Banner2CPE

Fernando, Antonio, David

26 de abril de 2016

Abstract

El propósito de esta aplicación es encontrar un CPE relacionado con un Banner indicado.

Análisis de datos estructurado

Definir la pregunta

Dado un Banner, ¿qué CPE tiene asociado?

```
banner <- "Microsoft IIS 7.0"

CPE --> ???
```

Definir el conjunto de datos ideal

Un conjunto de datos ideal sería tener una matriz en la que estuvieran relacionados los nombres de los CPE's con las palabras claves del título del propio CPE, y por otra parte tener unos banners que coincidieran exactamente con los títulos de los CPE's.

Determinar los datos que tenemos accesibles

Los datos que tenemos accesibles son, por un lado, un XML que contiene información sobre los CPE's (nombre, referencias, títulos, etc), y por otro unos banners cuyo contenido no podemos asegurar que contengan toda (o parte) de la información que necesitamos para hacer un match exacto con los títulos de los XML que tenemos.

Obtener los datos

Fichero XML de CPE's de la National Vulnerability Database (http://static.nvd.nist.gov/feeds/xml/cpe/dictionary/official-cpe-dictionary_v2.3.xml). Los Banners los incorporamos a mano, aunque en un futuro se podrá usar otra fuente de datos que los introduzca de manera automática.

Limpiar los datos >> Datos elegantes

Del XML nos quedamos sólo con el name y el title de los CPE's. El Banner lo hacemos pasar por un proceso en el que se sustituyen caracteres especiales por espacios, se eliminan stopwords y se convierte todo en minúsculas. Además, eliminamos todas aquellas palabras que no salen en el conjunto de palabras formado portodos los titles de CPE's.

Análisis de exploración de datos

Hemos analizado los datos del fichero de CPEs desde tres puntos de vista:

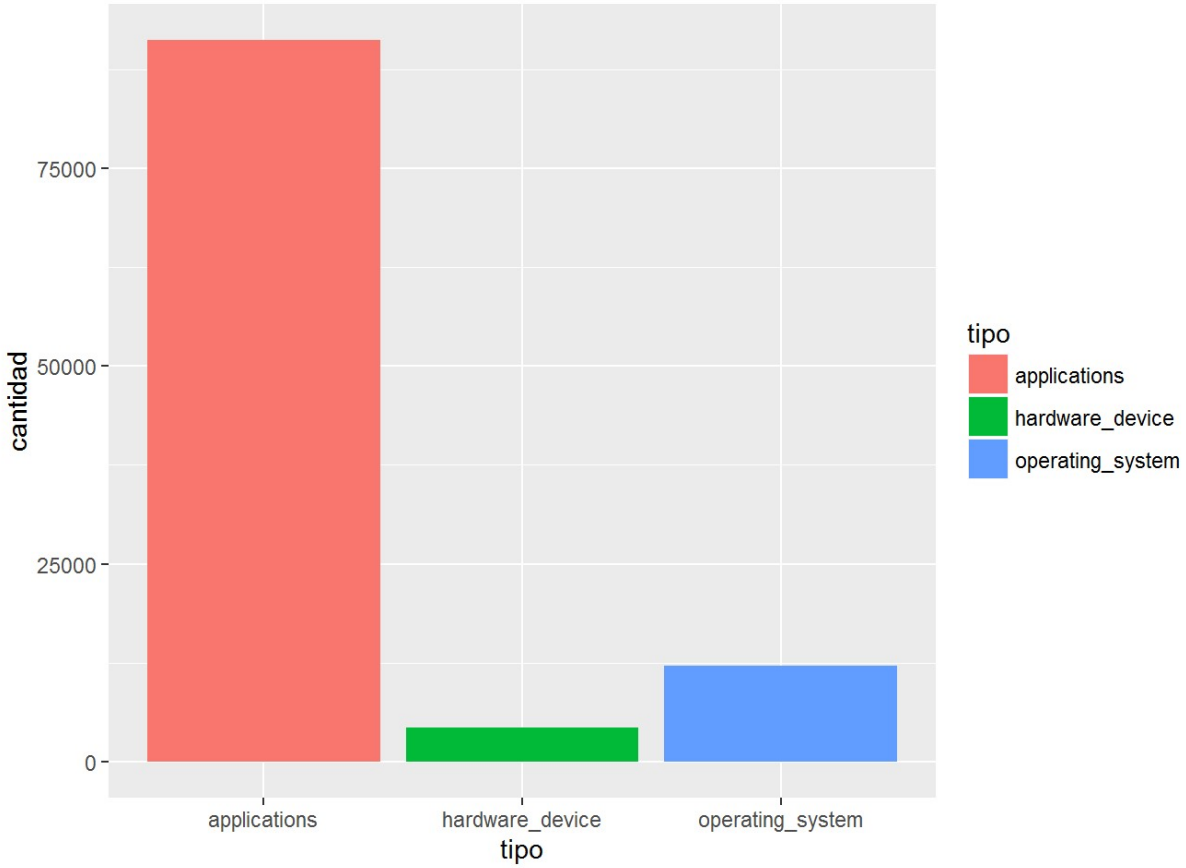
(Para todos los gráficos, sól mostramos los 10 más relevantes)

