TWITTER SENTINNEL

CS5344 Group Project

Group 15

Felipe Chapa Chamorro (A0179033E)

Gino Martelli Tiu (A0231956Y)

Susan Koruthu (A0231905L)

Widya Gani Salim (A0231857Y)

257M

Vulnerable Twitter users

- 70% more retweets
- 6X more viral
- 10-20x cascade depths

Hence, the onus on online social networks (OSNs) to:

- Monitor network health
- Intervene when required

'Misinformation' and fake news are major threats to civil liberties and democracy'.

- World Economic Forum Global Risk Report 2021

LEARN MORE

Objective and Approach

intent

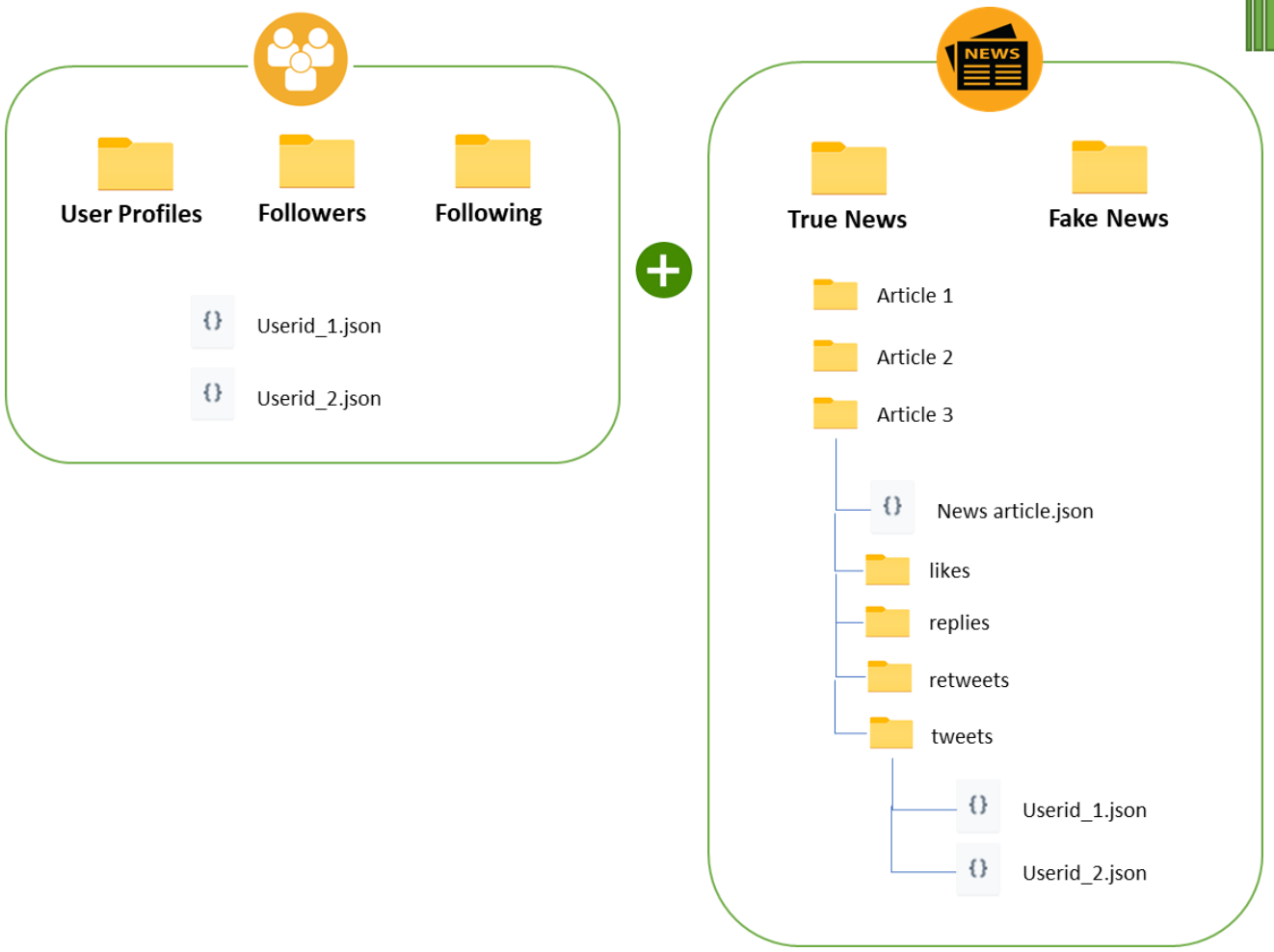
How does Twitter scalably do:

