# METODY EKSPLORACJI DANYCH

*PROJEKT: "Wykrywanie obiektów sportowych typu stadiony, boiska, hale sportowe według zagęszczenia wpisów na serwisie społecznościowym Twitter"*

**Aleksandra Knapik**

**Julita Musiał**

GiK 1 rok MSU

geoinformatyka

Wrocław, 17.01.2015r.

**Dane:**

Dane pochodzą z wiadomości (tweet'ów) wysyłanych przez użytkowników społeczności serwisu Twitter. Dostęp do tych danych możliwy jest za pomocą modułu Streaming API, udostępnionego przez serwis.

Dane do sprawdzenia poprawności lokalizacji obiektów sportowych zostaną pobrane z serwisu OpenStreetMap.

**Język:**

Do pobierania danych z serwisu Twitter i ich analizy zostanie wykorzystany język **Python**. Aby możliwa była komunikacja z API Twittera konieczna jest również biblioteka tweepy. Dodatkowo wykorzystamy również bibliotekę pandas (do analizy danych i obliczeń statystycznych) oraz matplotlib (do rysowania wykresów). Do połączenia z bazą danych MongoDB konieczna jest także biblioteka PyMongo, zawierająca zestaw narzędzi do pracy z bazą danych.

**Dostęp do danych:**

Dostęp do danych umożliwia Streaming API. Aby móc pobierać tweet'y należy posiadać konto na portalu Twitter, a następnie stworzyć swoją aplikację na stronie **www.apps.twitter.com**. Wygenerowane zostają wówczas kody dostępu i kody autoryzacyjne niezbędne podczas tworzenia aplikacji:

- consumer key
- consumer secret
- access token
- access secret

Za streaming tweet'ów odpowiadają odpowiednie metody i klasy z biblioteki tweepy.

Możliwa jest parametryzacja zapytań streamu. Podając odpowiedni filtr, można pobierać tylko wybrane tweet'y. Przykłady parametrów:

- język twitta (language)
- osoby, które się śledzi (follow)
- słowa kluczowe (track)
- lokalizacja (locations)
- odpowiedzi (replies)

```
from tweepy import Stream
from tweepy import OAuthHandler
from tweepy.streaming import StreamListener
import time


ckey = '[redacted]'
csecret = '[redacted]'
atoken = '[redacted]'
asecret = '[redacted]'

class listener(StreamListener):
    def on_data(self, data):
        try:
            print data
            return True
        except BaseException,e:
            print 'failed ondata,',str(e)
            time.sleep(5)

    def on_error(self, status):
        print status


auth = OAuthHandler(ckey, csecret)
auth.set_access_token(atoken, asecret)

twitterStream = Stream(auth, listener())
twitterStream.filter(locations = [-0.20, 51.40, -0.15, 51.50])
#twitterStream.filter(track = ["Sheeran"])
```

**Obszar badań:**

Dane zbierane są dla obszaru miasta **Los Angeles** w USA, ograniczonego BBox'em: -118.30876350402832, 33.99916579100914, -118.1356430053711, 34.07029354225064.