1. Tipo de activo: aplicación / Sistema Operativo / Hardware

```
library("ggplot2")
```

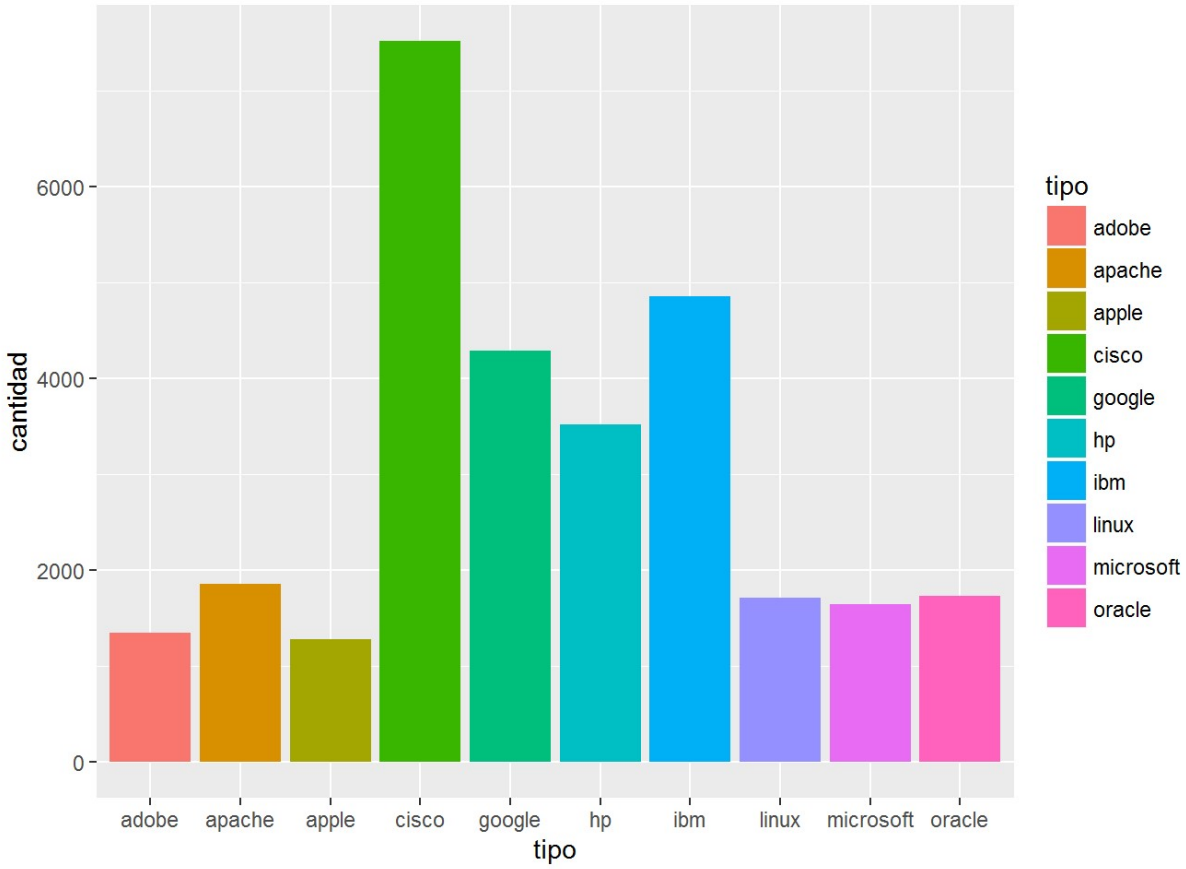
```
## Warning: package 'ggplot2' was built under R version 3.2.5
```

```
plot1 <- read.csv('inst/exdata/Part.txt', header= FALSE, sep=";")
colnames(plot1) <- c('cantidad','tipo')
ggplot(tail(plot1, n=10), aes(tipo,cantidad)) + geom_bar(aes(fill=tipo),posi
tion="dodge",stat="identity")
```



