

Twitter Data Visualization Models and Algorithm using Matplotlib

Arief Anbiya (anbarief@live.com)

November 2, 2018

- ▶ Name: Arief Anbiya
- ▶ Background: Applied Mathematics
- ▶ [Twitter](#) & [Github](#): @anbarief
- ▶ Experience:
 - ▶ LabMath Indonesia researcher (2014-2015)
 - ▶ Product Actuarial
 - ▶ Assistant Lecturer in Calculus (2017-2018)

Presentation Objectives

1. "*Why*"
2. the diagram of the process
3. tweets collection (using [Tweepy](#))
4. Cases, Data Vis models and [Matplotlib](#)

Why?

- ▶ Picture – story – concept

Why?

- ▶ Picture – **story** – concept
- ▶ Data visualization ⊂ **statistical analysis**

Why?

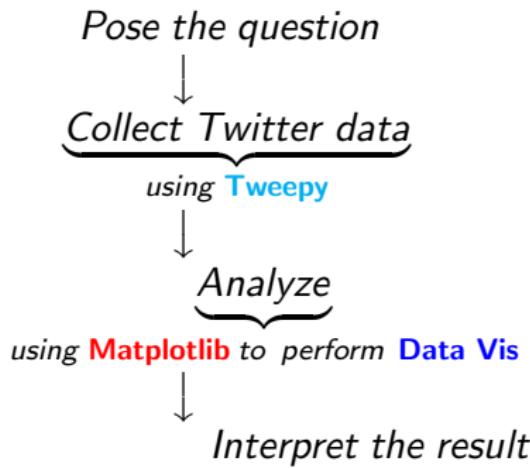
- ▶ Picture – story – concept
- ▶ Data visualization \subset statistical analysis
- ▶ New algorithms - creativity - problem-solving abilities

Why?

- ▶ Picture – story – concept
- ▶ Data visualization \subset statistical analysis
- ▶ New algorithms - creativity - problem-solving abilities
- ▶ Perspectives

The Statistical Process

P-C-A-I



Data Collection

Using Tweepy

*Need to have **consumer_key**, **consumer_secret**, **access_token**, **access_secret** by applying to <https://apps.twitter.com>

- ▶ Code example:

```
import tweepy

consumer_key = 'ecGxfboL66oO2ZwxfKkg7q3QK'
consumer_secret = 'exVRiv517gdwkPLP19PtIQMEIRjxgJr21JZCAAQYIqJCUW5vmh'
access_token = '3151279508-Ywd662Zv97le7E7I97dUm0e3s2X8yYBLoJQd6Gr'
access_secret = 'BH5REW4V7RdGadMr31NLY9ksFypG12m8BR04S32ZF7jO3'

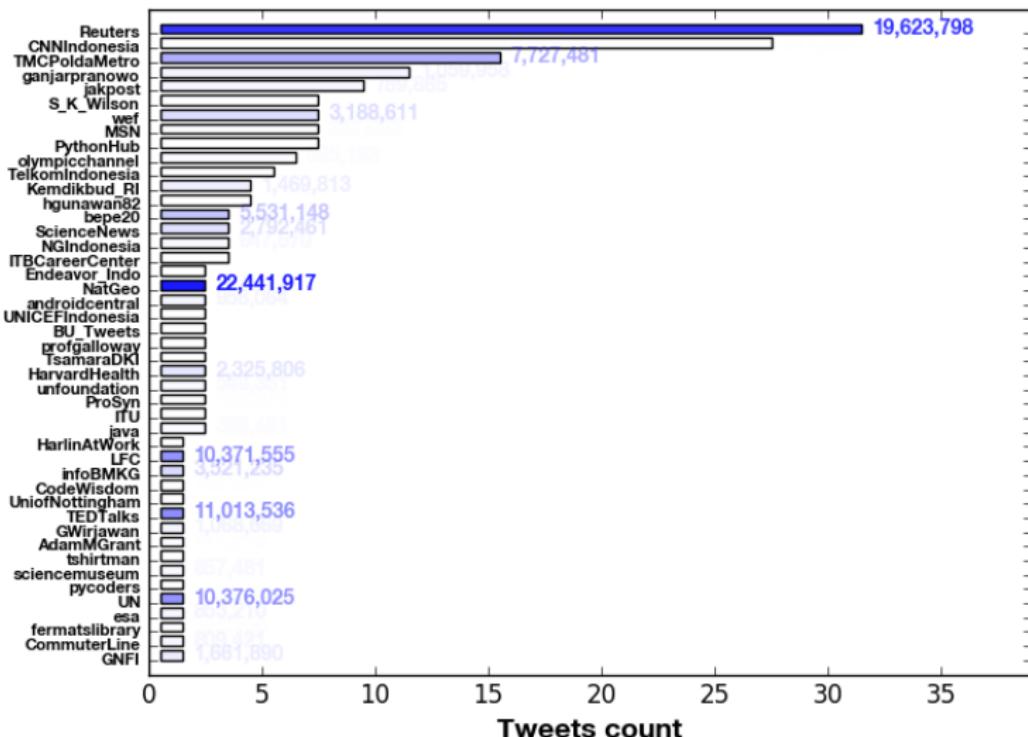
auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_secret)
api = tweepy.API(auth)
```