**Semantyka danych:**

Dane oprócz tego, że mogą być wyświetlane w konsoli mogą być również zapisane do pliku. Zapisywane są one w formacie JSON.

```
16  {"created_at":"Fri Jan 09 23:00:02 +0000 2015","id":553687671382765568,"id_str"
    :"553687671382765568","text":"BONG!  BONG!  BONG!  BONG!  BONG!","source":"\u003ca
    href=\"https:\/\/en.wikipedia.org\/wiki\/Big_Ben\" rel=\"nofollow\"\u003eFaultyBigB
    en\u003c\/a\u003e","truncated":false,"in_reply_to_status_id":null,"in_reply_to_stat
    us_id_str":null,"in_reply_to_user_id":null,"in_reply_to_user_id_str":null,"in_reply
    _to_screen_name":null,"user":{"id":1330109264,"id_str":"1330109264","name":"Faulty
    Big Ben Clock","screen_name":"FaultyBigBen","location":"Westminster, London","url"
    :null,"description":"I am wrong.","protected":false,"verified":false,"followers_cou
    nt":126,"friends_count":0,"listed_count":1,"favourites_count":0,"statuses_count"
    :13573,"created_at":"Fri Apr 05 21:22:52 +0000 2013","utc_offset":0,"time_zone"
    :"London","geo_enabled":true,"lang":"en","contributors_enabled":false,"is_translato
    r":false,"profile_background_color":"C0DEED","profile_background_image_url":"http:\
    /\/pbs.twimg.com\/profile_background_images\/378800000008520159\/d6ace022ff4650c8cd
    f23df98369f861.jpeg","profile_background_image_url_https":"https:\/\/pbs.twimg.com\
    /profile_background_images\/378800000008520159\/d6ace022ff4650c8cdf23df98369f861
    .jpeg","profile_background_tile":true,"profile_link_color":"0084B4","profile_sideba
    r_border_color":"000000","profile_sidebar_fill_color":"DDEEF6","profile_text_color"
    :"333333","profile_use_background_image":true,"profile_image_url":"http:\/\/pbs
    .twimg.com\/profile_images\/3709555224\/8e90e8045390de6bd838b8f328eee2dd_normal
    .jpeg","profile_image_url_https":"https:\/\/pbs.twimg.com\/profile_images\/37095552
    24\/8e90e8045390de6bd838b8f328eee2dd_normal.jpeg","default_profile":false,"default_
    profile_image":false,"following":null,"follow_request_sent":null,"notifications"
    :null},"geo":{"type":"Point","coordinates":[51.500753,-0.124680]},"coordinates"
    :{"type":"Point","coordinates":[-0.124680,51.500753]},"place":{"id":"457b4814b4240d
    87","url":"https:\/\/api.twitter.com\/1.1\/geo\/id\/457b4814b4240d87.json","place_t
    ype":"city","name":"London","full_name":"London, England","country_code":"GB"
    ,"country":"United Kingdom","bounding_box":{"type":"Polygon","coordinates":[[[-0
    .187894,51.483718],[-0.187894,51.5164655],[-0.109978,51.5164655],[-0.109978,51
    .483718]]]},"attributes":{}},"contributors":null,"retweet_count":0,"favorite_count"
    :0,"entities":{"hashtags":[],"trends":[],"urls":[],"user_mentions":[],"symbols":[]}
    ,"favorited":false,"retweeted":false,"possibly_sensitive":false,"filter_level"
    :"medium","lang":"tl","timestamp_ms":"1420844402534"}
```


**Baza danych:**

Ze względu na duży rozmiar pliku JSON z pobranymi tweetami konieczne będzie wykorzystanie bazy danych. Dla celów tego projektu zastosowana będzie baza danych **MongoDB**. MongoDB jest przykładem nierelacyjnego systemu zarządzania bazą danych. Dane składowane są jako dokumenty typu JSON w kolekcjach. Podobnie jak w relacyjnej bazie danych możliwe jest wykonywania zapytań, co zostanie wykorzystane do wstępnej filtracji danych.

