Statystyka użytkowników serwisu bit.ly

MICHAŁ KACZMARCZYK
ALEXANDROS CHANTELIDIS
MACIEJ DUDEK

Czym jest bit.ly?

Bitly jest to platforma której głównym celem jest skracanie linków (hiperlinków).

Usługa ta stałą się bardzo popularna na Twitterze gdzie w maju 2009 zostałą domyślną opcją.

Bitly oferuję też płatne rozwiązania zwane „Bitly Enterprise” które zapewniają różne funkcje pomocne w promowaniu marek. Przykładem tu może być firma Pepsi używająca skrótu pep.si. Firma zapewnia też dane statystyczne dla fukncji społecznościowych.

Technologia

- Firma używa HTTP 301 do przekierowania swoich skrótów. Są one trwałe i nie mogą być zmienione po ich stworzeniu.
- Od 2010 uzytkownicy mogą automatycznie generować kody QR (quick response) po zeskanowaniu przez mobilny skaner kodów dzieki czemu zostaną automatycznie przekierowaniu do skracanego linku
- W 2013 firma wypuściła nową fukncję wyszukiwania zwaną „Bitmarks” oraz aplikacją na IPhony

Program

- Program został napisany w IPytonie w środowisku Jupyter z użyciem bibliotek nympy, matplotib i pandas.

- Działanie zostanie pokazane w kilku krokach

Import potrzebnych bilbiotek

```
In [3]: #import bibliotek
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import json
from pandas import DataFrame, Series
```

Numpy – od obliczeń
Pandas – od stref czasowych
Matplotlib – od wykresów
Json – do czytania pliku

Wczytanie pliku i wyświetlenie jego zawartości

```
In [4]: #wczytanie pliku
path = 'usagov_bitly_data2012-03-16-1331923249.txt'
records = [json.loads(line) for line in open(path)]

In [5]: #wyświetlenie zawartości pliku
records[0]

Out[5]: {'a': 'Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/535.11
(KHTML, like Gecko) Chrome/17.0.963.78 Safari/535.11',
'al': 'en-US,en;q=0.8',
'c': 'US',
'cy': 'Danvers',
'g': 'A6qOVH',
'gr': 'MA',
'h': 'wfLQtf',
'hc': 1331822918,
'hh': '1.usa.gov',
'l': 'orofrog',
'll': [42.576698, -70.954903],
'nk': 1,
'r': 'http://www.facebook.com/l/7AQEFzjSi/1.usa.gov/wfLQtf',
't': 1331923247,
'tz': 'America/New_York',
'u': 'http://www.ncbi.nlm.nih.gov/pubmed/22415991'}
```

- Plik ma kilkanaście wpisów odnośnie użytkowników takich jak ich język, lokalizację, strefę czasową, system operacyjny itd..

Wybranie kolumny ze strefami czasowymi i ich zliczenie

```
In [6]: #pierwsze 10 stref czasowych
frame = DataFrame(records)
frame['tz'][:10]

Out[6]: 0      America/New_York
1        America/Denver
2      America/New_York
3     America/Sao_Paulo
4      America/New_York
5      America/New_York
6         Europe/Warsaw
7
8
9
Name: tz, dtype: object
```

