Social Property Based Media Data Set Creation

Yishi Wang, Youwen Zhu
Atul Prakash, Roya Ensafi
31 June 2019

***Abstract*— As tons of message rush to people frequently from various news websites and social media throughout the time, there is an urgent need of "neutral" and unbiased information for consumption by the general public. With the advancements in machine learning algorithms, more and more researchers have suggested more practical methods of characterizing and detecting false or misleading information. We aimed to look at the interaction between news articles' authors and their audiences on Twitter. We crawled tweets of major editors and reporters and user information of each account from Twitter and built up a data set. The data set provides communications properties, which can be used for further social media analysis. We also elaborate on the intricacies of dealing with data.**

I. Introduction

With the fast development of Internet, People nowadays can get information anywhere at any time, while with the spurt of social media, such as Twitter, Facebook, and Instagram, it costs little to spread information. Therefore, the social media has become one of the significant methods for people to share their daily life, opinions, and news and, in return, people now are receiving tons of information fragments every day from the social media.

On one hand, the convenience of sharing through social media does improve human's life quality by encouraging the communication among, for example, different generations, different organizations, and different countries, eliminating certain misunderstandings and help the society run smoothly. Many organizations, including governments, also use social media to broadcast their opinions and strengthen the connection between ordinary people for more support. By the time of 2013 Australian election, Australian Labor Party (ALP) Prime Minister (PM) Kevin Rudd had been returned to the Prime Ministership in a leadership spill against party colleague Julia Gillard earlier that year, arguably because his continuing social media popularity was seen as an indicator of his greater popular and electoral appeal [1]. By 2013, the majority of federal members and candidates operated Facebook and/or Twitter profiles (if with widely varying degrees of care and enthusiasm), and many party organizations paid close attention to their social media campaigning activities[1].

On the other hand, however, because of the little cost of spreading information on social media, people can be easily get exposed to biased, misleading, or even fake information. For example, the news reporters are likely inclined to report news more vividly that are more fit into its political bias while understating the news that they do not like. The politicians can also criticize their rivals for doing something immoral even if that is not true, or they omit the positive effects. What is worse, there are also lots of bot accounts that help to spread fake or biased news, and the high retweet counts of these fake tweets can be very convincing. With all the biased or unreal information spreading on the internet, people can have a hard time understanding what is truly going on. Though the social media do pay attention in dealing with those rumors for a healthier internet environment, the users can still often see this information before its effect gets too severe, which still can probably cause panic or misunderstanding. Moreover, in many cases, the massages can be painful to judge as true or false, which can exist on the internet for a longer time, affecting more users.

Many researchers have worked on information bias detection, and truth detection, including information from news websites and social media. We will introduce details at previous work part. However, there are still no practical ways in daily life for people to immediately get the truth hidden under all kinds of biased or fake information. Therefore, we get the idea to collect data from social media considering the spread of news and communication property. We focus on the US political-related information detection, including news from two major news websites(FOX NEWS and CNN NEWS) and the tweets of reporters or editors from these two websites. The language of these news websites is worthwhile to pay more attention to explore since the language they use is relatively more formal, and sometimes can be more confusing, making the hidden bias (if any) be more likely affect the viewers unconsciously compared to other kinds of news or tweets. Besides, since many researchers have worked on political bias and have got the political bias of websites, it is convenient to use these results as ground truth for our later information analysis.

For users, they can get more social background information(like media bias) about the news writer and other news articles written by editors with different bias about the same event.For other researchers, our database and codes can provide more social properties than common database.

II. previous work

A. Semantic Features

In the paper Credibility Assessment of Textual Claims on the Web [3] authors suggest clustering the news on their content first and then extract features from each cluster. They did not They consider the communication property and focused on the features instead: writing style and website source. So

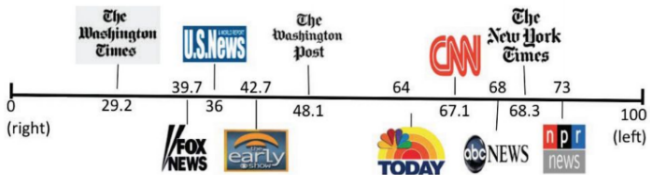

Fig. 1. The bias of medias, the score 0 means strongly conservative and 100, strongly liberal

they clustered an article($a_{ij} \in A$) with its claim($c_i \in C$) and source website($WS_j \in WS$).

