**Engagement:**

**Retweet**

To understand the Retweet pattern, we had few features in mind

1. Who was being retweeted max, can we differentiate the most retweeted celebrity between F and NF users?
2. Max Retweet frequency? (who was retweeting max outdegree of the user) How significantly different it is between F and NF

**Reply**

1. Volume**:** Total no of tweets / month
2. Mean proportion of reply for a given month /week/Day?

No of negative centric post /month for top negative users?

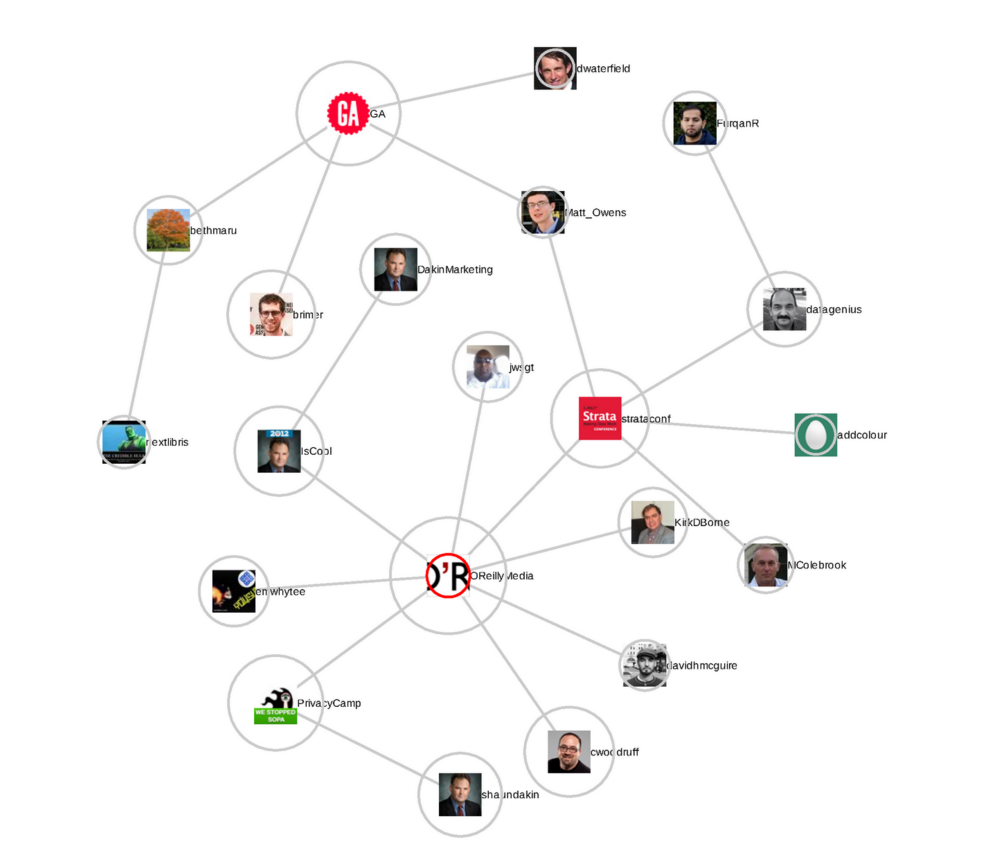
Ego network:

1. No of followers
2. Followee count

All the comparisons were done for Filtered vs Non Filtered users

**Retweet**

We wanted to figure out whose tweets were being max retweeted by Filtered and Non Filtered group and if there is a screaming difference between those celebrities/ famous users based on their max indegree



Figure

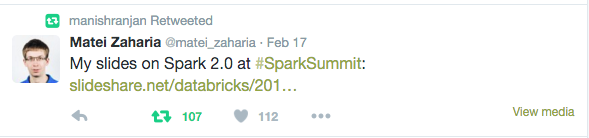
image source: <http://kalecoripple.blogspot.com/2012/10/where-did-my-tweet-go-tracking-retweets.html>

Say here we had u1, u2 and u3 in the Filtered list, we wanted to see what makes “OR” so famous among this set of users and also if there is an obvious difference between Filtered set and non filtered set.

Data Prep:

We had both set of users U\_F, U\_NF and we had randomly selected 1000 users to download last 3000 tweets (twitter api rate limitation)

And then say for a tweet text



Typically, it looks like

RT@ Matei Zaharia: text text text text text text

So we created a metrics of user to retweet mentions. And after extracting the user to retweet relation we loaded it to Gephi for further analysis.

