PART III: SENEKAL
POSITIONAL ANALYSIS

Positional analysis is the subfield of network science concerned with the identification of actors or groups of actors who occupy similar positions or roles based on some feature or structure of the network.

Data Source

#Senekal Reply Network - Largest Component

Data Transformation

I exclude observations with any missing data from the feature data set as they are typically not well handled in this clustering algorithm. Additionally, I use a data scaling method called "MinMaxScaler" to standardise the data.

Method

Using pythons scikit-learn library, I present feature based techniques that detect optimal **k-means algorithm** clusters.

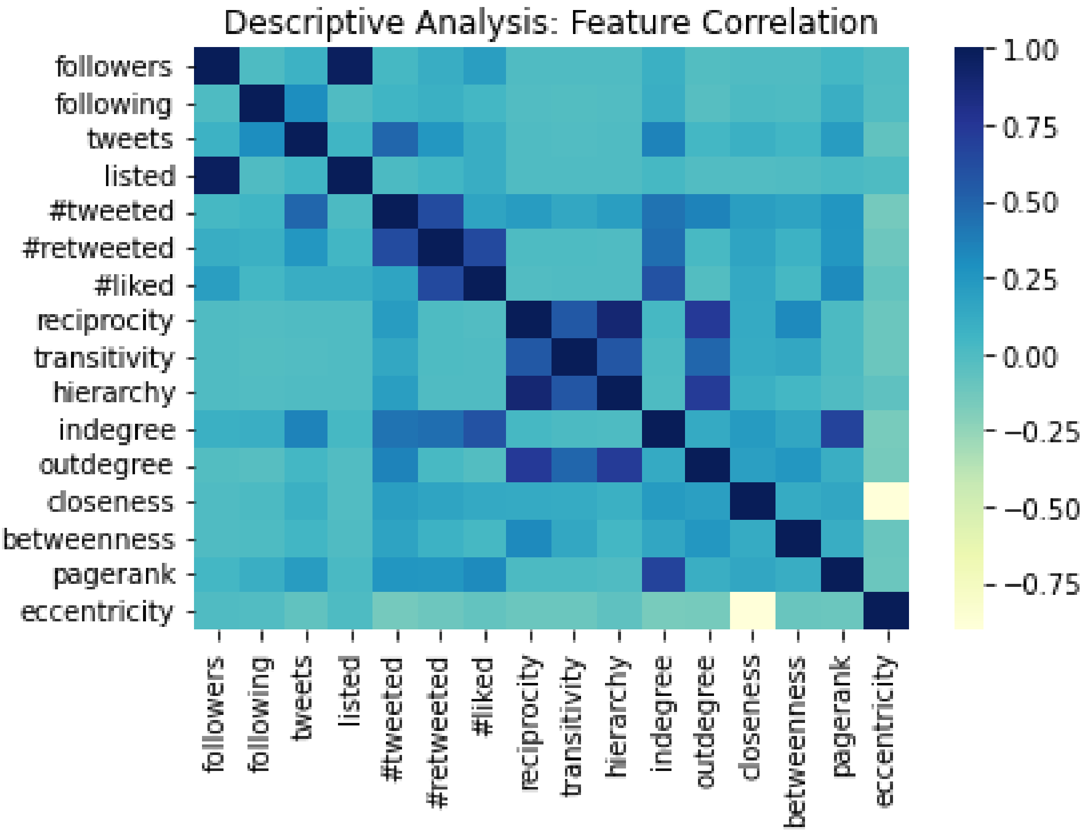
---

1 SELECTION

The 16 features selected for this analysis are detailed in the Part I: Descriptive Analysis. They cover node attributes of expansiveness (whether someone is a social butterfly or wallflower), attractiveness (whether someone is popular or unpopular), as well as relevance (whether someone is locally important or important gloably). For example, a person can be generally active on social media with a high number of tweets, while another person can be active with a high number of tweets at one particular point in time discussing one particular issue. Both of these traits are captured by the features: *tweets* (global) and *#tweeted* (local).

Notable observations from the correlation matrix:

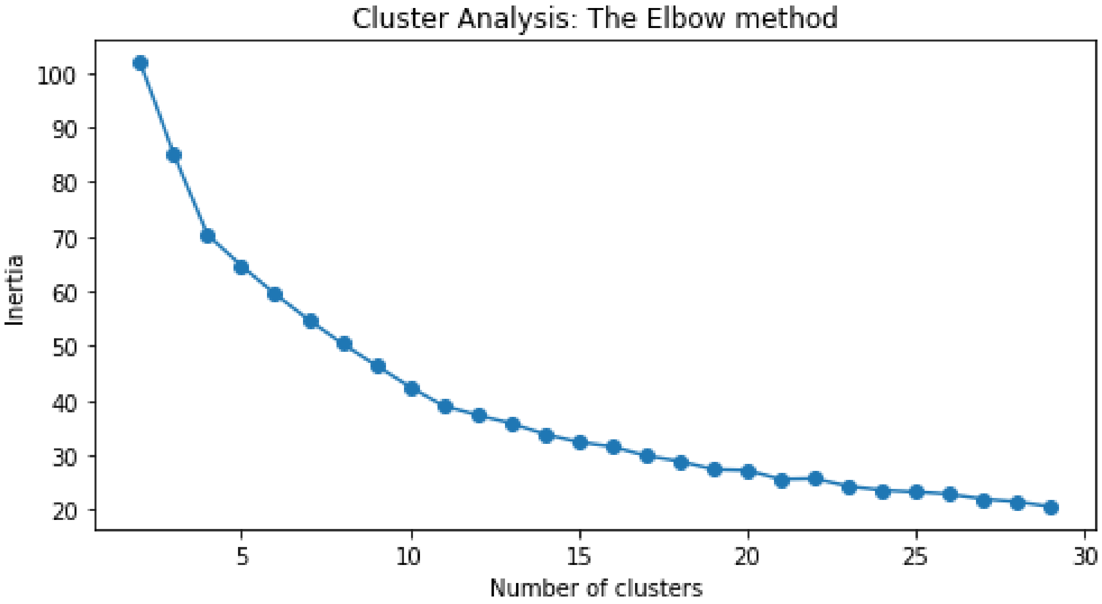
- being *listed* on a public list is positively correlated with the number of *followers*.
- *#tweeted*, *#liked* and *#retweeted* are positively correlated with the *indegree* of tweets being replied to as well as the *pagerank* likelihood of being connected to other high rank positions.
- *reciprocity*, *transitivity*, *hierarchy* are all positively correlated with the *outdegree* of replying to tweets.