- ▶ **api.home_timeline()**
- ▶ **api.user_timeline(screen_name = 'NatGeo', include_rts = True)**
- ▶ Tweet = **tweepy.models.Status**

Data Visualization: combining multiple Matplotlib features

Case 1: Accounts comparison

Tweets from Home Timeline (in an interval)



Case 1: Accounts comparison

Matplotlib features used:

- ▶ The bars

```
fig, ax = plt.subplots();
for i in range(..):
    ax.hbar(..., \
            color = (0, 0, 1, followers_count[i]/max_folls))
```

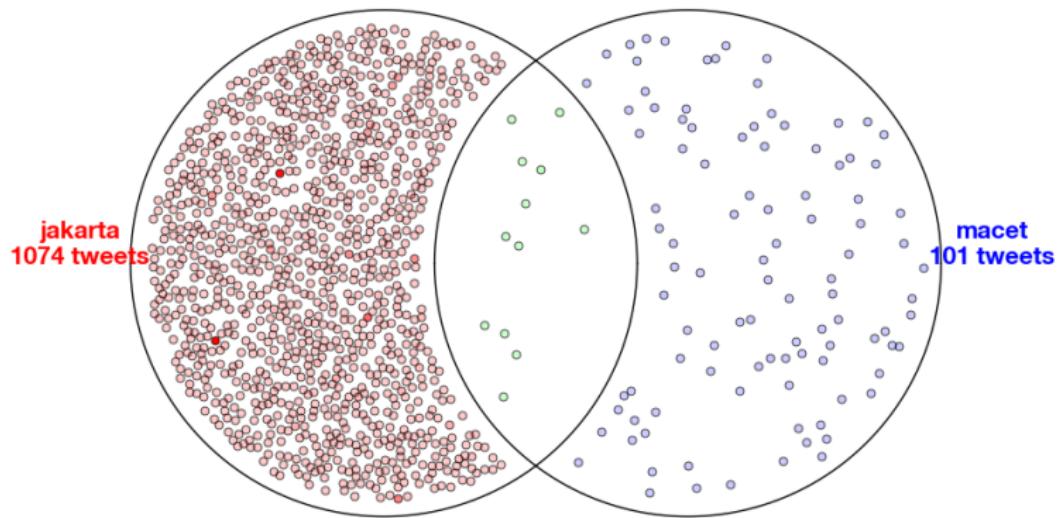
- ▶ The font style

```
import matplotlib.font_manager as fm
def style_the_num(number):
    ...
for i in range(..):
    ax.text(..., style_the_num(followers_count[i]))
ax.set_yticks(range(len(usernames)))
helvetica = fm.FontProperties(fname = 'Helvetica.ttf')
ax.set_yticklabels(usernames, fontproperties = helvetica)
```