- Community identification
- Influencer identification
- Community state of health
- Malignant network disruption

Steps	Objective	Techniques
1 Dataset and Insights	Generate master dataset from various source files	Tigergraph (Graph studio) Dataframe Analysis Data Mining
2 Network Form	Use of PySpark to shape data into package and algorithm requirements	MapReduce
3 Community Detection	Cluster node edge data into communities and identify points of influence in said communities	Leading Eigenvector Asynchronous Label Propagation Fluid Communities Kernighan Lin Maximization
4 Disruption	Identify malignant cluster and disrupt flow of fake news by removing the least number of nodes possible	Spark graph-frames Targeted attack based on highest degree value

Dataset & Initial Insights

Data pre-processing was done on the below original structure, with 2 key file outputs and statistics:



Tweet_retweet.csv

Follower_followee.csv

Key Statistics:

573,637 unique users

1B+ potential follower-followee matrix interactions

779 news — 47% / 53%

544,027 tweets — 28% / 72%

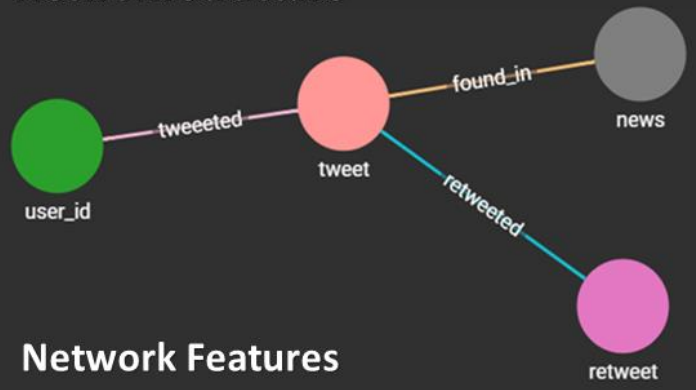
68,099 retweets — 27% / 73%

633,053 t-r edges — 28% / 72%

Fake news *True news*

Dataset & Initial Insights

Network Structure

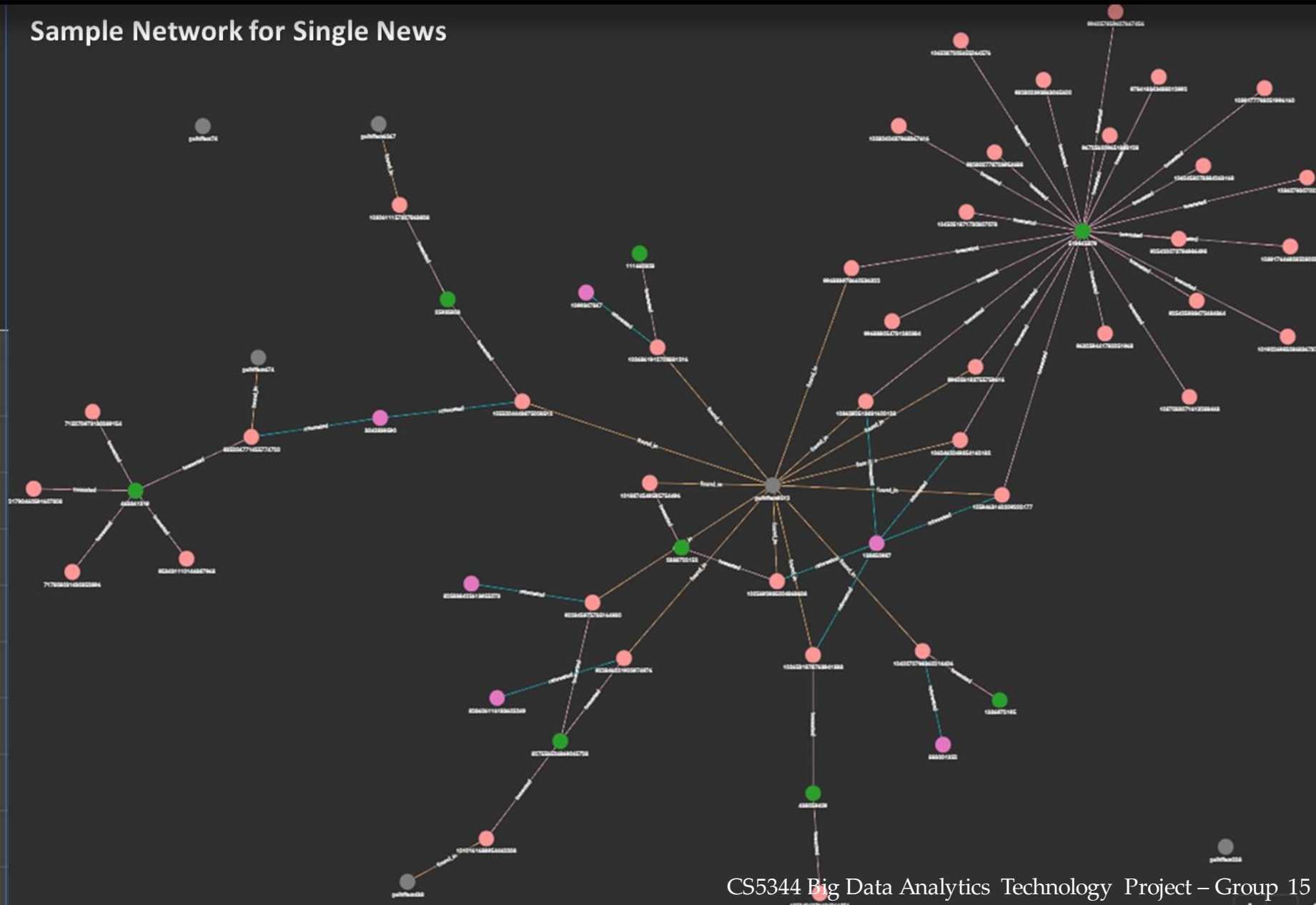


Network Features

Graph statistics

Type	Number
Total Vertex	926,227
Total Edge	1,179,603
Vertex "user_id"	307,584
Vertex "retweet"	68,099
Vertex "tweet"	549,765
Vertex "news"	779