2 DETECTION

Using k-means clustering techniques, I detect the optimal number of clusters to fit the network data.

2.1 The Elbow Method

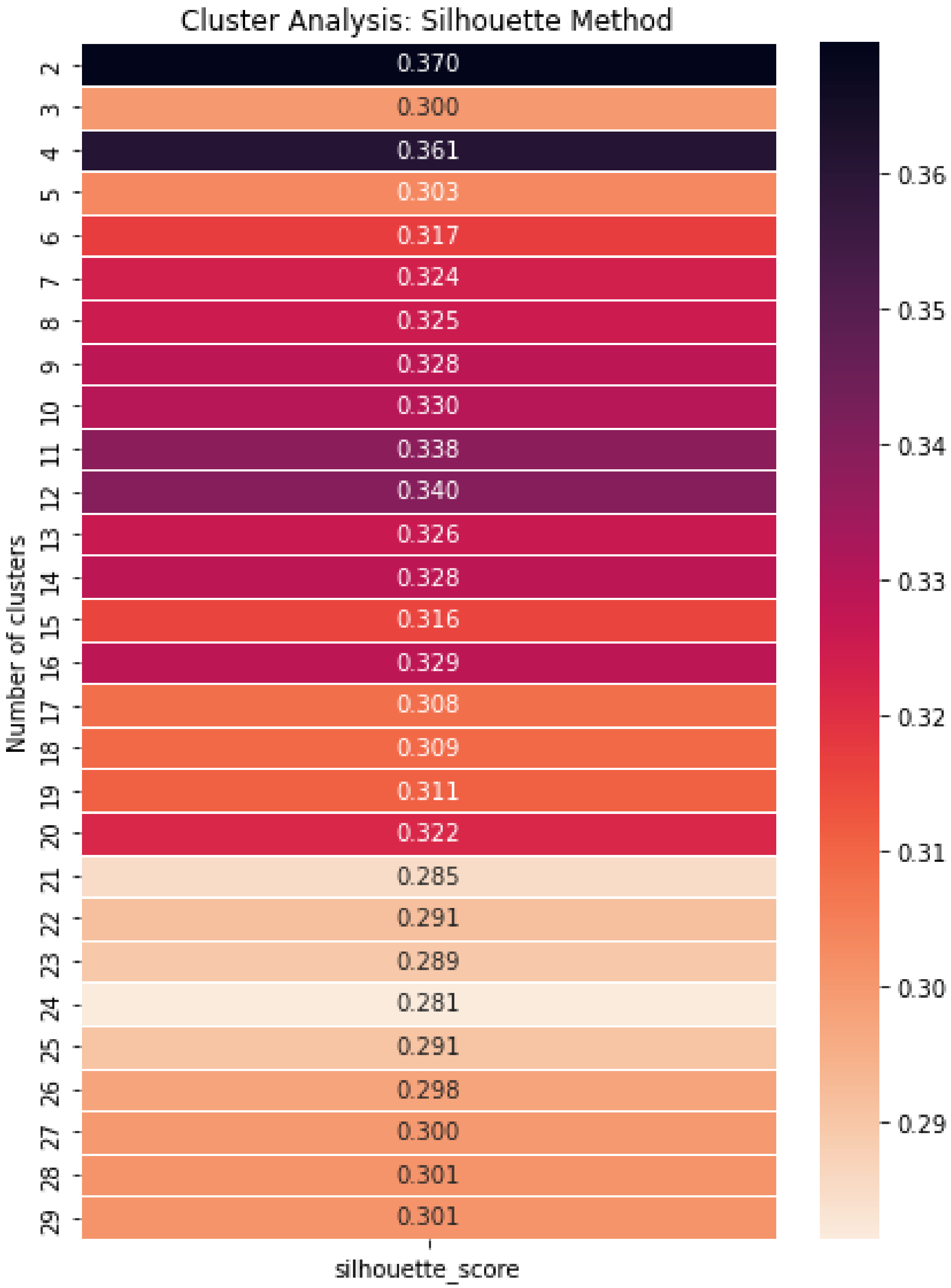


KneeLocator

Since it is not visibly clear, just by looking, which point of the curve is the knee or point of the maximum curvature. I run the *kneelocator* function and find the optimum number of clusters at k-cluster=11.

```
11
```

2.2 The Silhouette Method



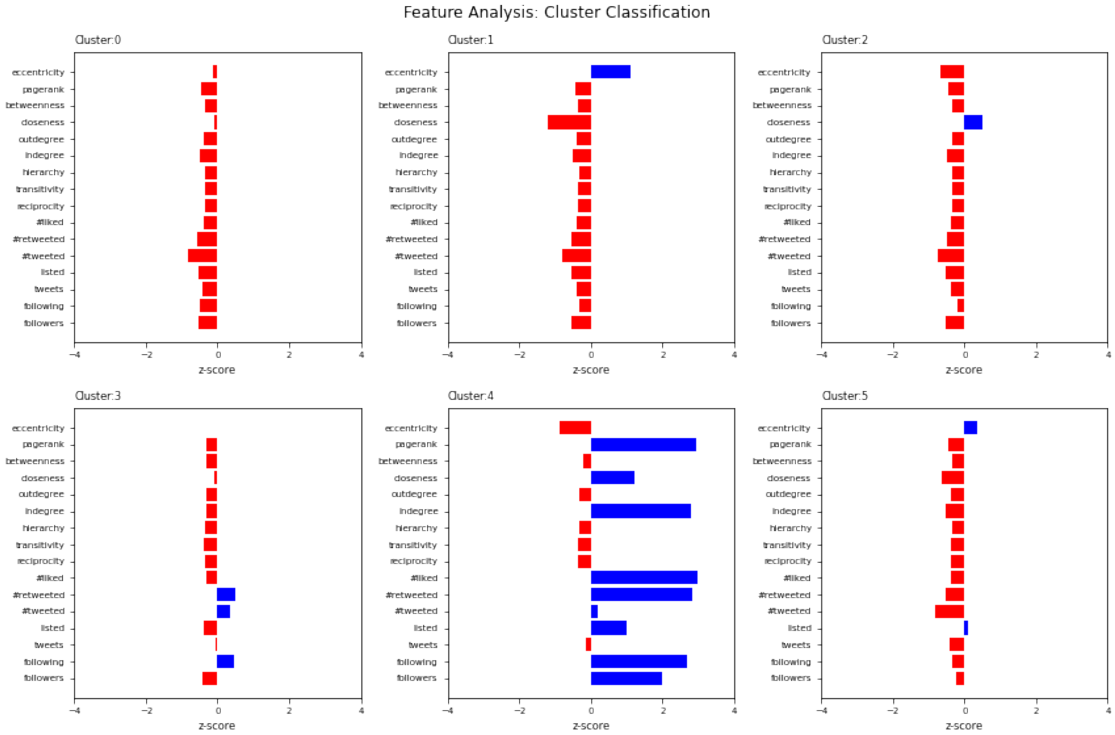
The highest silhouette score obtained for k-clusters between 2 and 30 is 0.370 at k-cluster=2.