B. Other Features

Hardalov1, et al. [2] also use the text contest as features to classify the credibility of news, but instead of words they worked with Credibility Features:

1) Length of the article (number of tokens);
2) Fraction of words that only contain uppercase letters;
3) Fraction of words that start with an uppercase letter;
4) Fraction of words that contain at least one uppercase letter;
5) Fraction of words that only contain lowercase letters;
6) Fraction of plural pronouns;
7) Fraction of singular pronouns;
8) Fraction of first person pronouns;
9) Fraction of second person pronouns;
10) Fraction of third person pronouns;
11) Number of URLs;
12) Number of occurrences of an exclamation mark;
13) Number of occurrences of a question mark;
14) Number of occurrences of a hashtag;
15) Number of occurrences of a single quote;
16) Number of occurrences of a double quote.

And there are some sentiment-polarity features which is popular in NPL researches:

1) Proportion of positive words;
2) Proportion of negative words;
3) Sum of the sentiment scores for the positive words;
4) Sum of the sentiment scores for the negative words.

Analyzing features mentioned above can get the attitudes of the text. But the authors did not take the communication property such as spreading network into consideration.

C. Visualizing Media Bias through Twitter

An et al. [4] worked on visualizing the media bias through Twitter. They created a political dichotomy map of media sources on Twitter, as shown in Fig. 1. Their basic idea is to calculate the closeness according to the audiences overlap. The closer the media are, the more audiences they share. The result reveals the relationship between audiences and bias of accounts, which is also roughly fit the ground truth of bias of each media evaluated by previous work. However, they mainly focused on the audiences of social media while did not care much about the contents of their tweets and the whole picture of the events which should exist in the original news websites. And they did not consider the different weights played in opinion shaping by different users.

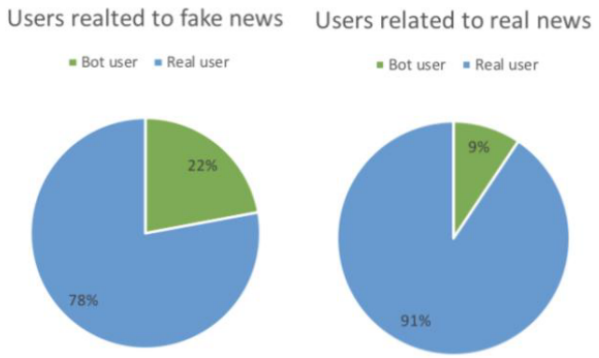

Fig. 2. Comparison of bot scores on users related to fake and real news on PolitiFact dataset

D. echo Chamber

Kai Shu et al. [6] introduced a new concept called echo Chamber. Social media provides a new paradigm of information creation and consumption for users. As Kai Shu et al. [6] illustrated, consumers are selectively exposed to certain kinds of news because of the way news feed appears on their homepage in social media. For example, users on Facebook always follow like-minded people and thus receive news that promotes their favored existing narratives. Therefore, users on social media tend to form groups containing like-minded people where they then polarize their opinions, resulting in an echo chamber effect, which means that people in the echo chamber are more likely to be biased and believe the fake news because they perceive people they are following as credible, and users can see the news frequently from the accounts they are following, increasing the spread of the biased or fake news.

E. FakeNewsNet

Shu et al. [5] built up a database containing information from news content, social context, and spatiotemporal information. They checked the news that had spread on the social media according to PolitiFact and GossipCop and attached a true or false label to each news they have collected. Besides, they collected information about user engagement such as user profile, their posts, replies, and networks as well as timestamp and geoinformation. By doing some fundamental analysis of their data set, they found that the bot users are more likely to involve in the information spreading of fake news compared to that of real news (showed in Fig.2). So those bots can, therefore, give a false impression that information is highly popular and endorsed by many people, enabling the echo chamber effect for the propagation of fake news mentioned above.

III. BACKGROUND

We use python3 as our language and following are the packages we used for our programming and other technologies we used during our research.

1) Tweepy
   Tweepy is an convenient tool for Twitter data extraction. It provides easy-to-use function to get access to Twitter data through Twitter handles.

2) JSON
   JSON (JavaScript Object Notation), specified by RFC 7159 (which obsoletes RFC 4627) and by ECMA-404, is a lightweight data interchange format inspired by JavaScript object literal syntax (although it is not a strict subset of JavaScript 1)[7].
   JSON exposes an API familiar to users of the standard library marshal and pickle modules.