Case 2: Twitter Venn

@CNNIndonesia ≈53,000 tweets from 1 Jan 2018 to 16 Okt 2018

Twitter Venn Diagram



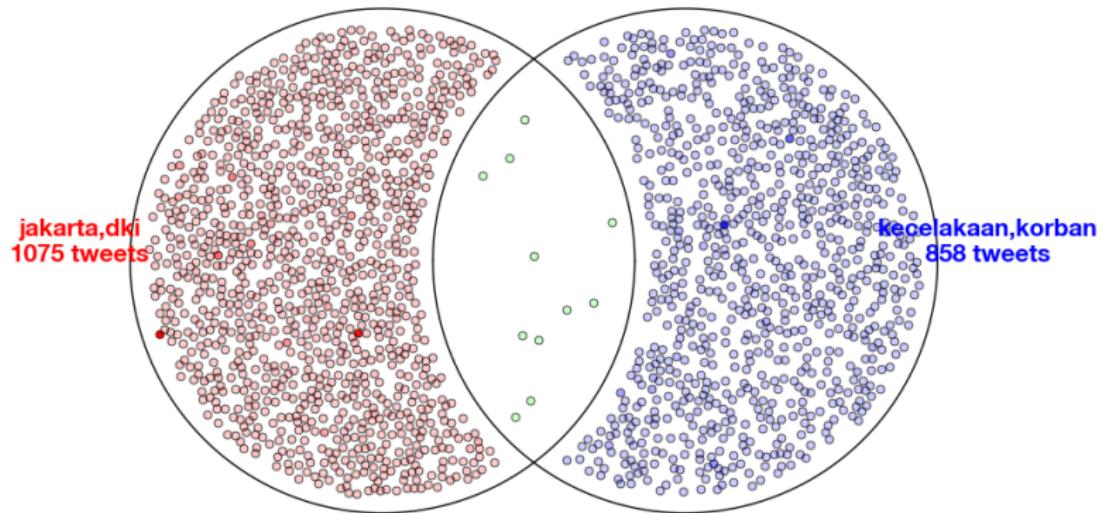
Dianggap telah menyebabkan kemacetan, Gubernur DKI Jakarta @ aniesbaswedan berencana memanggil perusahaan ojek online se...
RTs: 66 (2018-07-23 09:30:10)

Pengamat menilai penerapan ganjil genap di ruas jalan tol untuk mengurai kemacetan di Jakarta belumlah efektif. Menurutn...
RTs: 10 (2018-04-17 10:04:41)



Case 2: Twitter Venn

Twitter Venn Diagram



Bagaimana situasi terkini proses evakuasi korban robohnya selasar BEI Jakarta Selatan? Berikut laporannya. <http://cnn.id...>
RTs: 59 (2018-01-15 13:00:09)

Gubernur @ aniesbaswedan mengaku belum memikirkan naturalisasi sungai untuk menyiasati luapan Sungai Ciliwung. Evaluasi ...
RTs: 39 (2018-02-06 10:29:06)



Case 2: Twitter Venn

- ▶ Steps:
 - ▶ **numpy.load(...tweets data...)**
 - ▶ Plot the two large circles intersecting each other
 - ▶ Categorize each tweet (using **categorize** function, and also **TextBlob**)
 - ▶ Set points inside or outside the circles (using **set_points** and the other user defined funcs)
 - ▶ Plot the points
- ▶ User defined functions:

```
distance(point1, point2)
```

```
in_circle(point, center, radius)
```

```
out_circle(point, center, radius)
```

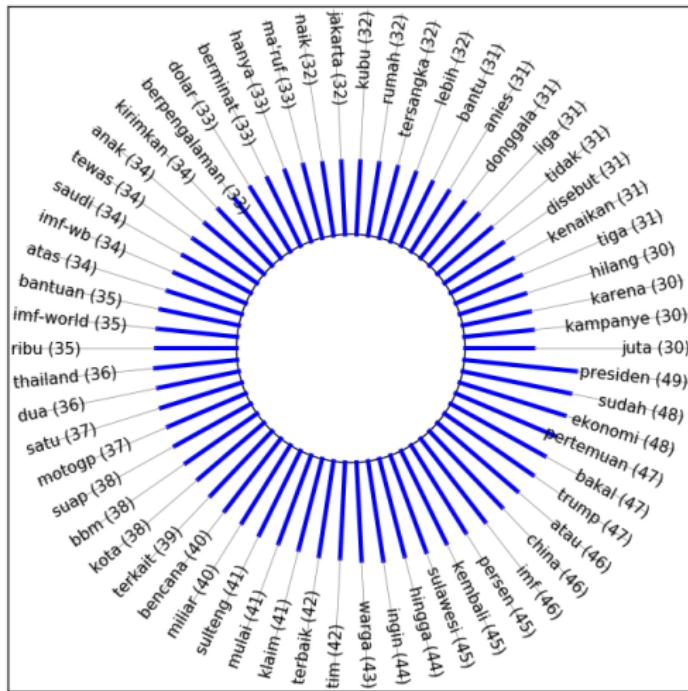
```
collide(rand, points)
```

```
categorize(tweets)
```

```
set_points(categorized_tweets, center1, radius1, center2, ra
```

