Individual Project Report

Indian Political System Network Analysis

Pranav Dixit

013838058

Abstract – This report details the work done to understand the network structure between Political Parties of India. The study also aimed to find the polarization of people in the political sphere using the network clustering algorithms.

Introduction :

The Indian Political system is a complex structure involving two main national parties and over 2000 small parties. The Bhartiya Janata Party or the BJP is a right-wing party; along with the support of other small parties is the current ruling party. The Indian National Congress Party is a center-left party which is the main opposition party. The general population of India are very zealous about politics and show active participation in the political system. The advent of social media has provided the people a platform to participate even more profoundly. Social media sites like Twitter play an important role in shaping the political system of a nation. It also provides an excellent tool for analysis.

This study aims to understand the network structure between the two main political parties of India using Twitter data. The study also aims to understand the flow information between different political communities using the retweet network. Finally, the study aims to find the polarity in the retweet network using the network clustering algorithms.

Data Understanding :

1. Twitter platform:

Twitter data is used for the study. Twitter is a microblogging platform where users can post micro blogs called tweets which are visible to all the user. The maximum length of a tweet is 280 characters. Twitter users consists of almost all import personalities on planet whether they be in Movies, Politics or Sports. Thus Twitter becomes an official source of first-hand information for the people and a platform for important people to make press releases.

Along with tweeting text, users get a lot of options to share information. Retweet is a way where a user posts another person's tweet on his timeline. Hashtags provide a way to tag your tweet with a specific topic. Tweets can be searched on the platform using the hashtags. Users can also mention other users in their tweets using the user mentions. Twitter also provides options to user to add media files in their tweets like photos, videos or gifs.

Twitter provides apis for developers to access their data. Twitter apis can be directly called using a third-party software or url-lib library. But a client library called as 'Tweepy' provides an easy way to access the apis from python code. A typical code snippet to get tweets for a query looks as follows-

```
api = tweepy.API(auth)
tweets = api.search(q='xyz',count=100)
```

A single search api request can give upto a 100 tweets. Standard users can hit only 150 requests in 15 mins which causes some limitations to large data collection.

2. Data attributes:

Each tweet returned is in JSON format. The tweet json, along with the actual tweet text, contains lot of meta data along with it. Some of the import attributes are-

1. created_at – Stores the timestamp at which the tweet was tweeted.
2. id – A unique id assigned to each tweet
3. full_text – The tweet text
4. entities – Contains hashtags, symbols, user mentions, urls and media contained in the tweet.
5. user – User information like name, screen_name, description
6. retweeted_status – If the tweet is a retweet then it contains the original tweet json with the original user

Following is the collapsed view of a tweet json –


```
{
  "created_at": "Tue Jul 16 22:15:15 +0000 2019",
  "id": 1151253989931061200,
  "id_str": "1151253989931061250",
  "full_text": "The ocean needs our help – overfishing and other human activity are disrupting marine ecosystems.
  "truncated": false,
  "display_text_range": [...],
  "entities": {...},
  "extended_entities": {...},
  "source": "<a href=\"https://www.hootsuite.com\" rel=\"nofollow\">Hootsuite Inc.</a>",
  "in_reply_to_status_id": null,
  "in_reply_to_status_id_str": null,
  "in_reply_to_user_id": null,
  "in_reply_to_user_id_str": null,
  "in_reply_to_screen_name": null,
  "user": {"id": 8161232...},
  "geo": null,
  "coordinates": null,
  "place": null,
  "contributors": null,
  "is_quote_status": false,
  "retweet_count": 25,
  "favorite_count": 94,
  "favorited": false,
  "retweeted": false,
  "possibly_sensitive": false,
  "possibly_sensitive_appealable": false,
  "lang": "en"
}
```


3. Data Collection:

The first aim is to construct a network of people following each major political party in India. To get that the followers of BJP and the Congress party are fetched using the Twitter friendship apis. The twitter apis require either the user id or the screen name of the user whose followers are to be fetched. The BJP and

the Congress party have the screen names ‘BJP4India’ and ‘INCIndia’ which is used in the apis. Following is a code snippet for the same-

```
bjpfollowers = api.followers_ids(screen_name='BJP4India')
congressfollowers = api.followers_ids(screen_name='INCIndia')
```

The apis returns the user id of the followers in the form of list.

To construct the retweet network tweets were captured using the twitter apis. The tweets were captured using two hashtags one was started by the ruling BJP party and the other by the opposition Congress party. The ‘#TripleTalaq’ was started by the BJP after a new law was passed which empowered the Muslim women of India. The ‘#UnnaoKiBeti’ was started by the Congress after a national scandal broke when a girl was raped and she accused a BJP’s member of parliament for the crime. Both the hashtags represent two sides of politics and women rights. One hashtag is regarding women empowerment while other one is about the crimes against women. One hashtag represents a positive aspect of the ruling BJP party while the other hurts the party’s reputation. Both these hashtags seem to be the best candidates for analyzing the retweet network and studying the polarization.

