

Python Setup and Data Analysis Model

The structure for data analysis model is as follows:

```
-- data analysis

-- raw_data (store downloaded json files from the ehims admin page)

    -- channel-***.json (channel information, including messages)

    -- survey_results.json

-- results (store analysis results)

    -- channel-***.txt (basic description of networks)

    -- channel-***.csv (including 2 sheets of networks analysis results)

    -- channel-***.png (visualized BFS of messages network)

    -- survey_results.csv (summary of surveys)

-- group-results.py (transform survey format data from .json to .csv)

-- survey-results.py (perform networks analysis on messages and gather stats on channel)
```

Those codes are written in Python3. Generally speaking, Python comes with OS X, so it is easy to check python version by typing *python --version* into terminal. If the terminal prints *Python2*, you can install the latest version of python 3 by inputting the following command:

```
brew install python3
```

Once python is installed successfully, having a package manager will make things easier and pip is a preferred one. Installing pip by entering the following command in a terminal.

```
curl -O https://raw.githubusercontent.com/pypa/pip/master/contrib/get-pip.py

python get-pip.py
```

Before we start to run python codes, it is necessary to install some packages at first. Copy and paste the following commands in a terminal:

```
pip install networkx

pip install matplotlib

pip install pandas
```

After python and packages are set up, it is time to perform some analysis. Open the file ‘*data analysis/group-results.py*’ and modify Line 15 to the id of the channel which you want to analyze.

```
1 import json
2 import networkx as nx
3 import matplotlib.pyplot as plt
4 import pandas as pd
5 from matplotlib import cm
6 import itertools
7
8 '''
9 perform basic networks analysis on messages
10 gather basic stats on channel
11 generate a .xlsx file consisted of 2 sheets (participants, messages) and a .txt file with basic info about this channel
12 '''
13
14
15 channel = '5d964cec26c1594270c72e08' # modify this line to process different channel messages
16 file_path = r'raw_data/channel-{}.json'.format(channel)
17 group_stats = r'results/channel-{}.txt'.format(channel)
18 group_info = r'results/channel-{}.xlsx'.format(channel)
19 plt_path = r'results/channel-{}.png'.format(channel)
```

Open a terminal and change into the same directory where the code is. Run the code by entering the following command in a terminal. A .xlsx file consisted of 2 sheets (participants, messages) , a .txt file with basic description about this channel and a .png file of the visualized BFS (Breadth-first Traversal) for this networks will be generated.

```
python group-results.py
```