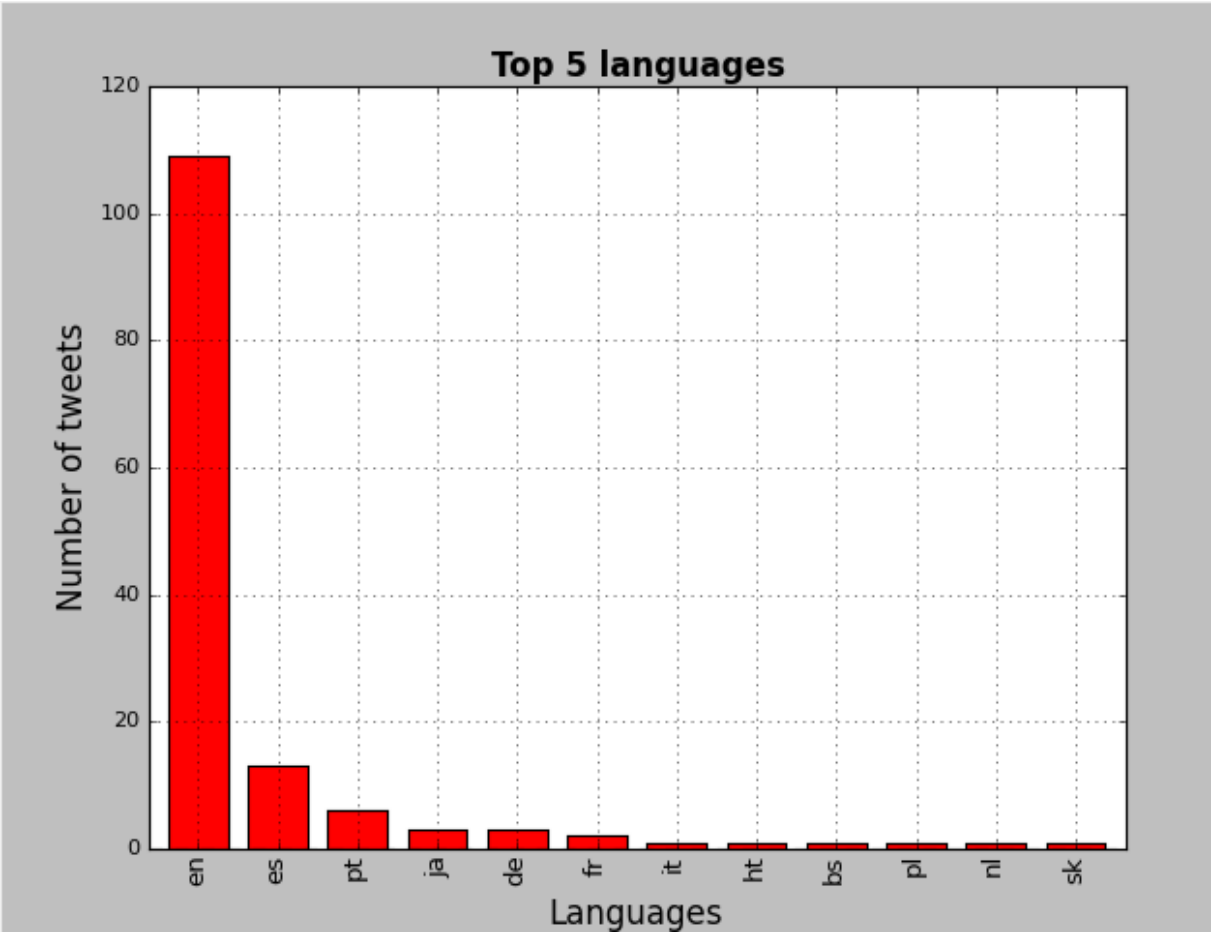
**Etapy opracowania projektu:**

1. Wybór obszaru badawczego
2. Pobranie danych za pomocą Streamu do pliku JSON dla zadanego BBoxa
3. Dodanie danych do bazy danych MongoDB
4. Wstępna filtracja danych:
   4.1. usunięcie tweet'ów nie posiadających geolokalizacji
   4.2. filtracja danych ze względu na słowa kluczowe: "football", "stadium", "sport center", "basketball", itp
5. Zapisanie dokumnetów, które spełniają zadane wyżej kryteria i zapisanie ich do nowej kolekcji
6. Połączenie się z bazą danych MongoDB za pomocą języka Python
7. Napisanie algorytmu grupującego wpisy (tweet'ty)
8. Pobranie lokalizacji stadionów i innych obiektów sportowych z serwisu OpenStreetMap
9. Porównanie wyznaczonej lokalizacji obiektów sportowych na podstawie wpisów z Twittera oraz danych z OSM
10. Wykonanie analiz statystycznych dotyczących tweet'ów

