Web scraping with Python on Jordanian news websites to see how many articles have been published about Orange

Dana Ghazal

1. Web Scraping: Getting Data

Web scraping is an automatic method to obtain large amounts of data from websites. Web scraping requires two parts, namely the crawler and the scraper. The crawler is an artificial intelligence algorithm that browses the web to search for the particular data required by following the links across the internet. The scraper, on the other hand, is a specific tool created to extract data from the website.

The first step is to get data from news websites, so in this code, we will used the most popular Python libraries for web scraping: urllib.request, requests and BeautifulSoup.

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
import requests
import mysql.connector

url = "http://www.almadenahnews.com/search?q=%D8%A3%D9%88%D8%B1%D9%86%D8%AC"
url1 =
requests.get("https://www.addustour.com/search.php?search=%D8%A3%D9%88%D8%B1%
D9%86%D8%AC")
url2 =
"https://www.jordanzad.com/index.php?page=tag&hashtag=%D8%A3%D9%88%D8%B1%D9%8
6%D8%AC"
url3 = "http://www.sarahanews.net/?s=%D8%A3%D9%88%D8%B1%D9%86%D8%AC"

client = urlopen(url)
client1 = url1.content
client2 = urlopen(url2)
client3 = urlopen(url3)

html = client.read()
html1 = client2.read()
html2 = client3.read()

soup = BeautifulSoup(html, "html.parser")
soup1 = BeautifulSoup(client1, "lxml")
soup2 = BeautifulSoup(html1, "html.parser")
soup3 = BeautifulSoup(html2, "html.parser")

containers = soup.find_all("div", {"class": "search_cart"})
containers2 = soup2.find_all("li")
containers3 = soup3.find_all("div", {"class": "jeg_postblock_content"})
containers1 = soup1.find_all("li", {"class": "search"})
```

```
title1 = containers[0].a.text.replace('\n', '').strip()
title2 = containers[3].a.text.replace('\n', '').strip()

date1 = containers[2].text.replace('\n', '').strip()
date2 = containers[5].text.replace('\n', '').strip()

title4 = containers2[1].a.text.replace('\n', '').strip()
title5 = containers2[2].a.text.replace('\n', '').strip()
title6 = containers2[3].a.text.replace('\n', '').strip()
title7 = containers2[4].a.text.replace('\n', '').strip()

datei = soup2.find_all("span", {"class": "date"})
date4 = datei[0].text.strip()
date5 = datei[1].text.strip()
date6 = datei[2].text.strip()
date7 = datei[3].text.strip()