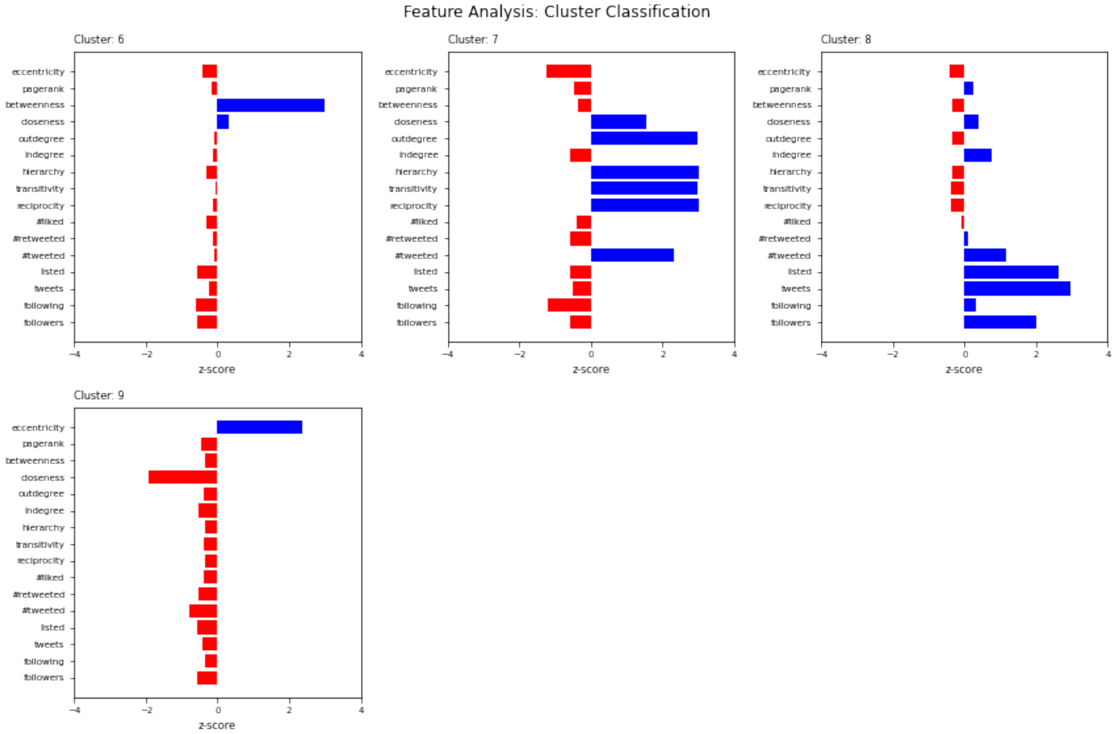
3 CLASSIFICATION

Using the elbow and silhouette method as a guide, I narrow the range of k-clusters and carefully examine the partitions that display the most consisent and meaningful result. I find k-cluster=7 and k-cluster=10 to have the best fit for this network data.

I classify the different clusters by plotting the mean scores of each feature in a cluster and comparing them to the mean scores of features in other clusters. Distinct clusters are identified by the features that display the largest deviation from the mean score.

3.1 Mean Score Plots





At k-cluster=10, 5 distinct clusters are detected. The remaining 5 exhibit non-distinct or low value features. Non-disinct clusters are therefore labelled as **Minor** types, while the 5 distinct clusters are labelled as such:

1. **Observer** -> periphery users that are far away from the centre of the discourse and have little participation or involvement.
2. **Spreader** -> bridge users that are connected to users in various positions and can easily spread information across the network.
3. **Activator** -> active users that reply to tweets and have a high interaction and engagement with other users.
4. **Informer** -> public users that have many followers, are publicly listed and have a high global tweet count.
5. **Leader** -> popular users that receive many replies to tweets and interact mainly with other users of similar status.

At k-cluster=7 (not displated), 3 distinct positions are well detected. These are basically the same as above, except that the *Activator* and *Spreader* types form a combined cluster, and so do the *Informer* and *Leader* types:

1. **Observer**
2. **Activator + Spreader**
3. **Informer + Leader**

Due to the significant outlier features exhibited by the **Tranced6** user skewing the partitions found at k-cluster=7, I proceed with k-cluster=10 to avoid biasing the results.

Below, I list the cluster sizes and usernames of members assigned to each distinct cluster group. I also re-plot the mean scores to include only the disctinct clusters.

```
--------
K-Cluster Sizes:
------------------------
          id
Cluster
0        591
1        365
2        232
3         88
4          6
5        633
6         13
7          1
8         12
9         76
------------------------

--------
K-Cluster Types:
------------------------
--------
Cluster 9: Observer
--------

List too large to display

Features: +ve eccentricity and -ve closeness
------------------------


--------
Cluster 6: Spreader
--------
```

```
             label
0        N_I_C_S_A
1      lilanichlsn
2      _Calculator
3       philcraig2
4     SaaymanBarry
5     PeterDermauw
6       ConCaracal
7       UnmovedLee
8          Dijosti
9   Teboho41703390
10     JohnWeak077
11   Ricky_ting777
12      ghandagand

Features: +ve betweenness
------------------------


--------
Cluster 7: Activator
--------

      label
0  Tranced6

Features: +ve outdegree, hierarchy, transitivity, reciprocity, #tweeted
------------------------
*Table list sorted by feature values.


--------
Cluster 8: Informer
--------

             label
0     ntsikimazwai
1         SABCNews
2     Unathi_Kwaza
3         ALETTAHA
4   SABreakingNews
5             eNCA
6           News24
7         JacaNews
8              IOL
9       ewnupdates
10       TimesLIVE
11     ewnreporter
```