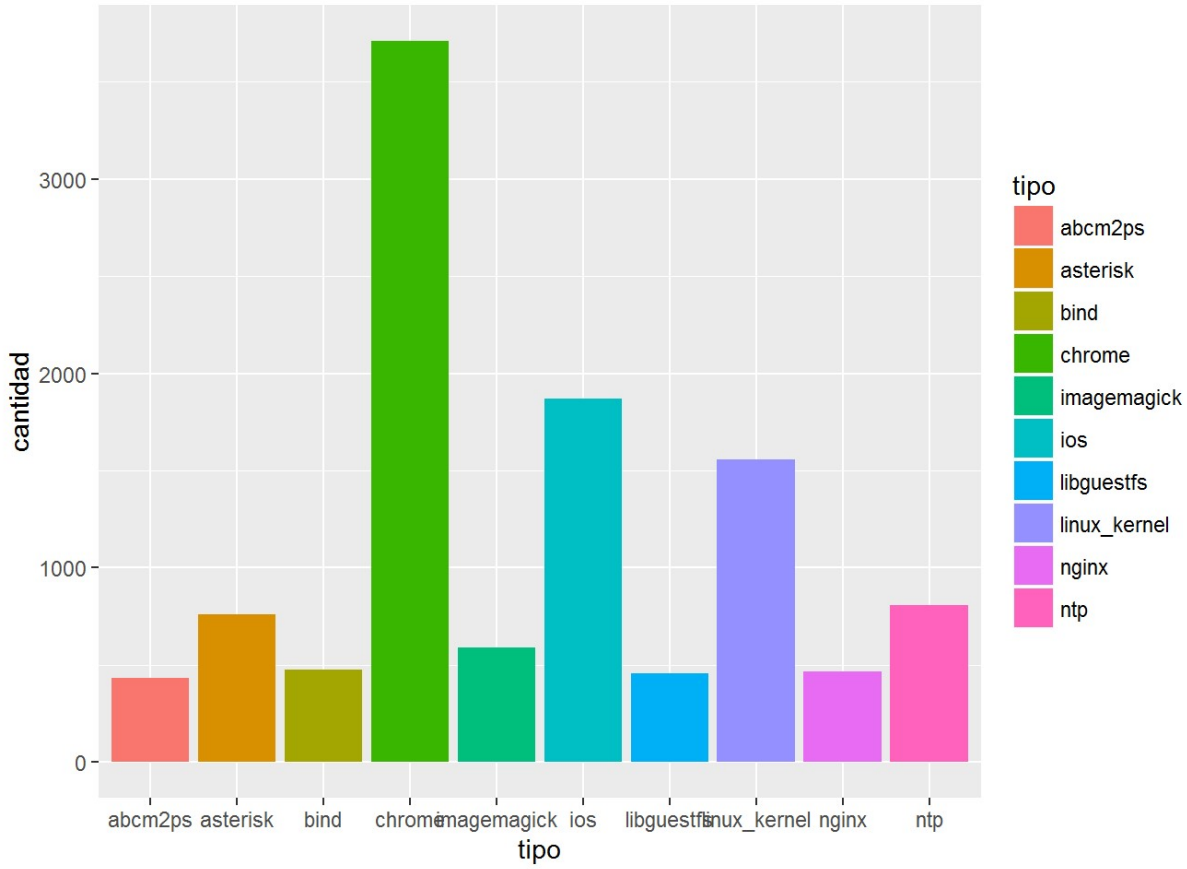
2. Fabricante

```
plot2 <- read.csv('inst/exdata/Vendor.txt', header= FALSE, sep=";")
colnames(plot2) <- c('cantidad','tipo')
ggplot(tail(plot2, n=10), aes(tipo,cantidad)) + geom_bar(aes(fill=tipo),posi
tion="dodge",stat="identity")
```



3. Producto

```
plot3 <- read.csv('inst/exdata/Product.txt', header= FALSE, sep=";")
colnames(plot3) <- c('cantidad','tipo')
ggplot(tail(plot3, n=10), aes(tipo,cantidad)) + geom_bar(aes(fill=tipo),posi
tion="dodge",stat="identity")
```



Interpretar los resultados

Los resultados dependen, como esperábamos, de la “calidad” del Banner. Aunque parezca una obviedad, no es posible encontrar con cierta fiabilidad un CPE si el Banner no cuenta con un mínimo de palabras que coincidan exactamente con el título del CPE. Si el contenido del Banner ha sido modificado y no se han mantenido palabras clave, no será posible encontrarle un CPE (y si se encuentra, su factor de similitud será bajo).