title8 = containers3[0].a.text.replace('\n', '').strip()
title9 = containers3[1].a.text.replace('\n', '').strip()
title10 = containers3[2].a.text.replace('\n','').strip()
date8 = containers3[0].div.text.replace('\n', '').strip()
date9 = containers3[1].div.text.replace('\n', '').strip()
date10 = containers3[2].div.text.replace('\n','').strip()
```

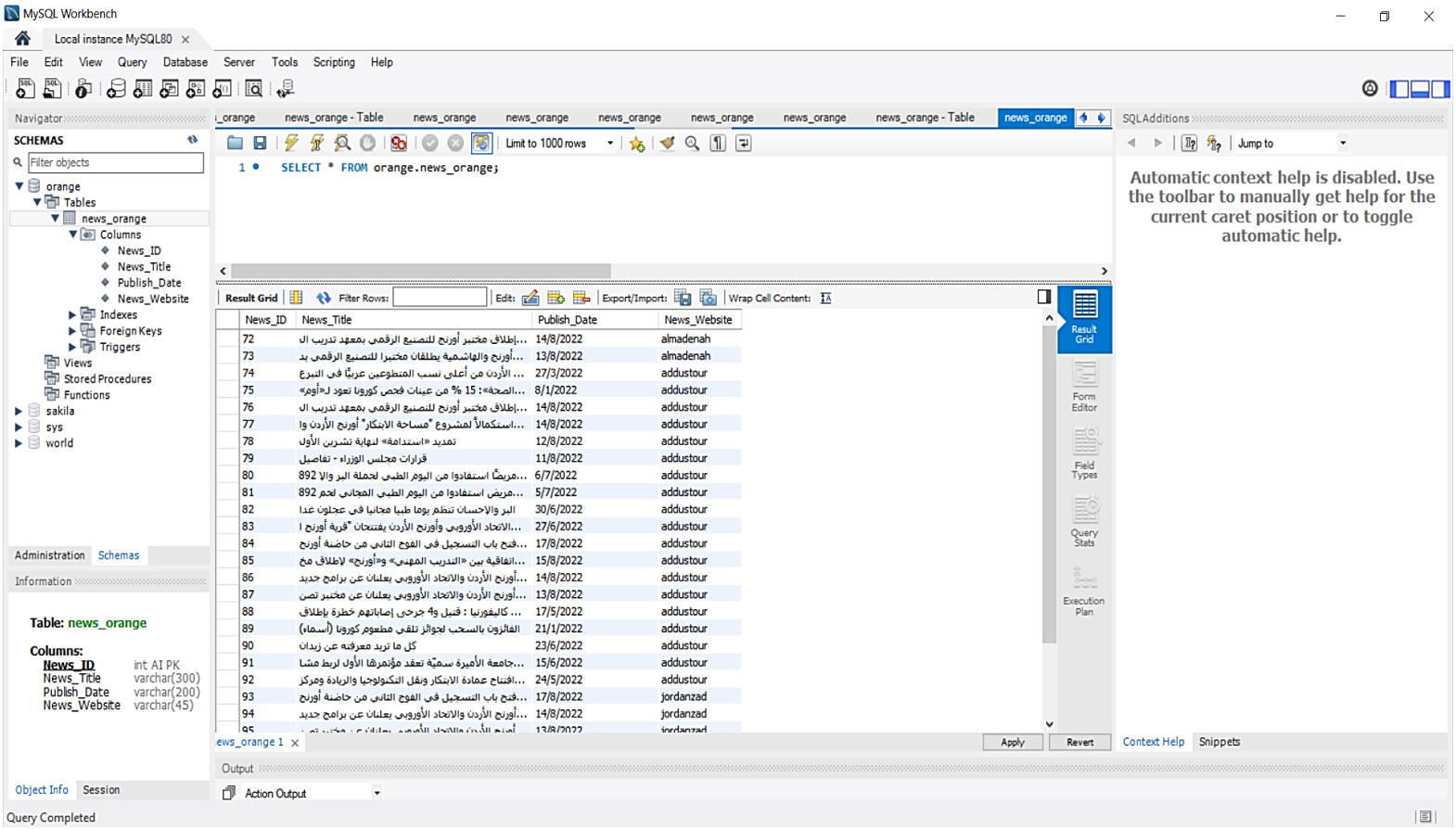
2. Saving our data

What do we have now? The second step we want to do now is to save that data so we don't have to make those requests again. We will store the scraped data in a MySQL database by this code.

```
"""**Saving data to Mysql Database**"""
mydb = mysql.connector.connect(
    host="localhost",
    user="root",
    password="123456",
    database="orange"
)

mycursor = mydb.cursor()

sql = "INSERT INTO news_orange (News_Title, Publish_Date, News_Website)
VALUES (%s,%s,%s)"
val1 = (title1, date1, "almadenah")
val2 = (title2, date2, "almadenah")
mycursor.execute(sql, val1)
mycursor.execute(sql, val2)

for i in containers1:
    titlee = i.findAll("h3")
    title3 = titlee[0].text.strip()

