Testu-meatzaritza R-n

Josemari Sarasola

Gizapedia

Zer den testu-meatzaritza

Testu-meatzaritza testu-multzo bateko azterketa da, teknika estatistikoak nahiz adimen artifiziala baliatuz, testuen azterketa kuantitatiboa (hitzen maiztasuna, adibidez) nahiz kualitatiboa (adibidez, testuak adierazten dituen ideiak eta sentimenduak, atzertuz). Zientzia politikoetan maiz erabiltzen da, politikarien diskurtsoa aztertzeko, eta marketinean ere bai, kontsumitzaileen iritziak aztertzeko-

Zer egin behar dugun

Testu bat hartu, hitzetan zatitu eta hitz horien maiztasunak kalkulatzeko prozesua garatu behar dugu. R softwareak pakete asko ditu testu-meatzaritza egiteko. Ondoko gardenkietan ordea, dena batera egin ordez, pausoz pauso joango gara, agindu arruntak baliatuz gehienetan, prozesua ondo ulertzeko.

Erabiliko dugun oinarriko testua

Egungo gazteak alper hutsak dira. Beti alperkerian. Zer egin behar dugu gazte horiekin? Egurra eman, onbideratu arte. Ikasi eta lan egin ordez, jolasa eta festa baino ez dute buruan. Gu ez ginen horrelakoak gaztaroan. Festetan ibiltzen ginen noski, baino beharrak agintzen zuenean, hortxe ginen beti, lanean. Ni 14 urterekin hasi nintzen lanean. Egungo gazteekin berriz, atarramendu onik ez. Soluzioa garbi ikusten dut, bereziki unibertsitatean dabiltzan horiekin: ikasi nahi ez, lanera orduan.

Testua gorde eta formateatu

Aurreko testua landuko dugu. Horretarako, kopia-pega egin, word-era eramanez, eta han `.txt` formatoan gorde (nik `solasaldia` izena aukeratu dut).
Orain, R instalatuta, testu hori nora eraman behar duzun zehaztu behar duzu. Horretarako, une horretan R-ko lan-direktorioa (work directory) zein den jakin behar duzu, R-ko agindu honen bitartez:

```
>getwd()
```

Testuaren `.txt` fitxategia aurreko aginduak erakusten duen karpetan gorde behar duzu. Beste irtenbide bat zure lan-direktorio propioa sortzea da, eta han gordetzea testua. Nik horrela egin nuen, direktorioaren *path* edo bidea adieraziz:

```
>setwd("C:/Users/PDI/Documents/RJosemari")
```

Testua R ekarri

Testua R-ra ekarri eta memorian izen batekin (nire kasuan, `testua` aukeratu dut) gordetzeko agindua aski sinplea da:

```
>testua=readLines("solasaldia.txt")
```

Orain, `testua` izena inbokatuz aterako zaizu testua:

```
>testua
```

Paketeak deskargatu

R programazio-lengoaia modularra da. Oinarrizko pakete bat du (base), baina horri pakete gehigarriak erantsi dakizkioke, funtzionalitate berriak eskuratzeko. Testu-meatzaritza egiteko, makina bat pakete daude R-n eskuragarri, baina guk pakete sinple bat baino ez dugu erabiliko: `tokenizers` izenekoa, testua hitzetan banatzeko erabil daitekeena. Paketea ez dago pakete basikoan, beraz deskargatu egin behar duzu. Honela egiten da:

```
>install.packages("tokenizers")
```

Gero paketea erabili behar duzun saio bakoitzean kargatu egin behar duzu:

```
>library("tokenizers")
```

Tokenizazioa

Tokenizazioa testu bat hitzetan edo bestelako unitatetan banatzea da. Horretarako noski, puntuazio-ikurrak, zenbakiak eta *stopword* edo garrantzirik gabeko hitzak ("ez","eta", gure kasuan adibidez) ezabatu behar dira. `tokenizers` paketetik agindu honekin egin dezakezu:

```
>hitzak=tokenize_words(testua,
stopwords=c("eta","ez","baino","egin","horiekin"),
strip_punc=T,strip_numeric=T)
>hitzak
```

Horrela, hitzak banaka agertuko zaizkizu, puntuaziorik, zenbakirik eta *stopwords* direlakoak ezabatuta.