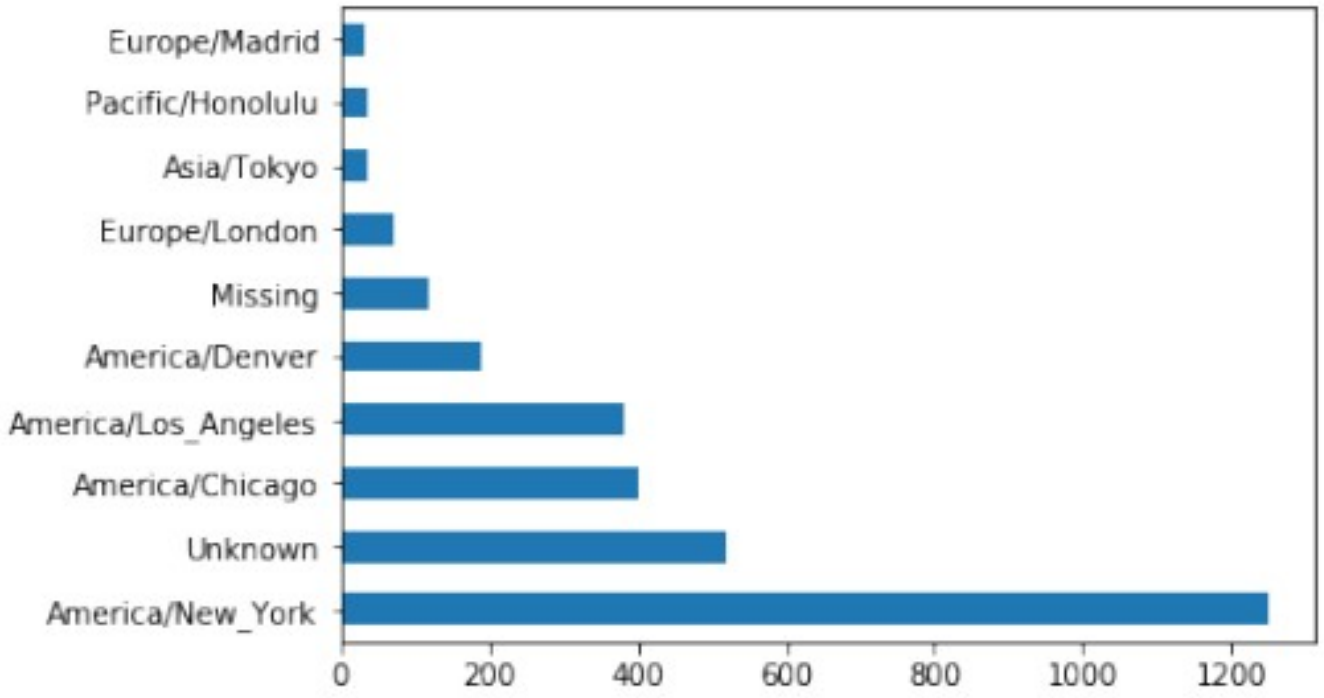
```
In [7]: #zliczenie stref czasowych i wyświetlenie najbardziej popularnych
tz_counts = frame['tz'].value_counts()
tz_counts[:15]

Out[7]: America/New_York          1251
521
America/Chicago                 400
America/Los_Angeles             382
America/Denver                  191
Europe/London                    74
Asia/Tokyo                       37
Pacific/Honolulu                 36
Europe/Madrid                    35
America/Sao_Paulo                33
Europe/Berlin                    28
Europe/Rome                      27
America/Rainy_River              25
Europe/Amsterdam                 22
America/Indianapolis             20
Name: tz, dtype: int64
```

Zastąpienie nieistniejących danych frazą „Unknown”

```
In [8]: #czyszczenie nieistnejących wartości
clean_tz = frame['tz'].fillna('Missing')
clean_tz[clean_tz == ''] = 'Unknown'
tz_counts = clean_tz.value_counts()
tz_counts[:3]

Out[8]: America/New_York     1251
Unknown                       521
America/Chicago               400
Name: tz, dtype: int64
```

Sporządzenie wykresu najbardziej popularnych stref czasowych

```
In [9]: #Wykres najbardziej popularnych stref czasowych
tz_counts[:10].plot(kind='barh')

Out[9]: <matplotlib.axes._subplots.AxesSubplot at 0x74ca2f0>
```