Case 3: Word frequency

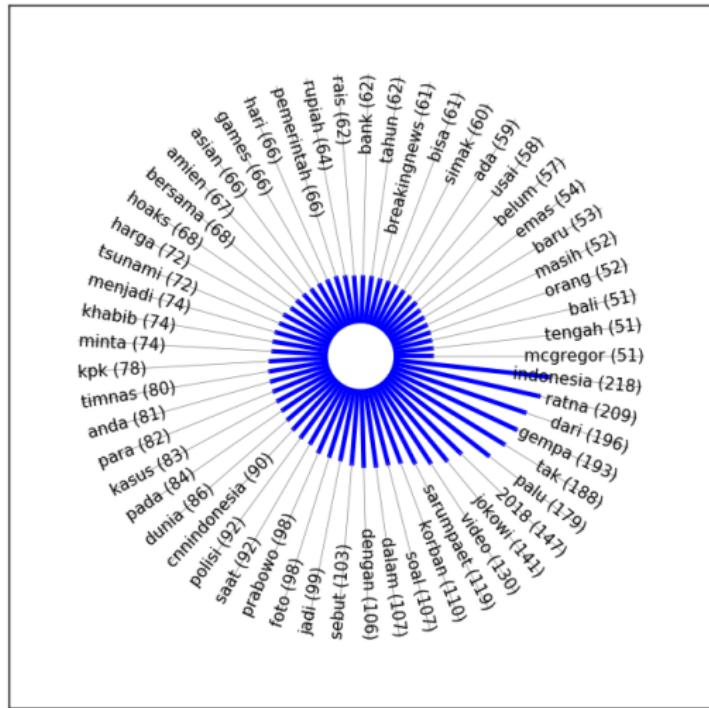
@CNNIndonesia 3,234 tweets.
2018-10-02 17:42:57 to 2018-10-16 10:17:23



30 - 50 frequency

Case 3: Word frequency

@CNNIndonesia 3,234 tweets.
2018-10-02 17:42:57 to 2018-10-16 10:17:23

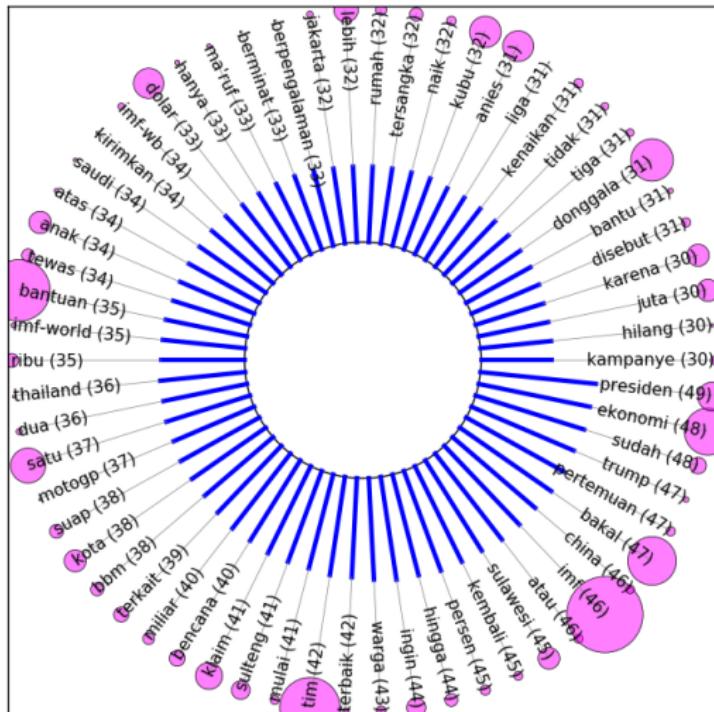


Frequency > 50

Case 3: Word frequency (with retweet size)