Edge "tweeeted"	521,689
Edge "retweeted"	85,336
Edge "found_in"	572,578

Sample Network for Single News



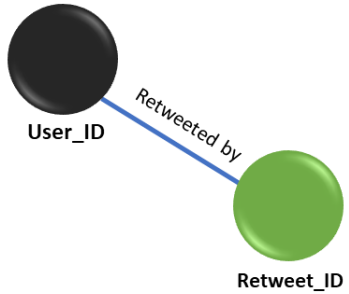
Network Formation

From the initial dataset, 2 network structures were created with the following objectives:

1 Tweet-Retweet Network

Explore community targeting based on retweeting connections

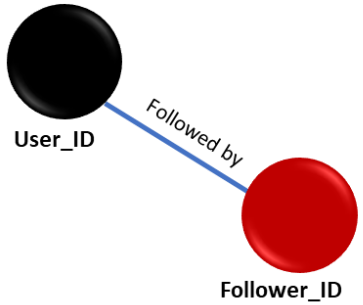
- Does fake news spread through retweets?
- Should we disrupt based on retweets?



2 Follower – Followee Network

Explore community targeting based on follower connections

- Does fake news spread through follows?
- Should we disrupt based on followers?



"It is industry practice to reduce network sizes when doing data mining. We implement a similar approach here."

Steps to reduce network size:

1. Extract only original user and retweet user IDs.
2. Filter to users that have spread fake news at any point in time + connections

Steps to reduce network size:

1. Extract only original user and follower user IDs.
2. Sparse network. Set threshold to users tweeting > 50% fake tweets/total tweets
3. Filter to users to threshold + connections

Community Detection Algorithms

Four network clustering algorithms were explored, with the highlighted one being our model of choice…

"Asynchronous Label Propagation"

- Repeatedly sets label to a node to be the label that appears most among its neighbors.
- Asynchronous because each node is updated without waiting for updates from remaining nodes.

