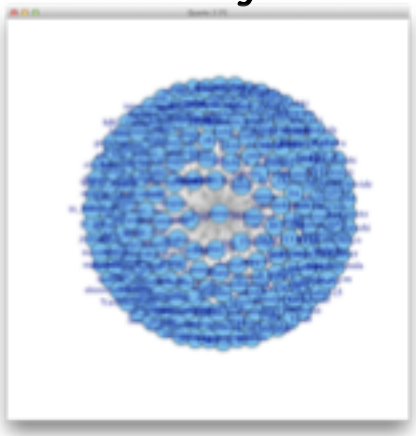
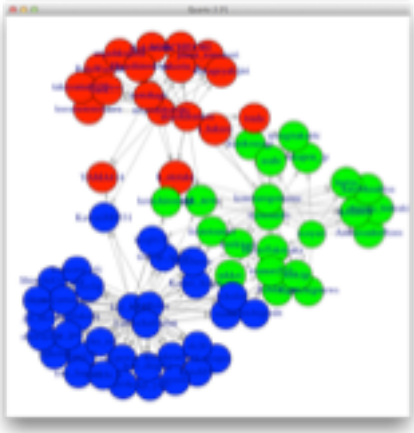
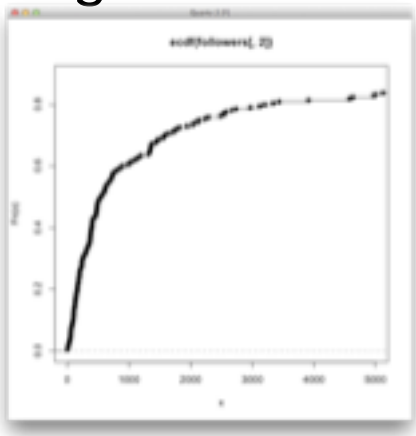
Twitter: コンテンツ解析
Twitter: Content Analysis

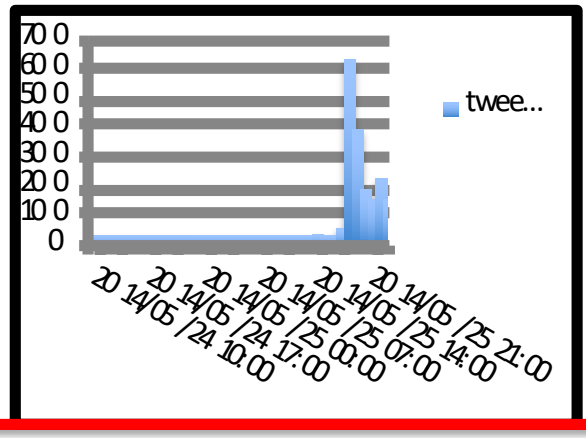
SNSの分析
Analysis of SNS

- ネットワーク分析: 人と人のつながりを分析する。
- Network Analysis: Analyze the relationship among comunities.

- コンテンツ分析: SNSで交換されているコンテンツそのものを分析する。
- Contents analysis: Analyze the content of posted message





PythonをつかってTwitterのAPIを使う準備
Preparation to use Twitter API from Ruby

- pipを使ってrequests_oauthlibをインストール
  # pip3 install requests_oauthlib
  - https://github.com/requests/requests-oauthlib
- Access tokenを取得する

- Install twitter requests_oauthlib using pip
  # pip3 install requests_oauthlib
  - https://github.com/requests/requests-oauthlib
- Get Access token

Pythonを使ってTweetを取得する
Get Tweets using Python

- "BigData"というキーワードが含まれるデータを取得してみる
- Acquire the tweets which a keyword called "BigData" is included in

getBigDataTweet.py

```
#!/usr/bin/env python3

#coding: UTF-8

from requests_oauthlib import OAuth1Session
import json

CONSUMER_KEY = "xxxxxxxxxxxxxxx"
CONSUMER_SECRET = "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
ACCESS_TOKEN = "xxxxxxxxx-xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
ACCESS_TOKEN_SECRET = "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"

twitter = OAuth1Session(CONSUMER_KEY, CONSUMER_SECRET, ACCESS_TOKEN, ACCESS_TOKEN_SECRET)

url = "https://api.twitter.com/1.1/search/tweets.json"
params = {'q': "BigData", 'count': 10, 'lang': "en"}
req = twitter.get(url, params=params)
if req.status_code == 200:
    tweets = json.loads(req.text)
    for tweet in tweets['statuses']:
        print(tweet['text'].replace('¥n', ' '))
else:
    print("ERROR: %d" % req.status_code)
```