    datee = i.findAll("div", {"class": "date"})
    date3 = datee[0].text.strip()
    val3 = (title3, date3, "addustour")
```

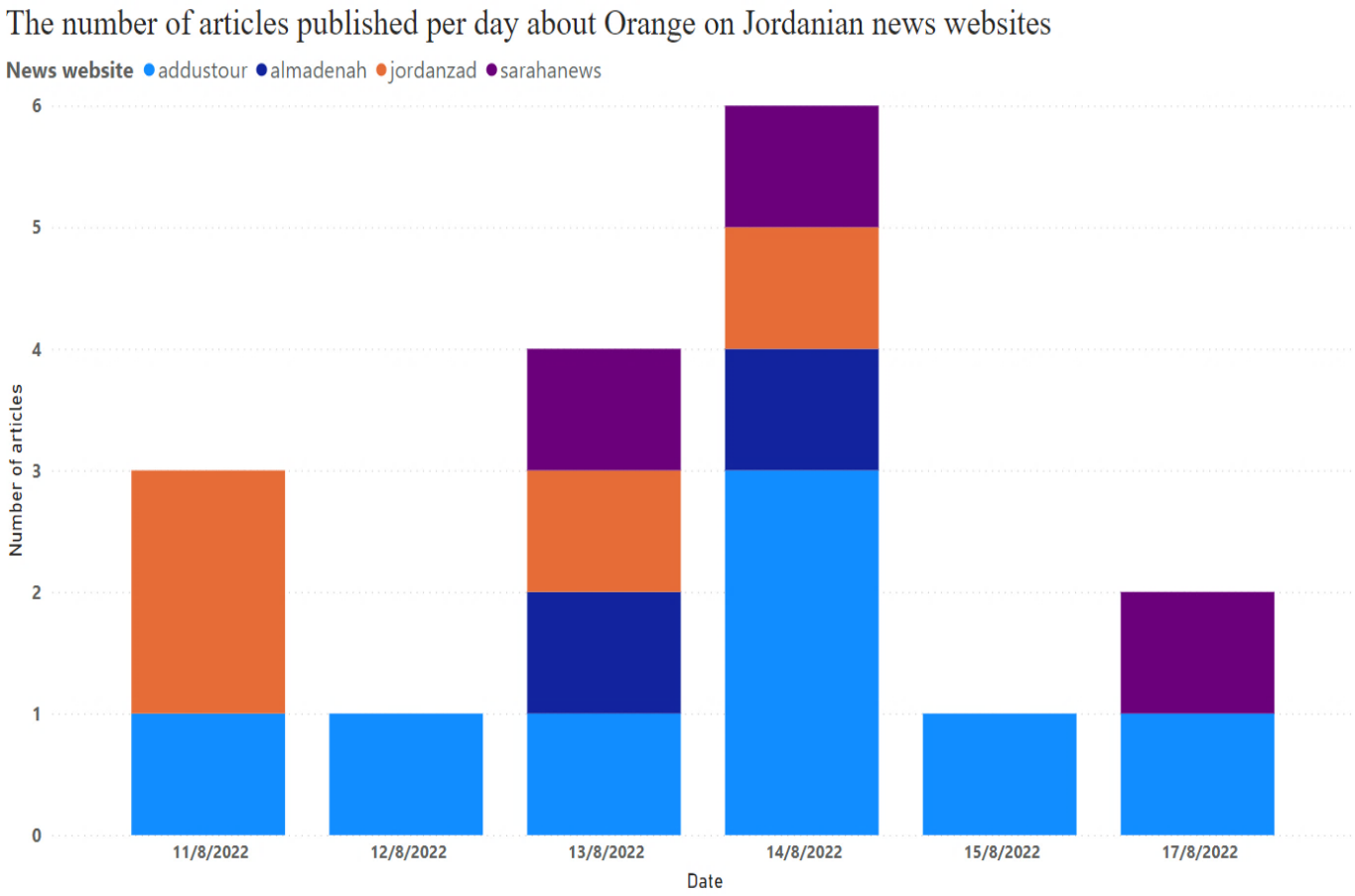
```
    mycursor.execute(sql, val3)