"Kernighan-Lin Maximization"

- Initially divide the community into ~ equal sized groups
- Calculate change in modularity if vertices were moved to another group.
- Choose vertex that increases modularity and move it.
- Repeat process until all vertices have been moved 1X.
- Go through above steps until modularity no longer improves

"Fluid Communities Algorithm"

- Randomly initialize k communities
- Iterate over all nodes as follows until convergence is achieved:
  - ❑ Sum w/ densities of node neighbors
  - ❑ Assign node to community with max density.
  - ❑ Where a node changes communities, adjust vertex density of affected communities

"Leading Eigenvector Algorithm"

- Use a leading vector of modularity matrix and use this to split the network graph.
- Split graph in 2 parts such that there is significant increase in modularity gain.
- Continue splitting until it converges.

Selected Community Detection Algorithm

Algorithm of Choice

Leading Eigenvector Algorithm

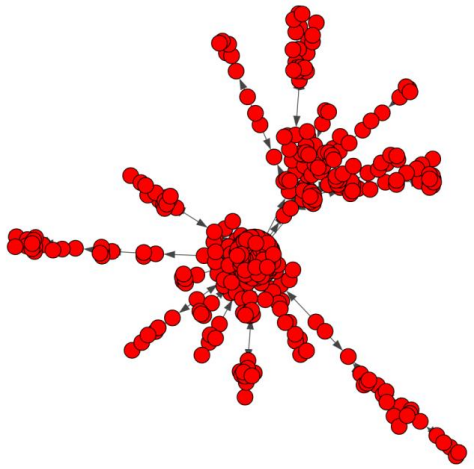
Results:

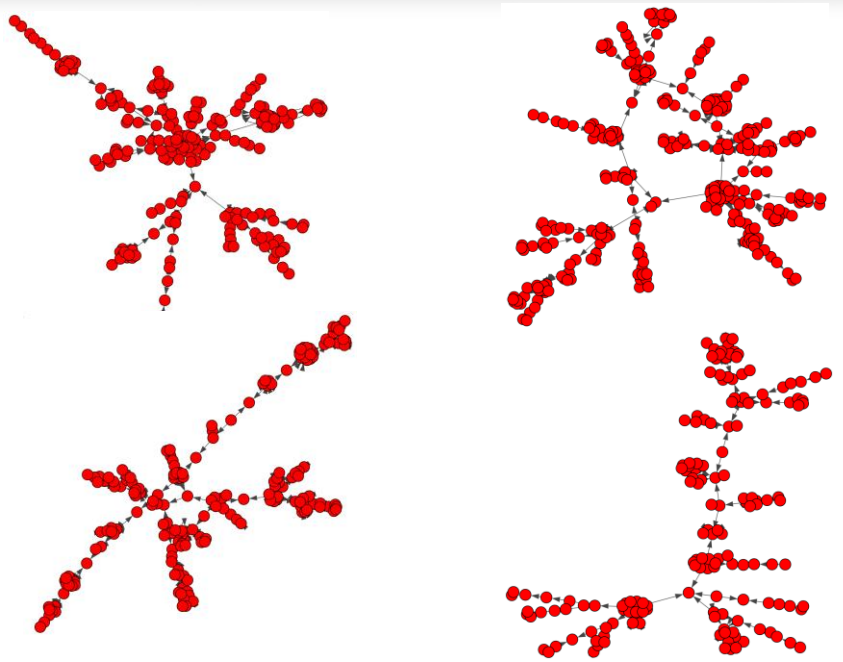
Follower-followee is a less connected network, meaning:

- Fake news sharing occurs across 2+ connection levels
- Tweet-retweet more efficiently links news -> user
- Page rank also showed that influential users in the fake news community were also ranked highly in the overall community. Hence, the importance of effective mitigation measures.

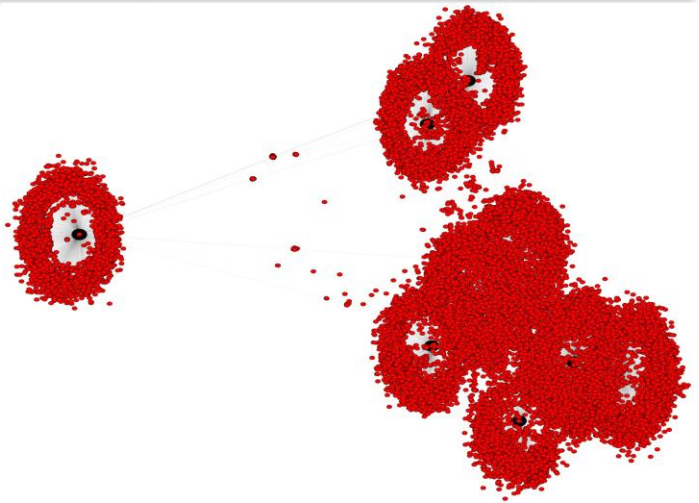
	id	all_news_rank	fake_news_rank
0	7.321097e+17	8	1
1	3.960211e+09	21	2
2	1.418901e+09	17	3
3	7.045636e+17	31	4

