ESSAI

Omar THIAM

2023-05-26

Plan

Introduction

I-Definition

II-Notion d'url,de htlm et de css

III.Les Packages

IV.extraction de donnees

V.Application du web mining

VI.Respect des politiques de confidentialites

Conclusion

Introduction

Les données et les informations sur le Web connaissent une croissance exponentielle. Aujourd'hui, nous utilisons tous Google comme première source de connaissances, qu'il s'agisse de trouver des avis sur un endroit ou de comprendre un nouveau terme. Toutes ces informations sont déjà disponibles sur le web. la seule chose qui vous empêche de les utiliser est la possibilité d'y accéder La plupart des données disponibles sur le Web ne sont pas facilement disponibles. En effet,elles sont présentes sous un format non structuré (format HTML) et n'est pas téléchargeable. Par conséquent, il faut des connaissances et une expertise pour utiliser ces données pour éventuellement construire un modèle utile .Dans cet expose, nous allons voir ensemble le processus de grattage Web dans R.

I-Definition

- **C'est quoi "web mining avec R ?**

Le **web mining** (ou exploration de données sur le web) est une méthode *d'extraction" de données à partir de sites web et d'autres sources en ligne. Il s'agit d'une technique utilisée pour collecter des données, analyser des informations et découvrir des tendances à partir de pages web et de bases de données en ligne.

II-Notion d'url,de htlm et de css

- **URL**

URL signifie Uniform Resource Locator (ou, en français, « localisateur uniforme de ressource »), est simplement **l'adresse d'une ressource donnée**, unique sur le Web. Adresse d'un site ou d'une page hypertexte sur Internet (ex: http://www.lerobert.com).

- **C'est quoi un HTML en informatique ?**

Designant HyperText Markup Language, HTML est un langage de balisage utilisé pour la création de pages web, permettant notamment de définir des liens.

HTML peut etre defini comme le code utilisé pour structurer une page web et son contenu. Par exemple, le contenu de votre page pourra être structuré en un ensemble de paragraphes, en une liste avec des images et des tableaux de données.

- **Que devons nous comprendre de CSS ?**

Les CSS (Cascading Style Sheets en anglais), ou « feuilles de style en cascade » appeles selecteurs le plus souvent,sont le codes utilisés pour mettre en forme une page web. Par Exemple on utilise le selecteur :

a pour sélectionner tous les liens dans une page HTML

table pour afficher les tableaux d'une page

p pour des paragraphes d'un texte

III. Les Packages

Le scraping de données peut être effectué à l'aide d'outils et de packages dédiés, tels que Beautiful Soup, Scrapy, rvest, Selenium, etc. L'approche la plus courante consiste à utiliser des packages pour extraire les données à partir du code HTML. Par exemple, en utilisant le package R rvest, vous pouvez extraire les données d'une page web en quelques lignes de code. Il est le package qui permet d'appeler les fonctions servant l'extraction des donnees. Pour les analyses et visualisations, on est amene a utiliser,selon le besoin, les packages: **tidyverse**, **tidytext**, **dplyr**, **ggplot2**,etc.

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```
library(av)

library(rvest)
```

```
##
## Attachement du package : 'rvest'
##
## L'objet suivant est masqué depuis 'package:readr':
##
##     guess_encoding
```

```
library(purrr)
```

```
#install.packages("magick")
library(magick)
```

```
## Linking to ImageMagick 6.9.12.3
## Enabled features: cairo, freetype, fftw, ghostscript, heic, lcms, pango, raw, rsvg, webp
## Disabled features: fontconfig, x11
```

```
#install.packages("jpeg")
library(jpeg)
library(purrr)
```

IV.Extraction de donnees

L'art d'extraire des données depuis un site web a un nom : c'est le **web scraping**, aussi appelé **harvesting**. Cette technique permet de récupérer des informations d'un site, grâce à un programme et de les réutiliser ensuite.

Le Web scraping est une technique de conversion des données présentes dans un format non structuré (balises HTML) sur le Web vers un format structuré facilement accessible et manipulable.

De quelles donnees extraires ?

Les données les plus demandées sur le web sont sous formats:

- Texte

```
library(rvest)
# Lire la page web
#url<-"https://www.scrapingbee.com/blog/web-scraping-r/
page <- read_html("https://www.scrapingbee.com/blog/web-scraping-r/")
# Sélectionner des paragraphe
paragraph <- html_nodes(page, "p")
# Extraire le texte des paragraphe
text <- html_text(paragraph)
# affichage texte
#print(text)
# determination du type de la var text
typeof(text)
```