※ もし、日本語のツイートを収集したい場合は、"en"を"ja"で置き換える。
※ If you want to get Japanese tweets, "en" must be replaced by "ja"

Pythonを使ってTweetを取得する
Get Tweets using Python

- streaming APIを使ってデータを収集
  Get tweets using streaming API

```
% ./getSample.py 10
```

getSample.py

```
#!/usr/bin/env python3

#coding: UTF-8

from requests_oauthlib import OAuth1Session
import json
import sys

CONSUMER_KEY = "xxxxxxxxxxxxxxx"
CONSUMER_SECRET = "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
ACCESS_TOKEN = "xxxxxxxxx-xxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
ACCESS_TOKEN_SECRET = "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"

twitter = OAuth1Session(CONSUMER_KEY, CONSUMER_SECRET, ACCESS_TOKEN, ACCESS_TOKEN_SECRET)

url = "https://stream.twitter.com/1.1/statuses/sample.json"
req = twitter.post(url, stream=True)
if req.status_code == 200:
    count = 0;
    for tweet in req.iter_lines():
        try:
            tweet = json.loads(tweet)
        except:
            continue
        if "text" in tweet:
            print(tweet['text'].replace('¥n', ' '))
            count = count + 1
            if count >= int(sys.argv[1]):
                break
else:
    print("ERROR: %d" % req.status_code)
```

Pythonを使ってTweetを取得する
Get Tweets using Python

- 下記のURLを参考。/ Document of Filter is available at the following URL
  - https://developer.twitter.com/en/docs/tweets/filter-realtime/api-reference/post-statuses-filter.html

```
% ./getSampleWord.py 10 BigData
```

getSampleWord.py

```
#!/usr/bin/env python3

#coding: UTF-8

from requests_oauthlib import OAuth1Session
import json
import sys

CONSUMER_KEY = "y1DiJ0IrRl6mFiYsRXA8ty6rp"
CONSUMER_SECRET = "ggSPOU8jk7OYbBuuwuLsvms0fpt0isk2jHQonlLBUFhKuS5sUO"
ACCESS_TOKEN = "107277720-0EFhkkq3GoxOT1nDDEyBZMMNOHaGqo3KGC6UM9HS"
ACCESS_TOKEN_SECRET = "TEtK8b6wcQYRREn8ywVYH98IArdMQzsfjjNdiORSjiJH7"

twitter = OAuth1Session(CONSUMER_KEY, CONSUMER_SECRET, ACCESS_TOKEN, ACCESS_TOKEN_SECRET)

url = "https://stream.twitter.com/1.1/statuses/filter.json"
req = twitter.post(url, stream=True, data={"track": sys.argv[2]})
if req.status_code == 200:
    count = 0;
    for tweet in req.iter_lines():
        try:
            tweet = json.loads(tweet)
        except:
            continue
        if "text" in tweet:
            print(tweet['text'].replace('¥n', ' '))
            count = count + 1
            if count >= int(sys.argv[1]):
                break
else:
    print("ERROR: %d" % req.status_code)
```

位置や言語を指定したTweetの取得
Get Tweets with location and language

- フィルターに位置(location)を設定する。/ Set location to filter.
- CSVにして出力する。/ output as CSV

```
% ./getSampleLoc.py 1000000 122.93361,20.42528,153.98639,45.55722 ja > all.csv
```

