PART III: BRACKENFELL
POSITIONAL ANALYSIS

Positional analysis is a 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

#Brackenfell Reply Network - Largest Component

Data Transformation

I exclude observations with any missing data from the feature data set as they are 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.

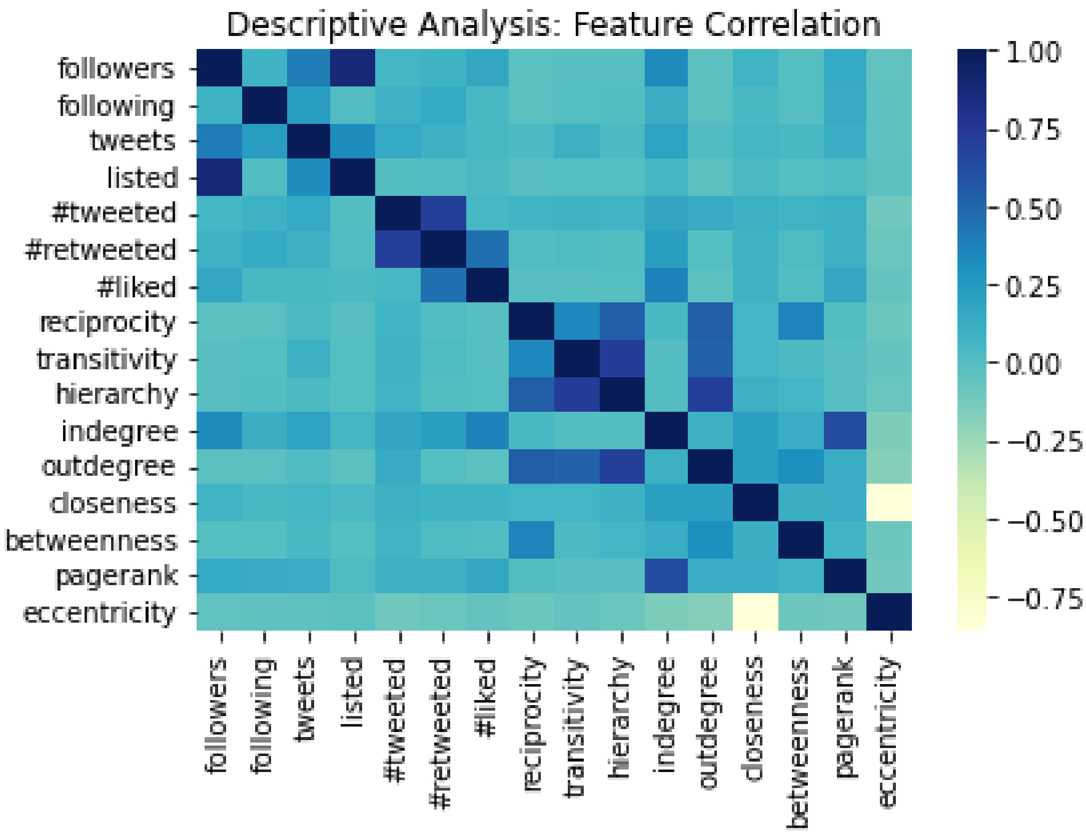
---

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:

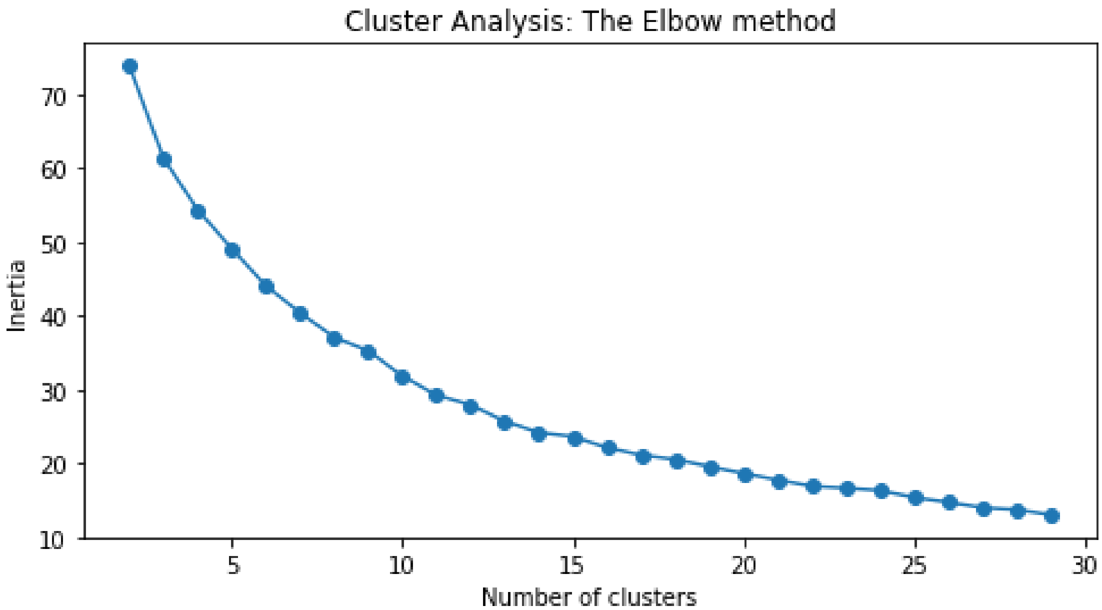
- being *listed* on a public list is positively correlated with the number of *followers*.
- *indegree* of tweets being replied to is positively correlated with the *pagerank* likelihood of being connected to other high rank positions.
- *reciprocity*, *transitivity*, *hierarchy* are all positively correlated 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