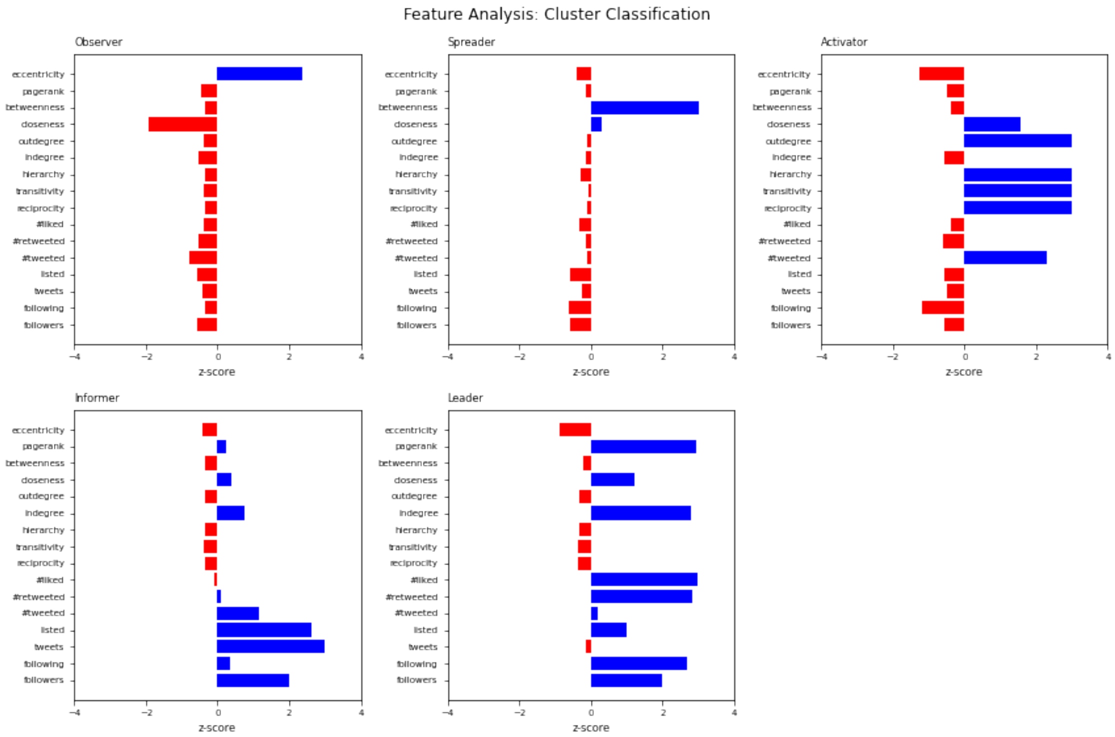
```
Features: +ve tweets, listed, followers*
------------------------
*Table list sorted by feature values.


--------
Cluster 4: Leader
--------

             label
0           Our_DA
1      PieterDuToit
2        ErnstRoets
3    EFFSouthAfrica
4   MbuyiseniNdlozi
5   Julius_S_Malema

Features: +ve pagerank, indegree, #liked, #retweeted, following,
followers*
------------------------
*Table list sorted by feature values.
```