val4 = (title4, date4, "jordanzad")
val5 = (title5, date5, "jordanzad")
val6 = (title6, date6, "jordanzad")
val7 = (title7, date7, "jordanzad")
val8 = (title8, date8, "sarahanews")
val9 = (title9, date9, "sarahanews")
val10 = (title10, date10, "sarahanews")
mycursor.execute(sql, val4)
mycursor.execute(sql, val5)
mycursor.execute(sql, val6)
mycursor.execute(sql, val7)
mycursor.execute(sql, val8)
mycursor.execute(sql, val9)
mycursor.execute(sql, val10)
mydb.commit()
```

Here, the figure shows how the data was stored in a table called news_orange in a database called orange.

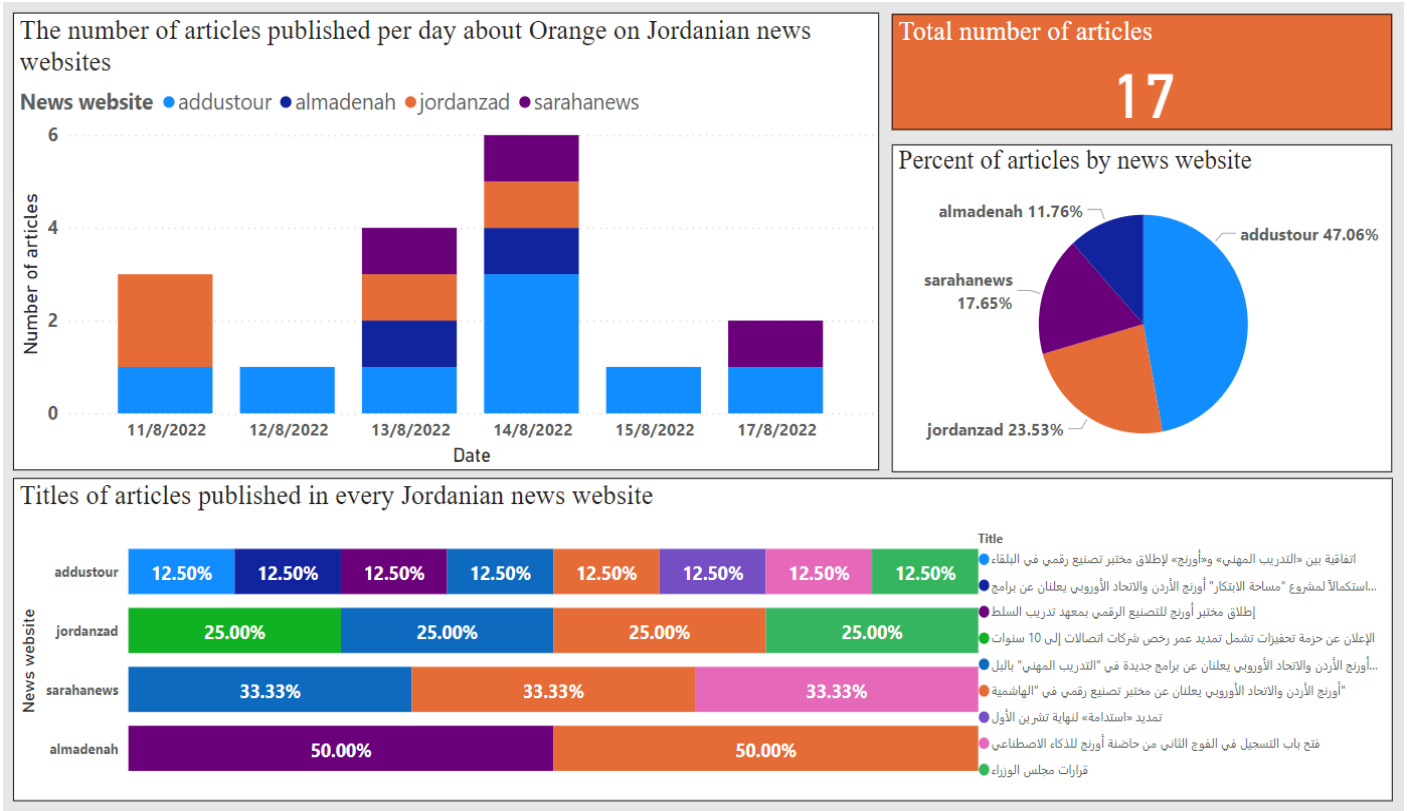


3. Visualizing the data

To create a stacked column chart showing the number of articles published per day about Orange on Jordanian news websites from 11/8/2022 to 17/8/2022, we will use the Power Bi tool.



As shown in the figure, this graph shows that Addustour News received the largest number of articles, followed by Jordanzad News, Saraha News and Almadenah News, and that on August 14, 2022, news websites accounted for 17.65% of the number of articles, while on August 16, 2022, no article was published on news websites, and we also note that Addustour News and Jordanzad News tied with the highest average number of articles at 1.33.



As shown in the dashboard, the total number of published articles is 17, Addustour News accounted for 47.06% of the articles. Also we note that the article titled “Orange Jordan and the European Union announce a digital manufacturing laboratory in Al Hashemite” is the most published by news websites.