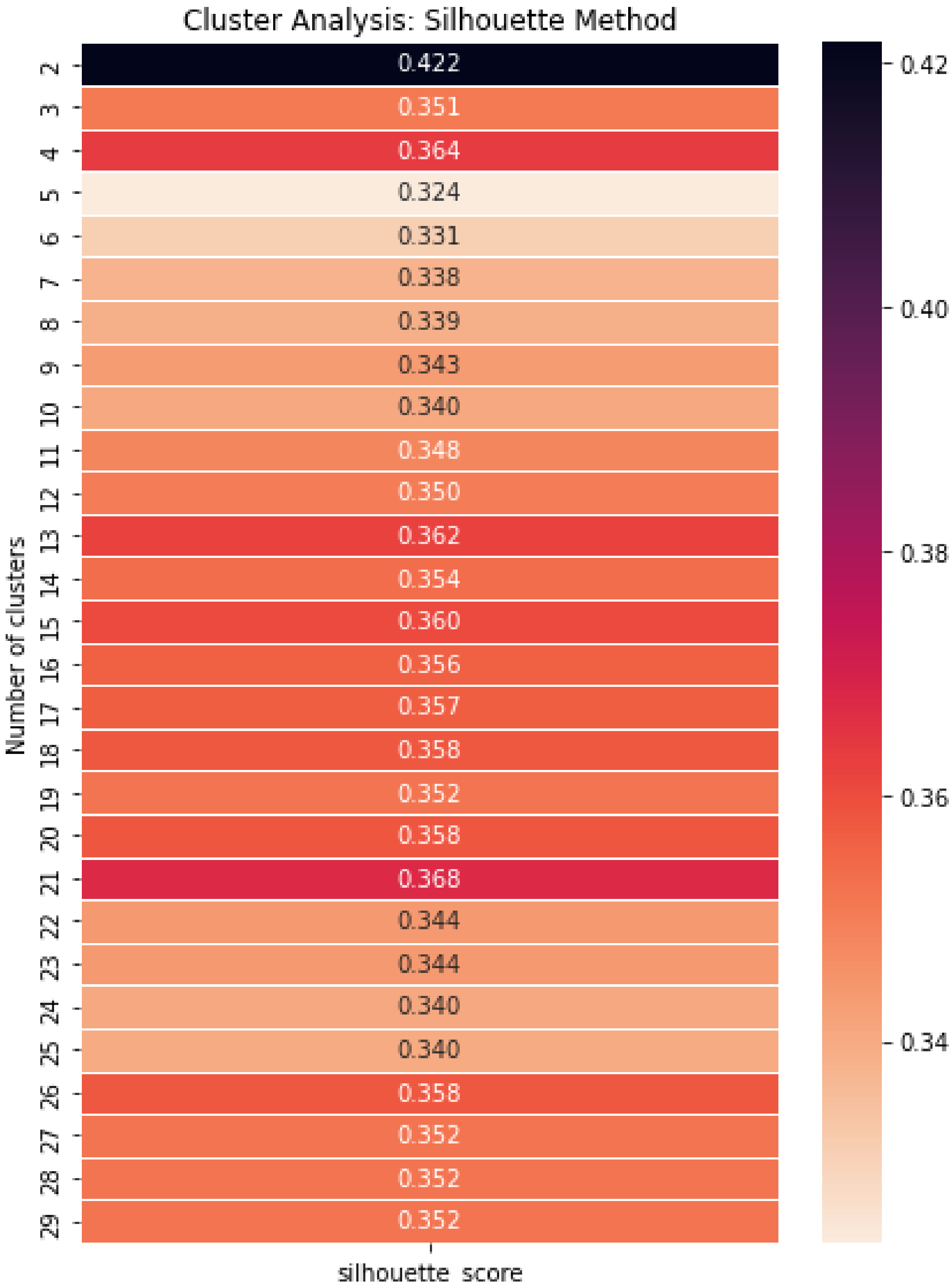


KneeLocator

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



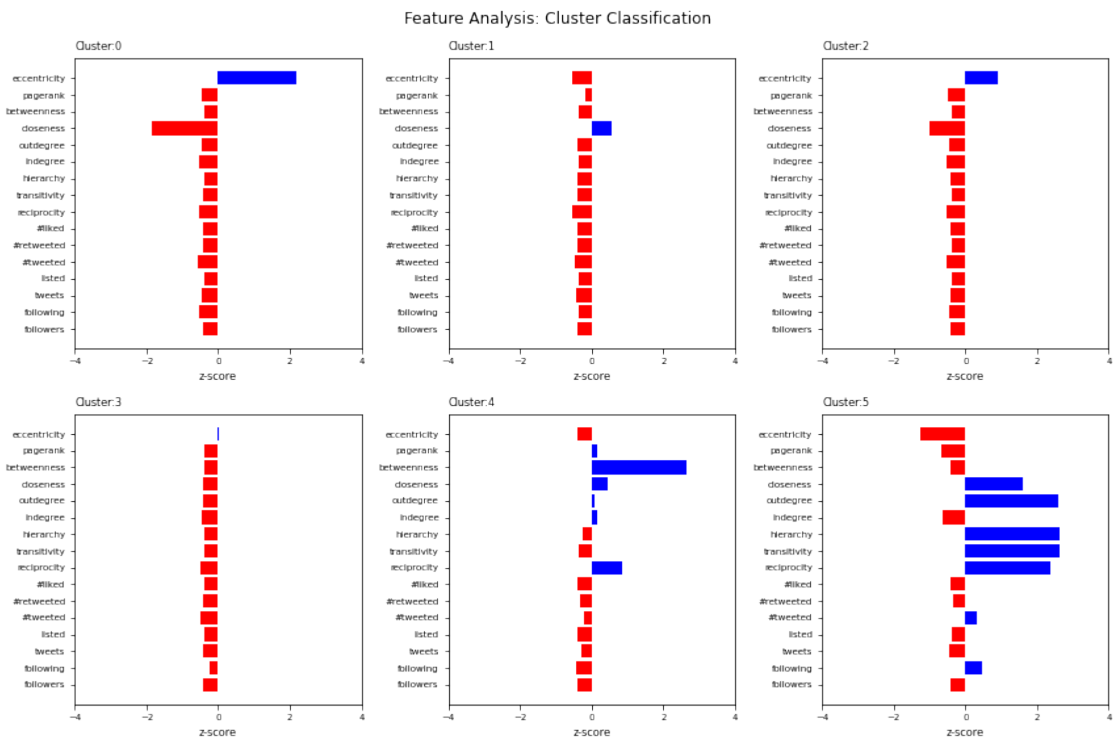
The highest silhouette score obtained for k-clusters between 2 and 30 is 0.422 at k-cluster=2.