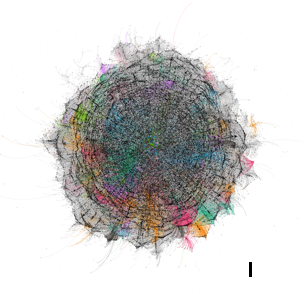


Figure Gephi retweet Graph

Nothing looks really clear, but when we zoom in a bit we can see the celebrities at the center. No surprise as people usually retweet famous people’s tweets.

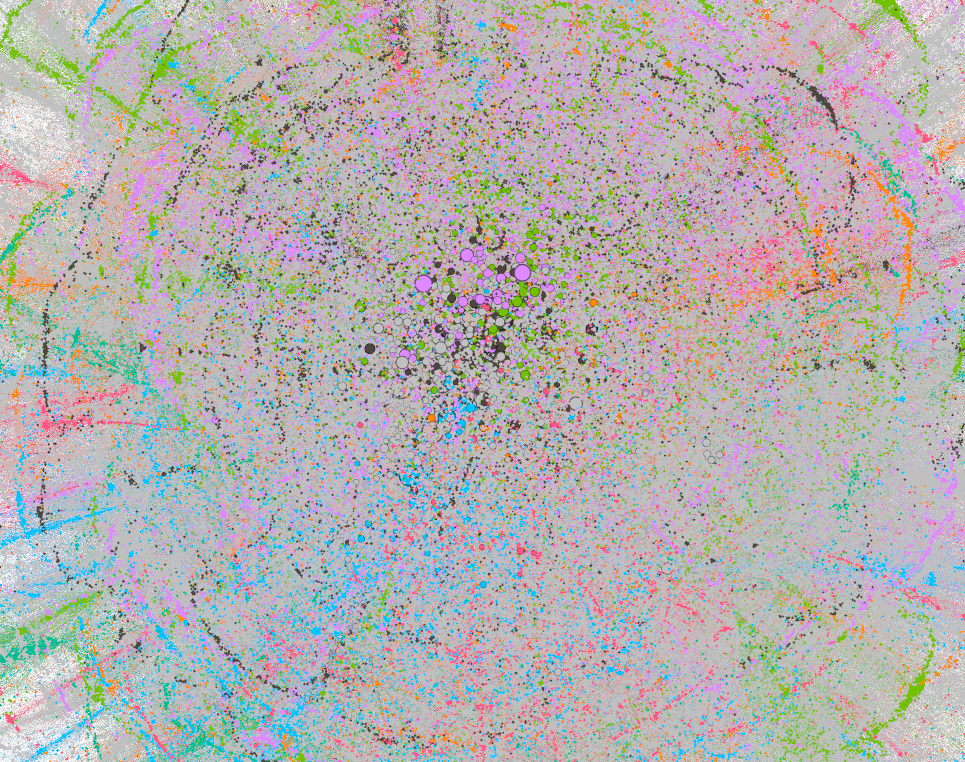


Figure 3Zoomed retweeted image

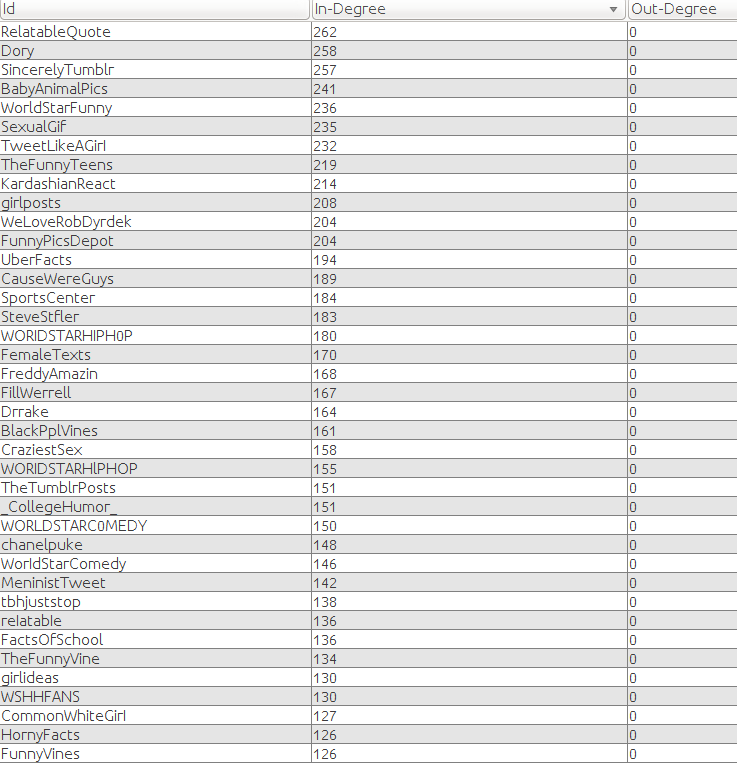
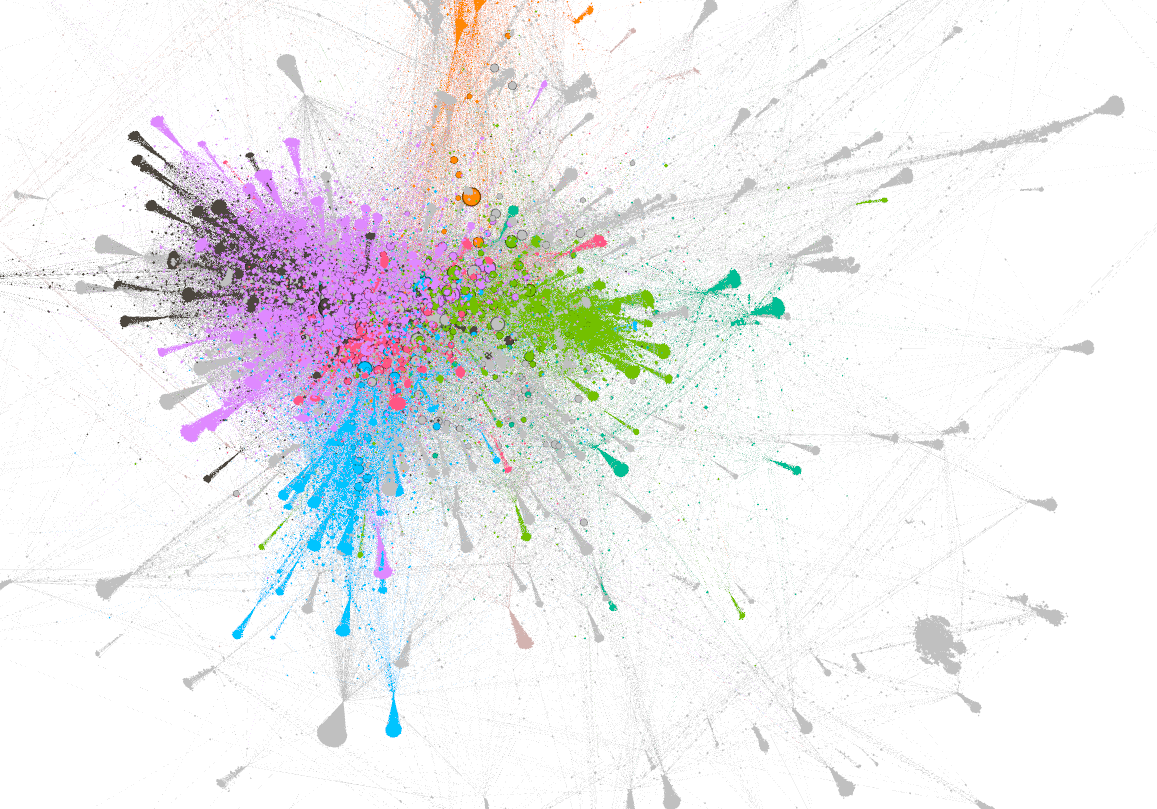
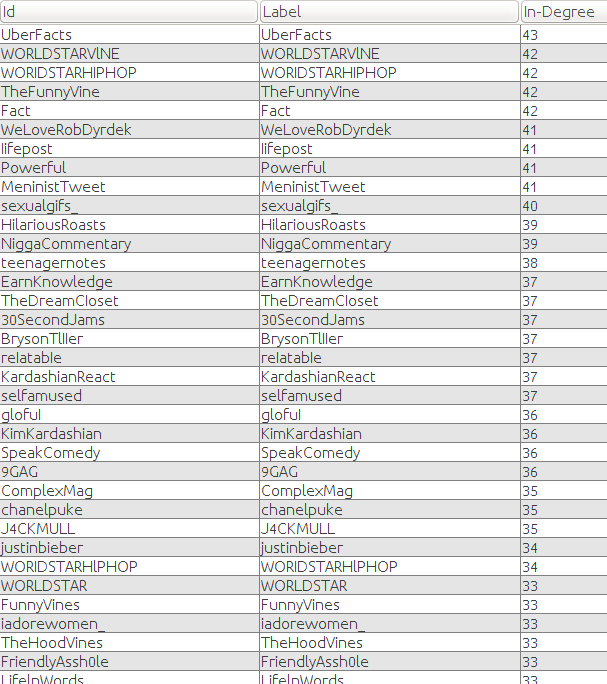