3) argparse
   The argparse module makes it easy to write user-friendly command-line interfaces. The program defines what arguments it requires, and argparse will figure out how to parse those out of sys.argv. The argparse module also automatically generates help and usage messages and issues errors when users give the program invalid arguments[8].

4) Google Cloud Platform
   Google Cloud Platform provides various tools for data analyze including the Bigquery and the Compute engine and can run all the time. Therefore, it is more convenient to store the data extracted either from news websites or social media on Google Cloud Platform for further study.

IV. METHOD

We have already crawled the news website from CNN and FOX news as well as finding the Twitter handle of each author to build up a network of news and tweets in the future.

Then we started to crawl tweets and user information from Twitter. Here is basic logic of our code:

1) Connect to Twitter using Twitter developer account
2) Visit targeted accounts' page
   a) Save the tweets containing full texts of targeted accounts on a regular basis into a JSON file
   b) Collect the followers and followings on a regular basis into a JSON file
3) upload the data to the Bigquery in Google Cloud Platform

After getting the raw data in JSON format, we started to analyze the form and try to find some characteristics or differences between different biases. We managed to formalize the columns of each tweet from original extended mode data and then read papers and some possible features that can be used for bias or fake news detection.

We structured the user table with the following schema:

CREATE TABLE users(

screen_name VARCHAR NOT NULL,

#the unique user name of twitter accounts

followers VARCHAR ARRAY,
#the followers, in ID form, that are following the target account

);

We also structured the tweets table with the following schema:

CREATE TABLE tweet(

tweet_id VARCHAR NOT NULL,
the unique number string of the tweet

time_stamp DATE
#time of the tweet was tweeted

screen_name VARCHAR NOT NULL,
#the name of the account that made this tweet

location VARCHAR,
#location of the users when the account made this tweet

text VARCHAR,
#full text of the twitter

retweeted_id VARCHAR,
#id of the original tweet if this tweet is a retweet. "none" if it is not

quoted_id VARCHAR,
#id of the original tweet if this tweet is a quote. "none" if it is not

mentions VARCHAR ARRAY,
#account(s) that is(are) mentioned in this tweet

hash_tags VARCHAR ARRAY,
#the hash tags that appear in the this tweet

urls VARCHAR,
#the links contained in this tweet.

media_urls VARCHAR,
#the links of videos or images contained in this tweet.

retweet_acount VARCHAR,
#the times of this tweet was retweeted .

);

In the beginning, we tested the python code on @realdonaldtrump(the screen name of the account), which is the account of Donald Trump, and we picked the columns we need using Twitter handle provided. Then, we made two lists based on the two news websites: FOX NEWS and CNN NEWS: After doing some research on these two websites, we picked 30 people each from those sites into the lists, including the editors and reporters that recently published political news or articles. Then we extract their twitter information into JSON files and upload them to the Google platform. Considering the time effect and the uncertainty of Twitter, in order to get the most updated information, we also tried to crawl information of the target Twitter accounts regularly. After getting the twitter information as well as the news articles information, we constructed a connected data set between news and tweets.

V. Results

Our work is mostly the exploration work of data extraction as well as data management. The following results shows that it is possible to build up a regularly updated Twitter database connecting the news websites for further study.

A. Tweets download

There are three types of tweet:

1. the tweet that was posted by the account itself
2. the tweet that was retweeted from other accounts without a comment
3. the tweet that was retweeted from other accounts with a comment

Fig.3, Fig.4, and Fig.5 are the examples mentioned above from @realdonaldtrump. Accounts that mentioned in tweets will start with @ sign, and the hashtag will start with # sign. Each original tweet contains the name in bold fonts, screen name, which is the unique name of each account put using small grey words, the timestamp, and the texts. The videos, images as well websites shown in the tweets will be stored as URLs in our final JSON file. Fig.6 represents a part of the follower list of @realdonaldtrump. Each row contains the name, screen name and a short self-introduction (can be null).