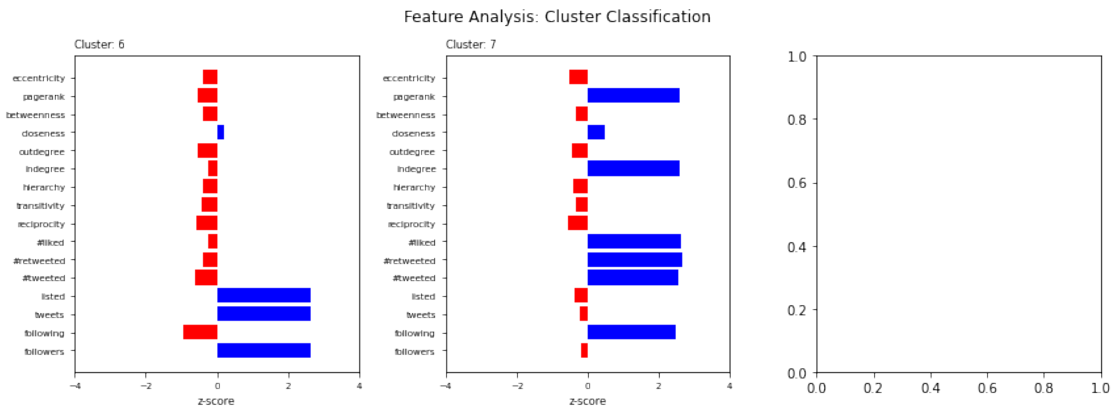
3 CLASSIFICATION

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=6 and k-cluster=8 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=8, 5 distinct clusters are detected. The remaining 3 exhibit non-distinct or low value features. Non-distinct 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=6 (not displayed), 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**

Considering that it is generally safer and likley more meaningful to set k-clusters at higher values, I find k-cluster=8 ideal.

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        194
1        454
2        381
3        403
4         17
5          2
6          2
7         11
------------------------

--------
K-Cluster Types:
------------------------
--------
Cluster 0: Observer
--------

List too large to display

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

--------
Cluster 4: Spreader
--------

              label
0       JoelMMalope
1         Godwinse1
2         ramalokot
3      Mellow_Rocker
4         sick6_six
5      Mutwanamba_SA
6      UnsceneDamian
7    karabonkabinde6
8         Janneman7
```

```
9        UnmovedLee
10  Blessed11587260
11  Thabisa23846794
12   BiancavanWyk16
13        DavidRyke
14       Cerned_Con
15     StantemUrsae
16          kiewiet
```

Features: +ve betweenness

Cluster 5: Activator

```
        label
0    rasencha
1  xeshamusiq
```

Features: +ve outdegree, hierarchy, transitivity, reciprocity*

----

*Table list sorted by feature values.

Cluster 6: Informer

```
       label
0   guardian
1  AJEnglish
```

Features: +ve tweets, listed, followers*

----

*Table list sorted by feature values.

Cluster 7: Leader

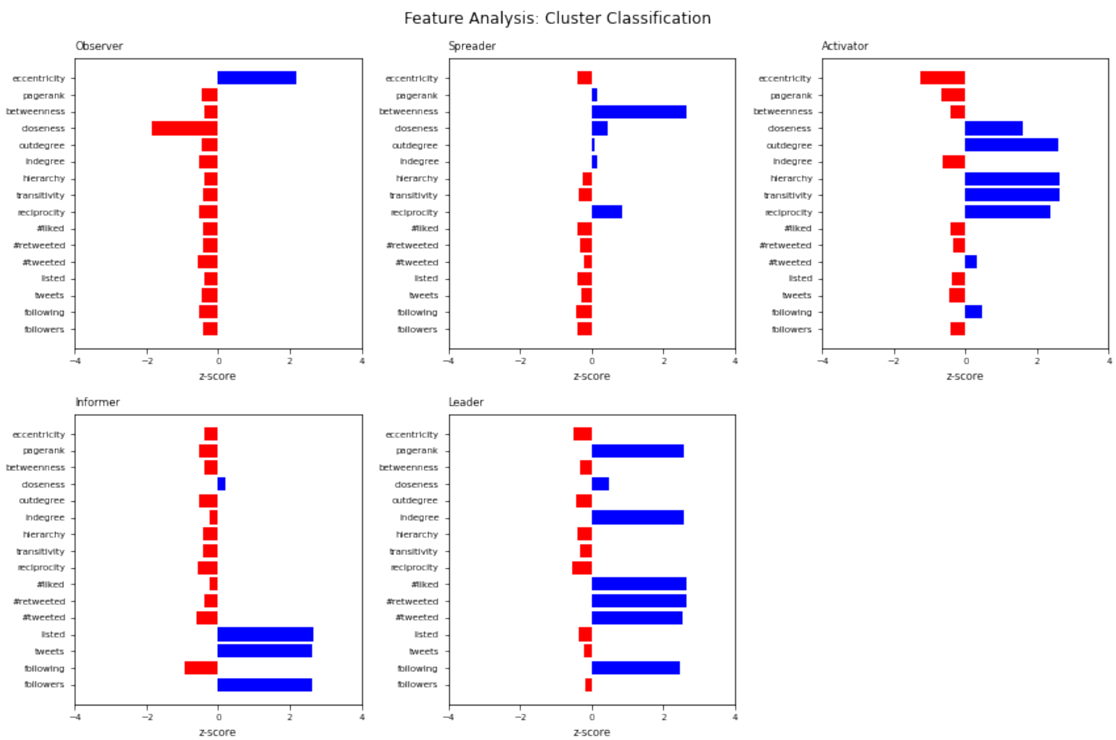
```
             label
0    EFFSouthAfrica
1   Julius_S_Malema
2     GardeeGodrich
3   MbuyiseniNdlozi
4          ThokoziP
5       SiyaRumbu21
6     Tyrone_Mkansi
7    FreeGeneraxion
```