Figure 4users based on maximum indegree connections



Based on the max in-degree, we observed that there wasn’t a significant difference in users tweet being retweeted, which were almost either sexual jokes, or single liner quotes.

However, the moment we got rid of the top 10, the difference was more obvious between top retweet by F and NF users.



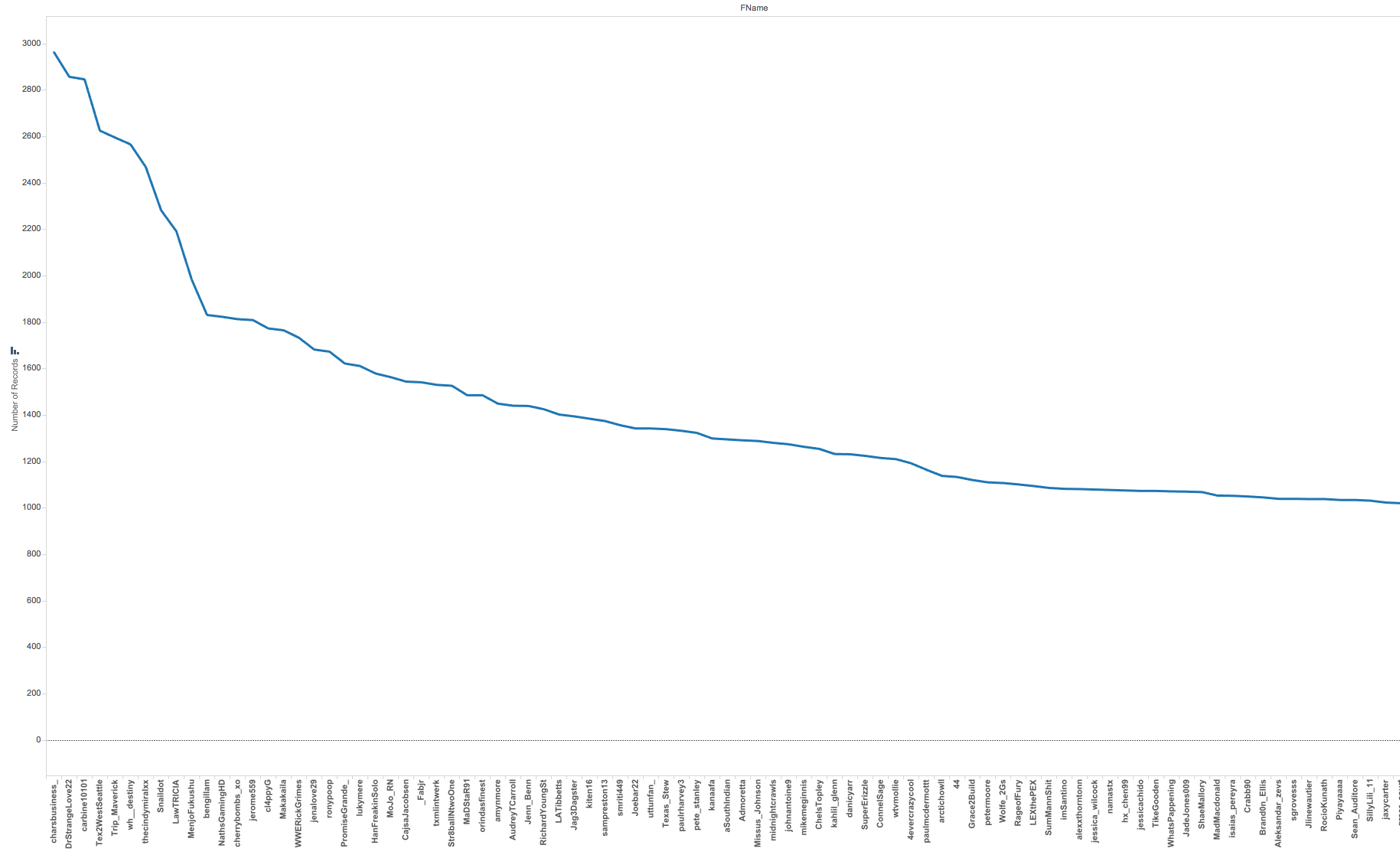
**Ahhh Finally Retweet Frequency makes sense:**