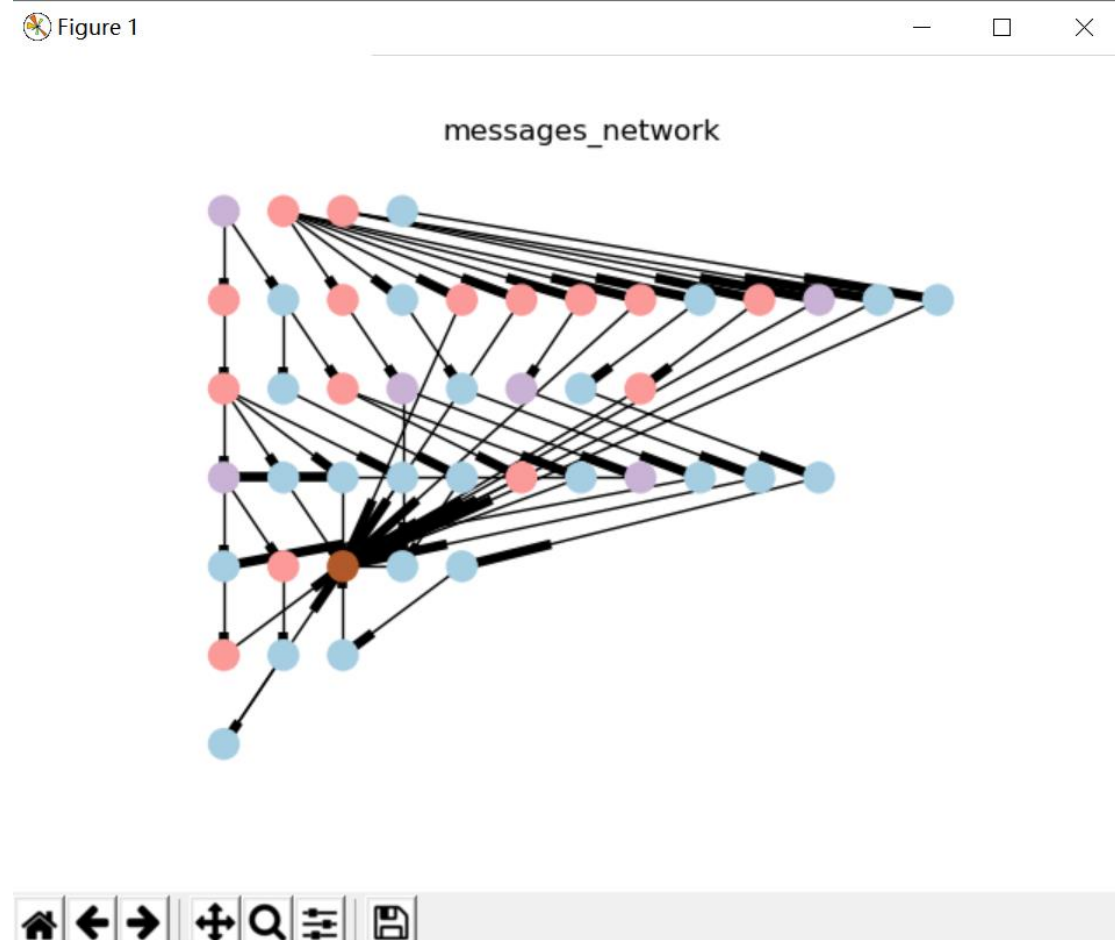


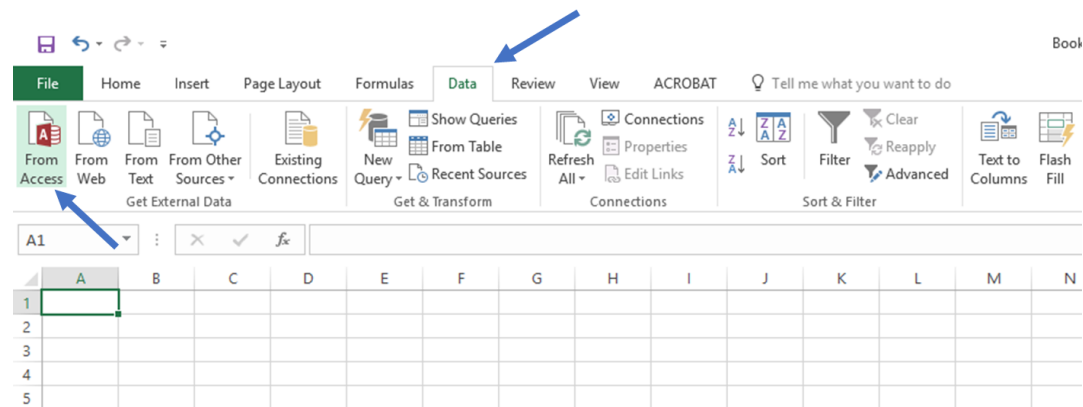
Figure 1 BFS of messages network. (Different colors represent different authors)

As for the '*data analysis/survey-results.py*' file, directly run it without any modification and a .csv file (delimiter is tab) will be generated.

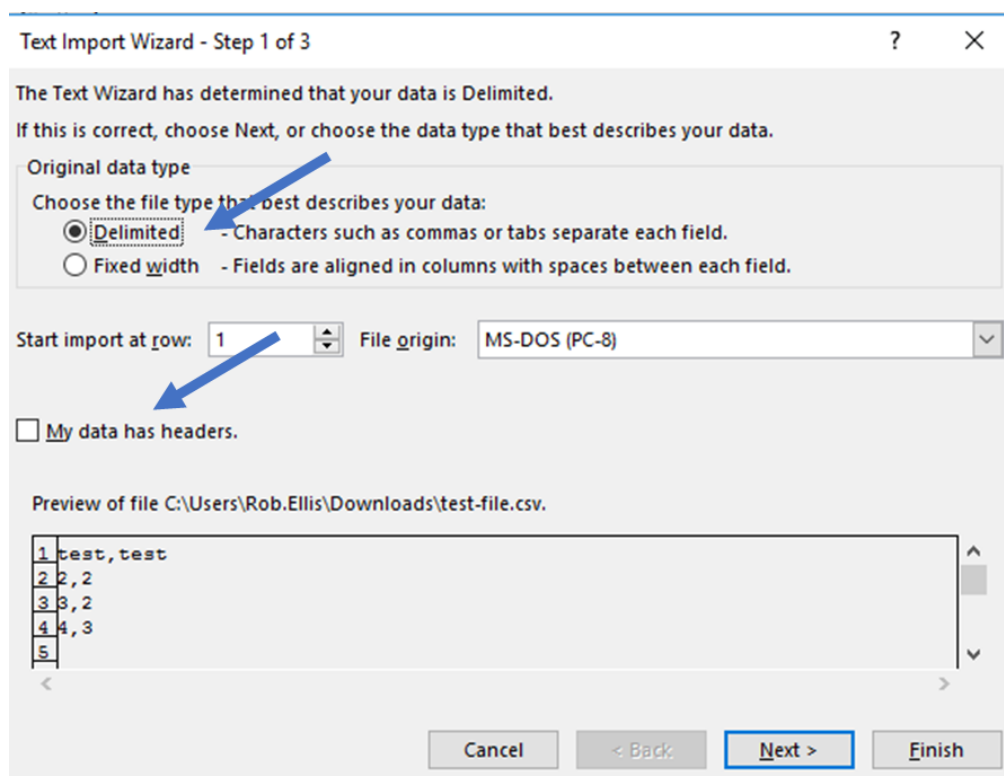
```
python survey-results.py
```

Since the delimiter is tab, it is convenience to open in Sublime. If you want to open it in Excel, please follow the following steps:

1. Select 'Data' on the ribbon, and then 'From Text'.



2. Browse for the *.csv file you want to open, and click 'Import'. In the Text import wizard, ensure the 'Delimited' option is selected. Click Next.



3. In the delimiters section, tick 'Tab'.

Text Import Wizard - Step 2 of 3 ? ×

This screen lets you set the delimiters your data contains. You can see how your text is affected in the preview below.

Delimiters

☐ Tab

☐ Semicolon

☒ Comma

☐ Space

☐ Other:

☐ Treat consecutive delimiters as one

Text qualifier: ▼