Following is a sample code for fetching tweets using hashtags-

```
tweets = api.search(q="#TripleTalaq")
```

A single api request only returns 15 tweets. Count parameter can be set to get upto 100 tweets. Tweepy also provides Cursor object for pagination which can be used to iteratively hit and fetch more tweets.

Data Preprocessing:

The tweets fetched are stored in the form of string. Python provides map data structure for json object. Only retweets are used for the graph creation. So the data is cleaned and the normal tweets are removed. From the retweets only the retweets’ user’s screen_name and the original user’s screen name are extracted.

Graph Generation and Modeling:

NetworkX library is used for graph generation. NetworkX is a Python library used for graph creation and network analysis. The BJP and the Congress party, for each, have a node in the graph . If a user A follows the BJP party then a edge is added from user A to BJP. Each node is also attached with an attribute ‘support’. If the person only followed the BJP the value of support is set to 1. If he followed both the value is set to 2. If he followed only the Congress the value is set to 3. The support attribute helps in network clustering and visualization. Only 5000 followers of each party are used due to memory and processing limitations.

Retweets are used to generate the retweet graph. If a tweet by user A is retweeted by user B then a edge is added from A to B. The whole network consists of 19134 nodes and 24843 edges. The largest connected

component has 17614 nodes and 23820 edges. As the largest component has more than 89% of the nodes analysis of retweet network is only conducted on the largest connected component.

Cluster Analysis:

Graph analysis is performed using Gephi. Gephi is a popular network analysis and visualization package.

1. Followers network:

Figure 1 show the followers network graph. The blue nodes represent the people following the BJP only. The red nodes represent the people following the Congress party only. The yellow nodes represent the people following both the parties. Looking at the network it is clearly visible that the network is divided into two separate clusters. Out of the 10000 people only 920 are common to both the parties. There is a lot of sparsity between the two communities.

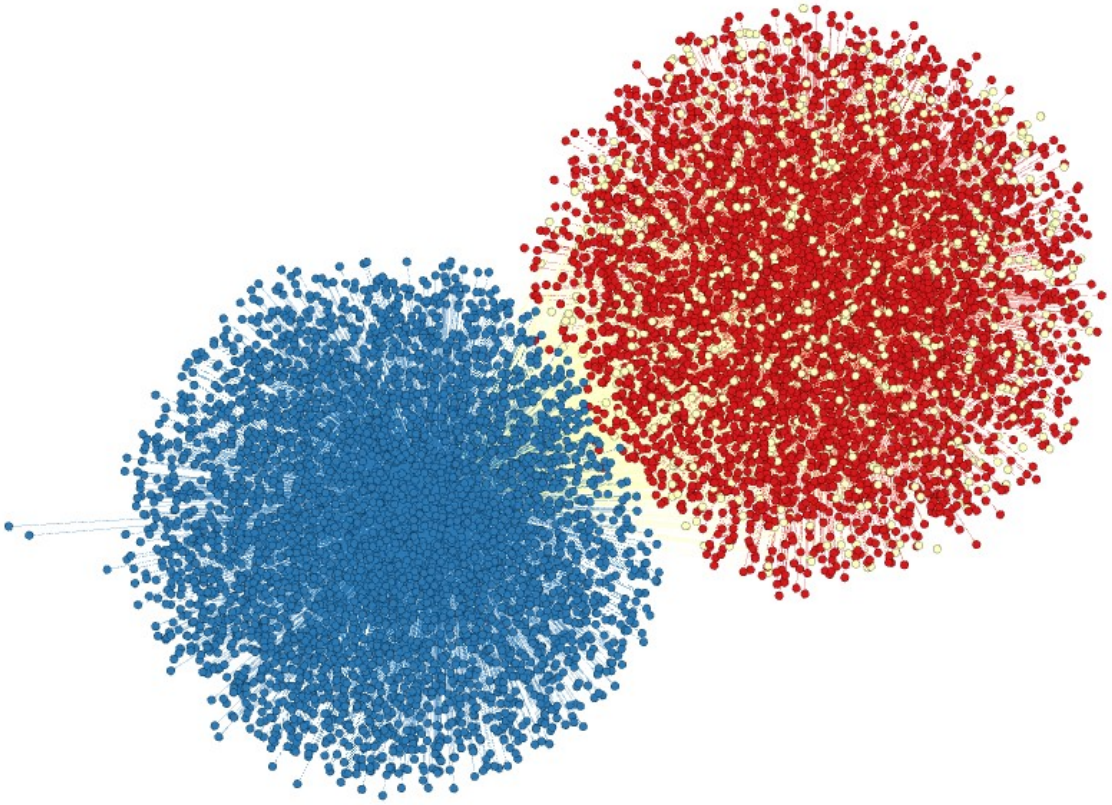

Figure 1: The followers network shown using force-directed graph

2. Retweet network:

The largest connected component of the retweet network consists of 17614 nodes and 23820 edges. This network represents the flow of the tweets between the users. Multiple techniques exist for community detection. These techniques are used to find strong communities inside a network. Some of the most

prominent ones are label propagation algorithm and Modularity measure. This project focuses on using Modularity for community detection.