```
8    tshepaMotshewa
9      alyssaraetho
10         Nananah2
```

Features: +ve pagerank, indegree, #liked, #retweeted, #tweeted, following*

```
------------------------
*Table list sorted by feature values.
```

3.2 Mean Score Plots (excl. Minor)



---

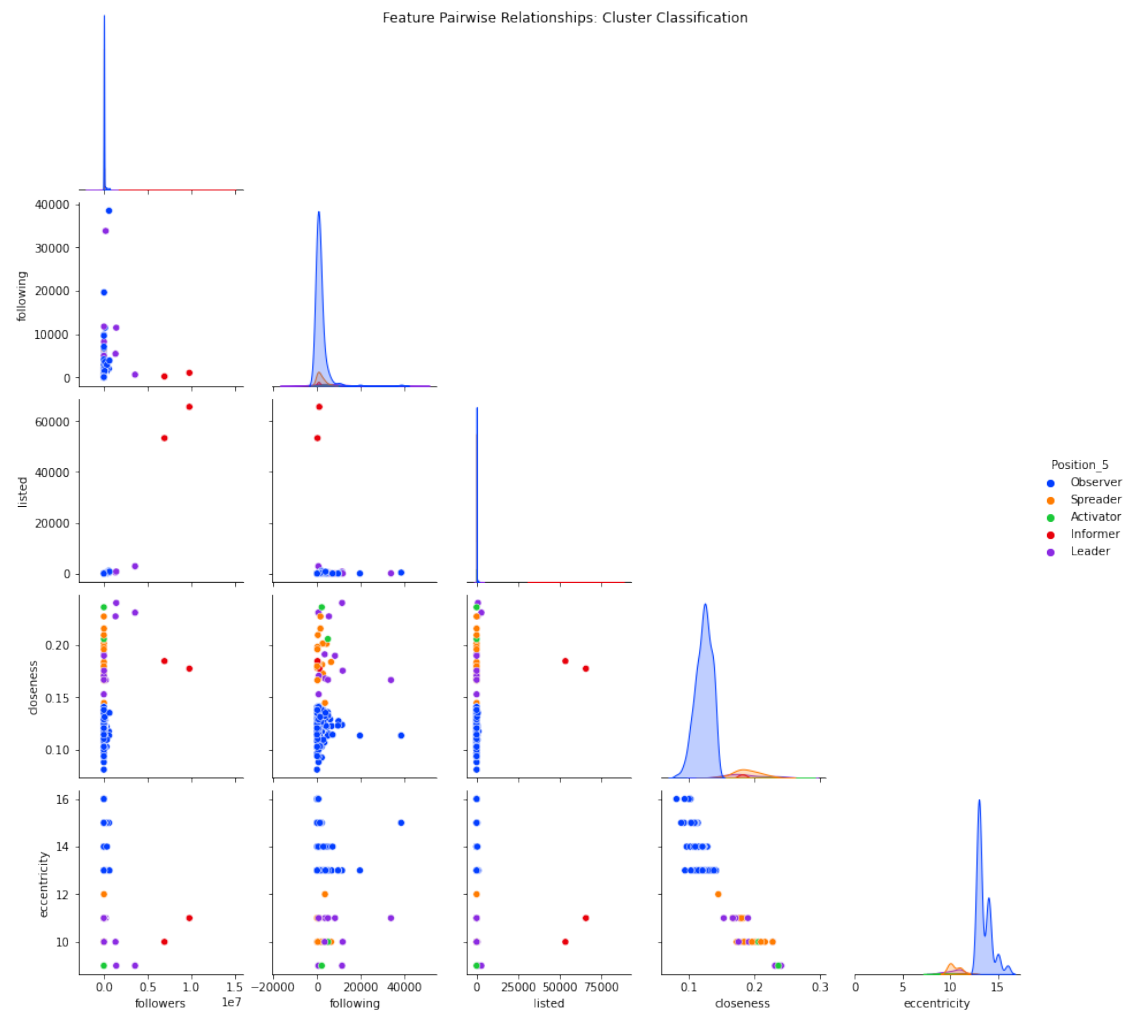
4 EVALUATION

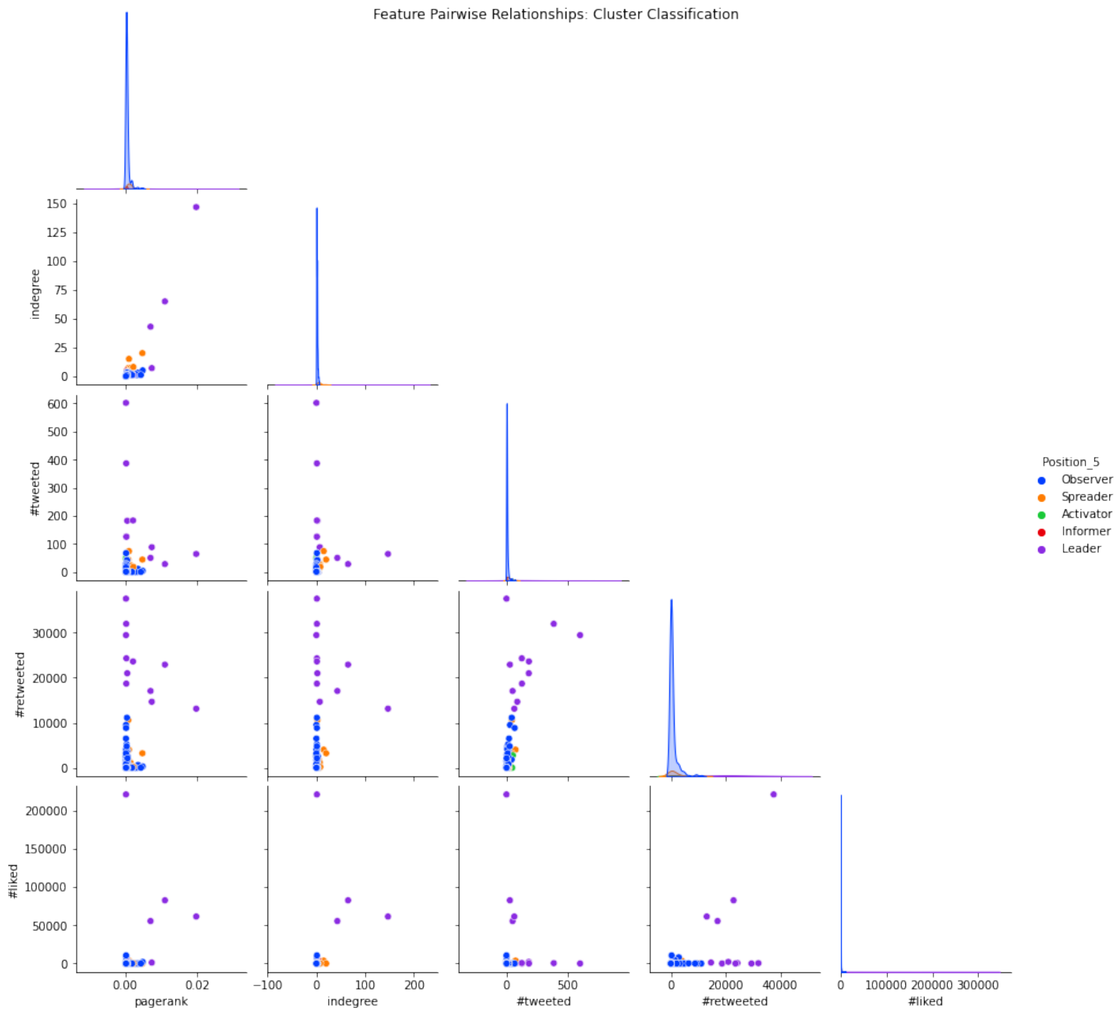
Using pythons seaborn library, I plot the pairwise relationships of detected clusters to assess the fit of nodes within a cluster and the relationships between nodes and their features.

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 receiving replies to tweets positively related to *Leader* type nodes. Significant relationships are observed for *Activator* and *Spreader* types when interacting *reciprocity* and *outdegree* features.

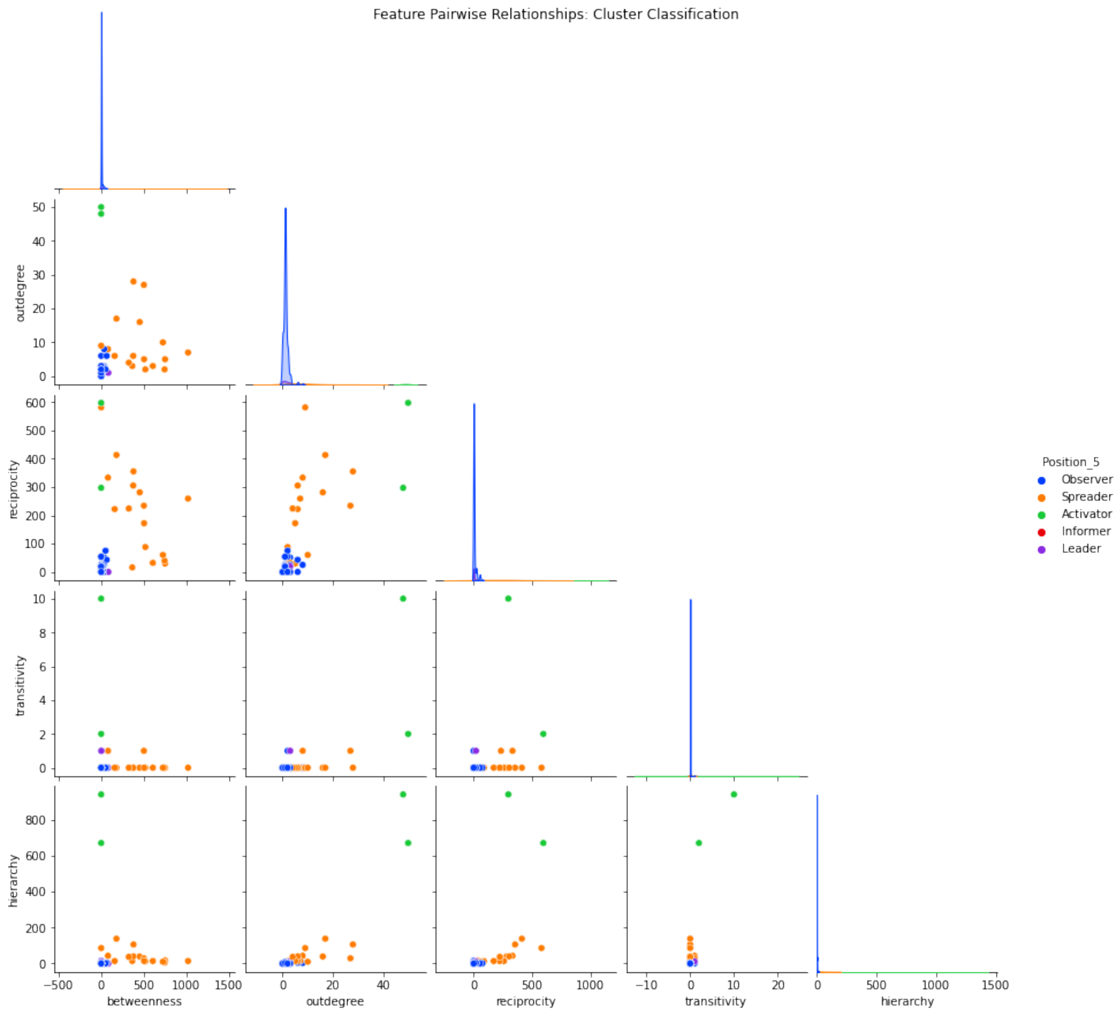
