프레임워크로 흔히 Hadoop MapReduce를 사용한다. MapReduce는 페타바이트 이상의 데이터를 여러 노드로 구성된 클러스터 환경에서 병렬 처리하는 기법으로, 분산형 프로그래밍에서 병렬적으로 사용되는 Map과 Reduce 방식을 사용한다. MapReduce는 대용량 데이터를 분산 처리할 수 있는 좋은 방법이지만, 배치 방식으로 데이터를 처리하기 때문에 실시간으로 데이터를 처리하기 어렵다. 이런 단점을 극복하기 위해 최근 몇 년간 실시간 분석을 위한 스트리밍 처리 기법이 많이 연구되었다."
[3] "실시간 분석 처리는 클라우드를 구성하는 노드가 각자 처리를 처리하게 (push down) 한 후에 처리할 데이터의 크기는 각기 다르지만 이를 병렬 처리해 중단 시간을 최소화하는 방식이다. Dremel의 논문은 기본적으로 Cloudera의 Impala와 Apache Tez, 그리고 최근 공개된 Facebook의 Presto가 이 방식에 속한다."

```

이 문장들을 전처리할 필요가 있었다. 그래서 정규표현식을 통해 전처리를 하였다.

AWS

ETC..

R 

공지사항

글 보실 때 주의사항

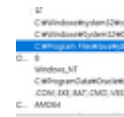
: 최근글 : 인기글

[R] R로 ...



2020.08.03

[R] R에...



2020.08.03

[R] R에...



2020.08.03

[Algorithm] 파이썬을 파..

2020.07.31

[R] R을 ...



2020.07.30

최근댓글

태그

cbind, SQL,

DDL, 날짜함수,

사용법,

```

337 #2. 전처리(정규표현식이 필요하다)
338 #문장 부호 제거[[:punct:]]하는 정규표현식 활용
339 #특수문자 제거[[:cntrl:]]
340 #숫자 제거 [[0-9]] \d+(숫자) , \w(단어) , \s+ (공백) , \n , \t
341 #gsub() 함수를 이용해서 전처리를 한다.
342
343 s1 <- gsub('[:punct:]', '', fbook_read) #문장부호 제거
344 s1
345
346 s2 <- gsub('[:cntrl:]', '', s1) # 특수문자 제거
347 s2
348
349 s3 <- gsub('\d+', '', s2) # 숫자제거
350 s3
351
352 s4 <- tolower(s3) #소문자로 변환
353 s4[1]
354

```

gsub을 통해서 정규표현식에 해당하는 데이터를 ""로 변환해주었다.

처음에는 문장부호를 제거하고 그다음에는 특수문자 , 숫자제거 이렇게 하였고 마지막에는 모든 대문자를 소문자로 바꿔주는 tolower를 사용하였다.

```

355
356 wordList <- str_split(s4, "\s+") #공백으로 분류
357 wordVec<-unlist(wordList) # vector로 만든다.
358

```

그다음은 str_split를 사용하여 공백을 통한 단어 분리를 하였다. 그다음 데이터 프레임 형식으로 된 데이터를

```

[[74]]
[1] "빅데이터란" "일차적으로" "데이터의" "영어" "방대함" "종래의" "방법으로는" "수집"
[2] "어려움" "것을" "결한다" "이차적으로도" "그럼" "큰" "데이터를" "어려"
[3] "정보도" "만들어내는" "과정까지" "포함한다" "단" "사실기부터" "우리나라에서도" "무관용영어"
[4] "데이터를" "데이터로서" "데이터란" "영어" "데이터를" "사실했다" "실용경제인구조사" "단"
[5] "빅데이터란" "지속하여" "어떻게" "미래의" "성장" "동력이" "될" "거라고"
[6] "--

[[75]]
[1] "전체" "사실상" "빅데이터는" "오래전부터" "유리" "세계" "이미" "동영상" "있는" "매우"
[2] "종류" "자료는" "결한다" "포함" "빅데이터" "시대" "매우능률" "위험" "통계적" "차이"
[3] "데이터를" "영어" "가장" "올바르고" "빠른" "답을" "알려주는" "실용적인" "학문"
[4] "근거" "외국어" "현대" "비즈니스" "지니" "말" "최상의" "무기" "것이다"

[[76]]
[1] "어" "정보" "일차적으로" "통계" "관련" "사적으로" "실용" "개별" "문제"
[2] "어려움" "데이터를" "데이터를" "데이터" "데이터" "데이터" "데이터" "데이터" "데이터"
[3] "데이터" "데이터" "데이터" "데이터" "데이터" "데이터" "데이터" "데이터"
[4] "데이터" "데이터" "데이터" "데이터" "데이터" "데이터" "데이터" "데이터"
[5] "데이터" "데이터" "데이터" "데이터" "데이터" "데이터" "데이터" "데이터"

```

unlist를 사용하여 vector 형식으로 바꿔주었다.

시계열 그래프,
R에서 DB 사용,
R DB 사용,
R 비정형 데이터 처리,
R과 DB,
전처리 예제,
Oracle, rbind,
테이블 생성,
설정,
파이썬을 파이썬
답게,
AWS,
R 데이터 전처리 예제,
R 데이터베이스,
데이터 전처리 예제,
R로 하는 비정형 데이터 처리,
Oracle SQL,
비정형 데이터 처리,
파이썬, R DB,
substr, 인스턴스,
ggplot, Python,
행렬

전체 방문자

129

Today : 1

Yesterday : 1

이 다음에는 긍정 단어와 부정 단어의 데이터를 통해 내가 가지고 있는 데이터와 매칭 하는 작업을 하였다.