*Przykłady analiz statystycznych:*

```
Python 2.7.5 (default, May 15 2013, 22:43:36) [MSC v.1500 32 bit (Intel)] on win
32
Type "copyright", "credits" or "license()" for more information.
>>> ================================ RESTART ================================
>>>
142
                                                  text lang
0   RT @vinceyue: Watch Pokémon R ... - http://t.c...   de
1   RT @charrington99: Comparison between #reactjs...   en
2   RT @DesignUXUI: I accidentally googled "Jabasc...   en
3   RT @DkaliRam: JavaScript and WebGL http://t.co...   en
4                     Ruby's gona go down so well ??   en
5   #GoogleMaps #API JavaScript Full Example Sourc...   en
6   I've just done three tweets about John Cleeses...   en
7   Shop this similar look JavaScript is currently...   en
8   Be sure to catch Sunday's Vision Call Hosted b...   en
9             RT @ZhaQuese: @R_Chocolatee bye ruby ??   ht
10  RT @BestMovieLine: Monty Python and the Holy G...   en
11  @Rubybeets You're welcome Ruby!  Are you manag...   en
12  a little more css this morning and then back t...   en
13  Watch and LIKE any of @AbdallahNATION's #Pokem...   en
14      State machine in #ruby http://t.co/8foAsLzW8z   de
15  RT @Pokemon_cojp: ?Pokemon Cafe ?Ruby??Sapphir...   ja
16  Is it coincidence that GAME FREAK released Ome...   en
17  Gostei de um vídeo @YouTube de @heitor_games h...   pt
18  @Noahpinion on a more practical level @t0nyyat...   en
19  .@link_lunq ??? @YouTube ???????????: http://t...   ja
```

```
[142 rows x 2 columns]
en    109
es     13
pt      6
ja      3
de      3
fr      2
it      1
ht      1
bs      1
pl      1
nl      1
sk      1
```




**Napotkane problemy:**

Na tym etapie realizacji projektu napotkano następujące problemy:

- trudności z instalacją niezbędnych bibliotek wykorzystywanych do dostępu do Streaming API
- problem z pobieraniem tweet'ów o zadanej lokalizacji ORAZ zadanych słowach kluczowych

**<u>Dotychczasowe efekty pracy:</u>**

### 1. Wybór obszaru badawczego

Obszar badawczy to miasto **Los Angeles** w Stanach Zjednoczonych Ameryki Północnej. Obszar został ograniczony BBoxem: -118.30876350402832, 33.99916579100914, -118.1356430053711, 34.07029354225064.


### 2. Pobranie danych za pomocą Streamu do pliku JSON dla zadanego BBoxa

Dane pobierane były w godzinach 15:00 - 10:00 czasu polskiego (UTC+1) , 8:00 - 24:00 czasu UTC-8. W tym czasie pobrany został plik testowy o rozmiarze 168 MB.

### 3. Dodanie danych do bazy danych MongoDB

Zaimportowano dane do bazy MongoDB. Kolekcja zawiera 59 719 dokumentów.