getSampleLoc.py

```
#!/usr/bin/env python3

#coding: UTF-8

from requests_oauthlib import OAuth1Session
import json
import sys
import csv

CONSUMER_KEY = "y1DiJ0IrRl6mFiYsRXA8ty6rp"
CONSUMER_SECRET = "ggSPOU8jk7OYbBuuwuLsvmsOfpt0isk2jHQonlLBUFhKuS5sUO"
ACCESS_TOKEN = "107277720-0EFhkkq3GoxOT1nDDEyBZMMNOHaGqo3KGC6UM9HS"
ACCESS_TOKEN_SECRET = "TEtK8b6wcQYRREn8ywVYH98IArdMQzsfjjNdiORSjiJH7"

twitter = OAuth1Session(CONSUMER_KEY, CONSUMER_SECRET, ACCESS_TOKEN, ACCESS_TOKEN_SECRET)
csvout = csv.writer(sys.stdout)

url = "https://stream.twitter.com/1.1/statuses/filter.json"
req = twitter.post(url, stream=True, data={"locations": sys.argv[2], "language": sys.argv[3]})
if req.status_code == 200:
    count = 0;
    for tweet in req.iter_lines():
        try:
            tweet = json.loads(tweet)
        except:
            continue
        if tweet['geo'] == None:
            continue
        if tweet['geo']['type'] != "Point":
            continue
        if "text" in tweet:
            csvout.writerow([tweet['geo']['coordinates'][0], tweet['geo']['coordinates'][1], tweet['text'].replace('¥n', ' ')])
            count = count + 1
            if count >= int(sys.argv[1]):
                break
else:
    print("ERROR: %d" % req.status_code)
```

Rubyを使ってTwitterのAPIを使う準備
Preparation to use Twitter API from Ruby

- gemを使ってtwitter gemをインストール(Version 6を入れる)
  # gem install twitter
  - ドキュメント: http://rdoc.info/gems/twitter
- Access tokenを取得する

- Install twitter gem using gem (Version 6)
  # gem install twitter
  - Document: http://rdoc.info/gems/twitter
- Get Access token

Rubyを使ってTweetを取得する
Get Tweets using Ruby

- "BigData"というキーワードが含まれるデータを取得してみる
- Acquire the tweets which a keyword called "BigData" is included in

getBigDataTweet.rb

```
#!/usr/bin/env ruby
require 'twitter'

client = Twitter::REST::Client.new { |config|
  config.consumer_key        = "$$API_Key$$"
  config.consumer_secret     = "$$API_secret$$"
  config.access_token        = "$$Access_token$$"
  config.access_token_secret = "$$Access_token_secret$$"
}

client.search("BigData", :lang => "en").collect { |tweet|
  p tweet.full_text
}
```

※ もし、日本語のツイートを収集したい場合は、"en"を"ja"で置き換える。
※ If you want to get Japanese tweets, "en" must be replaced by "ja"

Rubyを使ってTweetを取得する
Get Tweets using Ruby

- streaming APIを使ってデータを収集
  Get tweets using streaming API
  % ./getSample.rb 1000

getSample.rb

```
#!/usr/bin/env ruby
require 'twitter'

client = Twitter::Streaming::Client.new { |config|
 config.consumer_key        = "$$API_Key$$"
  config.consumer_secret     = "$$API_secret$$"
  config.access_token        = "$$Access_token$$"
  config.access_token_secret = "$$Access_token_secret$$""
}

count = 0
client.sample { |tweet|
  if tweet.is_a?(Twitter::Tweet)
    p tweet.full_text
    count = count + 1
  end
  break if ARGV[0].to_i > 0 && count >= ARGV[0].to_i
}
```

  - Twitter::Tweetオブジェクトの形式 / Twitter::Tweet Object
    http://www.rubydoc.info/github/sferik/twitter/Twitter/Tweet

キーワードを含むTweetのみを取得
Get tweets which includes keyword

- "client.sample"の部分を"client.filter()"にして、フィルターを設定する。
  Replace "client.sample" to "client.filter()" and set filter.
- 下記のURLを参考。/ Document of Filter is available at the following URL
  - https://developer.twitter.com/en/docs/tweets/filter-realtime/api-reference/post-statuses-filter.html
  - % ./getSampleWord.rb 10 BigData

getSampleWord.rb

```
#!/usr/bin/env ruby
require 'twitter'

client = Twitter::Streaming::Client.new { |config|
 config.consumer_key        = "$$API_Key$$"
  config.consumer_secret     = "$$API_secret$$"
  config.access_token        = "$$Access_token$$"
  config.access_token_secret = "$$Access_token_secret$$""
}

count = 0
client.filter(:track => ARGV[1]) { |tweet|
  if tweet.is_a?(Twitter::Tweet)
    p tweet.full_text
    count = count + 1
  end
  break if ARGV[0].to_i > 0 &&count >= ARGV[0].to_i
}
```