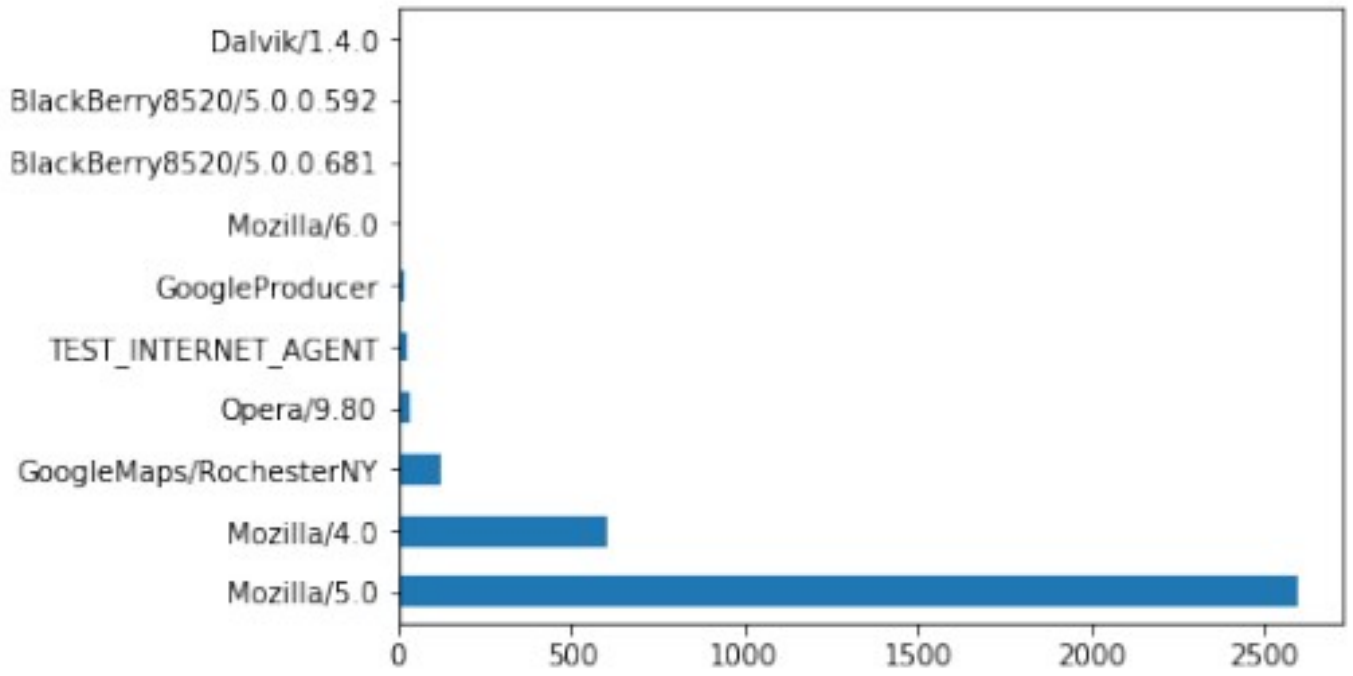


Podzielenie kolumny „a” na części (zawiera dane o przeglądarce)

```
In [10]: #Statystyka przeglądarek użytkowników
results = Series([x.split()[0] for x in frame.a.dropna()])
results[:5]

Out[10]: 0               Mozilla/5.0
1    GoogleMaps/RochesterNY
2               Mozilla/4.0
3               Mozilla/5.0
4               Mozilla/5.0
dtype: object

In [11]: #Top 10 przeglądarek
results.value_counts()[:10]

Out[11]: Mozilla/5.0                 2594
Mozilla/4.0                  601
GoogleMaps/RochesterNY       121
Opera/9.80                    34
TEST_INTERNET_AGENT           24
GoogleProducer                21
Mozilla/6.0                    5
BlackBerry8520/5.0.0.681       4
BlackBerry8520/5.0.0.592       3
Dalvik/1.4.0                   3
dtype: int64
```

Sporządzenie wykresu używanych przeglądarek

```
In [12]: #Wykres przeglądarek
results.value_counts()[:10].plot(kind='barh', rot=0)

Out[12]: <matplotlib.axes._subplots.AxesSubplot at 0xa1f2d90>
```



Co dalej?

- Program posiada możliwości przeprowdzania statystyk ze wszystkich zawartych danych w pliku, dzieki czemu możemy go poszerzyć o np. statystki odnośnie systemu operacyjnego lub lokalizacja użytkowników na mapie.