Modularity is a measure used to understand the structure of graph. It measures the closeness of the communities in the network. Modularity is defined as the fraction of edges that fall within a community minus the fraction of edges in the community if the edges had been distributed randomly in the network. It is given as –

$$Q = \sum_{i=1}^{n} (e_{ii} - a_i^2)$$

$$where\ n = number\ of\ communities$$

$$e_{ii} = the\ fraction\ of\ edges\ between\ two\ communities$$

$$a_i = fraction\ of\ edges\ between\ two\ communities\ for\ ramdon\ distribution$$

The value of score ranges between -1 to +1. A high positive number indicates that the intracommunity edges are high compared to the intercommunity. This shows that the communities are tightly bonded but have weak relations with other communities. A small value of modularity indicates less communal and more integral structure in the network.

Modularity analysis was applied on the retweet network. Figure 2 shows the retweet network with the different communities shaded with different colors.

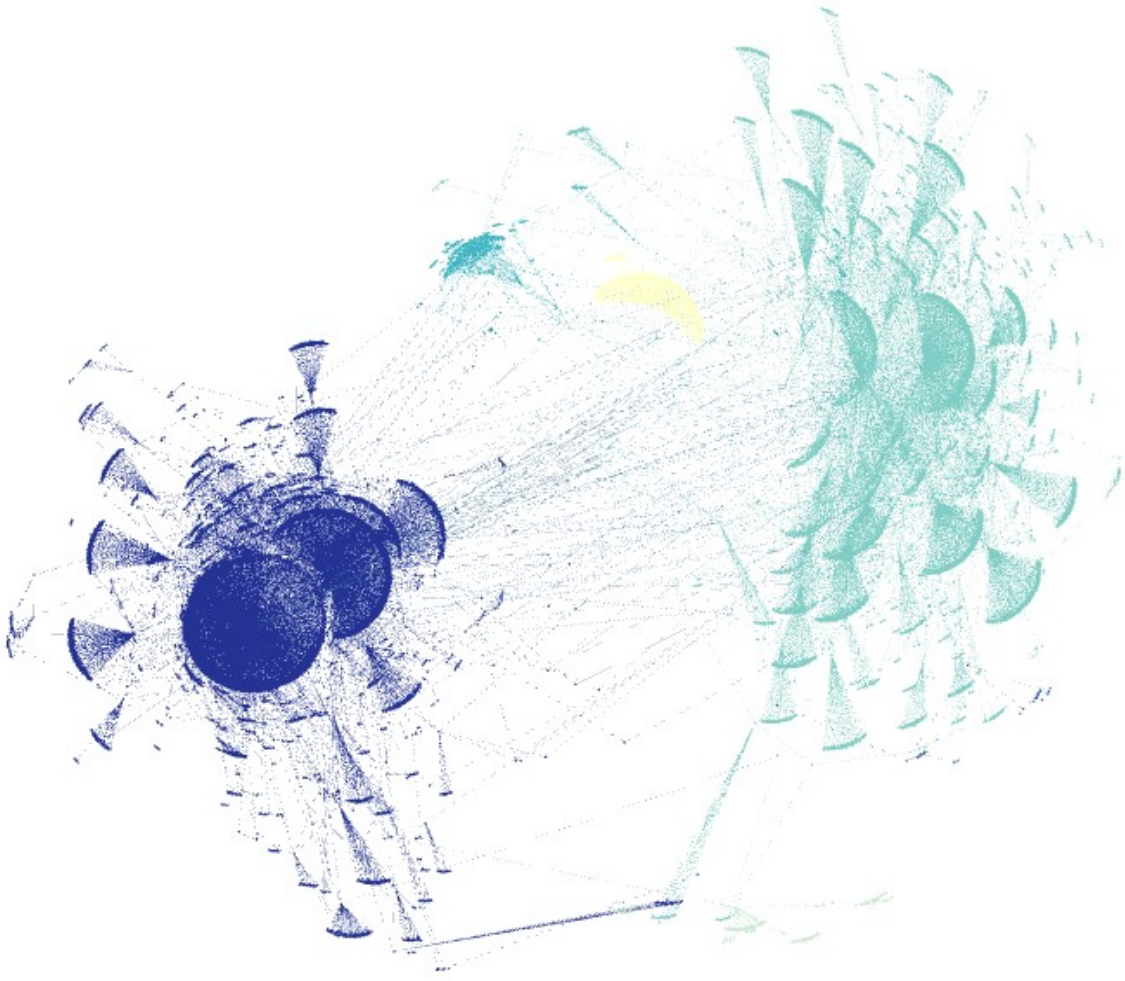

Figure 2:Force-directed graph of the retweet network

Two major communities were detected (shown in blue and green color) with 54.46% and 36.81% of the nodes in each of them respectively. The two communities have sparse connections with each other. The Modularity found was 0.534.

Network	**Modularity**
Retweet	0.534

Conclusion:

The Twitter followers network and retweet network were generated and analyzed. Each network showed clear evidence of strong clustering in to two groups. Analyzing the retweet network in depth revealed that the network shows high modularity. This suggests very weak connections between the two communities as compared to the links inside the communities. The analysis suggests strong polarization of the people in India.