It is inefficient to use the raw data got through the Twitter handle for further analysis. The extraction function has two modes —— normal mode and extended mode. The text of the target tweet will be truncated if the text is too long if using the normal model. And there are too many unnecessary columns for analysis if using extend mode raw data, and the columns set are different between three different kinds of tweets, which will lead to a crash when uploading to Bigquery of GCP(Google Cloud Platform). If a tweet is

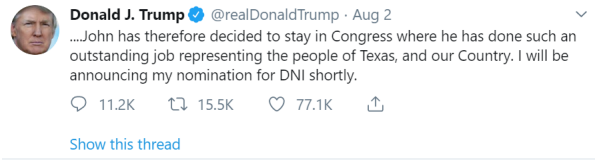

Fig. 3. One normal tweet

Fig. 4. Retweet without a comment

Fig. 5. Retweet with a comment

Fig. 6. Part of the follower list of @realdonaldtrump. Each row contains the name, screen name and a short self introduction (can be null)

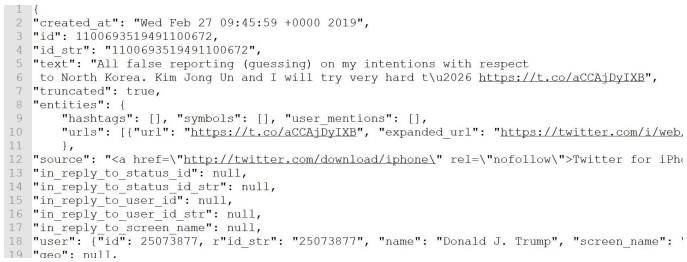

Fig. 7. Part of the non-extended tweet from @realdonaldtrump, the text is truncated and there are many redundant columns such as user information and columns with "str" at the end of their names

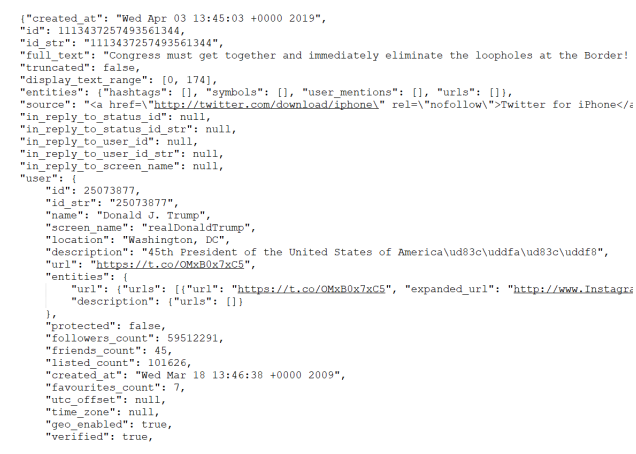

Fig. 8. Part of the extended tweet JSON data from @realdonaldtrump. Though it contains the full text even when the text is long, there are too many unnecessary columns that can waste storage space and even cause errors when uploading to Google server

a retweet, then its JSON data contains an element named 'retweet_status', and if a tweet is a quoted tweet, then its JSON data contains an element named 'quoted_status'. If the tweet contains any original media including images and videos, then its JSON data contains an element for detailed information about the media besides a link. All the elements mentioned above will not appear in a tweet JSON data that is not a retweet, or a quoted tweet or contains text only. Moreover, it is also quite often that one element of the previous tweet is null while the next tweet contains a dictionary in the same element. The problems mentioned above forestalls the auto-detection JSON input onto GCP, forcing us to develop our schema for uploading. Fig.7 is part of the JSON data sample got using normal mode Twitter function. Fig.8 is part of the JSON data sample got using extended mode Twitter function. Even the unextended Twitter data has too many columns that we would never use. Therefore, we built the schema table as the report mentioned above.

B. Build up Target account list

We explored the political panel of FOX NEWS and CNN NEWS. Then we finished the list containing the screen names of the Twitter accounts the reporter and editors that have recently posted articles. Just as Amy said in her report, some

Fig. 9. Part of the Twitter screen names of target reporters or editors of FOX NEWS

editors or reporters do not have twitter handles put on their profile pages of the news websites, especially the editors or reporters from FOX NEWS, then we found their Twitter screen name manually. We searched the political panel from the latest news and recorded the name of the author. Fig.9 is part of our list.

We found that the authors of the articles are often the same several people, especially FOX NEWS. So it took long time to find 30 editors or reporters from each website. Considering the time effect of the data want to analyze, we did not use the employee list though the websites do have lists. Because we want the latest information, and it is possible that someone on the list does not publish articles recently and do not represent the most up-to-date bias of his(her) organization. If we put these inactive employees onto our list, the bias analysis will be misled if researchers want to find out the latest bias of the websites.