3.2 Mean Score Plots (excl. Minor)



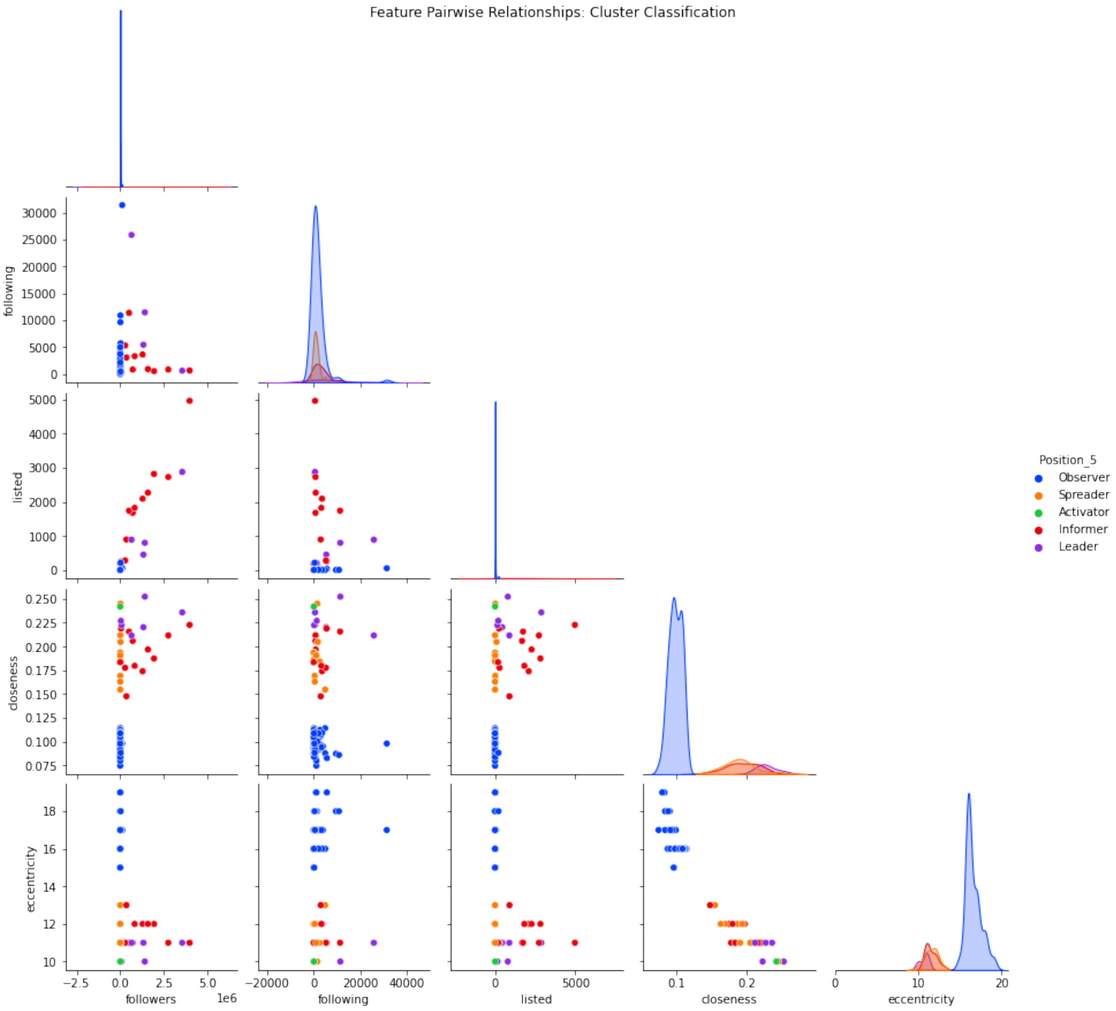
4 EVALUATION

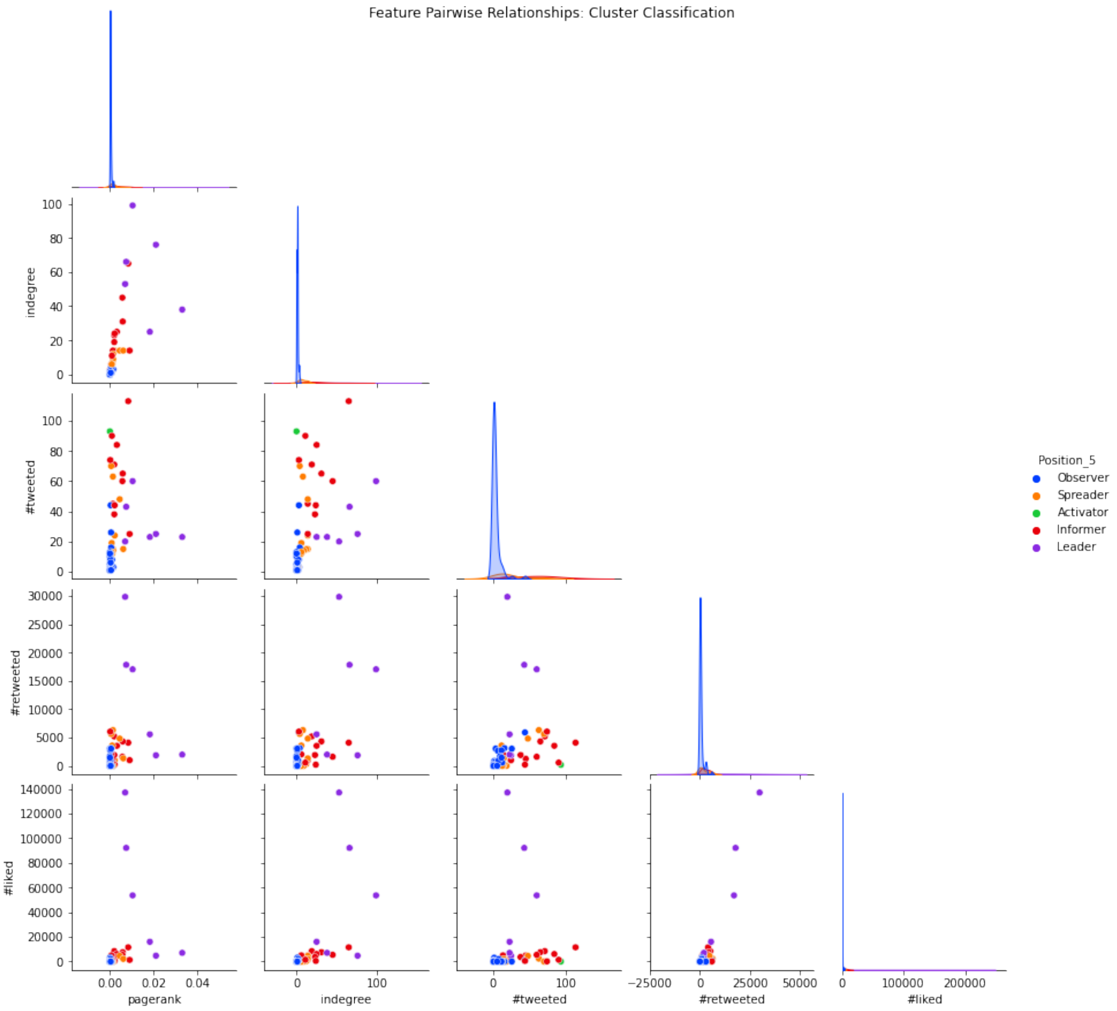
Using pythons seaborn library, I plot the pairwise relationships of detected clusters to assess the fit of nodes within a cluster and the relationships between nodes and their features.

