Mapping the echo-chamber
COVID-19 edition

Manuel Ivagnes
Riccardo Bianchini
Valerio Coretti

La Sapienza University of Rome — July 23, 2020

Contents

1 Background

"An echo chamber is a metaphorical description of a situation in which beliefs are amplified or reinforced by communication and repetition inside a closed system and insulates them from rebuttal." - wikipedia

Taking inspiration from this sentence, and the assigned paper [1], we built a simple model that can find the polarization of the communities inside the *Twitter* social network regarding *Coronavirus* related topics. Indeed, with social media, misinformation about COVID-19, or vaccines, for example, can reach huge audiences and circulate very quickly. Therefore, it is crucial to find ways to recognize reliable information.

2 Input Data & Preprocessing

Network seed dataset:

- The "**Coronavirus Tweet Ids**" [2] dataset from Harvard University [version 1][1]
  - Contains the ids of 51,798,932 tweets related to Coronavirus or COVID-19;
  - Collected between March 3, 2020 and March 19, 2020 from the Twitter API using Social Feed Manager;
  - Collected using the POST statuses/filter method of the Twitter Stream API, using the track parameter with the following keywords: #Coronavirus, #Coronaoutbreak, #COVID19;
  - The list of identifiers is split into 6 files of up to 10 million lines each, with a tweet identifier on each line;

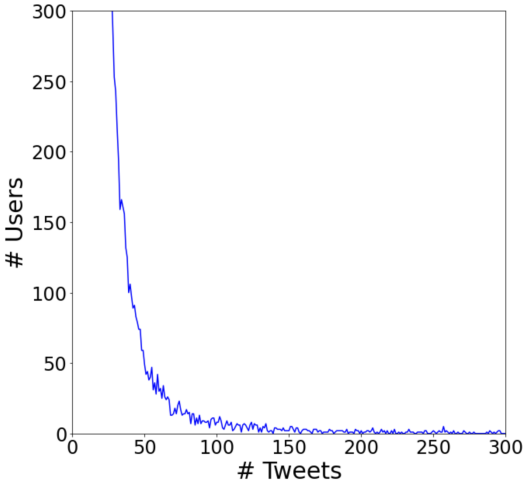
Per Twitter's Developer Policy, tweet ids may be publicly shared for academic purposes; tweets may not. Thus, as first step, we hydrated the tweet ids in the dataset using *Twarc* [3] and stored them as JSON Lines[2]. Then, to avoid preeminence of language related clusters [4], we deleted all the tweets having a language different than English. On average, this process has halved each file of the dataset. Unlike the paper, since our seed database is keyword driven, we did not search for bias or partnership with given domains. This resulted in $2,480,875$ complete tweets from $1,169,150$ unique accounts.

Distribution mapping how many accounts shared the same number of tweets

```
First 20 values:

#1 795383
#2 186736
#3 68835
#4 34746
#5 20654
#6 13716
#7 9470
#8 6900
#9 5168
#10 4056
#11 3209
#12 2534
#13 2107
#14 1796
#15 1463
#16 1253
#17 1074
#18 953
#19 869
#20 691

90.0% of accounts have less than 4 tweets
```



[1] This interval corresponds to the period around the declaration of pandemic by the OMS, which we believe leads to more interesting results in terms of semantic deviation.

[2] Since this procedure requires several hours, we decide to keep only 1 every 10 ids in each file. This may lead to information loss, however it is acceptable for the purpose of the experiment.

Most cited domains, and single pages

```
#1 => twitter.com: 239036
#2 => bit.ly: 23535
#3 => youtu.be: 7186
#4 => paper.li: 5696
#5 => ow.ly: 4154
#6 => www.instagram.com: 4079
#7 => www.pscp.tv: 3985
#8 => buff.ly: 3101
#9 => trib.al: 2919
#10 => www.youtube.com: 2150
#11 => reut.rs: 1873
#12 => www.theguardian.com: 1696
#13 => www.washingtonpost.com: 1503
#14 => www.nytimes.com: 1297
#15 => lnkd.in: 1286
#16 => dlvr.it: 1093
#17 => tinyurl.com: 1080
#18 => www.cnn.com: 1076
#19 => www.facebook.com: 1036
#20 => www.bbc.co.uk: 1024
```

```
#1 => https://twitter.com/i/events/1219057585707315201: 2720
#2 => https://twitter.com/maxbrooksauthor/status/1239624352305303552: 2537
#3 => https://twitter.com/BrookeGMcDonald/status/1238986272137502720: 2100
#4 => https://twitter.com/Reuters/status/1239637550828064769: 1761
#5 => http://bit.ly/337yabc: 1702
#6 => https://trib.al/vVSjvun: 1438
#7 => https://twitter.com/messages/compose?recipient_id=835740314006511618
#8 => https://twitter.com/redfishstream/status/1238436668102893568: 1210
#9 => https://twitter.com/LizSpecht/status/1236095180459003909: 944
#10 => https://twitter.com/adamclarkitv/status/1236289649737371648: 934
#11 => https://twitter.com/JasonWhitely/status/1238986444615618561: 884
#12 => https://twitter.com/julialindau/status/1235714275752267776: 858
#13 => https://twitter.com/silviast9/status/1236933818654896129: 777
#14 => https://twitter.com/DrPeckPNP/status/1244062665535864832: 749
#15 => https://twitter.com/balazscseko/status/1244612142831198209: 735
#16 => https://twitter.com/JohnCornyn/status/1238878952644624390: 707
#17 => https://twitter.com/weijia/status/1239923246801334283: 698
#18 => https://www.pscp.tv/w/cT7j5TFsWkVwZ2xwWnZvam58MU1ZeE5remVxRG9Ldww4F
#19 => https://twitter.com/AnaCabrera/status/1238126303238410243: 661
#20 => https://twitter.com/PalliThordarson/status/1236549305189597189: 642
```