In the next plot, I exclude possible outliers by removing the *Activator* and *Informer* roles. I don't find much difference in the results except that the *Spreader* cluster is now more pronounced when comparing the *outdegree* of replying to tweets and the *reciprocity* of a mutual response.

4.1. Pairwise Relationship Plots

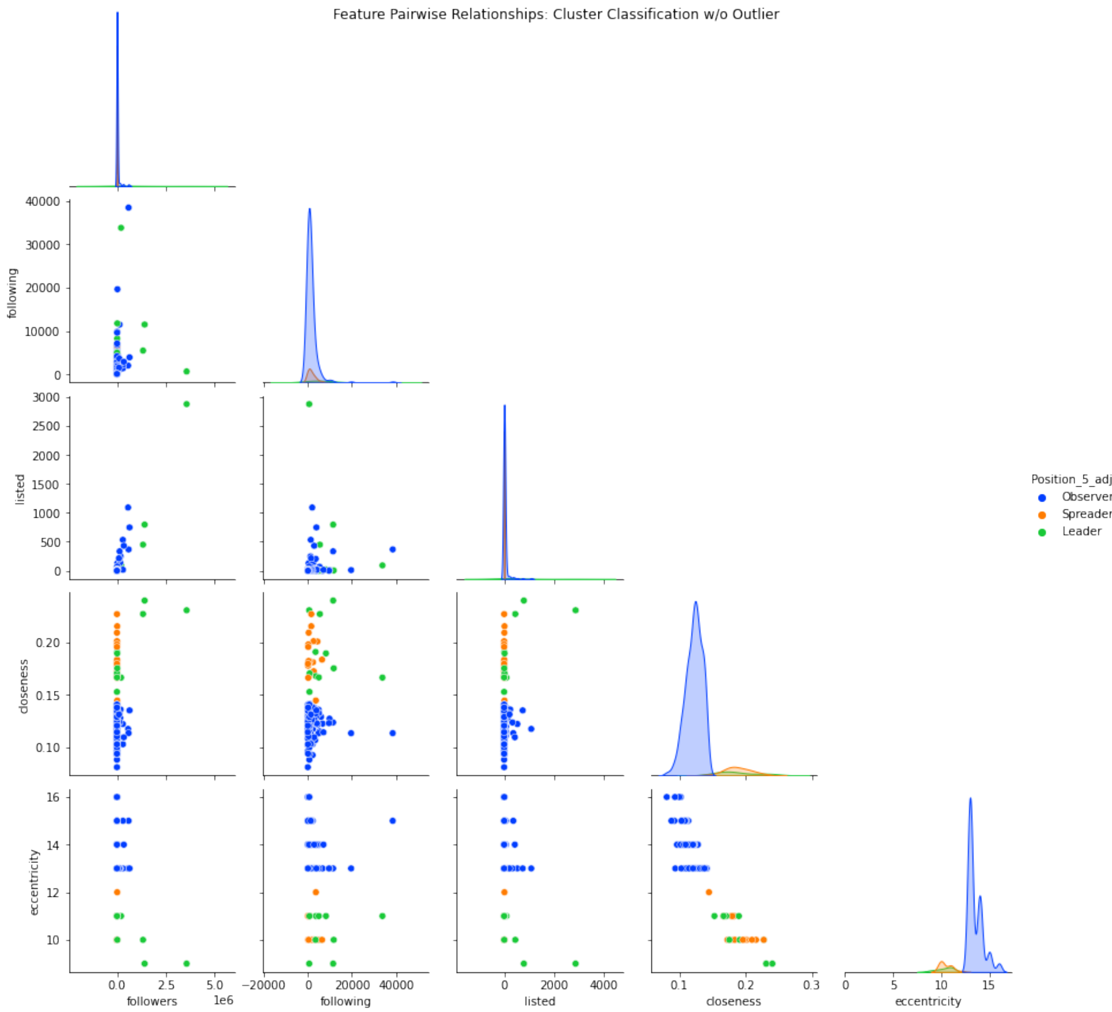


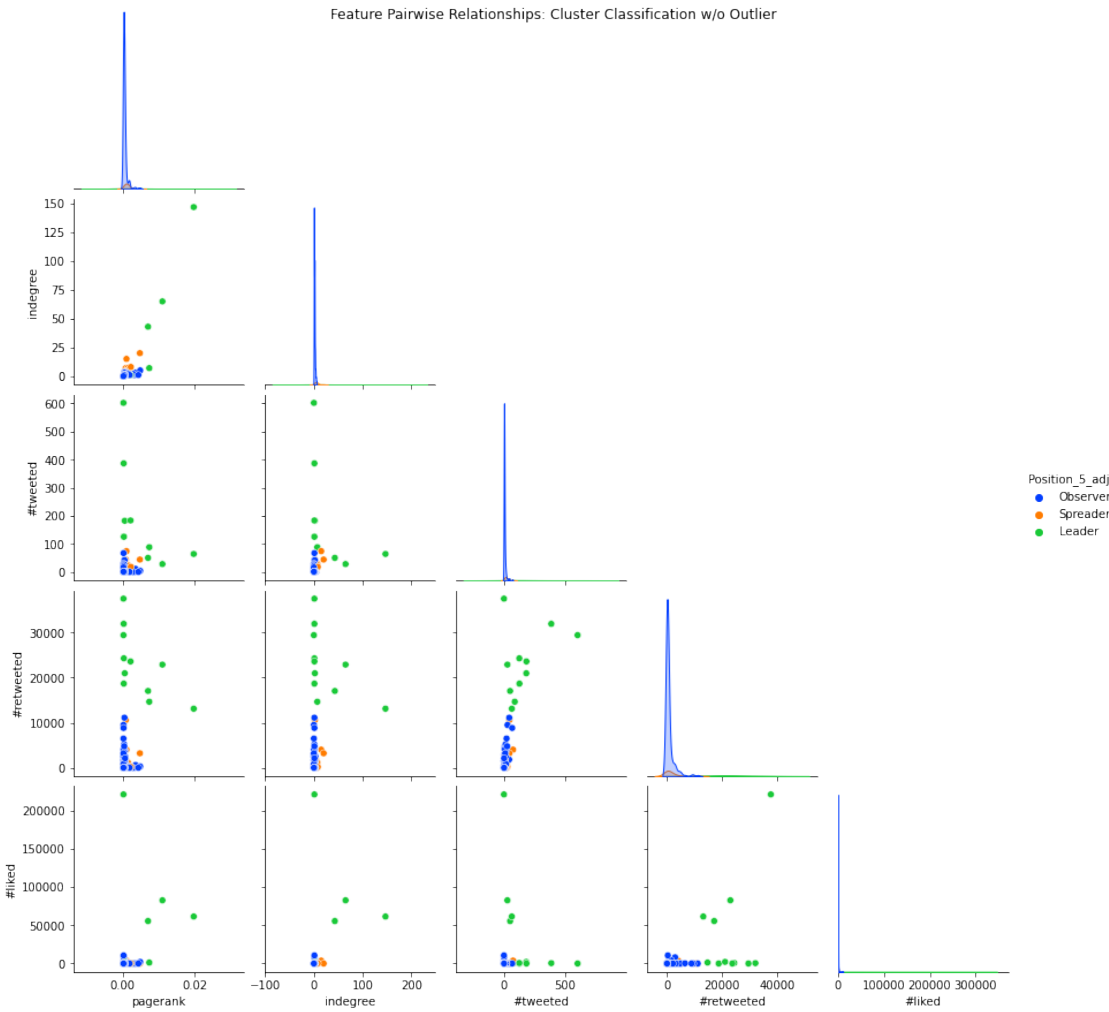


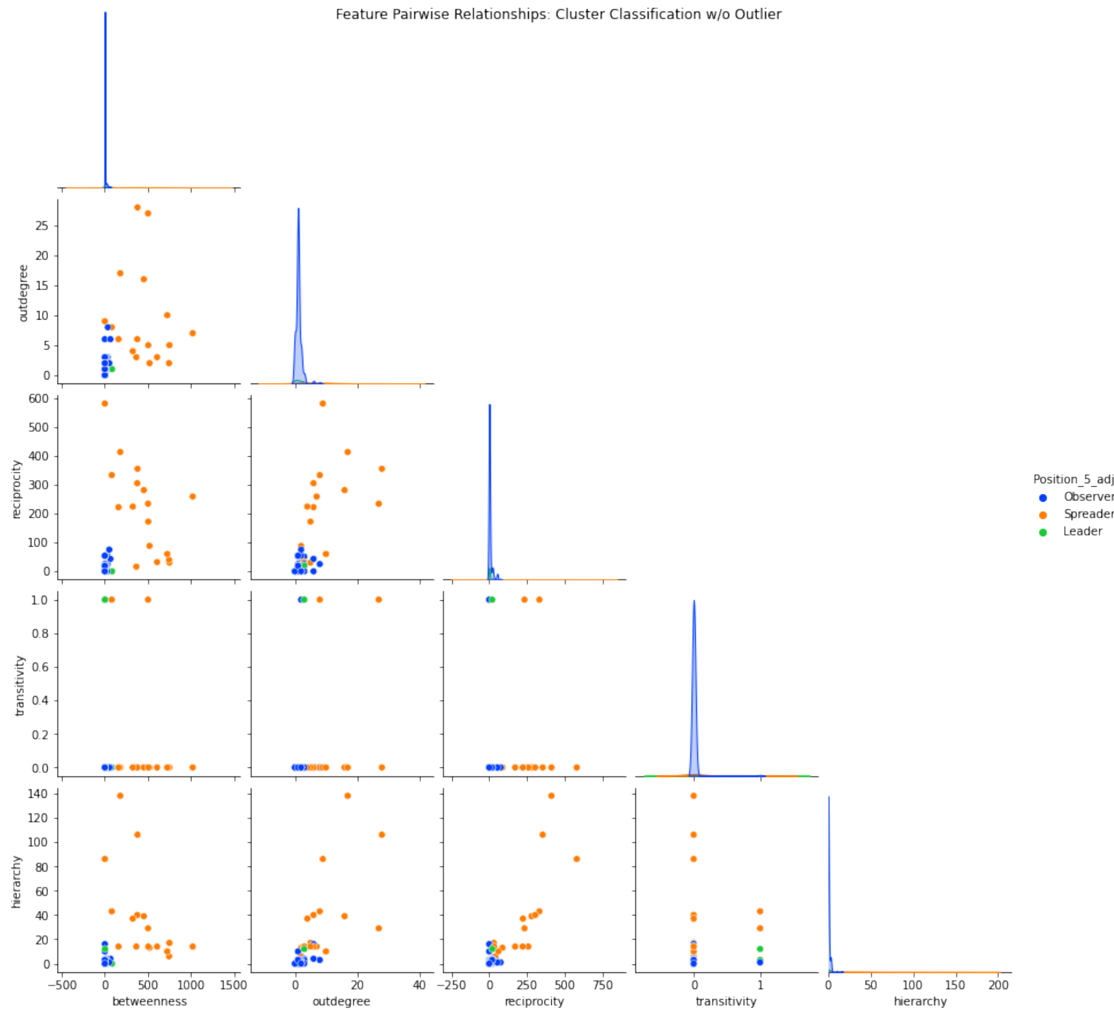
Feature Pairwise Relationships: Cluster Classification
Position_5
Observer
Spreader
Activator
Informer
Leader
indegree
#tweeted
#retweeted
#liked
pagerank



4.2. Pairwise Relationship Plots (excl. Outlier)


Feature Pairwise Relationships: Cluster Classification w/o Outlier
Position_5_adj
Observer
Spreader
Leader
followers
following
listed
closeness
eccentricity
1e6