```

447 library(stringr)
448 library(ggply)
449 library(tidy)
450
451 #여기서 생략된게 1. list를 for문에서 계속 돌리면 2-sum(true) 1의 값을 나타내고 3-lsply에서 function을 돌렸을때 매개변수로 앞에서 관련 키워드를 전달
452
453 # 이 함수를 정의해줘요
454 results <- function(words, positive, negative) { #여기서 함수에 리스트로 들어가지게 해줌을 for 문에서 계속 돌림.
455
456   scores = lsply(words, function(words, positive, negative) {
457     #match match(words, positive)
458     #match match(words, negative)
459
460     #match = if na(match) # true false 1을 # true 나타냄. 그래서 if true로 sum으로 되면 1을 나타내고 있겠거 생각해서 나타냄.
461     #match = if na(match)
462
463     score = sum(match) - sum(match) # 같이 앞부분의 긍정의 단어로 -일것을 보정의 단어로 # 일것을 줄임단어에서 나타냄.
464     #positive, negative) # 여기 두개는 문법이다. 원래 function(words, positive, negative) 해 준거는 앞에서 이원식이고 결과이다. 결과값으로 return하는거 score
465
466   })
467
468 scores.df = data.frame(score=scores, text=words)
469 return(scores.df)
470 }
471
472
473 result1bl <- results(wordvec, pblc, nblc)
474 head(result1bl)

```

```

477 resultTbl <- resultS(wordVec, pDic, nDic)
478 str(resultTbl)
479 head(resultTbl)
480
481 resultTbl$text
482 resultTbl$score
483 resultTbl$remark[resultTbl$score >= 1] <- "긍정"
484 resultTbl$remark[resultTbl$score == 0] <- "중립"
485
481:1 (Top Level)

```

Console

Terminal x

Jobs x

~/

489 0

490 0

491 0

492 0

493 0

494 0

495 0

496 0

497 0

498 0

499 0

500 0

중분합니다

스프레드시트

프로그램은

셀에

데이터를

입력하는

데

엑셀

에서는

약

만개행

x

[reached 'max' / getOption("max.print") -- omitted 1988 rows

head(resultTbl)

score

text

0

스마트

0

기기와

0

sns

0

덕분에

0

과거

0

어느

이런식으로 각 단어에 대한 긍정 부정을 볼 수 있다.

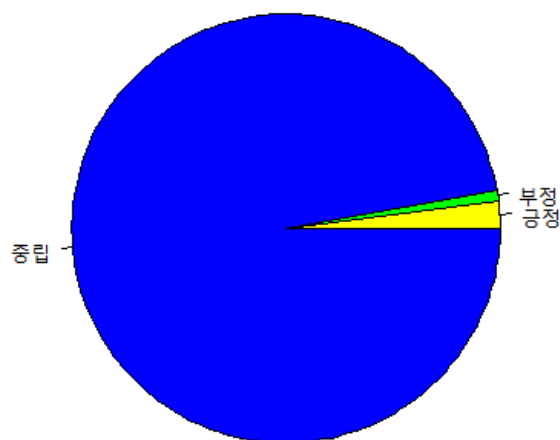
```
472
473 # 긍정부정에 따라서 파이차트 만들기
474 resultTbl <- resultS(wordVec, pDic, nDic)
475 str(resultTbl)
476 head(resultTbl)
477
478 resultTbl$text
479 resultTbl$score
480 resultTbl$remark[resultTbl$score >= 1] <- "긍정"
481 resultTbl$remark[resultTbl$score == 0] <- "중립"
482 resultTbl$remark[resultTbl$score < 0] <- "부정"
483
484 resultTbl$remark
485
486 table(resultTbl$remark)
487
488 pieResult <- table(resultTbl$remark)
489 pieResult <- table(resultTbl$remark)
490
491 ?pie
492 pie(pieResult,
493     labels=names(pieResult),
494     col = c('yellow','green','blue'))
```

```
> table(resultTbl$remark)
```

긍정	부정	중립
51	20	2417

이렇게 긍정 부정중립의 개수를 셀수 있다.

이걸 파이차트로 나타내면 아래와 같다.





'R' 카테고리의 다른 글

- [R] R로 하는 비정형 데이터 처리 (facebook 데이터를 통한 긍정/부정 나누기) (0) 2020.08.03
- [R] R에서 Database 사용하기 / DB 기본적인 구문 사용하기 (0) 2020.08.03
- [R] 예제를 통한 데이터 전처리 작업 (0) 2020.08.03
- [R] R을 통해 시계열 그래프 만들기 (자료 분석을 위한 시각화와 실습 예제) (0) 2020.07.30
- [R] 같은 형태의 ggplot 과 barplot 만들기 (차이 비교) (0) 2020.07.30
- [R] ggplot2 패키지 설치 에러시 해결 방법 (0) 2020.07.30

태그

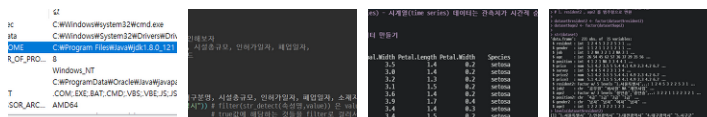
R 비정형 데이터 처리

R로 하는 비정형 데이터 처리

비정형 데이터

비정형 데이터 처리

관련글



[R] R에서 ... [R] 예제를 ... [R] R을 통... [R] 같은 형...

댓글 0



1

2

3

4

5

...

18



TEL. 02.1234.5678 / 경기 성남시 분당구 판교역로

© Kakao Corp.