Most popular hashtags (dataset keywords removed, everything lowercase)

```
#1 => covid—19: 42470
#2 => coronavirusoutbreak: 25829
#3 => dontbeaspreader: 19885
#4 => coronavirusupdates: 19679
#5 => breaking: 19472
#6 => covid_19: 19299
#7 => china: 18614
#8 => italy: 18133
#9 => covid2019: 16393
#10 => coronapocalypse: 12820
#11 => iran: 11021
#12 => corona: 10400
#13 => covid: 9712
#14 => coronaviruspandemic: 9559
#15 => coronavirusupdate: 9051
#16 => pandemic: 8362
#17 => socialdistancing: 8332
#18 => trump: 8230
#19 => flattenthecurve: 7899
#20 => wuhan: 7582
```

```
#21 => coronapocolypse: 5740
#22 => quarantine: 5328
#23 => stayhome: 5313
#24 => lockdown: 4350
#25 => quarantinelife: 4215
#26 => coronavirusuk: 4016
#27 => nyc: 3881
#28 => stayathome: 3684
#29 => sarscov2: 3545
#30 => coronavirusindia: 3481
#31 => india: 3177
#32 => coronavirususa: 3155
#33 => us: 3056
#34 => france: 2929
#35 => health: 2845
#36 => who: 2837
#37 => shipsgoingdown: 2827
#38 => wuhanvirus: 2584
#39 => auspol: 2573
#40 => covid19malaysia: 2566
```

Word2Vec dictionary dataset:

- "**Dataset7: TweetDataWithoutSpam+GeneralData_Word**" from the "Word Embedding Data Sets Learned from Tweets and General Data" by Research and Development at Thomson Reuters [5] (same as original paper)
  - Set of pre-trained vectors created on a corpus of 198 million tweets
  - 6.7 billion words from General Data, and 9.5 billion words in the whole training data
  - 1.7 million of unique words or (words+phrases) in the trained embedding model
  - Vector dimension size 300, Word and phrase frequency threshold 10, Learning context windows size 8

In this case, we did not need preprocessing. Here some insights on the reasons behind this dataset, directly from its reference paper [6]:

Tweets are noisy, short and have different features from other types of text. Because of the advantages of applying word embeddings in NLP tasks, and the uniqueness of tweet text, we think there is a need to have word embeddings learned specifically from tweets (TweetData).

[...]

Some applications or NLP tasks may need to deal with both tweet data and general-domain data, and a word embedding data set combining these two types of data may provide better performance than the one learned from just one of them.

3 Graph Construction & Analysis

We opted for a step by step construction, producing different intermediate output files. Thus, each script takes as input a json file that is the result of a previous script.

Step by step graph construction:

1. **accounts_extractor** makes a list of all the users producing the output file *accounts.jsonl*;
2. **sample_accounts_tweets** sample 30000 accounts, saving them in *sampled_accounts.jsonl*. Then, it extracts all the corresponding tweets from the dataset and save them in *sampled_tweets.jsonl*;
3. **domains_index** builds a pseudo inverted index $< domain : users >$, which makes easier to find the connections derived by common domains shared. It is saved in *inverted_domains.jsonl*;

   Note: For the 10 most popular domains, which corresponds mostly to social networks and URL shortening, we linked the users only if they shared the same page (full_url).
4. **hashtags_index** builds a pseudo inverted index $< hashtag : users >$, which makes easier to find the connections derived by common hashtags used. It is saved in *inverted_domains.jsonl*;