Feature Pairwise Relationships: Cluster Classification w/o Outlier
indegree
#tweeted
#retweeted
#liked
pagerank
Position_5_adj
Observer
Spreader
Leader



5 SUMMARY

In this section, I use feature based techniques to conduct analysis on the positions and roles that best describe users in the #Brackenfell 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.

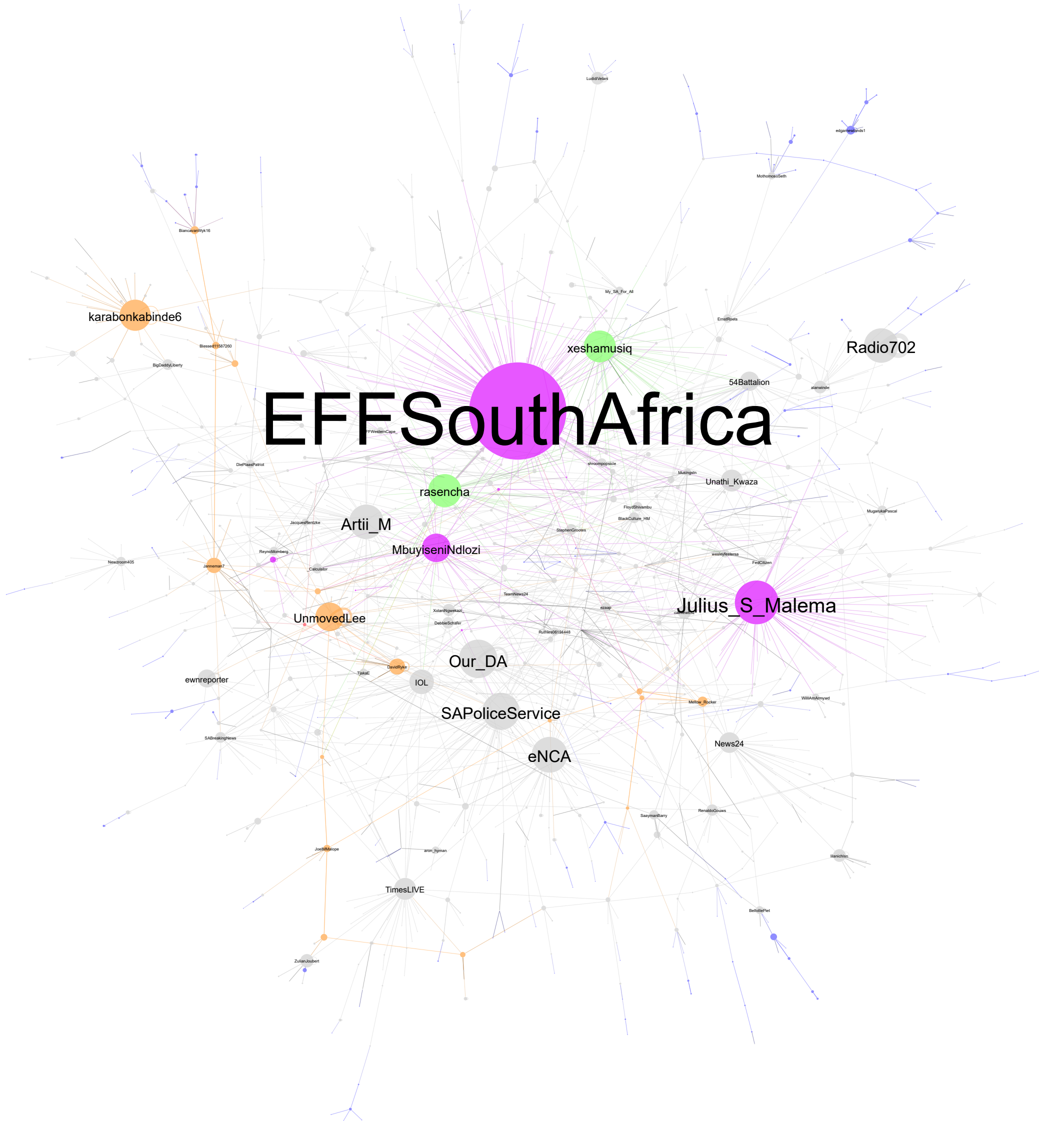
I find users in support of the protests well represented in the *Leader* role. At the top of this list are members of the Economic Freedom Fighters (EFF) party, namely: **Julius_S_Malema**, **GardeeGodrich**, and **MbuyiseniNdlozi**.

Instead of locally based journalists and media agencies, I find international and foreign news publications, **guardian** and **AJEnglish**, to be the only news media assigned to the *Informer* role. This is