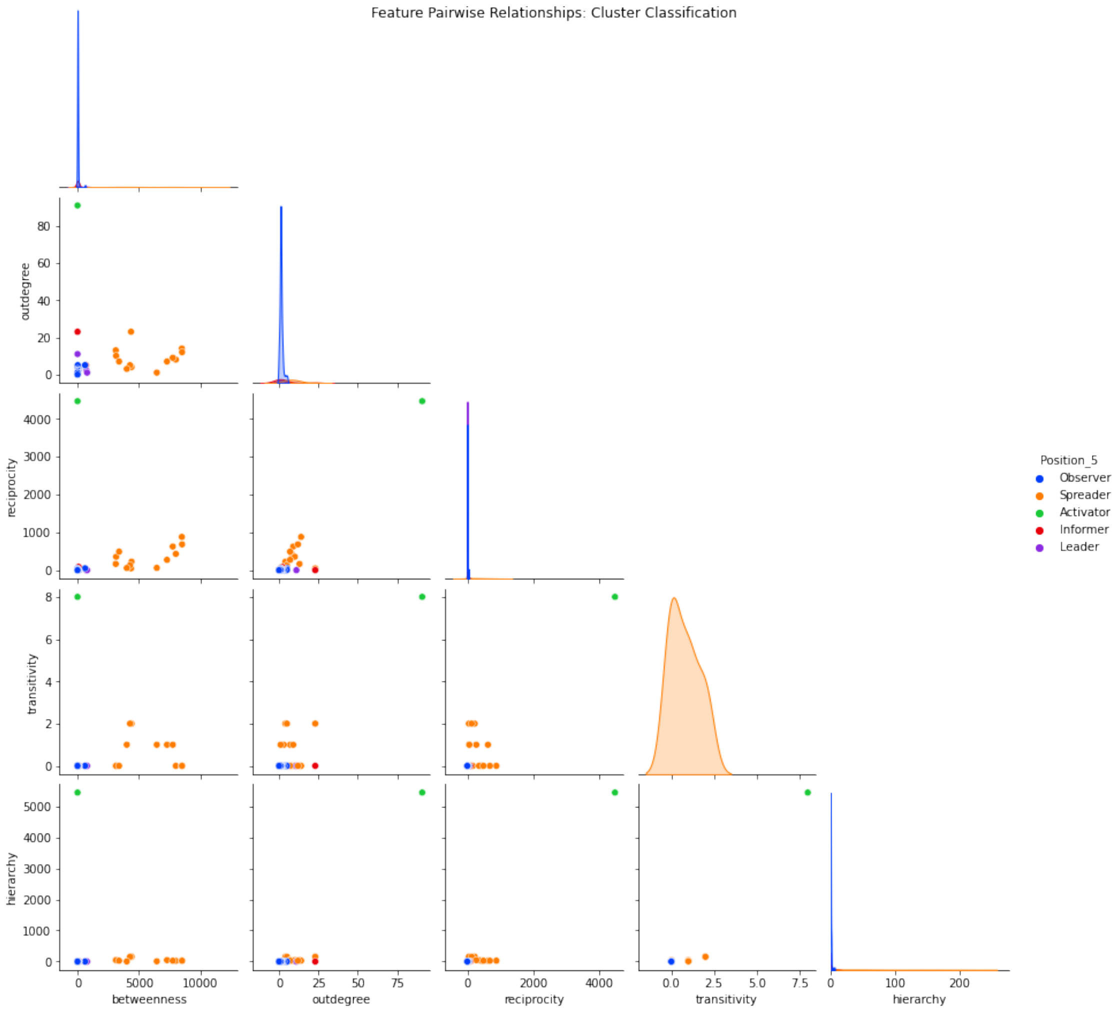
To interpret the results, lets say we want to consider the relationship between a node's distance from other nodes in the network and the probability of being added to a public list. From the plot, this can be found where the *closeness* feature on the y-axis meets the *listed* feature on the x-axis. Here, we can see that nodes in the *Informer* cluster share a positive relationship with being publicly listed and being close to other nodes in the network. Alternatively, if you are interested in the relationship between *indegree* and *pagerank*, then you would find the *pagerank* of interacting with high status users and the *indegree* of users tweets being replied to positively related to *Leader* type nodes. Additionally, when observing *reciprocity* and *outdegree* features, no significant relationships appear in any other groups except for the *Activator* group, shown by single green node in the top right corner that represents the **Tranced6** user account.

In the next plot, I exclude possible outliers by removing the *Activator* role. I find the *Spreader* cluster more pronounced when comparing the *outdegree* of replying to tweets and the *reciprocity* of a mutual response.