### 4. Wstępna filtracja danych:

#### 4.1. usunięcie tweet'ów nie posiadających geolokalizacji

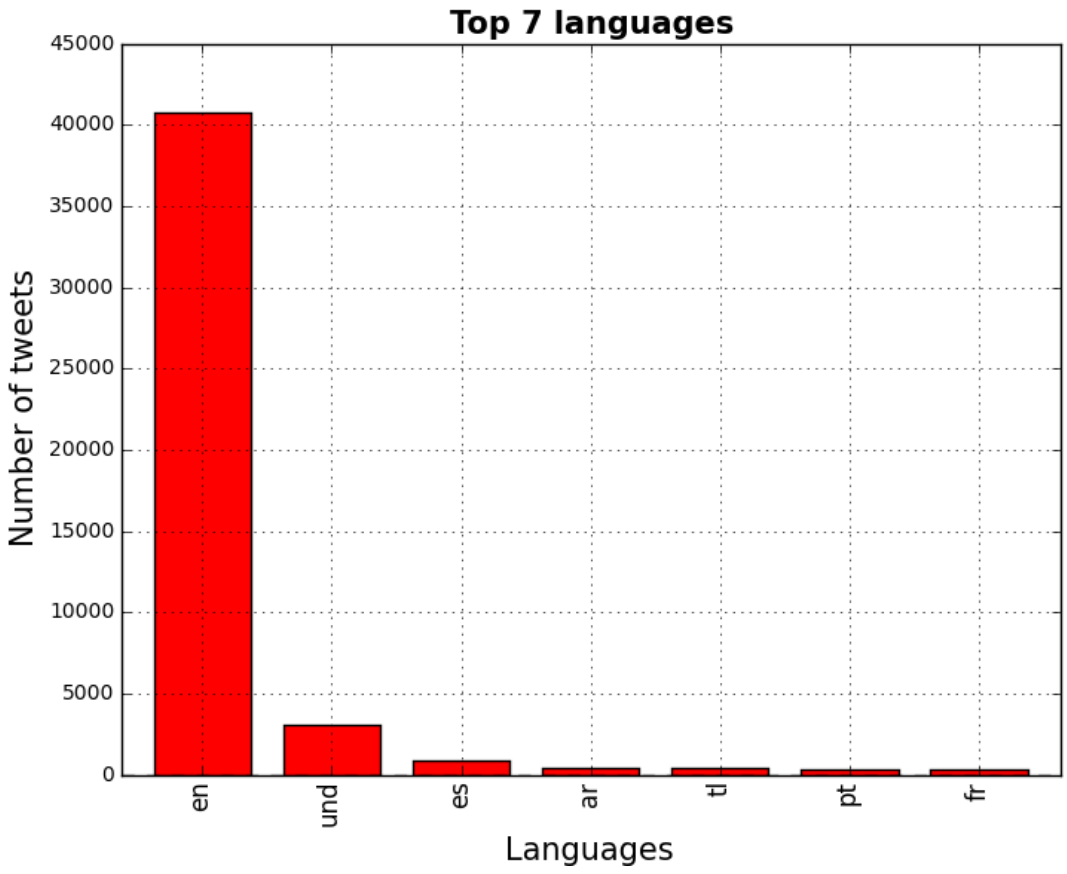
W wyniku zapytania na bazie danych otrzymano 47 506 dokumentów, które posiadają geolokalizację. Z kolekcji usunięto 12 213 dokumentów nie posiadających geolokalizacji.

**4.2. filtracja danych ze względu na słowa kluczowe: "football", "stadium", "sport center", "basketball", itp**

Wystąpienie słów kluczowych:

| | |
|---|---|
| sport: | 57 |
| football: | 24 |
| basketball: | 23 |
| hokey: | 20 |
| soccer: | 20 |
| tennis: | 7 |
| stadium: | 4 |

5. **Zapisanie dokumentów, które spełniają zadane wyżej kryteria i zapisanie ich do nowej kolekcji**
6. **Połączenie się z bazą danych MongoDB za pomocą języka Python**
7. **Wykonanie analiz statystycznych dotyczących tweet'ów**



Languages of tweets

| | |
|---|---|
| en | 40791 |
| und | 3073 |
| es | 875 |
| ar | 415 |
| tl | 384 |
| pt | 303 |
| fr | 301 |
| in | 211 |
| ht | 187 |
| tr | 94 |
| it | 89 |
| ja | 70 |
| de | 69 |
| et | 68 |
| nl | 65 |
| fi | 49 |
| pl | 45 |
| ro | 40 |
| vi | 39 |
| no | 39 |
| da | 37 |
| ru | 36 |
| sv | 35 |
| cy | 29 |
| ko | 22 |
| zh | 22 |
| sk | 21 |
| sl | 19 |
| hu | 13 |
| is | 13 |
| th | 10 |
| lt | 9 |
| lv | 8 |
| fa | 6 |
| hr | 6 |
| bs | 5 |
| uk | 2 |
| bg | 2 |
| bo | 1 |
| ta | 1 |
| iw | 1 |
| el | 1 |