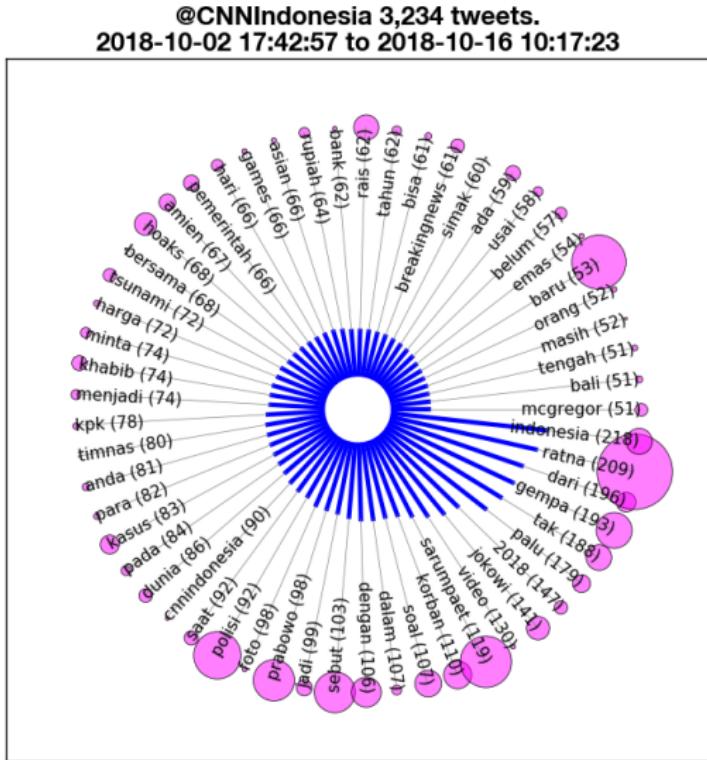
@CNNIndonesia 3,234 tweets.

2018-10-02 17:42:57 to 2018-10-16 10:17:23



30 - 50 frequency

Case 3: Word frequency (with retweet size)



Case 3: Word frequency

Steps:

- ▶ **numpy.load(...tweets data...)**
- ▶ Collect 'words' (**TextBlob**) - several adjustments
- ▶ Each unique 'words' frequency
- ▶ Plot radial bar chart (**rbar**)

