Sarah Kim
Anay Gupta
Group 12
CSE 472 - Prof. Han Liu
Project 1 Report

COVID-19 Disinformation Detection on Social Media

Step 1: Dataset Collection and Storage

First and foremost, a list of COVID-19 claims and ground truth labels were scraped from two fact-checking websites: Poynter and the World Health Organization. While Poynter was only crawled for the most recent claims in the United States, the World Health Organization website was used to extract their renowned mythbusters.

Using XPath rules, approximately 150 claims were extracted from Poynter and 27 claims were extracted from the WHO. The claim title and the validity of the claim were scraped from Poynter and the same was scraped from WHO. These were the **news nodes.**

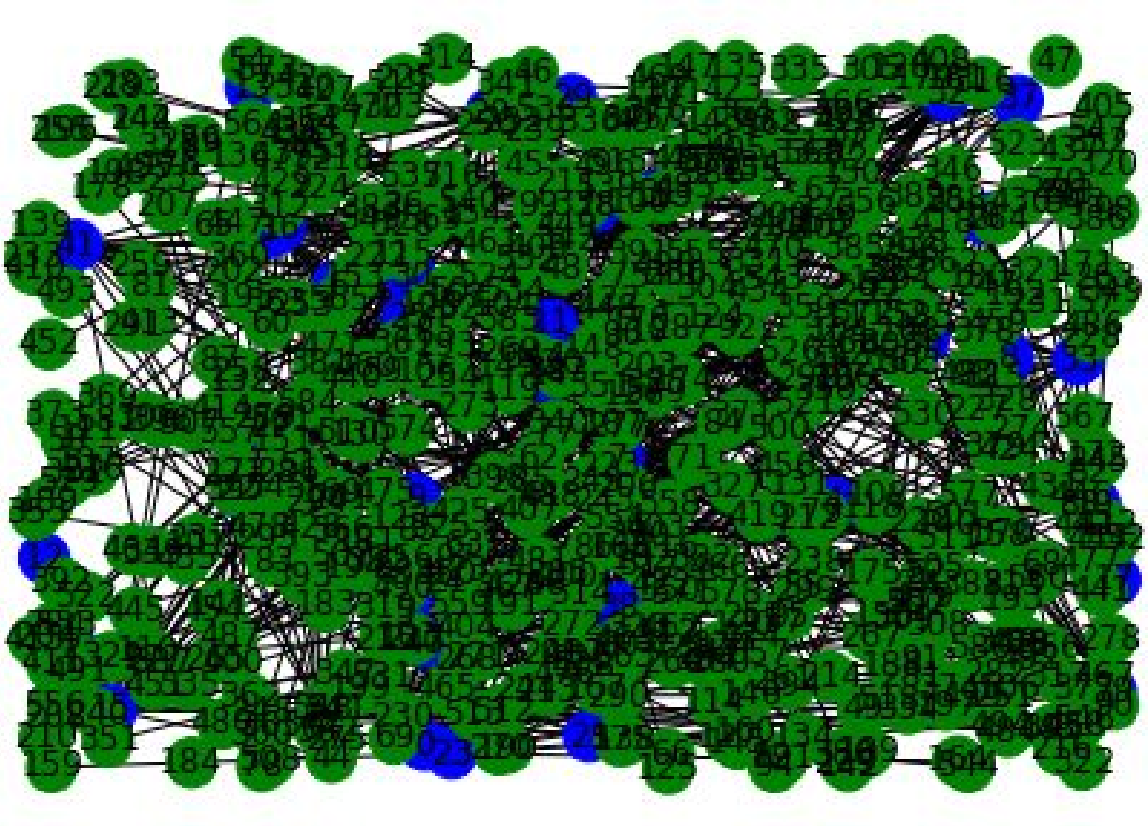
Once the ground truth dataset had been collected and processed, Twint was used to query public tweets on Twitter. Due to Twitter API bandwidth problems, only ~50 out of the 150 claims were searched against Twitter using Twint's python library and JSON results were retrieved. These JSON results indicated tweets that contained some substring of the fact-checked clai. The ids for the users associated with each tweet were extracted from this JSON and mapped to their respective claims. These were the **user nodes**.

Step 2: Graph Construction

Once the news and user nodes had been processed and consolidated, the NetworkX python library was used to create nodes and edges for an undirected graph. Approximately 50 claims (news nodes) were plotted and 533 tweets (user nodes) were plotted. This amounted to a total of 583 nodes and 589 edges in the fully built graph.

Below is the generated (via Networkx) heterogeneous graph with news and users as node types and the news-user tweeting relationship as edge type. The green nodes are user nodes and the

blue nodes indicate news nodes.



Step 3: Graph Analysis

Using the Networkx python package, the built graph's clustering coefficient and reciprocity were computed. The results are detailed below:

Clustering Coefficient: 0
Reciprocity: 0

*This was due to the lack of user-user edges in the built graph.