```
## [1] "character"
```

```
# convertir en list
text1<-as.list(text)
# typeof(text1) # verification dutyoe
# lire le premier paragraphe
print(text1[[2]])
```

```
## [1] "We will teach you from ground up on how to scrape the web with R, and will take you through func
```

- Tableaux

```
# Lire la page web
page <- read_html("https://fr.tradingeconomics.com/country-list/gdp-annual-growth-rate")
# Sélectionner le tableau
table <- html_nodes(page, "table")
typeof(table)
```

```
## [1] "list"
```

```
# lire le premier tableau
table_page<-table[[1]]
print(table_page)
```

```
## {html_node}
## <table class="table table-hover table-heatmap" data-sortable="">
##  [1] <thead><tr>\n<th style="cursor: pointer">Pays</th>\r\n                    ...
##  [2] <tr class="datatable-row">\n<td style="max-width: 120px; overflow: hidde ...
##  [3] <tr class="datatable-row-alternating">\n<td style="max-width: 120px; ove ...
##  [4] <tr class="datatable-row">\n<td style="max-width: 120px; overflow: hidde ...
##  [5] <tr class="datatable-row-alternating">\n<td style="max-width: 120px; ove ...
##  [6] <tr class="datatable-row">\n<td style="max-width: 120px; overflow: hidde ...
##  [7] <tr class="datatable-row-alternating">\n<td style="max-width: 120px; ove ...
##  [8] <tr class="datatable-row">\n<td style="max-width: 120px; overflow: hidde ...
##  [9] <tr class="datatable-row-alternating">\n<td style="max-width: 120px; ove ...
## [10] <tr class="datatable-row">\n<td style="max-width: 120px; overflow: hidde ...
## [11] <tr class="datatable-row-alternating">\n<td style="max-width: 120px; ove ...
## [12] <tr class="datatable-row">\n<td style="max-width: 120px; overflow: hidde ...
## [13] <tr class="datatable-row-alternating">\n<td style="max-width: 120px; ove ...
## [14] <tr class="datatable-row">\n<td style="max-width: 120px; overflow: hidde ...
## [15] <tr class="datatable-row-alternating">\n<td style="max-width: 120px; ove ...
## [16] <tr class="datatable-row">\n<td style="max-width: 120px; overflow: hidde ...
## [17] <tr class="datatable-row-alternating">\n<td style="max-width: 120px; ove ...
## [18] <tr class="datatable-row">\n<td style="max-width: 120px; overflow: hidde ...
## [19] <tr class="datatable-row-alternating">\n<td style="max-width: 120px; ove ...
## [20] <tr class="datatable-row">\n<td style="max-width: 120px; overflow: hidde ...
## ...
```

```
# Extraire le tableau en tant que dataframe
df <- html_table(table)
print(df)
```

```
## [[1]]
## # A tibble: 24 x 5
##    Pays            Dernier Précédent Référence Unité
##    <chr>             <dbl>     <dbl> <chr>     <chr>
##  1 Indonésie          5.03      5.01 2023-03   %
##  2 Chine              4.5       2.9  2023-03   %
##  3 Inde               4.4       6.3  2022-12   %
##  4 Arabie Saoudite    3.9       5.5  2023-03   %
##  5 Espagne            3.8       2.9  2023-03   %
##  6 Mexique            3.7       3.6  2023-03   %
##  7 Turquie            3.5       4    2022-12   %
##  8 Australie          2.7       5.9  2022-12   %
##  9 Canada             2.07      3.77 2022-12   %
## 10 Argentine          1.9       5.9  2022-12   %
```

```
## # i 14 more rows
```

```
# Determination du type
typeof(df)
```

```
## [1] "list"
```

```
# astuce cliquer sur df dans environnement
# enregistrer au format csv apres conversion
df<-as.data.frame(df)
```

- Images

```
# Lire une image dans R.
library(magick)
frink <- image_read("https://jeroen.github.io/images/frink.png")
print(frink)
```

```
## # A tibble: 1 x 7
##   format width height colorspace matte filesize density
##   <chr>  <int>  <int> <chr>      <lgl>    <int> <chr>
## 1 PNG      220    445 sRGB       TRUE     73494 72x72
```

```
# Afficher les caracteristiques de l'image avec la fonction image_info()
image_info(frink)
```

```
## # A tibble: 1 x 7
##   format width height colorspace matte filesize density
##   <chr>  <int>  <int> <chr>      <lgl>    <int> <chr>
```