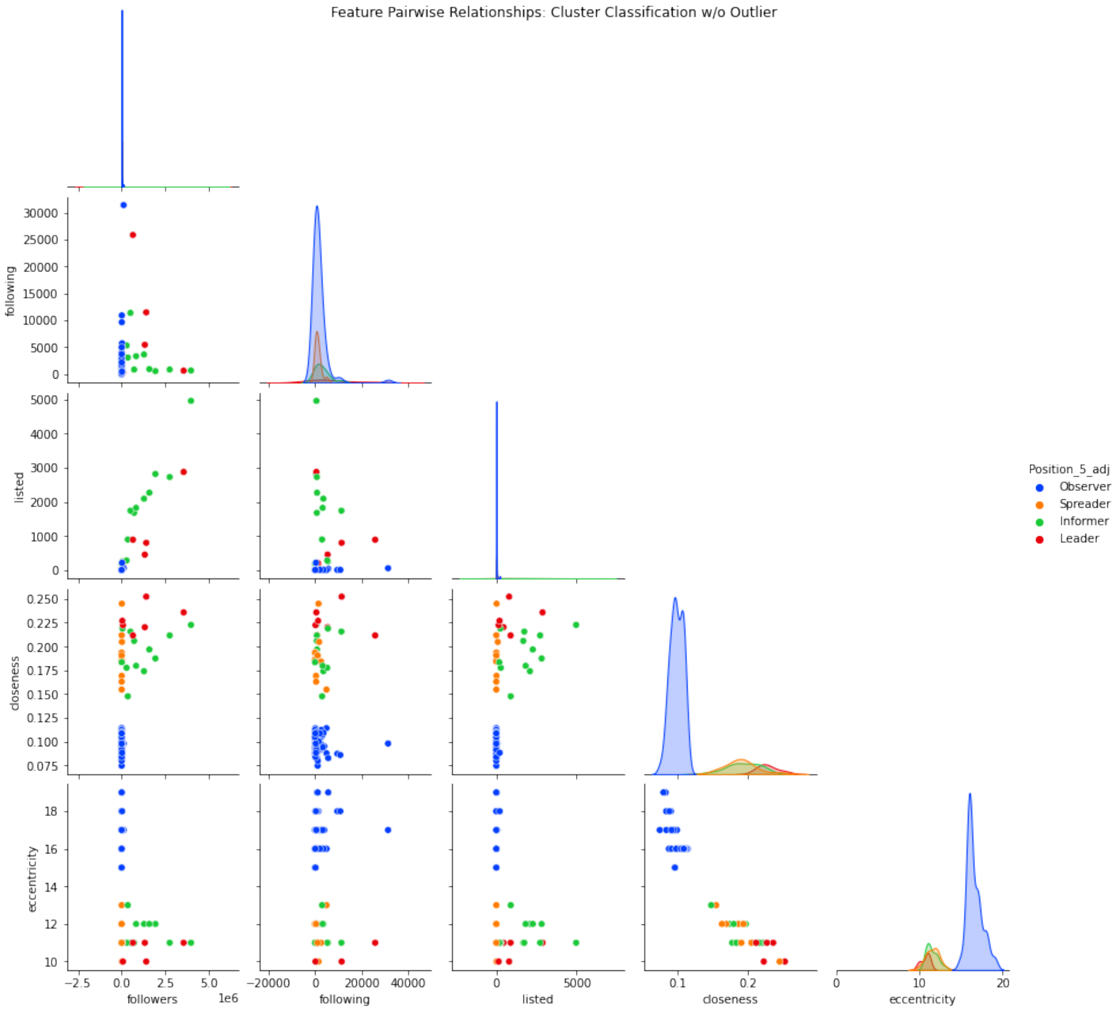
4.1 Pairwise Relationship Plots

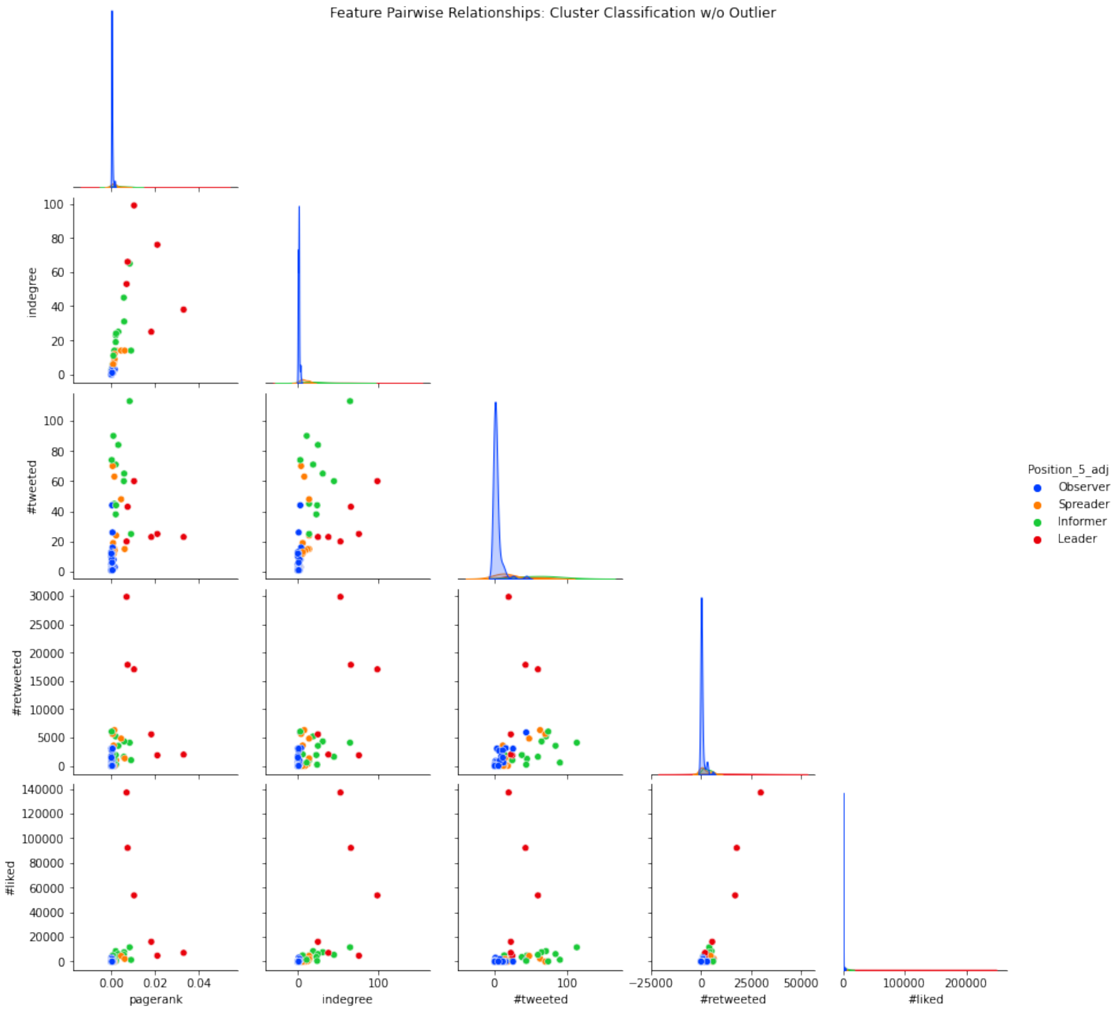



Feature Pairwise Relationships: Cluster Classification

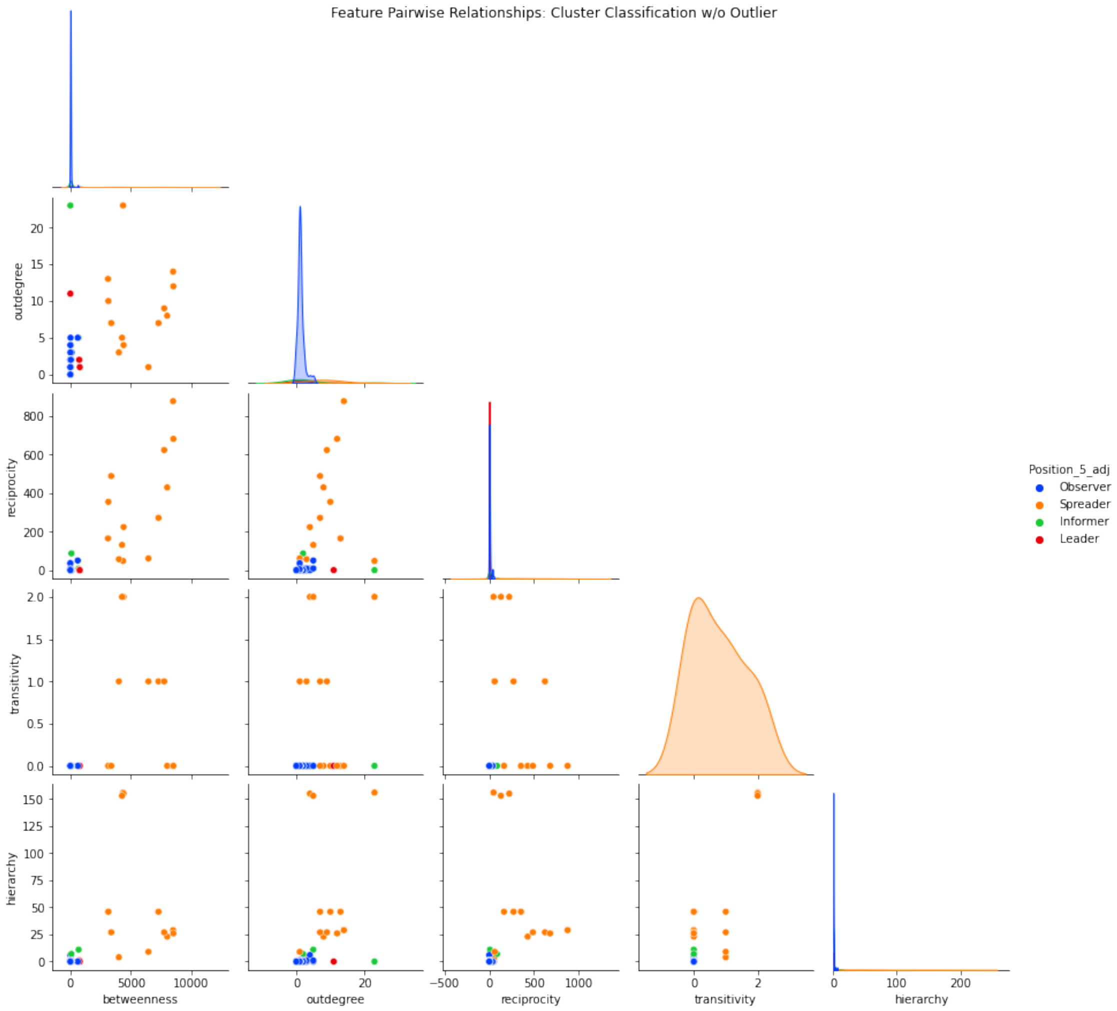

Feature Pairwise Relationships: Cluster Classification
outdegree
80
60
40
20
0
reciprocity
4000
3000
2000
1000
0
transitivity
8
6
4
2
0
hierarchy
5000
4000
3000
2000
1000
0
0
5000
10000
betweenness
0
25
50
75
outdegree
0
2000
4000
reciprocity
0.0
2.5
5.0
7.5
transitivity
0
100
200
hierarchy
Position_5
Observer
Spreader
Activator
Informer
Leader

4.2 Pairwise Relationship Plots (excl. Outlier)





Feature Pairwise Relationships: Cluster Classification w/o Outlier



5 SUMMARY

In this section, I use feature based techniques to conduct analysis on the positions and roles that best describe users in the #Senekal twitter reply network. Below is a summary of notable findings.

I identify 5 distinct clusters that fairly categorise the roles one could expect from a discourse network on a social media platform like Twitter and classify them as such:

1. **Observer** -> periphery users that are far away from the centre of the discourse and have little participation or involvement.

2. **Spreader** -> bridge users that are connected to users in various positions and can easily spread information across the network.

3. **Activator** -> active users that reply to tweets and have a high interaction and engagement with other users.

4. **Informer** -> public users that have many followers, are publicly listed and have a high global tweet count.
5. **Leader** -> popular users that receive many replies to tweets and interact with other important users.

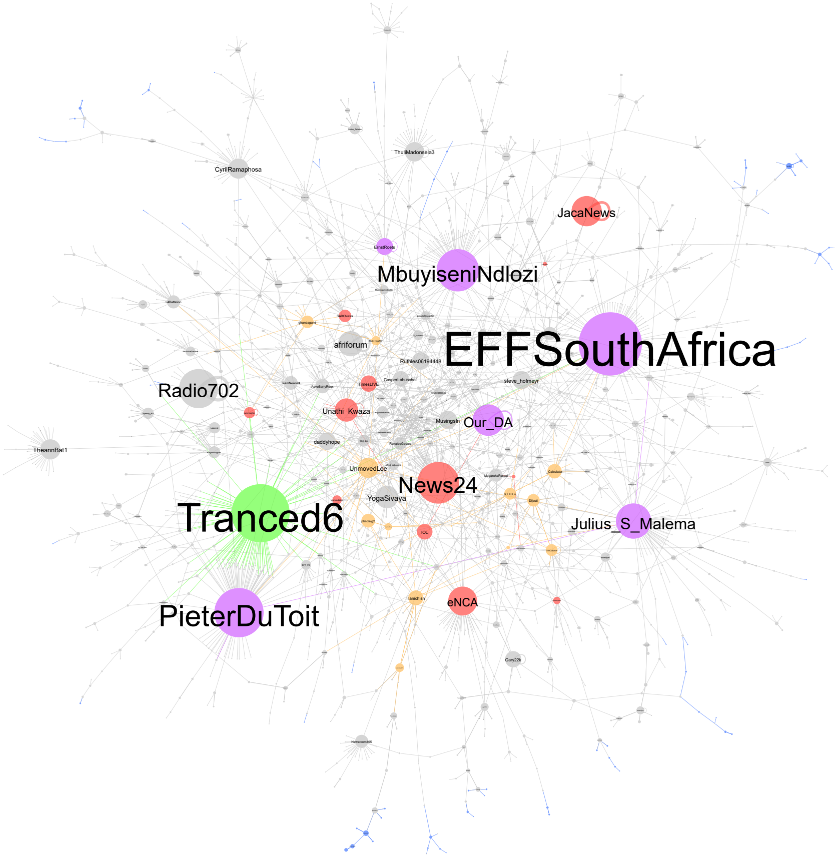