C. Scraping followers

When collecting the followers for target accounts, the code ran smoothly when the number of followers of one account is relatively small (less than about 100,000), while it often got stuck when the numbers of accounts' followers are enormous. It is because too many followers' data caused too many requests to the Twitter handle. Therefore, the request times soon reach rate limitation, and the program has to wait for 15 minutes before the request limitation was refreshed, but the request numbers can soon exceed the rate limitation again. Also, some accounts are not open to the public. When requesting these accounts, it took half hour until the connection failed itself and started to get data for the next account, though it does not affect the further analysis because those private accounts have no effect to the public or the whole bias of news websites, the refuse of visiting costs lots of time. What was worse, the connection was not stable, and can also fail no reason. In this way, the code has low efficiency, and it spent around 5 hours for getting data for all the followers of 60 accounts on the list.

However, the program never got stuck when scraping tweet data. We assume that the number of requests increases with the number of accounts that are aimed to collect information from. Therefore, though more columns or elements are asked when collecting tweet data, the request rate was still low, while the request rate got high when getting the followers since large quantities of accounts was collected.

D. Google Cloud Debugging

Considering that it is more convenient and safer to store or deal with data online, and also considering the limited computing power of a personal computer, we built a server on Google Cloud Platform for data storage and further study.

In the beginning, we uploaded the raw JSON data from Twitter handle onto the Bigquery manually. Then, as mentioned at the Twitter Data part, we found that a large part of the information is not useful, and the upload process can crush due to the different structure of extended tweet data. Therefore, we decided the schema and extract the columns we need from raw Twitter data and upload it manually to the GCP until there was no error.

Since the personal computer cannot keep running all the time, we tried to leave the task for GCP for us to run the code regularly instead of click button manually. In this way, our data can keep on increasing all the time and is always the latest. Besides using barely python, We also wrote a bash code to do the scheduled downloading and uploading with Google Cloud Platform Computer Engine. We created an instance in Compute engine to store and run the python, after collecting the data from Twitter handle, it can either input data into Bigquery or output JSON file into the bucket named 'Twitter' in Storage tool. Using Bigquery tool, we can handle the data using SQL command and can also output the table's JSON file into the Storage tool to for daily checking or backup.

VI. Discussion

When finding the targeted accounts at news websites, We found that some people have more than one Twitter account. So extra time was spent to distinguishing accounts between a causal one and a formal one which involves more about his(her) news work. We took the formal one into the list because the followers of the formal account of the editor or reporter are supposed to be more political-bias-oriented compared to that of the casual account.

After getting the tweets of the target accounts, we found the contents of tweets quite varies. There are not only political news broadcasting but also some personal life update such as the sports match and restaurants. Some accounts prefer making original tweets to express to share personal life or express their political understanding, while some accounts make more retweet or quote compared to the original contents. Some accounts share the whole story of an event often while some prefer not to use too many words and sometimes emotional. All these patterns are worth digging.

We also found a vast difference between the data collected from reporters and editors from FOX NEWS and that of CNN NEWS. Though we got the same number of accounts of employees from both websites, the size of tweets data from accounts of employees of CNN NEWS is almost is 50% more compared to that from accounts of employees of FOX NEWS. It shows that employees from CNN NEWS put more energy into their Twitter accounts.

There is also a big difference between the followers' data from CNN NEWS and FOX NEWS. Using the same number of accounts of employees from both websites, the size of followers JSON data of CNN NEWS is also 50% larger compared that of FOX NEWS.

Our work mostly focused on data collecting and formalizing, providing a fundamental understanding of the Twitter data structure. We began to observe the data since we got the first tweet's raw JSON data. To start with, we only have a vague concept about the information we need, including some essential elements such as the tweet id, timestamp, followers' id. During our implementation of the scraping code, we gradually figured out all the elements we need through the thoroughly observing all types of structure of tweet JSON data. Even though the Bigquery on Google Cloud Platform is robust to the entities with different elements, we insist on making the data have a unique format because fewer errors can occur with the same data format when uploading. Also, through the implementation of the schema for the tweets and the followers table we mentioned above, more precise analysis strategies can be applied to those data with the same structure. Also, with the building up of the basic framework on the Google Cloud Platform, more advanced work can be applied without too much preparation work.

During the process of uploading, GCP runs smoothly without errors when dealing with small data piece while became quite slow when the size of JSON files or a specific element, such as followers, are huge. We think that the tool AWS from Amazon is also worth trying.

VII. Future work

Our code can download the newest information from major social media websites , upload them onto the Google Cloud Platform and modify their into a format which can be extracted and managed easily.