Maiztasun-taula

Orain, testuko *token* edo hitzak kontatuko ditugu. Aurrena, hitzak datu moduan (hobe esanda, R-ko hizkeran, bektore moduan) hartu behar ditugu:

```
>datuak=unlist(hitzak)
```

Eta orain, datu hauek ordenatu egiten ditugu, antzekoak diren hitzak ikusteko:

```
>datuakord=sort(datuak)
>datuakord
```

Stemming

Aurreko taulan hitzak ordenaturik agertzen zaizkigu, eta zerrenda horretan "gazte","gazteak","gaztaroan" eta "gazteekin" hitzak ikus ditzakegu. Hitz horiek *stem* edo erro berdina dute eta beraz, kontatu baino lehen bateratu egin beharko genituzke, bestela behin bakarrik azalduko lirateke, 4 aldiz agertzen den erroa izan arren.
Horretarako, `stringr` paketea deskargatu eta aktibatu behar dugu, eta hitz horiek "gazteak" hitzarekin ordeztu:

```
>install.packages("stringr")
>library(stringr)
>datuakord1=str_replace_all(datuakord,
pattern="gazt.+",replacement="gazteak")
>datuakord1
```

Zerrenda berrian ikusiko dugu "gazteak" hitza 4 aldiz azaltzen dela.
`(.+)` ikurrek adierazten dute `gazt` erroa duten hitz horiek gutxienez letra bat dutela ondoren.

Stemming

Modu berean egiten dugu "alper" eta "alperkerian" hitzekin alde batetik, eta "lan", "lanean" eta "lanera" hitzekin bestetik, baina orain kontuan hartu behar dugu posible dela `alper` eta `lan` hitzek letra gehiagorik ez izatea ("alper" eta "lan" hitzekin, hain zuzen). Hori honela egiten dugu:

```
>datuakord2=str_replace_all(datuakord1,
pattern="alper.*",replacement="alperrak")
>datuakord3=str_replace_all(datuakord2,
pattern="lan.*",replacement="lana")
>datuakord3
```

`(.*)` eta `(.+)` ikurren efektu desberdina hobeto ikusteko aplikatu `(.*)` ikurrak "gazt" erroari eta ikusiko duzu zein den emaitza. Pentsatu zergatik geratzen den horrela.

Stemming

Badira hitzak erro berdina izan ez baino ideia berdina adierazten dutenak; adibidez, "ginen" eta "gu" alde batetik, "ikasi" eta "unibersitatea", "festa", "festetan" eta "jolasa" bestetik. Hiz horiek batera jartzeko honela egiten dugu:

```
>datuakord4=str_replace_all(datuakord3,
pattern="ginen|gu",replacement="gu")
>datuakord5=str_replace_all(datuakord4,
pattern="ikasi|unibertsitatean",replacement="ikasi")
>datuakord6=str_replace_all(datuakord5,
pattern="festa|festetan|jolasa",replacement="festak")
>datuakord6
```

Maiztasun-taula

Orain, testuko *token* edo hitzak kontatuko ditugu. Horretarako agindua oso sinplea da:

```
>table(datuakord6)
```

Eta orain, taula honi izen bat (*taula*, esaterako) emango diogu:

```
>taula=table(datuakord6)
```

Eta orain, taula honetako emaitzak datu moduan hartu R-n, taula manipulatu behar baitugu:

```
>maiztasunak=data.frame(taula)
>maiztasunak
```

Maiztasun-taula ordenatu

Orain maiztasun taula ordenatu behar dugu maiztasunen arabera, maiztasun handieneko hitzak hobeto bistaratzeko:

```
>maiztasunakord=maiztasunak[order(maiztasunak$Freq),]
>maiztasunakord
```

Maiztasun-taula ordenatu

Emaitzak ikusita, esan daiteke 2 edo maiztasun handiagoa duten hitzak direla esanguratsuak. Ezaba ditzagun taulatik beste guztiak, eta hitz berrien zerrendarako deia egin dezagun:

```
>azkenak=subset(maiztasunakord,Freq!=1)
>azkenak
datuakord6 Freq
gazteak 4
gu 4
lana 4
festak 3
alperrak 2
beti 2
egungo 2
ikasi 2
```

Maiztasun-taula ordenatu

Nahiko bazenu 2 aldiz errepikatzen diren hitzak ere kendu, agindua errepikatu baino ez duzu egin behar, aurreko datu multzoaren gainean betiere.