   Note: We removed all tags similar to dataset keywords in the list of 60 most used.
5. **retweet_mentions_extractor** finds all the connections given by retweets and mentions. Then save the list in *retweet_mentions.jsonl*;
6. **graph_construction** build the graph, then extract the *Giant component*, which is a connected component of a network that contains a significant proportion of the entire nodes in the network;

```
Initial                      # nodes: 0,     # edges: 0
After domains                # nodes: 2995,  # edges: 18252,   avg. clustering coeff: 0.731353829365398
After hashtags               # nodes: 10298, # edges: 441348,  avg. clustering coeff: 0.779130029889205
After retweets/mentions      # nodes: 10938, # edges: 442391,  avg. clustering coeff: 0.7294592359119989
Giant component              # nodes: 10162, # edges: 441707,  avg. clustering coeff: 0.7613322629146598
```

Once built the graph, we applied the **Louvain method**, which is a common algorithm for detecting communities in networks. It maximizes a modularity score for each community, where the modularity quantifies the quality of an assignment of nodes to communities. This means evaluating how much more densely connected the nodes within a community are, compared to how connected they would be in a random network[3] [7].

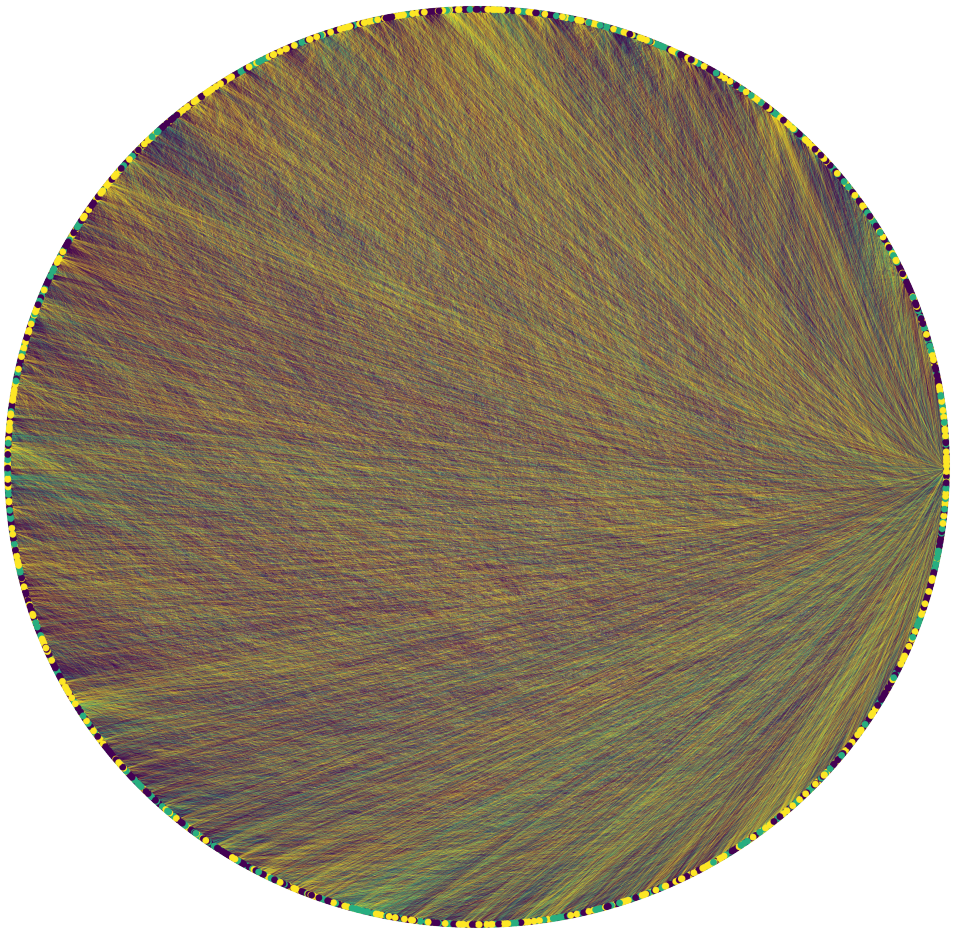
We first made a simple implementation from scratch of the algorithm[4], which found over 4000 communities, many of which with a very low number of nodes. Then, we used the optimized implementation provided by the *community detection* library for Python. The method found 39 *communities*, from which we selected the three ones with the highest number of nodes:

- *yellow*: 1577 nodes
- *violet*: 1240 nodes
- *green*: 990 nodes

For a total of 3807 nodes and 48328 edges.

[3] For the full explanation refer to the corresponding file inside the *community_detection* repository on the github experiment page

[4] It is only for academic purpose and does not take in consideration possible optimizations



4 Semantic Deviation

Once mined the communities, we made a model (using Word2Vec) to identify the semantic deviation of each community compared to the global norm, in order to know which subjects or topics are prone to be construed differently by a given echo-chamber[5] [8].