Obtener respuesta a la pregunta

Para obtener una respuesta a la pregunta que nos hemos formulado, debemos ejecutar la función findCPE, pasándole como argumento el conjunto de palabras que conforman el banner del que estamos buscando su CPE asociado.

```
banner <- "Microsoft IIS 7.5"
x <- banner2cpe::findCPE(banner)
```

```
## [1] "Microsoft IIS 7.5"
## {"microsoft", "iis", "7.5"}
## [1] "microsoft"
## [1] 1
## [1] "iis"
## [1] 0.6666667
## [1] "7.5"
## [1] 0.3333333
```

```
print(x)
```

```
##                                                  cpe     factor
## 9                          cpe:/a:microsoft:iis:7.5 1.00000000
## 2                          cpe:/a:microsoft:iis:1.0 0.83333333
## 3                          cpe:/a:microsoft:iis:2.0 0.83333333
## 4                          cpe:/a:microsoft:iis:3.0 0.83333333
## 5                          cpe:/a:microsoft:iis:4.0 0.83333333
## 6                          cpe:/a:microsoft:iis:5.0 0.83333333
## 7                         cpe:/a:microsoft:iis:5.06 0.83333333
## 8                          cpe:/a:microsoft:iis:5.1 0.83333333
## 10 cpe:/a:microsoft:internet_information_server:1.0 0.83333333
## 11 cpe:/a:microsoft:internet_information_server:2.0 0.83333333
## 1                  cpe:/a:microsoft:ftp_service:7.5 0.13333333
## 12               cpe:/a:microsoft:msn_messenger:7.5 0.11111111
## 13                         cpe:/a:drupal:drupal:7.5 0.08333333
## 14                 cpe:/a:searchblox:searchblox:7.5 0.08333333
```

Sintetizar y describir los resultados y el proceso

Requisitos Previos:

Preparar variables globales. Este proceso incluye:

Carga de XML de CPEs, carga de titles sin Stop Words y guardado de ambos en ficheros con formato .rda. Esta operación debe realizarse si no se dispone de los ficheros .rda o si se quiere actualizar la información que contiene. Esta operación tiene una duración aproximada de 15 minutos, pero a partir de entonces, la realización de búsquedas de CPEs se realiza en un tiempo muy reducido.

```
prepareGlobalVars <- function(){
  xmlFile <- 'inst/exdata/official-cpe-dictionary_v2.3.xml'
  matrix <- loadXML(xmlFile)
  dataFrame <- prepareDataframe(matrix)
  titlesWordList <- getFromFile('inst/exdata/titlesWithoutStopWords.txt')
  save(dataFrame, file = 'R/dataFrame.rda')
  save(titlesWordList, file = 'R/titlesWordList.rda')
}
```

Proceso:

1. Introducimos un banner, y lo normalizamos. Incluye quitar carácteres que no nos serán útiles (corchetes, dobles espacios, etc), lo pasamos todo a minúsculas y hacemos un ‘AND’ con la lista de palabras que tenemos en los títulos de los CPEs (aquellos que no se encuentren en la lista los eliminaremos, ya que sabemos que no harán match con nada de las palabras de los title, que es lo que nos interesa). Tras todo ello, nos quedamos con un conjunto de palabras que será la nueva representación de nuestro banner.

```
## {"microsoft", "iis", "7.5"}
```

2. Partiendo de la lista de títulos de los CPEs y el nuevo conjunto de palabras que es ahora nuestro Banner, creamos tantas tablas como palabras del banner aparecen en almenos 1 CPE, asignando para cada título de los CPEs un valor determinado obtenido a partir de dos factores: el primer factor es un valor que se obtiene en función del orden en el que se encuentra la palabra que analizamos en el Banner: entendemos que la primera palabra es la más importante, y la última la que menos. El segundo factor se basa en una función de similitud basado en el índice de Jaccard. Cada una de estas tablas las creamos con la función FindCPEWithWord.
3. Juntamos las anteriores tablas, mediante rbind, y sumamos los valores de aquellos registros que se repiten. Ese valor será el factor final de similitud que relacionará el banner inicial con un CPE determinado. Ordenamos la lista de CPE's por el factor final de similitud, de mayor a menor, obteniendo los 10 mejores, a partir de una puntuación mínima, y la mostramos el resultado al usuario.

Resultado:

El resultado es la lista de CPE's que obtenemos tras ejecutar la función findCPE. Podemos ver que está ordenada según el campo factor, de mayor a menor, que indica la afinidad que tiene el banner con el CPE indicado.