位置を指定したTweetの取得
Get Tweets with location

- フィルターに位置(location)を設定する。/ Set location to filter.
  - https://dev.twitter.com/docs/streaming-apis/parameters#locations
- CSVにして出力する。/ output as CSV

```
% ./getSampleLoc.rb 1000000 122.93361,20.42528,153.98639,45.55722 ja > all.csv
```

getSampleLoc.rb

```
#!/usr/bin/env ruby
require 'twitter'
require 'CSV'

client = Twitter::Streaming::Client.new { |config|
 config.consumer_key        = "$$API_Key$$"
  config.consumer_secret     = "$$API_secret$$"
  config.access_token        = "$$Access_token$$"
  config.access_token_secret = "$$Access_token_secret$$""
}
count = 0
client.filter(:locations => ARGV[1] , :language => ARGV[2]) { |tweet|
  if tweet.is_a?(Twitter::Tweet) && tweet.geo.coordinates.size == 2
    csv_string = CSV.generate(:force_quotes=>true) { |csv|
      csv << [tweet.geo.coordinates[0], tweet.geo.coordinates[1],
              tweet.user.screen_name, tweet.created_at, tweet.full_text.gsub(/(¥n)/, '¥¥¥n')]
    }
    puts csv_string
    count = count + 1
  end
  break if ARGV[0].to_i > 0 && count >= ARGV[0].to_i
}
```

RでGoogle Mapsを使う
Use Google Maps from R

- RgoogleMaps (http://cran.r-project.org/web/packages/RgoogleMaps/)
  - RからGoogleMapsを使えるようにするためのパッケージ
    This is a package to use GoogleMaps from R

- RでRgoogleMapsを使えるようにする / Install RgoogleMaps package

```
% R
> install.packages("RgoogleMaps")
```

- 実際にプロットしてみる

```
% R
> library(RgoogleMaps)
> lat <- c(35.388184, 35.386287)
> lon <- c(139.427367,  139.425976)
> sfc <- data.frame(LAT=lat, LON=lon)
> Map <- GetMap(c(35.388184, 139.427367),
+ destfile="map.png",zoom=16, sensor="false", hl="ja")
> PlotOnStaticMap(Map, lat=sfc[,1], lon=sfc[,2], col="black")
```

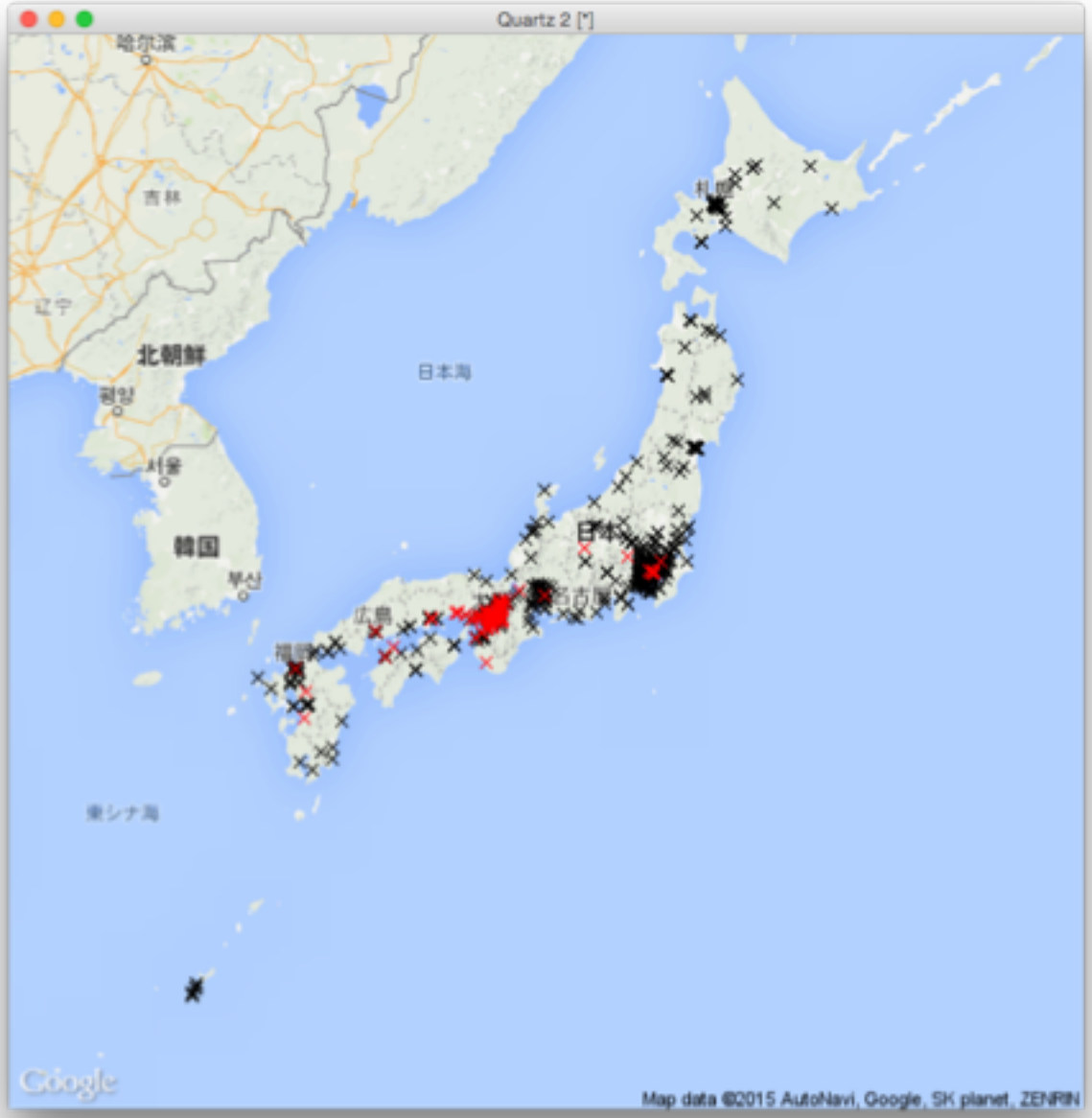
「マクドナルド」の呼び方は?
How to call McDonalds?