So, we got hold on Retweet frequency across two years 2014 and 15. For data cleanliness purpose we got rid of very small portion of tweets which were from 2008/09/10/11/12/13

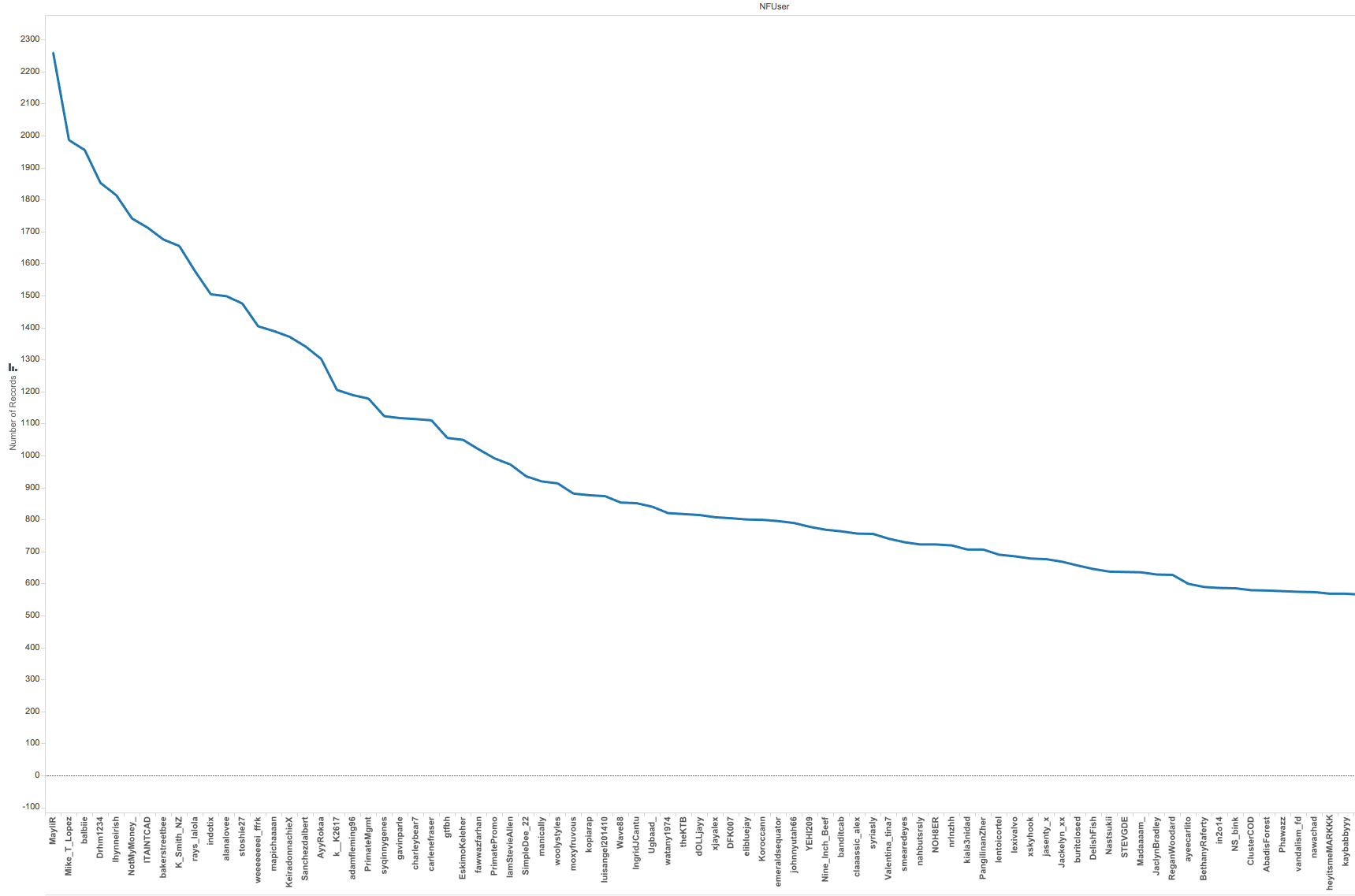
We created a map of user vs #Retweets for 2014 and 2015 of Filtered and Non Filtered users and results were indicating as if Filtered users were tweeting in higher frequency than non filtered ones