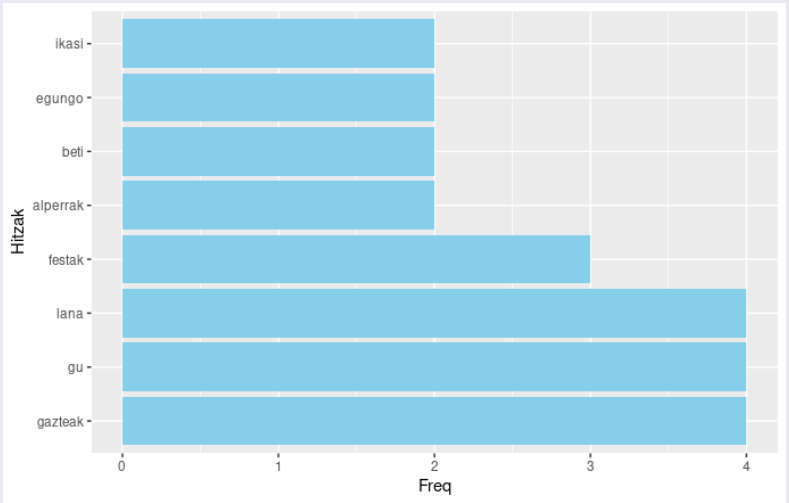
```
>azkenak2=subset(azkenak,Freq!=2)
>azkenak2
datuakord6 Freq
gazteak 4
gu 4
lana 4
festak 3
```

Barra-diagrama

Honela egiten dugu barra-diagrama maiztasun handieneko hitzei buruz:

```
>install.packages("ggplot2")
>library("ggplot")
>ggplot(azkenak,aes(x=reorder(datuakord6, -Freq),y=Freq))
+geom_bar(stat="identity",fill="skyblue")
+coord_flip()+xlab("Hitzak")
```

`reorder` barrak ordenatzeko da, maiztasun txikienetik handienera, eta `coord_flip` barrak horizontalak ezartzeko:



Interpretazioa

Emaitzak ikusita, garbi dago diskurtsoan dialektika bat ezartzen dela gazteen eta zaharren (gu) artean. Gazteak festazaleak eta alperrak direla diote zaharrek, beraiek langileak ziren bitartean. Dialektika garaiari buruzkoa ere bada (antzina eta "egun"), eta absolutua ("beti" hitzaren errepikapenarekin ikus daitekeenez)

Hitzen hodeia

Hitz-hodeia (egin klik) deitzen den grafikoak hitzen maiztasun-taula osotik errepikatuenak (esanguratsuenak) hartzen ditu zuzenean, maiztasun minimo bat hartu beharrik gabe.

Horretarako aurrez `wordcloud` eta kolorea emateko `RColorBrewer` paketeak instalatu eta kargatu behar dituzu:

```
>install.packages("wordcloud")
>library(wordcloud)
>install.packages("RColorBrewer")
>library(RColorBrewer)
```

Hitzen hodeia

Eta hitzen hodeia honela eratzen dugu:

```
>wordcloud(words=maiztasunak$datuakord6,
freq=maiztasunak$Freq,
colors=brewer.pal(8,"Spectral"))
```

Hitzen hodeia

Interpretazioa

Hitz hodeian berresten da belaunaldien arteko talka, zahar langileen eta gazte festazaleen artean.

Hitzen hodeia

Hitzen hodeia osatzeko ahal diogu maiztasun minimo bat eskatu, barra diagrama osatzeko egin genuen bezala:

```
>wordcloud(words=maiztasunak$datuakord6,
freq=maiztasunak$Freq,
colors=brewer.pal(8,"Spectral"),
min.freq=2)
```

Hitzen hodeia

Hitz guztiak hartzerakoan, agian eskala berregin beharko dugu, horrela egin ezean, agian hodeia moztuta azalduko baita:

```
>wordcloud(words=maiztasunak$datuakord6,
freq=maiztasunak$Freq,
colors=brewer.pal(8,"Spectral"),
min.freq=1,scale=c(3.5,0.15))
```

Bukatzeko ...

Testu-meatzaritza garatzeko ohiko pausoak ikasi ditugu. Egia esan, R-n badira paketeak dena batera egiten dutenak, hala nola Quanteda eta Tidyverse, baina arestian prozesuaren muina ikasi ahal izan dugu.