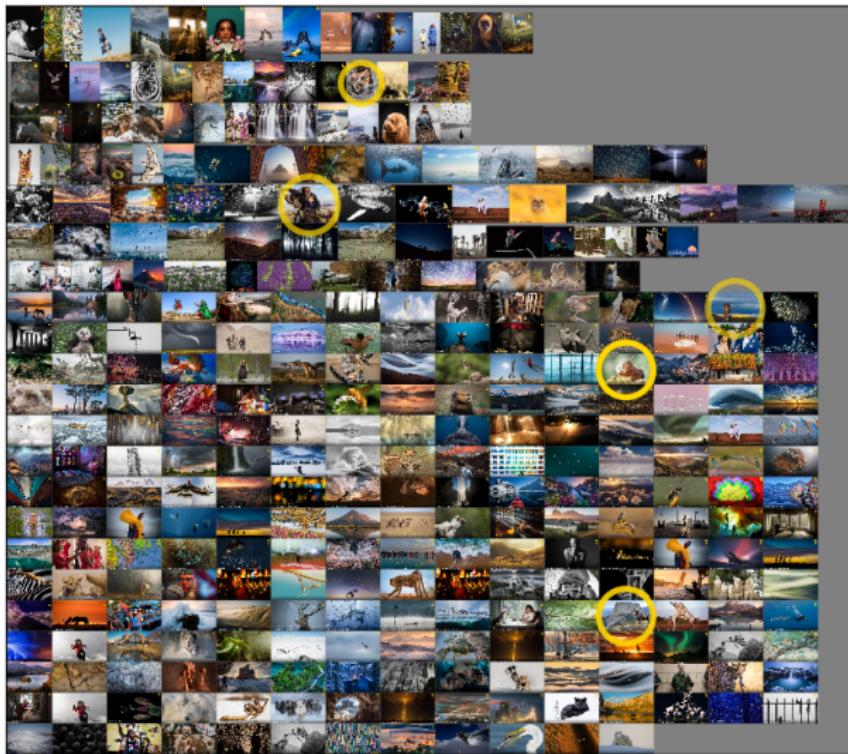
Case 3: Word frequency

Matplotlib and user defined function:

```
def rbar(ax, radii, label, radius, col, \
         bar_width = 5, text_size = 7):
    n = len(radii); indexs = range(n)
    ...
    ax.plot(...base circle...)
    for index in indexs:
        ...
        ax.plot([cx, rx], [cy, ry], \
                lw = bar_width, color = col[index])
        ...
        ax.text(..., ha = ha_split, va = 'center', \
                rotation = degree_split, \
                rotation_mode = 'anchor')
    ax.plot([rx, 1.5*Rx], [ry, 1.5*Ry], \
            lw = 0.5, color = 'gray')
```

Case 4: Image plot

@NatGeo images from 19 Dec 2017 to 19 Sep 2018

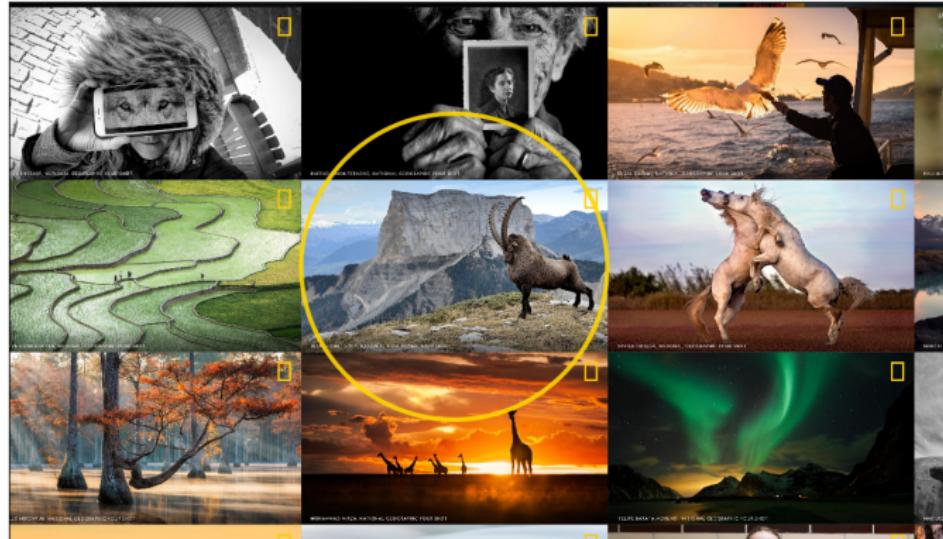


Case 4: Image plot



Zoom in (Image with most Retweets)

Case 4: Image plot



Zoom in (Image with 2nd most Retweets)

Case 4: Image plot

Steps:

- ▶ **numpy.load(...tweets data...)**
- ▶ Collect image urls from tweets
(tweet.entities['media'][0]['media_url'])

Case 4: Image plot

Steps:

- ▶ **numpy.load(...tweets data...)**
- ▶ Collect image urls from tweets
(tweet.entities['media'][0]['media_url'])
- ▶ Save (**urllib.request.urlretrieve**)