Replies:

Fuser Reply Freq:



NFUser Reply freq



Reply frequency of top 20%, excluding, what user is getting reply for. These are the replies user is sending

**Now analysis of how many replies they get:**

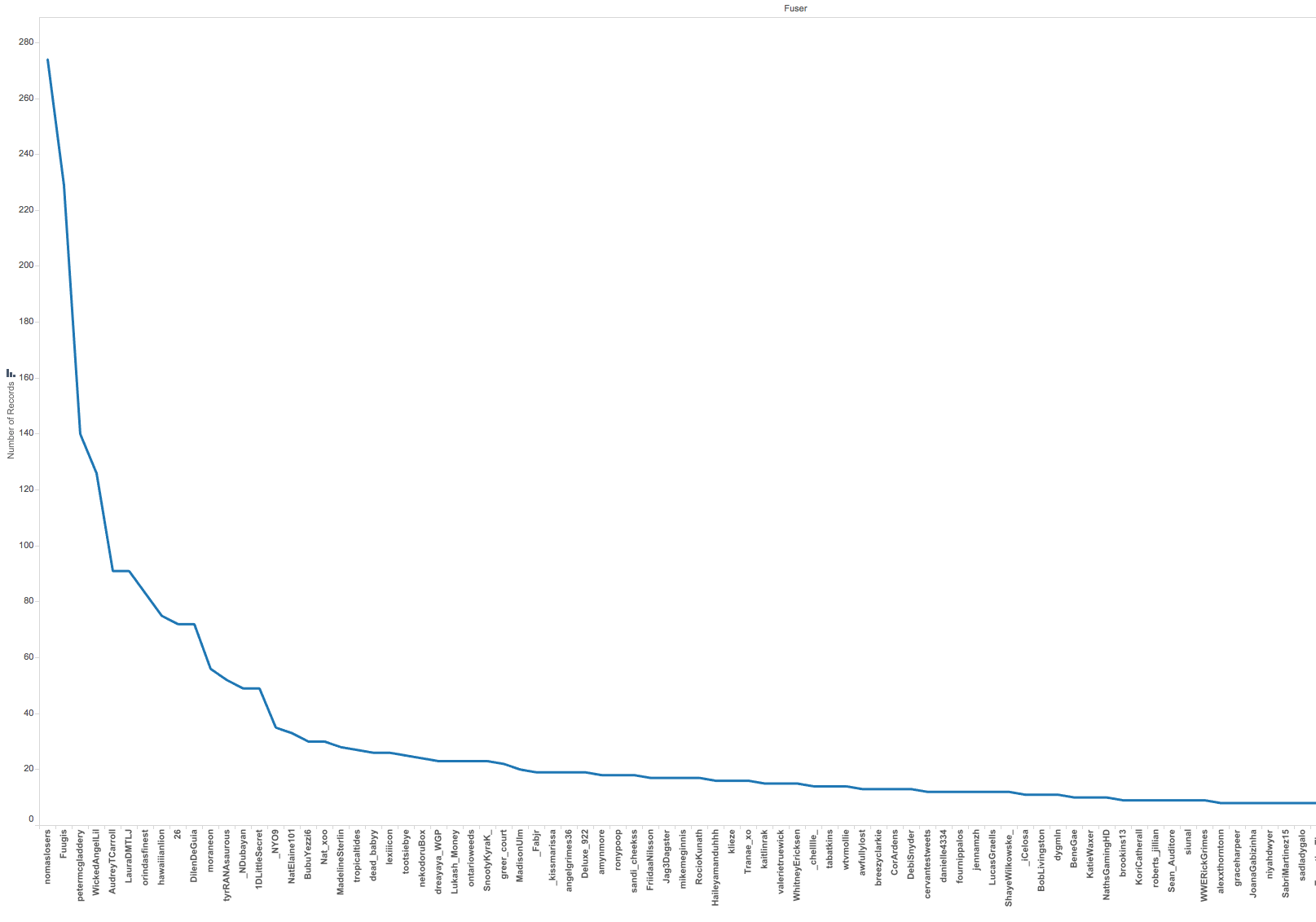


Figure Filtered suers : Freq of reply user ges

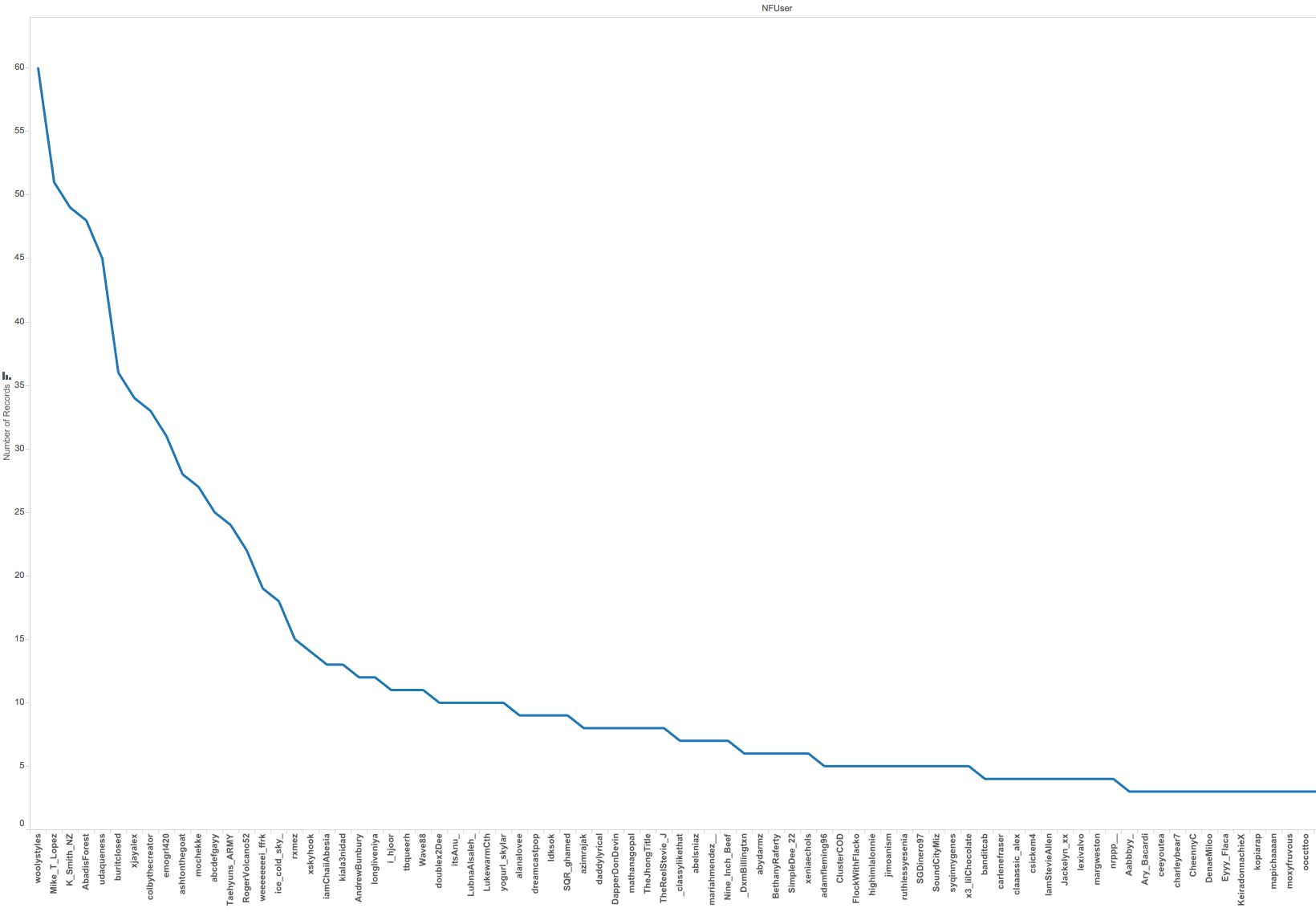


Figure NFUser-Freq of Reply user gets

So, the conclusion which could be drawn was no of reply user get in case of Filtered was higher than the non filtered ones

So we plotted another zoomed in graph, to depict that property

Figure FvsNF Chart of users who get replies

As the graph was clearly depicting that Filtered user were replying to other users in higher frequency as well as getting replies as well.