As first step, we split the tweets made by the accounts in each selected community on different files. Then, we applied on each file a specific preprocessing, composed by the following steps:

- Remove all duplicates and null text values in the data
- Remove emoticons, URLs and hashtag
- Remove stop words and alphanumerical words, apply lower-casing, remove punctuation and extra white spaces.
- Remove resulting empty rows

At this point, we trained the 3 different Neural Network implementing the Word2vec, one for each community, using the methods provided by the *Gensim* library.

[5]Also in this case, for the full explanation refer to the corresponding file inside the *Word2vec* repository on the github experiment page

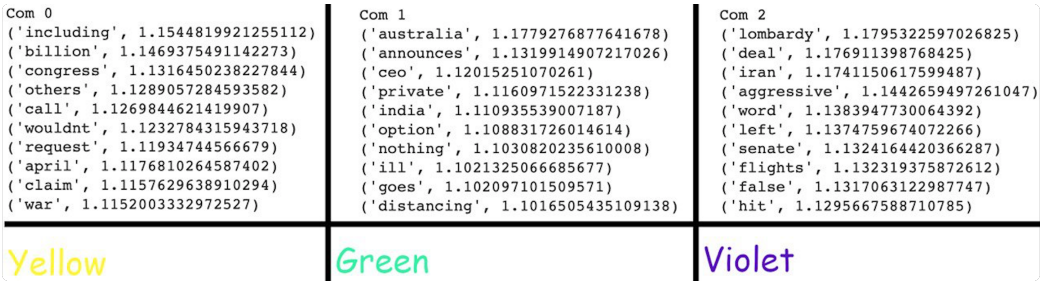
Finally, we computed the deviation of each term t from community c, with respect to the global dataset (section 2), using

$$dev(t, c) = cosine\_distance(v_t^c, v_t^g)$$

where v_t^g is the vector representation of t in the model created for community c, and g is the global model. Thus, we can define the maximum deviation of a term as the community with the largest deviation for that term

$$max\_dev(t) = \text{argmax}_c\, dev(t, c)$$

The 10 most deviating terms in each community are



The terms that show marked deviation in a certain community, but not in others, defined as clusters of 3 neighboring terms that all have the highest distance from the global vector are

Community	Yellow	Green	Violet
Deviated Terms	billion, telling, respond italians, decision, fighting wuhan, germany, feb	australia, positive, disease india, flu, things smart, negative, epidemic	flights, word, small country, iran, nation lombardy, director, later

5 Conclusion

In conclusion, our experiment confirmed the followings:

- We can identify latent networks or communities making echo-chambers inside the Twitter social network without any need of supervision;
- For a given echo-chamber, we can automatically identify topics or phrases that the community is vulnerable to spreading misinformation about, by analyzing the distribution of vector representations of messages;

Possible future work:

- Expand the dataset;
- Increase the number of accounts considered to make the graph;
- Classification of users attitudes and sentiment [9]

Note: We do not aim to open a debate on social media misinformation during the pandemic period, indeed our analysis considers only partial information. Thus, take in consideration that our result cannot be used as a real estimator.

References

[1] Armineh Nourbakhsh, Xiaomo Liu, Quanzhi Li, Sameema Shah, *Mapping the echo-chamber: detecting and characterizing partisan networks on Twitter*, Research and Development, Thomson Reuters

[2] https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/LW0BTB

[3] https://github.com/DocNow/twarc

[4] Svetlana S. Bodrunova, Ivan S. Blekanov, Mikhail Kukarkin, *Language and Sentiment Structure of Twitter Discussions on the Charlie Hebdo Case*, from *HCI International 2018 – Posters' Extended Abstracts: 20th International Conferences*, HCI International 2018, Las Vegas, NV, USA, July 15-20, 2018, Proceedings, Part I

[5] Li, Quanzhi. (2017). Word Embedding Data Sets Learned from Tweets and General Data [Data set]. Zenodo. http://doi.org/10.5281/zenodo.581402

[6] Quanzhi Li, Sameena Shah, Xiaomo Liu, Armineh Nourbakhsh, emphData Sets: Word Embeddings Learned from Tweets and General Data, Research and Development, Thomson Reuters https://arxiv.org/pdf/1708.03994.pdf

[7] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte and Etienne Lefebvre, *Fast unfolding of communities in large networks*, Department of Mathematical Engineering, Université catholique de Louvain

[8] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, *Efficient Estimation of Word Representations in Vector Space*

[9] Hywel T.P. Williams, James R. McMurray, Tim Kurz, F. Hugo Lambert, *Network analysis reveals open forums and echo chambers in social media discussions of climate change*