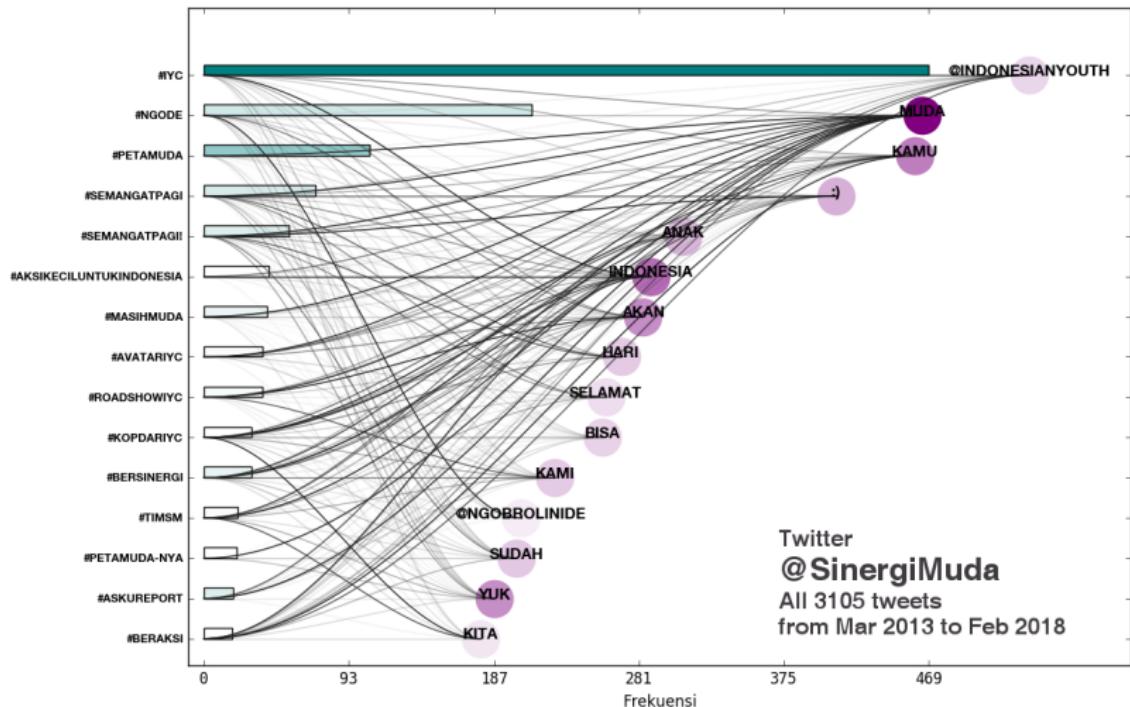
Case 4: Image plot

Steps:

- ▶ **numpy.load(...tweets data...)**
- ▶ Collect image urls from tweets
(tweet.entities['media'][0]['media_url'])
- ▶ Save (**urllib.request.urlretrieve**)
- ▶ Plot each **image** (**image = plt.imshow(plt.imread(image_path))**)
- ▶ Use **image.get_extent()**, and **image.set_extent()** to adjust the position of each image

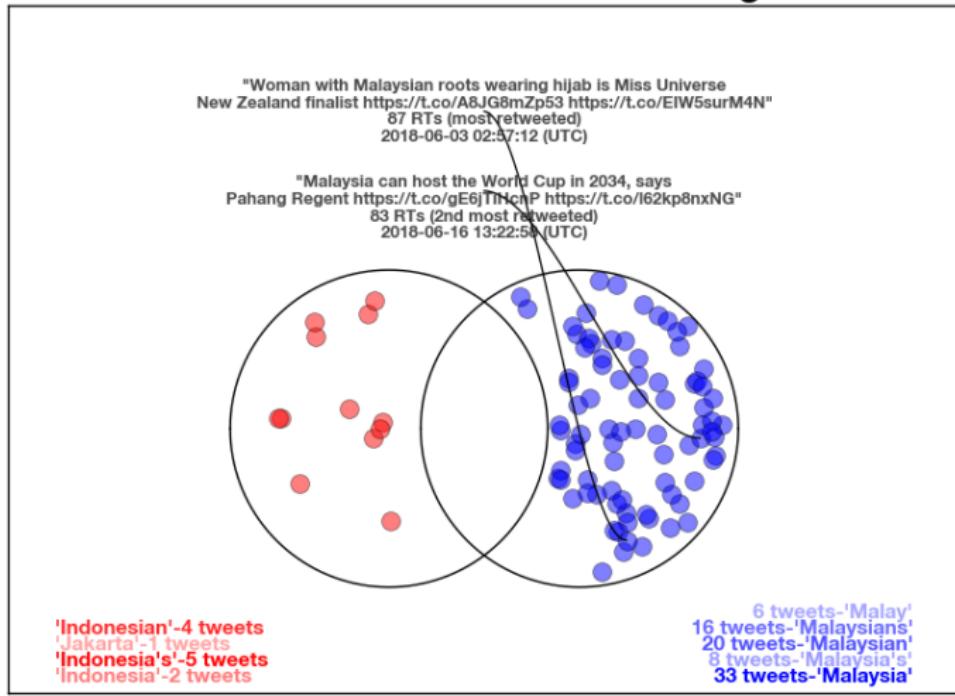
Others: TopWords-TopHashtags Connection

@SinergiMuda



Others: Twitter Venn (different version)

The Star Online - Twitter Venn Diagram



Part-time project: <https://github.com/anbarrief/statistweepy>