**Complete**

EVC(Eigen Vector Centrality): Keeping math aside, it is a useful way to find out how important a node is within its network.

To understand EVC, lets understand Degree centrality at first. Typically degree centrality is very useful if you want to get answers to question like

* People who will do favor to you
* People you can spend evening with

e.g :

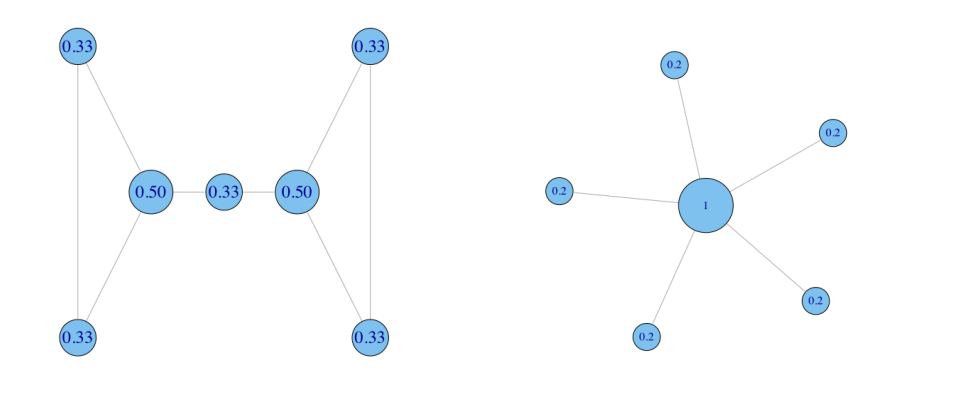


Figure Univ Of Cambridge class notes by Lada Adamic

Then there are other centrality measures like Betweenness centrality measure, Closeness Centrality measure. However what we want is not only to understand that which nodes are more important than others, but also those nodes which connect to more important nodes to figure out the network influencers.

**EigenVector Centrality:** Degree Centrality depends on having many connections. But what if these connections themselves are not well connected. Hence practically “A central node should be one connected to powerful nodes”

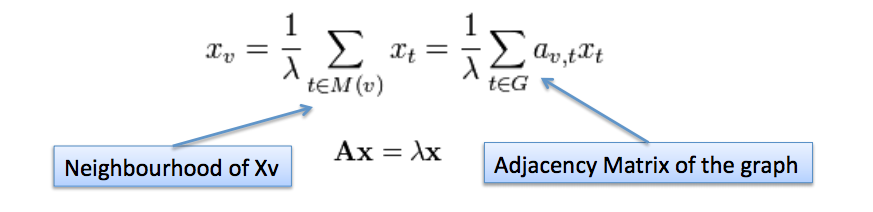


Figure Univ of Cambridge class notes by Lada Adamic

**EigenVector Centrality Algorithm:**

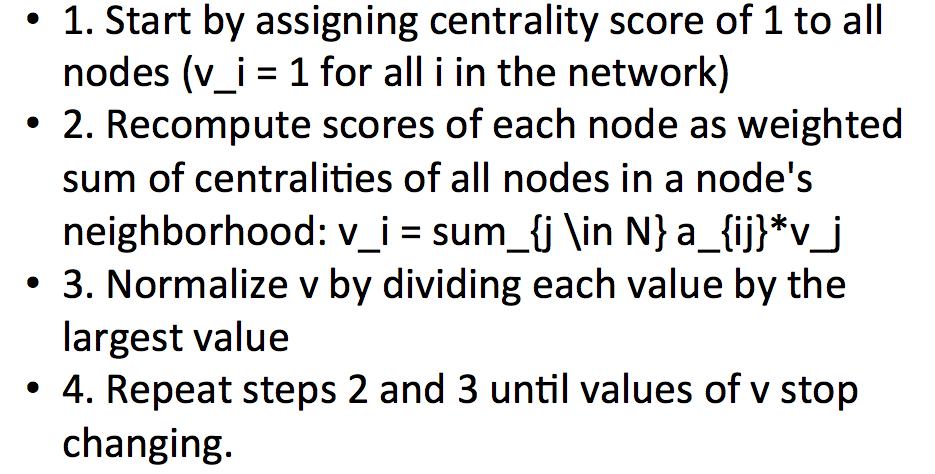
****

Figure Univ Of Cambridge class notes by Lada Adamic