in large part due to specific outlier features biasing the clustering algorithm. Given the extremely high number of *tweets*, *listed*, and *followers* values that their global reach affords them, it is not surprising that **guardian** and **AJEnglish** appear to stand out far beyond the other media accounts. In reality, the **guardian** only contributed 1 tweet to the discourse while **AJEnglish** contributed 3 tweets, which are significantly less than **eNCA** and **News24**'s 70 and 68 tweets, respectively. Considering that this cluster type is purely defined by its global features of *tweets*, *listed*, *following*, and *followers*, the assignment of the international publications, although contextually misplaced, is technically fitting.

As previously noted in the descriptive part of the analysis, features that classify **ransecha** and **xeshamusiq** in the *Activator* cluster are more true for **racensha** than **xeshamusiq**. From investigation, **xeshamusiq** appears to troll certain prominent users with the aim of promoting and plugging their music. While this strategy identifies them as a significant player in this analysis (mostly because they have strategically attached themselves to the neighbourhoods of those that are visibly active and influential), it does not seem to have resulted in any real or material response from other users in the network.

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.

```
--------
Brackenfell Positions:
```


EFFSouthAfrica
Julius_S_Malema
MbuyiseniNdlozi
xeshamusiq
rasencha
karabonkabinde6
Radio702
54Battalion
Unathi_Kwaza
Artii_M
UnmovedLee
Our_DA
IOL
SAPoliceService
eNCA
News24
ewnreporter
TimesLIVE