It can be further used to distinguish a tweet or news. Assuming that all the tweets are unbiased, we can build a model with the text of news as input and output of a binary classifier representing whether the input information is fake or not. Since we choose supervised learning, we first mark all the texts we collected with label "real" and "fake". To prepare features from the text, Stopwords library should be used to delete the frequent but meaningless words in both real and fake news. We can also extract Stem from the text and thus reduce and concentrate the Semantic features. After all the preparation, we can use the bag of words model and tf-idf model on the dataset and use 5-fold cross-validation to measure the performance of the model.

Meanwhile, the communicating property of Twitter should also be put into consideration. The distribution network should also be studied. Because there exist lots of Twitter

bots that can spread information pretty quickly and the distribution network they formed would be quite different compared to that created by individual accounts. We assume that the fake or highly biased information are more likely to use these bots to increase their influence on the Internet and, thus, cause echo chamber effect as mentioned in the previous work part and their distribution network can be identical. Besides, some biased tweets can use extreme words that can affect the retweet counts and the distribution network.

Then, after observing the quite different types of Twitter accounts, more research should be done about the effect of different types of tweets that can have on the viewers. For example, the original tweets may leave a more profound impression on the viewers compared to retweets. Also, the tones of texts and the sexual difference can play a role in the spreading of their tweets. Besides, when analyzing the bias of an organization, we should also figure out to what extent are the number of followers interact with the social opinions, i.e., does having more followers leads to more significant influence to people's opinion? Does the relationship linear or not? Does the composition of followers matters? The factors mentioned above may not affect personal bias analysis, but they can make a difference in an organization bias.

In this way, we can make a plugin for browsers. Unlike some previous work that shows the results on certain websites which require people to visit to check the validity or bias of information, this plugin can inform the users as soon as possible. When people want to read the news or search on Twitter, they can turn on the plugin, and the plugin would keep on real-time analysis of the texts the users are reading. We want to make the plugin analyze not only the related Twitter or news websites but also related source links. First of all, the plugin will check the bias or validity of this article and find other articles that talk about the same events but from different news websites or different perspectives, ensuring the users have a full understanding of the events with less bias. If the target article contains its authors with Twitter handles or news website links, the plugin will analyze the recent bias of authors using. Also, if the article contains media, the plugin will check the media on the Internet to make sure that the figures provide the same information as that in the article. Finally, the results, including the articles from different perspectives, the bias output and the validity of media will show in a widget on the screen as soon as the plugin finishes analyzing.

VIII. CONCLUSIONS

We managed to scrape data from Twitter and store them in JSON files following our schema. We also built a server on Google Cloud Platform for further analyzing and suggested some possibilities for our data. This project let us know how flexible research can be. In the beginning, we solely aimed at combining Twitter and news websites to analyze the bias of people and organization but did not think about the analyze method or ground truth or how to show the results. We have no idea about where our project would finally go, and there is an ample space for us to explore, which also made us realize the chance and the difficulty of such a big space when doing research. If scientists have their ideas and have specific plans, then the ample exploration space would be a great help for them. On the other hand, if scientists do not hold a firm direction, then as they keep on exploring the work others have done, they can lose directions got overwhelmed by large quantities of the information.

REFERENCES

[1] Bruns, Axel: Tweeting To Save The Furniture: The 2013 Australian Election Campaign On Twitter. Media International Australia, vol 162, no. 1, 2016, pp. 49-64. SAGE Publications, doi:10.1177/1329878x16669001.

[2] Hardalov M., Koychev I., Nakov P.: In Search of Credible News. In: Dichev C., Agre G. (eds) Artificial Intelligence: Methodology, Systems, and Applications. AIMSA 2016. Lecture Notes in Computer Science, vol 9883, pp.172-180, Springer, Cham (2016).

[3] Popat K., Mukherjee S., Strtgen J., Weikum G.: Credibility Assessment of Textual Claims on the Web. Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, pp. 2173-2178, Indianapolis, Indiana, USA (2016).

[4] An, J., Cha, M., Gummadi, K.P., Crowcroft, J., Quercia, D.Visualizing Media Bias through Twitter. 2012. AAAI Technical Report WS-12-01, The Potential of Social Media Tools and Data for Journalists in the News Media Industry. pp. 2-5

[5] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2018. FakeNewsNet: A Data Repository with News Content, Social Context and Dynamic Information for Studying Fake News on Social Media. arXiv preprint arXiv:1809.01286 (2018).

[6] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, Fake News Detection on Social Media, ACM SIGKDD Explorations Newsletter, vol. 19, no. 1, pp. 2236, Jan. 2017.

[7] https://docs.python.org/2/library/json.html

[8] https://docs.python.org/3/library/argparse.html