Not surprising, I find members of leading opposition parties assigned to the role of *Leader*, namely: **Our_DA** (DA), **ErnstRoets** (Afriforum), **EFFSouthAfrica** (EFF), **MbuyiseniNdlozi** (EFF) and **Julius_S_Malema** (EFF).

True to its name, I find the *Informer* role majorly skewed toward journalist and news agency accounts, e.g. **SABCNews**, **eNCA**, **SABreakingNews**, **News24**, and more.

Finally, I find the twitter account **Tranced6**, which is the only user assigned to the *Activator* role, to exhibit outlier properties accross the *outdegree*, *reciprocity*, *transitivity* and *hierarchy* features. These suggests that users falling within the neighbourhood are prone to replying to each other's tweets at a higher rate, as well as maintaining a consistency in the users they choose to reply to or not reply to. They also tend toward the same hierarchal agreement or consensus on which users they believe to be worth sending their ties to. Consistent with findings from part I, this shows a high probablity of circulation and redundancy of tweets being shared between the same users, thus making them more susceptible to confirmation bias and "groupthink".

In the next section, I simulate multiple diffusion processes using the discourse network to determine whether certain positions in highly clustered communities promote or hinder the spread of information and ideas.

```
--------
Senekal Positions:
```


CyrilRamaphosa
ThuliMadonsela3
JacaNews
ErnstRoets
MbuyiseniNdlozi
ghandagand
SABCNews
afriforum
Ricky_Ing777
EFFSouthAfrica
Ruthles06194448
Radio702
TeamNews24
AdvoBarryRoux
TimesLIVE
CasperLabuscha1
steve_hofmeyr
Unathi_Kwaza
MusingsIn
Our_DA
TheannBat1
daddyhope
RenaldoGouws
UnmovedLee
MugarukaPascal
Calculator
News24
YogaSivaya
Dijosti
Tranced6
philcraig2
IOL
Julius_S_Malema
ConCaracal
lilanichIsn
eNCA
PieterDuToit
Gary29