```
## 1 PNG      220     445 sRGB       TRUE      73494 72x72
```

```
# Conversion de format: Rendu png en jpeg
image_write(frink, path = "D:/ISEP2/Semestre_2/R/Cours_R_2023/frink.jpg", format = "jpeg", quality = 75)
# Faire pivoter
image_rotate(frink, 45)
```

```
# image_flip(frink)
# image_flop(frink)
# print(frink) # retour a la position iniatiale
```

-
Audios
et
Videos
**Et par
moyens
nous
ar-
rivons
à les
ex-
traire
?**
La
biblio-
theque
rvest
offre
une
varieté
de fonc-
tions
d'extractions
des
don-
nées
dont
les plus
util-
isées
sont:
read_html(url)
: ex-
traire
le con-
tenu
HTML
d'une
URL
donnée
html_nodes()
:pour
ex-
traire
des élé-
ments
spéci-
fiques
d'une
page
web.

html_attrs() :identifie les attributs (utiles pour le débogage), les plus connus sont “http://” ou “https://” pour les sites web, “ftp://” pour le transfert de fichiers,.com pour moteurs de recherche etc.

html_table() :transforme les tableaux HTML en blocs de données

html_text() :supprime les balises HTML et extrait uniquement le texte

V.Application du web mining: Analyse de sentiments des donnees Twitter

L'un des exemples concrets de cas d'utilisation du Web Mining dans R est l'analyse de sentiments à partir des commentaires sur les réseaux sociaux(*Twitter et Facebook* en particuliers) L'analyse de sentiment est utile pour comprendre la perception des utilisateurs à l'égard d'un produit d'une entreprise ou d'un service.

l'analyse de données web peut être utilisée pour avoir une idee ou prédire le comportement des utilisateurs. Par exemple, les entreprises peuvent utiliser des données web pour prédire les comportements d'achat des clients. Pour extraire et analyser des données Twitter dans R, vous pouvez suivre les étapes générales suivantes :

1.Creation de comptes:

Créez un compte de développeur Twitter.

Et créez une application sur le site du développeur Twitter (https://developer.twitter.com/). pour Obtenez les clés d'API (consumer key, consumer secret, access token, access secret) pour votre application.

2.Installer et charger les packages nécessaires :

3.Utilisez la fonction search_tweets()

pour effectuer une recherche de tweets en utilisant des mots-clés, des hashtags, des noms d'utilisateurs, etc.

Une fois que vous avez extrait les tweets, vous pouvez utiliser les fonctions du package *tidyverse* pour manipuler et analyser les données. Par exemple, vous pouvez : Utiliser mutate() pour ajouter de nouvelles colonnes calculées.

Utiliser *filter()* pour filtrer les tweets en fonction de critères spécifiques.

Utiliser *group_by()* et *summarize()* pour agréger les données.

Utiliser *ggplot()* pour créer des visualisations des données, etc.

4.Authentifier votre application Twitter

5.Effectuer une requête d'extraction de données

6.Manipuler et analyser les données

7.Visualisation des données

VI.Respect des politiques de confidentialites

Lors de l'extraction de données web à partir de plateformes telles que Twitter et Facebook, il est important de respecter les politiques de confidentialité et les conditions d'utilisation de ces plateformes. Chaque plateforme a ses propres conditions et restrictions quant à l'utilisation de leurs données.

Voici un aperçu général des politiques de confidentialité pour l'extraction de données web sur Twitter

-Les développeurs doivent se conformer aux règles de l'API Twitter et à la politique de développement de Twitter.

-Les données extraites à partir de Twitter doivent être utilisées conformément aux règles d'utilisation de l'API Twitter.

-Les données extraites ne doivent pas être utilisées pour l'identification personnelle ou la création de profils d'utilisateurs sans leur consentement.

-Les développeurs doivent respecter les limites d'utilisation de l'API Twitter, telles que le nombre de requêtes autorisées par période de temps.

Conclusion

En plus des contraintes des politiques de confidentialites,il est important de noter que l'analyse de sentiment à partir de données Facebook est souvent complexe en raison de la grande quantité de données à traiter, de la variété des types de posts et de la complexité du langage utilisé. Il est donc recommandé de travailler avec des données d'échantillonnage pour limiter le temps de traitement et de tester plusieurs méthodes pour déterminer celle qui est la plus appropriée pour l'analyse spécifique que l'on souhaite réaliser.