- 関東では「マック」、関西では「マクド」と言われているが本当か?
- In Tokyo McDonalds is called as "マック". But in Kansai, it's "マクド"

```
% grep "マック" all.csv | grep -v "マックス" > mac.csv
% grep "マクド" all.csv | grep -v "マクドナルド" > macdo.csv
% R
> library(RgoogleMaps)
> mac <- read.csv("mac.csv", header=F)
> macdo <- read.csv("macdo.csv", header=F)
> Map <- GetMap(c(35.65892, 136.95665), destfile="map.png",zoom=5,
sensor="false", hl="ja")
> PlotOnStaticMap(Map, lat=mac[,1], lon=mac[,2], pch=4)
> PlotOnStaticMap(Map, lat=macdo[,1], lon=macdo[,2], pch=4, col="red",
add=TRUE)
```

「マクドナルド」の呼び方は?
How to call McDonalds?

- 本当っぽい。
- Correct



「山手線」とつぶやく人は?
Where are peoples who tweet "山手線"

- 「山手線」とつぶやく人はどこにいるのか?
- Plot the tweets includes "山手線"

```
% grep "山手線" all.csv > yamate.csv
% R
> library(RgoogleMaps)
> yamate <- read.csv("yamate.csv", header=F)
> Map <- GetMap(c(35.68639, 139.73333),
+ destfile="map.png",zoom=12,
+ sensor="false", hl="ja")
> PlotOnStaticMap(Map,
+ lat=yamate[,1], lon=yamate[,2],
+ pch=16, col="red")
```

「AKB」とつぶやく人は?
Where are peoples who tweet "AKB"

- 「AKB」とつぶやく人はどこにいるのか?
- Plot tweets includes "AKB"

```
% grep "AKB" all.csv > akb.csv
% R
> library(RgoogleMaps)
> akb <- read.csv("akb.csv", header=FALSE)
> Map <- GetMap(c(35.68639, 139.73333),
destfile="map.png",zoom=12, sensor="false",
hl="ja")
> PlotOnStaticMap(Map, lat=akb[,1],
lon=akb[,2], pch=16, col="red")
```

AKB48「次の足跡」2014年5月24日（土）開催 劇場盤発売記念大写真会（東京ビッグサイト）詳細決定！！

an event

「AKB」とつぶやく人は?
Where are peoples who tweet "AKB"

- 「AKB」とつ
- Plot tweets

```
% grep "AKB" a
% R
> library(Rgoo
> akb <- read.
> Map <- GetMa
destfile="map.
hl="ja")
> PlotOnStatic
lon=akb[,2], p
```

本当は日本全国どこでもつぶやかれていて、
それなりにちゃんとした解析が必要。
There are many peoples in Japan