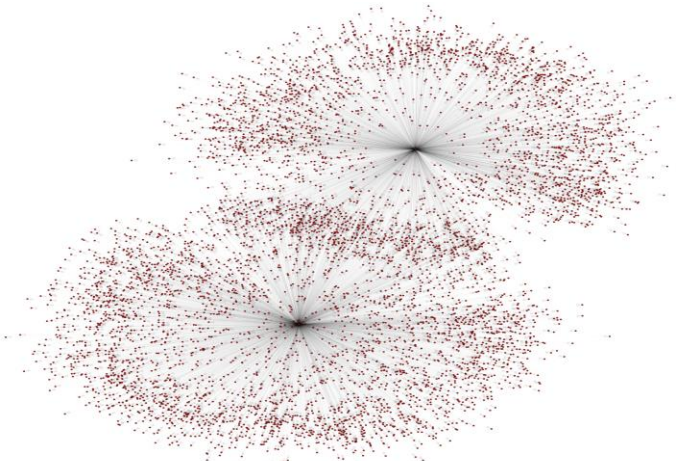
Tweet- retweet network





Follower – followee network





Network Disruption Simulation

- **Two** configurations were tested for network disruption:

Random attack:

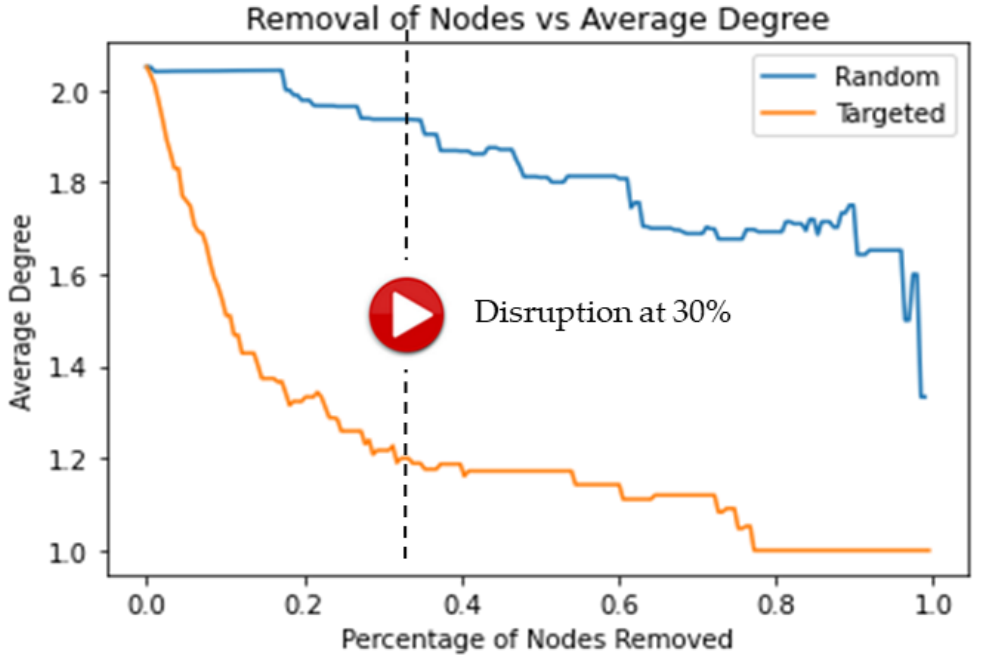
- Remove ~30% of nodes
- Reduce Avg. degree: 2 -->1.8

VS

Targeted attack:

- Remove ~30% of nodes
- Reduce Avg. degree: 2 -->1.2

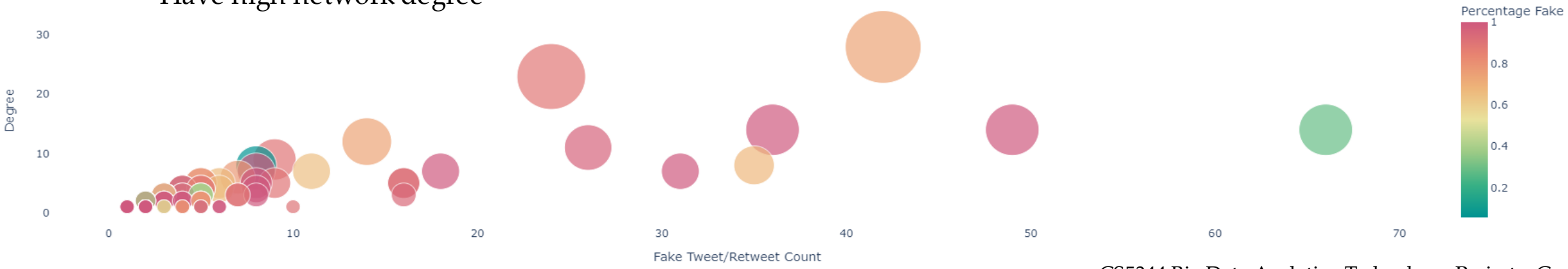
Targeting Tweet Source Nodes



- **How does the targeted attack work?**

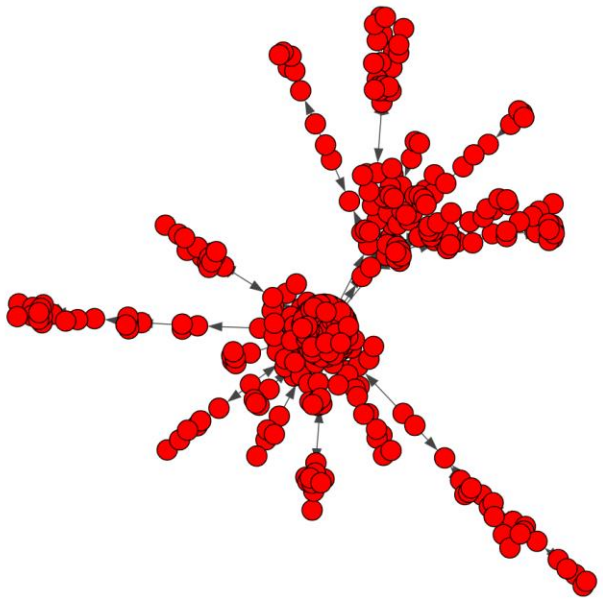
Prioritize nodes with largest bubble size from top-right to left. These nodes have:

- Have the highest fake retweet ratio.
- Have high network degree

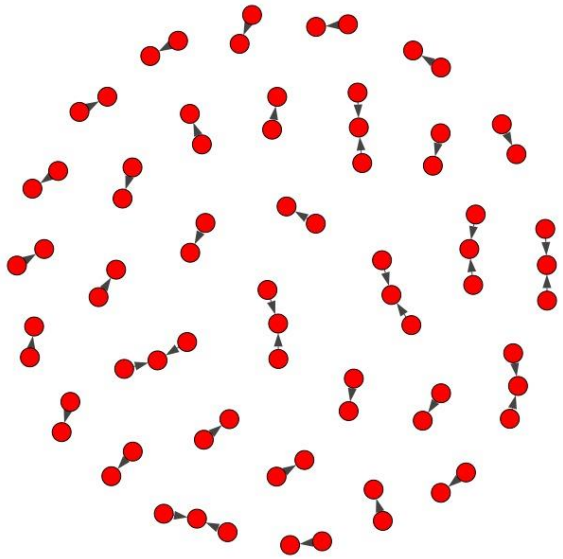


Result & Conclusion

Original



Disrupted



Conclusions:

- **Fake news spread** is better represented through Tweet – Retweet relationships
- Twitter communities can be identified using **Leading Eigenvector algorithm**, based on modularity
- **Identify communities** based on users with high % of fake news tweets
- **Community disruption is possible**
  - Targeted attack based on vertex degree
  - **Reduce spread of fake news by 40%** (Avg. degree 2→1.2)